Efecto de la distribución de la duración de las llamadas sobre los algoritmos de encaminamiento.

A. Ariza Quintana, E. Casilari Pérez y F. Sandoval
Departamento de Tecnología Electrónica. E.T.S.I. Telecomunicación, Universidad de Málaga.
Campus de Teatinos S/N. 29071 Málaga
Teléfono: 952 132 728 Fax: 95213 14 47
E-mail: alfonso@dte.uma.es

Abstract *Studies on routing have traditionally supposed the call holding time to be exponentially distributed. In this work it is studied the effects of contemplating a call holding time different from this classical exponential function. In particular, we analyse the influence of call holding time on the performance of QoS (Quality of Service) routing algorithms. The experiments are performed in a realistic environment under different conditions of traffic load. The results prove that the call holding time just slightly impacts on routing efficiency. These divergences from the exponential case are even smoothed if it is considered the actual impossibility of an immediate broadcast of changes in the resources availability of the links.*

1 Introducción

Tradicionalmente, el modelo de tráfico empleado para el estudio y simulación de los algoritmos y protocolos de encaminamiento determina que la función de distribución que modela las llamadas entrantes en la red sigue un modelo de *Poisson* y que la duración de las llamadas viene modelada mediante una distribución de tipo exponencial. Sin embargo, estudios recientes [3][4][5] rebaten éste modelo y determinan que la distribución que modela la duración de las llamadas debería seguir un modelo distinto de la distribución exponencial.

Cabe pensar que este hecho podría afectar en gran medida al rendimiento de las redes de comunicaciones puesto que gran parte de los algoritmos de encaminamiento, y prácticamente todos los algoritmos de dimensionamiento empleados en la actualidad, han sido diseñados y optimizados bajo la suposición de que la duración de las llamadas sigue una distribución estrictamente exponencial, entre otras razones, por tratabilidad analítica. Dada esta circunstancia, surge la necesidad de estudiar la dependencia del rendimiento final de la red con respecto al hecho de que la duración de las conexiones no siga el modelo exponencial.

En este trabajo se ha realizado un estudio del rendimiento de una red de comunicaciones orientada a conexión bajo distintas distribuciones de probabilidad para modelar la duración de las llamadas. Este estudio se ha realizado mediante simulación empleando como banco de pruebas una red de area extensa, de una complejidad tal que imposibilita el estudio analítico. De hecho, los modelos analíticos existentes en la actualidad sólo son válidos para redes sencillas formadas por un número muy pequeño de nodos.

2 Modelo de red y entorno de simulación

2.1 Modelo de red

Consideremos una red formada por un conjunto V de nodos unidos entre sí mediante un conjunto E de enlaces. Cada uno de estos enlaces $e_{ij} \in E$ es identificado por los nodos (i y j), que conecta y dispone de una capacidad de C_{ij} unidades de ancho de banda. Las llamadas entrantes pueden solicitar una conexión entre cualquier par de nodos origen-destino y el camino escogido para establecer la conexión entre este par de nodos es seleccionado de forma dinámica en función del estado de la red, de manera que el camino seleccionado para realizar la conexión entre un par de nodos evoluciona conforme cambia el estado de la red.

Cuando una llamada llega a la red, ésta precisa una serie de recursos para poder establecer una comunicación con éxito. En el caso de que la red disponga un camino con suficientes recursos para soportar la llamada, se procede a establecer la conexión reservando los recursos necesarios, en caso contrario la llamada es rechazada. En el modelo bajo estudio no se permite reencaminamiento ni renegociación de llamadas, es decir, la llamada utiliza durante toda su duración el mismo camino y los mismos recursos que el algoritmo de encaminamiento reservó para su establecimiento inicial.

Por simplicidad, consideraremos que el único recurso a reservar y a tener en cuenta es el ancho de banda y, que el ancho de banda de solicitado por las llamadas viene dado por una distribución uniforme de valor medio b unidades. No se tendrán en cuenta otros posibles requisitos de calidad de servicio, como puede ser el retardo para decidir si se acepta o rechaza una llamada. Así pues, en

cada enlace e_{ij} es posible transmitir $\frac{C_{ij}}{b}$ conexiones de media.

Bajo estas consideraciones es posible modelar una red de comunicaciones como un sistema de colas conectadas unas con otras, Fig. 1(a) y, puesto que las llamadas llegan a la red siguiendo una distribución de *Poisson*, y suponemos que la tasa de llegada es la misma para todos los nodos y se escogen los destinos con la misma probabilidad, es posible modelar el comportamiento de cada enlace como una cola $M/G/N/N$ [6] donde los N servidores representan las N porciones de capacidad del enlace y es igual a $\frac{C_{ij}}{b}$.

Así pues, cualquier red de comunicaciones orientada a conexión no es más que un conjunto de colas conectadas entre sí, por lo que todo camino que conecta un par de nodos está formado por una sucesión de colas donde cada conexión compite por una serie de recursos en cada una de las colas que atraviesa.

La dificultad del modelo radica en que parte de las conexiones servidas por las colas son finales (es decir, es el enlace final de la conexión) y parte son derivadas a otros enlaces con lo que aparece un sistema de colas en tándem, Fig. 1(b). Para complicar el modelo, el número de colas en tándem es variable y depende del camino. Además, una cola puede ser simultáneamente cola final, cola inicial e intermedia al formar parte de distintos caminos.

Actualmente no existen modelos analíticos capaces de modelar el comportamiento de un sistema de esta complejidad, y se espera que tarde todavía varios años en aparecer un modelo viable. Por esta razón, este estudio se ha basado en la simulación del rendimiento de un algoritmo de encaminamiento cuando el tráfico ofrecido a la red sigue distintas funciones de distribución para modelar la duración de las llamadas.

2.2 Entorno de simulación

Este estudio se ha realizado analizando, mediante simulaciones de una red de comunicaciones, el rendimiento final del algoritmo de encaminamiento. La única diferencia entre las distintas simulaciones ha sido la función de distribución de probabilidad utilizada para modelar la duración de las llamadas, siempre considerando que la duración media de las llamadas es la misma. Para medir el rendimiento del algoritmo de encaminamiento se ha utilizado la probabilidad de perdida de ancho de banda [7], determinada por la siguiente expresión:

$$P_p = \frac{BW_l}{BW_o} \tag{1}$$

donde BW_l (*ancho de banda perdido*) viene dado por el producto entre el número de llamadas rechazadas por la red y el ancho de banda que estas llamadas precisaban para establecer la conexión; mientras que BW_o (*ancho de banda ofrecido*) es el producto de todas las llamadas ofrecidas a la red por el ancho de banda que estas exigen para su conexión.

Para las pruebas se ha construido, con la herramienta de simulación de eventos discretos OPNET, un entorno de simulación orientado a llamadas. Así los eventos que tienen lugar en el sistema son la conexión y desconexión de llamadas, la reserva de recursos y la actualización del estado de los enlaces. La red empleada en las pruebas es una versión modificada del proveedor americano de internet MCI (Fig. 2) [1][7], empleada en diversos estudios de simulación. En esta versión se han limitado todos los enlaces a una capacidad de 50 Mbits/s.

En las pruebas también se ha tenido en cuenta la posibilidad de que los datos con los que se realiza la toma de decisión estén obsoletos en el tiempo (caso de encaminamiento "Impreciso"). Esta circunstancia es debida a la imposibilidad real de disponer de información completamente actualizada del estado de la red, circunstancia provocada, entre otros factores, por la necesidad de limitar el ancho de banda consumido por los mensajes de actualización. Para este estudio, la estrategia de actualización que se ha escogido ha sido la actualización por umbral proporcional. Se ha adoptado esta estrategia por ser la que mejor rendimiento presenta para un mismo ancho de banda consumido por los paquetes de información del estado de la red [2]. De acuerdo con esta estrategia, se actualizará el estado del enlace cuando se cumpla la siguiente condición:

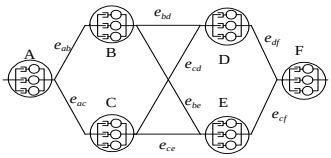
$$\frac{|B_{old} - B_{new}|}{B_{old}} > th \tag{2}$$

donde B_{old} es el ancho de banda notificado en el último mensaje de actualización, B_{new} el ancho de banda actualmente existente en el enlace y th es una constante que determina cuándo se debe generar un nuevo mensaje de actualización de estado.

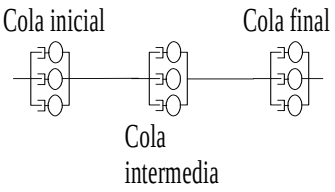
El algoritmo de encaminamiento utilizado en las pruebas asigna un coste, proporcional al uso del enlace e_{ij}, dado por la expresión:

$$Coste_{ij} = \frac{c_{ij}}{C_{ij}} \tag{3}$$

donde c_{ij} es el ancho de banda ocupado y C_{ij} es la capacidad total del enlace e_{ij}. El coste total del camino vendrá dado por la suma de los costes de los enlaces que componen dicho camino. Antes de proceder desde el nodo origen a la búsqueda de un camino válido se han eliminado de la red todos los enlaces que, en función de la información disponible por el nodo, no poseen de suficiente ancho de banda libre para satisfacer los requisitos solicitados por la conexión. Si no se encuentra un camino, la llamada es rechazada; en el caso de que el nodo detecte un camino (a partir del conocimiento que posee del estado de la red, que puede ser impreciso), se procede a comprobar, mediante los controles de admisión, si existen realmente los recursos requeridos en los enlaces que atraviesa el camino elegido. En el caso de que en alguno de los



(a) Esquema del modelo de colas de una red de comunicaciones



(b) Detalle del modelo de colas

Figura 1: Esquema del una red orientada a conexión y detalle de su sistema de colas

enlaces no exista suficiente ancho de banda disponible, la llamada es rechazada y se liberan todos lo recursos reservados por ésta en su proceso de establecimiento.

Para las simulaciones de las llamadas se han considerado dos posibles escenarios. En el primer escenario se ha supuesto la posibilidad de la existencia de una alta agregación de llamadas, es decir, el ancho de banda medio requerido por cada llamada es mucho menor que la capacidad del enlace. En el segundo escenario se ha supuesto la existencia de una baja agregación, para ello el ancho de banda que solicita cada llamada es grande con relación al ancho de banda del enlace. Por lo tanto, el número de llamadas que pueden ocupar un enlace de forma simultanea es bajo. En concreto, para el primer caso el ancho de banda que precisa cada conexión es escogido mediante una distribución uniforme entre 0,25 y 0,75 Mbits/s, mientras que para el segundo caso la distribución del ancho de banda requerido toma valores entre 16 y 20 Mbits/s.

2.3 Distribuciones para la duración de la llamada

La funciones de distribución que se han comparado en este trabajo han sido:

Exponencial: En este caso la función de densidad de probabilidad viene dada por

$$f_{exp}(x) = \begin{cases} \frac{1}{\mu} e^{-\frac{x}{\mu}} & si\ x \geq 0 \\ 0 & en\ otro\ caso \end{cases} \quad (4)$$

donde μ es la media de la distribución. Esta distribución se caracteriza porque su media y su desviación son iguales.

Pareto: En este caso la función de densidad de probabilidad viene dada por la siguiente expresión:

$$f_{par}(x) = \begin{cases} \frac{\alpha \beta^{\alpha}}{(\beta+x)^{\alpha+1}} & si\ x \geq 0 \\ 0 & en\ otro\ caso \end{cases} \quad (5)$$

Donde β está relacionada con la media μ mediante la expresión $\mu = \frac{\beta}{\alpha - 1}$ y α indica la intensidad de la caída de la función de densidad de probabilidad o, dicho de otro modo, el grado de subexponenciabilidad. Si la caída es muy lenta ($\alpha < 2$), la distribución tendrá una varianza infinita. En las pruebas consideramos dos casos de distribución de Pareto con $\alpha = 1,5$ y $\alpha = 1,8$.

Multimodal La duración de las llamadas solo puede tomar un número limitado y finito de valores. En este trabajo se han empleado dos distribuciones multimodales distintas con dos modas cada una (la distribución solo puede tomar dos valores) siendo las modas $1/K$ y K veces el valor medio, donde K es una constante que determina la separación de las dos modas. En las pruebas se han utilizado como valores de K 16 y 4 (multimodal 16 y 4). La probabilidad de generar una llamada de duración $\frac{1}{K} * \mu$ viene dada por la expresión:

$$p_1 = \frac{K - 1}{K - \frac{1}{K}} \quad (6)$$

mientras que la probabilidad de generar una llamada de duración $K * \mu$ viene dada por la siguiente expresión.

$$p_2 = \frac{1 - \frac{1}{K}}{K - \frac{1}{K}} \quad (7)$$

donde μ es el valor medio de la duración de las llamadas.

Determinista (Var 0): Todas las llamadas tienen exactamente la misma duración, y lógicamente coincidirá con el valor medio μ. Este tipo de distribución se caracteriza por tener varianza cero, con lo que, junto con la distribución de Pareto, constituyen los dos casos extremos.

En todos los casos se ha supuesto una duración media de llamada de 1200 segundos.

En la figura 3 se pueden observar las funciones de probabilidad correspondientes a cada una de las funciones de distribución de probabilidad utilizadas en las pruebas.

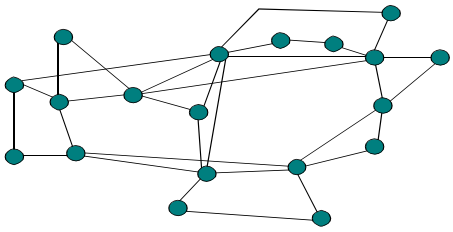


Figura 2: Red MCI utilizada en las pruebas

3 Resultados

En la figura 4 se pueden ver las probabilidades de pérdida normalizadas con respecto a las pérdidas de la función exponencial para los casos preciso e impreciso con alta agregación. Los distintos resultados obtenidos se han representado con un margen de confianza del 95% calculado mediante la distribución de T-Student.

El primer objetivo es estudiar el rendimiento en el caso ideal en el que las actualizaciones de los cambios son inmediatos y existe en los nodos un conocimiento preciso del estado de la red (caso "Preciso"). Esto lo hacemos tanto para una baja como para una alta agregación del tráfico en los enlaces. En las figuras 5, 7(a) y 4(a) se pueden ver los resultados en términos absolutos y relativos de este experimento. De estas figuras se pueden extraer una serie de observaciones. La primera y más importante es que, en principio, y al contrario de lo esperado y del comportamiento demostrado en otros sistemas de colas, la distribución empleada para modelar la duración no influye de forma determinante en el rendimiento del sistema. La segunda observación que se extrae es que el rendimiento empeora al disminuir la varianza de la distribución empleada, en este caso, la función de distribución con varianza cero se comporta como una cota superior, mientras que se puede observar que el comportamiento de las funciones subexponenciales (Pareto) mejora al disminuir su grado de subexponencialidad.

Sin embargo, uno de los resultados más interesantes aparece al comparar la figura 5(a) con la figura 5(b) y la figura 7(a) con la figura 4(a) . En estas figuras es posible observar que el rendimiento de las distintas funciones de distribución está mucho más cercano entre sí en el caso de existir una baja agregación de conexiones en los enlaces.

La respuesta que creemos que explica este comportamiento es la siguiente. Puesto que el comportamiento de los enlaces se puede modelar mediante un proceso de nacimiento y muerte de *Markov*, donde la probabilidad de estar en un estado i será P_i, la probabilidad de rechazar una llamada es igual al la probabilidad de que llegue una llamada cuando el enlace está en el estado P_N, donde N es el número máximo de conexiones que el enlace puede portar. Cuando se utilizan distribuciones de tipo subexponencial existirá una mayor probabilidad de llamadas con una enorme duración, aunque estas llamadas serán raras y, por lo tanto, la probabilidad de que existan simultáneamente varias conexiones activas en el mismo enlace es baja. Para compensar estas llamadas de gran duración existirá un gran número de conexiones de duración muy breve. Esta última circunstancia, sumada al hecho de que la tasa de llegadas es la misma en los dos casos, implica que si el valor de N es lo suficientemente grande se reduce la probabilidad de que el enlace llegue al estado P_N, ya que, para cuando llegue la siguiente llamada, la probabilidad de que hayan finalizado varias de las llamadas en curso es mayor. El hecho de que exista una conexión de larga duración afecta poco ya que ésta sólo toma un recurso de los N posibles. Por el contrario, cuando el valor de N es pequeño, Figuras 5(b) y 7(a), este efecto se diluye y la probabilidad de que el enlace se encuentre en el estado P_N cuando llegue la siguiente conexión tiende a igualarse para las distintas funciones de distribución de la duración de llamada.

En segundo lugar se estudia el comportamiento del sistema en condiciones de imprecisión. En las figuras 6, 7(b) y 4(b) se puede observar el rendimiento del sistema bajo estas circunstancias. En este caso, la tasa de imprecisión es de un 80%, es decir, $th = 0,8$. La necesidad de realizar estos experimentos en condiciones de imprecisión se ve justificada por el hecho de que estas condiciones de imprecisión forman parte del funcionamiento normal de cualquier red de comunicaciones. Al estudiar los resultados obtenidos se puede observar que en los resultados con imprecisión disminuye la importancia de emplear una determinada distribución para modelar la duración media de las llamadas. Sin embargo, todavía se mantiene la tendencia de un menor rendimiento cuanto menor es la varianza, de forma que el caso en que todas las llamadas tienen igual duración el rendimiento final es menor.

Por último se comparan los resultados de los casos preciso e impreciso (figuras 5 y 6) y se puede observar que, para el caso de alta agregación, mejora el rendimiento en presencia de imprecisión conforme aumenta el tráfico, presentando para tasas altas de tráfico un mejor rendimiento cuanto mayor es el umbral de actualización. Este fenóme-

no se puede observar con más claridad en la figura 8 y es fácilmente explicable por las oscilaciones del tráfico ofrecido a la red. Las rápidas oscilaciones del tráfico entre los distintos enlaces provocan inestabilidades. Al introducir un elevado grado de imprecisión en el sistema, el estado de los enlaces no se actualiza con cada nueva conexión ofrecida o finalizada. Este hecho provoca un menor balanceo en el tráfico, tendiéndose a utilizar caminos que requieren menor número de recursos (caminos más cortos) en detrimento de caminos de mayor longitud que en ese instante tienen un menor coste de establecimiento. A la larga, el menor número de recursos consumidos permitirá establecer una mayor número de futuras conexiones. Por el contrario, para bajas tasas de tráfico es más interesante redistribuir el tráfico a fin de evitar los posibles "puntos calientes" que existan en la red (para altas cargas prácticamente todos los enlaces son puntos calientes). Sin embargo, para que este fenómeno sea apreciable es necesario que exista la posibilidad de agregar múltiples llamadas en un enlace. En caso contrario, el comportamiento total se degrada ya que en circunstancias de baja agregación el peso de cada llamada es muy importante en el estado de la red. Un bajo conocimiento del estado de red dará lugar, en este caso, a pérdidas importantes con respecto al conocimiento preciso, siendo peor el resultado cuanto mayor sea el tráfico ofrecido.

4 Conclusiones

De este trabajo se pueden extraer varias conclusiones. La primera y más importante es que la función que modela la duración de las llamadas es poco significativa para el rendimiento de la red desde el punto vista de los algoritmos de encaminamiento y de dimensionado aunque es posible que esta función pueda tener una enorme importancia en el rendimiento de otros controles de la red. Teniendo en cuenta esta circunstancia, es perfectamente razonable emplear, por tratabilidad analítica, bien una función exponencial o una función determinista para modelar la duración de las llamadas, con la seguridad de que el resultado con el tráfico real será similar o incluso mejor que el esperado (por ejemplo, cuando la duración de las llamadas sigue una distribución de *Pareto* los resultados finales suelen ser mejores que los obtenidos con la distribución exponencial). De todas maneras esta parte precisa de un estudio mucho más detallado a fin de poder extraer una interpretación de los datos concluyente.

La segunda conclusión que extraemos de este trabajo es que, en condiciones de alta carga, es interesante introducir un cierto grado de imprecisión en el sistema a fin de limitar el balanceo del tráfico en la red y conservar recursos o, en su defecto, mecanismos diversos que limiten el balanceo del tráfico a través de la red. Este mecanismo puede ser decrementar la tasa de actualización conforme aumenta la ocupación de la red. Con esto se conseguirían dos efectos beneficiosos, el primero, mejorar el rendimiento ante las distintas cargas y, el segundo que el ancho de banda consumido por la señalización sea constante con independencia del tráfico ofrecido a la red.

Agradecimientos

Este trabajo ha sido parcialmente financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT) mediante el proyecto número TEL99-0755 y el proyecto FEDER-CICYT número IFD97-0918.

Referencias

[1] G. Apostolopoulos, R. Guerin, S. Kamat, and S. Tripathi, *Improving qos routing performance under inaccurate link state information*, Proc. of the 16th International Teletraffic Congress (ITC'16), North Holland Elservier Science Publisher, Junio 1999, pp. 7–11.

[2] A. Ariza, E. Casilari, and F. Sandoval, *Strategies for updating link states in qos routers*, Electronics Letters **36** (2000), no. 20, 1749–1750.

[3] F. Barceló and J. Jordán, *Channel holding time distribution in public cellular telephony*, Proc. of the 16th International Teletraffic Congress (ITC'16), North Holland Elservier Science Publisher, Junio 1999, pp. 107–116.

[4] V. A. Bolotin, *Modeling call holding time distributions for ccs network design and performance analysis*, IEEE JSAC **12** (1994), no. 3, 433–438.

[5] R. G. Garroppo, S. Giordano, and A. Vaccaro, *A teletraffic analysis of dial-up connections over pstn*, GLOBECOM'98 (Sydney), Noviembre 1998.

[6] K. R. Krishnan and T. J. Ott, *Forward-looking routing: A new state-dependent routing scheme*, Proc. of the 12th International Teletraffic Congress (ITC'12), North Holland Elservier Science Publisher, Junio 1988.

[7] Q. Ma, *Quality-of-service Routing in integrated Service Networks*, Ph.D. thesis, Carnegie Mellon University, Pittsburgh, January 1998.

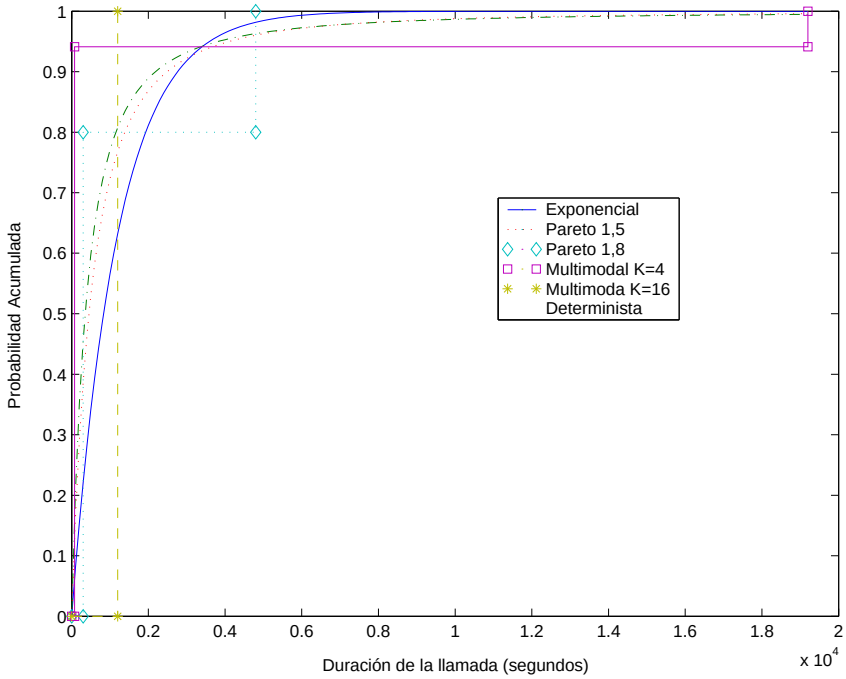
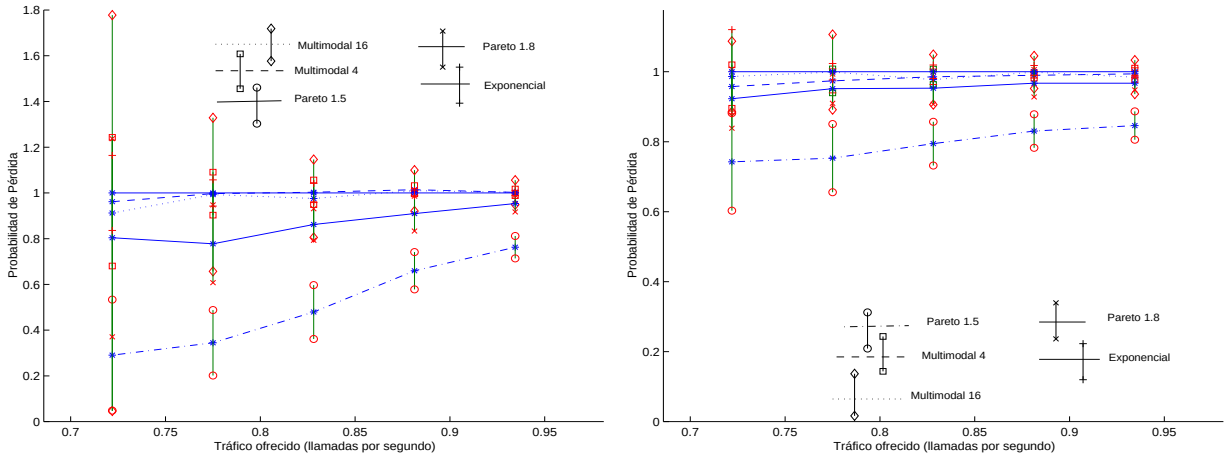


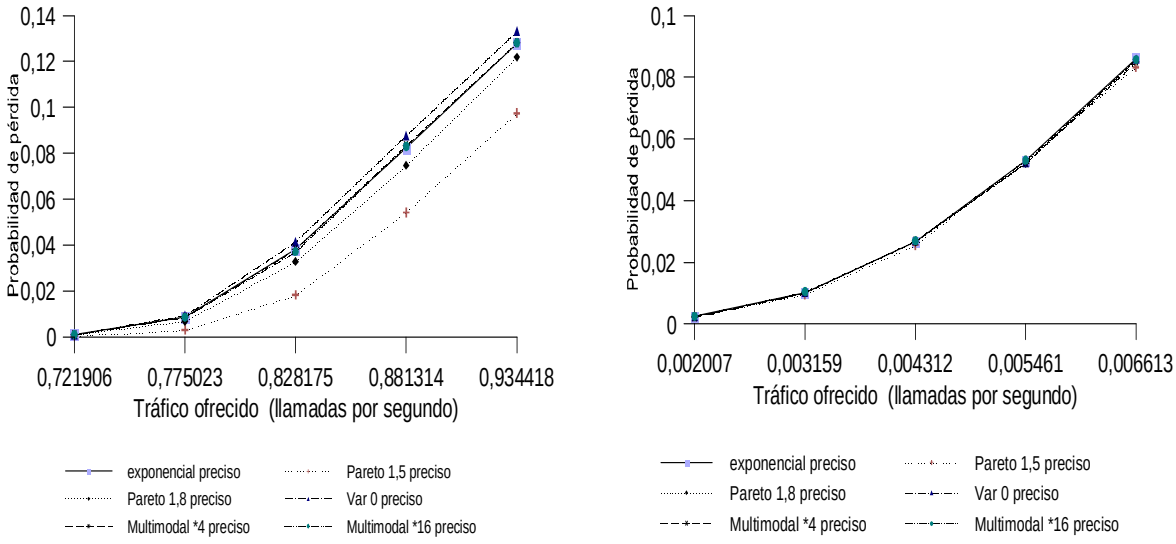
Figura 3: Funciones CDF de las distintas funciones de probabilidad utilizadas en las pruebas



(a) Caso preciso

(b) Caso impreciso

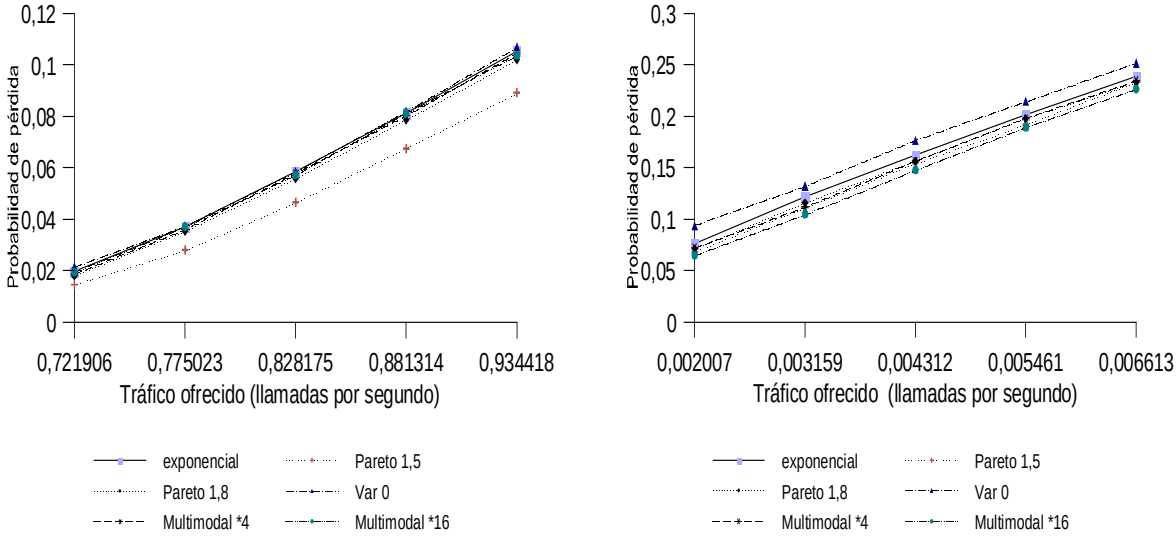
Figura 4: Comparación de las probabilidades de pérdida (así como sus márgenes de confianza) para los casos preciso e impreciso con alta agregación, los resultados se han normalizado por los que ofrece la distribución exponencial



(a) Alta agregación

(b) Baja agregación

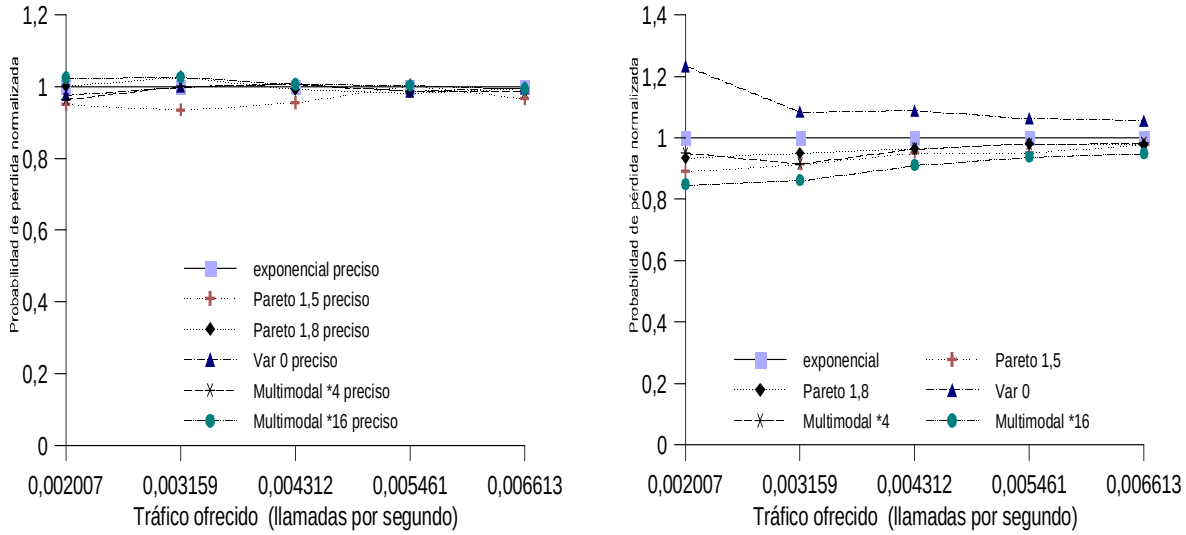
Figura 5: Comparación de las probabilidades de pérdida con un conocimiento preciso del estado de la red



(a) Alta agregación

(b) Baja agregación

Figura 6: Comparación de las probabilidades de pérdida con un conocimiento impreciso del estado de la red, tasa de actualización del 80%



(a) Caso preciso

(b) Caso impreciso

Figura 7: Comparación de las probabilidades de pérdida para los casos preciso e impreciso con baja agregación, los resultados se han normalizado por los que ofrece la distribución exponencial

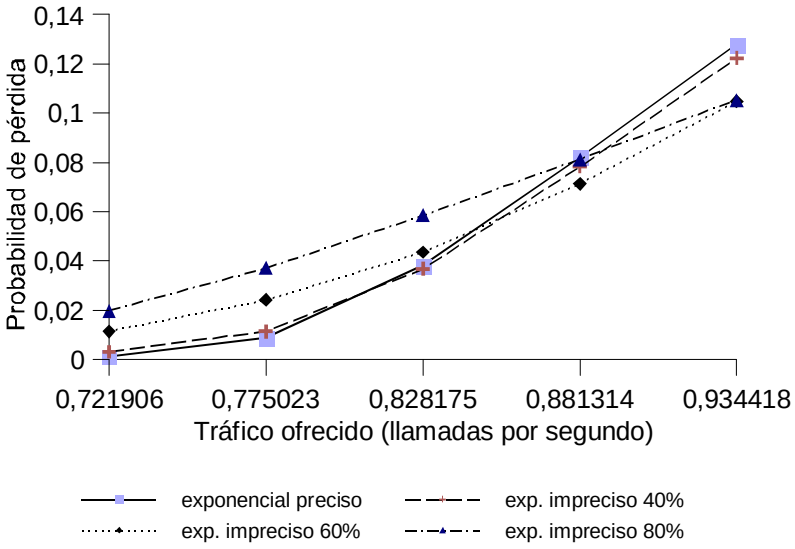


Figura 8: Evolución de la probabilidad de pérdida de ancho de banda para varios porcentajes de disparo de actualización

Algoritmo simple y eficiente de renegociación CBR a dos niveles para servicios de vídeo almacenado VBR en redes con QoS.

Josep R. Pegueroles, Juan José Alins, Jordi Mata, Luis J. de la Cruz.
Departamento de Ingeniería Telemática. Universitat Politècnica de Catalunya.
Jordi Girona 1 y 3. Campus Nord, Mód C3, UPC. 08034 Barcelona
Teléfono: 934 016 014 Fax: 934 015 981
E-mail: ljcruz@mat.upc.es / josep@mat.upc.es

Abstract. *Variable-bit-rate (VBR) compressed video can exhibit significant multiple-time-scale bit rate variability. Smoothing techniques remove the periodic fluctuations generated by the codification modes. However, global efficiency concerning network resource allocation remains low due to scene-time-scale variability. RCBR services seems to be a suitable approach in order to reach higher efficiency. In this paper, a new two level renegociated constant bit rate (RCBR) technique for stored video is presented. The technique uses the client buffer space to assure continuity-time constraint and to reduce the higher bandwidth required periods. Simulation results over 4 relevant sequences are presented.*

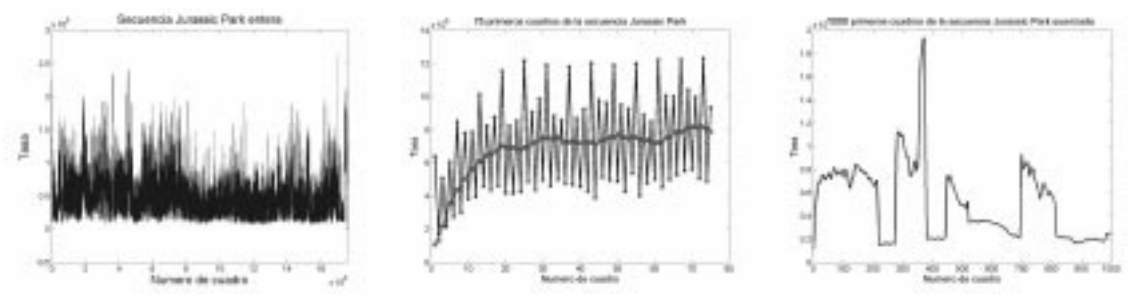
1 Introducción.

Es ampliamente conocido que los algoritmos de compresión de vídeo más utilizados actualmente, como la familia MPEG, generan tráfico de naturaleza a ráfagas. Estas fluctuaciones en la tasa a la salida del codificador son provocadas tanto por los distintos modos de codificación empleados por el algoritmo en los cuadros de la secuencia de vídeo como por la variación en la complejidad de las imágenes a codificar. La primera de las causas nos lleva a fluctuaciones periódicas de escala temporal reducida, del orden de milisegundos (Fig. 1.b), mientras que la segunda provoca variaciones no periódicas de escala temporal mayor, del orden de segundos (Fig. 1.c).

Numerosos trabajos han propuesto las técnicas de suavizado para solventar los problemas de las fluctuaciones periódicas a nivel temporal de cuadro. De cualquier forma, la mejora en eficiencia de ancho de banda debida al conformado de tráfico a nivel de cuadro sólo se debe a la disminución del rizado propio de la codificación en los intervalos de mayor actividad. La garantía de calidad de servicio obliga a contratar con la red los recursos necesarios para una correcta transmisión de los periodos de tasa máxima produciéndose un desperdicio de recursos contratados en los intervalos de menor actividad. La asignación dinámica de recursos de red se ha apuntado como solución a ese uso no eficiente de los recursos contratados.

En este artículo se propone y analiza un nuevo algoritmo de entrega de vídeo almacenado VBR a la red mediante intervalos a tasa constante de dos posibles niveles. Dicho algoritmo aprovecha el conocimiento previo de la secuencia de vídeo para calcular los instantes de renegociación o conmutación de tasa y el buffer del cliente para realizar técnicas de *work-ahead* buffering con el fin de disminuir al mínimo los intervalos de transmisión a tasas elevadas.

La bondad del algoritmo propuesto se ha comprobado mediante simulaciones sobre distintos patrones de vídeo MPEG-I de diferente longitud y sistema.



(a) Naturaleza a ráfagas de la Secuencia Jurassic Park entera.

(b) Rafagueo debido a la propia codificación

(c) Rafagueo debido a cambios de escena.

Figura 1: Distintas escalas temporales de la misma secuencia de vídeo Jurassic Park.

De las simulaciones realizadas se observó el buen comportamiento del algoritmo propuesto tanto en lo que se refiere a cumplimiento de las restricciones temporales como a número de renegociaciones necesarias. Los tamaños de buffer en el cliente para dichas realizaciones también fueron calculados.

El resto del artículo está organizado como sigue. En la siguiente sección se presenta la asignación dinámica de recursos como solución para incrementar la eficiencia de los recursos contratados a la red. A modo de ejemplo, se presentan dos técnicas de renegociación a tasa binaria constante presentadas en la literatura. En el apartado 3 se detalla el mecanismo de renegociación 2-RCBR, se introduce el modelo matemático utilizado para el estudio de la secuencia de vídeo y se formalizan las restricciones temporales a que debe ajustarse ésta para tener continuidad temporal, no saturación del buffer de recepción, retardo nulo y eficiencia máxima. En este mismo apartado se justifica la elección del número de niveles de contrato, su valor y el cálculo de los puntos de renegociación. También se presenta un algoritmo práctico de cálculo de los instantes de renegociación de tasa. En el apartado 4 se muestran los resultados obtenidos de la aplicación del algoritmo sobre distintas secuencias de vídeo. Finalmente, en el último apartado, se exponen las conclusiones y aportaciones derivadas de este trabajo.

2 Asignación dinámica de recursos

En servicios VBR, la solución más ampliamente adoptada para mantener la calidad de servicio requerida por el cliente es contratar con la red una disponibilidad de recursos que permita la correcta transmisión de los periodos de más elevada tasa. Esta solución conlleva una disminución considerable de la eficiencia de la conexión ya que, durante los periodos de menor actividad, se están reservando unos recursos que en realidad no son necesarios.

Diversos estudios han propuesto los servicios con asignación dinámica de recursos como solución para incrementar la eficiencia total en la transmisión de una secuencia de vídeo. De esta forma, en los momentos de baja actividad será posible llevar a cabo una liberación de recursos que podrán ser empleados por otras conexiones mientras que cuando la actividad sea más alta, se solicitarán mayores recursos con objeto de mantener la calidad de servicio deseada.

2.1 Renegociación a tasa binaria constante.

Del mismo modo que para servicios con asignación estática de recursos existe la posibilidad entre solicitar conexiones a tasa constante o variable (CBR o VBR), se puede llevar a cabo una distinción entre los servicios con asignación dinámica. Así, la renegociación podría implementarse sobre conexiones a tasa constante o a tasa variable.

De entre todos los mecanismos de asignación dinámica de recursos, el RCBR, o renegociación de conexiones a tasa constante, es el más sencillo que puede implementarse. Mediante RCBR, una fuente puede renegociar su tasa de servicio mediante el envío de un mensaje de señalización en el que se solicita el incremento o el decremento de la tasa actual. Si la renegociación es admitida por la red, la fuente puede enviar datos a una nueva tasa constante. En caso contrario ésta deberá adaptarse a la tasa que tiene disponible o la información excedente se perderá produciéndose una degradación de la calidad de servicio.

Los dos parámetros más importantes a la hora de realizar un servicio RCBR son los instantes de renegociación y los niveles de tasa solicitada. Las políticas seguidas para determinar dichos parámetros dependen en gran medida del tipo de servicio que se quiera prestar. Si lo que se quiere es transmitir una secuencia de vídeo previamente almacenada, se puede calcular un programa de renegociaciones óptimo a priori. Si por contra lo que se quiere es trabajar con codificación en tiempo real se deberá hallar mecanismos heurísticos, que permitan decidir el nivel de contrato con la red únicamente con la información disponible en tiempo real.

A continuación se presentan las soluciones 3-RCBR y Work-ahead correspondientes a cada uno de los casos citados anteriormente y se discuten sus ventajas e inconvenientes.

2.1.1 3-RCBR

En [1] se propone dividir la secuencia total de vídeo en 3 distintos niveles de actividad, dependiendo en cual de dichos niveles nos encontremos solicitaremos una tasa u otra. Dicha división en 3 niveles se realizó a partir de la caracterización de una secuencia de vídeo mediante un modelo de fluidos modulado por Markov. Para evitar renegociaciones excesivas se calcularon 4 umbrales para los 3 niveles, 2 de subida y 2 de bajada, que definían 2 ciclos de histéresis.

El sistema 3-RCBR está especialmente concebido para la transmisión de vídeo codificado en tiempo real ya que sólo se requiere del conocimiento de la tasa demandada instantáneamente por la fuente para resolver en cuál de los niveles de actividad nos encontramos y, por tanto, si se debe o no renegociar la tasa con la red. Además presenta un buen comportamiento en cuanto a número de renegociaciones se refiere.

De todos modos, para un perfecto ajuste de los umbrales y niveles de renegociación, se debe conocer la secuencia a priori o trabajar con valores aproximados. En cualquiera de los casos el precio que estamos pagando al no requerir el conocimiento previo de la secuencia se traduce en un decremento considerable de la eficiencia, del orden del 50%.

2.1.2 Optimal smoothing work_ahead algorithm.

En [2] se propone una técnica de suavizado óptima para un tamaño de buffer de cliente determinado. El sistema propuesto aprovecha el conocimiento del vídeo a transmitir para calcular la manera óptima de entregar la secuencia a la red mediante tramos a tasa constante. Se aprovecha el buffer del cliente para almacenar datos enviados "por adelantado", es decir, que aun no se necesitan para decodificar el vídeo, pero que se van a necesitar al cabo de un cierto intervalo de tiempo. Este tipo de técnicas recibe el nombre de Work-ahead buffering.

El algoritmo work-ahead optimal smoothing aunque es la forma más suave de entregar el tráfico a la red, y configurable para un tamaño de cliente dado, nos lleva a soluciones con excesivas renegociaciones de tasa y un número no determinado de niveles de renegociación.

3 Servicio 2-RCBR.

El presente trabajo pretende establecer un mecanismo simple y eficiente de entrega de tráfico de vídeo almacenado a la red mediante intervalos a tasa constante. En los servicios de vídeo bajo demanda las secuencias son conocidas a priori. Parece lógico aprovechar este hecho para realizar una entrega inteligente a la red. Por otro lado, se debe controlar que el estudio de la secuencia de vídeo no derive en una complejidad de cálculo excesiva. Las técnicas como la 3-RCBR fundamentan su éxito en la ausencia de cálculo de parámetros de entrega a la red y algoritmos de servicio muy simples. Dichos mecanismos, en cambio, nos llevan a eficiencias bajas en cuanto a aprovechamiento de ancho de banda. Parece razonable buscar un sistema que encuentre un compromiso entre eficiencia y simplicidad de cálculo. Este es el objetivo del mecanismo 2-RCBR.

3.1 Modelo matemático.

Tal como se define en [3] un servicio de vídeo bajo debe garantizar que no se produzca ni inanición ni saturación de buffers. Se debe procurar que en todo momento se tenga información suficiente para decodificar y a su vez se disponga de espacio de memoria suficiente para guardar la que no se decodificará de inmediato.

A continuación se presentará el modelo matemático que nos servirá para tratar estas condiciones de una manera formal.

Definamos la secuencia de vídeo como una función discreta en tiempo y amplitud *v(n)*, donde n indica el índice del cuadro codificado pudiendo tomar valores desde 1 hasta la longitud de la secuencia en número de cuadros *(N)*. Esta función tiene como imágenes el número de bits necesarios para codificar el cuadro de índice *n*.

Sea *V(n)* la función que representa el total de bits necesarios para codificar todos los cuadros desde 1 hasta n. Esta función se puede calcular a partir de *v(n)* tal como se expresa en (1).

$$V(n) = \sum_{i=1}^{n} v(i) \qquad (n = 1..N) \tag{1}$$

Por otra parte, definamos *c(n)* como el contrato en bps con la red para cada tiempo de cuadro y *C(n)* como el total contratado con la red desde el instante inicial hasta el tiempo de cuadro n. De igual forma que en (1) puede definirse *C(n)* como la función acumulativa de *c(n)* como indica la expresión (2).

$$C(n) = \sum_{i=1}^{n} c(i) \qquad (n = 1..N) \tag{2}$$

Finalmente definamos *e(n)* como el tráfico que realmente se entrega a la red en un determinado tiempo de cuadro n y que depende de la técnica de transmisión empleada. Similarmente a lo realizado con *v(n)* y *c(n)*, definiremos *E(n)* como el tráfico total entregado a la red desde el instante inicial hasta el tiempo de cuadro *n* , véase expresión (3).

$$E(n) = \sum_{i=1}^{n} e(i) \qquad (n = 1..N) \tag{3}$$

Si no se utilizan técnicas de buffering, el tráfico entregado a la red es VBR puro. Este hecho se puede expresar tal como se indica en (4).

$$e(n) = v(n) \tag{4}$$

En este caso, las restricciones de no violación del contrato y no inanición de buffer se pueden formalizar según la expresión (5).

$$c(n) \geq e(n) = v(n) \quad \forall n \tag{5}$$

Es decir, en todo momento se debe estar contratando como mínimo igual cantidad de recursos a la red que los requeridos instantáneamente por la secuencia de vídeo y en todo momento el tráfico entregado debe ser superior al requerido para decodificar la secuencia, que en este caso particular, al no realizarse técnicas

de buffering, coinciden. Esta restricción suele siempre cumplirse en exceso y ello conlleva decrementos drásticos de la eficiencia.

Cuando se usan técnicas de buffering el tráfico entregado a la red no tiene porqué coincidir con el codificado instantáneamente o con el necesario en decodificación en un determinado momento. La condición de uso de buffering queda reflejada en (6). En este tipo de mecanismos, el estudio de las funciones acumulativas nos lleva a resultados mucho más fáciles de entender.

$$e(n) \neq v(n) \qquad (6)$$

Si la técnica de buffering se realiza en emisión, como es el caso típico de todas las técnicas de conformación de tráfico, el tráfico total acumulado entregado a la red es menor que el total codificado en un cierto instante de tiempo (7).

$$E(n) \leq V(n) \qquad (7)$$

Esta diferencia se traduce en un retardo de decodificación en recepción (D expresado en tiempos de cuadro). Además se debe controlar el tamaño del buffer en emisión (que denominaremos B) para que no se produzca saturación de buffer. Esta condición se explicita en (8).

$$V(n) - E(n) < B \qquad (8)$$

En este caso la restricción de no inanición de buffer se puede expresar como que el total de datos entregados a la red debe ser siempre igual o superior a los totales decodificados en un cierto instante de cuadro menos el retardo introducido por el buffer de emisión, véase expresión (9).

$$E(n) > V(n - D) \qquad (9)$$

Si la técnica de buffering se realiza en recepción, como en el caso de todas las técnicas work-ahead, el tráfico total acumulado entregado a la red es mayor que el total decodificado en un cierto instante de tiempo, tal como se expresa en (10).

$$E(n) \geq V(n) \qquad (10)$$

Esta diferencia, aunque también nos lleva a un control del tamaño del buffer, expresión (11), produce un retardo nulo. En este caso, como el retardo es nulo, la no inanición de buffer se expresa igual que en (10) y se deriva de forma natural del propio mecanismo de work-ahead buffering.

$$E(n) - V(n) \leq B \qquad (11)$$

En cualquiera de los dos casos la expresión (12) define la eficiencia, calculada como el cociente entre el total contratado al operador y el total entregado a la red.

$$\eta = \frac{C(n)}{E(n)} \qquad (12)$$

Para una mayor comprensión de todo lo expuesto en la Fig. 2. se representan las condiciones discutidas.

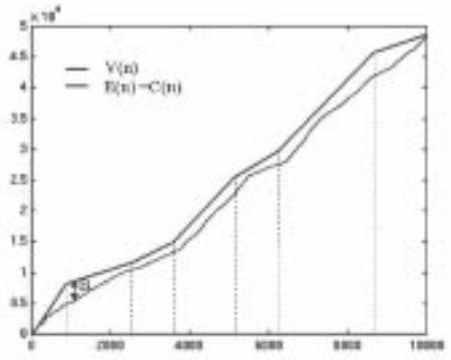


Figura 2: Esquema gráfico de las restricciones a cumplir por el sistema propuesto.

3.2 Número de niveles de contrato.

Tal como habíamos expuesto en la sección 2, los dos parámetros más importantes a la hora de realizar un servicio RCBR son los niveles de contrato, tanto su número como su valor, y los puntos de renegociación. El sistema 3-RCBR basa la elección del número de niveles en el modelo bidimensional de la secuencia de vídeo, que caracteriza de forma natural el tráfico VBR con tres niveles de actividad.

Dicha elección es adecuada para un sistema de codificación en tiempo real, pues pretende que la tasa contratada "siga" el nivel de actividad real e instantáneo de la fuente de vídeo. En un sistema off-line, en cambio, dicha correlación entre la tasa codificada y la tasa entregada a la red no tiene sentido, ya la secuencia es conocida a priori y consecuentemente se pueden aprovechan los recursos de memoria en recepción para "romper" dicha dependencia entre tasas codificada y entregada. Este hecho también se exploró en [2] pero el propio sistema resultaba en un número indeterminado de niveles de contrato, dependientes de la secuencia codificada y los recursos de memoria del cliente y de valor no determinístico.

En este trabajo se pretende encontrar el número mínimo de niveles fijo y de valor determinado que garantice una correcta entrega a la red del vídeo codificado y que cumpla todas las restricciones expuestas anteriormente.

De la observación de la Fig. 2 se deduce que nuestro problema se podría reformular como la aproximación de una función matemática *V(n)* mediante un número no determinado de tramos lineales de distinta pendiente de forma que la distancia entre la función original y su aproximación (*E(n)=C(n)*) sea mínima. Por otra parte, también queremos que el número de pendientes sea el mínimo posible, y siempre bajo las restricciones expuestas al comienzo de esta misma sección.

Al estudiar la naturaleza de la función *V(n)* se concluye que se trata de una función monótona creciente con pasos de concavidad a convexidad (y viceversa) producidos por los distintos ritmos de crecimiento que presenta a lo largo de su dominio. El comportamiento de esta función se asemeja al de la función error, en la Fig. 3 se observa que si se pretende aproximar una función de este estilo mediante un solo tramo (una única pendiente) se viola la restricción de inanición de buffer (caso 1) o disminuye la eficiencia final (caso 2). El número mínimo de tramos con el que podemos aproximar una función que siga el comportamiento de *V(n)* cumpliendo las condiciones expuestas es 2. Por tanto el número mínimo de tasas que debemos renegociar con el operador para servir una secuencia de vídeo de este estilo será 2. Nuestro problema ahora se traduce en hallar cuáles son esos dos valores.

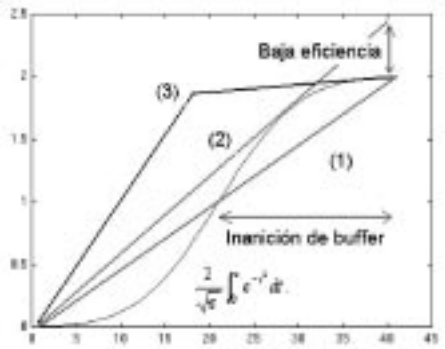


Figura 3: Modelado de la función V(n) como función de error.

3.3 Cálculo de los niveles del contrato

Una vez ya hemos decidido que aproximaremos la función *V(n)* mediante intervalos lineales de dos posibles valores de pendiente debemos hallar cuáles son esos valores. Para ello deberemos estudiar la función *V(n)* con un poco más de detalle. Las expresiones (13) y (14) se deducen de la misma definición de *V(n).*

$$V(n+1)-V(n) \le Max\{v(n)\} = tasa\ max. \quad (13)$$

$$V(n)-V(n-1) \ge Min\{v(n)\} = tasa\ min. \quad (14)$$

Para que el mecanismo de renegociación sea lo más general posible, consideremos que cualquier punto de la secuencia *V(n)* puede convertirse en un punto de renegociación. Existirán dos tipos de puntos de renegociación, puntos de buffer máximo y puntos de buffer mínimo. Los primeros se corresponden con la renegociación de nivel alto a nivel bajo mientras que los segundos se corresponden con la renegociación de nivel bajo a nivel alto. Los primeros no son críticos en cuanto a restricción de inanición de buffer ya que precisamente dichos puntos son los que requieren tamaños de buffer mayores. Los puntos de renegociación de nivel bajo a nivel alto, en cambio, sí que son críticos ya que por definición del algoritmo son los que tienen un tamaño de buffer requerido nulo, y por tanto la tasa servida en instantes precedentes o posteriores a dicho punto nos puede llevar al incumplimiento de las restricciones temporales.

Supongamos cualquier punto con índice *i* de la secuencia *V(n)* como punto de renegociación de segundo tipo. Índices precedentes a dicho punto se corresponden con índices pertenecientes a un intervalo de tasa contratada baja, y similarmente, índices posteriores al citado punto se corresponden con índices de un intervalo de tasa contratada alta, ecuaciones (15) y (16).

$$E(i)-E(i-1) = tasa\ baja \quad (15)$$

$$E(i+1)-E(i) = tasa\ alta \quad (16)$$

Como habíamos comentado, en estos puntos el tamaño del buffer es nulo (*E(i)=V(i)*). Como además debemos cumplir (10) para cualquier valor de *n*, se pueden plantear las inecuaciones (17) y (18) de las que se deduce que la tasa alta debe ser la tasa máxima de la secuencia codificada de vídeo y la tasa baja la tasa mínima de la misma secuencia.

$$E(i+1)-E(i) \ge V(i+1)-V(i) \le tasamax \quad (17)$$

$$E(i)-E(i-1) \ge V(i)-V(i-1) \le tasa\ min \quad (18)$$

3.4 Cálculo de los puntos de renegociación.

Por último sólo nos falta decidir cuáles serán los puntos en los que vamos a renegociar tasas con la red. Recordemos que nuestro objetivo es aproximar la función *V(n)* mediante un número aún indeterminado de tramos lineales de dos posibles pendientes de valores ya determinados. Ya comentamos que de la propia definición de *V(n)* se deduce que es una función monótona creciente con distintos ritmos de crecimiento en su recorrido. Estos cambios en el ritmo de crecimiento producen cambios de concavidad a convexidad y viceversa en la función *V(n*). Estudiando con detalle dichos pasos de concavidad a convexidad se puede

observar que son causados por cambios de complejidad en la codificación de la imagen de vídeo, o lo que es lo mismo, por cambios de escena. Cambios de escena significativos producirán puntos de inflexión pronunciados mientras que cambios de escena no significativos producirán puntos de inflexión poco pronunciados.

Sea *f(t)* una función continua y derivable en todo su dominio. Se definen los puntos de inflexión como los puntos en que la función experimenta un cambio en su sentido de concavidad. Por teorema de continuidad se puede demostrar que estos puntos presentan un valor de la segunda derivada nulo. A su vez, podemos clasificar los puntos de inflexión en dos grupos: fuertes y débiles dependiendo de la pendiente que presente la función f(t) en dichos puntos.

Nótese que según el umbral escogido la función *f(t)* presentará más o menos puntos de inflexión fuertes. De este modo se pueden usar los puntos de inflexión como indicativos de los puntos de renegociación de tasa y podremos controlar el número de renegociaciones mediante la variación del umbral de definición de punto de inflexión fuerte.

Para entender mejor la aplicación a nuestro problema de dichas afirmaciones consideraremos continua y derivable la función *V(n)*. Nótese que este cambio no afecta en la generalidad de las afirmaciones expuestas.

Sea *v(t)* la función continua y derivable que describe la cantidad de bits necesarios para codificar una secuencia de vídeo en un determinado instante de tiempo *t*.

Se define *V(t)* como la función continua y derivable que describe la cantidad acumulada total de bits requeridos para codificar una secuencia de vídeo hasta el instante *t*. Dicha función puede expresarse tal como se indica en (19), y consecuentemente se puede definir *v(t)* en función de *V(t)* tal como se refleja en (20). Finalmente en (21) se detalla la relación existente entre la secuencia acumulativa, *V(t)*, la secuencia patrón correspondiente a la codificación del vídeo, *v(t)*, y la función que indica el ritmo de crecimiento y decrecimiento de la secuencia patrón, $\dot{v}(t)$.

$$V(t) = \int_0^t v(s)\,ds \tag{19}$$

$$v(t) = \frac{dV(t)}{dt} \tag{20}$$

$$\dot{v}(t) = \frac{dv(t)}{dt} = \frac{d^2V(t)}{dt} \tag{21}$$

De estas expresiones se deduce que el cálculo de los puntos de renegociación se reduce a hallar los puntos en que $\dot{v}(t)$ se anula, y se correspondan con un valor de *v(t)* mayor que un determinado umbral. Dichos puntos nos proporcionarán una noción de dónde deben existir intervalos de renegociación, ya que estos puntos dividen la secuencia completa en intervalos que presentan un comportamiento aproximado a un punto de inflexión como el de la Fig 3. De todos modos, debe decidirse cuáles serán los puntos exactos de renegociación, o lo que es lo mismo, cómo aproximamos ese único punto de inflexión fuerte mediante un par de intervalos lineales.

Por una parte sabemos que debemos empezar a servir el tráfico a la red a tasa máxima. Del cálculo de los puntos de inflexión fuertes sabemos que la función integrativa presentará un salto pronunciado alrededor de dicho punto. Por diseño del sistema sabemos que debemos acabar el servicio en ese intervalo a una tasa mínima y con un tamaño de buffer nulo (*V(n)=E(n)*). Debemos encontrar un punto donde la función acumulativa vuelva a tener ritmos de crecimiento suaves. Ya hemos discutido ampliamente que dichos ritmos de crecimiento se corresponden con los valores de su derivada, es decir, con los valores de la secuencia patrón. Se considerará el punto final del intervalo de inflexión aquel en que el ritmo de crecimiento vuelve a su valor medio. El punto intermedio de renegociación se halla de forma natural de la intersección de los dos tramos lineales.

3.5 Algoritmo práctico de cálculo de los puntos de renegociación.

A continuación se presenta una forma práctica de calcular los puntos de renegociación a partir de la secuencia de vídeo conocida. Recordemos que en realidad las funciones de las que disponemos no son continuas, y que inicialmente sólo disponemos de los datos que nos proporciona v(n). Consecuentemente deberemos plantear un algoritmo que se base únicamente en datos conocidos.

Calcúlese V(n) a partir de v(n) según la expresión (1) para tantos puntos como longitud tenga v(n).

Calcúlese $\dot{v}(n)$ a partir de v(n) según la expresión $\dot{v}(n) = v(n) - v(n-1)$ para n tomando valores de 2 a N y con valor inicial nulo, $\dot{v}(1) = 0$.

Encuéntrese los puntos de inflexión fuertes como los que cumplan la condición $\dot{v}(n) \geq factor \cdot std\{v(n)\}$.

Encuéntrese los puntos inmediatamente posteriores al índice n de los puntos de

inflexión fuertes que cumplan $v(n) \leq media\{v(n)\}$.

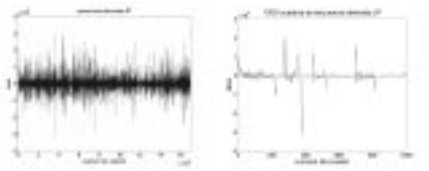
Todos dichos puntos junto con n=1 formarán parte del subconjunto de puntos de renegociación correspondientes a tamaño de buffer mínimo. Este subconjunto se puede expresar como un vector de índices ya que se corresponden con los puntos de renegociación impares. Consideramos siempre el primer elemento de los vectores con índice igual a 1.

Hállese los puntos de renegociación pares, según la expresión (22).

Los tramos comprendidos de índices impares a pares se corresponden con tramos de tasa entregada máxima, los tramos comprendidos de índices pares a impares se corresponden con tramos de tasa entregada mínima.

3.6 Justificación del algoritmo.

Aun quedan por justificar dos aproximaciones realizadas. En primer lugar, para el cálculo de los puntos de inflexión se toma aquellos que presentan un valor positivo y grande de la segunda secuencia $\dot{v}(n)$. Esto se justifica a partir de la observación de la propia secuencia $\dot{v}(n)$, Fig 4 .



(a) secuencia entera (b) 1000 primeras muestras

Figura 4: Secuencia incremental $\dot{v}$(n) de la película J.P.

Al no ser la función continua, los puntos en los que dicha función se anulan no se corresponden con índices naturales de la función sino a segmentos de extrapolación. Así, los puntos de inflexión no son valores nulos de la segunda derivada sino transiciones de valor negativo a positivo. Además existen numerosos de estos puntos que no indican cambios bruscos. Los puntos buscados son cambios de ritmo bruscos y positivos en la secuencia patrón. Dichos puntos son aquellos en que la primera derivada de la secuencia patrón toma valores positivos elevados, es decir, $\dot{v}(n) > umbral$.

4 Resultados obtenidos.

A continuación se presentan los resultados obtenidos de la simulación del algoritmo 2-RCBR sobre 4 secuencias de vídeo previamente conformadas. Las secuencias son: Jurassic Park y Concert by America Band, PAL y de 170000 y 34000 cuadros de longitud respectivamente, y Blade Runner y Concert by Neil Young , NTSC y de 156000 y 47000 cuadros cada una.

La tabla 1 muestra el número de renegociaciones para cada una de las secuencias en función de distintos umbrales de punto de inflexión fuerte. Se observa una reducción drástica cuando el umbral se toma alrededor de la desviación típica de la secuencia incremental. Este valor reducido en cuanto a número de renegociaciones conlleva un incremento de buffer en recepción, efecto no deseado, por lo que se debe hallar un compromiso entre número de renegociaciones y tamaño de buffer del cliente. Valores de umbral cercanos a 0.75 veces la desviacion típica ofrecen del orden de 1 renegociación cada 2500 cuadros, es decir, una media de 1 renegociación cada minuto y medio. Umbrales cercanos a la mitad de la desviación típica ofrecen valores del orden de 1 renegociación cada 750 cuadros, o 30 segundos en media. Éstos a su vez, presentan un comportamiento de buffer de cliente siempre inferior a los 100 MBytes.

	Std	0.75std	0.5std	0.25std
J.Park	13	71	211	1078
Neil Y.	4	6	16	309
Blade R.	91	217	551	2019
America	5	23	104	541

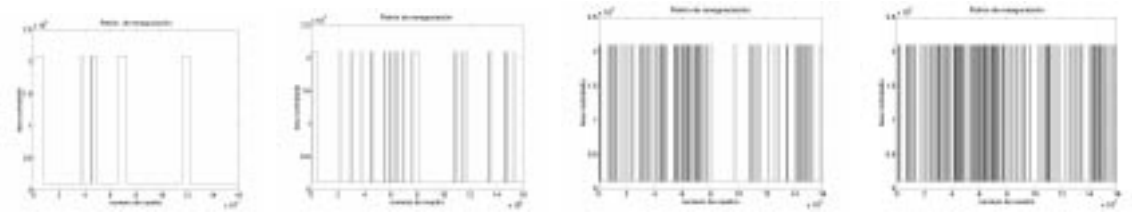
Tabla 1. Renegociaciones necesarias para distintas secuencias de vídeo y umbrales.

En la Fig. 5 se pueden ver los patrones de renegociación para distintos umbrales, mientras que en la Fig 6 se observa lo precisa que es la aproximación mediante tramos lineales de la función *V(n).* Finalmente en la tabla 2 y la Fig. 7 se detallan los tamaños de buffer de cliente necesarios para no producirse saturación.

	Std	0.75std	0.5std	0.25std
J.Park	150 MB	90 MB	41 MB	14.5 MB
Neil Y.	42 MB	38 MB	12 MB	3.5 MB
Blade R.	230 MB	114 MB	67 MB	22.5 MB
America	23 MB	12 MB	8 MB	2.5 MB

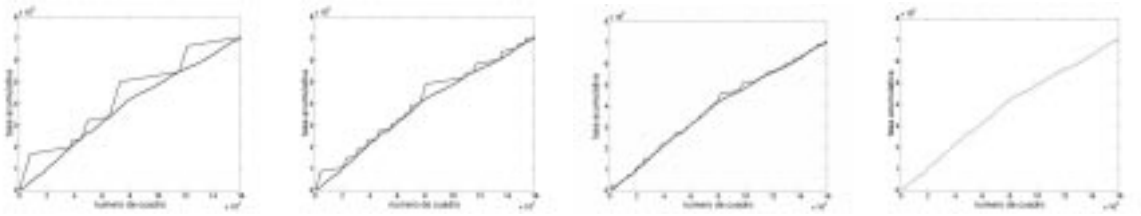
Tabla 2. Tamaño del buffer para distintas secuencias de vídeo umbrales.

$$\dot{p}(2n) = \frac{\{V(p(2n+1) - V(p(2n-1))\} - [tasamedia\{v\} \cdot (p(2n+1) - p(2n-1))]}{tasamaxima\{v\} - tasamedia\{v\}} \qquad (22)$$



(a) umbral std (b) umbral 0.75std (c) umbral 0.5std (d) umbral 0.25std

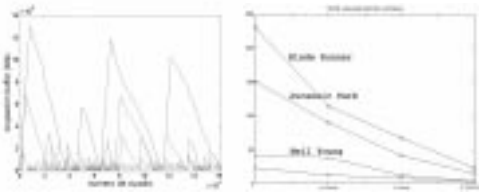
Figura 5: Patrones de los intervalos de renegociación para la secuencia Jurassic Park suavizada con distintos umbrales de punto de inflexión fuerte.



(a) umbral std (b) umbral 0.75std (c) umbral 0.5std (d) umbral 0.25std

Figura 6: Aproximaciones mediante tramos lineales de la secuencia Jurassic Park con cálculo de intervalos de renegociación para distintos umbrales de punto de inflexión fuerte.

5 Conclusiones.



(a) Distintos umbrales (d) Distintas secuencias.

Figura 7: Tamaño del buffer necesario.

En este artículo se ha presentado un nuevo mecanismo de entrega de tráfico de vídeo almacenado a la red mediante distintos tramos a tasa constante. Se ha demostrado que el número mínimo de niveles de renegociación debe ser 2 y que para cumplir las restricciones temporales impuestas en este tipo de servicios sus valores deben ser la tasa máxima y mínima de la secuencia a transmitir. El empleo de únicamente dos niveles simplifica en gran medida la señalización de renegociación . A su vez, en el diseño del algoritmo se ha forzado que presente una eficiencia máxima (del 100%), es decir, que no se contraten con el operador más recursos de los que realmente se necesitan para transmitir la secuencia, y que presente un retardo nulo. En otras palabras, en cualquier instante de tiempo, incluido el inicial, se dispone de suficientes datos en decodificación para poder decodificar la secuencia instantáneamente. Estas mejoras se consiguen gracias a técnicas de work-ahead buffering que utilizan los recursos de memoria del cliente para enviar datos "a priori" y no malgastar recursos contratados con la red.

Las prestaciones del nuevo sistema se han validado sobre distintas secuencia patrón. El número de renegociaciones es controlable por diseño. El aumento del número de renegociaciones disminuye drásticamente el tamaño del buffer en recepción.

Agradecimientos

Este trabajo ha sido soportado por el proyecto SSADE [CICYT, proyecto TEL 99-0322], dentro del Plan Nacional de I+D.

Referencias

[1] L.J de la Cruz, J. Mata, "Performance of Dynamic Resource Allocation with QoS guarantees for MPEG VBR Video Traffic Transmission over ATM Networks", Proceedings of the IEEE Global Conference on Communications (GLOBECOM99), pp. 1483-1489, Rio de Janeiro, RJ, Brasil, diciembre 1999.

[2] Salehi, James D. And Zang, Zhi-Li. "Supporting Stored Video: Reducing Rate Variability and End-to-End Resource Requirements Through Optimal Smoothing", IEEE/ACM Transactions on networking, vol 6 num 4 August 1998.

[3] Seungyup, Pae And Shih-Fu, Chang. "Video-Server Retrieval Scheduling and Resource Reservation for Variable Bit Rate Scalable Video", IEEE Transactions on circuits and systems for video technology. Vol 10, num 3 April 2000.

Análisis de Minimización de Coste en la Transmisión de Flujos Semi-Elásticos sobre Internet

Marcos Postigo Boix[1], Joan García Haro[2], Mónica Aguilar Igartua[1]
[1]Departamento de Ingeniería Telemática, Universitat Politècnica de Catalunya
C/ Jordi Girona 1-3, Mòd. C3, Campus Nord, 08034 Barcelona.
[2]Departamento de Tecnologías de la Información y las Comunicaciones, Universidad Politécnica de Cartagena
Campus Muralla de Mar s/n, 30202 Cartagena.
E-mail: mpostigo@mat.upc.es; joang.haro@upct.es; maguilar@mat.upc.es

Abstract. *Because the dramatic growth of the Internet and the requirement of Quality of Service (QoS) guarantees for the new and future applications, an Internet that ensures end-to-end QoS is essential. To provide this level of QoS, many protocols have evolved in the last years. Also, it is necessary to classify applications by its QoS requirements to understand how they reserve resources from the network. In this paper, we briefly review the existing end-to-end QoS protocols and an end-to-end QoS architecture that combines all them is proposed. In addition, we classify Internet data flows by their QoS requirements. We focus on the study of semi-elastic flows and the minimization of their transmission cost. First, we present a client-server system to transmit these flows while reducing the cost. Also, we analyze the buffer management required at the client. Finally, we implement the system in the Network Simulator 2 [1] to demonstrate that the analytical study can be used in a realistic scenario to minimize the cost.*

1 Introducción

El aumento exponencial del número de ordenadores, cantidad de tráfico generado y número de enlaces, es un buen indicador para mostrar el impresionante crecimiento de Internet. Además, se espera que Internet ofrezca nuevos servicios de telecomunicación que requieran nodos de conmutación de alta velocidad que además garanticen de Calidad de Servicio (QoS) [2].

Actualmente, Internet ofrece un servicio *best-effort* sin garantía de QoS, pero se han diseñado varios protocolos para proporcionar un servicio adecuado al tráfico con requerimientos temporales muy estrictos [3][4].

El protocolo de reserva de recursos RSVP (*ReSerVation Protocol*) es un protocolo de señalización que provee de reservas de recursos de red (modelo de Servicios Integrados) a flujos individuales y agregados [5][6]. En cada nodo la reserva de ancho de banda se mantiene mediante un algoritmo de planificación de servicio (*scheduling algorithm*), como el WFQ (*Weighted Fair Queueing*) [7] o el WF^2Q (*Worst-case Weighted Fair Weighted Fair Queueing*) [8]. Fundamentalmente, los Servicios Integrados se dividen en dos tipos: Garantizado (*Guaranteed Service*) [9] y Carga Controlada (*Controlled Load*) [10]. El servicio Garantizado es lo más parecido a una emulación de circuito virtual, limitando el retardo extremo a extremo debido a encolamientos y asegurando la disponibilidad de ancho de banda durante la transmisión. El servicio de Carga Controlada ofrece una QoS parecida a la que ofrece el servicio *best-effort* en una red infrautilizada, pero no se asegura ningún tipo de límite en el retardo extremo a extremo.

RSVP permite a las aplicaciones la reserva de ancho de banda con una gran granularidad y con las mejores garantías de QoS. Por otro lado, tiene serios problemas en cuanto a escalabilidad y manejabilidad [11]. Esto es debido a que cada nodo debe soportar RSVP y reservar recursos para cada flujo individual. Ello hace que sea difícil de gestionar, particularmente en la Internet troncal, donde puede haber miles de flujos.

Los Servicios Diferenciados [12][13] fueron propuestos por el IETF *(Internet Engineering Task Force*) para resolver los problemas de escalabilidad presentados por el esquema de Servicios Integrados. Proporcionan un método simple de priorizar agregados de tráfico usando etiquetas cortas. En este momento, hay definidas dos formas de proceder en los nodos (*per hop behavior*, PHB) creando dos niveles de servicio: Transporte Acelerado (*Expedited Forwarding*) y Transporte Asegurado (*Assured Forwarding*). El primero, ofrece el nivel más alto de QoS agregada, garantizando el ancho de banda como una "línea dedicada virtual" con retardo de espera en colas nulo o muy pequeño. El segundo, etiqueta los paquetes como *in* o *out* para indicar la conformidad con el perfil del tráfico. Los *routers* utilizan esta información en caso de congestión. Así, los paquetes *in* son raramente descartados, mientras que los paquetes *out* se descartan primero. Algunos de los mecanismos de gestión de colas usados para descartar paquetes son [14]: el descarte según un umbral, RIO [15] (*RED* [16] *with In and Out Packets*) y el descarte con cola llena.

Los Servicios Diferenciados tienen la ventaja de mover la complejidad hacia los extremos de la red. De esta forma, los paquetes de datos se pueden etiquetar y agregar en los *routers* de los extremos basándose en los perfiles del tráfico. Posteriormente, los *routers* de la red troncal pueden transportar los paquetes de acuerdo a las etiquetas sin la necesidad de examinar con detalle las cabeceras individuales de los paquetes.

MPLS (*Multi Protocol Label Switching*, MPLS) [17] es una técnica de transporte basada en el intercambio de etiquetas. En este caso, al igual que pasa con los Servicios Diferenciados, los paquetes se etiquetan en los puntos de ingreso y se desetiquetan en los de salida. Pero aquí, las marcas se usan para determinar el siguiente nodo en vez de usarse para priorizar el tráfico. En el primer nodo de una red que permite MPLS, se adjunta una etiqueta al paquete identificando la clase de equivalencia de Transporte (*Forwarding Equivalence Class*, FEC) basándose en las direcciones de origen y destino y en el nivel de prioridad. En el siguiente nodo, la etiqueta se usa para buscar en una tabla el siguiente nodo y una nueva etiqueta. De nuevo, se adjunta la etiqueta y el paquete se envía al siguiente nodo. Mediante este método, los nodos no tienen que analizar la cabecera completa del paquete ahorrando tiempo, y se pueden crear rutas explícitas eliminando el tiempo de decisión en los nodos intermedios.

SBM (*Subnet Bandwidth Management*) está diseñado para categorizar y priorizar el tráfico en redes de área local (LAN) conmutadas o compartidas. Es un protocolo de señalización [18] entre los nodos y conmutadores de la red presentada en el escenario SBM [19] y define el mapeo entre los protocolos de QoS de niveles superiores y las especificaciones de QoS en las tecnologías de subred [20].

Todas estas tecnologías se usan para proporcionar QoS extremo a extremo. En realidad, en las redes existentes se combinan y se utilizan conjuntamente. La Fig. 1 muestra una arquitectura que mezcla todas las tecnologías explicadas anteriormente [21]. Aquí, el RSVP se emplea en los extremos de la red para proveer reservas de recursos dentro de un mismo dominio. Por su parte, las aplicaciones emisoras pueden usar el mapeo SBM con RSVP para notificar a la red sus requerimientos de QoS. Los Servicios Diferenciados se aplican en la red troncal, ya que los paquetes se marcan en los *routers* de los extremos haciendo que los nodos de conmutación troncales necesiten menos requerimientos para ofrecer QoS. RSVP puede ayudar también a MPLS a proveer reservas de ancho de banda a sus caminos virtuales [22]. Por su parte, RSVP también puede utilizar MPLS para predeterminar los caminos que toman sus flujos [23]. Asimismo es posible mapear el tráfico de los Servicios Diferenciados en rutas MPLS [24].

Como se ha mostrado en la Fig. 1, hay otras tecnologías usadas en este escenario: Mediadores de Ancho de Banda (*Bandwidth Brokers*, BB) y COPS (*Common Open Policy Service*). Los BBs [25][26] se usan para reservar recursos entre dominios y para comunicar entre dominios vecinos el SLA (*Service Level Agreement*) o contrato entre cliente y proveedor o entre dominios. Finalmente, COPS [27] es un servicio de control de policía usado para intercambiar información de configuración entre BBs.

Las aplicaciones que utilizan esta arquitectura tienen diferentes requisitos de QoS. Idealmente, todas las aplicaciones emisoras desean que sus datos lleguen íntegros a su destino. Para conseguir esto es necesaria la reserva de recursos o en el caso de que hayan pérdidas y la aplicación pueda permitirse esperar, la retransmisión de paquetes puede ser suficiente. Algunas aplicaciones pueden soportar un cierto nivel de pérdidas degradando la QoS, pero realmente, esto no es deseable en una red con QoS extremo a extremo robusta. Por otro lado, las aplicaciones pueden ser sensibles al retardo. Así, las que tengan unos requerimientos temporales más estrictos usarán más recursos que las que no sean sensibles al retardo.

Obviamente, los Proveedores de Servicios de Internet (ISPs) obtendrán mayores beneficios si ofrecen servicios de QoS extremo a extremo. Por este motivo, las aplicaciones deben usar de forma eficiente los recursos reservados para minimizar el coste de la transmisión.

En este artículo, en la Sección 2, clasificaremos los flujos de datos por sus requerimientos en cuanto a reserva de recursos. En la Sección 3, nos centraremos en el estudio de la minimización del coste de transmisión de flujos semi-elásticos. Primero, definiremos el sistema cliente-servidor usado para transmitir estos flujos y seguidamente se mostrará el análisis de la gestión de la memoria necesaria para minimizar el coste. La Sección 4 muestra la implementación del sistema cliente-servidor en el simulador NS-2, y algunos de los resultados de simulación obtenidos del funcionamiento real del sistema. Finalmente, en la Sección 5 se muestran las conclusiones más significativas y algunas líneas de trabajo futuras.

2 Tipos de Flujos

En Internet, cada flujo tiene sus propios requerimientos de QoS. Estos requisitos son fijados por las aplicaciones que generan y usan estos flujos, aunque es realmente el usuario de la aplicación el que finalmente marca el criterio para especificar dichos requerimientos.

Cuando un flujo es elástico [28][29] (tolerante al retardo), no tiene requerimientos temporales específicos de QoS. En este caso, la información transmitida mediante este tipo flujo se espera que llegue al destino en cualquier instante (Fig. 2). Éste es el caso del servicio de correo electrónico, donde el mensaje se procesa (es leído por el destinatario) cuando el usuario advierte su llegada.

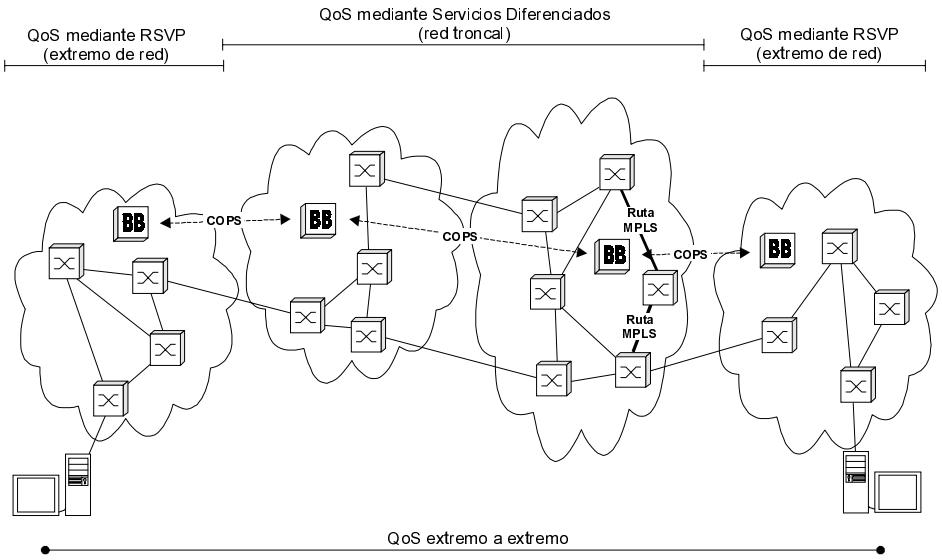


Fig. 1: Red con QoS extremo a extremo que usa diferentes tecnología de QoS.

Por otro lado, nos referimos a flujos con requerimientos temporales de QoS muy estrictos como flujos inelásticos. En este caso, la información tiene que llegar al destino en un instante de tiempo específico para poder ser consumida (Fig. 2). Normalmente, esta información se genera en directo y es esencial la reserva de recursos de la red durante toda la transmisión para asegurar la entrega correcta de la información sin degradar la QoS requerida. Un ejemplo de este tipo de flujo es la telefonía IP.

Finalmente, definimos otro tipo de flujo: el semi-elástico [30]. Este flujo requiere un cierto nivel de QoS, pero a menudo no es necesario mantenerlo durante toda la transmisión. En este contexto, la información necesita llegar al cliente en un instante muy específico, como pasa con los flujos inelásticos, pero la red puede entregarla antes (Fig. 2). Esto es lo que sucede con la información prealmacenada, como es el caso de los servidores denominados *continuous media servers* [31]. Estos flujos, necesitan que la información se almacene en una memoria y que se consuma a la tasa de lectura del receptor.

Como veremos a continuación, es posible minimizar el coste de la transmisión de los flujos semi-elásticos reduciendo la cantidad de información enviada cuando se reservan recursos. Ello es obvio, ya que si la red está infrautilizada no será siempre necesario reservar recursos, mientras que si la red está muy cargada, éstos se tendrán que mantener durante toda la transmisión.

3 Minimización del Coste de la Transmisión de Flujos Semi-Elásticos en un Sistema Cliente-Servidor

En este estudio estamos interesados en reducir el coste de la transmisión de los flujos semi-elásticos. En esta sección, describiremos un sistema cliente-servidor que transferirá un flujo semi-elástico de información (p.e., una transferencia de vídeo prealmacenado o un fichero) y qué es lo que se hace para minimizar el coste asociado a dicha transmisión. Seguidamente, definiremos el indicador de coste que utilizaremos y la eficiencia del método de minimización. Para finalizar, mostraremos el método de gestión de memoria que minimiza el coste de la transmisión.

3.1 Transmisión de Flujos Semi-Elásticos en un Sistema Cliente-Servidor

En un sistema cliente-servidor, los clientes generan peticiones de información al servidor. En este estudio, suponemos que la información que se pide se puede transmitir con un flujo semi-elástico. En la Internet actual, esta información se transmite con el servicio *best-effort*. Esto significa que si la red está saturada, la información llegará con un retardo alto y no será procesada si los requerimientos temporales son muy estrictos. En realidad, lo que pasa normalmente es que el cliente espera durante un cierto tiempo a que se llene su memoria antes de empezar a procesar la información para asegurar su disponibilidad. Además, la información se transmite con los protocolos TCP, si es posible retransmitir paquetes, o UDP con algún método añadido para permitir control de congestión y fiabilidad en la transmisión.

En una Internet con QoS extremo a extremo es posible reservar recursos permitiendo la correcta transmisión de estos flujos semi-elásticos. Como se mostró en la Fig. 1, en los extremos de la red, se utiliza

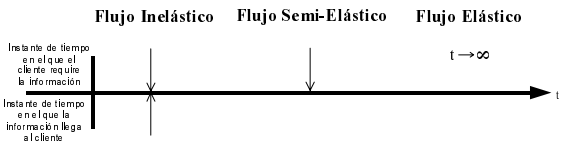


Fig. 2: Requerimientos temporales de los flujos.

RSVP para informar a la red de los requerimientos de reserva de recursos necesarios. Por tanto, las aplicaciones cliente-servidor podrán usar este protocolo para reservar ancho de banda y garantizar un retardo mínimo que asegure la entrega correcta de la información. Si el ancho de banda reservado es igual a la tasa de lectura del cliente, la reserva se mantendrá durante toda la transmisión y la memoria del cliente se mantendrá vacía. No obstante, si el ancho de banda reservado es mayor que la tasa de lectura, la información llenará la memoria del cliente. Así, sería posible enviar toda la información al cliente si la memoria fuese suficientemente grande. Como vamos a explicar a continuación, esta opción no es muy útil cuando no toda la información es necesaria al mismo tiempo (algunas partes de un vídeo interactivo pueden saltarse), y además se requiere reservar ancho de banda durante toda la transmisión incrementando innecesariamente el coste de la comunicación.

Es posible mejorar la transmisión fijando un umbral máximo, en vez de llenar la memoria del cliente hasta que todos los datos se hayan enviado. Cuando los datos almacenados en la memoria del cliente llegan al umbral máximo, el cliente informa a la red para que cambie al modo *best-effort* (BE) (supuesto más económico). Durante la transmisión en modo *best-effort*, los paquetes de datos llegan con una tasa que depende del estado de la red. Esta tasa puede ser menor o igual que la tasa de lectura del cliente. En el caso de ser igual, la cantidad de datos almacenados se mantendrá, mientras que si la tasa es menor, la cantidad de datos decrecerá. Lo más importante es la cantidad de datos que llegan durante el periodo de *best-effort*, ya que estos datos no se entregan mediante reserva de recursos (ReR), lo cual reduce el coste de la transmisión.

El coste de transmisión dependerá de la cantidad de datos entregados mientras se reservan recursos. Por tanto, definimos el coste (C) que queremos minimizar, como la cantidad de información entregada durante el modo de reserva de recursos (1). En (1), t_{ReR} es el tiempo durante el que se reservan recursos y α_{ReR} es la tasa de entrega de datos al cliente durante dicho tiempo. Naturalmente, el coste total dependerá de otros factores, pero nos centraremos en el coste relacionado con el uso de la reserva de recursos.

$$C = t_{ReR} \cdot \alpha_{ReR} \quad (1)$$

Por tanto, el máximo coste es el tamaño total de la información o archivo que se transmite (nos referiremos a este tamaño como *FS*, *File Size*). Por otro lado, definimos la eficiencia del método de minimización como sigue:

$$\eta = 1 - \frac{C}{C_{MAX}} = 1 - \frac{C}{FS} \quad (2)$$

La ecuación (2) muestra la mejora en la reducción del coste respecto del coste de mantener la reserva de recursos durante toda la transmisión ($C_{MAX} = FS$).

3.2 Gestión de la Memoria

Los datos enviados al cliente deben ser almacenados en una memoria, ya que suponemos que la tasa de entrega de datos durante la reserva de recursos es mayor que la tasa de lectura del cliente. Esta memoria almacenará los paquetes de datos antes de que el cliente los use. Por su parte, los paquetes llegan de capas de protocolos de transporte inferiores, como pueden ser TCP o UDP.

Para minimizar el coste es necesario entender el comportamiento de los datos almacenados en la memoria y también es necesaria una determinada gestión de la memoria en cuestión. Primero, definiremos los parámetros principales de la memoria. A continuación, determinaremos la dinámica de los datos almacenados, y finalmente, se definirá el criterio de minimización a utilizar.

3.2.1 Parámetros de la Memoria

En la Fig. 3 se muestra la memoria del cliente. Como se puede observar, hay tres parámetros principales que definen la memoria: tamaño de la memoria (M), umbral máximo (Max) y umbral mínimo (Min).

M es el espacio de memoria reservado por el cliente para almacenar los datos que llegan del servidor. Así, la cantidad de datos guardados debe estar siempre por debajo de este límite.

Como se explicó anteriormente, el umbral Max se usa para cambiar entre el modo con reserva de recursos y el *best-effort*. Como veremos después, este es el parámetro más relevante ya que es la clave para minimizar el coste.

Finalmente, el umbral Min se usa para cambiar del modo de *best-effort* al modo de reserva de recursos. Este valor debe ser definido por el cliente para garantizar la disponibilidad de datos durante todo el tiempo.

En este estudio nos centraremos en el control de Max para minimizar el coste de la transmisión de flujos semi-elásticos. El dimensionado de M y Min está fuera del ámbito de esta investigación.

3.2.2 Dinámica de los Datos Almacenados

La cantidad de información almacenada en la memo-

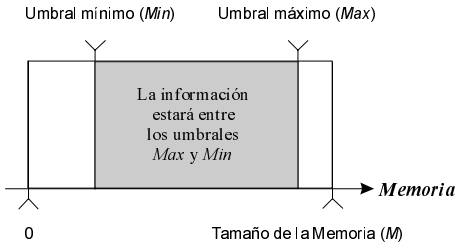


Fig. 3: Memoria del Cliente.

ria del cliente varía a través del tiempo dependiendo del modo de entrega de datos (modo de reserva de recursos o modo *best-effort*). Cuando el modo con reserva de recursos es el seleccionado, la tasa de entrega es α_{ReR}. De forma similar, definimos la tasa de entrega en el modo *best-effort* como α_{BE}. Es importante remarcar la dependencia de α_{BE} con la carga de la red. Obviamente, una red muy cargada entregará datos al cliente a una tasa menor que una red infrautilizada. Aquí, α_{BE} se define a nivel de la memoria del cliente (bits/s efectivos que entran en la memoria), no a nivel de transporte, por lo que se ve afectada por las pérdidas debidas a errores en la transmisión, descartes de paquetes en los *routers* y por las posibles retransmisiones.

La Fig. 4 muestra la variación en el tiempo de la cantidad de datos en la memoria del cliente. Como se observa, de 0 a t_0 es el primer periodo de tiempo en el que se usa reserva de recursos (al principio de la transmisión) y $t_1 - t_0$ es el primer periodo de tiempo usando el modo *best-effort*. Inicialmente, la memoria está vacía. Parece lógico enviar los primeros paquetes con reserva de recursos para proporcionar de forma rápida la información al cliente. En este tiempo, la cantidad de información almacenada en la memoria del cliente aumentará como si se llenara con una tasa de entrada α_i. Esta tasa (3) es la diferencia entre la tasa de entrega cuando se reservan recursos (α_{ReR}) y la tasa de lectura del cliente (α_r).

$$\alpha_i = \alpha_{ReR} - \alpha_r \tag{3}$$

Cuando los datos almacenados alcanzan *Max*, el modo de entrega de datos pasa a *best-effort*. Similarmente, en este periodo, la cantidad de información decrecerá si $\alpha_{BE} < \alpha_r$, por lo que la memoria parecerá que se vacía con una tasa de salida α_o. De nuevo, esta tasa (4) es la diferencia entre la tasa de entrega en *best-effort* y la tasa de lectura del cliente. Más tarde, cuando el nivel de datos almacenados llegue a *Min*, el modo de entrega cambiará otra vez a reserva de recursos.

$$\alpha_o = \alpha_{BE} - \alpha_r \tag{4}$$

Para determinar como afecta la carga de la red al comportamiento de la memoria, definimos la carga vista por el cliente (ρ) como,

$$\begin{cases} \rho = 1 - \dfrac{\alpha_{BE}}{\alpha_r} & \alpha_{BE} \le \alpha_r \\ \rho = 0 & \alpha_{BE} \ge \alpha_r \end{cases} \tag{5}$$

La ecuación (5) define una carga percibida de 0 cuando el modo *best-effort* es suficiente para mantener la memoria llena. Por el contrario, una carga percibida de 1 indica que no llega ningún paquete durante este periodo.

3.2.3 Criterio de Minimización

El tiempo total en que se reservan recursos se minimiza si el último paquete de información enviado al cliente llega a la memoria cuando el nivel de datos está en su umbral *Min*. Para demostrar esto basta con observar la Fig. 4. En ella, se muestra un ejemplo en que se da esta situación. En este caso, el tiempo en que se reservan recursos es t_0. Si *Max* es mayor, t_0 también lo será. Por otro lado, si *Max* es menor, t_0 será menor también, así como el intervalo de *best-effort* (t_1- t_0), por lo que la cantidad de datos entregados en este periodo decrecerá, incrementando la cantidad de datos enviados con reserva de recursos.

Si sólo hay una renegociación entre el modo de reserva de recursos y el *best-effort*, el valor óptimo de *Max* será el más grande (Max_0). Pero hay otros valores de *Max* que minimizan el coste aumentando el número de renegociaciones y reduciendo la memoria total usada. En la Fig. 5 se muestra el caso en el que el modo de reserva de recursos se utiliza tres veces. Obviamos la demostración analítica, ya que intuitivamente es fácil de entender que los umbrales *Max* que cumplen el criterio de minimización son:

$$Max_n = Min + \frac{Max_0 - Min}{n+1} \tag{6}$$

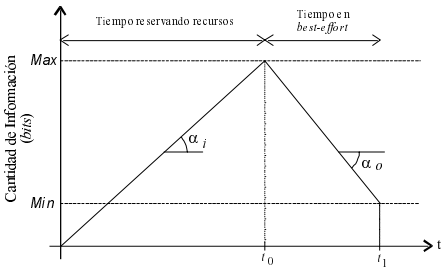


Fig. 4: Dinámica de los datos almacenados en la memoria del cliente.

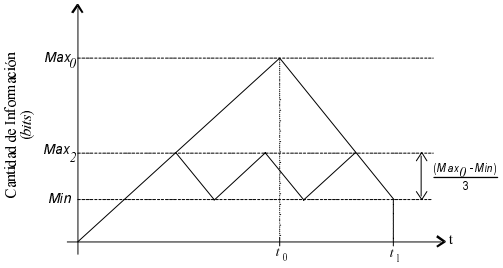


Fig. 5: Umbrales Max que cumplen el criterio de minimización del coste.

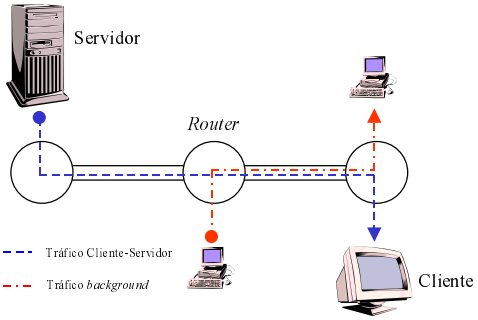


Fig. 6: Topología del Sistema.

donde *n* coincide con el número de veces en que se usa el modo de reserva de recursos menos uno. Además, se puede expresar Max_0 como en (7), ya que sabemos que el último paquete llega cuando el nivel de datos está en el umbral *Min*.

$$Max_0 = \frac{\alpha_r FS - \alpha_{BE}(FS - Min)}{\alpha_r(\alpha_{ReR} - \alpha_{BE})}(\alpha_{ReR} - \alpha_r) \qquad (7)$$

4 Análisis de Simulación

El sistema cliente-servidor diseñado para minimizar la transmisión de flujos semi-elásticos, ha sido implementado en el simulador NS-2 de carácter *freeware*. Esta implementación permite el estudio del sistema en un escenario controlado para validar la utilidad del análisis de la dinámica de los datos almacenados en la memoria del cliente.

La Fig. 6 muestra la topología del sistema. Como se puede ver, hay sólo tres nodos. El cliente accede a uno, mientras que el servidor se supone conectado a otro nodo distante. Utilizamos los protocolos TCP/IP para transmitir la información entre servidor y cliente de forma fiable. En el medio, un *router* permite añadir tráfico de fondo. En las simulaciones hemos usado una fuente de tráfico *best-effort* (random CBR en NS-2) entre los nodos *router* y cliente para simular la dinámica del tráfico de la red. Por otro lado, el protocolo de reserva de recursos utilizado en este escenario es el RSVP, por lo que el uso de los Servicios Diferenciados en la red troncal quedan fuera de nuestro ámbito de estudio. Hemos usado RSVP con WF^2Q como algoritmo de planificación de servicio en los nodos de la red ya que este algoritmo se comporta más eficientemente que WFQ con reservas de ancho de banda pequeñas (como es el caso de las reservas realizadas por RSVP para señalización). Estos nodos, además pueden perder paquetes IP debido a la carga creada por el tráfico de fondo.

En esta sección, se comentará el diseño del sistema cliente-servidor y también se mostrarán algunos de resultados de simulación obtenidos con dicho sistema.

4.1 Implementación del Servidor

La Fig. 7 muestra el diagrama de flujo que describe el comportamiento del servidor. Inicialmente, el servidor espera una petición por parte del cliente. Cuando la petición llega, el servidor activa una sesión RSVP y empieza la transmisión de paquetes de datos a la capa TCP para ser transmitidos. Estos paquetes se envían al TCP controlando la tasa para ajustarse al SLA negociado con RSVP y para no desbordar la memoria del cliente en redes infrautilizadas.

4.2 Implementación del Cliente

La Fig. 8 muestra el diagrama de flujo describiendo el comportamiento del cliente. Al principio, el cliente debe realizar una petición de información al servidor y esperar a que éste responda. Cuando los datos empiezan a llegar, el cliente comienza a leerlos inmediatamente ya que la reserva de recursos inicial asegura el llenado de la memoria. Además, el cliente debe empezar la gestión de la memoria a partir de la llegada del primer paquete. Esta gestión se realiza controlando cuándo la cantidad de información almacenada en la memoria del cliente alcanza el umbral *Max*. Cuando esto ocurre, el cliente cambia a *best-effort* indicando al RSVP el envío de un mensaje para liberar los recursos reservados, pero manteniendo la sesión RSVP. Seguidamente, el cliente controla cuándo el nivel de datos llega al umbral *Min*. De forma parecida, cuando esto sucede, el cliente cambia al modo de reserva de recursos indicando al RSVP la reserva de recursos para la sesión activa. Los mensajes RSVP se transmiten con garantía de retardo máximo, mediante la realización de una reserva de ancho de banda para realizar esta señalización.

Obviamente, al inicio de la transmisión es imposible fijar el nivel *Max* ya que no es posible conocer α_{BE} .

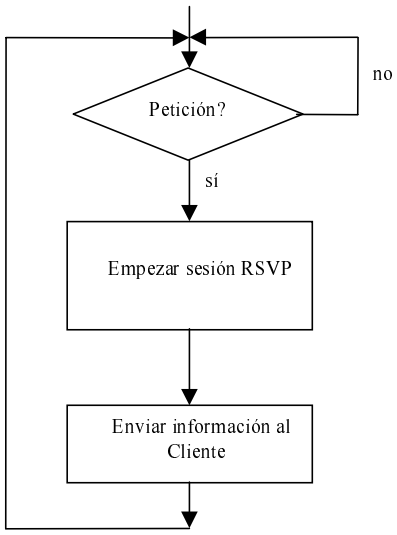


Fig. 7: Comportamiento del Servidor.

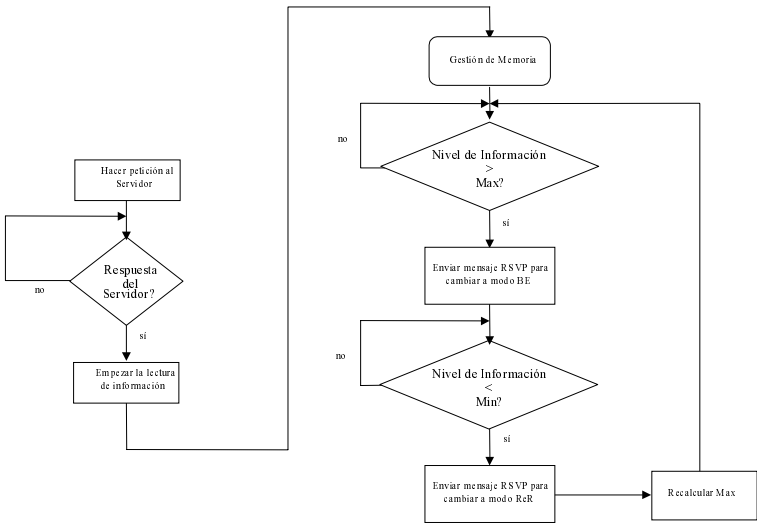


Fig. 8: Comportamiento del Cliente.

Esto complica la selección de un valor inicial (Max_{ini}) para *Max*. Por otro lado, Max_{ini} limita el rango de ρ donde el método es válido. Dicho de otra forma, si el valor de Max_0 es menor que Max_{ini} el sistema reservará recursos durante más tiempo del necesario. Por tanto, Max_{ini} debe cumplir (8) para evitar un coste residual.

$$Max_{ini} \leq \leq \left[FS - (1-\rho)(FS - Min)\right] \frac{\alpha_{ReR} - \alpha_r}{\alpha_{ReR} - \alpha_r(1-\rho)} \quad (8)$$

Además, α_{BE} debe estimarse en cada periodo de *best-effort* ya que puede variar. Por otra parte, la estimación de α_{BE} depende de la longitud de este periodo debido al comportamiento del TCP. Por todo ello, el valor de *Max* es recalculado en cada periodo de *best-effort* (Fig. 8) teniendo en cuanta el valor de *M*.

$$Max_n \leq M \Rightarrow n \geq \left\lceil \frac{Max_0 - Min}{MS - Min} - 1 \right\rceil \quad (9)$$

Finalmente, es importante mencionar que cuando este método se usa con dinámicas de red con varianza alta, puede ser difícil el cumplir el criterio de minimización. Luego, para prevenir esto y alcanzar lo más cerca posible el criterio de minimización es necesario reducir *Max* con respecto al calculado en los últimos cambios entre el modo *best-effort* y el de reserva de recursos. Esto previene la situación de recibir el último paquete en un nivel lejano a *Min*, pagando el coste de unas cuantas más (pocas) renegociaciones.

4.3 Implementación del Sistema Cliente-Servidor Óptimo

El objetivo de las simulaciones es el de comparar el comportamiento real del sistema cliente-servidor con el comportamiento analítico mostrado. Para compararlos, realizamos algunas simulaciones con niveles *Max* concretos (no se recalcula *Max* en el cliente) comparando el tiempo esperado en que se reservan

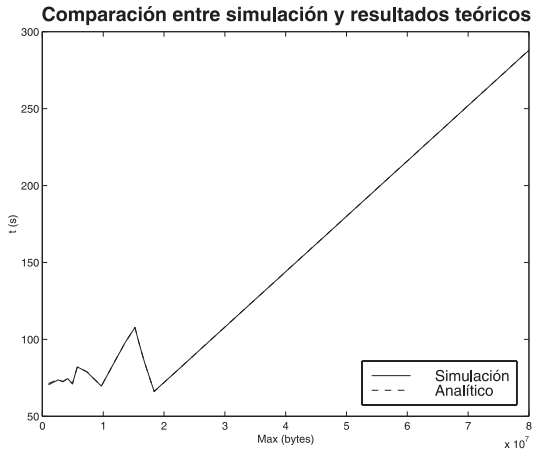


Fig. 9: Tiempo usando el modo ReR.

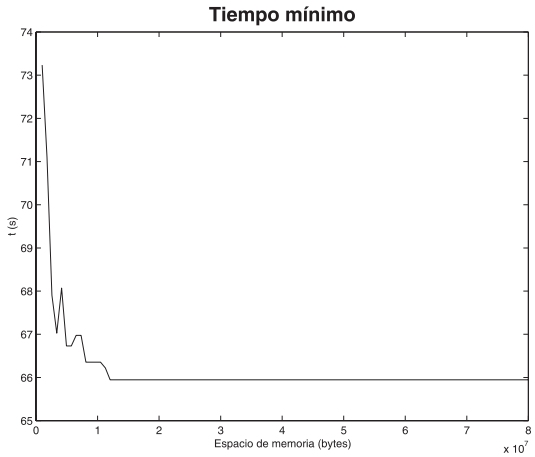


Fig. 10: Reducción del tiempo usando ReR para una transferencia con coste máximo de 150 Mbytes.

recursos usando estimaciones de α_{BE}, con los resultados de la simulación. Como se puede ver en la Fig. 9, los resultados son muy parecidos. Realmente, están superpuestos a excepción de los correspondientes a valores de *Max* pequeños. Por tanto, esto muestra que usando las expresiones analíticas junto con estimaciones de α_{BE}, es posible minimizar el coste de la transmisión.

Para ver el comportamiento del sistema cliente-servidor óptimo (*Max* se recalcula utilizando valores que minimizan el coste de la transmisión), realizamos varias simulaciones con diferentes parámetros. Aquí, mostramos los resultados para enlaces servidor-*router* y *router*-cliente de 5,8 Mb/s. El servidor envía un total de 150 Mbytes reservando 4 Mb/s en los periodos de reserva de recursos. La Fig. 10 muestra el tiempo en que se usa reserva de recursos (300 segundos corresponde al coste máximo) para *M* variando de 1 a 80 Mbytes. Como se puede observar, el tiempo decrece a medida que *M* aumenta, alcanzándose un valor mínimo constante cuando *M* es mayor que Max_0 (12 Mbytes). En este caso la eficiencia alcanza al 78%.

5 Conclusiones y Líneas Futuras

En este artículo, se han revisado los protocolos y tecnologías existentes para proporcionar QoS extremo a extremo en Internet y se ha descrito una de las arquitecturas propuestas que combina todos ellos. También, se han clasificado los flujos de datos que pueden circular por Internet según sus requerimientos en cuanto a reservas de recursos, esto es, flujos elásticos, inelásticos y semi-elásticos. Particularmente, hemos estudiado los flujos semi-elásticos, proponiendo un método eficiente para minimizar el coste de la transmisión de sus datos, reduciendo la cantidad de información enviada mediante reserva de recursos. Este método está gobernado por un sistema cliente-servidor. Concretamente, el cliente es capaz de minimizar el coste mediante el control de la ocupación de su memoria. Examinando cómo la información llega y conociendo los parámetros principales de la transmisión, el cliente puede minimizar el coste,

ajustando el umbral máximo de información en su memoria. Además este método tiene un criterio de minimización independiente del tráfico de fondo en la red que hace posible que el sistema consiga alcanzarlo para cualquier tipo de dinámica de red.

Como trabajo futuro, estamos actualmente interesados en la implementación de nuestro método manteniendo la complejidad del sistema baja y la compatibilidad con el comportamiento normal de Internet. También estamos interesados en estudiar la complejidad del servidor para controlar el servicio de un número arbitrario de flujos semi-elásticos.

Agradecimientos

Este trabajo ha sido parcialmente financiado por los proyectos de investigación SSADE (CICYT TEL99-0822), PRIME-IP (TIC2000-1734-C03-01) y FAR-IP (TIC2000-1734-C03-03).

Referencias

[1] The Network Simulator – ns-2, http://www.isi.edu/nsnam/ns/

[2] M. Hamdi, N. McKeown, "Scalable High-Speed Switches/Routers with QoS Support", IEEE Communications Magazine, December 2000, pp. 61.

[3] Stardust.com. White Paper – QoS Protocols & architectures. July 1999. http://www.qosforum.com.

[4] F. Cerdan, J. Malgosa-Sanahuja, J. Garcia-Haro, F. Burrull, F. Monzo-Sanchez, "Quality of service for TCP/IP traffic: an overview," in Proc. PROMS 2000, Cracow, Poland, October 2000.

[5] P. White, "RSVP and Integrated Services in the Internet: A Tutorial", IEEE Communications Magazine, May 1997.

[6] R. Braden, E., L. Zhang, S. Berson, S. Herzog, S. Jamin, "Resource ReSerVation Protocol (RSVP) – version 1 functional specification," Request for Comments 2205, IETF, Sept. 1997.

[7] A. Demers, S. Keshav, S. Shenker, "Analysis and Simulation of a Fair Queuing Algorithm", in Internetworking: Research and Experience, pp. 3-26, October 1999.

[8] J. C.R. Bennett and H. Zhang, WF^2Q: Worst-case Fair Weighted Fair Queueing, in Proceedings of INFOCOM '96, San Francisco, CA, March 24-28, 1996

[9] S. Shenker, C. Partridge, R. Guerin, Specification of Guaranteed Quality of Service, RFC 2212, September 1997.

[10] J. Wroclawski, Specification of the Controlled-Load Network Element Service, RFC 2211, September 1997.

[11] A. Mankin, F. Baker, B. Braden, S. Bradner, M. O'Dell, A. Romanow, A. Weinrib, L. Zhang. RSVP Version 1: Applicability Statement, Some Guideline on Deployment, RFC 2208, IETF, September 1997.

[12] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, An architecture for differentiated service. RFC 2475, IETF, December 1998.

[13] K. Nichols, S. Blake, F. Baker, D. Black. Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers. RFC 2474, IETF, December 1998.

[14] M. May, J.C. Bolot, A. Jean-Marie, C. Diot, Simple Performance Models of Differentiated Services Schemes for the Internet, in Proc. IEEE Infocom'99, New York, March 1999.

[15] D. Clark, W. Fang. Explicit allocation of *best-effort* packet delivery service. IEEE/ACM Transactions on Networking, vol. 6, no. 4, pp. 362-373, August 1998.

[16] S. Floyd, V. Jacobson, Random early detection gateways for congestion avoidance. IEEE/ACM Transactions on Networking, vol. 1, no. 4, pp. 397-413, August 1993.

[17] E. Rosen, A. Viswanathan, R. Callon, "Multiprotocol Label Switching Architecture", August 1999, <draft-ietf-mpls-arch-06.txt>.

[18] R. Yavatkar, D. Hoffman, Y. Bernet, F. Baker, "SBM (Subnet Bandwidth Manager): A Protocol for RSVP-based Admission Control over IEEE 802-style networks", RFC 2814, IETF, May 2000.

[19] A. Ghanwani, J. W. Pace, V. Srinivasan, A. Smith, M. Seaman, "A Framework for Integrated Services Over Shared and Switched IEEE 802 LAN Technologies", June 1999, RFC 2816, IETF, May 2000.

[20] H. Seaman, A. Smith, E. Crawley, J. Wroclawski, "Integrated Services Mappings on IEEE 802 Networks", RFC 2815, IETF, May 2000.

[21] Y. Bernet, R. Yavatkar, P. Ford, F. Baker, L. Zhang, K. Nichols, M. Speer, R. Braden, "Interoperation of RSVP/Int-Serv and Diff-Serv Networks", February 1999, <draft-ietf-diffserv-rsvp-02.txt>, Work in Progress.

[22] D. Awduche, L. Berger, D. Gan, T.Li, G. Swallow, V. Srinivasan, "Extensions to RSVP for LSP Tunnels", March 1999, Work in Progress.

[23] D. Awduche, D. Gan, T. Li, G. Swallow, V. Srinivasan, "Extensions to RSVP for Traffic Engineering", August 1998, <draft-swallow-mpls-rsvp-trafeng-00.txt>, Work in Progress.

[24] J. Heinanen, "Differentiated Services in MPLS Networks", June 1999, <draft-heinenen-diffserv-mpls-00.txt>, Work in Progress.

[25] K. Nichols, V. Jacobson, L. Zhang, "A two-bit differentiated services architecture for the internet", RFC 2638, IETF, July 1999.

[26] A. Terzis, L. Wang, J. Ogawa, L. Zhang, "A two-tier resource management model for the internet", in Proc. of Global Internet, December 1999.

[27] J. Boyle, R. Cohen, D. Durham, S. Herzog, R. Rajan, A. Sastry, "The COPS (common open policy service) protocol", RFC 2748, IETF, January 2000.

[28] J. Roberts, "Quality of Service Guarantees and Charging in Multi-service Networks", IEICE Transactions on Communications, May 1998.

[29] Stardust.com. White Paper – The Need for QoS. July 1999. http://www.qosforum.com.

[30] M. Postigo Boix, J. García Haro, M. Aguilar Igartua, "Transmisión Eficiente de Bloques en Tiempo Real sobre Redes IP", Proceedings of URSI 2000, Zaragoza, Spain, September 2000, pp. 405-406.

[31] C. Shahabi, M. H. Alshayeji, "Super-streaming: a New Object Delivery Paradigm for Continuous Media Servers," Journal of Multimedia Tools and Applications, vol.11, issue 1, May 2000, pp. 275-298.

Estudio de un mecanismo de control de congestión para comunicaciones punto a multipunto.

Marta Solera, Isabel Barbancho, JoséYufera, Sebastià Sallent
Departamento de Ingeniería de Comunicaciones. Universidad de Málaga.
Campus Universitario de Teatinos s/n. 29071 Málaga
E-mail: {msolera, ibp}@ic.uma.es
Departament d'Enginyeria Telemàtica. Universitat Politècnica de Catalunya.
Jordi Girona 1 y 3. Campus Nord, Mód C3, UPC. 08034 Barcelona
E-mail: {yufera, sallent}@mat.upc.es (1)

Abstract. *Reactive congestion control based on queue length is analysed for point to multipoint communications. Congestion control for multicast is more difficult than for unicast. The presence of multiple receivers, with different bottleneck bandwidths, makes it hard to define a target data rate to be used in the communication. To avoid the problem, the source should try to adapt to the slowest receiver. In this paper, we study the performance of a node which is subject to threshold-based feedback control policy by their average values and the fluid flow model. Also, we discuss the impact of two thresholds on the performance of the congestion control.*

1 Introducción

Las comunicaciones multipunto permiten a una o más fuentes transmitir información a un conjunto de usuarios. Desde el punto de vista del proveedor de servicios, la transmisión multipunto aporta una importante ventaja como es el ahorro de ancho de banda, sobretodo en aquellas redes de baja velocidad. Sin embargo, desde el punto de vista de usuario no aporta ningún valor añadido frente a la transmisión punto a punto. Por ello, la transmisión multipunto debe ofrecer las mismas características de estabilidad y accesibilidad que la transmisión punto a punto [1].

Debido al gran desarrollo de aplicaciones multipunto, como la distribución de audio y vídeo sobre Internet, se hace necesario el estudio de mecanismos de congestión que eviten la saturación de la red. El desarrollo de estos controles de congestión es mucho más complejo en comunicaciones multipunto que en punto a punto. En conexiones multipunto, los distintos receptores finales están conectados a la fuente a través de enlaces con capacidades distintas, lo que dificulta determinar la tasa a la que debe transmitir la fuente. Existen diferentes políticas para dar respuesta a este problema: adaptar la tasa de la fuente al receptor de menor capacidad; transmitir a una tasa determinada y eliminar del grupo multipunto a aquellos receptores que carezcan de los recursos necesarios; o dividir el tráfico generado por la fuente en diferentes flujos de distintas capacidades y permitir a los receptores asociarse a tantos flujos como recursos dispongan.

Otro punto a considerar en el desarrollo del control de congestión es conocer como responde el sistema. Para ello, es necesario establecer un mecanismo de realimentación que informe del estado de congestión de los receptores y elementos de la red. Debemos apuntar que en las comunicaciones multipunto, si la realimentación es demasiado lenta no será útil para conocer el estado de la red en un instante determinado, pero si es demasiado frecuente puede aparecer el problema de implosión de información de realimentación.

En la literatura podemos encontrar diversos estudios para el análisis de mecanismos de control de congestión reactivos para comunicaciones punto a punto. En [2] y [3] se analiza un control de congestión basado en dos umbrales a partir de un complejo análisis matemático.

En este artículo se va a abordar el estudio de un control de congestión simple para comunicaciones punto a multipunto. Un nodo intermedio reconoce congestión cuando detecta que la longitud de alguna de las colas de los enlaces que pertenecen a la conexión multipunto alcanza un umbral determinado que denominaremos H. El nodo congestionado informa a la fuente mediante una señal de control para que pare de transmitir. Mientras la fuente no envía tráfico a la red, las longitudes de las colas se van decrementando hasta alcanzar un umbral bajo que llamaremos L. En el instante en el que todas las colas tengan una longitud por debajo de L, el nodo informa a la fuente que puede volver a transmitir mediante una señal de control de activación.

En este trabajo se va a caracterizar en valores medios y mediante el modelo de fluidos un mecanismo de control de flujo basado en umbrales. Se determinará la probabilidad de pérdida de paquetes del sistema, el caudal y otros parámetros que caracterizan el control de congestión. Se estudiará cuales son los valores de los umbrales H y L más adecuados.

(1) Este trabajo ha sido parcialmente financiado por CICYT en el contrato TIC98-0495-C02-01.

La organización del trabajo es la siguiente: en la sección 2, se describirá el modelo que se ha utilizado para describir el control de congestión; en la sección 3, se determinarán los parámetros que caracterizan al sistema en valores medios, y se presentarán algunas gráficas que ilustran el comportamiento del mecanismo de congestión; seguidamente, en el apartado 4, se analiza el sistema mediante el modelo de fluidos. A continuación se estudiará la validez del modelo y por último se presentarán las conclusiones.

2 Descripción del modelo

El modelo que se va a considerar para caracterizar el sistema de control de congestión se presenta en la figura 1. Está compuesto por una fuente y un nodo intermedio con N colas de longitud finita e igual a B paquetes. Las colas se sirven a diferentes velocidades dependiendo de la capacidad del enlace de salida. Los paquetes llegarán al nodo después de un tiempo igual al tiempo de transmisión más el tiempo de propagación. Al llegar al nodo los paquetes se duplicarán y se encolarán si los servidores de salida están ocupados.

Para simplificar el sistema se va a considerar que el control de congestión está gobernado por el receptor de menos recursos, es decir, el nodo envía la señal de control de inactivación cuando la longitud de la cola del enlace más lento alcanza el umbral H y envía la señal de control de activación cuando la longitud de la cola del enlace de menos ancho de banda alcanza el umbral L. De esta manera, minimizaremos la probabilidad de pérdidas. La fuente genera tráfico a una tasa media de V paquetes/s y el enlace de capacidad V*C paquetes/s, con C<1, representa al enlace de menos recursos involucrado en la conexión punto a multipunto. D representa el tiempo de ida y vuelta entre la fuente y el nodo.

Para simplificar el análisis vamos a suponer que $(H-L)/VC > D$. Esta hipótesis es razonable cuando la carga de tráfico es alta, lo que coincide con la región de interés para el estudio de la congestión. Con esta condición, el comportamiento de la ocupación de la cola queda ilustrado en la figura 2. El sistema pasa cíclicamente por cuatro fases. Durante los periodos I y IV, la fuente está inactiva, y por lo tanto, la longitud de la cola se decrementa. Cuando alcanza el umbral L se envía una señal de control para la activación de la fuente. En este instante $X_{sig\text{-}on}$, el sistema pasa de la fase IV a la fase I. Durante un retardo de D, continúa la disminución de la longitud de la cola hasta la llegada del primer paquete generado por la fuente, momento $X_{lleg\text{-}on}$ en el que el sistema pasa de la fase I a la fase II. En los periodos II y III, la fuente genera tráfico. Cuando la cola alcanza el umbral H se envía la señal de control de inactivación a la fuente, instante $X_{sig\text{-}off}$ que coincide con el paso de la fase II a la fase III, pero durante un tiempo D todavía se reciben paquetes, instante $X_{lleg\text{-}off}$.

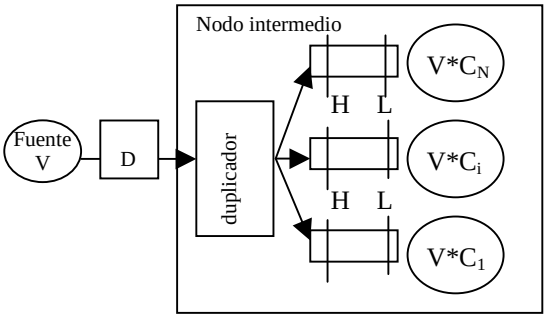


Figura 1. Modelo analítico

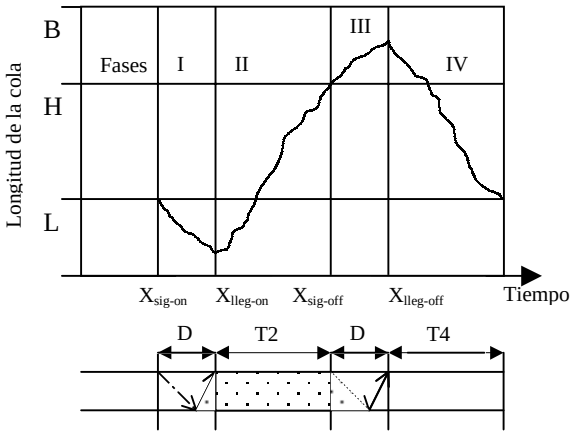


Figura 2. Comportamiento de la cola del nodo intermedio

3 Análisis en valores medios del sistema

Para la caracterización del sistema determinaremos la duración del ciclo, definido como la suma de la duración de las cuatro fases, la probabilidad de pérdidas del sistema y el caudal.

De la definición del sistema, se obtiene que el número medio de paquetes al inicio de la fase II es:

$$X_{sig-on} = L \qquad (1)$$

$$X_{lleg-on} = max(L - VCD, 0) \qquad (2)$$

Por lo tanto la duración del tiempo de actividad T2 queda determinado cuando el número de paquetes llega a H:

$$T2 = \frac{H - max(L - VCD, 0)}{V(1-C)} \qquad (3)$$

La fase III tiene una duración de D. Es en este periodo donde pueden producirse pérdidas. En valores medios la probabilidad de pérdidas puede calcularse como:

$$P_B = \frac{max(H + DV\ (1-C) - B, 0)}{V(T2 + D)} \qquad (4)$$

Al comienzo del periodo IV, se tiene un número medio de paquetes igual a $X_{lleg\text{-}off}$,

$$X_{lleg-off} = H + (V(1-P_B) - VC)D \qquad (5)$$

por lo que la duración de este periodo es T4:

$$T4 = \frac{H + (V(1-P_B) - VC)D - L}{VC} \qquad (6)$$

La expresión del caudal se obtiene a partir de la duración media del ciclo T2+T4+2D y de (4):

$$T(H,L) = \frac{(T2+D)V(1-P_B)}{T4+T2+2D} \qquad (7)$$

3.1 Resultados

A partir de las expresiones obtenidas en el apartado anterior se va a estudiar el comportamiento del sistema en función de sus parámetros de una forma heurística. Se analizará como afecta en la probabilidad de pérdida y en el caudal el retardo de propagación, y como escoger los valores de los umbrales H y L para minimizar la probabilidad de pérdida y la frecuencia de envío de señales de control.

3.1.1 Impacto del retardo sobre la probabilidad de pérdidas

Con B, H, L y C fijados, la figura 3 muestra el efecto del tiempo de ida y vuelta D en la probabilidad de pérdidas P_B para distintos valores de carga. Como se podía preveer, un aumento del tiempo de propagación supone un aumento en la duración del periodo III, única fase del sistema donde se pueden producir pérdidas. Por lo tanto, durante más tiempo el sistema está trabajando en los límites de la cola. Desde otro punto de vista, suponiendo fijada la probabilidad de pérdidas a la que el sistema debe trabajar, se observa que el sistema puede cursar más tráfico si el tiempo de propagación es mayor.

3.1.2 Elección de los umbrales

Una elección adecuada de los valores umbrales L y H es importante para maximizar el caudal, minimizar las pérdidas y reducir el envío de señales de control. En la figura 4 se observa como el valor del caudal se mantiene constante hasta un determinado tiempo D en el que cae drásticamente. Esto es debido a que el umbral L está por debajo del valor VCD. En está situación, la cola permanece vacía durante algún tiempo en la fase I lo que hace disminuir el caudal. Por otro lado, un valor del umbral L mayor que VCD no aumenta el caudal, y sin embargo, es posible que genere el envio de más señales de control. Además puede inducir a mayores pérdidas en el sistema si esto nos lleva a aumentar el valor del umbral H.

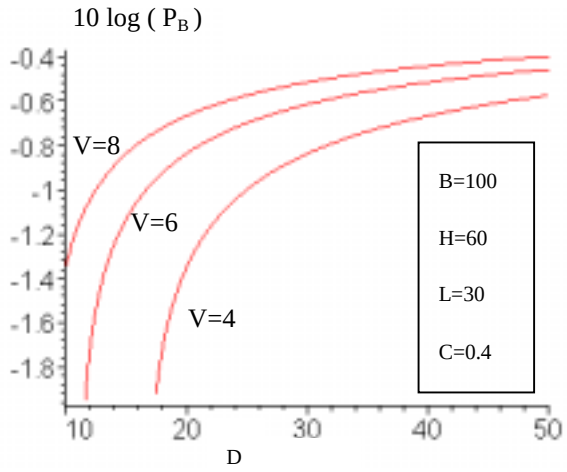


Figura 3. Probabilidad de pérdida en función del tiempo de propagación para varios valores de carga.

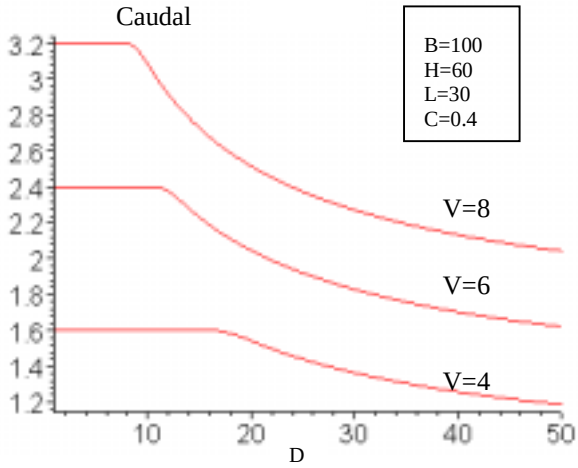


Figura 4. Efecto en el caudal del umbral L< VCD.

La elección del umbral H deberá ser lo más bajo posible para minimizar la probabilidad de pérdidas, llegando al compromiso entre la frecuencia de envío de señales de control, que vendrá dado por el tiempo de ciclo del sistema y las pérdidas. En las gráficas 5 y 6 pueden observarse estos comportamientos.

4 Análisis mediante el modelo de fluidos

Otra manera de describir el mecanismo de control de congestión es caracterizarlo como una fuente *on-off*, con un periodo de actividad T2+D, en el que la fuente genera V paquetes/s a tasa constante, y un periodo de inactividad de duración T4+D. Supondremos que el tiempo de permanencia en los estados activo, (estado 1), e inactivo (estado 0), sigue una distribución exponencial. Esto nos lleva a un modelo de nacimiento y muerte de dos estados. En la figura 7 está representado el modelo, donde λ y α son las tasas de transición y se calculan como la inversa del tiempo medio de permanencia en el estado.

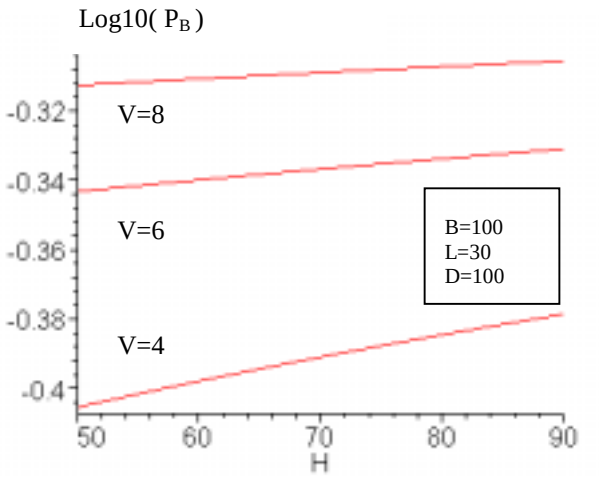


Figura 5. Probabilidad de pérdida en función del valor del umbral H.

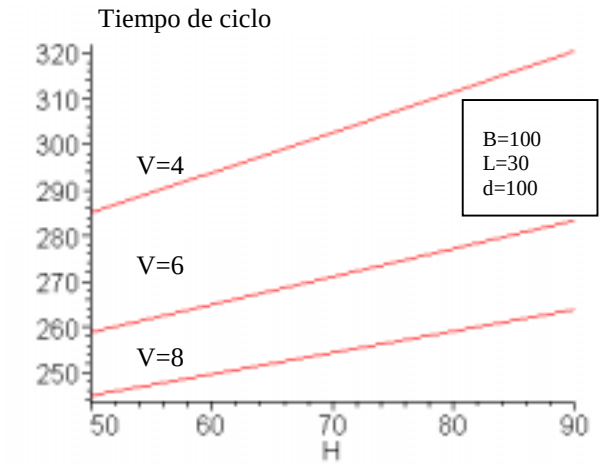


Figura 6. Tiempo medio de ciclo en función del umbral alto.

$$\lambda = \frac{1}{T4 + D} \tag{8}$$

$$\alpha = \frac{1}{T2 + D} \tag{9}$$

Π_o y Π_1 representan la probabilidad de que el sistema se encuentre en el estado 0 y 1 respectivamente, y se escriben como:

$$\Pi_o = \frac{\alpha}{\lambda + \alpha} \quad \Pi_1 = \frac{\lambda}{\lambda + \alpha} \tag{10}$$

La ecuación de balance que resulta del modelo descrito es:

$$\lambda \Pi_0 = \alpha \Pi_1 \tag{11}$$

A partir de la caracterización de la fuente y mediante el modelo de fluidos se va a estudiar el mecanismo de control de flujo descrito en la sección 2. Para ello, se define x como la variable aleatoria continua que describe el número de paquetes que llegan a la cola durante el periodo de actividad y αC como la capacidad normalizada respecto al número de paquetes que llegan en el periodo de actividad. La variable aleatoria que representa la longitud de la cola en paquetes la denominaremos m. La relación entre x y m vendrá dada por x=m*α/V. Además supondremos la cola infinita.

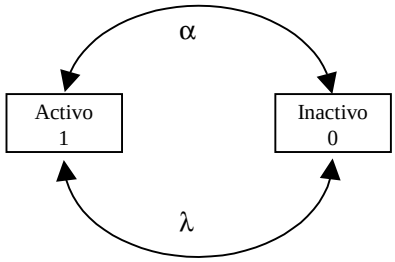


Figura 7. Modelo de dos estados: activo e inactivo.

Se define $F_i(t,x)$, con i=0,1, como la función de distribución de x en el instante t con el sistema en el estado i. A partir del análisis del modelo de la figura 7, podemos escribir:

$$F_0(t+\Delta t, x) = \alpha \Delta t F_1(t,x) + (1 - \lambda \Delta t) F_0(t, x + c\alpha \Delta t) + o(\Delta t) \tag{12}$$

$$F_1(t+\Delta t, x) = \lambda \Delta t F_0(t,x) + (1 - \alpha \Delta t) F_1(t, x - (1-c)\alpha \Delta t) + o(\Delta t) \tag{13}$$

Supondremos que $F_{-1}(\ )$ y $F_2(\ )$ son cero, y que $F_i(t,x)$ es continua y derivable. Desarrollando en serie de Taylor $F_i(t+\Delta t,x)$ y $F_i(t,x-\Delta x)$, y haciendo tender Δt a cero, nos quedan unas ecuaciones en derivadas parciales respecto a t y x. Si suponemos estacionariedad, $\partial Fi(t,x)/\partial t = 0$, entonces nos queda el sistema de ecuaciones siguiente:

$$-C\alpha \frac{dF_0(x)}{dx} = -\lambda F_0(x) + \alpha F_1(x) \tag{14}$$

$$(1-C)\alpha \frac{dF_1(x)}{dx} = \lambda F_0(x) - \alpha F_1(x) \tag{15}$$

La solución de este sistema es una suma de exponenciales, donde el cálculo de las constantes se determina a partir de las condiciones de contorno. Finalmente, y siguiendo el desarrollo de M. Schwartz en [4] obtenemos:

$$F_0(x) = \frac{\alpha}{\alpha + \lambda} - \rho (1-C) e^{-\frac{(1-\rho)(1+\gamma)}{(1-C)}x} \tag{16}$$

$$F_1(x) = \frac{\lambda}{\alpha + \lambda} - \rho C e^{-\frac{(1-\rho)(1+\gamma)}{(1-C)}x} \tag{17}$$

con $\rho = \frac{\lambda}{(\lambda + \alpha)C}$ y $\gamma = \frac{\lambda}{\alpha}$.

Para la caracterización del mecanismo de control de congestión, se ha de determinar la duración del ciclo del sistema y la probabilidad de pérdidas. El

tiempo medio de actividad de la fuente definido como T2+D es igual al obtenido en la expresión (3). Como la fuente genera V paquetes/s y la duración de este periodo está limitado entre m=L y m=H se obtiene la igualdad anterior. El cálculo de la duración del tiempo de inactividad, T4+D, vendrá dado a partir de imponer en la función de distribución de la ocupación de la cola, $F(m)=F_0(m) + F_1(m)$, la siguiente condición:

$$F(m \le L) = 1 - \rho\, e^{-\frac{(1-\rho)(1+\gamma)}{(1-C)}\frac{\alpha L}{V}} = \frac{D}{T4+T2+2D} \quad (18)$$

De aquí obtenemos T4 en función de T2 como:

$$T4 = \frac{H/VC}{\ln\left(\frac{D}{T2+D}\right) + \frac{H}{V(1-C)(T2+D)}} - D \quad (19)$$

El mecanismo de control de congestión sólo presenta pérdidas durante la fase III, por lo tanto la probabilidad de pérdidas se escribe como el producto entre la probabilidad de estar en esa fase por la probabilidad de pérdida condicionada a estar en la fase III.

$$P_B = \frac{D}{T2+T4+2D}\rho\, e^{-\frac{(1-\rho)(1+\gamma)}{(1-C)}\frac{\alpha(B-H)}{V}} \quad (20)$$

5.1 Validez del modelo

El análisis del mecanismo de control de congestión mediante el modelo de fluidos es válido para unos valores del tiempo de propagación que van desde $D1 \le D \ge L/VC$, donde D1 es el instante que minimiza la probabilidad de pérdidas F(m). En este rango, el análisis matemático se adecúa al funcionamiento del sistema. La probabilidad de pérdidas aumenta al aumentar la carga del sistema y el tiempo de propagación, como puede observarse en la figura 8. Al estudiar el caudal, definido en (7), en función del tiempo de propagación se observa que a partir de un valor de D igual a L/VC, el caudal se decrementa drásticamente (figura 9), comportamiento esperado si el umbral L se fija a un valor inferior a VCD.

6 Conclusiones

En este trabajo se ha analizado un mecanismo de control de congestión basado en la ocupación de la cola del enlace de menos recursos. Esta política que minimiza la probabilidad de pérdidas del sistema, y podía resultar viable en entornos homogéneos, es poco flexible y poco dinámica para entornos heterogéneos.

Se ha obtenido a partir del estudio de los valores medios del sistema, un ágil y sencillo mecanismo para analizar las características del sistema y seleccionar los valores de los umbrales adecuados.

Por último, se ha utilizado el modelo de fluidos para analizar el sistema como una fuente con dos estados: activo e inactivo.

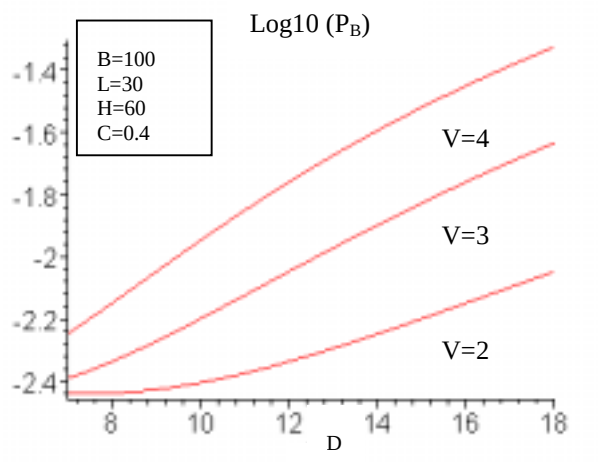


Figura 8. Probabilidad de pérdidas en función del tiempo de ida y vuelta para el modelo de fluidos.

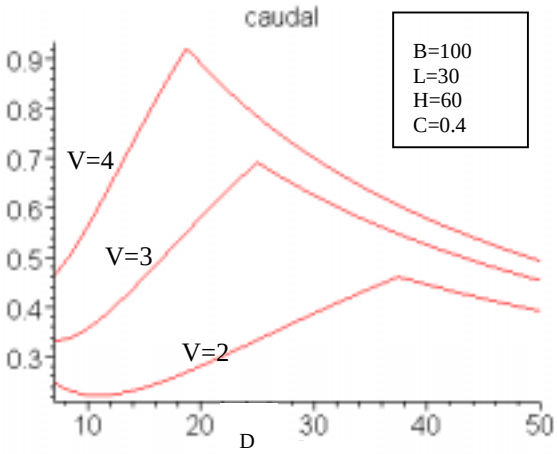


Figura 9. Caudal en función del tiempo de ida y vuelta para el modelo de fluidos.

Referencias

[1] C.Diot, B.N.Levine, B.Lytes et al, "Deployment Issues for the IP Multicast Service and Architecture", IEEE Network, January 2000.

[2] Y.T. Wang, B. Sengupta, "Performance Analysis of a feedback Congestion Control Policy Under Non-Negligible Propagation Delay", Proc. ACM SIGCOMM (1991). Pag. 149-157.

[3] M.Kato, Y.Oje, M.Murata, H. Miyahara, “ Performance Analysis of reactive congestion control based upon queue length threshold values”, Performance Evaluation, vol.29, nº 2, March 1997.

[4] M. Schwartz, “Broadband Intregated Networks”, Prentice Hall, New Jersey, 1996.

Reducción de la congestión y de la variación del retardo en redes Ethernet

J. M. Arco, D. Meziat,
Departamento de Automática Universidad de Alcalá
Escuela Politécnica 28871 Alcalá de Henares
Teléfono: +34 91 8856627
{jmarco, meziat@aut.alcala.es}

***Abstract**. This paper proposes a new scheduling algorithm called PTPV that reduce congestion, maximum delay and delay variation in Ethernet netwoks. This work is embraced into a line of research that intends to provide Ethernet users with the quality of service of ATM. To achieve this, an architecture called "Cells in Frame" (CIF) is used, which allows ATM cells to be carried in Ethernet frames. An implementation of this architecture has been made using the PTPV algorithm to reduce delay variation. Some measures have been carried out in order to check the cost and advantages of this algorithm.*

1 Introducción

Muchas aplicaciones multimedia necesitan que la red dé garantías de calidad de servicio para un correcto funcionamiento. Estas garantías se basan en que ciertos parámetros estén acotados. Los parámetros que habitualmente se usan son el ancho de banda, el retardo, la variación del retardo y la tasa de pérdidas.

Este trabajo, se engloba dentro de una línea de investigación que pretende mejorar la interconexión de ATM con redes de área local clásicas como Ethernet. Una de las posibilidades que hay para realizar dicha interconexión se basa en la arquitectura CIF [1]. La topología genérica de dicha arquitectura, se ilustra en la figura 1.

En el funcionamiento de CIF, el sistema final envía células ATM dentro de una trama Ethernet, además de una cabecera específica. El conmutador CIF extrae las células ATM de las tramas Ethernet y las envía a la red ATM.

Una limitación importante en este escenario, es que el retardo y la variación del retardo, no están convenientemente acotados en los segmentos Ethernet, para las necesidades de las aplicaciones multimedia, por lo que la QoS entre los usuarios finales no es la adecuada.

Con la inclusión del algoritmo de Paso de Testigo con Periodo Variable (PTPV) se reduce el retardo máximo y su variación en los segmentos Ethernet, por lo que finalmente se consigue soportar QoS entre usuarios.

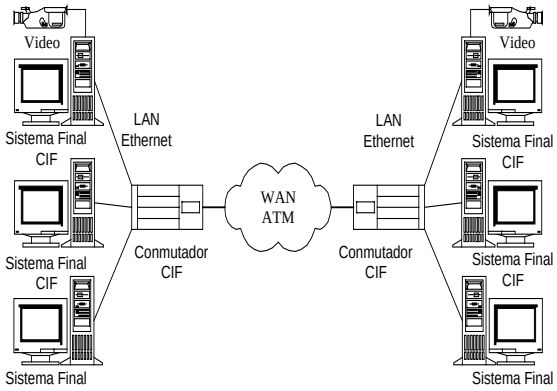


Fig. 1. Topología de red.

Para contrastar el funcionamiento de PTPV, se ha implementado en Linux dentro de la arquitectura CIF. También se han realizado unos ensayos, para ver como afecta a los parámetros relativos al tráfico en tiempo real.

El resto del artículo está divido en los siguientes apartados. En la siguiente sección se hace un planteamiento del problema, describiendo las soluciones propuestas y la que hemos adoptado en nuestro caso. En la sección 3, se muestra la implementación realizada del algoritmo PTPV. En la sección 4, se exponen las pruebas realizadas.

2 El algoritmo PTPV

En las redes Ethernet clásicas (compartidas) no se puede garantizar un límite superior para el retardo ni para la variación del mismo, debido a la existencia de colisiones y al algoritmo que se usa para solucionarlas. Este algoritmo, conocido como back off [2], obliga a esperar un número aleatorio de ranuras a cada uno de los que hayan colisionado. Las ranuras son de 2 veces el retardo máximo de transmisión, 2*56 microsegundos. Tras la primera colisión, se espera un número aleatorio entre 0 y 1

ranuras; tras la segunda se espera un número aleatorio entre 0 y 3; tras la enésima colisión se espera un número aleatorio comprendido entre 0 y 2^n-1, si n es menor o igual a 10; entre 0 y 1023 ranuras si n está comprendido entre 11 y 16. El algoritmo finaliza tras 16 colisiones descartando la trama. En teoría una trama puede esperar entre 952 y 459.256 microsegundos (es decir entre 17 y 8201 ranuras) antes de ser descartada. En simulaciones realizadas el tiempo medido está entre 75.000 y 400.000 microsegundos [3]. Cuando hay poco tráfico el algoritmo funciona correctamente, pero con mucho tráfico el back off favorece que vuelva a transmitir la estación que lo hizo en último lugar, debido a que empezará esperando un número aleatorio de ranuras entre 0 y 1 cuando el resto esperarán un número aleatorio mayor, es decir, tendrán que esperar más para volver a transmitir. Por lo tanto, con mucho tráfico, es muy probable que la estación que transmitió en último lugar lo haga de forma continua, esto se conoce como el efecto "captura". Esto provoca el incremento del retardo en el resto de estaciones, la variación del mismo, la pérdida de paquetes y de ancho de banda.

Para evitar este problema se han propuesto diversas soluciones que se van a analizar viendo su aplicación al escenario de trabajo.

Una opción es utilizar Ethernet full-duplex, con ésta técnica se puede transmitir y recibir al mismo tiempo, desapareciendo por tanto las colisiones. Tiene el inconveniente de que todos los elementos (adaptadores y concentradores) deben ser full-duplex, circunstancia que no se da en nuestro escenario.

Otra opción es la de utilizar un conmutador Ethernet con un control que evite que el tráfico del sistema final, supere el 50 % cuando hay mucha carga, con lo que desaparecería el efecto captura. Esto se podría hacer modificando el conmutador para que transmita antes de que finalice el tiempo de guarda (que se produce después de cualquier transmisión) del sistema final. De esta forma se evita que el sistema final pueda transmitir y no hay colisiones. Esto implica realizar un cambio en el conmutador, probablemente en el hardware del mismo, con lo que desde el punto de vista de nuestra investigación es inabordable.

Una idea similar utiliza la tecnología PACE de 3Com modificando el algoritmo back off en el conmutador [4]. Además PACE permite diferenciar dos tipos diferentes de tráfico a nivel Ethernet, aprovechando un bit del campo de dirección, aunque con poca granularidad.

Estas dos últimas soluciones tienen la ventaja de que no hay que cambiar la tarjeta de red del sistema final.

IsoEthernet (Isochronous Ethernet) es una variación de la Ethernet para soportar tráfico en tiempo real. Fue propuesta por el comité IEEE 802.9 con la idea de soportar en el mismo cable UTP la transmisión de datos de una LAN con varios canales isócronos tipo B de RDSI. No se ha sido tenido en cuenta, debido a que requiere nuevos adaptadores de red y conmutadores [5].

Rether (Real Time Ethernet) es una propuesta para dotar de calidad de servicio a Ethernet [6]. Se basa en la utilización de un testigo, de funcionamiento similar al del Token Ring [7]. Rether tiene dos modos de funcionamiento. El normal y el modo Rether al que se pasa cuando alguna estación quiere transmitir en tiempo real. Las estaciones Ethernet deben poseer el testigo (cuando las estaciones lo demandan) para poder transmitir desapareciendo por tanto las colisiones. El testigo es utilizado primero por las estaciones con tráfico más prioritario y después pasa al resto de estaciones. Una de las ventajas de Rether es que no requiere grandes cambios en la red Ethernet existente, tan solo el incluir driver especial en cada estación. Tiene el problema de la escalabilidad para varios segmentos Ethernet y su relativa complejidad de implementación.

Otras iniciativas para dotar a Ethernet de QoS son el estándar IEEE 802.1p, que es una extensión del 802.1D estándar del funcionamiento de los puentes, para tratar con distintas prioridades al tráfico Ethernet. Esta solución implica el uso de puentes Ethernet, que por su escaso uso y la necesidad de emplear un conmutador CIF específico se ha descartado. Otro estándar es el 802.1Q que definen el funcionamiento de las VLANs, que además pueden soportar QoS. En este caso se ha descartado por que se necesita nuevas tarjetas de red.

En el algoritmo PTPV propuesto utilizamos un testigo como en Token Ring para poder transmitir. Hay un período máximo de posesión del testigo, pero cuando no se tiene información que transmitir se devuelve inmediatamente para dar oportunidad al otro a transmitir. Nuestro algoritmo da más prioridad al tráfico de salida de la red que al de entrada, y de esta forma se ayuda a disminuir la congestión de la red. Para ello, se envía dentro del testigo información del tráfico encolado en el conmutador CIF, con lo que el sistema final, si es necesario, puede acortar su período para dar salida al tráfico del conmutador CIF.

En entornos Ethernet compartidos se puede garantizar la QoS al aplicar el PTPV, si está implementado en todas las máquinas y limitando el número de las mismas para evitar el aumento del retardo, por la espera hasta recibir el testigo. Por tanto, para su aplicación con CIF se ha considerado que en cada segmento existe sólo un sistema final.

3 Implementación del PTPV

La implementación se ha realizado dentro de una arquitectura CIF implementada con anterioridad [8]. Dicha implementación se ha hecho en el escenario mostrado en la figura 2, que es una simplificación de la red de la figura 1. La simplificación consiste en unir dos PCs, (que actúan de sistemas finales) mediante un único conmutador CIF. Aunque este no es el escenario final, puede considerarse válido para comprobar la provisión de QoS. Además este es un escenario real, que se da cuando se quiere transmitir con calidad de servicio en comunicaciones entre LANs. Se ha desarrollando la implementación de un sistema final y un conmutador CIF. El desarrollo se ha realizado sobre el sistema operativo Linux con la distribución Red Hat 4.2.

A la hora de realizar la implementación en Linux estándar hay tener en cuenta dos limitaciones. La primera es que puede existir un desfase entre el tiempo en el que se envía la trama desde el núcleo y el tiempo en el que se transmite la trama a nivel físico Ethernet. Este desfase es debido a los *buffers* de la tarjeta de red y al funcionamiento del bus PCI, que permite al núcleo transmitir varias tramas a velocidad superior a la de la línea, como se ilustra en la figura 3.

Para solucionar este problema, se utiliza un contador de tiempo virtual donde se simula el tiempo que transcurre en la línea. Inicialmente, cuando se recibe el testigo se iguala el tiempo virtual con el actual, se calcula cuando se debe devolver el testigo y se transmite la primera trama incrementando el tiempo virtual con la duración de la misma. Después cada vez que se transmite una trama, si el tiempo actual es menor que el virtual (lo que indicaría que el núcleo transmite a más velocidad), se incrementa este último con el tiempo de transmisión de la trama. En el caso contrario, lo que ha sucedido es que el sistema operativo ha ido más lento que la tarjeta de red y el tiempo virtual se actualiza con el tiempo actual. El testigo se devuelve (enviando los bytes encolado en ese momento) cuando el tiempo virtual supera el límite calculado.

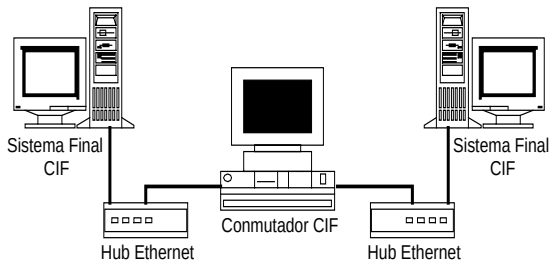


Fig. 2. Escenario considerado.

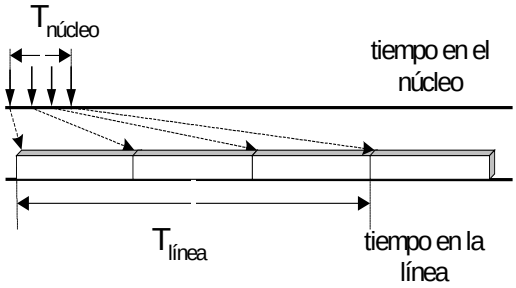


Fig. 3. Diferentes escalas de tiempo.

La segunda limitación se da cuando el sistema operativo se vea ralentizado pudiendo estar mucho tiempo hasta que devuelva el testigo, por lo que es necesario poner un temporizador que acote el período máximo de posesión del testigo. Interesa un período bajo, ya que en el peor caso, un paquete puede sufrir un retardo adicional de un período, hasta que le devuelvan el testigo y puedan ser transmitido. La limitación está en que los temporizadores en Linux son como mínimo de 10 milisegundos, que es el que se ha adoptado. Este retardo adicional es tolerado por la mayoría de las aplicaciones multimedia [9].

La implementación descrita anteriormente es la del conmutador, la del sistema final es lo mismo salvo que para dar más prioridad al tráfico que la red debe entregar, al recibir un testigo si hay más bytes encolados en el conmutador que el sistema final, entonces se toma como límite del período el tiempo que se tarda en transmitir los bytes encolados en el sistema final en ese momento.

El testigo se devuelve cuando sucede alguno de los tres siguientes supuestos, expira el temporizador, el tiempo virtual alcanza el límite superior del período (10 milisegundos) o cuando no haya tramas esperando a transmitir.

3.1 Funcionamiento del núcleo

En este apartado veremos como funciona la recepción y transmisión de paquetes en Linux y los cambios que se han efectuado para poder implementar el algoritmo. Cuando se recibe una trama se realiza una serie de tareas llamando a varias funciones que se muestran en la figura 4.

Al recibir una trama, la tarjeta de red interrumpe a la CPU. El tratamiento de la interrupción se hace en *xx_interrupt()*, donde *xx* son dos caracteres que dependen de la tarjeta de red, en esta función se determina el tipo de interrupción. En este caso llama a *xx_receive()*, que copia el paquete de la tarjeta de red al núcleo en una estructura *sk_buff*, extrae el campo tipo de la cabecera Ethernet y suprime la misma, por último llama a una función llamada *netif_rx()*. Está función encola el paquete en una cola central (llamada *backlog*) en donde se encolan todos los paquetes que hayan llegado por cualquier interfaz.

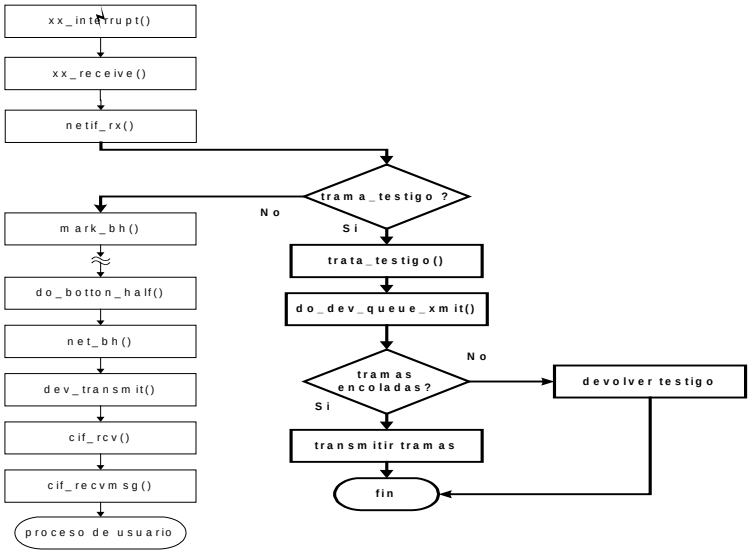


Fig. 4. Proceso de recepción

Por motivos de eficacia el sistema operativo Linux divide el tratamiento de las interrupciones en dos partes, una primera parte prioritaria, llamada *top-half* y en una segunda parte, llamada *bottom-half*, que se encarga de ejecutar las tareas menos urgentes y que requieren más tiempo.

El tratamiento *top-half* de recepción del paquete, termina dejando aviso con *mark_bh(NET_BH)*, para que cuando el núcleo pueda se realicen el resto del tratamiento de la interrupción.

El núcleo regularmente realiza el resto del tratamiento de las interrupciones mediante *do_bottom_half()*. En nuestro caso, al quedar pendiente el tratamiento de la interrupción de red, llamará a la rutina *net_bh()*, que comprueba si hay paquetes pendientes de transmisión en la cola de cualquier dispositivo de red. A pesar de estar tratando la llegada de un paquete, se aprovecha para ver si hay algún paquete pendiente, con objeto de reducir el tiempo medio de transmisión. Si hay un paquete se transmiten mediante *dev_transmit()*. Después *net_bh()* procesa el siguiente paquete de la cola *backlog* y en función del protocolo al que pertenezca el paquete, lo entrega a la entidad de red correspondiente. En nuestro caso se entrega a *cif_rcv()* perteneciente a la entidad CIF. En esta función comprobamos si el mensaje debe ser entregado a un proceso de usuario o debe ser retransmitido por otro interfaz.

Nosotros hemos modificado el proceso de recepción introduciendo dentro de *netif()*, la función *trata_testigo()* que es llamada si la trama recibida es un testigo. En este caso, lo primero que se hace es recuperar la cabecera Ethernet que había sido dejada atrás anteriormente. Hay que indicar, que las cabeceras asociadas a una trama almacenados en un *sk_buff*, no se eliminan totalmente, dejan de considerarse al apuntar los punteros a la siguiente cabecera [10]. Tras recuperar la cabecera, se intercambian las direcciones de origen y destino Ethernet para cuando haya que devolver el testigo. Por último, termina la función y con ella el proceso de recepción, llamando a *do_dev_queue()* donde el testigo será encolado empezándose a ejecutar el algoritmo PTPV, transmitiéndose las primeras tramas encoladas.

Otra modificación, que afecta sólo al conmutador CIF, es que al recibir la trama por un interfaz y tener el testigo en el otro interfaz *dev_transmit()* intentará transmitirla, se ha modificado para que no la transmita.

En el proceso de transmisión, es mostrado en la figura 5 en la parte que no está en negrita. Al enviar un mensaje de usuario, este llega a las funciones *cif_sendmsg()* y *cif_send()* donde se realizan diversas tareas CIF [8]. Después llaman a *do_dev_xmit()* que realiza el encolado de los paquetes. Con *dev->hard_start_xmit()* se entrega el paquete al driver para su transmisión. La modificación que se ha hecho es que si no tenemos el testigo no intenta transmitir.

Si tenemos el testigo y la tarjeta de red está libre, se envía el paquete. En caso contrario, se encola el paquete y se dan instrucciones a la tarjeta, para que cuando esté libre avise interrumpiendo.

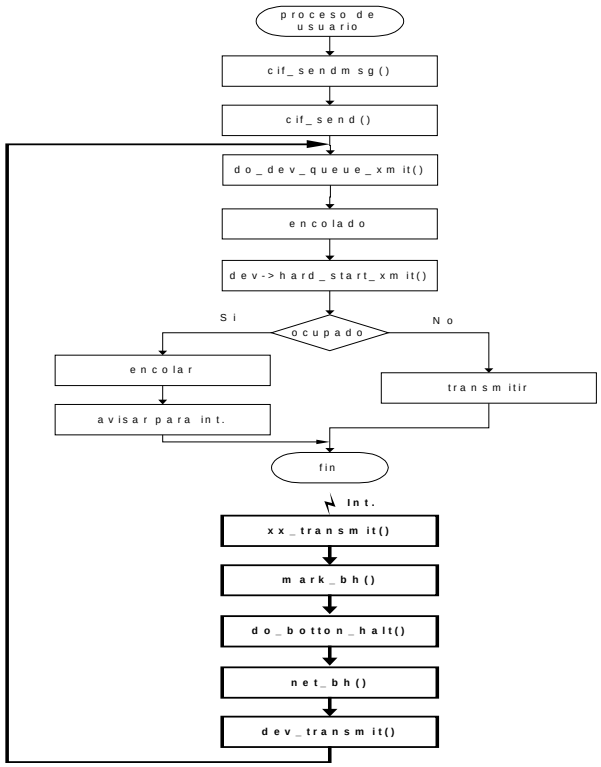


Fig. 4. Proceso de *retransmisión*.

El proceso de atención a las anteriores interrupciones se denomina *retransmisión* y está indicado en la figura 5 en negrita. En este proceso se ejecuta en primer lugar *xx_interrupt()* y se deja aviso con *mark_bh(NET_BH)*, para que cuando el núcleo pueda se realicen el resto del tratamiento de la interrupción. Como sucedía en la recepción de un paquete se llama a *do_bottom_half()*, *net_bh()* y *dev_transmit()* que llama a *do_dev_queue_xmit()* para que se transmita un paquete. La modificación efectuada es que si no tenemos el testigo no se transmite.

4 Pruebas de validación

Se han realizado ensayos para comprobar el impacto producido al implementar el algoritmo PTPV, para ello se han medido parámetros relativos al tráfico en tiempo real, de *throughput*, de retardo, y de variación del retardo. Es necesario indicar, que se han desarrollado los programas para realizar dichas pruebas, ya que aunque existen varias aplicaciones que miden el rendimiento en red [11][12] no han podido utilizarse por estar diseñadas para protocolos de TCP/IP y en nuestro desarrollo no hemos utilizado dichos protocolos.

Los PCs utilizados en las pruebas tienen CPU a 266 MHz y 32 Mbytes de RAM.

Las pruebas se han repetido al menos 4 veces. Con éste número se ha comprobado que los resultados son bastante aproximados, ya que las desviaciones típicas son muy bajas, por lo que el intervalo de confianza puede considerarse muy estrecho para un margen de confianza elevado.

Los principales factores que afectan a las parámetros medidos son las siguientes: tamaño del mensaje, número de sesiones, sentido del tráfico, número de procesos, tipo de tráfico, tiempo de las pruebas. De los parámetros anteriores sólo el tamaño del mensaje es variable, el resto se fijan atendiendo a diversas razones que a continuación se describen.

Se han hecho pruebas con diferente número de sesiones y se ha observado que no influyen en los resultados.

El sentido del tráfico y el tipo de tráfico depende de cada prueba. El sentido del tráfico puede ser unidireccional o bidireccional. Lo habitual es que el tipo de tráfico, se haga emitiendo paquetes de forma periódica. El período es calculado a partir de la velocidad de emisión introducida por teclado. No se implementa un control de la velocidad en bucle cerrado.

El número de procesos hará que la máquina tenga más o menos memoria ocupada, pero no influye en la pérdida de velocidad que pueda introducir el algoritmo implementado, ya que el sistema al enviar mensajes, cambia entre un proceso de usuario y el núcleo, por lo que al retornar a un proceso de usuario en teoría da igual que sea el mismo o que sea otro diferente. Por tanto, se fija un

proceso emisor en un sistema final y el otro proceso en el receptor del otro sistema final.

Por último, el tiempo de las pruebas es el habitual de 20 segundos aproximadamente.

El tamaño del mensaje es de 40, 64, 128, 256, 512, 1024 y 1480 bytes, y cuando se fija vale 1480.

4.1 Pruebas de *throughput*

La prueba de *throughput* se hace para hallar la máxima velocidad de transferencia sin pérdidas. La pérdida de velocidad es debida fundamentalmente a los tiempos de espera hasta recuperar el testigo para transmitir. La tabla 1 muestra la evolución del *throughput* con el tamaño de usuario.

El algoritmo PTPV introduce unas pérdidas de *throughput* inferiores a 0,4 Mbps, lo que supone en la mayoría de los casos unas pérdidas inferiores al 5%.

4.2 Pruebas de retardo

La medida del retardo se ha hecho de forma indirecta midiendo el RTT entre dos sistemas finales CIF, conectados a través del conmutador CIF. En cuanto al tipo de tráfico generado ha consistido en mandar un paquete de una sesión y esperar la respuesta para mandar el paquete de la siguiente sesión.

La tabla 2 muestra los resultados de la variación del retardo con el tamaño del mensaje.

Las diferencias se mantienen más o menos constantes, como era de esperar, ya que la diferencia es debida, al igual que en la prueba de *throughput*, al tiempo de espera para tener el testigo y poder transmitir, siendo éste tiempo independiente del tamaño de los datos a transmitir. Estas diferencias se justifican teniendo en cuenta que el número de testigos que se intercambian en ausencia de tráfico, es de 3.975 testigos por segundo, lo que significa que se tarda 125 microsegundos en recibir el testigo procesarlo y devolverlo. Como en el viaje de ida y vuelta se transmite 4 veces la trama, en el peor caso, se incrementaría el tiempo en unos 500 microsegundos aproximadamente, como queda reflejado en los resultados.

Tabla 1: *Throughput* en función del tamaño.

Tamaño	Vmax. sin PTPV	Vmax. con PTPV	Diferencias
40	3,81	3,42	0,39
64	4,45	4,26	0,19
128	6,68	6,34	0,34
256	7,65	7,3	0,35
512	8,89	8,41	0,38
1024	9,28	9,05	0,23
1480	9,48	9,4	0,08

Tabla 2: Variación del retardo con el tamaño del mensaje.

Tamaño	Retardo con PTPV	Retardo sin PTPV	Diferencia
40	977	502	475
64	1120	624	496
128	1280	811	479
256	1740	1272	468
512	2477	2104	373
1024	4260	3851	409
1480	5720	5333	387

Hay que indicar que el retardo máximo, se reduce de 500 milisegundos a 10 milisegundos a costa de incrementar el retardo normal en 400/2 microsegundos, aproximadamente.

4.3 Pruebas de variación del retardo

Las variaciones del retardo son debidas a la red de transmisión y al sistema operativo que puede tardar mas o menos tiempo en atender la emisión o recepción de un mensaje.

Las variaciones del retardo en la parte de transmisión por Ethernet se deben a dos motivos, a las colisiones que se producen al intentar emitir a la vez el sistema final y el conmutador y a que para transmitir, la tarjeta de red debe esperar a terminar de recibir. Se han hecho varias pruebas con distintos sentidos de tráfico, con distintos tamaños y velocidades, observándose los mayores variaciones con tráfico bidireccional con los mensajes más grandes y velocidades máximas, ya que en estas condiciones se da la mayor probabilidad de colisión.

En las figuras 6 y 7 se muestra la variación del retardo a la máxima velocidad sin pérdidas sin y con PTPV respectivamente.

El valor medio medido sin PTPV es de 35.544 microsegundos este valor se reduce 6 veces (6.092 microsegundos) con testigo. En el caso de los valores máximos, 205.000 microsegundos sin testigo, se reduce 10 veces (20.000 microsegundos).

La variación del retardo cuando se utiliza el algoritmo PVPT se debe al sistema operativo puesto que no hay colisiones.

También calculamos y representado, figura 8, el *interarrival jitter*, un estadístico que se define en [13]. El *interarrival jitter* es promedio de la variación del retado de varios paquetes consecutivos. Si $D_{i-1,i}$ es la desviación del retardo:

$$D_{i-1,i} = (R_i - R_{i-1}) - (S_i - S_{i-1})$$

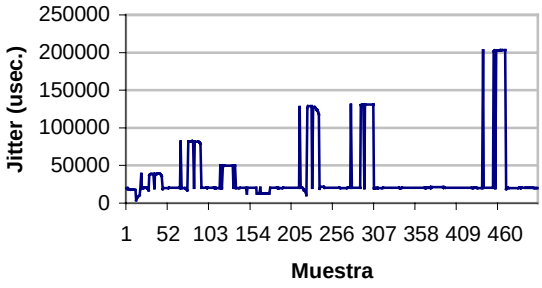


Fig. 6: Jitter sin PTPV a Vmax sin pérdidas (4,95 Mbps.)

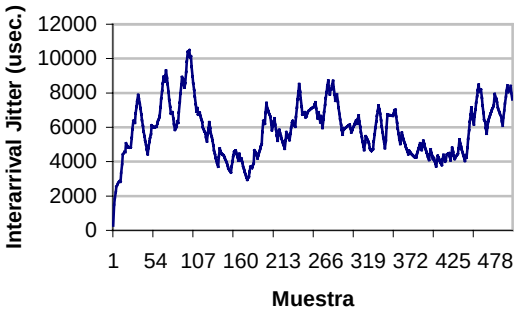


Fig. 8: Interarrival Jitter con PTPV a Vmax sin pérdidas (4,95 Mbps.)

donde *i-1*, *i* son dos paquetes consecutivos, con un tiempo de emisión de S_{i-1} y S_i respectivamente y un tiempo de recepción de R_{i-1} y R_i. El *interarrival jitter* es una función de $|D_{i-1,i}|$:

$$J_i = \frac{15}{16} * J_{i-1} + \frac{1}{16} * |D_{i-1,i}|$$

La figura 8 representa muestras del interarrival jitter para un tamaño de mensaje de 1480 bytes y 3,57 Mbps de velocidad de emisión en cada sentido.

Con objeto de ver la evolución de la variación de retardo cuando se transmite por encima de la máxima velocidad sin pérdidas, se ha hecho una prueba transmitiendo a 7 Mbps, cuyos resultados transmitiendo con y sin PTPV se muestran en la figuras 9 y 10.

El valor medio medido sin PTPV es 35.280 de microsegundos este valor se reduce 3,5 veces (10739 microsegundos) con testigo. En el caso de los valores máximos, 450.000 microsegundos sin testigo, se reduce 14,5 veces (31.019 microsegundos).

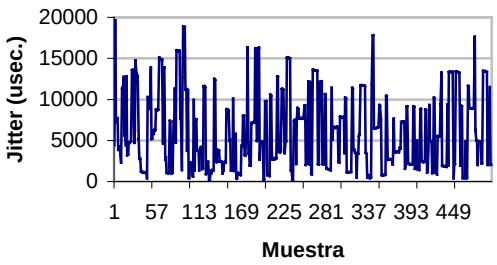


Fig. 7: Jitter con PTPV a Vmax sin pérdidas (4,95 Mbps.)

En esta prueba se puede observar que a mayores velocidades aumenta la contribución del sistema operativo a la variación del retardo, puesto que con el algoritmo PTPV sigue sin haber colisiones y ha aumentado la variación del retardo.

4.4 Análisis global de los resultados obtenidos

El algoritmo PTPV introduce unas pérdidas de *throughput* inferiores a 0,4 Mbps. El retardo adicional de los algoritmos es de baja magnitud. El retardo total de transmisión con la inclusión del PTPV es muy pequeño con relación al total recomendado, que es de 300 milisegundos.

El algoritmo PTPV consigue eliminar totalmente la componente de la variación del retardo debida a la red de transmisión Ethernet.

5 Conclusiones y líneas de futuros trabajos

En este articulo, se ha propuesto un nuevo algoritmo, el PTPV, para reducir el retardo la variación del retardo y la congestión en redes Ethernet que han sido implementados en Linux.

Se han diseñado unas pruebas, para ver el comportamiento de los parámetros relativos al

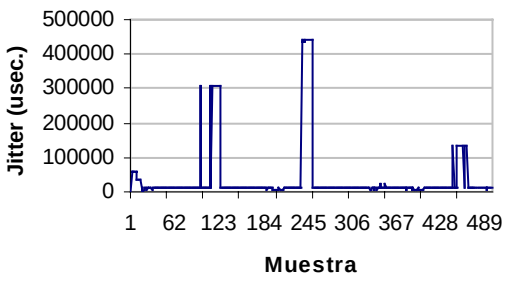


Fig. 9: Jitter sin PTPV 7 Mbps.

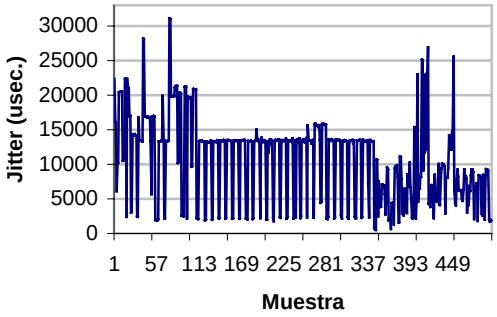


Fig. 10: Jitter con PTPV 7 Mbps.

tráfico en tiempo real. Para realizar dichas pruebas se han desarrollado los programas necesarios.

Las conclusiones más importantes del presente trabajo son las siguientes. El algoritmo PTPV reduce el retardo máximo y elimina la componente de la variación de retardo debida al funcionamiento de la red Ethernet, a costa de introducir unas ligeras pérdidas en el resto de parámetros.

Actualmente estamos trabajando en completar el escenario real, mediante la integración de una red ATM en el escenario de trabajo. De esta forma se podrán utilizar además de las aplicaciones ATM nativas las aplicaciones TCP/IP mediante el soporte de IP clásico.

Referencias

[1] J. M. Arco, A. Martínez, B. Alarcos, A. García, D. Meziat. “ATM Sobre Ethernet Mediante el Protocolo CIF". Libro de ponencias de las II Jornadas de Ingeniería Telemática JITEL´99, pp. 105-110, Septiembre 1999.

[2] A. Tanenbaum "Computer Networks, third edition", Prentice Hall, 1997.

[3] B. Whetten, S. Steinberg, D. Ferrari, "The Packet Starvation Effect in CSMA/CD LANs and a Solution", Proc. of IEEE Local Computer Networks Minneapolis, MN, pp. 206-217, October 1994..

[4] K. Chang "PACE Technology, Making Multimedia and Real-Time Networks Possible Today". http://mirror.dlut.edu.cn/3comEN/nsc/501316.html, 1997.

[5] F. Kuo, W. Effelsbeg, J. Garcia-Luna-Aceves “Multimedia Communications, Protocols and Applications". Prenticel Hall, 1998.

[6] C. Venkatramani, “The design, implementation and evaluating of RETHER: Areal-Time Ethernet Protocol". PhD. thesis, University of New York, Enero 1997.

[7] W. Stalling, “Local & Metropolitan Area Networks, Fifth Edition". Prentice Hall, 1997.

[8] J. M. Arco, “Propuesta de optimización de la interconexión de redes con calidad de servicio para aplicaciones multimedia".Tesis Doctoral, Universidad de Alcalá, Enero 1997.

[9] J. Rusell, capítulo "Multimedia Networking Performance Requirements", del libro "ATM Networks". Editores I. Viniotis and R.O. Onvural, Plenum, 1993, pp. 187-198.

[10] A. Cox, “Network Buffers and memory Management". Linux Journal, September 1996.

[11] R. Jones "Netperf: A Benchmark for Measuring Network Performance. Homepage". http://www.netperf.org/netperf/NetperfPage.html.

[12] B. Adamson. "The MGEN Toolset". http://manimac.itd.nrl.navy.mil/MGEN/.

[13] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications". Internet Request For Comment nº 1889, Enero 1996.

Evaluación de Rendimiento de Algoritmos de *Scheduling* en Redes ATM Basada en una Metodología Genérica.

Reinaldo Vallejos[1], Marta Barría[2], Alejandra Zapata[1], Isabel Martín[1,3].
[1]Departamento de Electrónica, UTFSM, Chile.
[2]Departamento de Computación, Universidad de Valparaíso, Chile.
[3]Departamento de Ingeniería Telemática, UPC, España.
reinaldo@elo.utfsm.cl, marta.barria@uv.cl, azb@elo.utfsm.cl, isabelm@mat.upc.es

Abstract.

In this paper a generic methodology is proposed for emulating and mathematically evaluating the performance of various scheduling algorithms in ATM networks. To emulate the operation of a given algorithm, a meta-algorithm named LDS (Load Dependent Scheduling) is proposed. The emulation of various algorithms using LDS is simple, and is achieved by specifying the appropriate LDS operating parameters. To evaluate the performance of a given algorithm X, the performance of LDS using the parameters that enable it to emulate that algorithm is evaluated. Thus, the performance measures obtained for LDS will correspond to those of algorithm X.

Key Words*: scheduling, ATM, Performance Evaluation.*

1 Introducción.

Para explotar eficientemente los recursos de una red es necesario atender simultáneamente a la mayor cantidad posible de usuarios, tratando de utilizar la menor cantidad posible de recursos. El uso eficiente de los recursos se logra usando multiplexación estadística. Sin embargo, debido a la conducta aleatoria de los usuarios, al utilizar multiplexación estadística existe la posibilidad de que en un determinado momento la tasa instantánea de llegada de información a un cierto conmutador de la red supere la tasa de servicio de éste. Para disminuir la pérdida de información que puede ocurrir en estos casos, los conmutadores implementan filas donde almacenan temporalmente el exceso de información recibida, para poder transmitirla posteriormente.

Debido a que una red ATM transporta tipos de tráficos que tiene diferentes requisitos de *Calidad de Servicio* (*QoS*), para almacenar el exceso temporal de información recibida, los conmutadores implementan filas distintas para los tráficos que poseen diferente *QoS*. Por este motivo surge la necesidad de diseñar e implementar una política de atención de paquetes (o política de *scheduling*), que es la encargada de determinar el orden en el cual se deben transmitir los paquetes de las distintas filas.

La política de *scheduling* implementada determina, en el largo plazo, el ancho de banda asignado a cada tipo de tráfico. La asignación de ancho de banda a cada tipo de tráfico es muy importante, ya que la *QoS* con que es atendida una conexión depende en gran medida de este ancho de banda [17].

En la literatura se han propuesto variados mecanismos de *scheduling* [2], [3], [5], [6], [11], [12], [15], [18], y continúan apareciendo nuevas propuestas. Cada vez que se propone un nuevo algoritmo de *scheduling*, los autores deben evaluarlo y compararlo con el resto de los algoritmos ya propuestos, con el objeto de decidir cuál de ellos constituye la mejor alternativa para ser implementada en las redes actuales. Para realizar este trabajo es necesario modelar el algoritmo propuesto y evaluar su desempeño. El análisis de desempeño se realiza utilizando herramientas matemáticas o de simulación. Dicho trabajo puede llegar a ser bastante largo y complicado matemáticamente. Con el objeto de simplificar el tipo de trabajo recién descrito, en este artículo se propone una metodología genérica que permite emular y evaluar el desempeño de variados algoritmos de *scheduling*.

La metodología consiste en emular el algoritmo de interés y luego evaluar el desempeño del algoritmo emulador. Para emular la operación de cualquier algoritmo de *scheduling* se propone un meta-algoritmo, que hemos denominado LDS: *Load Dependent Scheduling*. La emulación de un determinado algoritmo utilizando LDS es simple, y se consigue especificando apropiadamente los parámetros de operación de LDS. Luego, para evaluar el rendimiento de un determinado algoritmo X, se evalúa el desempeño de LDS para el caso en que éste opera con los parámetros que le permiten emular a X y, por lo tanto las medidas de rendimiento obtenidas para LDS corresponden a las del algoritmo X. En consecuencia, LDS ofrece la ventaja de poder evaluar el desempeño de cualquier algoritmo de *scheduling* a partir de un único conjunto de ecuaciones.

En lo que resta, este trabajo está organizado de la siguiente manera: En la sección 2 se describe el Meta-Algoritmo LDS. Luego, en la sección 3, a modo de ejemplo se muestra la emulación de algunos algoritmos de *scheduling* utilizando LDS. Posteriormente, en el sección 4 se realiza la

evaluación de rendimiento LDS, para el caso de un multiplexor que atiende dos tipos de tráficos diferentes (voz y datos). Después, en la sección 5 se entregan algunos resultados numéricos. Por último, en la sección 6 se presentan las conclusiones del trabajo realizado.

2 Descripción del Meta-Algoritmo LDS

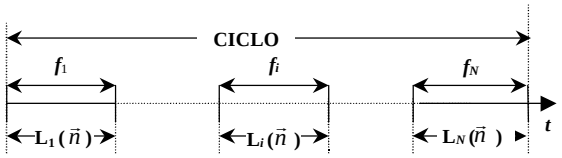


Figura 1: Diagrama temporal de la operación de LDS.

En la Fig. 1 se muestra el diagrama temporal de la operación del Meta-algoritmo LDS. LDS opera cíclicamente. Cada ciclo está compuesto por *N* fases, donde *N* corresponde al número de tipos de tráfico diferentes atendidos por el multiplexor. Durante la fase f_i ($1 \le i \le N$) el multiplexor transmite exclusivamente celdas del tipo de tráfico *i*. La duración de cada una de las fases depende del estado que tiene el multiplexor (cantidad de celdas que contiene cada uno de sus *buffers*) al inicio del ciclo. Esto implica que, en general, el tamaño de la fase f_i puede ser distinto en ciclos de operación distintos. La duración de un ciclo corresponde a la suma de la duración de cada una de sus fases.

En este escrito se supone que una unidad de tiempo corresponde al intervalo necesario para transmitir una celda ATM. Por este motivo, de aquí en adelante se hablará indistintamente del tiempo que dura una fase o de la cantidad de celdas transmitidas en dicha fase. A continuación se presenta más en detalle la operación de LDS.

El Meta-algoritmo LDS opera de acuerdo a la forma que se especifica en una tabla que tiene almacenada el multiplexor, la cual se denomina Tabla LDS. Esta tabla se compone de *2N* columnas, donde *N* corresponde a la cantidad de tráficos distintos atendidos por el multiplexor (ver, por ejemplo, la Tabla 1 en la que se muestra una tabla LDS para el caso de *N* tipos de tráfico). Las primeras *N* columnas se rotulan como n_i ($1 \le i \le N$), donde n_i es un parámetro que designa el número de celdas de tipo *i* que posee el multiplexor al inicio del ciclo. Las segundas *N* columnas se rotulan como $L_i(\vec{n})$, donde $L_i(\vec{n})$ es un parámetro que designa el máximo número de celdas de tipo *i* que pueden ser transmitidas durante la fase f_i. $L_i(\vec{n})$ depende del vector de estado $\vec{n} = (n_1, n_2, ..., n_N)$, es decir de la cantidad de celdas que contiene el multiplexor en cada uno de sus *buffers* al inicio de un ciclo. Por último, la tabla LDS posee tantas filas como situaciones de carga distintas puedan ser distinguidas. De acuerdo a estas definiciones, LDS opera de la siguiente forma:

1. Al inicio de un ciclo el multiplexor lee el vector de estado $\vec{n}$, en el que se encuentra en ese instante.
2. Si $\vec{n} = \vec{0}$, el sistema (multiplexor) espera que termine la unidad de tiempo actual, en cuyo instante reinicia este algoritmo (en el punto 1). Esto implica que para el caso $\vec{n} = \vec{0}$, el ciclo de LDS se compone solamente de una unidad de tiempo, en la cual obviamente no transmite ninguna celda, pero pueden llegar celdas de distinto tipo al multiplexor.
3. Si $\vec{n} \neq \vec{0}$, el algoritmo lee las primeras *N* columnas de cada fila de la Tabla LDS, hasta encontrar la situación de carga que corresponde al vector $\vec{n}$. Luego se leen las segundas *N* columnas de la fila seleccionada, las cuales especifican la duración máxima que puede tener cada fase de ese ciclo, es decir especifican el valor de $L_i(\vec{n})$, $1 \le i \le N$.
4. A continuación, se transmiten las celdas de la fase f_1, luego de la fase f_2 y así sucesivamente hasta la fase f_N.
5. Durante la fase f_i, $1 \le i \le N$, se transmiten $L_i(\vec{n})$ celdas de clase *i* sólo si el multiplexor puede transmitir ininterrumpidamente esta cantidad de celdas. Nótese que aunque al inicio de un ciclo el número de celdas de tipo *i* sea menor que $L_i(\vec{n})$, la fase f_i puede durar $L_i(\vec{n})$ unidades de tiempo. Esta situación ocurre si durante el tiempo que transcurre desde el inicio del ciclo hasta que termina de atenderse la fase f_i, llegan suficientes celdas de tipo *i* como para mantener ocupado el multiplexor hasta que termine de transmitir las $L_i(\vec{n})$ celdas.
6. En el caso que las celdas de la fase f_i se terminen antes de que expire el tiempo $L_i(\vec{n})$, el algoritmo aborta la fase f_i en el instante en que se agotan las celdas de tipo *i*. En ese mismo instante inicia la atención de la fase f_{i+1}. Un caso particular de esta situación ocurre cuando al comienzo de la fase f_i no existen celdas del tipo *i*, en cuyo caso LDS se salta la fase f_i.

3 Emulación de Algunos Algoritmos de *Scheduling* utilizando LDS

Con el objetivo de explicar con mayor detalle la capacidad que tiene LDS de emular otros algoritmos de Scheduling, a continuación se presentan dos ejemplos.

3.1 Algoritmo de Scheduling WRR (*Weighted Round Robin*)

En la Tabla 1 se muestra la tabla LDS que permite emular el algoritmo WRR para *N* clases diferentes de tráficos. El valor de T_i ($1 \le i \le N$) que aparece en la

Tabla corresponde al número de celdas que se deben transmitir durante la fase f_i. Los valores de T_i se escogen de forma que sean proporcionales al ancho de banda que se desea asignar a cada tipo de tráfico.

Tabla 1: Tabla LDS para algoritmo WRR.

n_1	n_2	.....	n_N	$L_1(\vec{n})$	$L_2(\vec{n})$	.....	$L_N(\vec{n})$
>0	>0	.....	>0	T_1	T_2	.....	T_N

El algoritmo WRR especifica que en el caso en que el número de celdas de clase *i* se agote antes de que se transmitan las T_i celdas destinadas a esa fase, el multiplexor debe comenzar a atender inmediatamente la fase siguiente. Nótese que LDS emula correctamente esta situación, debido al punto 5 del algoritmo de LDS.

3.2 Algoritmos de *Scheduling* tipo PS (*Processor Sharing*)

Según se mencionó en la introducción, además de emular otros algoritmos, LDS se puede programar para que se comporte de acuerdo a la propuesta de un nuevo algoritmo. Por ejemplo, en la Tabla 2 se presenta un nuevo algoritmo, el cual intenta operar en forma aproximada a PS y que hemos denominado Algoritmo Proporcional. En la notación utilizada en la Tabla 2, *T* corresponde al tamaño máximo de un ciclo.

Tabla 2: Tabla LDS para Algoritmo Proporcional.

n_1	n_2	$L_1(\vec{n})$	$L_2(\vec{n})$
$n_1 \le u$	≥ 0	$T\left(\frac{n_1}{\sum_{\forall i} n_i}\right)$	$T\left(\frac{n_2}{\sum_{\forall i} n_i}\right)$
$n_1 > u$	≥ 0	n_1-u	0

En este algoritmo, el número de celdas transmitidas en cada fase es proporcional al número de celdas de ese tipo que contiene el multiplexor al inicio del ciclo. Sin embargo, con el objetivo de mantener acotado el retardo experimentado por el tráfico de tipo 1 (por ejemplo tráfico de tiempo real), en los casos en que el número de celdas tipo 1 es mayor que un determinado umbral (denominado *u*), el multiplexor sólo atiende a este tipo de tráfico. Esta situación se mantiene hasta que la ocupación del *buffer* correspondiente vuelve a ser menor que dicho umbral, en cuyo caso el algoritmo retoma su operación normal.

4 Evaluación de Rendimiento de LDS

El análisis de rendimiento de LDS que se presenta a continuación es válido para *N* tipos de tráficos diferentes. Sin embargo, con el objeto de simplificar la explicación, se ha particularizado el análisis para dos tipos de tráficos distintos: uno de estos tráficos es de datos y el otro es un tráfico de tiempo real, como por ejemplo voz.

4.1 Descripción del Multiplexor.

En esta sección se explica la operación del algoritmo LDS, para el caso particular en que se implementa en un multiplexor como el de la Fig. 2. Este multiplexor atiende dos tipos de tráfico: voz y datos. Dado que en este caso sólo existen dos tipos de tráfico, cada ciclo de operación de LDS se divide en 2 fases: la fase de voz (f_v) y fase de datos (f_d).

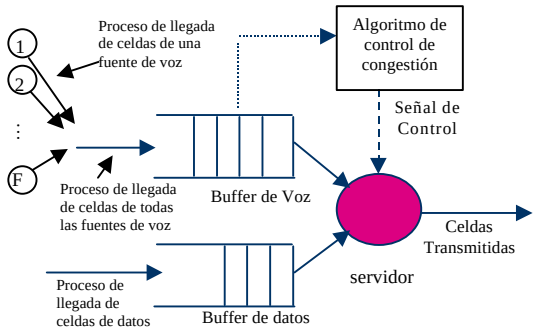


Figura 2: Multiplexor de Voz y Datos

4.1.1 Paquetización de la Información de Voz

La información de voz es almacenada en las celdas ATM de la forma que se describe a continuación. Cuando una fuente desea transmitir información de voz, en primer lugar digitaliza la señal original utilizando un esquema ADPCM embutido (4 bits por muestra) [4], [7]. En este esquema cada fuente es muestreada solamente durante los períodos de actividad, a una tasa constante de 8KHz. Las muestras de voz son empaquetadas en pares de celdas ATM, cada una de las cuáles se compone de 5 bytes de encabezado, 4 bytes de encabezado AAL2 [1] (para que la información de voz sea atendida como tráfico de tiempo real con tasa de transmisión variable) y 44 bytes de información. Entonces, para llenar un par de celdas ATM es necesario obtener 176 muestras. Los dos bits menos significativos de cada una de las 176 muestras se almacenan en forma secuencial en una celda de baja prioridad la cual se etiqueta como "descartable" (escribiendo un 1 en el bit CLP de la celda ATM). Análogamente, los dos bits más significativos de cada una de las 176 muestras se almacenan en forma secuencial en una celda de alta prioridad que se etiqueta como "no-descartable" (escribiendo un 0 en el el bit CLP de la celda ATM). En lo restante del artículo se designará "par de celdas" al par de celdas que se genera a partir de 176 muestras consecutivas de voz.

4.1.2 Control de Congestión para el Tráfico de Voz

El multiplexor posee un mecanismo de descarte de celdas de voz, que se activa en los períodos en que éste tipo de tráfico se encuentra en congestión. Mediante el descarte de celdas durante estos períodos, el multiplexor disminuye la degradación de *QoS* que inevitablemente experimenta el tráfico de voz en estos períodos. El mecanismo de control de

congestión usado en esta sección fue extraído de [14]. En este mecanismo se define (arbitrariamente) que el tráfico de voz está en congestión cuando la cantidad de celdas que contiene el *buffer* de voz supera un cierto umbral *u*. Específicamente, si al comienzo de la atención de un par de celdas el multiplexor contiene a lo más *u* pares de celdas de voz, entonces se transmiten las dos celdas del par. En caso contrario, es decir cuando hay más de *u* pares de celdas en el *buffer* de voz al comienzo de la transmisión de un par de este tipo de celdas, se descarta la celda de baja prioridad (es decir, se transmite sólo la celda de alta prioridad). Por último, si el *buffer* está lleno, se descarta cualquier par de celdas que llegue al *buffer* mientras se mantenga esta condición.

Note que la relación existente entre la política de *scheduling* y el control de congestión es la siguiente: la política de *scheduling* especifica la duración de cada fase; mientras que el control de congestión especifica que si durante la fase de voz este tipo de tráfico entra en congestión, sólo se deben transmitir las celdas de alta prioridad de este tipo de tráfico.

4.2 Modelo del Multiplexor

A continuación se describe el modelo utilizado para representar la llegada de celdas al multiplexor, el almacenamiento y atención de celdas, y la dinámica de la operación de LDS.

4.2.1 Modelo del proceso de llegadas de celdas

Para modelar la llegada de celdas de cada tipo de tráfico se puede utilizar cualquiera de los modelos propuestos en la literatura, como por ejemplo: Fluidos, MMPP, Poisson, etc.

Es bastante conocido que, en general, la superposición de celdas de voz proveniente desde varias fuentes no es un proceso de Poisson [10]. Sin embargo, en [13] se demostró que para el caso en el cual las celdas menos significativas de las muestras de voz se descartan cuando el multiplexor se encuentra en congestión, la llegada de pares de celdas de voz se puede modelar apropiadamente utilizando un proceso de Poisson. Entonces, debido a que en el caso bajo análisis LDS opera bajo las condiciones en que dicho modelo es válido, en este trabajo se utiliza el mismo modelo propuesto en [13] para representar la llegada de pares de celdas de voz.

En cuanto al proceso de llegada de celdas de datos, éste se caracteriza como un proceso de Poisson [13].

En resumen, el proceso de llegadas de pares de celdas de voz se modela como un proceso de Poisson de parámetro l_v y el proceso de llegadas de los datos se modela como un proceso de Poisson de parámetro l_d, independiente del proceso anterior.

4.2.2 Modelo de Almacenamiento y Atención de Celdas

Para almacenar la información de voz, el multiplexor mantiene un *buffer* de tamaño K_v pares de celdas. La atención de los pares de celdas de voz se realiza sólo durante la fase de voz, durante la fase de datos el multiplexor sólo almacena este tipo de celdas. La disciplina de atención es FIFO y el tiempo que el multiplexor demora en atender un par de celdas de voz es determinista, pero dependiente de la carga, de la manera que se define a continuación. Sea s_{n_v} el tiempo necesario para transmitir un par de celdas de voz cuando hay n_v pares de celdas de voz en el multiplexor, entonces, debido a la política de descarte de celdas, se tiene que:

$$s_{n_v} = \begin{cases} \mathrm{D}_2 & n_v = 1, 2, \ldots, u \\ \mathrm{D}_1 & n_v = u+1, \ldots, K_v \end{cases} \qquad (1)$$

Donde *u+1* es el mínimo número de pares de celdas que debe tener el *buffer* de voz para que se considere que éste se encuentra en congestión. La ecuación de arriba implica que, en los casos en que $n_v > u$, el multiplexor atiende sólo una celda del par, lo que realiza en un intervalo de tiempo igual a D_1. (nótese que, según el acuerdo de notación establecido anteriormente, se tiene que $D_1 = 1$). Por otro lado, D_2 (=2 D_1) es el intervalo de tiempo necesario para que el multiplexor transmita las dos celdas del par, que es lo que ocurre cuando no hay congestión (es decir, cuando el número de pares de celdas de voz en el *buffer* es menor o igual a *u*).

Respecto al almacenamiento de celdas de datos, el multiplexor posee un *buffer* de tamaño K_d celdas. El servicio de las celdas de datos sólo se realiza durante la fase de datos, durante la fase de voz el multiplexor, sólo las almacena. La disciplina de atención es FIFO, y el tiempo que el multiplexor se demora en atender una celda de datos es igual a D_1.

4.2.3 Modelo de la dinámica de la operación del Meta-Algoritmo LDS.

La Fig. 3 muestra la operación cíclica del meta-algoritmo LDS. En ella puede verse que cada ciclo está compuesto por una fase de voz y una fase de datos. En la fase de voz sólo son transmitidas celdas de voz. Análogamente en la fase de datos sólo se transmiten celdas de datos.

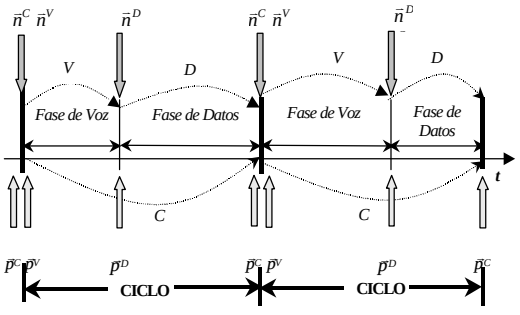


Figura 3: Cadenas de Markov *C, V y D* , y vectores de probabilidad $\vec{p}_C$, $\vec{p}_V$ y $\vec{p}_D$.

Sea *C* una cadena de Markov de parámetro discreto cuyas transiciones ocurren sólo en los instantes en que se inicia un ciclo (ver Fig. 3). El estado de la cadena *C* (en los puntos embutidos) corresponde al vector $\vec{n}^C = (n_v^C, n_d^C)$, donde la primera componente identifica el número de pares de celdas de voz que contiene el *buffer* de voz, y la segunda componente corresponde al número de celdas de datos contenidas en el *buffer* de datos. La matriz de probabilidades de transición asociada a la cadena *C* se denomina **C**.

Para modelar el comportamiento del sistema en la fase de voz, en los puntos embutidos correspondientes a los instantes en que se inicia y en que termina una fase de voz (ver Fig. 3), se define la cadena de Markov *V*. Un estado de la cadena *V* está dado por el vector $\vec{n}^V = (n_v^V, n_d^V)$. La matriz de probabilidades de transición asociada a la cadena *V* se denomina **V.**

Finalmente, para modelar el comportamiento del sistema en la fase de datos, en los puntos embutidos correspondientes a los instantes en que se inicia y en que termina una fase de datos (ver Fig. 3), se define la cadena de Markov *D*. Un estado de la cadena *D* está dado por el vector $\vec{n}^D = (n_v^D, n_d^D)$. La matriz de probabilidades de transición asociada a la cadena *D* se denomina **D**.

De las definiciones de las cadenas *C*, *V* y *D* es fácil ver que los elementos de la matriz **C** se obtienen a través de la siguiente ecuación matricial:

$$\mathbf{C}=\mathbf{V}\mathrm{x}\mathbf{D} \qquad (2)$$

El autovector izquierdo de la matriz **C**, denotado por $\vec{p}_C$, se obtiene a través de la ecuación:

$$\vec{p}_C = \vec{p}_C \mathbf{C} \qquad (3)$$

Sea $\vec{p}_V$ el vector de probabilidad, en estado estacionario, del estado del multiplexor al inicio de la fase de voz. Por definición, este vector está dado por:

$$\vec{p}_V = \vec{p}_C \qquad (4)$$

Sea $\vec{p}_D$ el vector de probabilidad, en estado estacionario, del estado del multiplexor al inicio de la fase de datos. Este vector está dado por:

$$\vec{p}_D = \vec{p}_V \mathbf{V} \qquad (5)$$

Observe que las filas $(0,i)$, $0 \le i \le K_d$, de la matriz **V** no son estrictamente necesarias, ya que para que exista la fase de voz el multiplexor debe contener al menos un par de celdas de voz al inicio del ciclo. Sin embargo, en **V** se crearon los estados $(0,i)$, $0 \le i \le K_d$, como estados absorbentes con el objeto de mantener dimensionalmente válida la ecuación (2). Respecto a los elementos de la fila (0,0) de la matriz **C** se obtienen considerando que la llegada de celdas está modelada por un proceso de Poisson y que el ciclo dura D_1 unidades de tiempo.

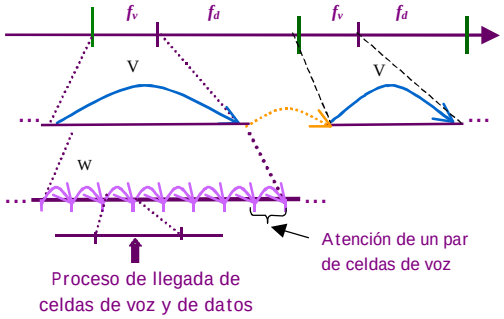


Figura 4: Transmisión de Voz durante la fase de voz.

Tal como se puede observar en la Fig. 4, durante la fase de voz no sólo se transmiten pares de celdas de voz, sino que además llegan pares de celdas de voz y celdas de datos. Entonces, para evaluar el comportamiento del sistema durante la fase de voz, se define la cadena de Markov *W* (ver Fig. 4). Las transiciones de *W* ocurren cada vez que se termina de atender un par de celdas de voz. Un estado de la cadena *W* corresponde al número de pares de celdas de voz que contiene el multiplexor. El estado inicial de la cadena *W* corresponde al número de pares de celdas de voz que contiene el multiplexor en el instante en que comienza la fase de voz. La matriz de probabilidades de transición asociada a la cadena *W* se denomina **W**. Nótese que, de acuerdo a la definición recién dada, se cumple que el estado de la cadena *W* al iniciar y al finalizar la fase de voz es igual a la primera componente del estado de la cadena *V* en esos mismos instantes.

	0	1	2	. . .	u	u+1	. . .	K_v-2	K_v-1	K_v
0	1	0	0	. . .	0	0	. . .	0	0	0
1	$a_v(0)$	$a_v(1)$	$a_v(2)$	. . .	$a_v(u)$	$a_v(u+1)$	. . .	$a_v(K_v-2)$	1-Σ	0
2	0	$a_v(0)$	$a_v(1)$	. . .	$a_v(u-1)$	$a_v(u)$	. . .	$a_v(K_v-3)$	1-Σ	0
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
u	0	0	0	. . .	$a_v(1)$	$a_v(2)$	. . .	$a_v(K_v-u-1)$	1-Σ	0
u+1	0	0	0	. . .	$b_v(0)$	$b_v(1)$	. . .	$b_v(K_v-u-2)$	1-Σ	0
.	.	.	.	.	.	.	.	.	.	
.	.	.	.	.	.	.	.	.	.	
.	.	.	.	.	.	.	.	.	.	
K_v-1	0	0	0	. . .	0	0	. . .	$b_v(0)$	1-Σ	0
K_v	0	0	0	. . .	0	0	. . .	0	1	0

Figura 5: Matriz de probabilidad de transición **W**.

En la Fig. 5 se muestra la matriz de probabilidades de transición *W*, donde el término $a_v(j)$ corresponde a la probabilidad de que durante un intervalo de tiempo D_2 lleguen *j* pares de celdas de voz. Análogamente, el término $b_v(j)$ corresponde a la probabilidad de que durante un intervalo de tiempo D_1 lleguen *j* pares de celdas de voz. Debido a que, en este análisis el proceso de llegada de pares de celdas de voz se modela como un proceso de Poisson de parámetro l_v, las expresiones para $a_v(j)$ y $b_v(j)$ son las siguientes:

$$a_v(j) = e^{-l_v D_2} \frac{(l_v D_2)^j}{j!} \qquad (6)$$

$$b_v(j) = e^{-l_v D_1} \frac{(l_v D_1)^j}{j!}$$

Nótese que, debido a que el sistema atiende sólo un par de celdas por vez, en la matriz **W** la probabilidad de transición desde un estado *i* a un estado menor que *(i-1)* tiene un valor igual a cero. Nótese también que, en los instantes en que la cadena *W* efectúa transiciones, no es posible que el *buffer* de voz se encuentre lleno, ya que en ese instante el par de celdas que acaba de ser atendido abandona el *buffer*. Por este motivo la columna K_v de la Fig. 5 tiene asociada una probabilidad igual a cero. Por otro lado, debido a que **W** es una matriz estocástica, el término 1-Σ que aparece en la columna rotulada (K_v-1) de la fila *i*, corresponde a: $\left(1 - \sum_{j=0}^{K_v-2} W[i, j]\right)$.

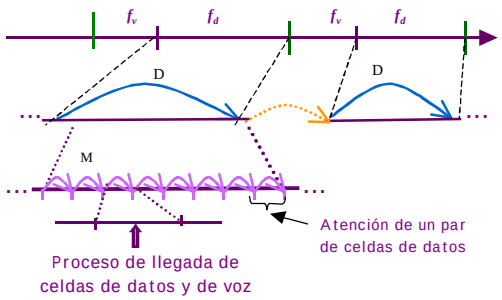


Figura 6: Transmisión de datos durante la fase de datos.

Para estudiar el comportamiento del sistema durante la fase de datos se procede de manera a lo que se hizo para la fase de voz. En este caso se define la cadena de Markov *M* (ver Fig. 6) que evoluciona sólo durante la fase de datos. Las transiciones de *M* ocurren cada vez que el multiplexor termina de atender una celda de datos. Para este caso, $p_d(j)$ corresponde a la probabilidad de que durante un tiempo D_1 lleguen *j* celdas de datos. Además, debido a que el proceso de llegada de celdas de datos se modela como un proceso de Poisson de parámetro l_d, $p_d(j)$ está dada por:

$$p_d(j) = e^{-l_d D_1} \frac{(l_d D_1)^j}{j!} \qquad (7)$$

5 Medidas de Desempeño

Las medidas de rendimiento que se evaluaron son: tamaño medio del ciclo, tamaño medio de cada fase, número medio de pares de celdas de voz que contiene el *buffer* de voz, el número medio de celdas de datos que contiene el *buffer* de datos, retardo medio experimentado por cada tipo de tráfico, número medio de celdas atendidas por unidad de tiempo (*throughput)* de cada tipo de tráfico, y la probabilidad de pérdida de celdas de voz y de datos. Sin embargo, debido a limitaciones de espacio, a modo de ejemplo de la forma en que fueron calculadas las medidas de rendimiento, a continuación se explica la obtención del número medio de celdas de datos durante la fase de voz.

5.1 Número medio de celdas de datos en la fase de voz

Sea N_d^V el número medio de celdas de datos en la fase de voz. Debido a que N_d^V depende del estado $\vec{n}^V$ del multiplexor al inicio de la fase de voz, para evaluar esta medida se condiciona y descondiciona en el valor de $\vec{n}^V$. Entonces, utilizando el teorema de probabilidades totales, se obtiene:

$$N_d^V = \sum_{\forall \vec{n}} \vec{p}_V\left[\vec{n}^V\right] N_d^V(\vec{n}^V) \qquad (8)$$

donde $\vec{p}_V\left[\vec{n}^V\right]$ corresponde a la probabilidad en estado estacionario de que la fase de voz se inicie en el estado $\vec{n}^V$; y $N_d^V(\vec{n}^V)$ corresponde al número medio de celdas de datos durante la fase de voz, dado que la fase de voz se inicia en el estado $\vec{n}^V$. Para evaluar $N_d^V(\vec{n}^V)$ se condiciona en el número *j* de celdas de datos recibidas durante la fase de voz y se aplica el teorema de probabilidades totales:

$$N_d^V(\vec{n}^V) = \sum_{j=0}^{\infty} p(j) \quad N_d^V(\vec{n}^V, \quad j) \qquad (9)$$

donde *p(j)* corresponde a la probabilidad de que durante la fase de voz lleguen *j* celdas de datos. Debido a que *p(j)* se distribuye según una variable aleatoria Poisson de parámetro l_d, para evaluar (9) es necesario conocer la duración de la fase de voz. Por este motivo, en la ecuación (9) se aplica el teorema de probabilidades totales, condicionando en el valor *t* de la duración de la fase de voz:

$$N_d^V(\vec{n}^V) = \sum_{\forall t} \Pr\left\{ t_v = t \mid \vec{n}^V \right\} \sum_{j=0}^{\infty} p(j \mid t) \; N_d^V(\vec{n}^V, \; j, \; t) \qquad (10)$$

donde $\Pr\left\{ t_v = t \mid \vec{n}^V \right\}$ corresponde a la probabilidad de que la fase de voz dure *t* unidades de tiempo, dado que el estado del multiplexor al inicio de la fase de voz es $\vec{n}^V$; y $N_d^V(\vec{n}^V, j, t)$ es el número medio de celdas de datos durante la fase de voz, cuando se cumple que: al inicio de esta fase el estado del

multiplexor es $\vec{n}^V$, durante la fase llegan j celdas de datos, y la fase dura τ unidades de tiempo. Debido a que $N_d^V(\vec{n}^V, j, t)$ no depende de la duración de la fase de voz, sino que solamente del número j de celdas de datos que llegan durante esta fase y del estado del multiplexor al inicio de la misma fase, $N_d^V(\vec{n}^V, j, t)$ se denota simplemente como $N_d^V(\vec{n}^V, j)$. Esta observación permite re-escribir (10) de la siguiente forma:

$$N_d^V(\vec{n}^V) = \sum_{\forall t} \Pr\left\{ t_v = t \mid \vec{n}^V \right\} \sum_{j=0}^{\infty} p(j \mid t)\ N_d^V(\vec{n}^V,\ j) \qquad (11)$$

Para evaluar $N_d^V(\vec{n}^V, j)$ se distinguen dos situaciones: una corresponde al caso en la cual el *buffer* de datos no se satura con la llegada de j celdas de datos, y la otra ocurre cuando el *buffer* de datos se satura con estas llegadas. Para el primer caso, en la evaluación de $N_d^V(\vec{n}^V, j)$, se observa que este valor medio corresponde a la altura media de un gráfico en el cual las abscisas representan el tiempo que dura la fase de voz y las ordenadas representa el número de celdas que contiene el multiplexor. De este análisis se concluye que:

$$N_d^V\left(\vec{n}^V,\ \ j\right) = n_d^V + \frac{j}{2}; \qquad 0 \le j < (K_d - n_d^V) \qquad (12)$$

Por otro lado, para evaluar $N_d^V(\vec{n}^V, j)$, para el caso en que el *buffer* de voz se satura durante la fase de datos, el raciocinio es similar al caso anterior. De donde se obtiene que:

$$N_d^V(\vec{n}^V, j) = K_d - \frac{(K_d - n_d^V + 1)(K_d - n_d^V)}{2(j+1)}; \quad j \ge (K_d - n_d^V) \qquad (13)$$

Descomponiendo la segunda sumatoria del lado derecho de la ecuación (11) en las dos situaciones recién explicadas, reemplazando las ecuaciones (12) y (13) en (11), y usando el hecho de que las llegadas de celdas de datos se modelan por un proceso Poisson de parámetro l_d, se concluye que:

$$N_d^V(\vec{n}^V) = \sum_{\forall t} \Pr\left\{t_v = t \mid \vec{n}^V\right\} \left[\sum_{j=0}^{K_d - n_d^V - 1} \left(n_d^V + \frac{j}{2} \right) e^{-l_d t} \frac{(l_d t)^j}{j!} + \sum_{j=K_d - n_d^V}^{\infty} \left(K_d - \frac{(K_d - n_d^V)\ (K_d - n_d^V + 1)}{2\ (j+1)} \right) e^{-l_d t} \frac{(l_d t)^j}{j!} \right] \qquad (14)$$

Para terminar de evaluar N_d^V, falta solamente obtener la expresión para $\Pr\left\{ t_v = t \mid \vec{n}^V \right\}$. Con este objetivo, se asignan recompensas a las transiciones de la cadena *W*, de acuerdo a la siguiente función:

$$h(i,\ j) = \begin{cases} 0; & i = 0 \\ D_2; & 1 \le i \le u;\ \ i-1 \le j < K_v \\ D_1; & u < i \le K_v;\ \ i-1 \le j < K_v \end{cases}$$

Con lo cual se llega a que:

$$\Pr\left\{ t_v = t \mid \vec{n}^V \right\} = g(L_v, t) \qquad (15)$$

donde $g(L_v, t)$ corresponde a la probabilidad de que la recompensa total acumulada por la cadena *W* en L_v transiciones (en cada transición *W* gana recompensas de la forma que especifica la ecuación 15) sea igual a t. El método de evaluación de $g(L_v, t)$ se puede encontrar en [16].

5.2 Ejemplos numéricos

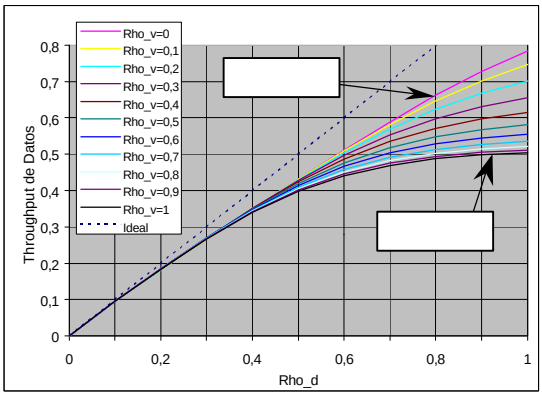


Figura 8. *Throughput* t normalizado de datos en función de la carga de datos.

En la Fig. 8 se muestra el gráfico del *throughput* normalizado de datos, es decir el *throughput* dividido por la carga total del canal, en función de la carga de datos. En esta figura la carga de datos es denotada Rho_d y la carga de voz es denotada Rho_v. Ambas cargas están normalizadas por la carga total del canal. Además, en línea discontinua está graficado el *throughput* ideal, es decir el *throughput* que se obtiene con un *buffer* infinito y una carga de voz normalizada igual a 0.

Para este ejemplo se utilizó un *buffer* que puede almacenar dos pares de celdas de voz y cuatro celdas de datos. Estos valores fueron escogidos con el propósito de mostrar el efecto que tiene un *buffer* de poca capacidad sobre el *throughput*. Además, debido a la poca capacidad del *buffer* de voz no fue necesario utilizar control de congestión. Por último, la política de *scheduling* que se utilizó fue la Política Proporcional, ver sección 3.2.

En el gráfico puede observarse que, para el caso Rho_v=0 el *throughput* de datos presenta una degradación en relación con el caso ideal, aún para muy baja carga del canal de voz. Esto se debe al hecho de que el *buffer* es de poca capacidad. También puede observarse que para una misma carga de datos (Rho_d), a medida que la carga de voz (Rho_v) aumenta, el *throughput* de datos disminuye. Esta situación es bastante notoria cuando la suma de las cargas normalizadas de voz y datos (Rho_v + Rho_d) es cercana a 1. Por ejemplo, para una carga de datos Rho_d=0,6 y una carga de voz Rho_v =0,2, el

throughput normalizado de datos es 0,503; en cambio cuando la carga de datos es 0,6 y la carga de voz es 0,4 el *throughput* normalizado de datos es 0,48.

6 Conclusiones

Se propuso una nueva metodología para evaluar el desempeño de algoritmos de *scheduling* en redes ATM. La metodología se basa en un meta-algoritmo, denominado LDS, el cual permite emular la operación de distintos algoritmos de *scheduling* simplemente cambiando sus parámetros de operación.

LDS posee las siguientes características: es flexible, ya que puede atender diferentes tipos de tráfico; coopera en garantizar la *QoS* de cada conexión, ya que es posible asegurar un cierto ancho de banda para cada clase de tráfico; es adaptivo, ya que la cantidad de celdas transmitidas de cada tipo de tráfico depende del estado de carga del multiplexor; es dinámico, ya que permite redistribuir en forma rápida el ancho de banda que no es empleado momentáneamente por algún tráfico que lo tenía asignado; es general, ya que al determinar en forma adecuada la tabla LDS, puede emular diferentes tipos de políticas de *scheduling* propuestas en la literatura.

La evaluación de desempeño de LDS fue llevada a cabo utilizando la técnica de Cadenas de Markov con Recompensas y el resultado obtenido es un conjunto de ecuaciones que se aplica a cualquier algoritmo de *scheduling* representable a través de una Tabla LDS.

Debido a que en la propuesta de un nuevo algoritmo siempre es necesario evaluarlo y comparar su desempeño respecto de otros algoritmos ya existentes, este trabajo resulta ser de gran utilidad en el desarrollo de nuevas propuestas en el área de *scheduling*, ya que permitirá en el futuro evaluar fácilmente nuevos algoritmos, solamente cambiando las tablas LDS. De este modo se evita desarrollar nuevos análisis matemáticos y tener que crear nuevos programas, lo que significa un ahorro de esfuerzo y una simplificación del trabajo en esta área.

Agradecimientos

Este trabajo ha sido parcialmente financiado por proyecto FONDECYT 100055/2000, CONICYT, Chile.

Referencias

[1] Recommendation Project ITU-T I.363.2. Specification about adaptation layer ATM type 2. Technical Report CCITT, April 1997.

[2] J. Bennet and H. Zhang, “WF2Q: Worst-Case Fair weighted Queueing”, Proceedings of IEEE INFOCOM ’96, San Francisco, pp.120-128, March 1996.

[3] J. Bennet and H. Zhang, “Hierarchical Packet Fair Queueing Algorithm”, IEEE/ACM Transactions on Networking, Vol.5, No.5 , 675-689, October 1997.

[4] CCITT. Recomendation G.726-40, 32, 24, 16 kbits/s Adaptive Differential Pulse code Modulation (ADPCM), December 1990.

[5] S. Golestani, “A Stop-and-Go Queueing Framework for Congestion Management”, ACM SIGCOMM Computer Communication Review, vol. 20, No. 4, pp. 8-18, Sept. 1990.

[6] S. Golestani, “A Self-Clocked fair Queueing Scheme for Broadband Aplications”, Proceedings of IEEE INFOCOM ‘94, pp. 636-646, toronto, June 1994.

[7] ITU-T. Recomendation I.363.2 B-ISDN ATM Adaptation Layer Specification: Type 2 AAL, September 1997.

[8] J.F. Meyer. “On evaluating the performability of degradable computing systems”. IEEE Transactions on Computers, C-29(8);720-731,1980.

[9] K. Nichols et. al. , “Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers”, RFC 2474, Dec. 1998.

[10] R. Onvural, "Asynchronous Transfer Mode Networks Issues", Artech House, 1994.

[11] A. K. Parekh and R. Gallager, “ A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single Node Case”, IEEE/ACM Transactions on Networking, Vol.1, No.3, pp. 344-357, June 1993.

[12] Shreedar M, Varghese G, “Efficient Fair Queueing Using Deficit Round Robin”, IEEE/ACM Transactions on Networking, Vol. 4, pp.375-385, June 1996.

[13] K. Sriram and W.Whitt. Characterizing superposition arrival process in packet multiplexers for voice and data. IEEE Journal on Selected Areas in Communications, SAC-4;833-846, Sept 1986.

[14] K.Sriram, D.Scott McKinney and M.Hashen Sheriff. Voice packetization and compression in a broadband ATM networks. IEEE JSAC, SAC-9, April 1991.

[15] K. Sriram, "Methologies for bandwidth allocation, transmission *scheduling*, and congestion avoidance in broadband ATM networks", Computer Networks and ISDN Systems 26 (1993) North-Holland.

[16] R. Vallejos C. “Modelos de Performability e suas aplicaçoes a sistemas Computacionais” Tesis de Doctorado, Universidade Federal do Rio de Janeiro, Brasil, 1993.

[17] de Veciana G, Kesidis G and Walrand J. , “Resource Management in Wide Area networks Using Effective Bandwidtdths. IEEE JSAC, Vol. 13 No..6, August 1995.

[18] L. Zhang, “Virtual clock: A New Traffic Control Algorithm for packet Switching Networks”, ACM SIGCOMM Computer Communications Reviews, Vol. 20, No. 4, pag. 19-29, September 1990.

Hacia una Facilidad de Dominio CORBA para Sistemas Distribuidos de Teleformación

L. Anido, M. Llamas, M.J. Fernández, M. Caeiro, J. Rodríguez y J. Santos
Área de Ingeniería Telemática
ETSI de Telecomunicación, Universidad de Vigo
E-36200 Vigo, SPAIN
{lanido,martin,manolo,mcaeiro,jestevez,jsgago}@ait.uvigo.es

***Abstract**. Institutions like the US Department of Defense or the Aviation Industry are currently specifying standards to support information models used in e-learning environments. From these works we present a Domain CORBA Facility for e-learning. Our proposal includes: (1) the application of an OO methodology to derive a reference architecture and design model from those conceptual entities identified by the learning technology standardization process; (2) support for standards-driven information models (e.g. course structures, student records, etc.) allowing easy reuse at the data level; (3) support for development of large-scale distributed systems in an incremental and scalable fashion; and (4) identification and specification of IDL interfaces offered by the domain facility, providing interoperability at the business logic level among components from different vendors.*

1 Introducción

Los avances en las Tecnologías de la Información y las Comunicaciones y especialmente en multimedia, redes de comunicaciones e ingeniería del software han propiciado la aparición de una nueva generación de sistemas de aprendizaje basados en ordenador. Muchas instituciones, tanto públicas como privadas, aprovechan las nuevas tecnologías para ofrecer servicios y productos de formación a todos los niveles utilizando el WWW. Como consecuencia, los sistemas de teleformación proliferan y surge la necesidad de la estandarización: muchos sistemas se diseñan partiendo de cero para cubrir las necesidades de una determinada institución, sin ningún mecanismo de interoperabilidad con otros sistemas externos. Así, los cursos que se desarrollan para un determinado sistema no pueden ser utilizados en otro, los registros y perfiles de los estudiantes no se pueden transferir fácilmente y no hay ningún procedimiento establecido para compartir servicios en tiempo de ejecución.

Este artículo trata sobre cuestiones relacionadas con la interoperabilidad entre sistemas de teleformación distribuidos a gran escala. Proponemos un marco orientado a objetos y guiado por estándares para desarrollar tales sistemas. Los modelos de información subyacentes se basan en el trabajo que actualmente están desarrollando sobre estandarización de las tecnologías de aprendizaje instituciones como el Departamento de Defensa de EEUU, el IEEE o la industria de la Aviación, consumidores muy importantes de software educativo. Estas instituciones proponen estándares sobre esquemas de metadatos orientados al dominio del aprendizaje, estructuras de contenidos, registros y perfiles de estudiantes, etc. Estos facilitarán la reutilización y transferencia de recursos educativos entre distintas instituciones o entre departamentos internos de una misma institución. La aplicación de CORBA[1], y sus capacidades de computación distribuida, nos permite añadir interoperabilidad de software y una reducción del tiempo de desarrollo de nuevos sistemas de aprendizaje.

CORBA es una arquitectura distribuida basada en objetos que permite la interoperabilidad entre aplicaciones heterogéneas y distribuidas en una red. Es un estándar definido por más de 800 instituciones formando el Object Management Group (OMG). Sobre un bus software, el Object Request Broker (ORB), los objetos CORBA interactúan mediante contratos definidos en IDL (Interface Definition Language). La interacción en entornos tipo Internet se soporta mediante el protocolo Internet Inter-ORB Protocol (IIOP).

La OMG Object Management Architecture (OMA) [2] (c.f. Fig. 1) organiza los objetos en torno al ORB en cuatro categorías: servicios CORBA[3], facilidades CORBA [4], objetos de dominios CORBA [5] y objetos de aplicación.

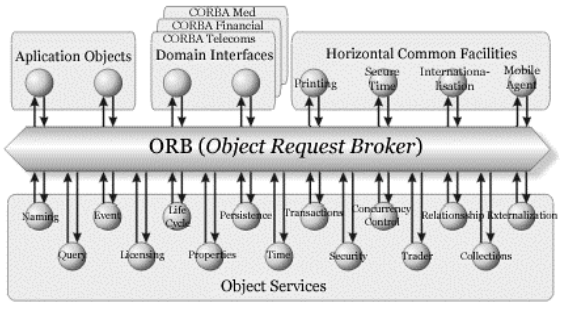


Figura 1: OMG Object Management Architecture.

El trabajo más interesante que está realizando actualmente el OMG se centra en las facilidades CORBA. Como se ha mencionado anteriormente, IDL ha sido seleccionado como el lenguaje común para proporcionar definiciones de interfaces. IDL es, probablemente, el modo más conveniente de definir interfaces estándar para objetos estándar que cada institución en un dominio puede compartir. Muchas compañías acudieron al OMG en busca de soporte, y esta organización estableció el Domain Technology Committee, a principios de 1996, para coordinar estas actividades. Actualmente, 10 industrias tienen su propia OMG Domain Task Force. Cada Domain Task Force define especificaciones útiles para un determinado dominio. Muchas se benefician de trabajos anteriores en el campo de la estandarización en el correspondiente área. Las facilidades de Dominio ofrecen nuevas posibilidades para la interoperabilidad y reutilización del software. Son marcos adecuados para desarrollar aplicaciones orientadas a objetos, distribuidas y estandarizadas para un campo concreto.

Por otra parte, soportan interoperabilidad entre aplicaciones desarrolladas por diferentes vendedores, una vez probado que cumplen con la consiguiente facilidad de dominio. Esto es particularmente importante para sistemas de gran escala, que necesitan tratar con un gran número de componentes diferentes y heterogéneos. Más aún, la práctica ha demostrado que la aprobación de una nueva facilidad de dominio significa un nuevo paso hacia la estandarización de ese dominio.

El artículo está organizado de la siguiente manera: la sección 2 presenta la metodología utilizada para derivar nuestro marco: CORBAlearn. La sección 3 trata sobre la fase de análisis de nuestro trabajo. Las secciones 4 y 5 muestran la arquitectura de referencia y los servicios de la Facilidad de Dominio CORBA propuesta. Finalmente presentaremos algunas conclusiones.

2 Metodología

El proceso de diseño de nuestro marco ha sido guiado por el Proceso de Desarrollo Software Unificado [6] y modelado mediante el Lenguaje de Modelado Unificado (UML) [7]. Combinamos el Proceso de Desarrollo Software Unificado con las recomendaciones de Bass et al. [8] para derivar nuestra arquitectura software.

El proceso unificado identifica un conjunto de iteraciones en el proceso de desarrollo software: requisitos, análisis, diseño, implementación y pruebas. Los diseñadores empiezan capturando los requisitos del cliente en el modelo de casos de uso. Después analizan y diseñan el sistema para cumplir los casos de uso, creando así primero un modelo de análisis, después un modelo de diseño y un modelo de despliegue; e implementando el sistema según un modelo de implementación. Finalmente, los desarrolladores preparan un modelo de pruebas que les permite verificar que el sistema proporciona las funcionalidades descritas en los casos de uso.

Por otra parte, Bass establece que un modelo de referencia y una arquitectura de referencia son pasos previos hacia una arquitectura software final. El modelo de referencia es una división de la funcionalidad, junto con el correspondiente flujo de datos entre los componentes identificados. Una arquitectura de referencia es la descomposición de los elementos funcionales en elementos de sistema. En las próximas secciones mostraremos como estas metodologías fueron aplicadas al diseño de un marco distribuido para teleformación.

3 Modelo de Análisis

Los casos de uso se llevan a cabo mediante colaboraciones de clases. El objetivo es hacerlo de un modo eficiente y escalable. El modelo de análisis crece iterativamente cuando más casos de uso son analizados. En cada iteración seleccionamos un conjunto de casos de uso que se estudian en el modelo de análisis. El sistema se construye como una estructura de clasificadores (clases de análisis) con las relaciones entre ellos.

Utilizamos tres estereotipos diferentes sobre las clases: <<boundary>> (para modelar interacción entre el sistema y sus actores), <<control>> (para representar coordinación, secuenciamiento, transacciones, etc.) y <<entity>> (para modelar información). Presentamos aquí varios ejemplos del modelo de análisis. Por ejemplo, la Fig. 2 muestra el diagrama de clases de análisis para realizar aquellos casos de uso relacionados con la localización y búsqueda de contenidos educativos. Hay una clase de control que implementa la lógica de búsqueda (posiblemente colaborando con otras instancias de la misma clase). Las clases *Searcher* utilizan clases <<entity>> que encapsulan las descripciones de cursos, las preferencias de los estudiantes y los datos de los proveedores de cursos. Los clientes acceden a los servicios de búsqueda mediante una clase <<boundary>>.

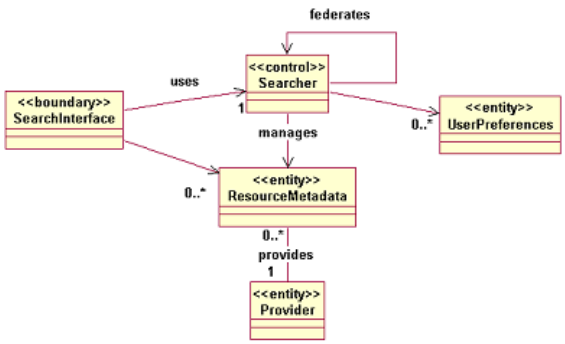


Figura 2: Diagrama de clases para la búsqueda de cursos.

La dinámica de las realizaciones de casos de uso se modela mediante diagramas de colaboración. La Fig. 3 representa el diagrama de colaboración para implementar un caso de uso de *búsqueda de curso*. Varias instancias de la clase *Searcher* colaboran para ampliar el alcance de la búsqueda.

Los entornos de aprendizaje son responsables de la entrega de contenidos y del seguimiento de alumnos. El correspondiente diagrama de clases de análisis se muestra en la Fig. 4. Los estándares sobre formatos de estructuras de cursos y sobre trazas son utilizados respectivamente por las clases de control *Navigation* y *Tracking Manager*.

4 La Arquitectura de Referencia

Se supone que los dominios maduros tienen arquitecturas de referencia que guían a los desarrolladores software en la construcción de nuevos sistemas. Las clases del modelo de análisis deben agruparse en subsistemas, especialmente para sistemas grandes. Un subsistema es una agrupación semánticamente útil de clases u otros subsistemas. Los subsistemas se utilizan también para modelar grupos de clases que tienden a cambiar conjuntamente. Sólo es posible instalar un susbsistema si se hace en su totalidad. En nuestro caso, cambios en los modelos de información estandarizados afectarían al menor número posible de subsistemas. Los subsistemas identificados conforman la arquitectura de referencia (c.f. Fig. 5). Debido a limitaciones de espacio, presentaremos solo parte de la arquitectura de referencia.

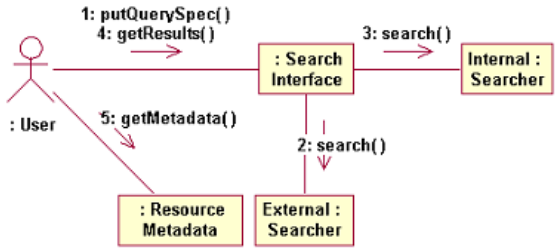


Figura 3: Diagrama de colaboración para la búsqueda de cursos.

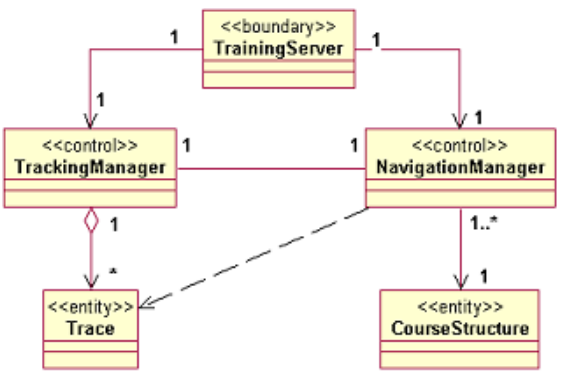


Figura 4: Diagrama de clases del entorno de aprendizaje.

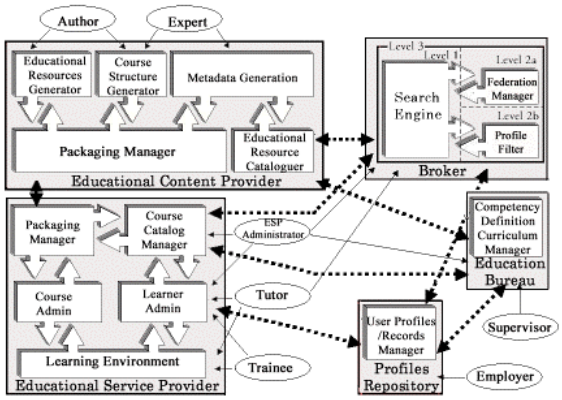


Figura 5: Arquitectura de Referencia.

4.1 Broker Reference Architecture

La arquitectura del Broker se compone de tres subsistemas software: el motor de búsqueda (*Search Engine* , SE), el gestor de federación (*Federation Manager*, FM) y el filtro de perfiles (*Profile Filter*, PF). SE utiliza el almacén de metadatos para la búsqueda de objetos de aprendizaje. No se impone ningún esquema particular de metadatos. De hecho, la interfaz ofrecida a los módulos externos debería ser la misma independientemente del modelo subyacente de metadatos que se utilice para describir los objetos de aprendizaje. Al mismo tiempo, debe existir un mecanismo para informar a los módulos de los clientes sobre los esquemas de metadatos soportados. El modelo de diseño debería definir explícitamente este procedimiento de introspección.

Los resultados de la búsqueda pueden filtrarse de acuerdo a las preferencias del usuario. Los clientes proporcionan al broker su perfil (o una referencia al mismo) que se utiliza para adaptar las búsquedas según las preferencias en él definidas. El PF es responsable de esto. No se impone ningún modelo de perfiles específico. Los métodos de introspección deben proporcionar información acerca de los modelos soportados por la implementación de componentes.

Mientras en el modelo de referencia el broker es un único módulo, en la arquitectura de referencia su funcionalidad puede tener que implementarse mediante la colaboración de un grupo de brokers federados. El subsistema software FM gestiona las federaciones de acuerdo a la topología con la que fueron configurados. Las búsquedas enviadas a través del Gestor de Federaciones son reenviadas a Motores de Búsqueda externos y, posiblemente, a Gestores de Federaciones externos.

Para ser compatible con nuestra arquitectura, solo es necesario implementar el componente del Motor de Búsqueda. La arquitectura de Brokerage define varios niveles de servicio. Dependiendo del nivel de compatibilidad habrá que implementar Filtros de

Perfil y/o Gestores de Federación. La tabla 1 muestra los niveles de servicio definidos para un broker y los componentes que se necesitan para cada uno de ellos.

Tabla 1: Niveles de Servicio del Broker.

	Búsqueda sencilla	Búsqueda filtrada	Búsqueda federada
Nivel 1	X		
Nivel 2a	X	X	
Nivel 2b	X		X
Nivel 3	X	X	X

Esta arquitectura sería totalmente escalable, un broker simple con incluye únicamente un SE podría añadir en el futuro subsistemas PF o FM. Nuestro modelo de diseño, presentado en la sección 5, describe los mecanismos automáticos de configuración de componentes para un entorno de implementación CORBA. El modelo estandarizado de información que cada componente trata, guió nuestro criterio de diseño. Un cambio o actualización en el modelo de datos que definen los metadadatos de aprendizaje o la información del perfil de preferencias del usuario sólo afectaría al los componentes SE o PF, respectivamente. De esta forma, cambios en un modelo de información afectarían al menor número posible de subsistemas de la arquitectura de referencia.

4.2 Arquitectura de Referencia ESP

La arquitectura de referencia del *Proveedor de Servicios Educativos (Educational Service Provider*, ESP) se compone de cinco subsistemas software (Fig. 5): Entorno de Aprendizaje (*Learning Environment*, LE), Administración de Estudiantes (*Learner admin.*, LA), Administración de Cursos (*Course admin.*, CA), Gestor de Empaquetado (*Packaging Manager*, PM) y el Gestor de Catálogos de Cursos (*Course Catalogue Manager*, CCM). El subsistema LE es responsable de guiar al estudiante a través de un curso, de la entrega de contenidos educativos y de las trazas del alumno. *Learner Admin* gestiona las matrículas de los estudiantes, la autenticación de usuarios y los registros y perfiles curriculares. *Course Admin* gestiona los cursos disponibles en la institución. Distribuye contenidos a LE y recibe nuevos recursos de Proveedores de Contenidos a través del *Packaging Manager*. CCM mantiene información actualizada sobre los cursos actuales que está ofreciendo la institución y los publica a través de brokers educativos.

5. CORBAlearn: Modelo de Diseño y Especificaciones

Nuestro propuesta –CORBAlearn- se compone de un conjunto de especificaciones que definen la conducta esperada para cada objeto que pertenezca a la arquitectura software. La arquitectura software de CORBAlearn se especifica utilizando el modelo de diseño de Proceso Unificado. Este modelo se crea utilizando el modelo de análisis como entrada primaria, pero se adapta al entorno de implementación seleccionado, tal como un ORB, un kit de construcción de elementos gráficos, o a un sistema de gestión de bases de datos. Similarmente al modelo de análisis, en el modelo de diseño también se definen clases, relaciones entre ellas, y colaboraciones que materializan los casos de uso. Sin embargo, los elementos definidos en el modelo de diseño difieren de los elementos más conceptuales definidos en el modelo de análisis, en el sentido de que los primeros son adaptados al entorno de implementación, mientras que los últimos (elementos de análisis) no. En otras palabras, el modelo de diseño es más físico en su naturaleza, mientras que el modelo de análisis es más conceptual.

En este punto fijamos una plataforma final de soporte, como CORBA en nuestro caso; y el producto final a entregar es identificado como una facilidad de dominio CORBA. Esta es la razón por la que aparecen patrones comunes de diseño en entornos CORBA a este nivel (e.g. Perfil UML CORBA[9], objetos factoría o mecanismos de navegación de interfaz).

El proceso de diseño utilizado para la elaboración de nuestra arquitectura software fue guiado por la arquitectura de referencia establecida y por los modelos de información estandarizados disponibles: cambios en estos modelos de información implicarían sólo cambios locales que afectan al menor número posible de objetos. La inclusión de nuevos componentes de la arquitectura de referencia debería ser directa y no afectar a los objetos que ya se están ejecutando en el sistema.

CORBAlearn cubre todos los elementos de un sistema distribuido de teleformación identificados por la arquitectura de referencia. Cada uno de ellos es soportado por una especificación diferente de la facilidad de dominio CORBAlearn. Debido a las limitaciones de espacio, sólo presentaremos aquí parte de la vista estática y dinámica de la arquitectura software y sus interfaces de objeto IDL. Particularmente, introduciremos los servicios de *Brokerage* y *Learning Environment*. Animamos al lector a pedir a los autores el conjunto completo de especificaciones.

5.1 Servicios del Broker

Se han definido tres niveles de servicio de brokerage (c.f. Fig. 5 y tabla 1): el nivel 1 trata las búsquedas sobre almacenes de metadatos locales. El nivel 2a tiene en cuenta las preferencias que el usuario ha predefinido para filtrar los resultados de la búsqueda. El nivel 2b posibilita la colaboración entre diferentes brokers. El nivel 3 soporta tanto filtrado como federación.

Todo broker conforme con CORBAlearn debe implementar la interfaz *BrokerComponent*. Ésta proporciona métodos que permiten conocer el nivel de servicio con el que el broker es conforme. Además, posibilita que un cliente obtenga una referencia a cualquiera de los objetos del broker de un modo sencillo a partir de cualquier otro. El servicio CORBA Trader [10] y la facilidad CORBA para el dominio de la Sanidad [11] utilizan este patrón de diseño, que es usado también en el Modelo de Componentes de CORBA (CCM) [12].

La actualización del nivel de servicio proporcionado por un broker se puede realizar en tiempo de ejecución. Los objetos nuevos que implementen un servicio más alto buscarán el *BrokerComponent* en el servicio de nombres CORBA [13], para actualizar sus referencias a las interfaces de los nuevos servicios disponibles. Los accesos posteriores de los clientes al *BrokerComponent* podrían utilizar los servicios instalados más recientemente. En la Fig. 6 se muestra el diagrama de clases UML para el subsistema *Broker* con los servicios ofrecidos. Este modelo de diseño se deriva directamente de su correspondiente diagrama de clases de análisis (c.f. Fig. 2) teniendo en cuenta los entornos de implementación definitivos.

Los brokers conformes con el Nivel-1 deben implementar la interfaz *BrokerManager*, para alimentar el almacén de metadatos del broker, y la interfaz *Searcher*, que define los servicios de búsqueda y localización. Esto permite a los desarrolladores de brokers educativos implementar políticas de introducción (*push*): los proveedores de servicios educativos o los proveedores de contenidos introducen las descripciones de cursos mediante metadatos en el broker. Las especificaciones CORBAlearn para estos subsistemas también definen servicios de acceso a los metadatos. Por tanto, también es posible una política de adquisición (*pull*): los subsistemas de intermediación (brokers) recogen descripciones de metadatos de los proveedores.

Los brokers de Nivel 2-a necesitan las interfaces *ProfileSearcher* y *ProfileManager*. La primera se utiliza para tratar el filtrado con preferencias y la última para acceder o actualizar a estas preferencias. Los datos de preferencia del usuario se pueden obtener directamente de un ESP o de un PM, o bien se puede crear un nuevo perfil de preferencias (i.e. los datos se pueden obtener por referencia o por valor).

La conformidad con el nivel 2b requiere la implementación de la interfaz *FederatedSearch*. Ésta define, en uno de sus atributos, una referencia al objeto *FederationManager*, el cual mantiene la información de la topología de la federación. Finalmente, el nivel 3 requiere la implementación de todas las interfaces presentadas anteriormente.

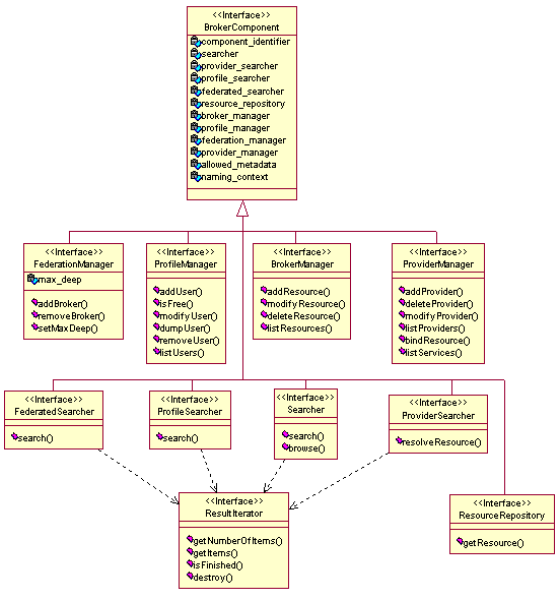


Figura 6: Diagrama de clases del Broker.

El *ResourceRepository* ofrece acceso a las descripciones de metadatos de los objetos de aprendizaje. Finalmente, los objetos *Provider Searcher* y *ProviderManager* definen servicios para localizar el Proveedor de Servicios Educativos (*Educational Service Provider)* o el Proveedor de Contenidos Educativos (*Educational Content Provider)* donde se puede acceder al objeto de aprendizaje. Estos tres objetos se utilizan en la fase de localización de objetos de aprendizaje.

La especificación CORBAlearn para el servicio de intermediación define, mediante IDL, todos aquellos servicios identificados en el modelo de diseño. Los diferentes vendedores que implementen componentes software de acuerdo con el interfaz pueden argumentar que las prestaciones ofrecidas por sus productos es mejor que las de sus competidores. Sin embargo, aunque las prestaciones pueden ser diferentes, el servicio especificado en la interfaz ha de ser exactamente el mismo. Observe que las interfaces actúan como un contrato entre la implementación del componente y los usuarios del mismo. Este es el punto clave tanto para la interoperabilidad entre diferentes aplicaciones distribuidas como para la reutilización de componentes software.

Las especificaciones CORBAlearn también definen mecanismos de introspección para averiguar los modelos de información que un determinado componente puede gestionar. Por ejemplo, el atributo *allowed_metadata* permite conocer a los clientes los esquemas de metadatos que soporta el broker.

En cualquier caso, las interfaces de servicio son independientes del modelo de información que se utilice: las definiciones de métodos no hacen referencia a ningún modelo en particular. Para

permitir a la implementación utilizar atributos concretos, las especificaciones CORBAlearn contienen definiciones de modelos de información potencialmente utilizados por diferentes implementaciones de componentes. Este patrón de diseño está tomado de otras facilidades de dominio CORBA (e.g. la interfaz de dominio para Sanidad). De esta manera, los clientes de CORBAlearn saben cómo invocar operaciones de acuerdo a modelos de información ya existentes. Por ejemplo, la especificación del esquema de metadatos Educacional Dublín Core [14] en IDL se puede ver en la Fig. 7.

5.2 Learning Environment Services

CORBAlearn define el diagrama de clases representado en la Fig. 8 para modelar el subsistema de Entorno de Aprendizaje (*Learning Environment)* (ver diagrama del modelo de análisis en la Fig. 4). Las responsabilidades se dividen principalmente entre un objeto *TrackingManager* que realiza el seguimiento del alumno durante una sesión de aprendizaje, y un objeto *NavigationManager* que controla la navegación por el curso en función de la estructura definida para el mismo y de las acciones previas del estudiante. Para realizar las tareas de navegación y seguimiento estos dos objetos deben trabajar conjuntamente. El *Learning Server Factory* crea un objeto *Learning Server* por cada estudiante que accede a un curso diferente.

5.2.1 Especificación IDL del Entorno de Aprendizaje

Los usuarios (desarrolladores de sistemas distribuidos de aprendizaje a distancia) de la facilidad CORBAlearn pueden contar con la existencia de los servicios ya definidos en la arquitectura software y en su interfaz IDL. La Fig. 9 muestra parte de la especificación IDL para el componente de Entorno de Aprendizaje.

```
module DublinCoreEducation
{
 typedef MetadataTypes::MetadataName DCEdFieldName;
 typedef string DCEdFieldValue;

 const DCEdFieldName TITLE = "DC/Title";
 const DCEdFieldName CREATOR = "DC/Creator";
 const DCEdFieldName SUBJECT = "DC/Subject";
 ...
 ...
 const DCEdFieldName COVERAGE = "DC/Coverage";
 const DCEdFieldName RIGHTS = "DC/Rights";
 const DCEdFieldName AUDIENCE = "DC-Ed/Audience";
 const DCEdFieldName STANDARD = "DC-Ed/Standard";
 const DCEdFieldName INTERACIVITYTYPE = "DC-Ed/InteractivityType";
 const DCEdFieldName INTERACIVITYLEVEL = "DC-Ed/InteractivityLevel";
 const DCEdFieldName TYPICALLEARNINGTIME = "DC-Ed/TypicalLearningTime";
};
```

Figura 7: IDL del esquema de metadatos Educational Dublín Core.

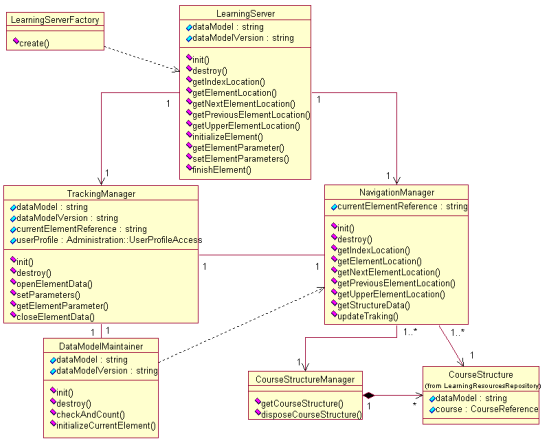


Figura 8: Diagrama de Clases UML del Entorno de Aprendizaje.

```
module LearningEnvironment{

//Type Definitions
  typedef string URL;
  typedef LearningResourcesRepository::CourseReference CourseReference;

  struct Trace {
    string name;
    string value;
  };

  typedef sequence<Trace> TraceSeq;

//Exceptions
  exception NoElementOpened {};
  exception ElementNotClosed {};
  exception NotFound {};
  exception DataModelError {
    long code;
    string description;
  };
  exception InvalidReference {};
  exception NoMoreElements {};

//Interfaces
  interface CourseStructureManager {
    struct StructureRecord {
       LearningResourcesRepository::CourseStructure structure;
       Long useCount;
    };
    typedef sequence<StructureRecord> StructureRecordSeq;
    LearningResourcesRepository::CourseStructure getCourseStructure (
       in CourseReference reference);
    void disposeCourseStructure(in CourseReference reference);
  };

  interface TrackingManager {
    readonly attribute string dataModel;
    readonly attribute string dataModelVersion;
    readonly attribute string currentElementReference;
    readonly attribute Administration::UserProfileAccess userProfile;
    void init(in Administration::UserProfileAccess user, in CourseReference course,
       in NavigationManager navigationReference);
    void destroy();
    void openElementData(in string elementReference) raises(ElementNotClosed);
    void setParameters(in TraceSeq paramSet, in boolean check)
       raises( NoElementOpened, DataModelError );
    string getElementParameter(in string elementReference, in string param)
       raises( NoElementOpened, DataModelError, NotFound );
    void closeElementData() raises( NoElementOpened );
  };

  interface NavigationManager {
    readonly attribute string currentElementReference;
    void init(in CourseReference course, in TrackingManager trackingReference);
    void destroy();
    URL getIndexLocation();
    URL getElementLocation(in string elementReference)raises( InvalidReference);
    URL getNextElementLocation() raises( NoMoreElements );
    URL getPreviousElementLocation() raises( NoMoreElements );
    URL getUpperElementLocation() raises( NoMoreElements );
    string getStructureData(in string elementReference, in string path)
       raises( InvalidReference );
    void updateTraking();
  };
  ...
  ...
```

Figura 9: Especificación IDL del Entorno de Aprendizaje.

5.2.2 Ejemplo de Aplicabilidad

Los servicios definidos incluyen funcionalidades comunes para entornos de aprendizaje. Los desarrolladores de sistemas particulares de aprendizaje basados en Web se benefician de los servicios ofrecidos y de su reutilización. De esta forma, se reduce el tiempo de implementación. Como ejemplo de aplicabilidad, hemos desarrollado un sistema de cursos basado en Web que es conforme con el entorno de ejecución definido por la iniciativa ADL del Departamento de Defensa de EEUU [15]. Este modelo seguramente será aceptado por la comunidad de estándares en tecnologías del aprendizaje como la vía común para lanzar y adquirir información de lecciones en entornos de aprendizaje a distancia basados en web. Para ello, solamente fue necesario desarrollar una pequeña capa entre el navegador web y los objetos del servidor CORBAlearn.

La Fig. 10 muestra la interfaz de usuario de la herramienta de teleformación desarrollada a partir de CORBAlearn. Los contenidos del curso son aquellos proporcionados como ejemplo por el ADL. La Fig. 10 muestra el índice del curso que se crea automáticamente a partir de la estructura del curso y de los datos del estudiante. Este tipo de funcionalidades, común en los entornos de teleformación, son las que ofrecen los servicios de CORBAlearn.

6. Conclusiones

Junto con el comercio electrónico y la banca electrónica, la teleformación se está convirtiendo en una de las aplicaciones estrella de la Web. Desgraciadamente, el diseño e implementación de sistemas distribuidos de aprendizaje carece de un marco adecuado. Como consecuencia, se desarrollan una y otra vez sistemas independientes con funcionalidad similar.

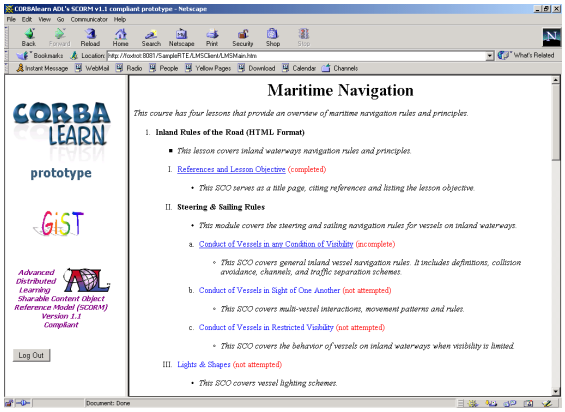


Figura 10: Entorno de ejecución basado en CORBAlearn

La tecnología CORBA del OMG proporciona un entorno adecuado donde es posible construir marcos para aplicaciones en dominios específicos. Las *Domain Task Forces* de CORBA identifican servicios software para dar soporte al desarrollo de software basado en componentes en entornos distribuidos. Los servicios de dominio se definen utilizando interfaces IDL que actúan como contratos entre los desarrolladores de componentes y los usuarios de tales componentes. Estos usuarios son a su vez desarrolladores de sus propios productos software en el dominio. El tiempo de puesta en el mercado del producto se ve drásticamente reducido ya que las facilidades de dominio proporcionan las funcionalidades comunes. Además, los servicios claramente identificados permiten la interoperabilidad en tiempo de ejecución entre componentes de distintos vendedores.

En este artículo hemos presentado una propuesta para una nueva Facilidad de Dominio CORBA: CORBAlearn. Su campo de aplicación es el aprendizaje electrónico distribuido, donde se están realizando grandes esfuerzos hacia su estandarización. La definición del servicio está basada en recomendaciones de instituciones involucradas en este proceso. Hemos discutido la aplicación del Proceso de Diseño Unificado y de la metodología de Bass para derivar marcos de desarrollo orientados al dominio. De la implementación del modelo de análisis obtenido, hemos identificado un modelo de referencia independiente del entorno y la correspondiente arquitectura de referencia.

El modelo de diseño del marco desarrollado sirve como base para CORBAlearn. Sus especificaciones facilitan el desarrollo y el despliegue de los subsistemas de la arquitectura de referencia de una forma escalable. Los mecanismos de navegación de interfaces, basados en el Modelo de Componentes de CORBA, se utilizan para permitir la instalación en tiempo de ejecución de nuevos elementos o la actualización de los ya existentes.

CORBAlearn también define especificaciones IDL para los modelos de información y estándares más comunes en el dominio de la teleformación. Las implementaciones que las utilicen deben ser conformes con los formatos y vocabularios definidos. Sin embargo, las interfaces de servicios son independientes del modelo de información utilizado. También hemos identificado mecanismos de introspección para permitir que los clientes conozcan los modelos de información, niveles de servicio y funcionalidad que la implementación es capaz de tratar.

Resumiendo, nos hemos basado en técnicas orientadas a objetos para construir un marco específico del dominio de teleformación, tanto para el diseño, como para el desarrollo y despliegue.

Como contribución marginal, las líneas perfiladas en este artículo pueden ayudar al lector en el desarrollo de nuevos marcos en otros dominios

Las pruebas de prototipado e interoperabilidad fueron realizadas en plataformas WinNT/Windows 2000 y GNU Linux Debian Potato 2.2. Los ORBs CORBA fueron Orbacus 4.0.3 y Orbix 2000 v1.1, ambos para C++ y Java.

Agradecimientos

Queremos dar las gracias a la XUNTA DE GALICIA por su apoyo parcial bajo la subvención PGIDT00TIC3220PR "Arquitecturas Distribuidas para Teleservicios".

Referencias

[1] J. Siegel. "CORBA 3 Fundamentals and Programming". Wiley & Sons. 1999.

[2] OMG. "A Discussion of the Object Management Architecture". Disponible en http://www.omg.org/technology/documents/formal/object_management_architecture.htm

[3] OMG. "CORBA Services Specification". Disponible en http://www.omg.org/technology/documents/formal/corba_services_available_electro.htm

[4] OMG. "CORBA Facilities Specification". Disponible en http://www.omg.org/technology/documents/formal/corba_facilites_specific.htm

[5] OMG. "Domain Facilities". Disponible en http://www.omg.org/technology/documents/formal/

[6] I. Jacobson, G. Booch, J. Rumbaugh. "The Unified Software Development Process". Addison Wesley. 1999.

[7] I. Jacobson, G. Booch, J. Rumbaugh. "The Unified Modelling Language User Guide". Addison Wesley. 1999.

[8] L. Bass, P. Clemence, R. Kazman. "Software Architecture in Practice". Addison Wesley. 1999.

[9] OMG. "UML Profile for CORBA Specification". Disponible en http://cgi.omg.org/cgi-bin/doc?ptc/00-10-01

[10] OMG. "Trading Object Service". Disponible en http://www.omg.org/technology/documents/formal/trading_object_service.htm

[11] OMG. "OMG Healthcare Domain Task Force". Sitio web en http://www.omg.org/homepages/healthcare/index.htm

[12] OMG. "OMG CORBA Component Model Specification". Disponible en http://www.omg.org/techprocess/meetings/schedule/CORBA_Component_Model_RFP.html

[13] OMG. "OMG Interoperable Naming Service Specification". Disponible en ftp://ftp.omg.org/pub/docs/format/00-11-01.pdf

[14] J. Mason, S. Sutton. "Education Working Group: Report of Deliberations". Disponible en http://www.ischool.washington.edu/sasutton/dc-ed/Dc-ac/DC-Education_Report.html

[15] P. Dodds. "ADL Sharable Content Object Reference Model (SCORM)". Disponible en http://www.adlnet.org/Scorn/docs/SCORM1.1.zip

Plataforma Distribuida para Monitorización y Control de Sistemas B2B con restricciones de tiempo real

Justo Hidalgo, Vicente Orjales, Víctor Carneiro
Departamento de Tecnologías de la Información y las Comunicaciones, Universidad de A Coruña.
Campus de Elviña S/N, A Coruña.
Teléfono: +34 981 167000 Ext.1213 Fax: +34 91 2775860
E-mail: {jhidalgo, vorjales}@denodo.com, viccar@udc.es

Abstract. *Business-to-business applications have inherent temporal and load restrictions which require non-intrusive monitoring and control systems for an adequate management, therefore the necessity of lightweight agents. Nevertheless, the monitoring architecture must allow several non-trivial characteristics such as filtering, scope, scalability and load distribution. This work presents an architecture which permits rapid development of highly scalable management platforms, with configurable distribution of responsibility between the agents and the manager system. This is obtained by creating and using an XML-based object-oriented management information model and a communications framework, thus allowing any extension of the architecture simply by adapting it to the management protocol for a particular domain. This framework has proven its usefulness in the management of a real world B2B financial trading system.*

1 Introducción

El comercio electrónico entre empresas o B2B (*business-to-business*) se está convirtiendo en uno de los ejes de las comunicaciones mercantiles en Internet. Los sistemas B2B conforman el subconjunto del área de comercio electrónico con mayor relevancia en los últimos tiempos. Tal y como se describe en [1], el mercado B2B en EEUU se incrementará de 109.000 millones de dólares en 1999 a 2.7 billones de dólares en 2004.

La competencia y las necesidades propias obligan a realizar aplicaciones y servicios B2B a una velocidad creciente. La complejidad y al mismo tiempo criticidad de estos sistemas hacen más necesarios que nunca su monitorización y control.

Las aplicaciones de gestión de sistemas B2B requieren de una arquitectura genérica y modular, debido a la actual heterogeneidad de estos. Por otra parte, la intrusión de la monitorización en estos sistemas ha de ser mínima, debido a que muchos de ellos integran elementos ya existentes que trabajan en condiciones extremas (tiempo real blando con restricciones temporales y espaciales) [2], lo cual invalida la mayoría de las soluciones existentes en el mercado que sobrecargarían el propio sistema gestionado. Sin embargo, esta monitorización y control es absolutamente necesaria en estos sistemas, debido precisamente a la alta heterogeneidad, débil acoplamiento y alta criticidad de sus componentes, lo que dificulta en gran medida la observación de la aplicación en conjunto. Se necesita por tanto un modo de gestión que nos permita esa visión general, pero sin agredir el comportamiento propio de la aplicación.

En este trabajo se presenta un *framework* de gestión cuya principal característica es la independencia del agente, permitiendo que tenga diferentes grados de inteligencia y distintos niveles de intrusismo sobre los recursos que monitoriza y controla. Una vez extendido con el protocolo de gestión que el agente requiera el *framework* permite un rápido despliegue de nuevas plataformas de gestión.

Aunque el *framework* se ha desarrollado íntegramente en Java no presupone nada sobre el lenguaje de programación utilizado para el agente. De esta forma se evita el inconveniente de otras soluciones actuales como JMX (*Java Management eXtensions*) [3] que exige el utilizar la plataforma Java también en el agente, con la consecuente sobrecarga que ello puede introducir sobre los elementos de aplicación que se desea gestionar, inadmisible en dominios con restricciones de tiempo real como en el caso real mostrado en este trabajo.

La independencia del agente de gestión nos permite utilizar el *framework* para construir soluciones de gestión que utilicen el agente con el grado de intrusismo y nivel de inteligencia que mejor se adapten a nuestras necesidades.

La segunda de las características más relevantes es la utilización de un modelo de información de gestión orientado a objetos, lo cual facilita el modelado del sistema a gestionar, importante en sistemas complejos como ocurre en el caso del B2B. La mayoría de las soluciones de gestión actuales se basan en el protocolo SNMP (*Simple Network Management Protocol*) [4], lo cual lleva implícito un modelo de información de gestión plano, dificultando el modelado, y por tanto

gestión, de sistemas complejos. El modelo utilizado recoge los conceptos del propuesto en el ámbito OSI (*Open Systems Interconnection*) [5] pero lo extiende mediante XML (*eXtended Markup Language*) [6] y Java, eliminando la principal carencia de la solución de gestión OSI: lo limitado de las herramientas y soluciones que la soportan en comparación con otros más extendidos como SNMP.

En la sección segunda se estudia más en detalle la arquitectura implementada y las tecnologías y componentes empleados en ello. A continuación se expondrá la aplicación de la misma para un caso concreto real; la gestión de un sistema B2B de intermediación financiera con restricciones de tiempo real blando. Para extender el *framework* en este entorno resultó un factor clave su independencia del protocolo de comunicaciones, ya que las restricciones del entorno obligaron a utilizar un agente lo más reducido posible con un protocolo de comunicaciones binario muy compacto, características que invalidaban la mayoría de las soluciones de gestión existentes en el mercado como las antes descritas.

2 Arquitectura del Framework de Gestión

En la arquitectura de gestión tenemos tres componentes básicos como se puede apreciar en la Fig. 1. El primero y más importante es el Servidor Mediador (MS, *Mediator Server*). Se trata de un pool de gestores (*Managers*), cada uno de los cuales se ejecuta sobre una máquina virtual Java diferente, logrando una escalabilidad alta por el débil acoplamiento existente. Además en el MS encontramos un *Manager Factory* responsable de lanzar nuevos gestores a medida que son necesarios.

Los clientes de gestión son sistemas ligeros (*thin clients*) cuya única finalidad es ofrecer una interfaz al operador para visualizar y operar sobre la información gestionada.

En cuanto al agente, esta arquitectura es capaz de manejar cualquier agente de gestión independientemente de su nivel de inteligencia. Para ello ha de implementarse el protocolo de comunicaciones de gestión adecuado a través de una serie de interfaces que el *framework* nos brinda.

2.1 Servidor Mediador

Es la pieza principal de la plataforma de gestión construida y en la que se centra la arquitectura aquí

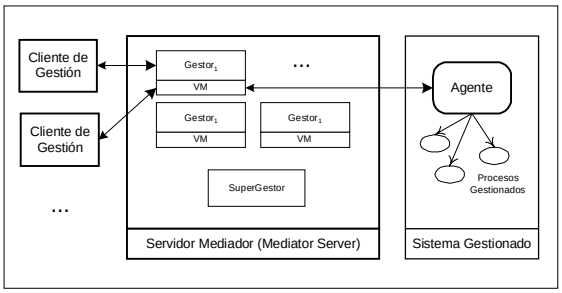


Figura 1: Arquitectura General del Sistema

descrita. Se trata del *middleware* de gestión, responsable de la obtención de la información de los objetos gestionados proporcionada por los agentes y su filtrado y comunicación a los clientes. Además también es el responsable de transmitir a los agentes las peticiones necesarias para llevar a cabo las operaciones solicitadas por los clientes.

Para cada cliente conectado mantiene un *proxy* [7] que determina la información que le interesa a dicho cliente y por tanto la única que le será enviada y sobre la que podrá operar. Además también puede mantener diferentes niveles de log de la información de gestión enviada por el agente.

Aunque ya en la primera versión del sistema se desarrolló una plataforma de gestión completa, el núcleo de la arquitectura consiste en un *framework* de clases Java para gestión de sistemas, de manera que la plataforma final no es más que una extensión del mismo.

La idea ha sido la de construir un *framework* con capacidades avanzadas de gestión de sistemas mediante el ensamblaje de otros más simples (Fig 2). En concreto se han combinado un *framework* para el tratamiento de información de gestión basado en XML y otro para comunicaciones, tanto de gestión (entre gestor y agente) como entre cliente y gestor. Las principales características del *framework* resultante son: manejar un modelo de información de gestión orientado a objetos basado en XML; y la independencia del protocolo de comunicaciones. Como ya se mencionó, junto con el *framework* se proporciona una implementación por defecto del mismo en la que tanto agente de

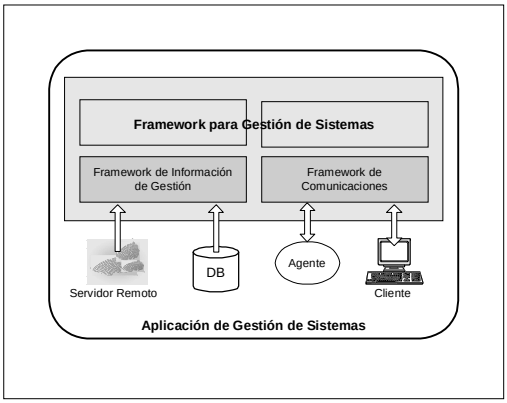


Figura 2: Composición del framework de gestión.

```
        <xsd:keyref name="keyContainedClassName" refer="keyClassName">
            <xsd:selector xpath="ManagementInformationModel/NameBinding"/>
            <xsd:field xpath="subordinateObjectClass"/>
        </xsd:keyref>
    </xsd:element>

    <!-- ******************************************************* -->
    <!-- Comparaciones admitidas por un atributo.       -->
    <xsd:simpleType name="AttributeMatchingOperation">
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="EQUALITY"/>
            <xsd:enumeration value="ORDERING"/>
            <xsd:enumeration value="SET-INTERSECTION"/>
            <xsd:enumeration value="SET-COMPARISON"/>
        </xsd:restriction>
    </xsd:simpleType>

    <xsd:simpleType name="AttributeMatchingOperationList">
        <xsd:list itemType="AttributeMatchingOperation"/>
    </xsd:simpleType>
```

Figura 3: Fragmento de la definición de SM3L mediante XML Schema.

gestión como cliente han sido también desarrollados en Java, aunque podrían ser desarrollados en cualquier otro lenguaje y plataforma.

En las siguientes secciones se describen en más detalle estas características principales. En la sección 3 se detalla la extensión de este *framework* para una solución avanzada de monitorización y control de sistemas B2B en tiempo real.

2.2 Modelo de Información de Gestión

El componente más importante de cualquier plataforma de gestión es el modelo de información de gestión. Se trata de la representación de los recursos gestionados y que determina por tanto la capacidad de control y monitorización de los mismos.

Para la elaboración del modelo de información de gestión se ha desarrollado un lenguaje de modelado orientado a objetos denominado SM3L (*System Management Modelling Markup Language*). Este lenguaje parte del definido en la norma ISO X.700 [8] eliminando ciertos aspectos del mismo como algunas etiquetas dependientes del protocolo CMIP (*Common Management Information Protocol*) y extendiéndolo mediante XML. La definición del propio lenguaje ha sido utilizando *XML Schema* [9], un estándar alternativo a los DTD (*Document Type Definition*) que ofrece unas capacidades considerablemente superiores como mayor soporte de tipos de datos, restricciones de integridad o características de orientación a objetos. En la Fig 3 se aprecia un fragmento del XML Schema que determina la validez de un modelo de información de gestión.

El utilizar XML como base para el modelo de información de gestión dota a nuestra plataforma de algunas de sus principales ventajas. En primer lugar separa el modelo de información de la visualización del mismo; es posible obtener diferentes vistas del modelo y a diferentes niveles. Por otra parte facilita el intercambio de información de gestión con cualquier otra plataforma de gestión como WBEM [10] (*Web-based Enterprise Management*) u OSI. En ambos casos la transformación del modelo de información en SM3L tiene lugar mediante un conjunto de reglas XSL (*eXtended Sylesheet Language*) (Fig 4). Para cada conversión o transformación a la que queramos someter nuestro modelo definimos un conjunto de reglas independiente y que evita cualquier necesidad de programación [10].

Junto con el modelo de información basado en XML se desarrolló un *framework* de componentes que facilita el acceso al mismo proporcionando a clases de niveles superiores una representación de los elementos gestionados como objetos Java independientemente de su naturaleza real. Este *framework* define interfaces para el acceso a información de gestión independientemente del medio en el que se encuentre almacenada. Es posible su recuperación desde ficheros XML, bases de datos relacionales mediante JDBC (*Java*

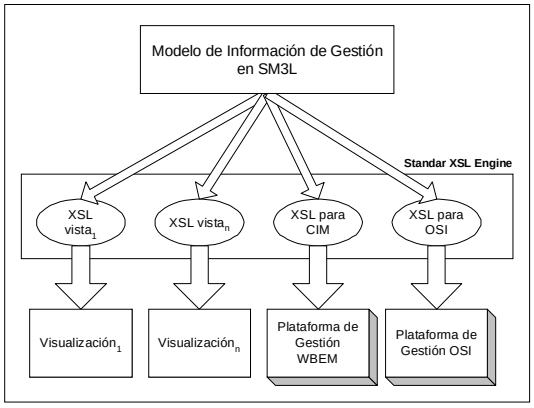


Figura 4: Transformación del modelo SM3L mediante XSL

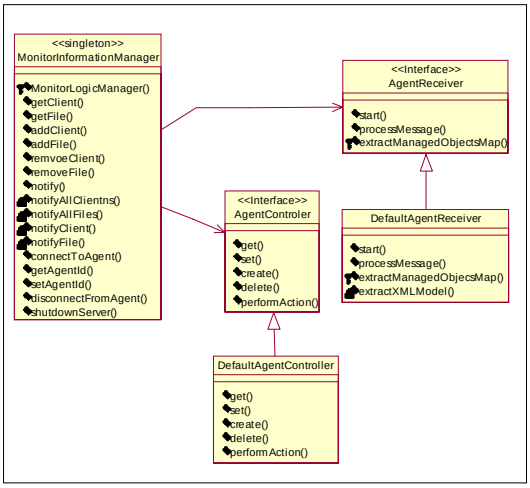


Figura 5: Interfaces de comunicación de gestión.

DataBase Connectivity), servidores HTTP remotos o cualquier otro medio. En la implementación proporcionada por defecto y utilizada en este trabajo el modelo se lee desde un fichero XML utilizando la API DOM (*Document Object Model*) [12].

2.3 Agente de Gestión

Como ya se mencionó, uno de los principales aspectos de la arquitectura aquí presentada es su independencia del protocolo de comunicaciones de gestión y por tanto del nivel de distribución y el agente utilizado. Es decir, en lugar de proporcionar una implementación de un protocolo concreto ofrece unas interfaces que el implementador del protocolo (o sea, el desarrollador que hace uso del *framework*) deberá cumplir. De esta forma la arquitectura no presupone nada sobre las capacidades del agente y resulta totalmente válida para la construcción tanto de soluciones centralizadas como distribuidas.

Las interfaces para comunicaciones de gestión proporcionan los métodos para solicitar cualquier operación del agente y responder a cualquiera de sus notificaciones. Dependiendo del nivel de autonomía del agente la implementación de las mismas será más o menos costosa (cuanta mayor sea la autonomía del agente, menos responsabilidades recaerán sobre el gestor).

En la implementación por defecto se provee un agente básico en Java que se limita a implementar las operaciones GET, SET, CREATE y DELETE sobre elementos gestionados tal como se define en el modelo de gestión OSI incorporando capacidades de alcance y filtrado [13]. El protocolo de comunicaciones de gestión proporcionado en esta implementación por defecto se basa en la transmisión de información XML y soporta tanto comunicaciones síncronas como asíncronas.

2.4 Cliente de Gestión

La definición del cliente no entra dentro del *framework*. En la implementación por defecto se provee un cliente basado en interfaz web. Esto se logró instalando un contenedor web (en concreto se utilizó el *Apache Tomcat*) [14] responsable de la ejecución de las JSP (*Java Server Pages*) que conforman la capa de presentación [15]. Sobre el mismo se desplegaron JSPs que a través de una serie de JavaBeans accedían a la información de gestión en XML. Las JSPs transforman dinámicamente el modelo XML en páginas HTML accesibles para el operador desde cualquier navegador web a través del que pueden llevar a cabo la gestión remota del sistema

3 Extensión del Framework para un sistema B2B real

Tal y como se ha comentado anteriormente, el *framework* de gestión ha sido extendido para abordar la monitorización y control de un sistema B2B de intermediación financiera en el mercado de renta fija española. Este sistema se ocupa de mediar en todas las operaciones que se realizan entre empresas permitiendo operaciones de compra, venta, posición en demanda y oferta, etc.

Un servidor atiende un conjunto escalable de clientes de mercado, cada uno de los cuales tiene capacidad de operación sobre el mercado de renta fija. El servidor se comunica mediante líneas punto a punto con el módulo que se ocupa de recibir todas las operaciones de los clientes, denominado *Trading System*, responsable de interaccionar con el sistema de *Matching* que empareja todas las operaciones, y las procesa para su posterior ejecución en el mercado de renta fija. Todos los módulos son sistemas replicados con capacidad de tolerancia a fallos y distribución automática de carga. Nuestra extensión del *framework* de gestión ha de ser capaz de monitorizar y controlar servidores, *Trading System* y sistema de *Matching*.

La gestión se ha visto sometida a restricciones temporales y espaciales de la aplicación que obligaban a minimizar la carga añadida a la aplicación gestionada. Por otra parte es vital que el ancho de banda consumido por las aplicaciones de gestión sea mínimo. En estas condiciones el *framework* ha resultado la mejor alternativa gracias al débil acoplamiento entre el agente y el resto del sistema, lo que permitió su extensión para la creación de una plataforma con un protocolo de comunicaciones propietario, optimizado para la situación descrita. El agente que utiliza este protocolo es ligero minimizando el intrusismo en la aplicación B2B. En cuanto al cliente de gestión se trata de un *front-end* realizado con los componentes

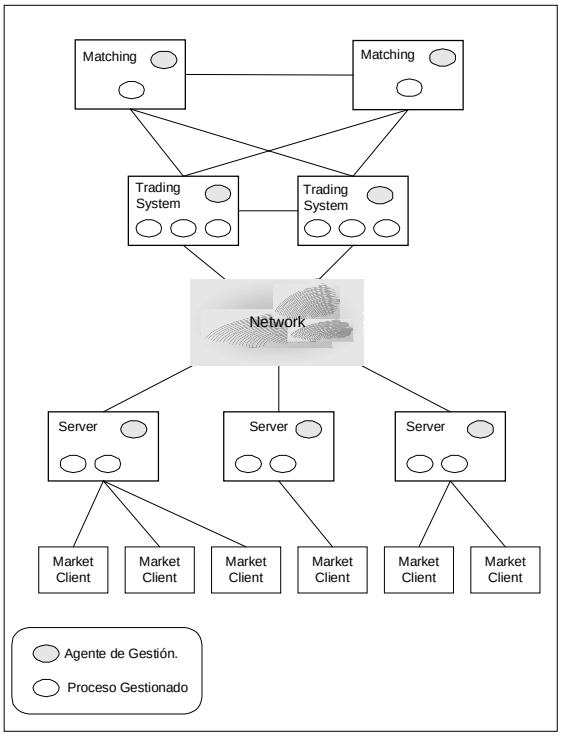


Figura 6: Sistema B2B para el que se ha extendido la arquitectura de gestión.

gráficos Swing de Java. El modelo de información de gestión se almacena en ficheros XML como en la implementación por defecto del *framework* descrita en el apartado anterior.

4 Conclusiones y trabajo futuro

En este proyecto se ha desarrollado una plataforma de gestión que permite crear rápidamente aplicaciones de monitorización y control de sistemas B2B, gracias a su composición de diferentes *frameworks* [16] [17] perfectamente instanciables; además, debido a su capacidad de distribución de carga -gracias a la escalabilidad de la arquitectura, con una máquina virtual Java para cada gestor- y a la distribución de responsabilidad entre módulos, las herramientas creadas con este *framework* se ajustan absolutamente a las necesidades de gestión particulares de cada B2B; a esto hay que añadir la utilización de XML como lenguaje de intercambio y almacenamiento de información en entornos heterogéneos, que facilita tanto la generación de diferentes vistas de los sistemas gestionados, como la exportación del modelo de información desarrollado en SM3L a otras plataformas de gestión.

La plataforma se ha implementado con éxito en la herramienta de gestión de un sistema real de intermediación financiera, con unas restricciones máximas tanto temporales como de carga. Ha sido crucial para ello la independencia del protocolo de comunicaciones que nos brinda el *framework*.

El trabajo futuro consiste en optimizar las prestaciones del sistema, que actualmente dependen del intérprete Java HotSpot [18]. Por otra parte, se ha iniciado ya el estudio de la viabilidad de J2EE como tecnología a utilizar en la base del *framework*, de manera que nos proporcione directamente tolerancia a fallos [15] (extendiendo la ya existente), capacidad transaccional atómica y distribuida, y un entorno multiusuario seguro.

Agradecimientos

Los autores desean agradecer a David Sánchez y Mª Cruz Martín la ayuda y asesoramiento prestado en el desarrollo de esta arquitectura.

Referencias

[1] The Report on European E-Business. Copyright 2000 AMR Research Inc.

[2] Carol Wallace, Mira E. Genser," Gartner Sets Industry Standard for Real-Time Monitoring for B2B E-Business". GartnerGroup Technical Report.

[3] Java Management Extensions. http://java.sun.com/products/JavaManagement

[4] Willian Stallings. "SNMP, SNMPv2, SNMPv3 and RMON 1 and 2". Addison Wesley, 3ª Edición

[5] Lakshi Raman. "OSI Systems and Network Management". IEEE Communications Magazine. Marzo 1998.

[6] Elliotte Rusty. "The XML Bible". IDG Books. Agosto 1999.

[7] Gamma, Erich, Richard Helm, Ralph Johnson, and John Vlissides. "Desing Patterns,Elements of Reusable Object-Oriented Software".Addison-Wesley. 1995

[8] ISO/IEC 7498-4. "Basic Reference Model – Part 4: Management Framework". 1992

[9] David C. Fallside. "XML Schema Part 0: Primer". W3C Recommendation. World Wide Web Consortium. 24 Octubre 2000.

[10] Robin Cover. "DMTF Common Information Model - CIM". OASIS. Junio 2000.

[11] James Clark. "XSL Transformations (XSLT) version 1.0". W3C Recommendation. World Wide Web Consortium. Noviembre 1999.

[12] Armaud Le Hors. "Document Object Model(DOM Level 2 Core Specification" World Wide Web Consortium. Noviembre 2000.

[13] Baha Hewrai. "GDMO: Object Modeling & Definition for Network Management" Technology Appraisals. Junio 1995.

[14] The Jakarta Project. http://jakarta.apache.org

[15] Subrahmanyam Allamaraju et al. "Profesional Java Server Programming J2EE Edition". Wrox Press. 2000.

[16] M.E. Fayad, D.C. Schmidt, R.E. Jonson. "Building Application Frameworks". Wiley. Computer Publishing.1999

[17] M.E. Fayad, D.C. Schmidt. "Domain-Specific Application Frameworks". Wiley Computer Publishing. 2000

[18] Java HotSpot Technoloy. http://java.sun.com/products/hotspot

Implementación de un protocolo de gestión cooperativa distribuida de redes con interfaz WEB y movilidad

Alvaro Suárez y Mario Marrero
GAC (Grupo de Arquitectura y Concurrencia)
Departamento de Ingeniería Telemática, Universidad de Las Palmas de Gran Canaria
Campus de Tafira S/N, Pabellón C (219), 35017 Las Palmas
Tel.: 928-45 89 70, Fax: 928-45 12 43
E-mail: alvaro@dit.ulpgc.es mario@ulpgc.es

Abstract. *Nowadays the management of telecomunication networks is a very important task. Firstly a centralized model of management was used. A Client-Server programming scheme was also used for programming the management applications. In order to solve these problems distributed management applications have been proposed some of them using WEB interfaces. In this p aper we propose a novel management architecture that combine a hierarchical and distributed vision of the network. Each domain of the network is managed in a distributed way by the application software and all the domain managers can share information of theirs domains in order to cooperatively achieve the global administration of complex and large networks. We also think that our strategy is a novel one because it combines a WEB interface, a cooperative module and mobility functions in order to facilitate the cooperation among the different system managers. We present the formal specification (using different tools, part of them implemented by our research group), and the verification of some fundamental properties of the protocol. We have tested that our protocol is efficient and it is safe and also it can support faults in the managed resources. When a fault is discovered all the other software applications are inmediatly reconfigured for continuing the management*.

1 Introducción

La creciente evolución técnica experimentada por las redes de comunicación modernas ha hecho que cada vez más usuarios utilicen sus servicios. Para soportar esta gran cantidad de usuarios es necesario gestionar los recursos de comunicación de forma eficiente para proporcionar Calidad de Servicio aceptable. La gestión de sistemas informáticos es el proceso de control de una red de datos compleja con el propósito de maximizar su eficiencia y productividad. Un Sistema de Administración de Redes [1] es un conjunto de hardware y software diseñados para realizar la tarea fundamental de la gestión de sistemas informáticos.

Las aplicaciones software de gestión de red se componen de las siguientes partes: modelo y arquitectura, protocolos de comunicación, información de gestión, modelo de abstracciones, interfaz de usuario y la metodología [15].

Si bien en un principio las redes de comunicación eran lentas y se justificaba el esquema de gestión Cliente-Servidor para minimizar el número de mensajes de gestión, hoy en día, al aumentar la velocidad de comunicación, se utilizan aplicaciones distribuidas que pretenden eliminar el cuello de botella de estos esquemas Cliente-Servidor. Sin embargo el diseño de estas aplicaciones es muy complejo y se requiere de su especificación formal para verificar su correcto funcionamiento. En los últimos años se ha experimentado una evolución espectacular en los métodos de gestión de redes. En [15] se presentan varios métodos novedosos como: RMON-2, SNMP-v3, Arquitecturas CORBA, JMX, aplicación de la tecnología de redes activas y gestión WEB, que actualmente se utilizan para la gestión de redes. En concreto el WEBM pretende llevar a cabo la gestión de red utilizando tecnologías WEB. Además, el uso de tecnologías de comunicación móviles se puede aplicar a la gestión de redes para flexibilizar su gestión.

La Internet y la WEB han experimentado un avance espectacular en los últimos años incrementándose cada año el número de usuarios. Este avance ha llevado a la implantación de nuevos servicios y a la redefinición de otros existentes. Uno de estos servicios son el *Groupware* and *CSCW* (trabajo Cooperativo Soportado por Computador) que son dos técnicas que se pueden usar eficientemente para resolver cooperativamente problemas con la ayuda de los computadores [10]. Un hecho importante es que la *WEB* ha cambiado por completo la forma en la que los investigadores piensan en *CSCW* [9].

En este artículo presentamos la especificación formal de un nuevo protocolo de gestión de red distribuido. Además hemos implementado una herramienta de gestión y explicamos su interfaz WEB y sus características cooperativas haciendo que varios gestores de redes puedan colaborar activamente en la gestión de diferentes partes de una red de grandes dimensiones. Por último

presentamos las características de movilidad de la misma haciendo que la gestión de la red sea muy flexible ya que permite su gestión (de determinadas operaciones) desde cualquier punto de conexión a Internet utilizando terminales móviles GPRS o WAP sobre GSM.

En el apartado 2 se describe brevemente los pasos seguidos en la especificación formal del protocolo, su estructura, funciones y las herramientas que se han utilizado para la verificación formal del mismo. En el apartado 3 se describe la implementación del protocolo y la herramienta Web diseñada para el funcionamiento del mismo. En el apartado 4 se realizan algunas consideraciones relacionadas con el trabajo cooperativo de los distintos gestores de sistemas y se finaliza con algunas conclusiones y líneas de trabajo futuro.

2 Especificación del protocolo

En este apartado presentamos las ideas básicas de la especificación formal del protocolo que hemos diseñado. Primero presentamos las ideas generales de diseño de protocolos distribuidos y justificamos nuestra elección y después presentamos la forma en que hemos verificado el funcionamiento del mismo.

2.1 Protocolos de gestión distribuidos

En la Fig. 1 se muestra un diagrama básico de las dos partes de todo sistema de administración. La *arquitectura* del sistema define la estructura de los módulos hardware, mientras que la *aplicación* define la disposición del software del sistema sobre esa arquitectura.

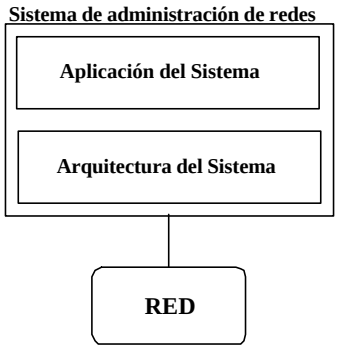


Fig 1. Diagrama básico de un sistema de administración de redes

Básicamente, las reglas definidas para el diseño de la arquitectura pueden ser de tres tipos: *centralizada*, *jerárquica* y *distribuida*. En cualquier caso es clave el concepto de agente de gestión (módulo software instalado en el recurso a gestionar que realiza las tareas locales de gestión). En la práctica la mayoría de las aplicaciones de gestión de red utilizan agentes SNMP debido a que es un estándar basado en TCP/IP que se aplica a la mayoría de los equipos de redes actuales.

Dado que: 1) la gestión centralizada tiene problemas de cuellos de botella, 2) que la gestión pura distribuida es muy difícil de controlar y 3) la gestión jerárquica tiene problemas con el control de equipos "*lentos*" de comunicación. Nosotros en este artículo presentamos un trabajo para el que hemos definido una estrategia de gestión jerárquica (definiendo un Administrador de administradores y dominios). Dentro de cada dominio seguimos una estrategia distribuida. En [11] se puede encontrar una explicación completa de esta estrategia.

Hemos definido esta estrategia porque aprovechamos las ventajas de la gestión jerárquica (alto rendimiento, escalabilidad y control sencillo) y las ventajas de la gestión distribuida en dominios con una cantidad de recursos no elevado (se pueden controlar bien las aplicaciones de administración que funcionan en paralelo).

En este trabajo hemos mejorado el mecanismo anterior considerando la forma en que habitualmente se lleva a cabo la gestión de sistemas con muchos recursos, como son las Universidades: en cada dominio existe un responsable de administración y estos responsables están organizados jerárquicamente. Además hemos pensado que este hecho fundamenta la gestión cooperativa usando el WEB ya que permite implementar mecanismos de cooperación naturales haciendo uso del WEB. Esto es, cada administrador puede informar en cualquier momento de incidencias importantes a sus compañeros del mismo nivel jerárquico o bien a su superior depositando información de gestión en un servidor WEB conocido por todos. Con un servicio de información instantáneo a un terminal móvil (mediante el envío de mensajes cortos que avisaran cada vez que se deposite información compartida en el servidor anterior) se logra que todos los administradores pudieran tener información de gestión, o pudieran consultarla en los servidores WEB, en tiempo real. Dado que la información está en WEB se podría acceder a ella desde terminales móviles.

2.2 Especificación formal del nuevo protocolo

En la Fig. 2 se muestra el esquema jerárquico diseñado así como los principales componentes de nuestra arquitectura de gestión.

En esta arquitectura, inicialmente, todos los sistemas están situados al mismo nivel de jerarquía. Por lo tanto, cada uno se ocupa de un dominio de agentes. A cada uno de estos sistemas los denominamos **SARDs** (*Sistema de Administración de Red Distribuido*). Adicionalmente, uno de estos SARDs actúa como árbitro (puede ser cualquiera de los ellos), y sólo uno de ellos puede estar realizando las funciones de árbitro. Además llamamos **SGA**

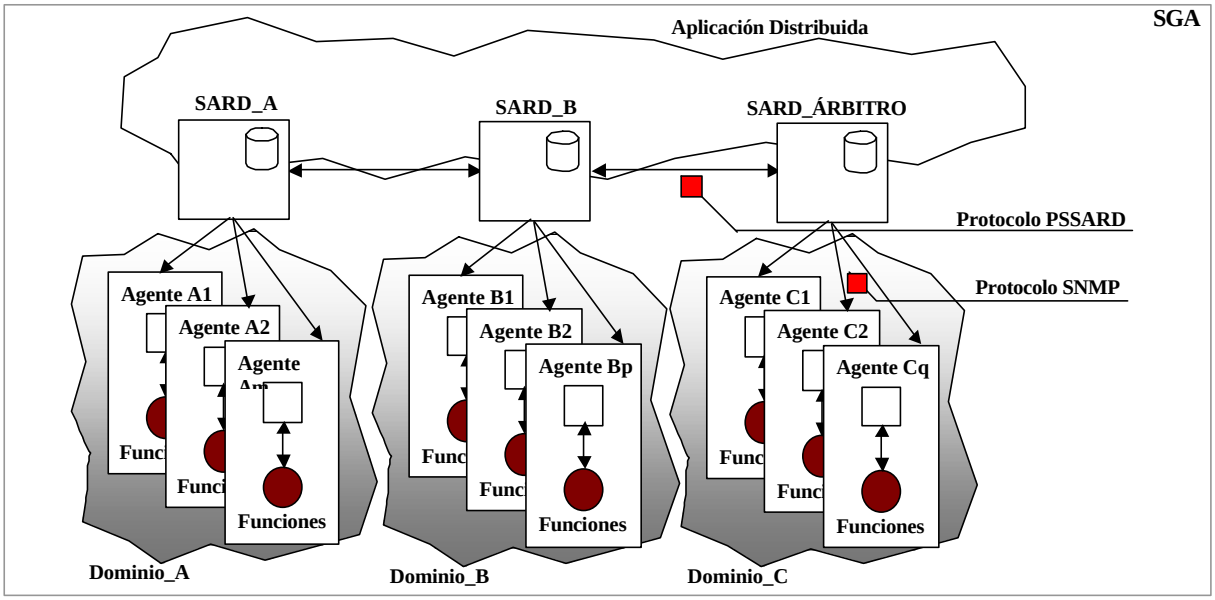


Fig 2. Sistema global de administración (SGA) distribuido con árbitro

(*Sistema Global de Administración*) a todo el conjunto o cluster de SARDs.

En la Fig. 3 se muestran los componentes de un SARD. El *Subsistema de Acceso al Agente* (*SAA*) incluye todos aquellos módulos que actúan sobre el protocolo SNMP y atacan a los agentes obteniendo los datos propios de su dominio. El *Subsistema de Aplicación* (*SDA*) incluye todos aquellos módulos encargados de realizar las partes de la aplicación distribuida correspondiente a este SARD. Cada SARD debe realizar una serie de procedimientos o funciones dentro de su dominio, asignados por el árbitro en funciones. A estos procedimientos los denominamos *PAD* (*Procedimientos de administración sobre dominio*). El *Subsistema de Sincronización y Autonomía* (*SSA*) es el encargado de implementar el protocolo *PSSARD*, mediante el cual se comunica y sincroniza con el resto de SARDs, incluido el árbitro. Adicionalmente, dependiendo de las condiciones del entorno, este subsistema puede hacer que el SARD se convierta en árbitro del sistema global. La parte de *Control* es el encargado de coordinar los subsistemas anteriores.

Informalmente, el protocolo PSSARD se divide en tres fases críticas. *Fase de inicialización*: se lleva a cabo cuando se inicia el sistema por primera vez, o tras un fallo producido por la caída de algún SARD (incluyendo al árbitro). *Fase de operación normal*: En esta fase los SARDs operan normalmente realizando las partes de la aplicación distribuida que le correspondan, es decir realizan los PADs que le han sido asignados. *Fase de test de SARDs*: en la se detecta si algún SARD "ha caído" o ha dejado de funcionar correctamente.

La especificación formal de este sistema se realizó en varias fases utilizando las siguientes herramientas: Herramientas de especificación formal visual en LOTOS [2] llamada SGLOT [4] y GRCLOT [5], una herramienta de visualización gráfica de especificaciones LOTOS [3], y por último herramientas para la verificación formal, el LOLA [6] y el ARA tools [7]. En [11] se puede encontrar los detalles de los resultados obtenidos en la verificación de este protocolo. A continuación presentamos sólo las conclusiones obtenidas que son más interesantes.

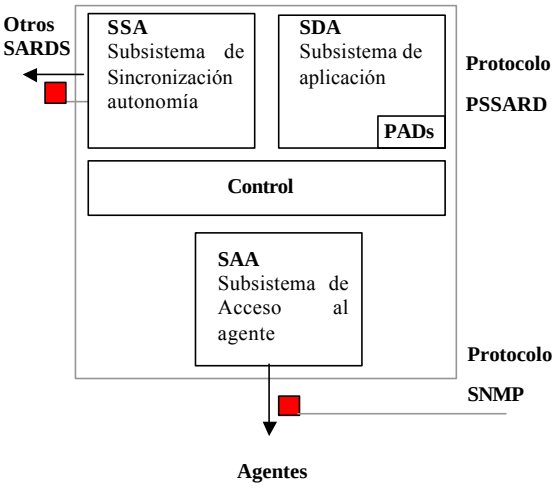


Fig 3. Subsistemas de un SARD

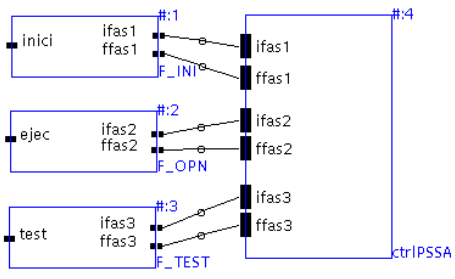


Fig 4. Especificación estructural del protocolo PSSARD a nivel 0 (menor complejidad).

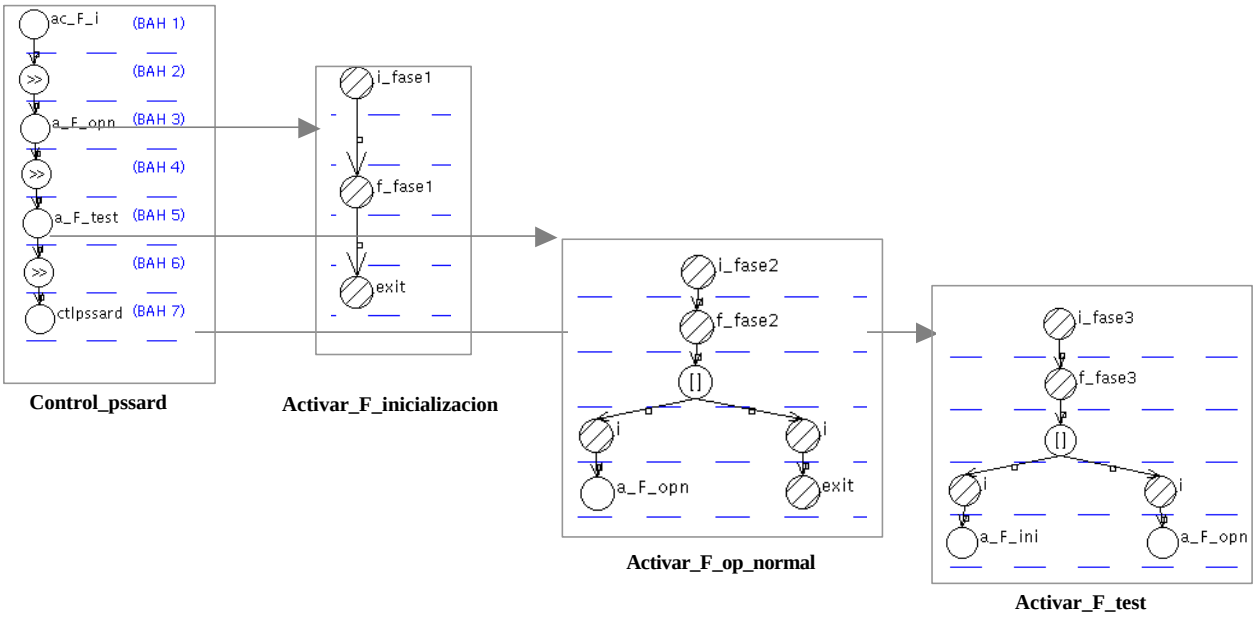


Fig 5. Comportamiento del proceso Control_PSSARD

En la Fig. 4 se muestra la especificación estructural de la arquitectura del PSSARD (usando SGLOT) a nivel de abstracción elevado. En la Fig. 5 se muestra la especificación de la parte de control usando GRCLOT y la herramienta presentada en [3]. Finalmente se pudo comprobar que el protocolo estaba libre de problemas de control de la concurrencia como *deadlock*, *starvation*, etc. En la Fig. 6 se muestra esto mediante un gráfico resultado de la herramienta ARA (el LOLA se usó para verificar estas propiedades cuando la especificación era muy grande en número de líneas de código).

3 Implementación del protocolo

En este apartado presentamos la implementación de una herramienta WEB que lleva a cabo todas las acciones especificadas en el protocolo. Esta herramienta está basada en una previa que se utilizó para la gestión de la red corporativa de la ULPGC [13] [14].

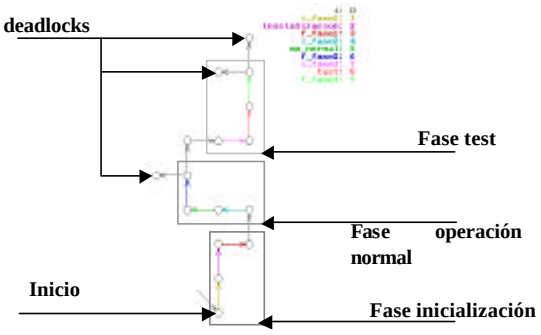


Fig 6. LTS de la especificación pssard1.lot

La implementación del protocolo se ha hecho siguiendo la siguiente metodología:

Para implantar los SARDs se ha desarrollado un demonio (*sardd*) escrito en perl. Dicho demonio se ejecuta en todas y cada una de las máquinas que foman parte del cluster.

La implementación de este demonio sigue exactamente las mismas fases definidas en la especificación formal del protocolo.

Con todo esto se ha diseñado una herramienta WEB cuya interfaz principal se muestra en la Fig.7.

Los objetivos fundamentales de esta herramienta son:

- Posibilitar la administración remota de redes desde el WWW.
- Diseño modular adaptable a todo tipo de cambios en la red, así como a todo tipo de sistemas críticos que soporten el protocolo SNMP.
- Automatizar operaciones básicas de gestión sobre estos sistemas críticos, y operaciones no tan básicas mediante la utilización de PADs.
- Permitir un estudio estadístico del uso de la red.
- Ofrecer varios niveles de seguridad, así como versiones "*read-only*".

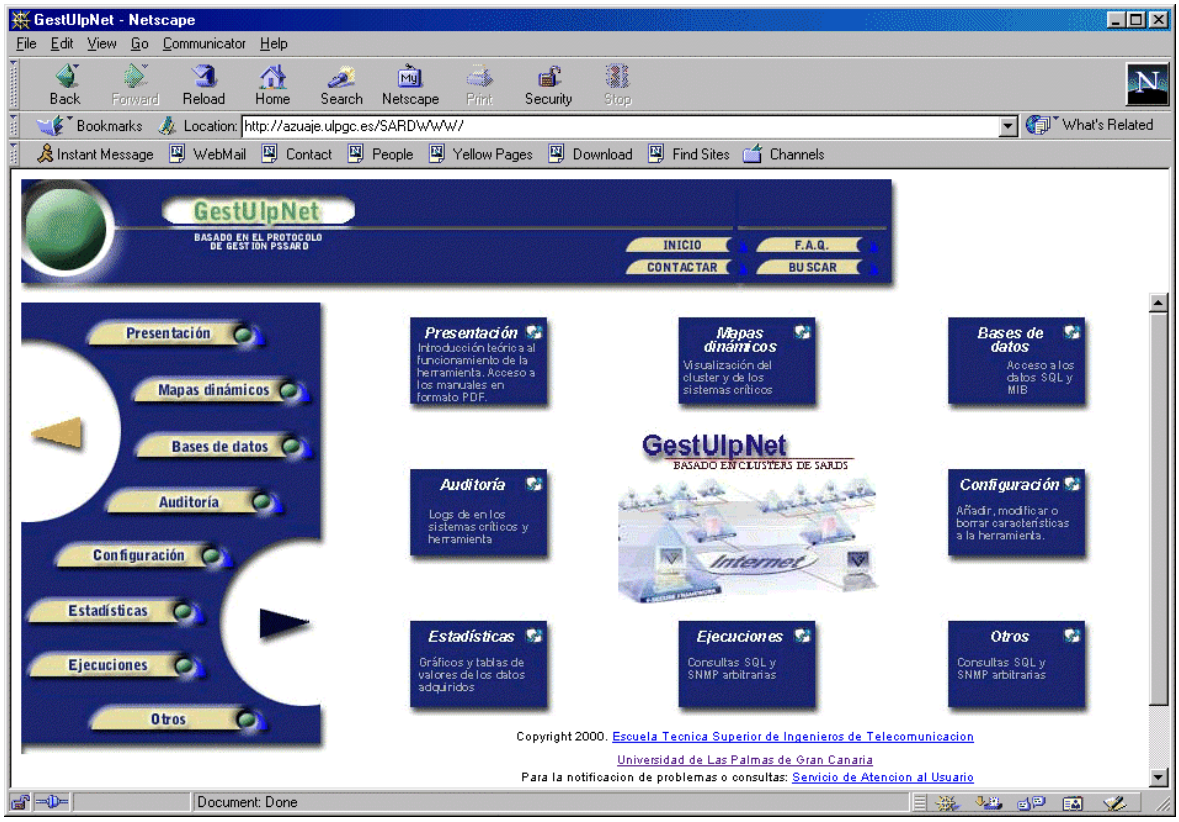


Fig 7.Herramienta Web de gestión

- Permitir el trabajo cooperativo de distintos gestores de sistemas en base a determinados privilegios de seguridad según perfiles sobre dominios.

- Acceso a notificación de fallos vía móviles.

Una vez se accede al URL principal de la herramienta se solicita la identificacion del usuario. Dependiendo de dicha identificacion se podrá acceder a determinadas opciones de la herramienta en funcion de determinados privilegios asignados a dicho gestor o usuario.

Desde la pantalla inicial se tiene acceso a todas las funciones. Dentro de la presentación, se encuentra información general sobre el proyecto, manuales de utilización, artículos relacionados, así como informacion general sobre el protocolo PSSARD y su funcionamiento.

3.1 Módulo de mapas dinámicos

Dentro de esta opción se accede a una captura instantánea de la red, segun la topología que se haya definido. Mediante la definición de topologías, se pueden tener distintas vistas de la red a distintos niveles de abstracción.

Cada mapa dinámico contiene aquellos sistemas críticos que se hayan definido para él: *switches*, *routers*, máquinas servidoras, etc. El único requerimiento para estos sistemas críticos es que deben soportar el protocolo SNMP.

Para cada uno de los sistemas críticos que se encuentren dentro del mapa dinámico, se pueden realizar una serie de operaciones: Ver el status de funcionamiento del mismo, cambiar parámetros de configuración en la topología actual, o ejecutar cualquier PAD que se haya definido para dicho sistema, de forma arbitraria.

De igual forma, el mapa dinámico permite la entrada al área de documentación privada del usuario o gestor. En esta área se puede disponer de un repositorio de documentos, mensajes de otros usuarios o un listado de capturas que nos hayan enviado. Las capturas se definen como fotos instantáneas de la red. Pueden ser realizadas por cualquier usuario y ante cualquier problema detectado, reenviarla para su posterior tratamiento y análisis.

Cada vez que un gestor realiza una captura y se la envía a otro usuario, se le notifica a este mediante un mensaje SMS de móvil y por e-mail.

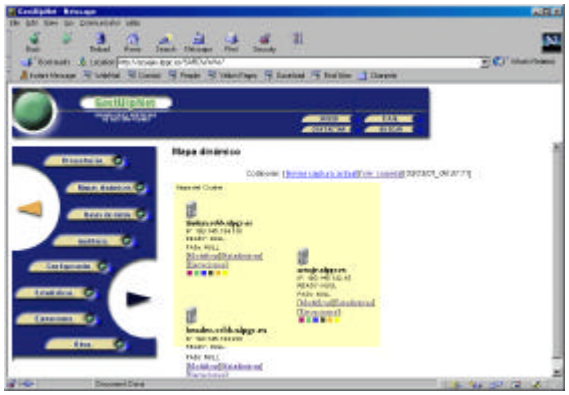


Fig 8. Mapa dinámico de un cluster de SARDs

Fig 9. Ejemplo de código de colores indicando fallos en dos máquinas servidoras

En la figura 8 se representa un mapa dinámico del cluster de SARDs para la Universidad de Las Palmas de Gran Canaria, consistente en tres máquinas situadas en tres edificios distintos del campus de Tafira.

Dentro de la herramienta se pueden definir tantas topologías como sean necesarias, asignándoles privilegios de seguridad para el acceso de los gestores a la misma. De esta forma las topologías pueden ser consideradas distintos dominios de acción en la red global de la organización. El nivel más alto de abstración se constituye como el nivel *WAN* (*Wide Area Network*).

De igual forma el número de sistemas críticos definidos dentro de cada topología es configurable por el gestor. De esta forma, la herramienta está preparada para definir un número no definidido de topologías o vistas, así como un número no definido de sistemas críticos dentro de cada una de estas vistas, por lo que la herramienta está preparada para los contínuos aumentos de la red.

Por otro lado, los mapas dinámicos incluyen un código de colores que ofrecen información visual y rápida de problemas que puedan surgir en los sistemas críticos.

Por ejemplo, en la figura 9 podemos ver de forma intuitiva como las máquinas *ciebas.ulpgc.es* y *capo2.ccbb.ulpgc.es* aparecen en símbolo rojo, indicando que poseen algun tipo de problema o fallo grave (desconexión de la red o caída de la máquina).

3.2 Módulo de Gestión de las bases de datos

En función de los privilegios asignados al gestor de sistemas, éste puede realizar una consulta a bajo nivel de todas las bases de datos implicadas en la herramienta. El motor de bases de datos utilizado es el MySQL. No obstante, existe un listado de consultas "*tipo*" que dispone el gestor para efectuarlas de una forma rápida.

Por otro lado, todas las bases de datos de la herramienta están replicadas en cada una de las máquinas del cluster. Las consultas que se realizan desde la herramienta se realizan sobre la máquina está actuando en ese instante como árbitro, dentro del cluster.

3.3 Módulo de auditoría

Está dividido en dos bloques fundamentales: auditoría del cluster y auditoría de la herramienta en si.

Dentro de la auditoría de la herramienta, se lleva un control exhaustivo de todas las operaciones que realiza el gestor de sistemas dentro de la herramienta: altas, modificaciones, listados, bajas, accesos, etc, sobre cualquier parámetro modificable en la misma: usuarios, máquinas, topologías, etc.

Dentro de la auditoría del cluster se lleva un control del funcionamieto del protocolo PSSARD en cada una de más máquinas donde esté ejecutándose el demonio que lo implementa: estado actual (configuración, inicio, normal y test), etc.

3.4 Módulo de configuración

Dentro de este módulo se pueden configurar parámetros relativos a (Fig. 10):

- Usuarios o gestor de sistemas que pueden acceder a la herramienta o a partes de la misma.
- *Hosts* o sistemas críticos en general que soporten el protocolo de gestión SNMP.
- PADs que se desarrollen. Estos pueden ser tanto sencillas consultas SNMP, como tareas de administración completas que se creen en un determinado lenguaje de script y que posteriormente se puedan ejecutar vía SNMP.

Fig 10. Módulo de configuración

- Topologías (definición de vistas). Crear, modificar o borrar, así como la asignación de hosts a cada topología y su posición.

3.5 Módulo de estadísticas

Para la administración de sistemas, es indispensable disponer de estadísticas de funcionamiento y evolución de todos los sistemas críticos de la red.

Este módulo permite hacer la programación teniendo en cuenta programar sobre qué PADs se realizan las capturas periódicas de datos, y como se guardan en la bases de datos. Adicionalmente, se ofrece una salida gráfica para el estudio y análisis de dichas estadísticas.

3.6 Módulo de ejecución

Se pueden ejecutar PADs arbitrarios e individuales de forma interactiva para obtener su resultado istantáneo desde el Web.

También se pueden ejecutar dichos PADs desde cualquiera de los subgrupos que se hayan definido.

La ejecución de cualquier PAD se constituye como una petición que en ese instante esté actuando según las funciones de árbitro. Este árbitro ejecuta o reenvía la petición a cualquier otro miembro del cluster, dependiendo del dominio donde se encuentre. En su última instancia, el PAD se convierte a una petición SNMP que viaja por la red hasta el sistema crítico y devuelve el resultado. Este resultado se representa en la herramienta web, de forma textual o gráfica.

4 El trabajo cooperativo en SARDWWW

Dado que la gestión de sistemas informáticos en una red WAN es una tarea compleja, es normal encontrar distintos gestores de sistemas cooperando entre sí para la solución de un determinado problema o avería. Muchas de las tareas de gestión en una red de cierta envergadura se pueden realizar de forma centralizada o distribuida.

La herramienta posibilita la cooperación de los gestores vía web, de una forma asíncrona, mediante el envío de mensajes, documentos o capturas instantáneas de la red, con notas asociadas. Los tipos de cooperación que se establecen dentro de la herramienta son:

- Cooperación uno a uno: Mensajes individualizados. Capturas, etc.

- Cooperacion uno a todos: Mediante la utilización de un foro de debate en temas relacionados con la herramienta.

5 Conclusiones

En este artículo hemos presentado la implantación de una herramienta de gestión de red. Previamente habíamos diseñado un protocolo de gestión de red distribuido que fue verificado formalmente para eliminar los errores de diseño y posterior implantación.

El protocolo previamente diseñado hubo de ser modificado (en su fase de implantación) para añadir nuevas tareas de cooperación entre gestores de red. La idea (que es novedosa y la planteamos por primera vez en este artículo) es que diferentes gestores se encargan de controlar la gestión distribuida en dominios con pocas máquinas para que el protocolo de buenos resultados. Entre estos gestores se permite que exista cooperación a través de la WEB usando nuestra herramienta. De esta manera una red compleja se puede gestionar de forma cooperativa de manera muy sencilla.

También hemos añadido el concepto de movilidad para hacer que nuestra herramienta pueda ser usada desde terminales móviles (interactuando con la WEB) y además enviamos mensajes cortos a los diferentes usuarios cuando se detecta un mal funcionamiento.

Esta herramienta ha sido usada para la gestión de la red Corporativa de la Universidad de Las Palmas de Gran Canaria, dando unos resultados excelentes y produciendo muchas mejoras en dicha tarea.

Actualmente estamos trabajando en las modificaciones oportunas al protocolo para contemplar redes de mayor área basadas en Internet. Esto es, estudiar redes que tengan nodos en sitios muy distantes y utilicen IP para conectar los equipos remotos. La ocultación de la latencia de las comunicaciones es crucial para obtener la gestión en tiempo real. Otro tópico en el que estamos interesados es estudiar el acceso inalámbrico a nuestra herramienta desde terminales

móviles de última generación localizados a mucha distancia desde el servidor.

Referencias

[1] Aiko Pras, Network Management Architectures, Tesis, Centre for Telematics and Information Center. University of Twente, Holanda, 1995.

[2] Bolognesi T., Brinksma E. Introduction to the ISO specification language LOTOS. Conputer Networks ISDN systems, pp. 25-59, 1987.

[3] Mario Marrero, Alvaro Suárez, A tool for visualizing LOTOS Behavioural Specifications, 4th FTRTFT, LNCS 1135, pp. 475-478, Sep. 1996.

[4] Mario Marrero, Alvaro Suárez, Elena Carrión and Elsa Macías, SGLOT: A visual tool for structural LOTOS specifications, 5th FTRTFT, LNCS, Lyngby, Sep. 1998.

[5] Elisa Barrena, Generador de Código LOTOS desde Grafos de Comportamiento (GRCLOT), Proy. Fin Carrera, EUITT-ULPGC, Julio 1998.

[6] S. Pavón, D. Larrabeiti, G. Rabay, "LOLA: LOtos LAboratory", Departamento de Ingeniería Telemática. Universidad Politécnica de Madrid. Febrero-1995.

[7] AATOS Research Group, "ARA Tools 2.0, A set of tools for analysing the behaviour of concurrent systems", Electronics Jukka Kemppainen, Agosto-1994.

[8] Greenberg, S., "Computer supported cooperative work and groupware: An introduction to the special edition", *International Journal of Man Machine Studies*, 34(2), pp. 133-143, February. Also describes 34(3), 1991.

[9] Greenberg, S. "Collaborative Interfaces for the Web", In C. Forsythe, E. Grose and J. Ratner (editors), Human Factors and Web Development, Chapter 18, p241-254, LEA Press, ISBN 0-8058-2823-0, 1997.

[10] Grudin, J., "Computer-Supported Cooperative Work: Its History and Participation". IEEE Computer, 27, 5, 19-26, 1994.

[11] Marrero Mario, Suarez Alvaro, Alvarez Alberto, Pego Eva Mª, Macias Elsa, Gonzalez Antonio, "Formal Specification and Verification of a Distributed Network Management Protocol". 5th International Conference on Infomation Systems Analysis and Synthesis. July 31 – August 4, 1999. Orlando, Florida.

[12] Marrero Mario, Suárez Alvaro, "Especificación y Verificación Formal de un Protocolo de Gestión de Sistemas de Administración de Redes distribuido con árbitro". Libro de ponencias de las II Jornadas de Ingeniería Telemática, pp. 319-326, Leganés (Madrid), 15-17 de Septiembre de 1999.

[13] A. Ocón, M. Marrero, A. Alvarez, A. L. Gonzalez, M. Galan y E. Rubio. "GESTnet: Entorno de gestión de red basado en WWW para la ULPnet". Jornadas Técnicas Red Iris 98. Barcelona, 23 y 24 Noviembre. 1998.

[14] Alvarez, Alberto. "Herramienta de gestión para la red de la Universidad de Las Palmas de Gran Canaria (GESTnet)". PFC de la EUITT. Diciembre 1998.

[15] Francisco Fontes, "Contribución a la gestión distribuida, dinámica y flexible de redes utilizando elementos cooperativos", Tesis Doctoral, DIT-UPM, Septiembre 2000.

El Papel del Servicio de Directorio LDAP en Entornos Virtuales Colaborativos *

M. Amor, M. Pinto, L. Fuentes, J.M. Troya
Dpto. de Lenguajes y Ciencias de la Computación, Universidad de Málaga (ESPAÑA)
Campus de Teatinos, s/n. cp. 29071 Málaga (ESPAÑA)
email: {pinilla,pinto,lff,troya}@lcc.uma.es

Abstract *Nowadays, the interest in collaborative virtual environments has increased considerably, probably due to the current technological advances specially on Internet computing. The main challenge in the development of these systems is to construct a configurable, extensible, adaptable and integrated shared environment that takes into account different users preferences. The aim of this paper is to analyze the advantages of using an LDAP Directory Service in the configuration and persistence of virtual environments and propose a solution for virtual office applications. We illustrate our approach through different use cases.*

1. Introducción

Los avances tecnológicos de los últimos años hacen posible el uso de los ordenadores para comunicarse y colaborar con personas geográficamente dispersas formando *grupos de trabajo virtuales*. El trabajo de estos grupos sin pérdida de eficiencia y productividad es posible si se proporciona un *entorno compartido* que integre los recursos necesarios - herramientas individuales, herramientas colaborativas y documentos de distintos tipos, mediante los cuales los miembros del grupo puedan comunicarse y colaborar entre ellos. La interacción entre los miembros del grupo se realiza usando distintos medios de comunicación - texto, gráficos, audio o vídeo, y distintos modos de comunicación - asíncrono o síncrono, con un único envío, múltiples envíos o difusión de información, planificada o informal.

El desarrollo de cualquier sistema distribuido complejo debe procurar producir sistemas con un alto grado de *configurabilidad*, *extensibilidad*, *escalabilidad* y *adaptabilidad*, para poder incorporar de forma fácil y rápida los nuevos requisitos y cambios que afectan continuamente a este tipo de sistemas y para ser capaz de adaptar el sistema, estática y dinámicamente, a restricciones del tipo *preferencias de los usuarios* [1]. En el caso de los *entornos virtuales colaborativos* (EVC) estos objetivos son aún más importantes, debido a que el sistema debe manejar diferentes tipos de usuarios, tamaño del entorno, aplicaciones y dominios de aplicación.

Si estudiamos la literatura [2][3][4][5] sobre EVC, hay muchas variantes en el desarrollo de este tipo de sistemas. Por ejemplo, la mayoría de ellos usan la metáfora de *lugar* para representar el entorno, pero mientras algunos de ellos modelan una habitación de conferencia simple, otros modelan entornos más complejos como edificios donde se puede navegar por diferentes habitaciones o espacios abiertos como una calle o un parque. También se diferencian en el número y tipo de herramientas colaborativas, aunque normalmente hay un conjunto de herramientas por defecto como *chats*, *pizarras electrónicas* y *audioconferencias* que están presentes en la mayoría de los sistemas. Sin embargo, ninguno de ellos propone una arquitectura o marco de trabajo de referencia a partir del cual diseñar este tipo de entornos, y mucho menos proponen un mecanismo a partir del cual se pueda instanciar el entorno *automáticamente*, dependiendo de todos los factores de escalabilidad mencionados.

El EVC más característico es la *oficina virtual* (OV), donde se representa el *entorno de trabajo* de una oficina real, modelando sus recursos típicos - habitaciones, sistemas de ficheros, documentos, herramientas individuales, herramientas colaborativas, trabajadores, y sus modos de trabajo.

Como *entorno de trabajo* se entiende no sólo las aplicaciones y datos con los que el usuario trabaja normalmente sino también la información de configuración que personaliza el entorno para cada usuario. Los usuarios esperan que el entorno *recuerde* información sobre las *preferencias de usuario* independientemente desde dónde y cuando se conecte. Los ejemplos más típicos los podemos encontrar en la Web, donde el objetivo es que los usuarios tengan disponibles, independientemente del lugar de trabajo, información como su lista de marcadores en el explorador Web, su configuración de acceso al correo electrónico o sus listas personales de correo electrónico (ej. *Roaming Access* de Netscape).

*Esta investigación ha sido financiada en parte por el proyecto CICYT TIC99-1083-C02-01, y en parte por la organización "Fundación Retevisión"

En el caso de una OV la información que debe estar disponible de forma independiente a la localización del usuario es mucho más amplia. Algunos ejemplos son la información de conexión (normalmente nombre de usuario y password), la información de control de acceso a los recursos del entorno (ej. listas de acceso) y las preferencias de cada usuario (ej. representación gráfica, combinación de colores, nivel de presencia y nivel de persistencia). La forma de proporcionar toda esta información es utilizar un *almacén estructurado* compartido por todos los componentes de la OV y accesible desde cualquier localización. Tradicionalmente se utilizaban bases de datos aunque hoy en día, sobre todo en aplicaciones sobre la Web, son muchos más utilizados los *servidores de directorios* y especialmente el *Servidor de Directorios LDAP* [6].

Este trabajo es una experiencia real en el uso de LDAP en la organización de OVs propuesto por la "Fundación Retevisión". El diseño del marco de trabajo de EVC sigue el paradigma de orientación a componentes de la plataforma MultiTEL [7] que hemos extendido con la programación orientada a aspectos [1][8], pero que no es objeto de este artículo. El objetivo de este trabajo es justificar el uso de un almacén estructurado y compartido de información en el desarrollo de OVs. En primer lugar vamos a introducir los requisitos de organización de una OV y justificaremos el uso del *Servidor de Directorios LDAP*, destacando las ventajas que proporciona a estos entornos. Para ver realmente su potencia, presentamos diversos casos de uso de LDAP en la configuración y ejecución de una OV. Las conclusiones de este trabajo se pueden extender también a otros EVCs.

2. Requisitos de Organización de una OV

Una OV es un tipo de aplicación de trabajo colaborativo que favorece la formación de equipos de trabajo en los que sus miembros están geográficamente dispersos. Sin embargo, si el nivel de colaboración y de presencia proporcionado por la OV a los usuarios es el adecuado, se favorece también el trabajo en equipo de personas geográficamente cercanas reduciendo así el número de encuentros físicos y agilizando y haciendo más dinámica la colaboración. Entre los requisitos de desarrollo más importantes de un entorno colaborativo cabe destacar como se lleva a cabo la *comunicación*, *cooperación*, *presencia* y *organización*. En este trabajo nos vamos a centrar primero en los aspectos organizacionales de una OV, comenzando por la necesidad de definir un *Directorio de Oficinas Virtuales* (DOV), como un sitio en el cual localizar nuestra oficina y participar en ella. Posteriormente nos centraremos en la *organización interna* de la oficina y por último, en dos propiedades básicas que debe proporcionar cualquier EVC, *persistencia* y *control de acceso*.

2.1. DOV

La forma de configurar y acceder a una OV es buscarla en el DOV e iniciar la aplicación cliente correspondiente. Cada entrada del DOV representará un tipo de oficina, que se describirá mediante los campos *nombre* y *descripción* (Figura 1). Igualmente se indicará el *estado* en el que se encuentra cada servicio de OV, que podrá ser de cuatro tipos: sin configurar (*unbound*), inactivo (*idle*), activo (*active*) o suspendido (*suspended*). La búsqueda de una OV en cualquiera de estos estados se lleva a cabo usando LDAP.

Un servicio *sin configurar* será en realidad una plantilla para definir un tipo concreto de OV, que podrá ser instanciada por un usuario con permisos de organización. Cuando la oficina está organizada, el DOV añade una entrada con estado *inactivo*. Los servicios que están en este estado están listos para ser ejecutados por un usuario iniciador de un servicio con los permisos adecuados. Una vez iniciada, la oficina pasa a estado *activo*, es decir en ejecución, y los usuarios con permisos para unirse a esa oficina podrán hacerlo y colaborar. Por último, una OV en ejecución podría suspenderse para reanudarse en otro momento, por ejemplo tras finalizar la jornada de trabajo. Mientras un servicio está en estado *suspendido* no consume recursos y toda la información relativa a los usuarios o la ejecución del entorno en general, y los datos necesarios para reanudarlo desde el momento en que se suspendió su ejecución, se hará *persistente* almacenándolo en el servidor LDAP.

2.2. Organización Interna de una OV

El segundo nivel de organización de una OV es su *organización interna* modelando lo más fielmente posible una oficina real. Será necesario determinar: (1) las *personas* que constituyen la oficina, (2) su *estructura*, organizando de forma jerárquica las personas en departamentos y grupos de trabajo, (3) la *asignación* de personas a proyectos, (4) el *control de acceso* a los recursos, (5) la *administración* de la oficina y (6) los *documentos* que podrán ser accedidos por los usuarios y aplicaciones con permisos adecuados. Toda esta información es la que caracteriza en un momento dado una OV concreta y por tanto debe ser *persistente*.

2.3. Persistencia

La persistencia es una necesidad en los EVCs. Hemos visto que el DOV debe hacer persistente la OV cuando su estado sea *suspendido*, ya que debe existir incluso cuando no exista ningún usuario conectado a ella. Otros niveles de persistencia son la configuración del entorno para cada usuario o

las acciones realizadas por los usuarios sobre el entorno, por ejemplo, modificaciones de sus preferencias, lista de reuniones planificadas o la creación o modificación de documentos. Para incrementar la flexibilidad del sistema, el nivel de persistencia de cada componente en el entorno debería ser configurable. El servidor LDAP es un almacén apropiado para hacer persistente toda esta información que estará disponible independientemente de la localización del usuario y de los recursos físicos de la propia OV.

2.4. Control de Acceso

Dentro de la OV residirán una serie de recursos (carpetas, documentos, habitaciones, notas, etc.) a los cuales sólo podrán tener acceso un conjunto limitado de personas. El *control de acceso* aparece distribuido entre todos los componentes del entorno, ya que todos los recursos de la OV tendrán propietarios y, por tanto, restricciones de acceso. Existirán componentes que sean dependientes del entorno siendo el sistema quién establece los permisos de acceso (ej. una habitación de reuniones) y habrá componentes propiedad de usuarios específicos que serán los que determinen los permisos de acceso (ej. el despacho de un usuario). Estos permisos podrán cambiarse en tiempo de ejecución (ej. acceso a documentos, a habitaciones, etc.) y se almacenan en el servidor LDAP. Uno de los principales servicios del servidor LDAP es el de autenticación, ya que permite definir los usuarios y sus permisos en el entorno de forma jerárquica y única. La información de autenticación de un usuario podrá ser la misma para todas las aplicaciones y componentes del entorno y estará almacenada en el servidor LDAP.

3. Escenarios de Uso de LDAP en una OV

Para estudiar algunos escenarios de uso del servidor LDAP en el desarrollo de una OV vamos a presentar nuestra experiencia en la creación del DOV y en la organización y configuración de una OV. Veremos también como es posible controlar el acceso de los usuarios a la oficina y a sus recursos y como hacer persistente las preferencias de usuario mediante el uso del servidor LDAP.

En LDAP la información se organiza de forma jerárquica y se almacena en *entradas* del esquema de base de datos del servidor [9]. Cada *entrada* tiene un *nombre distinguido* único dentro del directorio y un conjunto de *atributos*. LDAP define una serie de *entradas* y *atributos* por defecto, que pueden ser extendidos si es necesario. En nuestro caso tenemos que definir nuevos atributos y entradas para almacenar toda la información de *organización*, *configuración*, *control de acceso* y *preferencias de usuarios* en la oficina. Siempre que sea preciso, se indicará el tipo de modificaciones que ha sido necesario realizar sobre el esquema de base de datos.

Los tipos de entrada qué aparecen en los ejemplos que veremos a continuación, son algunas de las más características de los niveles superiores del servidor LDAP:

a) *o*: modela una organización.

b) *ou*: modela una unidad organizativa dentro de la organización (departamentos, filiares, etc.). También es posible usarla para organizar un conjunto de recursos compartidos de las oficinas (personas, grupos, documentos, plantillas de oficinas, oficinas instanciadas, oficinas en ejecución, etc.).

c) *cn*: determina el nombre común de grupos dentro de una organización o unidad organizativa, de personas dentro de grupos, etc.

3.1. Persistencia de OVs

Como vimos anteriormente, el DOV proporciona información sobre los servicios de OVs que un usuario puede organizar, iniciar, conectarse o suspender. Para poder implementar el DOV y hacer *persistente* toda la información sobre las OV usamos las entradas del servidor LDAP.

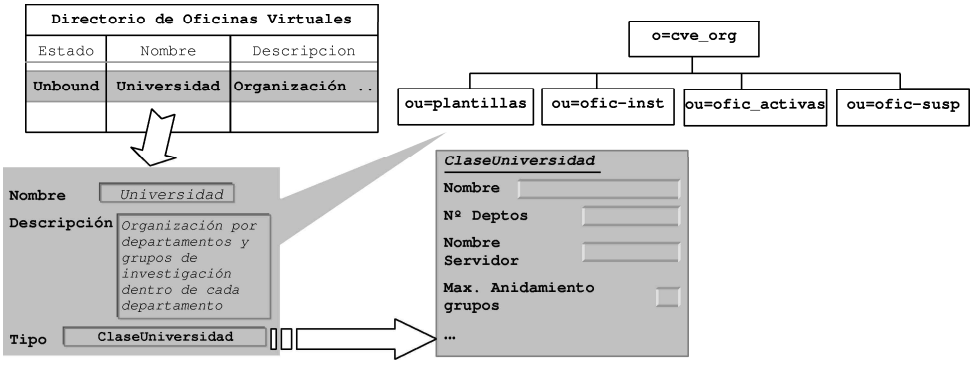


Figura 1: Proceso de Organización de la Oficina Virtual

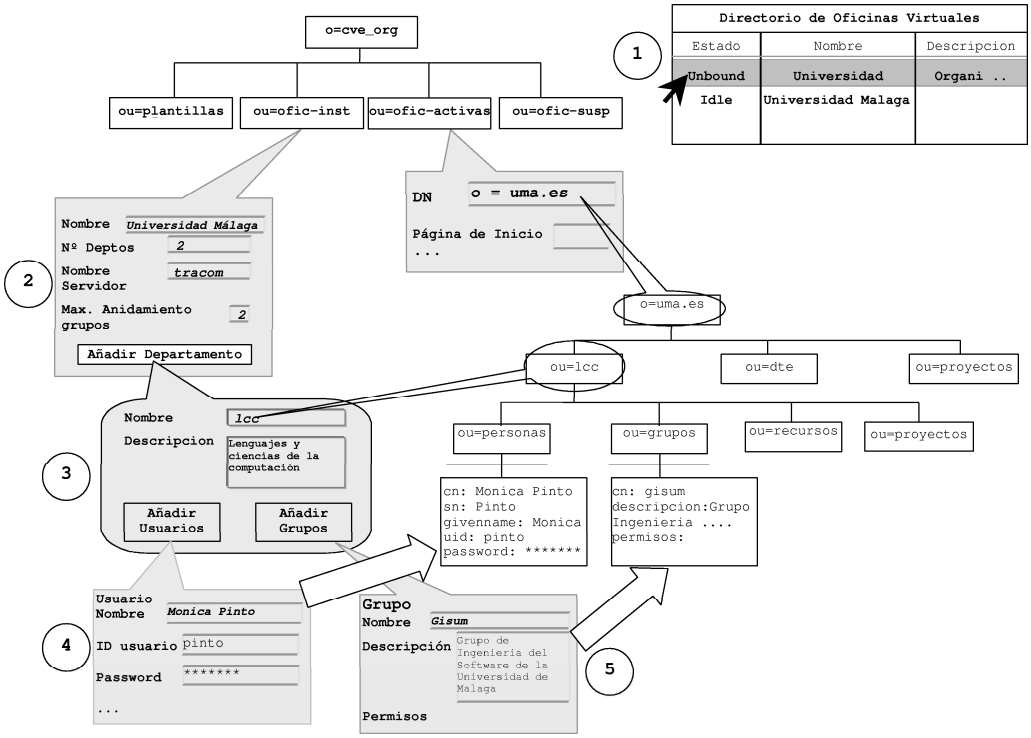


Figura 2: Directorio CVE

En nuestra propuesta, el servidor LDAP almacena un conjunto de *plantillas* que un usuario *organizador* puede utilizar o extender para organizar su propia oficina virtual. Una vez organizada, se podrán crear distintas instancias de la misma OV. La información de las oficinas instanciadas y en ejecución también se almacena en el servidor LDAP. En la Figura 1 podemos ver la organización de las OVs con más detalle.

En primer lugar es necesario crear la estructura jerárquica que mantendrá toda la información de configuración de la oficina. Como vemos en la Figura 1, hay cuatro unidades organizativas principales, *plantillas*, que contiene todas las plantillas creadas, *ofic-inst*, que contiene las oficinas instanciadas, *ofic-activas*, que contiene las oficinas activas y *ofic-susp*, que contiene las oficinas suspendidas. Para hacer persistente toda esta información en el servidor LDAP es necesario extender el esquema de la base de datos con entradas nuevas. Para las plantillas la entrada almacenará el *nombre*, *descripción* y *tipo*de cada plantilla. Por ejemplo, la entrada *Universidad* en la Figura 1. El *tipo* será el nombre de otra *entrada* en el servidor LDAP que contenga como atributos toda la información que el usuario debe proporcionar para instanciar una OV basada en esa plantilla. Esta última *entrada* también será necesario crearla, de forma que hay que añadir una *entrada* diferente en el servidor LDAP para cada plantilla definida. Por ejemplo, la plantilla *ClaseUniversidad* en la Figura 1. El resto de las entradas que hay que añadir se verán más adelante.

3.2. Organización y Configuración de una OV

El siguiente paso sería *instanciar* una OV a partir de una de las plantillas. Para instanciar una oficina concreta a partir de una plantilla, el DOV se comportará como un *cliente LDAP*, accediendo al servidor y mostrando todas las entradas almacenadas en la entrada *plantillas* como *no configuradas*. Una vez que el usuario organizador seleccione la plantilla que desea usar, el componente organizador le solicitará todos los datos que es necesario proporcionar para instanciar una oficina a partir de dicha plantilla. En la Figura 1 el usuario selecciona la plantilla con nombre *Universidad* y el DOV consulta el *tipo* de esa plantilla para saber qué información debe solicitar al usuario para configurar una oficina según esta plantilla. En nuestro ejemplo, el *tipo* de plantilla es *ClaseUniversidad*.

En general, para organizar una OV la información que el usuario organizador debe proporcionar es la siguiente:

1. *Estructura de la oficina.* Dentro de la organización de la OV se modelará su *organización estática*, de forma jerárquica como en cualquier empresa. Al instanciar una OV concreta en el DOV, un usuario con rol de organizador dará de alta todos los departamentos de la oficina, los grupos de cada departamento y las interrelaciones entre ellos.

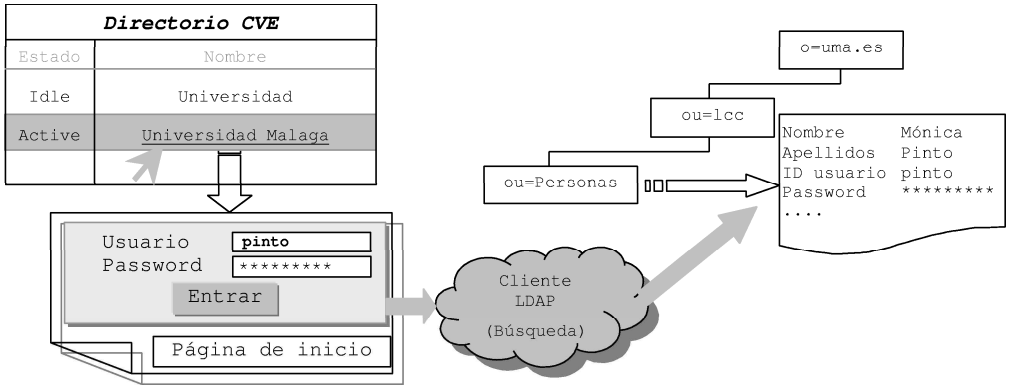


Figura 3: Control de Acceso

2. *Personal.* Los usuarios que forman parte de la oficina. Cada persona tendrá un perfil asociado que identificará tanto información personal como su papel dentro de la oficina, o sea, a qué departamento pertenece, en qué proyecto trabaja en la actualidad, y sus preferencias sobre el entorno. Entre los datos personales, son muy relevantes el *login* y la *password* de conexión a la oficina.

3. *Asignación de personal.* Dentro de una OV se debe incluir mecanismos para la asignación de personal a cada elemento constitutivo de la oficina, tanto a elementos estáticos, por ejemplo departamentos, como a elementos organizativos casuales como proyectos, reuniones, etc.

4. *Gestión de grupos de trabajo.* Es importante agrupar a los usuarios según su categoría profesional, las actividades de colaboración que realicen y los permisos de acceso de qué dispongan. Esto permite gestionar a estas personas globalmente, por ejemplo, enviando un aviso a todos los miembros de un mismo grupo.

5. *Administración de la oficina.* Es necesario establecer los parámetros globales de la oficina y los permisos de acceso del resto de los usuarios a los recursos de la OV.

La Figura 2 muestra paso a paso el proceso de configuración de una OV que modele una Universidad. En primer lugar el usuario selecciona la entrada *Universidad* en estado no configurado (*unbound*). A continuación introduce el *nombre* de la OV (*Universidad de Málaga*), el *número de departamentos* (2), el *nombre del servidor* encargado de implementar las tareas de colaboración (en nuestro caso es un servidor Sametime [10] llamado *tracom*) y el máximo anidamiento de grupos (2). Por último, añade y configura los departamentos, grupos de trabajo y personas que pertenecen a la oficina. En nuestro ejemplo, se añade el departamento *lcc*, donde uno de los grupos es el grupo *gisum* y se da de alta el usuario *Monica Pinto*.

Con la información anterior se crea la OV en el servidor LDAP mediante un árbol jerárquico. La Figura 2 muestra un ejemplo de esta representación, en la que se ha creado la oficina *uma.es* con dos departamentos *lcc* y *dte* y una entrada *proyectos* que representa los proyectos interdepartamentales. En cada departamento tendremos una entrada *personas* con toda la información personal de los usuarios, una entrada *grupos* con los grupos y los permisos de acceso de dichos grupos, una entrada *recursos* con los recursos de la oficina (habitaciones, documentos, etc.) y una entrada *proyectos* con los proyectos propios de cada departamento.

Una vez que la oficina ha sido configurada, se añade una entrada que haga persistente esa configuración en la entrada *ofic-inst* del árbol de configuración. Esa entrada se representa en el DOV como inactiva (*idle*). Cuando un usuario iniciador quiere comenzar la ejecución de una oficina ya configurada, selecciona la entrada (ej. Universidad de Málaga) en el DOV creando una entrada activa (*active*). Además, se añade una entrada en el árbol de configuración bajo *ofic-activas*. Para añadir esta entrada será necesario extender nuevamente el esquema de base de datos del servidor LDAP con una entrada que contenga el *nodo raíz* del árbol que modela la oficina y la *página de inicio* de la misma, así como cualquier otra información que sea necesaria para conectarse a la oficina.

3.3. Control de Acceso

Para unirse a una oficina virtual ya creada, el DOV mostrará toda la información almacenada en la entrada *ofic-activas* del árbol de organización. Cuando el usuario seleccione una oficina en estado activo (*active*) (Figura 3) se comprueba si tiene los permisos adecuados para conectarse. Para realizar la autenticación, en el servidor LDAP debe existir una entrada para ese usuario bajo la entrada *Personas* y debe coincidir el *login* y la *password* introducidas por el usuario con las almacenadas en su entrada de usuario.

Pero el control de acceso no se realiza únicamente cuando el usuario se une a la oficina sino también cuando accede a sus recursos, como

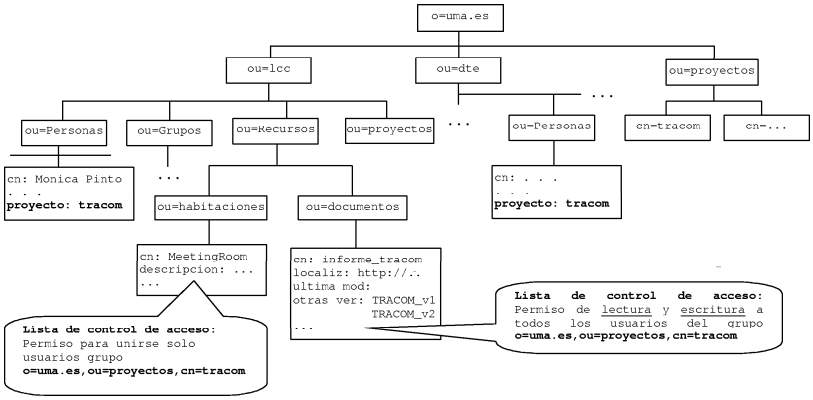


Figura 4: Organización del Control de Acceso

habitaciones, *documentos* y *aplicaciones*. Los permisos establecidos por el usuario que creó la oficina determinarán los recursos a los que cada usuario puede acceder una vez conectado a la oficina.

Siguiendo con nuestro ejemplo (Figura 4), supongamos que se está realizando el proyecto *tracom* entre los departamentos *lcc* y *dte*. Para representar estos proyectos, se creará un grupo de trabajo *dinámico* con nombre *tracom* en la entrada *o=uma.es, ou=proyectos* que agrupa a los proyectos interdepartamentales. Para asignar usuarios a un grupo dinámico en LDAP se establece que los usuarios que pertenecen a este grupo son aquellos para los que uno de los valores de su atributo multivaluado*proyecto* sea *tracom*.

Supongamos también que los miembros del grupo *tracom* están creando un informe semestral del proyecto, que se encuentra almacenado en la entrada *documentos* del departamento *lcc*. Para que todos los miembros del proyecto puedan acceder a ese documento, su lista de control de acceso debe establecer que todas las personas que pertenecen al grupo *tracom* tienen permisos de lectura y escritura sobre él. De la misma forma, se puede reservar una habitación de reuniones (ej. MeetingRoom) para los miembros del proyecto, indicando en la lista de control de acceso de la habitación que sólo los usuarios que pertenecen al grupo *tracom* pueden entrar.

3.4. Preferencias de usuario

La primera vez que un usuario se une a la oficina virtual, la configura estableciendo sus *preferencias de usuario*. Entre las más relevantes, cada usuario podrá establecer el *nivel de presencia* que desea, el *nivel de persistencia*, la *representación gráfica* del entorno y su *combinación de colores*. La utilización del servidor LDAP para almacenar estas preferencias permite hacerlas *persistentes* y disponibles desde cualquier localización. Esto permite que el usuario pueda disponer de su propia configuración del entorno independientemente del lugar de conexión a la oficina. Las preferencias de un usuario se almacenan en su entrada de usuario. Para poder hacer esto se añade a la entrada de los usuarios un atributo *preferencias* cuyo valor será una instancia serializada de un objeto (en nuestro caso un objeto Java) (Figura 5) conteniendo los valores establecidos para cada preferencia por el usuario. Almacenando las preferencias como un objeto serializado en lugar de como atributos independientes, permitimos que estas preferencias sean distintas para cada OV de forma transparente al esquema de base de datos del servidor LDAP. Esto último significa que en el valor del atributo *preferencias* en la entrada para un usuario en la Figura 5, puede ser cualquier objeto serializado.

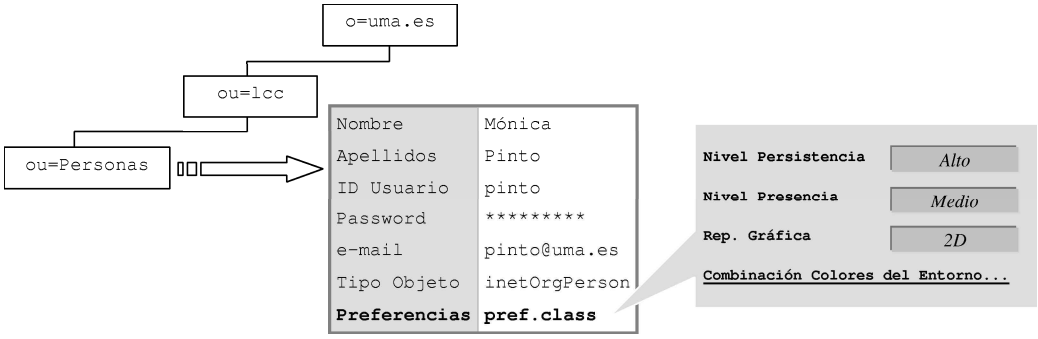


Figura 5: Persistencia de las Preferencias de Usuario

4. Conclusiones y Trabajo Futuro

En este trabajo hemos analizado la utilidad del servidor de directorios LDAP como medio de almacenamiento de la información necesaria para organizar, instanciar y ejecutar una OV. Las principales ventajas que proporciona es la facilidad y flexibilidad que aporta para la definición de nuevas *entradas* y *atributos* y la alteración dinámica en tiempo de ejecución no sólo de la información almacenada en el directorio sino también del esquema de base de datos del directorio.

Otra ventaja proporcionada por el servidor LDAP es que nos permite tener toda la información relativa a la oficina en un mismo sitio a pesar de que dicha información es muy dispar entre sí. Esto hace posible la unión de los usuarios a cualquier OV de forma completamente independiente de la máquina desde la que se conecte. Por último haciendo persistente las preferencias de los usuarios en el servidor LDAP es posible tener configuraciones diferentes del mismo entorno para distintos usuarios de forma rápida y fácil.

Disponemos de un prototipo desarrollado en Java con *Directory SDK for Java* incluido con el servidor LDAP de Netscape (*Netscape Directory Service*). Nuestro siguiente paso será estudiar diferentes tipos de entornos virtuales extrayendo las características más importantes de configuración de cada uno de ellos para proporcionar un conjunto de plantillas que cubran un rango lo más amplio posible de entornos. Otro objetivo será definir e implementar un mecanismo de extensión que permita modificar las plantillas predefinidas para la configuración de cualquier tipo de entorno.

Referencias

[1] M. Pinto, et. al. Collaborative Virtual Environment Development: An Aspect-Oriented Approach. *Próxima Publicación en Proceedings of DDMA Workshop*, Phoenix, Arizona, 2001.

[2] M. Roseman and S. Greenberg. Teamrooms: Network places for collaboration. *Proceedings of ACM CSCW*, 1996.

[3] M. Sohlenkamp and G. Ghwelos. Integrating communication, cooperation and awareness: The diva virtual office environment. *Proceedings of ACM CSCW*, 1994.

[4] H. Shinkuro, et. al. A virtual office environment based on a shared room realizing awareness space and transmitting awareness information. *Proceedings of the 10th annual ACM symposium on user interface software and technology*, 1997.

[5] C. Steinfield, C. Y. Jang, and B. Pfaff. Supporting virtual team collaboration: The teamscope system. *International ACM SIGGROUP Conference on Supporting Group Work (GROUP'99)*, 1999.

[6] M. Wahl,T. Howes and S. Kille Lightweight Directory Access Protocol (v3). RFC 2251, 1997.

[7] L. Fuentes and J. M. Troya. Coordinating distributed components on the web: an integrated development environment. *Software-Practice and Experience*, 31, 2001.

[8] M. Pinto, et. al. Supporting Heterogeneous Users in Collaborative Virtual Environments using AOP. *Enviado a Sixth International Conference on Cooperative Information Systems*, Trento, Italy, 2001.

[9] R. Weltman and T. Dahbura LDAP Programming with Java. Addison-Wesley, 2000.

[10] Lotus Corporation Sametime 2.0 Java Toolkit Developers Guide. 2001.

Análisis y comparativa de las alternativas propuestas para la Gestión Basada en *Web*

Jorge E. López·de·Vergara, Víctor A. Villagrá, Juan I. Asensio, Julio Berrocal
Departamento de Ingeniería de Sistemas Telemáticos. Universidad Politécnica de Madrid.
E.T.S.I. de Telecomunicación. Av. Complutense, s/n. 28040 Madrid
Teléfono: 91 549 57 00, Fax: 91 336 73 33
E-mail: {jlopez, villagra, jasensio, berrocal}@dit.upm.es

Abstract. *The introduction of web technologies in the network management field has contributed with some new ideas that will improve the existing management systems. Some proposals, coming from different organizations, are becoming web based management standards. The differences between these proposals imply the need of studying and comparing them, making it possible to choose the best alternative for certain circumstances. For this, this paper tries to perform this analysis, comparing technologies such as CORBA/JIDM, CIM/WBEM and JMX, using an architectural framework based on the general characteristics that a management system is supposed to have, following the guidelines proposed by some network management organizations.*

1 Introducción

1.1 Motivación

La interfaz *web*, tras su rápido despliegue en el mundo Internet, se ha revelado como paradigma de interfaz de usuario, gracias a sus características que la hacen ser amigable, intuitiva, independiente de arquitectura y con una curva de aprendizaje rápida. Es por ello por lo que está siendo utilizada actualmente por las casas de *software* como interfaz de sus servidores de aplicaciones (Microsoft BackOffice, Lotus Domino, iPlanet Application Server, Oracle...), posibilitando una utilización óptima de los recursos *software* de una compañía. Esta tecnología suele estar basada en el uso de lenguajes como Java y mecanismos de comunicación distribuida tales como DCOM (*Distributed Component Object Model*, Modelo de Objetos de Componentes Distribuidos), CORBA (*Common Object Request Broker Architecture*, Arquitectura Común de Intermediarios de Peticiones de Objetos), RMI (*Remote Method Invocation*, Invocación de Métodos Remotos) o SOAP (*Simple Object Adapter Protocol*, Protocolo Simple de Adaptadores de Objetos), que posibilitan que el usuario interactúe, mediante clientes ligeros o páginas generadas dinámicamente, con servidores distribuidos de forma que se aprovechen los recursos eficientemente.

La Gestión Basada en *Web* (WBM, *Web Based Management*) también trata de aplicar estas ideas, pero a herramientas de gestión de red. Así, arquitecturas tales como JMX (*Java Management Extensions*, Extensiones de Gestión de Java), antigua JMAPI (*Java Management API*, Interfaz de Programación de Aplicaciones de Gestión de Java), definen los componentes que deben poseer un sistema que pretenda utilizar este nuevo paradigma a la gestión. El DMTF (*Distributed Management Task Force*, Grupo de Trabajo de la Gestión Distribuida) también apuesta por una gestión basada en *Web* usando XML y HTTP, pretendiendo la implantación de CIM (*Common Information Model*¸ Modelo de Información Común) como modelo de información que unifique los estándares tradicionales en la arquitectura llamada WBEM (*Web Based Enterprise Management*, Gestión de Empresas Basada en *Web*). Por otro lado, OMG (*Object Management Group*, Grupo de Gestión de Objetos), a través del grupo de trabajo de JIDM (*Joint Inter-Domain Management*, Gestión Inter-Dominios Unificada) ha tratado de definir cómo se debe traducir especificaciones e interacciones CORBA con dominios de gestión tales como CMIP o SNMP, permitiendo la compatibilidad hacia atrás con sistemas ya existentes.

Por otro lado, también hay que conseguir modularizar las aplicaciones de gestión, aprovechando las posibilidades que dan estas nuevas tecnologías: Es posible el desarrollo de gestores que funcionen sobre DPEs (*Distributed Processing Environments*, Entornos de Procesamiento Distribuido) y a los que se acceda mediante una interfaz basada en *web* usando *applets* encapsulados en páginas HTML, o bien páginas HTML generadas dinámicamente. Los servicios de una plataforma tradicional, tales como el acceso a la pila de protocolos de gestión o un servicio de eventos, podrían ser en este caso servicios estandarizados del DPE, como ocurre con los servicios de CORBA.

1.2 Objetivos y estructura del documento

Las diferencias entre las distintas propuestas para la Gestión Basada en *Web* implican la necesidad de estudiarlas y compararlas, posibilitando la elección de la mejor alternativa para cada caso particular. Para ello, este artículo hace un análisis comparando CORBA/JIDM, CIM/WBEM y JMX, usando un marco arquitectónico basado en las características

generales que un sistema de gestión debe poseer, siguiendo las directrices propuestas por algunas organizaciones involucradas en la gestión de red.

La forma en que se desarrollan los objetivos propuestos es como sigue: A continuación se tratará de caracterizar los sistemas de gestión, en términos de arquitectura, servicios y otras cuestiones adicionales. Tras ello se presentará un marco arquitectónico de los sistemas de gestión basada en *web*, haciendo corresponder los sistemas existentes, CORBA/JIDM, CIM/WBEM y JMX, con dicho marco. Así, se podrá proceder a su comparación, en la que se expondrán los puntos a favor y en contra de cada uno de ellos. El documento finaliza mostrando las conclusiones que se han obtenido de este estudio.

2 Características de sistemas de gestión

En este apartado se pretende dar una visión a las características generales que debe cumplir una arquitectura genérica de gestión. Estas características han sido extraídas al analizar aquellas cuestiones más relevantes de [10], [12] y [13], y se refieren a la arquitectura de un sistema de gestión, los servicios que debe poseer, así como otras características generales. Su utilidad es relevante en dos cuestiones que serán de interés en los siguientes apartados:

1. A la hora de definir una arquitectura de gestión.

2. A la hora de comparar distintas implementaciones que se ajustan a dicha arquitectura.

Este estudio debería completarse con sendos análisis de los modelos de información de gestión y de la seguridad, dada la importancia que tienen en las arquitecturas. Sin embargo no se han incluido debido a que son temas con entidad propia y se salen del ámbito de este documento.

2.1 Arquitecturas

En lo que se refiere a arquitectura, el OpenGroup ha definido el Modelo de Referencia XSM (*X-Open Systems Management*, Gestión de Sistemas X-Open) tal y como la ilustrada en la Figura 1.

Esta arquitectura se sustenta en los servicios que se detallan en el subapartado 2.2, teniendo una connotación especial los servicios de comunicaciones entre el gestor y los objetos gestionados. En esta arquitectura se adopta el uso de tecnología orientada a objetos, aunque se incluye, por cuestiones de compatibilidad, la posibilidad de interfaces no orientadas a objetos entre las entidades implicadas en el sistema de gestión.

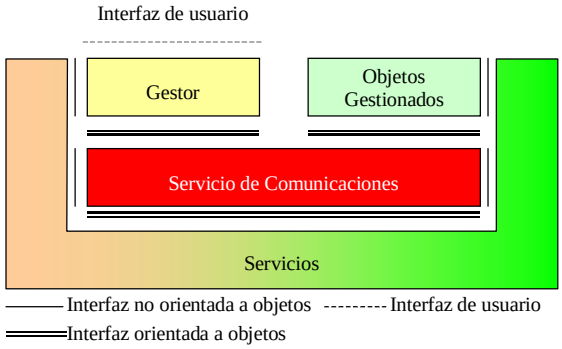


Figura 1. ***Modelo de Referencia XSM [10]***

Esta arquitectura, particularizada a OMA (*Object Management Architecture*, Arquitectura de Gestión de Objetos de CORBA), pasaría a ser:

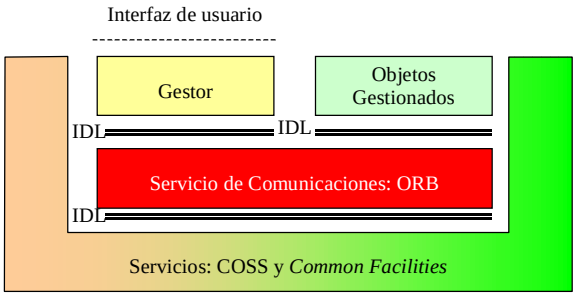


Figura 2. ***Modelo de Referencia XSM particularizado a OMA/CORBA [10]***

Además, ha definido un modelo de interoperabilidad entre XSM y OMA basada en pasarelas, ilustrado a continuación, con lo que se puede tener un punto de referencia de arquitectura de gestión que aprovecha la funcionalidad de plataformas de procesamiento distribuido.

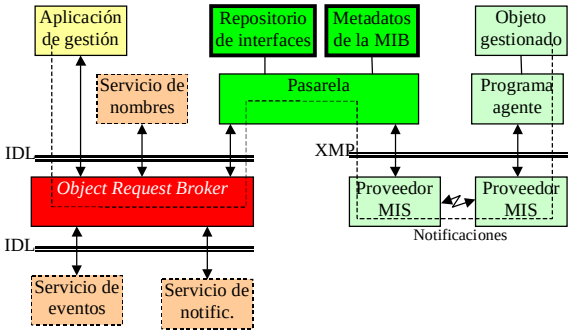


Figura 3. ***Interoperabilidad entre modelos distribuidos según XSM [10]***

En este modelo se ve claramente una división entre lo que serían servicios de gestión, señalados con línea punteada, de lo que son servicios para la pasarela que permite la interoperabilidad de ambos modelos, señalados con línea gruesa. También se ha añadido una línea punteada que indica el camino a seguir entre una aplicación de gestión y un objeto gestionado. Si se pusieran *en serie* estos módulos, tendríamos una aplicación de gestión que funciona en

un entorno distribuido, ayudada de varios servicios; esta aplicación accedería a una pasarela que accedería, a través de los servicios adecuados, a los recursos gestionados.

Por su parte, el TeleManagement Forum, a partir del conjunto de tecnologías de gestión existentes que son útiles para gestión TMN (*Telecommunication Management Network*, Red de Gestión de Telecomunicaciones), ha definido un conjunto de puntos de integración tecnológica para desarrollar sistemas de gestión. A continuación se incluye el diagrama que muestra estos puntos:

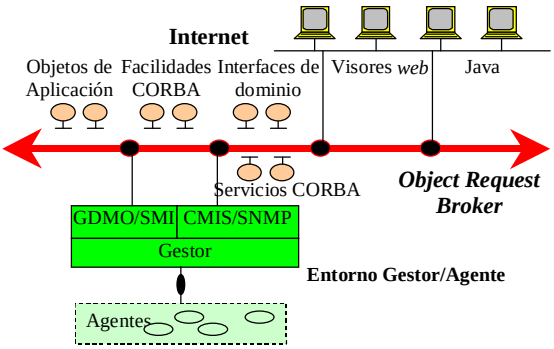


Figura 4. ***Puntos de integración en una arquitectura de gestión híbrida [13]***

Estos puntos son cinco y se refieren a:

1. Traducción entre IDL (*Interface Definition Language*, Lenguaje de Definición de Interfaces de CORBA) y GDMO/SMI, lenguajes de definición de la información en los entornos tradicionales de gestión.

2. Proporcionar servicios CORBA para CMIS/SNMP.

3. Acceso a CORBA desde *web browsers*.

4. Traducción entre Java y objetos CORBA.

5. Proporcionar un entorno de programación para el desarrollo de interacciones gestor/agente basadas en TMN.

Como se aprecia, se pueden encontrar similitudes entre esta arquitectura y la que propone el Open Group para la interoperabilidad con CORBA.

El comité T1 de ANSI también ha definido un marco de gestión, pero su unión con CORBA es aún mayor que la mostrada en el caso del TeleManagement Forum, con lo que no se puede considerar un marco de referencia.

2.2 Servicios

OpenGroup ha definido, desde el punto de vista de XSM, una serie de servicios, los cuales están especificados en [9], que básicamente pueden dividirse en los siguientes: servicios generales, servicios de gestión y servicios de comunicaciones.

En lo que se refiere a servicios para la gestión distribuida, son necesarios los que siguen:

- Servicios de comunicaciones: Con servicios confirmados y no-confirmados, codificación de las peticiones en una sintaxis concreta, seguridad de autenticación entre las partes, descripción de las operaciones y transparencia de localización.

- Servicio de almacenamiento persistente.

- Servicio de seguridad: elementos adicionales a la autenticación antes nombrada.

- Servicio de consistencia: ante el acceso de múltiples gestores a datos compartidos o bien, acceso a múltiples objetos desde un único gestor.

- Servicio de colección.

- Servicio de selección.

- Servicio de eventos.

- Servicio de nombrado.

Por su parte, el comité T1 de ANSI, en un intento por estandarizar interfaces de gestión particularizadas a CORBA, ha definido la necesidad de los siguientes servicios:

- Servicios comunes de CORBA: Nombrado, Notificación, Registro, Mensajería y Seguridad.

- Servicios adicionales: Búsqueda de factorías, Terminación, Operaciones sobre múltiples objetos (para realizar operaciones de ámbito y filtrado).

2.3 Otras características

En los documentos mencionados también se ha incluido un conjunto de características generales que serían deseables en los sistemas de gestión. En el caso del OpenGroup, un sistema de gestión debe tratar de ser: portable, interoperable, transparente, extensible y robusto.

El TeleManagement Forum propone para los sistemas de gestión que se aplique el uso de sistemas distribuidos, enfocándose en datos corporativos (enterprise management), reutilizando componentes, usando diseño orientado a objetos, manteniendo sistemas heredados, y dando acceso al sistema con herramientas de propósito general y bajo coste.

3 Un marco arquitectónico unificado

3.1 Presentación

Tras lo visto en el punto anterior se deduce que, sea cual sea la tecnología a emplear en un sistema de gestión basado en *web*, el marco arquitectónico con el que se corresponda dicho sistema deberá tener los niveles mostrados en la Figura 5. Así, se pueden distinguir cuatro niveles, que se enumeran de arriba a abajo, y dónde se entremezclan los paradigmas cliente-servidor y gestor-agente:

1. Nivel del cliente: Incluye un visor de páginas HTML con capacidad para ejecutar código embebido en ellas.

2. Nivel de servicios de gestión: Se encarga de actuar de intermediario entre el cliente y los recursos subyacentes, dando también soporte a aplicaciones de gestión que existan en el sistema.

3. Nivel de adaptación: Es necesario un nivel que adapte los servicios generales de gestión a los distintos marcos de gestión existentes

4. Nivel de recursos gestionados: Serían aquellos recursos con agentes tradicionales de gestión de red, o bien otras entidades que actúen como tales, facilitando una interfaz de acceso a información de gestión.

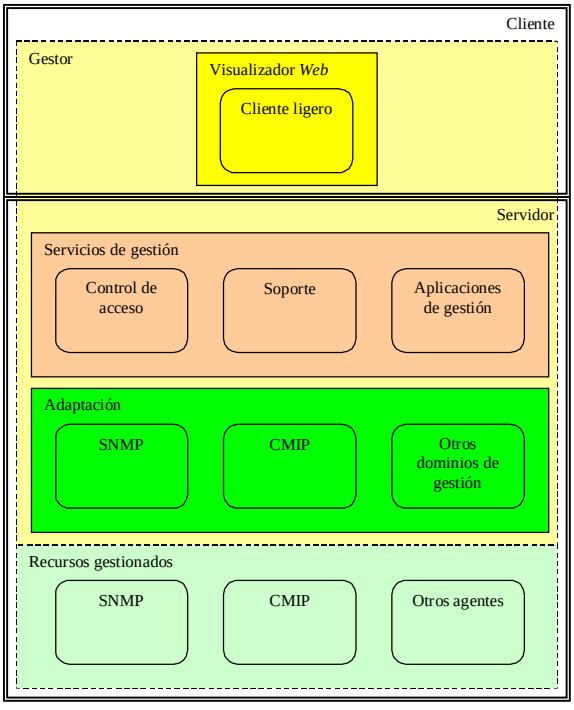


Figura 5. Marco Arquitectónico de la Gestión Basada en *Web*

Aunque este marco arquitectónico parece no tener mucho en común con el modelo propuesto por el OpenGroup e ilustrado en la Figura 1, sí que se pueden ver muchas similitudes, sobre todo, al ver la propuesta de una pasarela de interoperabilidad con CORBA, mostrada en la Figura 3. En la arquitectura de XSM existe, al igual que aquí, una parte dedicada a interfaz de usuario, que en este caso, se encontraría en el visor *web*. Por otro lado, también hay una distinción entre los gestores y los objetos gestionados, con una serie de servicios que median entre ellos. Por tanto, aunque la distribución que se hace es distinta, los conceptos permanecen igual.

Además, esta similitud es más evidente con la arquitectura propuesta por el TeleManagement Forum (ver Figura 4), dónde sí que se pueden distinguir cuatro niveles: Interfaz de usuario, servicios de gestión, usando CORBA en este caso, Adaptación a otros dominios de gestión, en los que existen los recursos a gestionar.

A continuación se muestran las distintas tecnologías existentes, enumeradas en la introducción, y cómo se ejemplarizan según el marco arquitectónico propuesto.

3.2 CORBA/JIDM

Para permitir la interoperabilidad entre los marcos de gestión tradicionales y plataformas de procesamiento distribuido basadas en CORBA, el Open Group creó el grupo de trabajo JIDM (*Joint Inter-Domain Management*, Gestión Inter-Dominios Unificada) [11], que ha sido acogido posteriormente por OMG. Este grupo ha estado estudiando cómo llevar a cabo dicha interoperabilidad, llegando a la conclusión de que ésta se puede posibilitar resolviendo dos cuestiones:

- Normalizar la Traducción de Especificaciones de información de gestión, que detalla la traducción entre los tipos y estructuras de datos utilizados en CMIP y SNMP, protocolos de gestión de red tradicionales, con los usados en CORBA. Es decir: a partir de una MIB, GDMO en el caso de OSI y SMI en el de Internet, es posible generar un módulo IDL que defina qué interfaces CORBA debe implementar un objeto que vaya a ser gestionado mediante esta información de gestión. Así mismo, también es posible hacer una traducción inversa de un módulo IDL a GDMO.

- Normalizar la Traducción de Interacciones entre los distintos dominios, detallada entre CORBA y CMIP, y CORBA y SNMP. Esto significa definir una serie de algoritmos y servicios que permitan traducir y encaminar las peticiones y respuestas generadas en dominios diferentes. Por ejemplo, en el caso de la interacción entre SNMP y CORBA, se detallan servicios con los que se puede traducir un identificador de objeto ASN.1 (OID, *Object Identifier*) a su nombre asociado y, a partir de dicho nombre, obtener la referencia al

objeto CORBA (IOR, *Interoperable Object Reference*) que mantiene la información relacionada con dicho nombre.

La tecnología descrita permitiría particularizar el marco arquitectónico de la Figura 5 en los siguientes términos, ilustrados en la Figura 6.

En este caso, como cliente puede actuar cualquier visor *web*. Las aplicaciones de gestión se apoyarían en los servicios COSS (*CORBA Object Services*, Servicios de Objetos CORBA) definidos por OMG. Todas las cuestiones referentes pasarelas serían servidores CORBA que tuvieran en cuenta las reglas y algoritmos especificados en los documentos antes mencionados. Se unifica el lenguaje de especificación de la información de gestión mediante el uso de IDL. Sin embargo, esta tecnología no define cómo interactuar con otros dominios de gestión, quedando únicamente la puerta abierta a aquellos recursos que posean una interfaz CORBA.

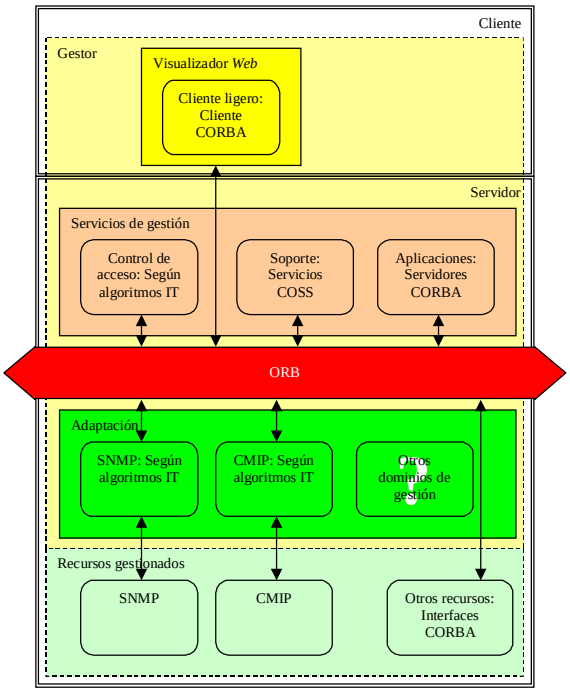


Figura 6. ***Arquitectura usando CORBA/JIDM***

3.3 CIM/WBEM

Para resolver el problema de interoperabilidad entre los múltiples marcos de gestión existentes (SNMP, CMIP/TMN, DMI...) el DMTF ha propuesto lo que se ha dado en llamar CIM [3] y WBEM [4].

- CIM es el *Common Information Model*, o modelo común de información. Aporta un lenguaje de modelado de información, como puedan ser SMI o GDMO, basado en UML (*Unified Modelling Language*, Lenguaje de Modelado Unificado) [8], con el que se trata de modelar toda la información de gestión existente, incluyendo la definida con los lenguajes anteriores.

- Los esquemas CIM son MIBs que tratan de definir varias áreas de la gestión: Sistemas, Dispositivos, Red, Aplicaciones, Inventario..., pero que no tienen una correspondencia exacta con las MIBs de los otros marcos de gestión.

- WBEM, *Web Based Enterprise Management* (Gestión de Empresa Basada en *Web*), es la arquitectura sobre la que se sustenta CIM. Su objetivo es llevar a cabo la gestión integrada de los recursos de una empresa (recursos de red que se gestionan con SNMP, recursos telefónicos que se gestionan con CMIP, recursos de PCs que se gestionan con DMI...), en términos FCAPS (*Fault, Configuration, Accounting, Performance and Security*, Fallos, Configuración, Contabilidad, Rendimiento y Seguridad) empleando las tecnologías que han dado éxito al *web*. Posee una arquitectura en cuatro niveles, similar a la expuesta al principio del documento, que se ilustra y compara en la Figura 7. En principio, el DMTF ha definido el uso de HTTP/XML como mecanismo de comunicaciones entre los distintos módulos, si bien gran parte de las implementaciones existentes hacen uso de otro tipo de tecnologías tales como RMI o DCOM.

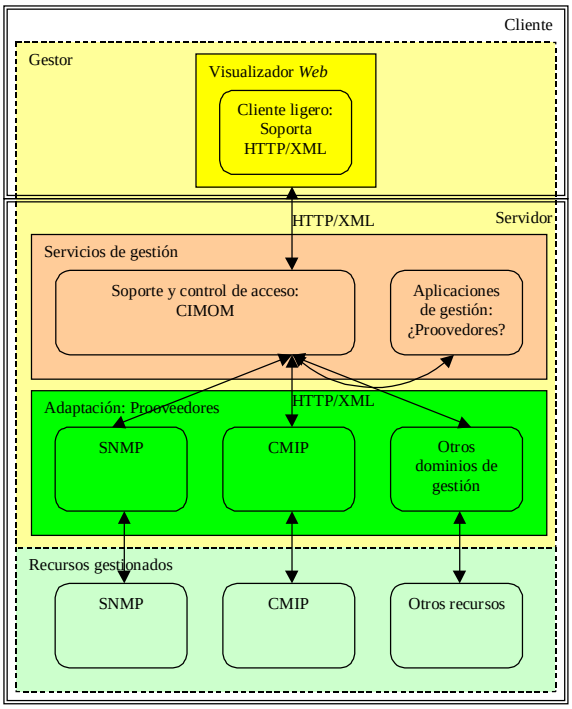


Figura 7. ***Arquitectura utilizando tecnología CIM/WBEM***

Si se usa esta tecnología, toda la funcionalidad se descarga sobre el CIMOM (*CIM Object Manager*, Gestor de Objetos CIM). Las pasarelas están integradas dentro de la arquitectura WBEM como proveedores. Hay poca capacidad de aumentar la

funcionalidad del sistema, dada su carácter monolítico, a no ser que se añada otro sistema que se integre de cierta manera con el CIMOM, como proponen algunos vendedores [1], [2]. Otra solución para integrar las aplicaciones de gestión es considerarlas como proveedores a los que accediera el cliente a través del CIMOM, con su propio modelo de información, como ocurre en la implementación de Microsoft WMI (*Windows Management Instrumentation*, Instrumentación de Gestión de Windows) [6].

En lo que respecta al modelo de información, el uso de calificadores facilita el trabajo al CIMOM a la hora de escoger el proveedor adecuado para la obtención de la información relativa a cierto dominio, y al proveedor a la hora de llevar a cabo la traducción de la información entre los modelos de cada dominio.

Actualmente no existe un documento de estandarización sobre la traducción entre las distintas especificaciones, aunque el objetivo del DMTF es una traducción de todas las especificaciones existentes.

3.4 JMX

A diferencia de JMAPI, la propuesta anterior de gestión con Java en que existía una arquitectura parecida a la propuesta en WBEM, JMX (*Java Management eXtensions*, Extensiones de Gestión Java) [15] no es realmente una arquitectura de gestión, sino de instrumentación de la gestión. De hecho, JMX es únicamente un conjunto de bibliotecas de Java que posibilitan la instrumentación de aplicaciones de una manera más sencilla, sin importar el protocolo de intercambio de información. Sin embargo, a partir de este conjunto de bibliotecas se podría diseñar una arquitectura, no sólo de instrumentación, sino de gestión integrada.

Dicha arquitectura de instrumentación posee los siguientes componentes, separados por niveles:

- Adaptadores de protocolos para la comunicación con la instrumentación, adaptándola a protocolos tales como SNMP, o bien únicamente realizando una comunicación remota Java con RMI o soluciones intermedias que usan HTTP/HTML.

- Marco de instrumentación, que contiene por un lado los adaptadores y por otro, los componentes de instrumentación de gestión. El marco de instrumentación también puede tener una serie de servicios para persistencia, registro, búsqueda,...

- Los componentes de instrumentación de gestión o M-beans (*Management beans*) usan el paradigma de componentes Java o JavaBeans aplicándolo a la instrumentación de la gestión. Estos M-beans pueden ser diseñados directamente en Java, o bien haber sido creados a partir de una MIB.

Actualmente las bibliotecas de JMX dan soporte a algunos protocolos de gestión existentes: SNMP y WBEM/CIM. Otros, como CMIP, están en proceso de desarrollo.

A continuación se propone e ilustra en la Figura 8 cómo se podrían utilizar las bibliotecas JMX para proyectar la arquitectura propuesta en el marco de Java.

El cliente puede ser un *applet* Java. No tiene por qué ser necesario que utilice las bibliotecas de gestión (JMX) sino que utilice únicamente las estándares de Java, que incluyen RMI o CORBA. También existe la posibilidad de que el cliente simplemente interprete las páginas HTML que recibe, y pasar la complejidad de su generación al servidor. Los servicios de gestión se implementarían a partir de las bibliotecas JMX, que facilitan las tareas de gestión. También es posible utilizar algunas de las bibliotecas que han sido definidas en el marco de la J2EE (*Java2 Enterprise Edition*, Java2, Edición Empresarial) [14], para aquellas funciones que no posea JMX, pero sí estén desarrolladas en Java.

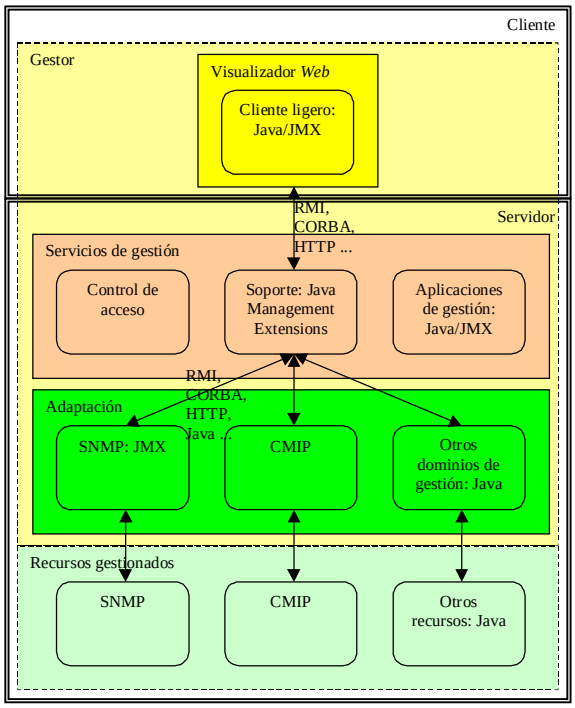


Figura 8. ***Arquitectura usando JMX***

En lo que se refiere a la interoperabilidad con otros dominios de gestión, como se ha dicho anteriormente, existe únicamente interoperabilidad con SNMP y WBEM/CIM. La interoperabilidad con CMIP está en desarrollo, aunque existen bibliotecas Java ya desarrolladas por terceros para realizar operaciones CMIP [5]. El acceso a otros dominios de gestión pasa por hacer uso otra vez de Java, en este caso, en

conjunto con su biblioteca de acceso al sistema o con JNI (*Java Native Interface*, Interfaz Nativa de Java).

4 Comparativa

A continuación se realiza un análisis en el que se señalan las fortalezas y debilidades de cada tecnología con respecto al resto a la hora de implementar la arquitectura propuesta. Para ello se tiene en cuenta aquellos puntos que se han presentado en el apartado 2, relativos tanto los servicios que puedan dar estas implementaciones como las cuestiones más generales que también se han descrito.

4.1 JIDM/CORBA

Puntos a favor

El uso de CORBA permite reutilizar servicios ya existentes e integrar los servicios nuevos en un entorno de un ámbito más general. Además, se puede extender su funcionalidad fácilmente gracias a la modularidad inherente de CORBA. Se podrían aplicar los conceptos de la Facilidad de Meta Objetos (MOF, *Meta Object Facility*) de OMG [7] para mantener y manejar la información de gestión, sin restringirse al uso de IDL. En definitiva, con CORBA son posibles todas las características deseables para un sistema de este tipo: la portabilidad, interoperabilidad, transparencia, extensibilidad y robustez, según el Open Group, y su enfoque en datos corporativos (enterprise management), reutilización de componentes, diseño orientado a objetos, mantenimiento de sistemas heredados y acceso al sistema de propósito general y bajo coste según el Tele Management Forum.

Puntos en contra

CORBA usa IDL, que es menos potente que CIM para el diseño específico de información de gestión, aunque el hecho de que exista un perfil UML sea un punto a su favor, pues es posible describir la información con herramientas CASE estándar; además, se podría emplear la Meta Object Facility de OMG, como se ha comentado anteriormente, si se pretende utilizar un modelo de información más potente. Tampoco existe una definición de objetos gestionados, aunque el comité T1 de ANSI está trabajando en este punto; además, los algoritmos de JIDM permiten redefinir en IDL todas las MIBs ya existentes en GDMO y SMI. No tiene definidas interacciones con otros dominios que no sean SNMP y CMIP, aunque tampoco tiene excesivo sentido: CIM/WBEM está orientado a entidades gestoras y el resto de los dominios son prácticamente propietarios. Otra cuestión negativa es la necesidad de servicios específicos a cada dominio para llevar a cabo las traducciones de interacciones, si bien esto también ocurre en CIM/WBEM con el uso de proveedores específicos de cada dominio.

4.2 CIM/WBEM

Puntos a favor

Con esta iniciativa existe la intención de unificar todos los posibles modelos de información existentes. Para ello, se hace uso de CIM, un modelo bastante potente y orientado a objetos y basado en UML, aunque posea un metamodelo algo diferente. Además, existe una integración total de las tecnologías *web* en esta arquitectura, cumpliendo las exigencias de reusabilidad y bajo coste. Con respecto a JIDM añade un modelado de información estandarizada, que se suma a los ya existentes. Aporta el uso de calificadores para añadir metadatos que completen el modelado de los objetos.

Puntos en contra

El mayor problema de esta arquitectura es su falta de modularidad. No es posible desplegar aplicaciones de forma que un cliente tenga una interfaz de acceso única, a no ser que estas aplicaciones se modelen como proveedores, como ocurre en el caso de WMI ya referenciado anteriormente (existen proveedores, como el monitor de rendimiento, que en una plataforma de gestión serían aplicaciones). También, varios servicios deseables para un sistema de este tipo se deben modelar como proveedores (notificaciones, por ejemplo). Además, existe una falta de consenso en los fabricantes a la hora de utilizar HTTP/XML, ya que, por ejemplo, Microsoft está utilizando DCOM y Sun, RMI, como sistemas de acceso al CIMOM. En lo que se refiere a CIM, se le puede achacar el que su metamodelo no se corresponda con un perfil particularizado del metamodelo de UML, lo que supone tener que trabajar en la adaptación entre ambos modelos.

4.3 JMX

Puntos a favor

Las múltiples bibliotecas definidas en JMX dan la posibilidad de usar cualquier protocolo, desde cualquier punto (gestor, agente, cliente o servidor), y no únicamente Java RMI. Se está trabajando en su adaptación con los estándares de gestión: SNMP, CIM/WBEM y CMIP. Además, el uso de Java permite su despliegue en cualquier sistema operativo, lo que ocurre en el caso del *web*, donde máquinas de distintas arquitecturas intercambian datos libremente. La información se puede definir en un lenguaje orientado a objetos, utilizando la estructura de M-beans, pero no existe, al igual que ocurre con CORBA ninguna información definida a priori, a no ser la ya existente de modelos tradicionales de gestión.

Puntos en contra

JMX está centrado en Java, lo que limita su aplicabilidad con otros lenguajes de programación, si bien, el uso de IIOP solventa la interoperabilidad entre códigos escritos con distintos lenguajes. A diferencia de JMAPI, no define una arquitectura de gestión, sino únicamente una arquitectura de instrumentación de la gestión. Esto supone que tenga grandes limitaciones a la hora de proporcionar una infraestructura de servicios, aunque el resto de las especificaciones que se están desarrollando para Java e incluidos en J2EE (JDBC, JNDI, ...) puede suplir esta carencia.

5 Conclusiones

A pesar de que parece existir una tendencia generalizada hacia el *web*, las tecnologías existentes que pretenden utilizarla para la gestión difieren en varias cuestiones, que posiblemente sean debidas a política de mercado. Cada una de las tecnologías presentadas es fácilmente proyectable sobre la arquitectura propuesta basándose en los conceptos generales descritos por el Open Group y el TeleManagement Forum, lo que demostraría la posibilidad de llevar a cabo una gestión basada en *web* con cualquiera de ellas. La cuestión importante desde un punto de vista técnico es conocer las fortalezas y debilidades de cada una para utilizar en cada caso la tecnología más adecuada.

Otra cuestión que también merece la pena estudiar es la heterogeneidad de modelos de información que se crea al usar estas tecnologías. Por un lado, es necesario evaluar su capacidad expresiva, comparándolos desde su meta-modelo. También se plantea la falta de interoperabilidad de la información definida a un nivel semántico. Por ejemplo, CIM ha definido un conjunto de esquemas que no se corresponden directamente con las MIBs de GDMO o SMI, con lo que los proveedores de estos protocolos no pueden realizar una traducción directa de los mismos. Para conseguir un modelo realmente común debiera ser posible hacer una proyección de este modelo en los de GDMO y SMI haciendo uso del significado de los datos especificados, lo que supone tener que utilizar técnicas ontológicas que modelen el comportamiento de los recursos gestionados, independientemente del modelo de información que se utilice.

Referencias

[1] BMC, *Making WBEM Work for You*, http://www.dmtf.org/download/presentations/conf1999/v101.ppt, DMTF Annual Conference, 1999.

[2] Bull, *Integration of WBEM into a standard management platform, Bull OpenMaster*, http://www.dmtf.org/download/presentations/conf1999/v102.ppt, DMTF Annual Conference, 1999.

[3] Distributed Management Task Force, Inc. *Common Information Model (CIM) SpecificationVersion 2.2*. DMTF Standard, junio de 1999.

[4] Distributed Management Task Force, Inc. *WBEM initiative*, http://www.dmtf.org/wbem/index.html, 1999.

[5] O. Festor, *The RESEDAS Free Java Management Software Homepage*. INRIA, http://www.loria.fr/~festor/JAM/JAM.html, 1997.

[6] Microsoft Corporation, *Windows Management Instrumentation*, http://msdn.microsoft.com/downloads/sdks/wmi/default.asp, 2000

[7] The Object Management Group, *Meta Object Facility (MOF) Specification*. OMG Document ad/99-09-05, septiembre de 1999.

[8] The Object Management Group, *Unified Modeling Language (UML) 1.3 specification*. OMG Document formal/00-03-01, marzo de 2000

[9] The Open Group, *System Management: Identification of Management Services*. Open Group Snapshot S190, mayo de 1992.

[10] The Open Group, *Systems Management: Reference Model*. Open Group

[11] The Open Group, *Inter-Domain Management: Specification & Interaction Translation*. Open Group Specification C802, enero de 2000.

[12] T1 Committee, *Working Document for Draft Standard ANSI T1.2xx-2000, CORBA Generic Network and NE Level Information Model*. T1 Document 0m150300, enero de 2000

[13] Tele Management Forum, *Smart TMN Technology Integration Map*. Tele Management Forum GB909, octubre de 1998.

[14] Sun Microsystems, Inc. *Java™2 Enterprise Edition (J2EE)*. http://java.sun.com/j2ee, 1999

[15] Sun Microsystems, Inc. *Java™ Management Extensions (JMX)*. http://java.sun.com/products/JavaManagement/, 1999

Integración de tecnologías de acceso de banda ancha: Proyecto HIAD

C. López Bravo, J. García Reinoso[1], F.J. González Castaño[1],
P.S. Rodríguez Hernández[1], J.M. Pousada Carballo[1]
Área de Ingeniería Telemática
Universidad Politécnica de Cartagena
Cristina.Lopez@upct.es

***Abstract**: The HIAD (High Integration Access Device) is a project for the development of a central device, with the main object to integrate broad band access tecnologies. The main device has a board, integrating two different interfaces: ISDN or xDSL, to connect the users and the OC-3 OF, connecting the HIAD to the ATM backbone. Using xDSL in the user loop, you can utilize the existing telephone net to provide broad band services. Another ventage is to unify the protocol from the user to the server, since we use the ATM protocol all the way. In the following sections, we'll describe the architecture of HIAD in the very first phase, where we just implement ADSL in the user part.*

1. Motivación

Este trabajo está financiado por la Comisión Europea y la Comisión Interministerial de Ciencia y Tecnología, a través del proyecto TIC 1FD97-2248-C02-01. En el proyecto participa la empresa Versaware S.L. (Vigo), y se cuenta con una expresión de interés de Telenor (Noruega), del año 1999.

Además del desarrollo del equipo HIAD, el proyecto persigue la obtención de un prototipo de funcionalidad reducida, para demostraciones de tecnologías de sistemas telemáticos de banda ancha: Motorola MPC860 [1] y EVB92501 [2], sistemas operativos pSOS+ [3] y VxWorks [4]. En lo sucesivo, nos referiremos a este prototipo como prototipo HIAD. En la figura 1 podemos ver el dispositivo HIAD dentro de una red de comunicaciones.

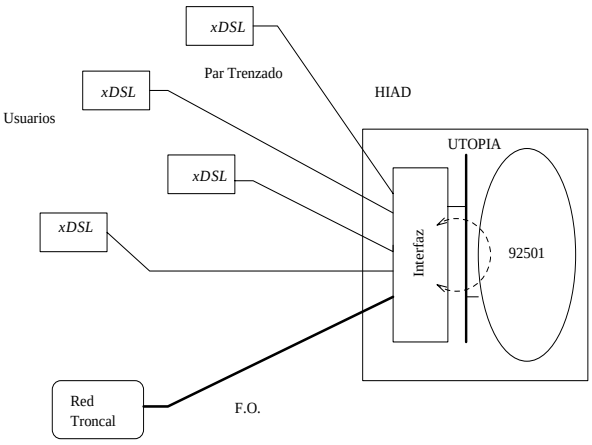


Fig. 1: HIAD

2. Arquitectura del prototipo HIAD

2.1 Arquitectura *hardware*

Para la construcción de un prototipo del equipo de central HIAD se ha decidido utilizar la placa de desarrollo EVB92501 de Motorola (ATMC-EVB). Esta placa:

- Incorpora numerosos puertos de entrada y salida: Ethernet, RS-232, ADI, etc., que facilitan la comunicación con otros sistemas. Desde un PC se puede descargar los módulos de inicio, preparar la memoria FLASH, etc.
- Incluye en una sola placa los dos tipos de interfaces que necesitamos (ADSL y FO OC-3).
- Y, razón fundamental de nuestra elección, cuenta con un circuito -MC92501- capaz de procesar celdas ATM, y concentrar el tráfico generado por distintos usuarios en una sola línea de salida.

Los componentes principales de la placa son:

- Un microprocesador MPC860, de la familia PowerPC. Es el encargado del

1 Departamento de Tecnologías de las Comunicaciones, Universidad de Vigo, javier/pedro/chema/reinoso@ait.uvigo.es.

control de los demás subsistemas de la placa.

- Un controlador de nivel ATM MC92501 [5], que realiza funciones de gestión de tráfico (celdas OAM), control de flujo de datos (UPC), conversión de direcciones y generación de estadísticas. Está formado por dos procesadores de celdas, *ingress* y *egress*, y sus interfaces UTOPIA (niveles 1 y 2). El MC92501 gestiona su propia memoria, en la que almacena la información relativa a las conexiones que se abran.

- Un transceptor AX ATM-SONET/SDH, CY7C955 [6]. Es la interfaz entre el equipo HIAD y la red ATM propiamente dicha. Este circuito se encarga de convertir las celdas ATM (recibidas desde la interfaz UTOPIA) en tramas serie SONET para transmisión por FO, y viceversa.

- Un bus UTOPIA (niveles 1 y 2), que permite la conexión de la placa con otros dispositivos (por ejemplo los modems ADSL con interfaz utopia). Es el nexo de unión entre los usuarios y el MC92501, y entre éste y la FO.

- Un generador de tráfico que, combinado con los dispositivos *ingress* y *egress*, permite comprobar el funcionamiento del equipo HIAD sin necesidad de trabajar con tráfico real. Una vez que se trabaje con tráfico real, el conjunto permite encaminar el tráfico procedente de los usuarios hacia la FO, y viceversa (las celdas almacenadas en el grabador *ingress* se envían al generador de tráfico *egress*).

Y, finalmente, un conjunto de componentes auxiliares: memorias y puertos de comunicación (RS-232, Ethernet, ADI).

Para realizar desarrollos sobre la placa EVB92501, se puede conectar un PC a través de un puerto ADI (modo depuración), del puerto RS-232 o del puerto Ethernet (modo de funcionamiento normal). En la figura 2 se muestra la disposición de todos los componentes (incluido el PC) y su interconexión.

2.2 Arquitectura *software*

Tras una evaluación de pSOS+ y VxWorks, se eligió éste último para el desarrollo del equipo HIAD. VxWorks dispone de un entorno de

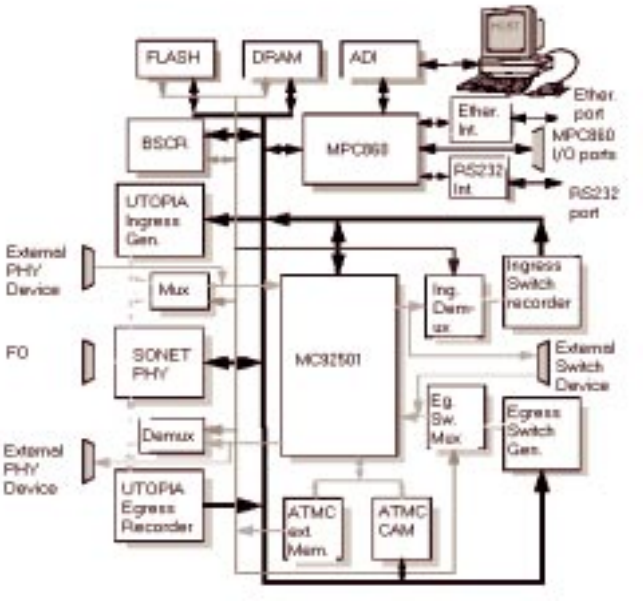


Fig. 2: ATMC-EVB

desarrollo, llamado Tornado, que se instala en el PC. Tornado dispone de editores, compiladores y diversas aplicaciones que permiten descargar imágenes (núcleo y aplicaciones) en la placa ATMC-EVB.

Incluye, además, una herramienta para desarrollo de BSPs (*Board Support Package*), que permite añadir periféricos específicos, o portar el sistema operativo a placas no soportadas. Aunque la distribución de VxWorks incluye un BSP genérico para MPC860, ha sido necesario realizar adaptaciones para la placa ATMC-EVB. En especial, se ha reflejado la existencia del MC92501, su memoria, los generadores y grabadores de tráfico y la interfaz de FO.

2.2.1 Funciones del prototipo HIAD

2.2.1.1 Configuración inicial

Los procedimientos de inicio siguen las órdenes de un NMS externo (ver apartado 4). Para ello, se utiliza un agente SNMP que se ejecuta en el núcleo de VxWorks. La configuración incluye el número de interfaces ADSL activas (o, en otras palabras, número de modems ADSL conectados a HIAD) y el número de conexiones que se va a poder abrir, tanto VPC como VCC. La configuración se puede cambiar, pero es preciso reiniciar todas las tablas de conexiones y los contadores de celdas.

2.2.1.2 Apertura de conexiones y enlaces

Los circuitos virtuales que se crean son permanentes. Es decir, se asume que el control de admisión es ajeno al equipo de central. Sin embargo, sí puede ser necesario mantener un control de flujo que garantice que los usuarios no sobrepasen las tasas de transferencia contratadas. Los parámetros relacionados con la calidad de

servicio (PCR, MSB, etc.) y el tipo de tráfico (UBR, ABR, CBR, etc.), se especifican cuando se activa una conexión para el transporte de datos.

El número de conexiones, el sentido de la comunicación para cada una de ellas y la categoría de servicio ofrecido son especificados por el NMS (según sus tablas, previamente generadas a mano).

2.2.1.3 Gestión del tráfico del sistema

Una vez creadas y activadas las conexiones, los usuarios pueden utilizarlas. A partir de este momento, las funciones del HIAD son:

- Encaminamiento de las celdas de la red a los usuarios, y viceversa.
- Control del flujo.
- Monitorización del tráfico que circula en ambos sentidos.

Cuando se recibe una celda, en sentido *ingress*, el MC92501 lee su cabecera y comprueba si la conexión correspondiente está activada en su tabla de configuración. La tabla de memoria está dividida por puertos, por lo que, conocido el puerto por el que ha entrado la celda, se reduce el espacio de búsqueda. Cuando se recibe una celda, en sentido *egress*, ésta viene acompañada de un identificador de conexión (ECI), que es un puntero directo a las tablas de configuración, donde se comprueba si la conexión correspondiente está activada. A partir de aquí hay dos posibilidades:

- La conexión está activada. En este caso, el MC92501 cambia la cabecera de la celda y la envía por el puerto correspondiente.
- La conexión no está activada o forma parte de un *multicast*. El MC92501 no puede seguir procesando esa celda. No obstante, se le puede programar para que realice las acciones oportunas, como veremos más adelante.

2.2.1.3.1 Encaminamiento de celdas

Conexiones punto a punto: El tráfico generado por los usuarios se encamina hacia la interfaz FO a través del HIAD. La salida FO se dirige al conmutador de acceso a la red. En sentido inverso, el tráfico de red de la FO llega al HIAD, desde donde es encaminado hacia el usuario correspondiente.

Este tipo de configuración supone que:

- Primero, hay que enlazar los dos tramos de la comunicación, es decir, la conexión entre los usuarios y el HIAD, y la conexión entre el HIAD y la red. Por cada conexión que se abre entre un usuario y el HIAD debe abrirse otra, entre el HIAD y la interfaz de FO (salvo en el caso de *multicast*, que veremos más adelante). A continuación, hay que unir ambas conexiones, actualizando las tablas de contexto que el MC92501 mantiene por cada conexión abierta. El HIAD permite, si así se desea, que se active solo uno de los dos sentidos de la comunicación.
- Luego, se debe realimentar el tráfico. Todas las celdas procesadas por el ATMC-EVB en sentido *ingress* (ya sean generadas por los usuarios, o procedentes de la red) vuelven a ser procesadas en sentido *egress*. La realimentación se puede hacer de dos formas: por *hardware*, utilizando el dispositivo correspondiente [2], o por *software*, sirviéndose de los generadores y grabadores. Es decir: todo lo que se almacena en el grabador de *ingress* es enviado por el microprocesador al generador de *egress*, de donde regresa al MC92501. Para la elaboración del prototipo esto es suficiente. La realimentación consigue dos objetivos:

  Por un lado, como se explicará en el siguiente apartado, posibilita la realización de control de flujo en el tráfico de los usuarios y en el tráfico que procede de la red. Por otro, la modificación de las cabeceras de las celdas, cuando proceda (por ejemplo, en el tratamiento de celdas *multicast*).

Conexiones punto a multipunto: En el caso de conexiones punto a multipunto, se abren varias conexiones entre los usuarios y el HIAD, y una sola conexión entre éste y la red. Consecuentemente, todas las conexiones de usuario se enlazan con la conexión de red, y solo se activa uno de los sentidos de comunicación, red→usuarios. En nuestra implementación, esta configuración es programada por el NMS.

La gestión del tráfico *multicast* se basa en la capacidad del MC92501 para extraer celdas del flujo de datos y entregarlas al microprocesador, así como para recibir celdas de éste e insertarlas de nuevo en el flujo de datos. Las conexiones *multicast* pueden distinguirse por sus identificadores. Si el MC92501 detecta uno de esos identificadores en la cabecera de una celda, la

envía a una cola de extracción. Una vez en la cola, el generador de interrupciones de la placa ATMC-EVB avisa al microprocesador. Entonces, el MPC860 extrae la celda de la cola, busca en sus tablas las conexiones de usuario por donde debe enviarla, y hace tantas copias como sea necesario en la cola de inserción. Una vez allí, las celdas replicadas son progresivamente introducidas en el flujo de datos, aprovechando ranuras libres.

2.2.1.3.2 Control de flujo

El MC92501 permite la realización de control de flujo. En principio, esto es posible para los dos sentidos del MC92501, *ingress* y *egress*. El problema es que el control es sobre un sentido o sobre el otro, pero no sobre los dos a la vez. Se podría pensar entonces que, por ejemplo, se controla el tráfico generado por los usuarios, pero no el tráfico que solicitan a la red. Sin embargo la arquitectura que hemos seleccionado evita este tipo de situaciones, como se explica a continuación.

Se ha dicho que, cuando se abre una conexión, ésta tiene asociados unos parámetros relativos a la categoría del servicio. Cada sentido de la comunicación tiene su propio conjunto de parámetros. A partir de estos parámetros se construyen las estructuras que definen el control de flujo UPC. Con los parámetros de sentido usuario→red se construye el control UPC en sentido *ingress* para los usuarios, controlando la salida hacia la red. Con los parámetros de sentido red→usuarios se construye el control UPC para el *ingress* de la FO, controlando así la demanda de los usuarios. De esta forma, el control se hace en ambos casos en sentido *ingress*, pero en realidad se controlan los dos sentidos de la comunicación, usuario↔red.

2.2.1.3.3 Cómputo del tráfico

El MC92501 mantiene tablas de contadores, con entradas para cada una de las conexiones que estén abiertas. La estructura de estas entradas se puede configurar, de forma que las celdas se cuenten conjuntamente o separándolas en función de su bit de prioridad CLP.

Además, el MC92501 también mantiene una tabla de contadores para cada línea activada, es decir, para cada modem ADSL conectado, y para la interfaz de FO. Esto permite tarificación por tráfico global, independientemente de las conexiones que tenga abiertas un usuario.

3. El agente SNMP en el proyecto HIAD

En este apartado se describe la implementación del agente SNMP del prototipo HIAD. En los RFCs de SNMP se detallan las estructuras de datos que se deben crear para el control de un dispositivo mediante este protocolo de gestión de redes, pero las descripciones son un tanto oscuras. Por lo tanto, se han seguido las recomendaciones en los casos que se explican con rigor, pero se han hecho suposiciones en los casos en los que las especificaciones no eran del todo claras o no existían.

3.1 El protocolo SNMP

El SNMP (*Simple Network Management Protocol*) es un protocolo sencillo de administración de redes. Tanto es así, que en la primera versión sólo se incluían cuatro posibles llamadas: *set*, *get*, *get-next* y *trap*. La complejidad reside casi exclusivamente en el NMS (*Network Management System*), que es el encargado de administrar y monitorizar la red.

Así, la red puede verse como un conjunto de nodos administrados (que deben poseer al menos un agente SNMP), y al menos un NMS.

Existe un NMS que controla los dispositivos y algunos nodos poseen un agente SNMP. Estos nodos son potenciales puntos administrables. Aun así, un NMS no administra necesariamente cualquier nodo con agente SNMP, puesto que existe un sistema de claves.

Cada nodo con agente SNMP posee una estructura de datos llamada MIB (*Management Information Base*), jerárquicamente distribuida. Así, dependiendo de qué se desee administrar, se puede incluir una parte del árbol MIB o no. Por ejemplo, *iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1).system(1)* tiene objetos que muestran información del nodo administrado: nombre, ubicación, tiempo desde el último *reset*, etc.

3.2 Misión del agente SNMP en el prototipo HIAD.

El prototipo HIAD es un dispositivo concentrador de líneas ADSL hacia ATM. Es preciso iniciar los circuitos virtuales, antes de que los datos fluyan. Para ello, se ha optado por PVC (*Permanent Virtual Circuits*) -aunque la implementación de SVC (*Switched Virtual Circuits*) no se excluye en una fase posterior-. Los PVC son fijos y no tienen que generarse cada vez que se desee comunicar dos entidades. Por tanto, cada vez que se inicie el agente, se deben crear estos circuitos, y esta misión se encomienda al NMS.

Además, se puede tener control SNMP sobre todas las interfaces del dispositivo, como Ethernet, puerto serie, etc, por lo que se puede llevar una estadística de los paquetes que las atraviesan.

3.3 Modificación de las conexiones.

Para que el tráfico pueda fluir entre dos puntos de una red orientada a la conexión -como es el caso de ATM- es necesario establecer circuitos. El NMS

solicita las conexiones a realizar, y el agente SNMP del HIAD las establece.

Una conexión se define mediante una serie de parámetros básicos. En la terminología de los RFCs SNMP se distinguen varias definiciones que son importantes para comprender el mecanismo de activación de conexiones.

Una *interfaz* es un punto por el que pueden entrar datos. Con esta definición, podemos llamar interfaz a una línea ADSL, a una línea ethernet o incluso a un generador de tráfico. En nuestro caso, tendremos dos tipos de interfaces: las líneas ADSL y las líneas FO. El equivalente en terminología de circuitos es el puerto.

Una *conexión* es el segmento de línea entre dos dispositivos por el que fluyen los datos. Es decir, entre los modems ADSL y el HIAD se define una conexión.

Una *conexión cruzada* es la intersección lógica de dos conexiones. Es decir, cuando el flujo proveniente de una conexión se encamina hacia otra.

La representación del *tipo de tráfico* definida en los RFCs difiere de las del ATM Forum, pero se puede establecer una relación sin mayor dificultad.

La secuencia que sigue el NMS para comunicar un usuario con la red es la siguiente:

- Se define un grupo de tipos de tráfico que podrá usarse en cualquier conexión.
- Se abren las conexiones en cada interfaz.
- Se cruzan las conexiones necesarias.

Se debe tener en cuenta que las conexiones pueden ser tanto bidireccionales como unidireccionales. Estas últimas son útiles para implementar *multicast*.

3.4 Ejemplo de configuración

En la figura 3 se muestra una posible configuración del HIAD.

El NMS envía las instrucciones necesarias para que el agente SNMP del HIAD cree 6 conexiones. Dos conexiones para la interfaz 1, a la que está conectado un modem ADSL, otras dos para la interfaz 2, a la que está conectado otro modem ADSL, y dos más en la interfaz FO. Luego, se cruzan todas ellas. El resultado es una conexión *multicast*, entre una FO y dos lineas de modems.

3.5. Monitorización de información en redes ATM.

En toda red de datos es necesario realizar estadísticas de comportamiento a lo largo del tiempo. Esto es interesante para futuras ampliaciones, ya que las estadísticas reflejan cuáles son los recursos más utilizados; es útil para poder facturar el gasto de cada usuario, e incluso para el propio NMS, para poder tomar decisiones sobre el comportamiento de un determinado dispositivo.

Los RFCs de monitorización ATM proporcionan mecanismos para la definición de los tipos de datos y el momento y manera de recolectarlos.

El método descrito en el RFC2513 [7] es genérico y permite recolectar más tipos de datos que los necesarios en nuestra implementación, por lo que se han eliminado algunos aspectos.

Existen dos formas básicas de recolección. Se puede seguir un temporizador o reaccionar ante determinados eventos. Si se escoge la primera opción, el agente SNMP crea un temporizador que se conecta a una función encargada de la recolección.

En el RFC2512 [8] se enumeran los datos que se pueden almacenar sobre una determinada conexión. En el RFC2513 se describen los pasos a seguir para la obtención de dichos datos.

El funcionamiento con temporizador sería el siguiente:

1. El NMS le pide al agente que cada *n* segundos se recolecte la información solicitada. Esta información puede escogerse entre todo el espectro de

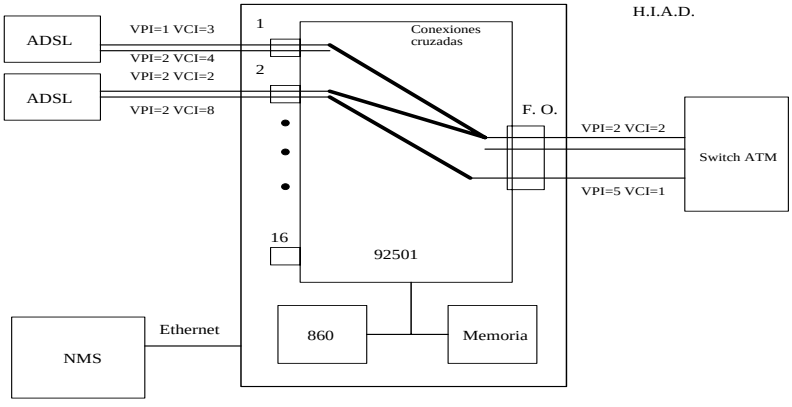


Fig. 3: Ejemplo de configuración HIAD

posibilidades del RFC2512.

2. El agente recolecta esta información y la guarda en un archivo, cuyo nombre ha sido asignado por el NMS.

3. El NMS, mediante FTP, adquiere el fichero creado por el agente SNMP.

4. Análisis cuantitativo *multicast*

Recordemos que en ATM se transmiten celdas de un tamaño fijo de 53 B. Los calculos que siguen se refieren a una única conexión multicast.

El proceso para producir las replicas *multicast* es el siguiente:

- Llega una celda al MC92501.
- El MC92501 detecta que la celda es *multicast*, y por lo tanto la deja en la *cola de extracción* (previamente, se ha programado al MC92501 para que dispare una interrupción cuando hay un determinado número de celdas almacenadas).
- Ante dicha interrupción, el MPC860 ejecuta la rutina de servicio correspondiente. Se cambia la cabecera de las celdas de la cola y se copian en la *cola de inserción* del MC92501.
- El MC92501 envía las celdas replicadas.

Tanto la cola de extracción como la de inserción tienen una capacidad de 16 celdas. Esto limita la cantidad de conexiones que pueden formar parte de un *multicast*: el producto entre el número de usuarios del *multicast* U_m y el de celdas necesarias para generar una interrupción C_i debe ser menor o igual que 16. Es decir, en la rutina de servicio de réplica se desea aprovechar al máximo la cola de inserción, pero sin saturarla.

La fórmula que rige las interrupciones es:

$$C_i = \lfloor 16/U_m \rfloor \text{ celdas} \qquad 2 \le U_m \le 16 \qquad (1)$$

Cuando la cola de extracción almacena un número de celdas igual a C_i, se comienza a ejecutar la rutina de servicio, donde se leen las celdas. El número de *bytes* B_{ce} a transferir desde la cola de extracción es igual a $53 \times C_i$.

Según el manual del MC92501 [4], se necesitan 3 ciclos de reloj para acceder a cada *byte* de las colas. Con un reloj de 33 MHz, se necesitan 90 ns por *byte*. De (1), el tiempo necesario para leer B_{ce} *bytes*, T_{ce}, será entonces:

$$T_{ce} = 90 \text{ ns/}byte \times B_{ce} = 4.77 \lfloor 16/U_m \rfloor \ \mu s \quad (2)$$

Para estimar el tiempo necesario para realizar la réplica de celdas, realizamos una aproximación basada en el código:

```
for (celda = 0; celda <Ci; celda++) {
    buf_aux = buf + celda*53;
    for (usuario = 0; usuario < Um; usuario ++) {
      nueva_ECI = buscar_ECI(usuario);
      buf_aux[POS_ECI] = nueva_ECI;
      write_extraction_queu(buf_aux);
    }
}
```

Las funciones más críticas en tiempo de procesado son:

- nueva_ECI = buscar_ECI(usuario): Hay que efectuar una lectura en la posición RAM ECI + BASE_tabla + usuario.
- `write_extraction_queu(buf_aux)`: Una vez preparada la celda para transmisión, es preciso reincorporarla al flujo, insertándola en la cola de inserción.

El tiempo necesario está dominado por la escritura de celdas en la cola de inserción, de modo que despreciamos el resto.

En total, tenemos que escribir $C_{ci} = U_m \times \lfloor 16/U_m \rfloor$ celdas en la cola de inserción, en un tiempo T_{ci}:

$$T_{ci} = 90 \text{ ns} \times 53 \times U_m \times \lfloor 16/U_m \rfloor =$$

$$= 4.77\ U_m \times \lfloor 16/U_m \rfloor \ \mu s \qquad (3)$$

De (2) y (3) resulta un tiempo total de procesado de celdas T_t:

$$T_t = 4.77 \lfloor 16/U_m \rfloor (1 + U_m) \ \mu s$$

En la figura 4 se representa esta igualdad. Se puede observar que el tiempo mínimo de procesado se produce para una conexión *multicast* de 9 usuarios. Como curiosidad, se ha repetido el análisis para el caso en que tuviésemos 32 celdas en vez de 16. Resultaría la segunda curva, con un tiempo mínimo para 11 usuarios.

A continuación, analizaremos si el retardo introducido es aceptable o no, para tráfico en

tiempo real. Para ello, calcularemos la tasa máxima de entrada λ tolerable por el sistema.

Si llamamos C_r al número de celdas que llegan a la cola de extracción mientras se ejecuta la rutina de servicio, para que no se produzcan interrupciones anidadas se debe cumplir $C_r < C_i$. En caso contrario, se saturaría la cola de extracción.

- $C_r \cong \lambda \times T_t / (53 \times 8) < C_i$
- $\lambda \times 4.77 \times \lfloor 16/U_m \rfloor \times (1+U_m)/(53 \times 8) <$

  $< \lfloor 16/U_m \rfloor \times U_m$
- $\lambda < 88.89\ U_m / (1+U_m)$ Mbps

De la cota anterior, podría entenderse que el número de registros disponibles en las cola de inserción es irrelevante. Esto no es cierto, ya que U_m está acotado por ese número.

En la figura 5 se muestra la cota teórica de tráfico frente al número de conexiones *multicast*. La zona permitida está situada bajo la curva. No se refleja el límite máximo de 16 usuarios, impuesto por nuestro sistema.

5. Conclusiones

El proyecto HIAD es un proyecto ambicioso, validado por uno de los principales operadores europeos de telecomunicación, en el que se quiere concentrar, en un mismo dispositivo, distintas tecnologías de acceso. No obstante, uno de nuestros principales objetivos es un prototipo para adquirir experiencia en el desarrollo de sistemas telemáticos de banda ancha.

En la fase actual se ha logrado implementar la parte de concentración de ADSL, lo que ha generado amplia información y experiencia para poder añadir más funcionalidad y poder llegar a un dispositivo HIAD pleno en un futuro cercano.

La placa de evaluación del circuito MC92501 de Motorola, ATMC-EVB, ha sido de gran ayuda, ya que nos permitió manejar los conceptos básicos. Sin embargo, según se avanzaba, se ha detectado falta de potencia para el equipo HIAD final. Definitivamente, se hace necesaria la inclusión de otro circuito que realice tareas complementarias. Una solución posible es el MC92400, anunciado por Motorola, pero no nos consta disponibilidad en el mercado.

6. Líneas futuras

Resta mucho trabajo para conseguir un equipo HIAD definitivo. Se tiene que tomar una decisión sobre el sucesor del MC92501. Se ha citado al MC92400, pero otra opción es un cambio de tecnología. Una solución interesante es la familia de circuitos de Sierra, aunque también hemos barajado otras alternativas de Lucent o Transwitch. La migración desde el prototipo actual hacia los nuevos circuitos no sería difícil. La parte de control de SNMP es independiente del circuito que se elija. El resto del *software* deberá sufrir ligeros retoques, pero la arquitectura se mantiene.

En el caso de *multicast*, las prestaciones pueden mejorarse sensiblemente. Un algoritmo más elaborado mejoraría las prestaciones. Si tenemos en cuenta que lo único que hay que cambiar es la cabecera de la celda, se podría pensar en no replicar la carga útil. Un paso adicional sería considerar las características del tráfico. Los estándares de vídeo MPEG transmiten distintos tipos de paquetes. Algunos de ellos son más prioritarios que otros. En momentos de alta congestión, se podría optar por descartar paquetes de baja prioridad. Un algoritmo posible sería el siguiente:

- Se lee el registro CR0 de la cola de extracción para comprobar que una celda

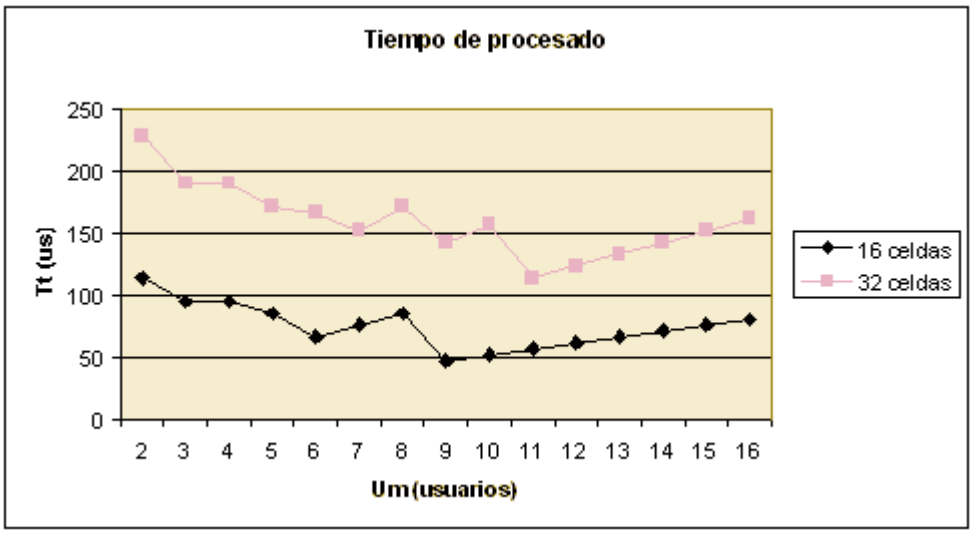


Fig. 4: Tiempo de procesado

pertenece a una conexión *multicast* (pueden existir celdas *unicast* que vayan a esa cola). Si es así:.

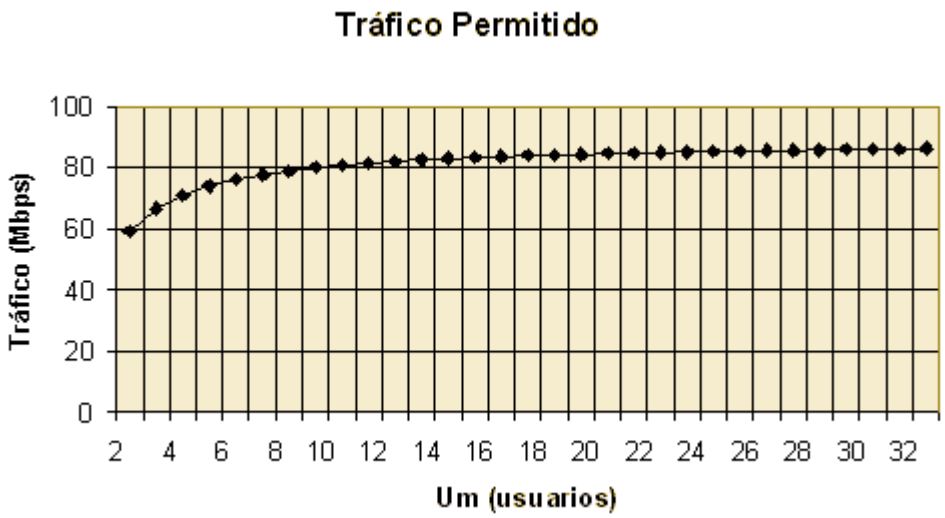


Fig. 5: Tráfico soportado

- Se lee el registro CR1 para obtener el ECI.

- En una situación de fuerte carga, se puede optar por el descarte. Leyendo la información exacta de la celda, podemos averiguar si es importante o no. En caso afirmativo se prosigue con el algoritmo:

  Comienza la réplica. Se cambia la cabecera de cada celda, pero la carga útil se escribe una sola vez . Para *N* usuarios, sólo tendremos que escribir N×5 + 48 B, ya que la cabecera ocupa 5 B y la carga útil 48.

La reducción del tiempo de escritura es considerable, aunque como vimos en párrafos anteriores, nuestro principal problema es la falta de espacio en la cola de inserción.

Por último, citar que existen algunos mecanismos del MC92501 que no se han probado en el prototipo. El caudal máximo permitido es de 155 Mbps [4], por lo que, si el tráfico de pico es excesivo, se deben activar los mecanismos necesarios para controlar la salida y evitar pérdidas.

El MC92501 posee *leaky buckets* para realizar este control. Se dispone de hasta 4 *leaky buckets* por conexión, lo que permite soportar pequeñas ráfagas sin pérdidas. El tamaño de cada *leaky bucket* se configura en la iniciación de cada conexión.

Referencias

[1] MOTOROLA, “MPC860 PowerQUICC – User’s Manual”, 1996.

[2] MOTOROLA, “ATM Cell Processor Evaluation Board User´s Manual”.

[3] Integrated Systems, “pRISM+ for pSOSystem”.

[4] WindRiver Systems, “VxWorks Programer’s Guide”.

[5] MOTOROLA, “MC92501 – ATM Cell Processor User´s Manual”.

[6] CYPRESS Semiconductor Corporation, “AX ATM-SONET/SDH Transceiver”, 1999.

[7] McCloghrie K., Heinanen J., Greene W. y Prasad A., “Managed Objects for Controlling the Collection and Storage of Accounting formation for Connection-Oriented Networks”, RFC2513, Feb. 1999.

[8] McCloghrie K., Heinanen J., Greene W. y Prasad A., “Accounting Information for ATM Networks”, RFC2512, Feb. 1999.

Diseño de un Sistema para el Desarrollo de Aplicaciones en Entornos LAN-WLAN*

Elsa Mª Macías, Alvaro Suárez, Carmen Nieves Ojeda, Ernesto Robayna
Departamento de Ingeniería Telemática. Universidad de Las Palmas de Gran Canaria.
Edificio de Electrónica y Telecomunicación. Campus Universitario de Tafira, ULPGC.
35017 Las Palmas de Gran Canaria
Teléfono: 928 45 80 54 Fax: 928 45 12 43
E-mail: elsa@cic.teleco.ulpgc.es

Abstract. *In the recent past there has been an important development in communications technologies. For example we can mention the Internet and the World Wide Web (WWW). On the other hand, recent advances in wireless communications have become the Wireless Local Area Networks (WLANs) a viable option for performing distributed computing. Wireless technology is rapidly becoming a crucial component of computer networks, and its use is growing by leaps and bounds. In this paper it is presented a whole system for implementing distributed applications. The novel system includes a new nodes management protocol that hides the communications latency overlapping calculations and communications. It has also been developed a communication library that encapsulates this protocol to make easy the programming to the user. Finally, a Web interface to access to the system via Internet has also been implemented.*

1 Introducción

En la actualidad, las comunicaciones están experimentando un avance espectacular. Aunque la Internet y el WEB [1] sea quizás la red de dominio público más extendida, existen también otras infraestructuras de red que están alcanzando un importante desarrollo. Un ejemplo son las Redes de Área Local Inalámbricas (WLANs, de las siglas en inglés *Wireless Local Area Networks*) [2]. Junto a las tecnologías de comunicación inalámbricas se han desarrollado una serie de protocolos portadores como son: Bluetooth [3], HomeRF [4], y recientemente se ha liberado el estándar IEEE 802.11b [5]. El interés despertado por este último se basa en que permite la comunicación a 11 Mbps, existiendo ya diversos fabricantes que construyen tarjetas de comunicación que funcionan a esta velocidad. Estas tarjetas se pueden acoplar a computadores personales de sobremesa o portátiles que proporcionan una relación coste/rendimiento aceptable. Sobre estos protocolos se instala el TCP/IP [6] para obtener conectividad a Internet.

El uso de redes de área local (LANs, de las siglas en inglés *Local Area Networks*) para la ejecución de aplicaciones distribuidas se ha experimentado en repetidas ocasiones, desde principios de la década pasada [7]. El trabajo realizado en esta área ha sido muy activo, y últimamente, con el aumento creciente de potencia en los computadores personales portátiles y el aumento de la velocidad de las comunicaciones inalámbricas, han motivado que varios autores hayan pensado en usar las WLANs para ejecutar aplicaciones distribuidas intensivas en cálculo [8][9][10]. Estas aplicaciones distribuidas se suelen programar usando bibliotecas de comunicación y gestión de máquinas virtuales (estas bibliotecas están implantadas usando el TCP/IP como base). Una biblioteca estándar es MPI [11] (*Message Passing Interface*), que facilita el desarrollo de aplicaciones distribuidas sobre LANs. Recientemente se ha liberado la última versión del estándar MPI, denominado MPI-2 [12]. Sobre estas bibliotecas se implementan entornos integrados como el MPICH [13] que es uno de los desarrollos de libre distribución más ampliamente usado dentro de los que cumplen con el estándar MPI-1.

Para ejecutar programas sobre una WLAN se requiere, como característica fundamental, la creación dinámica de procesos (la máquina virtual varía en tiempo de ejecución debido a que los portátiles pueden "moverse", "entrando" y "saliendo" de la máquina virtual). Actualmente, MPICH no soporta esta funcionalidad, y según la información que obra en nuestro poder sobre las posibles implementaciones de la biblioteca MPI de libre distribución, solamente la implementación LAM/MPI [14] da soporte para el lanzamiento dinámico de procesos sobre máquinas específicas una vez iniciada la aplicación. Esta funcionalidad se consigue mediante la utilización de la función

* Este trabajo está parcialmente subvencionado por el Proyecto CICYT: TIC98-1115-C02-02.

MPI_Comm_spawn (definida en MPI-2) que expande un proceso específico sobre un nodo de la máquina virtual.

El sistema internacional más importante relacionado con nuestro trabajo es Condor [15]. En la actualidad, el sistema Condor es capaz de administrar recursos en un entorno heterogéneo y cambiante como podría ser el sistema que se propone en este artículo. Condor incluye un módulo opcional denominado MW (*Master Worker*) que asiste a los usuarios en el desarrollo e implementación de aplicaciones distribuidas que siguen el paradigma master-worker. La herramienta MW se apoya en el sistema Condor y el middleware de comunicación PVM [16]. Puesto que PVM no es un estándar, a diferencia de MPI, nosotros estamos interesados en utilizar la biblioteca de comunicación estándar MPI para el desarrollo e implementación de aplicaciones (que tienen dependencias de cálculo más complejas que las del modelo MW) en entornos LAN-WLAN. Bajo el sistema Condor también se pueden ejecutar aplicaciones que hagan uso de la biblioteca MPI, con la salvedad de que las tareas que se ejecutan en una máquina no pueden ser migradas a otra, debido a la restricción impuesta por la implementación de MPI utilizada por Condor [13]. Condor no permite la incorporación de nodos en la máquina virtual, dinámicamente en tiempo de ejecución. Uno de los logros de nuestro sistema es justamente ese.

En este artículo presentamos un sistema completo de desarrollo de aplicaciones distribuidas en un entorno heterogéneo y dinámico. En el apartado 2 se presenta la arquitectura del sistema utilizado. En el apartado 3 se presenta el protocolo propuesto para la gestión de los recursos inalámbricos en el que se logra ocultar la latencia de las comunicaciones no cableadas solapándolas con los cálculos de las aplicaciones. En el apartado 4 se presenta la biblioteca de comunicación diseñada que encapsula este protocolo para facilitar el diseño de las aplicaciones en el entorno LAN-WLAN. En el apartado 5 se presenta una interfaz WEB de acceso al sistema de desarrollo a través de Internet. Finalmente, en la sección 6 se resumen las conclusiones y se exponen algunas líneas de trabajo futuro.

2 La Infraestructura de Red

En la Fig. 1 se presenta la arquitectura del sistema que se utiliza para el desarrollo de aplicaciones en entornos LAN-WLAN. Los recursos de este sistema son tres: nodos en una LAN, conectados entre sí a través de interconexión cableada, portátiles en una WLAN que utilizan tarjetas de interfaz de red conectadas a un emisor/transmisor de radio, y un nodo de acceso que se conecta a ambas redes por medio de una interfaz cableada y otra interfaz de red inalámbrica.

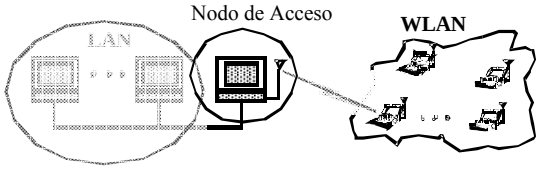


Figura 1: La Infraestructura de Red

En la implementación práctica de nuestro sistema hemos utilizado servidores Linux (RedHat 6.2 y 7). El host anfitrión (nodo de acceso) es un dual Pentium Pro 266 MHz (Kernel 2.2.16). Como nodos fijos hemos integrado un tetra Pentium Power Dell y varios monoprocesadores Pentium Celeron 600 MHz. Los nodos portátiles son también monoprocesadores Pentium Celeron 600 MHz. Los interfaces alámbricos se realizaron mediante la interconexión de tarjetas EtherPro10/100 a un switch (10 Mbits exclusivos) y los inalámbricos mediante tarjetas ImasDé [17] (tarjetas ISA y PCMCIA TDMA-CDMA 2 Mbits nominales pero altamente dependiente de la situación y el número de máquinas interconectadas). Las tarjetas inalámbricas cumplen con el estándar IEEE 802.11, operando en la banda ISM a 2.4 Ghz. La escasez de ancho de banda para las comunicaciones inalámbricas nos obliga a evitar la transmisión de paquetes no necesarios, por lo cual, en el nodo de acceso, que hace las tareas de enrutamiento y reencaminamiento entre ambas subredes (LAN y WLAN), se implementó un filtrado de paquetes (*ipchains* e *ip-masquerade*).

3 El Nuevo Protocolo

En este apartado presentamos las ideas básicas del nuevo protocolo diseñado para soportar la modificación, en tiempo real de ejecución, de la máquina virtual.

3.1 Entidades del Protocolo

Las entidades del protocolo de gestión de nodos son tres [18]:

Entidad Nodo de Acceso (ENA), encargada de: permitir el acceso remoto al entorno LAN-WLAN, conectar el resto de entidades del sistema, intercambiar mensajes entre las distintas entidades (de igual o diferente tipo), y gestionar la vinculación y desvinculación de entidades tipo *ENI*.

Entidad Nodo Cableado (ENC), que se comunica con otras entidades de su mismo tipo o con entidades de tipos diferentes (en este caso, a través de la *Entidad Nodo de Acceso*).

Entidad Nodo Inalámbrico (ENI), que se comunica con otras entidades a través de la *Entidad Nodo de Acceso*.

3.2 Primitivas del Protocolo

Las primitivas del protocolo de gestión de la máquina virtual son:

Vincular. Esta primitiva lleva a cabo la incorporación de un nodo inalámbrico en la WLAN. Los mensajes involucrados en esta primitiva se comunican entre la *ENI* y la *ENA:* petición de vinculación (*ENI* ® *ENA*) y confirmación de vinculación (*ENA* ® *ENI*). Esta situación se produce cuando un usuario introduce su portátil en el área de cobertura radio de la WLAN y ejecuta la aplicación WEB de adhesión a la máquina virtual.

Desvincular. Esta primitiva desvincula un nodo inalámbrico de la WLAN. Los mensajes involucrados en esta primitiva se realizan entre la *ENI* y la *ENA:* petición de desvinculación (*ENI* ® *ENA*) y confirmación de desvinculación (*ENA* ® *ENI*). Esta situación se produce cuando un usuario retira su portátil del área de cobertura radio de la WLAN ejecutando la aplicación WEB de desvinculación de la máquina virtual.

Enviar_Estado. Esta primitiva es la encargada de enviar a todos las entidades del sistema, el estado actual del mismo (*ENA* ® *ENI* y *ENA* ® *ENC*). Esta situación se produce cuando las entidades pueden aceptar la creación de nuevas *ENI*s y/o la liberación de *ENI*s antiguas.

Recibir_Estado. Esta primitiva es la encargada de recibir de la *ENA* el estado actual del sistema (*ENA* ® *ENI* y *ENA* ® *ENC*). Esta situación se produce cuando las entidades pueden aceptar la creación de nuevas *ENI*s y/o la liberación de *ENI*s antiguas.

Enviar_Datos. Esta primitiva es la encargada de enviar el código de la aplicación y demás información necesaria para su normal funcionamiento. Los mensajes involucrados en esta primitiva se realizan entre el acceso remoto y la *ENA* o entre la *ENA* y las entidades *ENI* /E*NC*. Esta situación se produce cuando se inicia una aplicación.

Recibir_Datos. Esta primitiva es la encargada de recibir el código de la aplicación y demás información necesaria para su normal funcionamiento. Los mensajes involucrados en esta primitiva se realizan entre la *ENA* y el resto de entidades (*ENI* y *ENC*). Esta situación se produce cuando se inicia una aplicación (en el caso de una *ENC*) o se vincula una *ENI*.

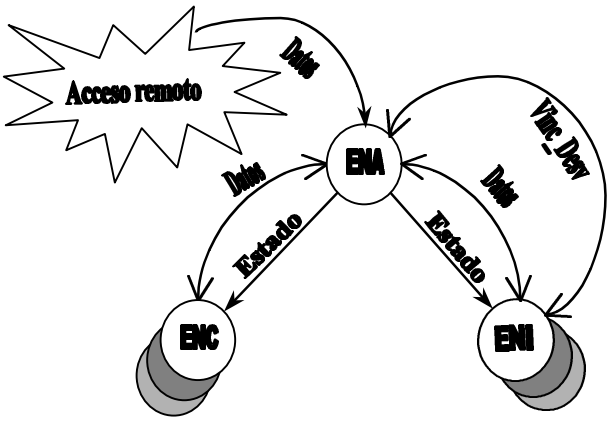
Finalizar. Esta primitiva es la encargada de fijar el estado de la WLAN porque mantiene las *ENCs* y *ENIs* existentes. Esta situación se produce cuando la *ENA* no permite la vinculación de nuevas entidades y la desvinculación de entidades existentes.

En la Fig. 2 se presenta una visión general de las distintas entidades y las primitivas de comunicación. En la Fig. 3 se muestra un ejemplo de intercambio de mensajes entre el nodo de acceso y los nodos cableados e inalámbricos.

4 Biblioteca de Comunicación

La programación de aplicaciones distribuidas con variación en tiempo real de la máquina virtual, es muy complicada haciendo uso de nuestro nuevo protocolo y de los mecanismos IPC (*Inter Process Communications*) soportados por el LINUX [19] y la biblioteca LAM/MPI. Por este motivo se ha implementado una biblioteca sencilla de usar y que facilita enormemente la programación de aplicaciones distribuidas en un entorno heterogéneo y dinámico.

La biblioteca de comunicación implementada, que la hemos denominado *LAMGAC*, proporciona al usuario un conjunto de tres funciones para atender las peticiones de vinculación y desvinculación de portátiles, modificación en tiempo de ejecución del entorno LAN-WLAN (estado de la red), y liberación de los recursos que se activan para atender a dichas peticiones. La biblioteca ha sido implementada en el lenguaje de alto nivel C, y utiliza dos de los tres mecanismos de comunicación entre procesos soportados el sistema operativo UNIX: semáforos y memoria compartida. La biblioteca también hace uso del middleware MPI para la comunicación entre entidades y el entorno LAM (envío/recepción de señales de usuario entre distintos procesos).



Leyenda

Estado: primitivas *Enviar_Estado* y *Recibir_Estado*

Vinc_Desv: primitivas *Vincular* y *Desvincular*

Datos: primitivas *Enviar_Datos* y *Recibir_Datos*

Figura 2: Entidades y Primitivas del Protocolo

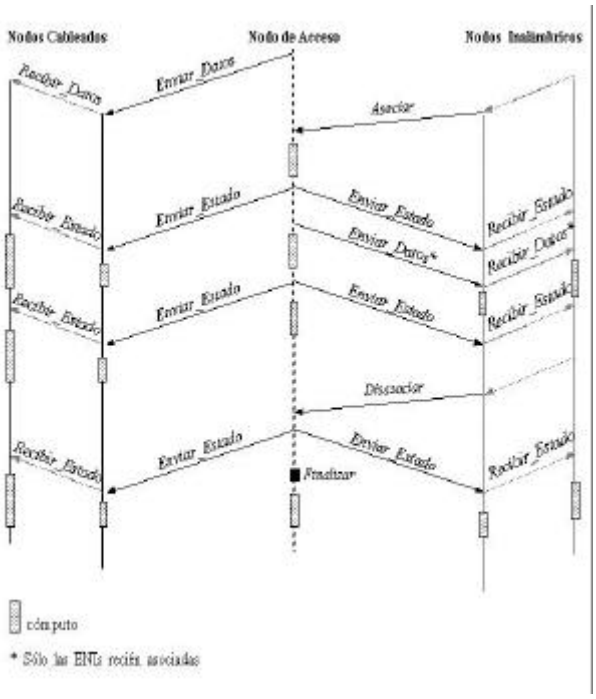


Figura 3: Intercambio de Mensajes

4.1Función *inicializar_LAMGAC*

La función *inicializar_LAMGAC* atiende peticiones de vinculación y desvinculación de nodos portátiles. Debe ser llamada por la *Entidad Nodo de Acceso* una sóla vez y antes de cualquier llamada a otra función de la biblioteca.

La declaración de esta función es:

*void inicializar_LAMGAC(char *argv[]);*

El parámetro *argv* se utiliza para crear la memoria que la *ENA* comparte con los procesos encargados de atender las peticiones de vinculación y desvinculación, así como la creación y utilización de los semáforos que manejan dicha memoria compartida. El parámetro *argv* se corresponde con el parámetro *argv* de la función *main*.

4.2 Función *actualizar_LAMGAC*

La función *actualizar_LAMGAC* actualiza el estado de la red, determinando el número de *ENIs* presentes en el sistema. Esta función también comunica dicho estado al resto de entidades, y asigna un identificador consecutivo a cada una de las *ENIs*.

La declaración en C de la función es la siguiente:

*tipo_LAM *actualizar_LAMGAC (int *my_id, int *NA, int *NI, char *prog_wireless, tipo_LAM *vector_LAM, MPI_Comm COMM);*

El valor devuelto por la función es un vector de tamaño igual al número de *ENIs* (*NI*) donde cada elemento es una estructura que almacena información relativa a un portátil: su nombre, su intercomunicador con la *Entidad Nodo de Acceso*, y un tercer campo que indica si el portátil es un nodo previamente vinculado (en una llamada anterior a la función *actualizar_LAMGAC*), o es un nodo recién vinculado (en la llamada actual a dicha función). Esta información le sirve al programador, después del retorno de la función, para diferenciar entre nuevos portátiles y portátiles ya presentes en el sistema. Por lo general, esta información suele utilizarse para enviar datos iniciales a los nodos recién vinculados.

El significado de los parámetros de la función *actualizar_LAMGAC* varía según el tipo del entidad que invoque a la función.

Para la *ENA*, el significado de los parámetros es el siguiente:

- *my_id* (parámetro de entrada: IN). Representa el identificador de la *ENA* (valor constante e igual a 0).
- *NA* (parámetro de entrada/salida: IN/OUT). Representa el número de *ENCs*.
- *NI* (IN/OUT). Representa el número de *ENIs* (inicialmente es 0).
- *prog_wireless* (IN). Nombre del ejecutable (y sus parámetros) que se lanza en los nodos portátiles.
- *vector_LAM* (IN). Vector que almacena el nombre e intercomunicador de cada portátil con la *ENA*. Asimismo indica si el portátil se acaba de vincular o ya existía en el sistema.
- *COMM* (IN). Comunicador formado únicamente por la *ENA*.

Para las *ENCs*:

- *my_id* (IN). Representa el identificador de la *ENC* (valor constante en el rango *1..NA*).
- *NA* (IN/OUT). Representa el número de *ENCs*.
- *NI* (parámetro de salida: OUT). Representa el número de *ENIs*.
- *prog_wireless.* No tiene efecto.
- *vector_LAM*. No tiene efecto.
- *COMM* (IN). Comunicador con el resto de *ENCs* y con la *ENA*.

Para las *ENIs*:

- *my_id* (IN/OUT). Representa el identificador de la entidad (valor constante en el rango *NA+1..NA+NI*).
- *NA* (IN/OUT). Representa el número de *ENCs*.
- *NI* (OUT). Representa el número de *ENIs*.
- *prog_wireless*. No tiene efecto.
- *vector_LAM*. No tiene efecto.
- *COMM* (IN). Intercomunicador con la *ENA*.

La función *actualizar_LAMGAC* retorna el vector *vector_LAM* actualizado. Debe ser llamada por todos las entidades actualmente involucradas en la ejecución de la aplicación, en el momento en que estén disponibles para actualizar el entorno LAN-WLAN (por ejemplo, cuando finalicen una determinada tarea de cálculo).

4.3 Función *finalizar_LAMGAC*

La rutina *finalizar_LAMGAC* es invocada por la *Entidad Nodo de Acceso* para liberar los mecanismos de comunicación de vinculación y desvinculación. Después de la llamada a esta función, la ejecución de la aplicación continúa en los recursos que en ese momento forman parte del entorno, no atendiéndose ninguna petición de vinculación o desvinculación posteriores de portátiles.

La declaración de esta función es:

void finalizar_LAMGAC();

5 Interfaz WEB

En este apartado presentamos la implementación de la interfaz de trabajo que usa el programador o un usuario que quiere observar cómo funciona el sistema LAN-WLAN. También presentamos la implementación WEB de estas interfaces. Para hacer esta presentación distinguimos varios tipos de usuarios y el software que se ejecuta en el nodo de acceso.

5.1 Tipos de Usuarios

Distinguimos dos tipos de usuarios: el que cede su máquina y su cuenta para que él mismo y otros ejecuten parte de la aplicación sobre ella (*usuario invitado*), y el que entrega trabajos para que sean ejecutados en el sistema (*usuario de computación*). Estos usuarios utilizan el sistema haciendo uso de una interfaz *WEB* cliente-servidor con soporte para *CGIs* (*Common Gateway Interfaces*) mediante los cuales se envían aplicaciones al sistema o se oferta una nueva máquina.

El *usuario invitado* puede ofertar una nueva máquina al sistema si cumple los siguientes requisitos:

- Debe tener instalado la pila de protocolos TCP/IP y permitir ejecución remota de procesos mediante SSH [20]. Se elige SSH por la facilidad de implementación por parte del cliente (publicamos en el servidor WWW la clave pública del usuario LAM del nodo de acceso), y por la seguridad que conlleva sin introducir una excesiva latencia.
- Debe disponer del *scp* del SSH para donar la cuenta de usuario sin ayuda del supervisor, y así ejecutar una copia del código fuente de la aplicación distribuida. De esta manera se podrá compilar, ejecutar y soportar sistemas de arquitecturas heterogéneas. Esto tiene la ventaja frente a NFS y CODA [21] de que no se sobrecarga el escaso ancho de banda inalámbrico.
- Debe tener instalada la biblioteca LAM/MPI y nuestra biblioteca de comunicación y gestión de la máquina virtual.

Si el usuario cumple todos estos requisitos entonces puede enviar (mediante un CGI invocado desde su página WEB) la oferta de la nueva máquina. Para ello se han diseñado dos mecanismos diferentes: en el primero, el nodo de acceso envía una página WEB al usuario invitado con todas las aplicaciones en ejecución. El usuario, mediante un formulario WEB, elige la aplicación a la que quiere ofertar su máquina (el nodo de acceso modifica dinámicamente la máquina virtual de esa aplicación). En el segundo caso, el usuario simplemente oferta su máquina y el nodo de acceso actualiza la base de datos de máquinas disponibles para cualquier aplicación. En la Fig. 4 se muestra la interfaz WEB usada por este tipo de usuarios.

Figura 4: Interfaz Web para el *usuario invitado*

Por otro lado, el *usuario de computación* debe estar autentificado en el servidor WEB del nodo de acceso. Cuando accede al servidor WEB, éste le devuelve una página de envío de trabajos de computación (*página de subscripción*).

En esta página existe un CGI con el mecanismo *upload* de *HTTP* para enviar los archivos con el código fuente y los datos de entrada de la aplicación distribuida. Es importante tener en cuenta que este tipo de usuario no tiene por qué tener su máquina dentro del sistema LAN-WLAN, sino que puede ser cualquier usuario conectado a Internet que tenga y que necesite usar nuestro sistema. Se permite de esta forma que cualquier usuario de Internet pueda utilizar, de forma segura, el entorno desarrollado.

En la Fig. 5 se muestra una interfaz para el envío de los archivos de código y los datos de entrada.

5.2 Software WWW del Nodo de Acceso

En el nodo de acceso existe un software básico WWW para manejar toda la gestión de la ejecución de las aplicaciones. Además, existe un software de presentación de estos datos a los usuarios, en tiempo real. Esta información de gestión es la siguiente: nodos que se encuentran disponibles, aviso de la entrada o salida de un nodo, procesos en ejecución y su estado en cada nodo, datos de la monitorización de las comunicaciones, etc. Esa información se mantiene en una base de datos *PostgreSQL*. Esta base de datos es accesible desde cualquier punto de Internet puesto que a partir de ella se generan páginas WEB dinámicas de monitorización y un mapa gráfico que muestra el estado de las máquinas, así como las comunicaciones.

En este nodo existe un software base para controlar la oferta de una nueva máquina al sistema o la retirada de una existente (por parte del *usuario invitado*). También regula la incorporación a la máquina virtual, en tiempo real, de nuevos procesos de ejecución que forman parte de la aplicación distribuida. Esta adición se puede hacer desde cualquier nodo de Internet si el usuario está bien autentificado para poder controlar el número de procesos en ejecución. Cuando el sistema esté disponible, el nodo de acceso distribuye el código fuente entre los nodos (creando directorios temporales), compila el código tanto en la máquina local como en las remotas, crea los ficheros esenciales para la ejecución inicial de la aplicación (*application schema* de la LAM) y ordena su ejecución. Al mismo tiempo actualiza las tablas pertinentes de la base de datos (almacenando información de nodos, programas, directorios, etc) y lanza los demonios de monitorización de comunicaciones en el nodo local y en los remotos. Tanto si todas las acciones se han realizado con éxito como si no, el nodo de acceso notificará al *usuario de computación*, mediante correo electrónico, los resultados de la ejecución o los errores detectados.

Asimismo, existe un servidor WEB que se encarga de la comunicación con los usuarios y de la ejecución de los CGIs. Por ejemplo, al *usuario de computación* le envía datos sobre el estado de la máquina virtual o los resultados de una ejecución. Por otro lado, se hace uso del usuario *wwwrun* de la versión 1.3 del servidor Apache [22], para poder lanzar órdenes en el nodo de acceso (como si fuera el usuario propietario de la aplicación distribuida). Esta facilidad se consigue con un módulo externo llamado *suEXEC*.

6 Conclusiones y Trabajo Futuro

En este artículo hemos presentado un nuevo entorno (*framework*) para la programación efectiva de aplicaciones distribuidas intensivas en cálculo que integra un middleware de comunicación, cuyo diseño es novedoso y que permite utilizar nodos cableados e inalámbricos de una infraestructura de comunicación que combina LANs y WLANs. Para facilitar la programación hemos diseñado e implementado una biblioteca sencilla de usar por los programadores. Finalmente integramos una interfaz WEB que permite un fácil uso de la máquina virtual para vincular y/o desvincular los portátiles dentro de la máquina virtual, así como la gestión de la misma desde cualquier nodo de Internet. Nuestro sistema tiene la ventaja, sobre los otros que conocemos, de permitir la variación de la máquina virtual en tiempo de ejecución, de gestionarla eficientemente a través del WEB y de conocer su estado desde cualquier nodo de conexión a Internet.

El interés de este trabajo es diverso: por un lado permite experimentar con los problemas intrínsecos de las comunicaciones inalámbricas, el manejo de la movilidad y los protocolos básicos de comunicación inalámbricos. Diversos autores

reconocen que el siguiente paso necesario en este tipo de redes es la experimentación con diversas aplicaciones para demostrar que su uso es efectivo frente a las redes cableadas. Por otro lado, nuestro sistema nos permite experimentar algoritmos de acceso inalámbrico a Internet y plantear formas novedosas de gestión distribuida de recursos de comunicación cuando éstos son móviles.

En la actualidad estamos experimentando con diversas aplicaciones.

Referencias

[1] The World Wide Web Consortium. http://www.w3.org/[1].

[2] J. Geier. "Wireless LANs. Implementing Interoperable Networks". Macmillan Technical Publishing, 1999. ISBN: 1-57870-081-7.

[3] The Official Bluetooth SIG Website. http://www.bluetooth.com/[1].

[4] The HomeRF Website. http://www.homerf.org/[1].

[5] IEEE Standards Products Catalog: Wireless (802.11). http://standards.ieee.org/catalog/IEEE802.11.html[1].

[6] D. E. Comer. "Redes Globales de Información con Internet y TCP/IP". Prentice Hall (3ª Edición.), 1996. ISBN 968-880-541-6.

[7] V. S. Sunderam. "PVM: A framework for Parallel Distributed Computing". Concurrency: Practice and Experience, 27 páginas, Diciembre de 1990. http://www.mathcs.emory.edu/~vss/pvmsystem.ps.Z[1].

[8] Z. Haihong, R. Buyya, S. Bhattacharya. "Mobile Cluster Computing and Timeliness Issues". Informatica 17, 1999.

[9] S. Janche, C-H M. Lin, M-C Wang. "Experiences with Network-Based Computing over Wireless Links". International Journal of Parallel and Distributed Systems & Networks, vol. 2, nº. 2, 79-87, 1999.

[10] A. Suárez, E. Macías. "Management of Portable Nodes in a WLAN-LAN Cluster". Proceedings on Systemics, Cybernetics and Informatics. SCI 2000. Orlando, Florida, USA, 151-155, Julio 2000. ISBN: 980-07-6693-6.

[11] P. S. Pacheco. "Parallel Programming with MPI". Morgan Kaufmann Publishers, Inc., 1997. ISBN: 1-55860-339-5.

[12] Message Passing Interface (MPI) Forum Home Page. http://www.mpi-forum.org/index.html[1].

[13] MPICH- A Portable MPI Implementation. http://www-unix.mcs.anl.gov/mpi/mpich/[1].

[14] LAM / MPI Parallel Computing. http://www.mpi.nd.edu/lam/[1].

[15] Condor Project Homepage. http://www.cs.wisc.edu/condor/[1].

[16] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, B. Mancheck, V. Sunderam. "PVM:Parallel Virtual Machine. A users's guide and Tutorial for Network parallel Computing". The MIT Press, 1994.

[17] ImasDé: WLAN IEEE 802.11 ISA Plug&Play. http://www.imasde.com/wlan/wlan_e[1].

[18] E. Macías, A. Suárez, C. N. Ojeda, L. Gómez. "A Novel Programming Strategy Considering Portable Wireless Nodes in a Heterogeneous Cluster". Proceedings on the Distributed and Parallel Systems. DAPSYS 2000. Balatonfüred, Hungría, 185-194, Septiembre 2000. Kluwer Academic Publishers. ISBN: 0-7923-7892-X.

[19] The Linux Home Page at Linux Online. http://www.linux.org/[1].

[20] SSH Front Page. http://www.ssh.com/[1].

[21] Coda File System. http://www.coda.cs.cmu.edu/[1].

[22] The Apache Software Foundation. http://www.apache.org/[1].

[1] Información disponible a fecha 28 de marzo de 2001

Análisis de la integración entre tráfico IP y redes ATM. Simulador MPLS

Miguel Ángel Martín Tardío – Miguel Gaspar Rodríguez - José Luis González Sánchez
Área de Ingeniería Telemática. Departamento de Informática. Universidad de Extremadura
Campus de Badajoz. Facultad de Ciencias. Edificio de Matemáticas.
Teléfono: 924207541 Fax: 924205604 E-mail: matardio@unex.es

Abstract. *Current IP networks are a long way form meeting the requeriments of service providers an their customers. Multi-Protocol Label Switching (MPLS) represents the next level of standars-based evolution in combining layer 2 switching technologies with layer 3 routing technologies. MPLS is designed to work with a wide variety of transport mechanism; however, the initial implementations will focus on leveraging ATM which are already deployed in large-scale service provider networks. This documents provides an overview of the IP-ATM integration technologies, reviews the capabilities of MPLS and presents a MPLS Simulator making with Java. In other way, this open simulator will want to be a common educational tool for MPLS Technologies research.*

1 Introducción

Actualmente, IP es la solución que destaca a la hora de proporcionar servicios a través de Internet. Esto supone que con unos adecuados cambios tecnológicos podrá fortalecerse hasta convertirse en los cimientos que asienten la *transferencia de datos* a través de Internet, permitiendo la difusión de servicios avanzados en tiempo real y de respuesta prioritaria, entre los que podemos destacar:

- Proyectos de Teleformación y Aula virtual interactiva.
- Servicios avanzados de atención médica y hospitalaria. Telemedicina, teleasistencia o telecirugía.
- Video a la carta, multidifusiones, servicios multimedia interactivos o Telespectáculos.
- Transmisión de voz a través de Internet y las construcción de redes privadas virtuales (VPNs).
- Servicios de calidad para voz y fax sobre IP.

Estos servicios exigirán una serie de necesidades crecientes para los ISP y carriers que deberán incorporar a sus redes troncales mecanismos para mantener un *control sobre el tráfico*, una *gestión de recursos*, un *ancho de banda disponible*, así como una adecuada *calidad de servicio (QoS).*

A su vez, los avances en las tecnologías de transporte basadas en *switching*, encabezadas por la tecnología ATM, proporcionan alta velocidad, calidad de servicio y facilitan la gestión de los recursos en la red, que hemos comentado anteriormente. Comprobando cómo el protocolo de Internet actual, IP, puede beneficiarse de las características que aportan estas nuevas tecnologías parece claro buscar soluciones para conseguir un escenario de integración IP-ATM.

El principal objetivo, por tanto, de este trabajo es analizar y dar a conocer las soluciones que existen actualmente al problema de integrar tráfico IP dentro de la tecnología ATM; destacando a MPLS como la solución más completa desde una visión práctica a partir de un simulador MPLS implementado en Java, como herramienta destinada al estudio e investigación de esta tecnología.

2 Soluciones IP sobre ATM

A continuación, hacemos una breve revisión a las tecnologías surgidas para la integración IP-ATM, que sirven de justificación para la búsqueda de una solución definitiva como es MPLS.

2.1 IP Clásico y ARP sobre ATM

El modelo clásico ha sido la primera solución empleada para adaptar IP–ATM, basado en la recomendación RFC 1577 y 2225 [1], que trata cómo implementar una red de tal forma que sea posible la transmisión de datagramas IP junto con tramas ATMARP sobre la capa de adaptación 5 de ATM (AAL5).

La base del modelo es el LIS (Logical IP Subnet) ó Subred Lógica IP (Fig. 1), que representa a un conjunto de equipos que se comunican entre sí mediante ATM. Los equipos extremos del LIS hacen de puente entre los equipos internos del LIS y equipos externos. Para comunicarse con el exterior, emplean IP.

2.2 IP Switching

Nació como respuesta a los requerimientos de mantener una convergencia entre el routing de IP y la switching ATM [2]. El nombre de *IP switching,* (conmutación IP) engloba dos términos antagónicos. *IP*: un protocolo no orientado a conexión, basado en routing, y *Switching*: método empleado por las redes ATM, gracias a la cual son posibles todas sus funcionalidades. Se basa en un mecanismo de intercambio de etiquetas al igual que Tag Switching.

Se caracteriza por no ocultar la verdadera topología de la red a la capa IP, evitando, entre otras cosas complejidad y duplicación de funcionalidad.. Sin embargo, ciertos artículos que analizan tests sobre este sistema [3], [4] hablan de que el protocolo no ha sido lo suficientemente cuidadoso con el manejo de direcciones. Esto conlleva que su escalabilidad se ve limitada en redes grandes debido al gran número de circuitos virtuales que se requieren para cada conexión.

2.3 Cell Switch Routers (CSR)

Muy parecida a IP Switching, se diferencia de ésta en que su intención, según Toshiba [5], es conectar subredes IP locales en ATM (LIS), corriendo bajo LANE o IP Clásico sobre ATM (pero no subredes IP que no posean enlaces ATM y empleen routing estándar).

La conexión entre los LIS ATM y las redes no ATM se lleva a cabo mediante routers estándar, como LANE e IP Clásico sobre ATM. Los datos de control entre los CSR emplean el estándar UNI 3.1 Q.2931, y la resolución de direcciones depende de servidores ATMARP e InATMARP.

Como IP Switching, el router CSR puede realizar tanto Cell Switching como envío de paquetes IP. La descripción del envío de paquetes IP por defecto y los procedimientos de establecimiento de conexión ATM son idénticos a los de IP Switching. Las condiciones bajo las que se crean Circuitos Virtuales específicos ATM no han sido definidas claramente en la descripción de CSR. Las ventajas y desventajas de CSR son similares a IP Switching.

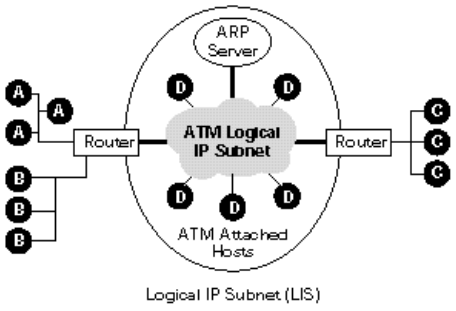


Fig. 1: Subred lógica IP (LIS)

2.4 LAN Emulation

LAN Emulation (LANE) fue elaborada por el ATM forum [6] con el propósito de definir mecanismos que posibiliten la coexistencia de las redes ATM con los sistemas LAN ya instalados.

LANE opera en el nivel MAC y permite que el tráfico Ethernet, Token Ring o FDDI se transmita sobre ATM sin alterar el código de las aplicaciones ni los sistemas operativos de red o las tarjetas de red de las estaciones de trabajo.

2.5 Multi-Protocol Over ATM (MPOA)

El ATM Forum [7] ha trabajado en cooperación con la IETF para desarrollar MPOA (Multi Protocolo Sobre ATM) [8]. Se trata de una solución de routing para la capa de red que integra los protocolos existentes y los estándares para dar funciones de routing sobre las redes ATM. Confiere escalabilidad y flexibilidad introduciendo un concepto conocido como *router virtual*. Éste emula la funcionalidad de los routers tradicionales y elimina las limitaciones en el rendimiento que tiene el routing *salto a salto*.

Para ello se establecerán atajos sobre el conmutador ATM entre cualquier host o dispositivo en los extremos con capacidades MPOA, independientemente de la subred a la que pertenezcan. En resumen: MPOA identifica flujos de datos y los mapea directamente en canales virtuales ATM. MPOA emplea tres técnicas complementarias: LANE, NHRP y el concepto del r*outer virtual*.

2.6 NHRP (Next Hop Resolution Protocol)

Definido en el RFC 2332 puede admitir rutas de atajo para eliminar los saltos adicionales que se necesitan en el modelo clásico. Puede emplearse tanto sobre subredes NBMA[1] [8] no orientadas a conexión (SMDS), como sobre subredes NBMA orientadas a conexión (ATM), de modo que no incluye mecanismos para el establecimiento de la conexión en este último caso. Estos deberán ser dados por otros protocolos, como MPOA. Aunque NHRP resuelve el problema de los saltos, introduce otros:

- La necesidad de una implementación contigua de los NHS, que pueden no durar mucho en los backbones ATM existentes en Internet.

- Está enfocado solamente al routing unicast. Puede soportar multicast donde puedan ser empleados los atajos *punto a multipunto* para evitar saltos extras, pero esta solución es probable que no sea escalable.

- El emisor se verá saturado si intenta establecer atajos para un número muy grande de receptores este

[1] Non Broadcast Multi Access Networks

problema es incluso peor en entornos con IPv6, donde el multicast es un elemento indispensable.

2.7 Tag Switching

Es una solución al direccionamiento de paquetes en el nivel de red. Su funcionamiento coincide en varios aspectos con IP Switching [9] [10], fundamentalmente en que ambos emplean *etiquetas* para identificar flujos de datos. También coincide en muchos aspectos con ARIS (Aggregate Route based IP Switching), fundamentalmente en que la asociación de Etiquetas con Circuitos Virtuales es determinada por la topología de la red.

Existen dos componentes principales en Tag Switching: la *componente de envío* y la *componente de control*. El envío es llevado a cabo usando técnicas de intercambio de etiquetas. Posteriormente, la componente de control se encarga de distribuirlas. Tag switching puede mantener las propiedades de IP, ayudar a mejorar la escalabilidad en estas redes y puede ser aplicado de manera directa sobre los conmutadores ATM.

2.8 Aggregate Route – Based IP Switching (ARIS)

Se trata de una solución que está destinada a ser usada con tecnologías que empleen el switching, como es ATM, conmutadores que lleven Frame switching o conmutadores LAN.

ARIS permite que el switching (en la capa 2) sea empleado para el envío de datagramas IP. Aumenta el empleo de los conmutadores ATM que hayan sido diseñados específicamente para mezclar circuitos virtuales (aunque no es un requerimiento obligatorio).

2.9 Multi-Protocol Label Switching (MPLS)

MPLS es la solución más reciente de todas las citadas. Diseñada originalmente por un grupo de trabajo: Ross Callon (Ascend), Arun Viswanathan (IBM), Eric Rosen (Cisco) [11], ha asimilado los puntos fuertes de soluciones como MPOA o IP Switching para conseguir una solución sólida y con muchas posibilidades de convertirse en un estándar de la industria, avalado por la IEEE [12].

Seguidamente pasaremos a ver las características de esta solución con más detalle, continuando con una explicación sobre la herramienta del Simulador desarrollado.

3 MPLS. Hacia la integración efectiva IP-ATM

3.1 Orígenes

MPLS nace con la idea de integrar las mejores características de la Capa 2 y la Capa 3, sustituyendo a las actuales técnicas de túneles (*tunneling*) para encaminar tráfico IP sobre ATM. Tecnologías como las anteriormente vistas, han servido de base, incluyendo alguna variación para su estandarización.

Las principales premisas a la hora de definir el estándar, han sido las siguientes:

- Debía funcionar sobre cualquier tecnología de transporte, no sólo sobre ATM. Es decir, deberá ser independiente de la capa que corra por bajo.
- Soportaría tanto envíos Unicast como Multicast.
- Compatible con RSVP y los servicios integrados de la IETF.
- Compatible con los actuales modos de funcionamiento de las redes IP existentes.
- Deberá admitir herramientas de soporte, administración y mantenimiento al menos tan fiables como las que soportan las redes actuales IPv4.
- Permitir un crecimiento constante de Internet, ampliando su capacidad para convertirse en un medio universal para el transporte de datos, no sólo para la distribución de contenidos.

Los primeros trabajos mediante MPLS se centraron en integrar IPv4 sobre ATM (también sobre Frame Relay) demostrando que se podían salvar los problemas que hasta ahora habían aparecido para llevar a cabo esa integración (y que se han expuesto a lo largo de los puntos anteriores).

3.2 Visión general

En sí, trata de emplear los conmutadores como Routers de *Label Switching* o Routers de conmutación de etiquetas. Los conmutadores ATM ejecutan algoritmos de routing de la capa de red, y el envío de sus datos se basa en los resultados de esos algoritmos de routing. No se necesita direccionamiento ni routing específico para ATM. Los conmutadores ATM que se emplean de esta manera son conocidos como ATM-LSRs.

Los puntos principales de MPLS son:

- Etiqueta: clasificación de paquetes que se enviarán por el mismo camino.
- Las etiquetas se asocian en la entrada de la red MPLS.
- El envío de paquetes se basa en la etiqueta.
- Las etiquetas se eliminan en la salida de la red MPLS.

- El criterio empleado para clasificar los paquetes en etiquetas se puede basar en una decisión local, al entrar en la red MPLS o en base a decisiones preestablecidas.

- Las etiquetas asignadas deben comunicarse a todos los nodos a lo largo del camino de la clase de paquetes asociados con la etiqueta.

- Las etiquetas pueden apilarse: un paquete puede tener varias etiquetas.

- LSR: router – conmutador que soporta MPLS.

Por tanto, la idea clave de funcionamiento de MPLS se encuentra en la identificación y mapeado de los paquetes IP con etiquetas, las cuales se utilizan a partir de ese instante para el establecimiento de los caminos LSP (Label Switching Path) por la red.

Un red que utiliza este tipo de tecnología (Fig. 2) se dice que forma parte del "Dominio MPLS" constituido por los LSP que se establecen entre los LSR (Label Switching Router) y los LER (Label Edge Router).

Los LER son los dispositivos que se encuentran en los límites del dominio MPLS, y tienen la capacidad de utilizar la información de enrutamiento para asignar las etiquetas a los paquetes IP y reenviar éstos a través de este dominio.

Los LSR son los dispositivos que residen dentro del dominio MPLS con la capacidad de reenviar y distribuir paquetes con etiquetas. Un LSR o "conmutador de etiquetas", se configura como un router especializado en el envío de estos paquetes etiquetados por MPLS. Podemos considerar a un LSR como un típico conmutador ATM modificado para el envío de este tipo de datagrama.

Por tanto, la identidad del paquete IP original queda enmascarada durante el transporte por el dominio MPLS. Los LSR sólo miran las etiquetas de esos paquetes para tomar la decisión de salto (hops) que configuran los LSP.

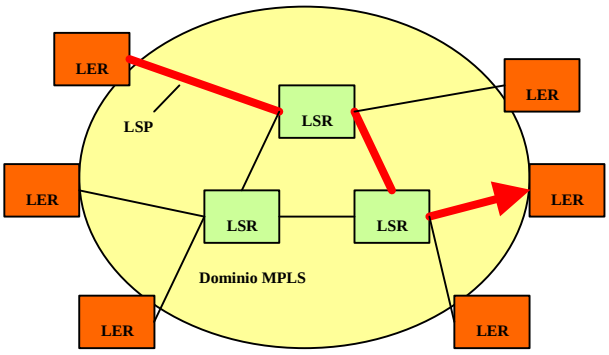


Fig. 2: Dominio MPLS

3.3 Conceptos erróneos sobre MPLS

Se ha extendido la idea de que MPLS es un estándar que permite a los fabricantes transformar los conmutadores ATM en routers backbone de altas prestaciones. Sin embargo actualmente, los avances en la tecnología están consiguiendo que motores de búsqueda de rutas IP basados en ASIC (application-specific integrated circuit) sean tan rápidos como MPLS o motores de búsqueda ATM. De modo que la ventaja que MPLS aporta al rendimiento del envío de paquetes no es la primera ventaja.

También se piensa que MPLS fue diseñado para eliminar por completo el routing IP. Esto no ha sido nunca un objetivo del Grupo de Trabajo de MPLS, porque sus miembros entienden que el routing tradicional en la capa 3 siempre será necesario en Internet:

- El filtrado de paquetes es un componente fundamental para la seguridad y administración en la red, y requiere un examen detallado de las cabeceras de los paquetes; el routing de la capa 3 sigue siendo necesario.

- Es improbable que un gran número de hosts implementen MPLS, por tanto, cada paquete transmitido necesita ser enviado por un dispositivo en la capa 3, donde la cabecera pueda ser examinada antes de enviarse al destino. Este router puede, tanto enviar el paquete empleando el routing convencional, como asignarle una etiqueta y enviar el paquete por un camino etiquetado conmutado (LSP).

- En el último salto anterior al host de destino, el paquete debe ser enviado empleando routing porque no resulta práctico asignar una etiqueta por separado para cada host en la subred de destino.

4. Simulador MPLS: Una visión práctica

4.1 Planteamiento

Cuando nos planteamos el estudio de MPLS, consideramos la necesidad de contar con una herramienta de simulación que nos sirviese:

- Como análisis del comportamiento del tráfico y de la red.

- Como utilidad para explicar el funcionamiento de la integración IP – ATM.

- Como escenario para probar cambios en la topología de la red, o en el comportamiento de los algoritmos de encaminamiento.

Además, esta herramienta debería ser:

- Configurable: Poder alterar los parámetros de la red y del tráfico que circula a través de ella.

- Gráfica: Poder obtener conclusiones con facilidad.

- Rigurosa y analizable: los resultados revelarán el comportamiento de la red y los posibles problemas que pueden surgir.

- Portable: a ser posible, deberá ser código abierto y fácil de ejecutar en varios sistemas operativos, con la idea de dar pie a posibles ampliaciones en el futuro.

Los elementos mas característicos que esta aplicación debería manejar serían:

- Entrada de tráfico IP.

- Circulación de datos IP, ATM e IP-ATM.

- Circulación de signalling ATM e IP-ATM.

- Modificación del número de Nodos de la red y de los enlaces entre los nodos.

- Velocidad de envío de IP y de switching ATM.

- Modificación de los parámetros específicos de IP (tabla de routing) y ATM (tabla de switching) en cada nodo, calidad de servicio.

- Adaptación de los datagramas IP en paquetes ATM, empleando la capa AAL5 entre ambos protocolos.

- Modificación de los parámetros específicos de la solución elegida para adaptar IP – ATM.

A partir de estos elementos y su correspondiente análisis de interrelaciones, consideramos que en nuestro análisis de la solución simulada, y por el objetivo de la integración que se estudia, se debían considerar dos entidades principales y diferenciadas:

- *Entidad externa tipo IP*: los nodos frontera interaccionan directamente con routers IP, de tal manera que deben manejar datagramas IP, clasificarlos en Clases de Envío Equivalentes y asociarlas con etiquetas.

- *Entidad externa tipo ATM*: los datos procedentes de conmutadores ATM que no soporten MPLS deberán llevarse por los VP específicos.

4.2 Funcionamiento

4.2.1 Entradas del Simulador

Las únicas entradas al sistema serán las modificaciones sobre el simulador. Pueden venir de un fichero de configuraciones o directamente sobre la simulación según lo siguiente:

Datos genéricos de la red: Es una ventana que contiene los datos relativos a la información que maneja la red al realizar la simulación (Fig. 3):

- Número de nodos en la red: modificará cuantos LSR existen en la red.

- Velocidad de envío de datagramas IP: modificará la velocidad en paquetes/sec a la que serán enviados los datagramas IP a la red.

- Retardo de procesamiento de los nodos: cifra genérica que representa el tiempo que un nodo tarda en realizar un chequeo en todos sus interfaces de entrada sumado al tiempo que emplee en procesar un paquete ATM y enviarlo.

- Tamaño de datos de los datagramas IP: modificará el tamaño de datos, sin incluir la cabecera, que tendrán los datagramas IP, de tal manera que si el tamaño de datos, junto con la cabecera supera los 53 bytes del tamaño de célula ATM, será necesario realizar segmentación al entrar estos datos en la red.

- Debug: si se desean mostrar todos los datos, cuando el simulador entre en funcionamiento, o sólo mostrar el modo gráfico.

- Nombre del fichero de entrada: modificará el fichero que se está utilizando para obtener los datos de la red.

Datos específicos de cada nodo: Esta entrada aparece cuando se deseen visualizar los datos específicos de un LSR perteneciente a la red. Los datos que se podrán editar serán (Fig. 4):

- Tabla de Routing: los datos de la tabla de routing del nodo.

- LIB: los datos de la tabla Label Information Base del nodo. Esta tabla se mostrará en el caso de que el nodo no sea interno (los nodos internos no la necesitan).

- NHLFE: los datos de la tabla NHLFE, que asocia las etiquetas con las salidas de los datos.

Fig. 3: Entrada de datos genéricos

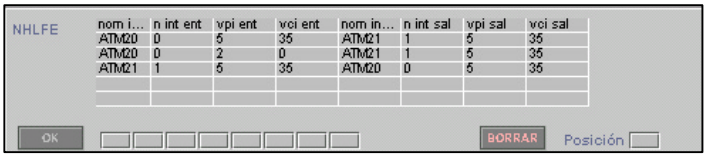


Fig. 4: Entrada de datos de la tabla NHLE

- Enlaces de entrada: mostrará los nodos de entrada a él junto con los interfaces de entrada asociados.
- Enlaces de salida: mostrará los nodos de salida a él junto con los interfaces de salida asociados.
- Frontera: indicará si es un nodo frontera de la red ó interno. En el caso de los nodos frontera, estos podrán tener enlaces con entidades IP.

Disposición de la red: Esta entrada visualizará:

- Localización de los nodo.
- Enlaces de nodos: indicará si se trata de un enlace de entrada o de salida (o mixto).
- Localización de las entidades IP.
- Enlaces de los nodos frontera con las entidades IP.

Funcionamiento de la red: Esta entrada dispone de los botones necesarios para controlar el funcionamiento de la red:

- Reset: recarga todos los datos originales de la red, respecto del fichero de configuraciones.
- Activar: pone en funcionamiento la red.
- Traza: ejecuta paso a paso el funcionamiento de la red.
- Detener: pone en pausa el funcionamiento de la red.
- Salir: saldrá de la aplicación

Además, los datos genéricos de la red, podrán modificarse sobre la marcha, durante el funcionamiento de la red

Fichero de configuraciones: Este fichero contiene toda la información necesaria para poner en marcha la simulación sin necesidad de tener que entrar datos por pantalla. Es de tipo texto (mpls.cfg).

4.2.2 Salidas del Simulador

Las salidas visibles del sistema serán descritas a continuación:

Datos genéricos de la red: Una ventana que contiene los datos relativos a la información que maneja la red al realizar la simulación (Fig. 5):

- Número de nodos en la red: indicará cuántos LSR existen en la red.
- Velocidad de envío de datagramas IP: indicará la velocidad en paquetes/sec a la que serán enviados los datagramas IP a la red.
- Retardo de procesamiento de los nodos: cifra genérica que representa el tiempo que un nodo tarda en realizar un chequeo en todos sus interfaces de entrada sumado al tiempo que emplee en procesar un paquete ATM y enviarlo.
- Tamaño de datos de los datagramas IP: indicará el tamaño de datos, sin incluir la cabecera, que tendrán los datagramas IP, de tal manera que si el tamaño de datos, junto con la cabecera supera los 53 bytes del tamaño de célula ATM, será necesario realizar segmentación al entrar estos datos en la red.
- Debug: indicará si se desean mostrar todos los datos, cuando el simulador entre en funcionamiento, o sólo mostrar el modo gráfico.
- Nombre del fichero de entrada: fichero utilizado para obtener los datos de la red (si se usa).

Datos específicos de cada nodo: Esta salida aparece cuando se desean visualizar los datos específicos de un LSR perteneciente a la red. Los datos que se muestran son (Fig. 6):

- Identificador: dirección específica del nodo único, y coincide con la dirección IP.
- Tabla de Routing: los datos de la tabla de routing del nodo.

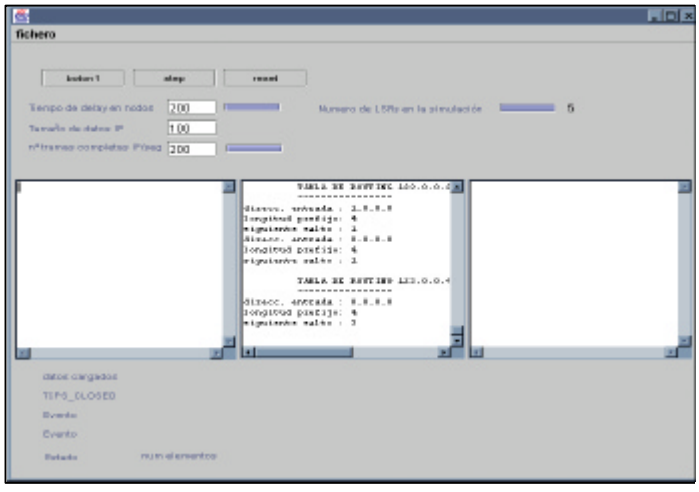


Fig. 5: Ventana de datos genéricos

Fig. 6: Datos específicos del nodo

- LIB: los datos de la tabla Label Information Base del nodo. Esta tabla se mostrará en el caso de que el nodo no sea interno (los nodos internos no la necesitan).
- NHLFE: los datos de la tabla NHLFE, que asocia las etiquetas con las salidas de los datos.
- Enlaces de entrada: mostrará los nodos de entrada a él junto con los interfaces de entrada asociados.
- Enlaces de salida: mostrará los nodos de salida a él junto con los interfaces de salida asociados.
- Frontera: indicará si es un nodo frontera de la red o interno. En el caso de los nodos frontera, estos podrán tener enlaces con entidades IP.

Disposición de la red: En esta salida se visualizan:

- Localización de los nodo.
- Enlaces entre los nodos: indica si se trata de un enlace de entrada o de salida (o mixto).
- Localización de las entidades IP.
- Enlaces nodos frontera con entidades IP.

Funcionamiento de la red: Los datos que se visualizan son (Fig. 7):

- Localización de los nodos.
- Enlaces entre los nodos: además deberá indicarse de alguna manera si se trata de un enlace de entrada o de salida (o mixto).

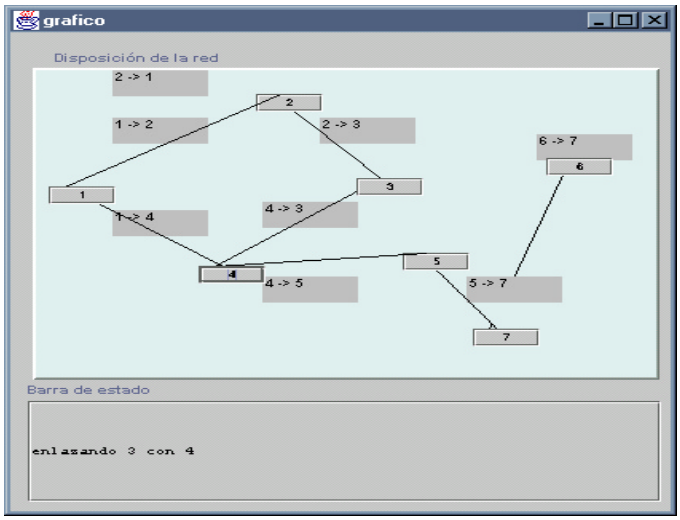


Fig. 7: Ventana de Simulación

- Localización de las entidades IP.
- Enlaces de los nodos frontera con las entidades IP.
- Paquetes enviados por los nodos.
- Paquetes recibidos por los nodos.
- Paquetes de control enviados.
- Paquetes de control recibidos.
- Operaciones que están realizando los nodos.

Como podemos observar, este conjunto de salidas nos proporcionan todas la respuestas necesarias con respecto a los parámetros de comportamiento de la red funcionando bajo MPLS.

4.3 Notas sobre el desarrollo en Java

La mayoría del desarrollo se ha realizado usando la herramienta *Symantec Visual Café Database Edition* bajo la *Java 2 SDK, Standard Edition, Version 1.2.2*.

Algunas de las clases que se emplearon para la codificación de la simulación fueron las siguientes:

- **Paquete:** clase genérica de la cual heredarán todos los tipos de paquetes empleados en la aplicación. Un paquete es una estructura del tipo matriz en la que se van insertando datos de tipo entero, short y string.
- **Paquete ATM:** hereda de *paquete*, con funciones y campos específicos de la célula ATM.
- **Paquete IP:** hereda de *paquete*, con funciones y campos específicos del datagrama IP.

- **General:** clase empleada para contener todas las constantes de la aplicación y todos los procedimientos útiles: conversión de datos, etc.

- **TCP:** representa a las entidades productoras o receptoras de paquetes TCP/IP.

- **MonitorBuffer:** estas clases son empleadas en todas las entidades que envían o reciben datos. Representan los buffers de entrada de datos y los enlaces entre unas entidades y otras. Los buffers almacenan paquetes genéricos, de tal manera que pueden instanciarse para tratar, tanto con datos IP como con datos ATM. La estructura es una *cola*, es decir, un sistema *FIFO.*

- **Sim:** es la clase principal que llama a los procedimientos de inicialización del resto de clases.

- **Grafico:** representa a la ventana de simulación de la red. Esta clase utiliza objetos gráficos, que representan a los nodos y a sus enlaces:

    - **DibujoNodo:** incorpora todos los elementos necesarios para representar a un nodo de la red y sus enlaces.

- **NodoMPLS:** es la clase más amplia, representa a un nodo de la red. Contiene las variables locales y los procedimientos necesarios para simular el comportamiento de un nodo. Dentro de ella, existen las siguientes clases:

    - **VectorPDUAAL5:** estructura de datos necesaria para almacenar las células ATM fragmentadas en el formato AAL5.

    - **TablaRouting:** representa a la tabla de routing del nodo.

    - **LIB:** representa a la tabla de Label Information Base del nodo, caso que sea un nodo frontera.

    - **TablaSwitching:** representa a la NHLFE. Next Hop Label Forwarding Entry del nodo interno.

    - **FEC:** representa a las Clases de Envío Equivalentes.

    - **TipoEtiqueta:** representa a las etiquetas que se asocian con los paquetes.

Conclusiones

En lo que respecta a la integración de IP sobre ATM, nunca podremos dar un 'sí' rotundo y sin objeciones a una solución. Lo que deseamos poner de manifiesto es que de todas éstas, la más interesante de cara al futuro es MPLS, y lo consideramos como tal, por su capacidad para asimilar características de las demás soluciones que ya han sido probadas. Por tanto, con vistas a determinar si MPLS puede llegar a convertirse en el estándar para la integración de IP-ATM, decidimos realizar esta experiencia de simulación. Disponemos, así de una herramienta de estudio pormenorizado (y a la vez didáctica) de su funcionamiento e intentar plantear posibles evoluciones de la tecnología que optimicen el proceso de integración de estos dos tipos de tráfico. Esta herramienta tiene un carácter totalmente abierto, y se está trabajando en su mejora y ampliación de capacidades.

Referencias

[1] "Classical IP and ARP over ATM ". M.Laubach, rfc 1577. Enero 1994

[2] "IP Switching – ATM Under IP". Peter Newman,Greg Minshall y Thomas L. Lyon IEEE ACM Transactions on networking, vol 6 nº 2. Abril 1998.

[3] "IP over ATM: a switch for the better? ". Andy Bray. Octubre 1.998. http://www.telecoms-mag.com/issues/199810/tci/bray.html

[4] "ATM switching and IP routing integration: the next stage in Internet Evolution? ". Paul Patrick University College London. IEEE. Abril 1998

[5]. "Flow Attribute Notification Protocol" RFC 2129. Abril 1997.

[6] "LAN Emulation over ATM 1.0" rtf 0021. ATM Forum. Enero 1995

[7] "MPOA Baseline". The ATM Forum. RFC 1932. Julio 1995

[8] "NBMA Next Hop Resolution Protocol (NHRP)" RFC 2332. J. Luciani, D. Katz, D.Piscitello, B.Cole, N.Doraswamy. Abril 1998

[9] "Tag Distribution Protocol". P Doolan et. Julio 1997

[10] "An Overview of IP Switching and Tag Switching ". Xipeng Xiao, Lionel M. Ni, Vibhavasu Vuppala. Department of Computer Science Michigan State University East. ICPADS'1997. 1997

[11] "Multi Protocol Label Switching Architecture". Eric C. Rosen (Cisco), Arun Viswanathan (Lucent), Ross Callon (IronBridge Networks). Agosto 1999

[12] "A framework for Multi Protocol Label Switching". Callon, Doolan, Feldman, Fredette, Swallow, Viswanathan. Julio 1999.

Metodología de Diseño para la Planificación de Redes Conmutadas en Entornos Locales.

Esteve Pallarès Segarra[1] y Joan García Haro[2]
[1]Departament d'Enginyeria Telemàtica, Universitat Politècnica de Catalunya.
C/ Jordi Girona 1 y 3, Campus Nord, Mod. C3, 08034 Barcelona.
[2]Departamento de Tecnologías de la Información y las Comunicaciones, Universidad Politécnica de Cartagena.
Campus Muralla de Mar s/n, 30202 Cartagena.
E-mail: esteve@mat.upc.es; joang.haro@upct.es

Abstract*. The deployment of new telematic services requiring high bandwidth entails that current telecommunication networks have to be prepared to support this traffic growth. In Local Area Networks the solution to these new bandwidth exigencies was the increase of link capacity as well as the segmentation in several subnets. Consequently, LANs that originally were shared-medium, shared-bandwidth ones have became switched networks. In this paper, we propose a heuristic methodology to design these new topologies. The traffic is modelled as a fluid flow generated by ON-OFF sources and the quality of service constraint is the information loss probability. The topology design parameters are the capacity of the links and the ports in each node.*

1 Introducción

La incorporación de nuevos servicios telemáticos que requieren mayor ancho de banda hace que las tecnologías de red utilizadas deban estar preparadas para soportar este aumento del tráfico.

En las redes de área extendida es necesario dimensionar adecuadamente las capacidades de los enlaces que conectan los nodos de conmutación. De esta manera es posible garantizar los parámetros de calidad de servicio que los usuarios precisan. En dichas redes la ubicación de los nodos de conmutación suele venir determinada por consideraciones geográficas, encontrándose éstos cerca de los núcleos de tráfico. El diseño de la topología de red, en este caso, se reduce a determinar la interconexión entre los nodos. Para ello pueden utilizarse algoritmos que garanticen la fiabilidad de la red, es decir, que la red sea capaz de seguir funcionando después de haberse producido algún fallo en algún enlace o nodo [1]. El principal parámetro de calidad de servicio para este tipo de redes es el retardo que pueda sufrir la información. Existen técnicas de asignación óptima de capacidades que permiten minimizar dicho retardo [1][2].

Por otro lado, en entornos locales, las nuevas exigencias de ancho de banda han conllevado un aumento de la capacidad de los enlaces y una mayor segmentación de dichas redes. De esta forma, las redes de área local que originalmente eran de medio compartido tienden a convertirse en redes conmutadas. Además, estas redes que cubren un área geográfica reducida, utilizan enlaces de alta velocidad de manera que el retardo de la información no suele ser un parámetro de calidad de servicio decisivo. Este tipo de redes suelen manejar grandes volúmenes de información, pudiéndose producir situaciones de congestión, siendo la probabilidad de pérdida de la información un parámetro de calidad de servicio a tener en consideración.

En este artículo se propone un procedimiento heurístico para el diseño de topologías de redes conmutadas en entornos locales. Para ello será necesario conocer y modelar el tráfico que fluirá por la red entre cada par de estaciones en función de los servicios que éstas deban soportar. Se ha considerado que la principal figura de mérito para el diseño de la red es acotar la probabilidad de pérdida de la información, aunque la metodología aquí propuesta es suficientemente genérica como para considerar otros parámetros de calidad de servicio.

El artículo se estructura de la siguiente manera. En la sección 2 se describe el modelo de tráfico utilizado y las expresiones utilizadas para el cálculo aproximado de la probabilidad de pérdida de la información en un nodo de conmutación. En la sección 3 se describe la metodología utilizada para el diseño de topologías. Dicha metodología consta de dos fases, primero se obtiene una topología inicial que es mejorada en una segunda fase mediante métodos heurísticos. En la cuarta sección se muestra un ejemplo de diseño y los resultados obtenidos al aplicar el algoritmo propuesto. Finalmente, en la última sección se incluyen las conclusiones más relevantes derivadas de este trabajo.

2 Modelo de tráfico

Existen múltiples modelos para caracterizar los distintos tipos de tráfico que se pueden encontrar en una red de telecomunicaciones [3]. Además, para nuestros propósitos será necesario utilizar un modelo que tenga en consideración la integración de los distintos servicios que deba soportar la red. Un modelo aceptado, para caracterizar el tráfico debido a la agregación de distintos servicios, es el de fluidos. En este caso, el concepto de paquete desaparece, de manera que el tráfico es considerado como un fluido continuo, consecuencia de la agregación de los fluidos generados por los distintos servicios soportados por la red. El hecho de no trabajar con paquetes permite independizar la caracterización del tráfico de los protocolos utilizados para la transmisión de la información. El fluido es generado por un conjunto de fuentes ON-OFF. El tiempo de permanencia en cada uno de los dos estados, ON y OFF, se supone distribuido exponencialmente con media t_{on} y t_{off} respectivamente. Solamente cuando la fuente esté en estado ON generará fluido con una tasa constante *r*. Los puertos de los conmutadores de la red se modelan mediante un buffer finito de tamaño *m*, en el cual se almacena el fluido generado por las fuentes antes de ser transmitido por el canal [4]. Dicho *buffer*, cuando contiene fluido, se vacía con una tasa constante *C*, la cual representa la velocidad de transmisión de los enlaces de la red. En la figura 1 se muestra dicho modelo.

Si el fluido es generado por *N* fuentes idénticas e independientes (caso homogéneo) se obtiene un proceso de Markov de *N*+1 estados. Para determinar la ocupación del *buffer* en este caso, es necesario resolver un sistema lineal de *N*+1 ecuaciones diferenciales de primer orden [5]. Ello supone calcular *N*+1 autovalores con sus correspondientes autovectores y además resolver un sistema lineal de *N*+1 ecuaciones para encontrar los coeficientes a_k que cumplen las condiciones de contorno del sistema. Existen expresiones cerradas para el cálculo de los autovalores y autovectores para el caso homogéneo [6], evitando así resolver la ecuación característica de la matriz del sistema de ecuaciones. También es posible calcular fácilmente los coeficientes a_k si el tamaño del buffer puede considerarse infinito.

Para el caso heterogéneo es necesario agrupar las fuentes ON-OFF en clases. Para un sistema de *R* clases con n_u fuentes iguales e independientes en cada clase se obtiene un proceso de Markov *R*-dimensional de *n* estados, siendo *n*

$$n = \prod_{u=1}^{R} (n_u + 1) \qquad (1)$$

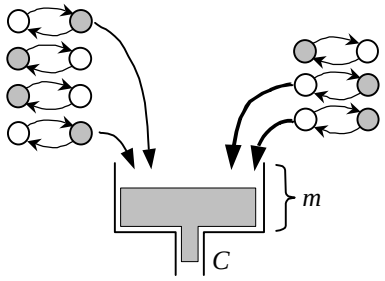


Fig. 1: Modelo de fluidos.

En este caso para calcular la ocupación del *buffer* será necesario resolver un sistema lineal de *n* ecuaciones diferenciales de primer orden [7]. Es posible calcular los autovectores y autovalores de la matriz del sistema de ecuaciones mediante técnicas de interpolación inversa [8], evitando así el cálculo de las *n* raíces del polinomio característico. El principal inconveniente en el caso heterogéneo es la rapidez con que aumenta la complejidad de la solución final a medida que aumenta el número de fuentes. Por ejemplo, para el caso homogéneo con 8 fuentes solamente hay que calcular 9 autovalores con sus correspondientes autovectores. En cambio para el caso heterogéneo con 4 clases y 2 fuentes de cada clase (8 fuentes en total) es necesario calcular 81 autovalores con sus correspondientes autovectores. Para evitar este problema suelen utilizarse expresiones aproximadas.

Si el objetivo es calcular la probabilidad de pérdida de información en un *buffer* finito se pueden utilizar las expresiones aproximadas desarrolladas en [9][10]. Dichas expresiones están desarrolladas para caso heterogéneo pero se pueden utilizar también para el caso homogéneo simplemente considerando que sólo existe una clase de fuente (*R*=1). Dependiendo del número de fuentes activas de cada clase, existen *n* posibles estados. Sea p_i la probabilidad de que el sistema esté en el estado *i* y B_i el flujo generado por las fuentes activas en dicho estado. Si ordenamos los estados en función del flujo generado (B_i), podemos clasificarlos en dos grupos. Los primeros *T* estados son los estados de infracarga (*underload*), para los cuales el flujo generado siempre es inferior a la capacidad del enlace (B_i<*C*). Cuando el sistema esté en uno de dichos estados no se producirán pérdidas de información. Los (*n*-*T*) estados restantes son estados de sobrecarga (*overload*) para los cuales se producirán pérdidas de información cuando la cantidad de fluido almacenado supere la capacidad del *buffer* (*m*). Suponiendo que la capacidad del enlace es superior al fluido medio generado por todas las fuentes (condición de estabilidad) e inferior al fluido generado por las fuentes cuando todas ellas están activas, la probabilidad de pérdida de información se puede aproximar por (2). Siendo z_d el autovalor negativo más cercano a cero de la matriz del sistema de ecuaciones diferenciales. Dicho autovalor, al cual denominamos autovalor

dominante, se puede calcular utilizando técnicas de interpolación en la función (3) [8].

$$prob.\ de\ pérdida \approx \frac{1-\frac{C}{\sum_{i=1}^{n} B_i p_i}}{1+\frac{\sum_{i=1}^{T}(B_i-C)p_i}{\sum_{i=T+1}^{n}(B_i-C)p_i}e^{-z_d m}} \quad (2)$$

$$C=\frac{1}{z_d}\sum_{u=1}^{R}\frac{n_u}{2}\left[z_d r_u + m_u + l_u - \sqrt{(z_d r_u + m_u - l_u)^2 + 4 l_u m_u}\right] \quad (3)$$

En la fórmula (3) se expresa la capacidad del enlace de salida como una función del autovalor dominante (z_d). Los parámetros l_u^{-1}, m_u^{-1} y r_u son el tiempo medio en el estado OFF, el tiempo medio en el estado ON y el fluido generado por una fuente de clase *u*. Dicha función es monótona decreciente, de manera que es posible interpolar el valor del autovalor dominante que da como resultado la capacidad del enlace.

Queda fuera del alcance de este trabajo el desarrollo completo de las aproximaciones aquí presentadas, pero el lector puede referirse a [9] y [10] para profundizar más en los detalles de las mismas.

3 Diseño de topologías

Para el diseño de las topologías de red se utiliza un algoritmo heurístico. Dicho algoritmo consta de dos fases. En una primera fase se genera una topología inicial la cual cumple los requisitos de conexión entre estaciones y de calidad de servicio, en este caso se pretende acotar la probabilidad de pérdida de la información. En una segunda fase se intenta diminuir el número de nodos y enlaces que forman la topología. Para ello se intentan combinar pares de nodos formando uno único, el cual seguirá cumpliendo con los criterios de conexión y de calidad de servicio de la red. Los enlaces de la red se suponen bidireccionales y todos iguales con la misma capacidad. Asimismo los conmutadores también se suponen todos iguales, con el mismo número de puertos y el mismo tamaño para todos sus *buffers*. Tanto el numero de puertos como las capacidades de los *buffers* y los enlaces son figuras de diseño del método. Otro parámetro de diseño es la máxima probabilidad de pérdida de la información admitida en la red.

Antes de realizar el diseño es necesario conocer y modelar los distintos servicios que debe soportar la red. Cada servicio es modelado como una o varias fuentes ON-OFF. La integración de los distintos servicios en cada enlace de la red se obtendrá simplemente agregando las fuentes que modelan dichos servicios. Así pues, para cada servicio será necesario determinar su tiempo de actividad (t_{on}), su tiempo de inactividad (t_{off}) y la intensidad de fluido generado (r), la cual estará expresada en unidades de transmisión de información, por ejemplo bits/segundo.

3.1 Topología inicial

Para el diseño de la topología inicial se toma cada una de las estaciones de la red y se construye una subtopología con estructura de árbol. Las extremidades de dicho árbol representan los puntos de conexión con el resto de subtopologías. En la figura 2 se muestra un ejemplo de subtopología, las formas cuadradas representan estaciones, las redondas nodos de conmutación y las trapezoidales puntos de conexión con otras subtopologías. En este ejemplo sencillo se supone una red formada por seis estaciones y los nodos de conmutación tienen tres puertos bidireccionales cada uno. Inicialmente se une la estación a un nodo de acceso (en el ejemplo de la figura 2 se une la estación 1 al nodo 7). El enlace que une ambos dispositivos deberá cumplir las restricciones de calidad de servicio. Utilizando la expresión (2), y teniendo en cuenta que la probabilidad de pérdida es una función decreciente con la capacidad, se puede interpolar el valor de capacidad mínimo para dicho enlace que cumple los requisitos de calidad de servicio. Dado que los enlaces son bidireccionales, dicho cálculo debe realizarse tanto para el tráfico ascendente, como para el tráfico descendente. Una vez realizado el cálculo para todas las estaciones, se considerará el peor caso, o sea el valor máximo de capacidad. Dicho valor será el valor mínimo de capacidad que deberán tener los enlaces de la red. Debido a la

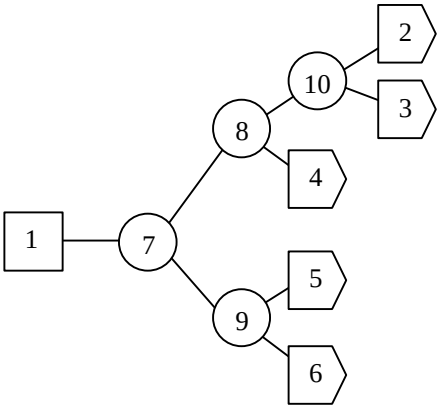


Fig. 2: Subtopología asociada a la estación 1.

propia estructura arbórea de las subtopologías, por el resto de enlaces de cada subtopología transitará un tráfico inferior al tráfico que circula por el enlace de acceso. Por lo tanto, si el enlace de acceso cumple las restricciones de calidad de servicio, el resto de enlaces de cada subtopología también lo cumplirá. La profundidad del árbol, definida como el número de nodos que hay entre la estación y el punto de conexión más alejado de la misma, dependerá del número de puntos de conexión que haya en cada subtopología. En general, dicha profundidad se puede calcular utilizando la expresión (4).

$$p = \left\lceil \frac{\ln(conexiones)}{\ln(puertos - 1)} \right\rceil \qquad (4)$$

Siendo *conexiones*, el número de puntos de conexión con las otras subtopologías y *puertos*, el número de puertos en cada nodo. El diseño de la subtopología se realiza de manera que sólo queden puertos libres en aquellos nodos más alejados de la estación. Ello facilitará la posterior combinación de estos nodos más alejados. El orden con que se ubican los puntos de conexión en los nodos de las extremidades es aleatorio, pudiendo obtener distintas topologías iniciales para diferentes ordenaciones.

Una vez generada la subtopología asociada a cada una de las estaciones, se procede a la construcción de la topología inicial. Para ello se unen los puntos de conexión de cada una de las subtopologías. En la figura 3 se muestra un ejemplo en el cual se interconectan 6 subtopologías formando la topología inicial de la figura 4. Esta topología inicial cumple con las restricciones de calidad de servicio impuestas a la red; en este caso que la probabilidad de pérdida de cada uno de los enlaces de la red esté acotada. Además también garantiza la existencia de un camino de comunicación entre aquellos pares de estaciones que lo precisen.

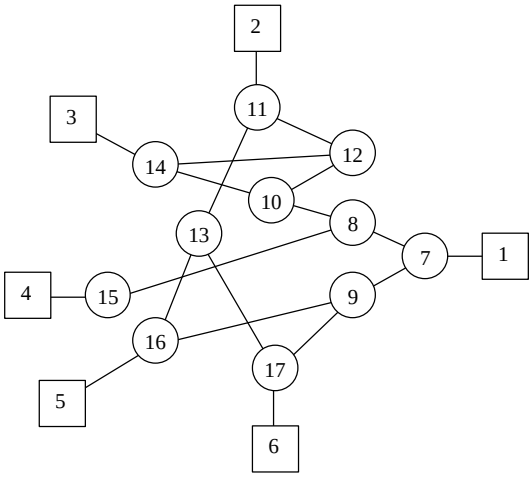


Fig. 4: Topología inicial.

3.2 Mejora de la topología

Como se observa en el ejemplo de la figura 4, a pesar de que la topología obtenida inicialmente cumple los requisitos de diseño, el número de nodos y enlaces obtenidos es muy elevado. Así pues, a continuación se procede a simplificar esta topología eliminando nodos y enlaces de manera que se sigan cumpliendo tanto los requisitos de conexión de la red, como los requisitos de calidad de servicio. Dicha simplificación se realiza combinando pares de nodos vecinos. Dos nodos son considerados vecinos cuando están conectados directamente (figura 5a), cuando están conectados a un nodo común (figura 5b) o bien, si se cumplen ambas condiciones (figura 5c). Al combinar dos nodos conectados directamente se produce un ahorro de un nodo y un enlace. La única condición que se debe cumplir para que dicha combinación sea posible es que el nuevo nodo disponga de suficientes puertos para mantener las conexiones ya establecidas (Figura 6a). Dicho de otra forma, que el número de conexiones del nuevo nodo con el resto de nodos de la red sea inferior o igual al

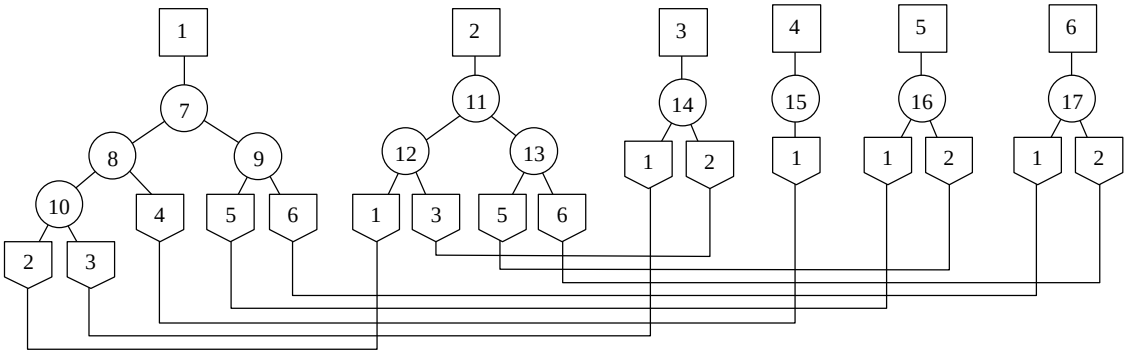


Fig. 3: Unión de subtopologías.

número de puertos de un nodo. Al combinar dos nodos conectados a un mismo nodo común, al cual llamaremos nodo padre, también se produce el ahorro de un nodo y un enlace (figura 6b). Si un par de nodos tiene más de un padre se produce el ahorro de un enlace por cada nodo padre (figura 6c). Además en este ejemplo también se produce el ahorro del enlace que une ambos nodos directamente. Para que dichas combinaciones sean posibles es necesario que el nuevo nodo combinado disponga de suficientes puertos para mantener las conexiones ya establecidas, pero además también deben cumplirse las condiciones de calidad de servicio en los enlaces que unen el nodo combinado con sus padres. Cada uno de estos enlaces es la unión de otros dos, por lo tanto, el tráfico que circula por cada uno de ellos es la agregación de los tráficos de los enlaces originales. En estos casos es necesario calcular la probabilidad de pérdida en el nuevo enlace y comprobar que se cumplan los criterios de calidad de la red. Si dichos criterios no se cumplen, no será posible unir los enlaces que se conectan al nodo padre. En este caso, si el número de puertos disponibles lo permite, se puede tener un doble enlace entre el nodo combinado y su padre. Aunque esta situación no se corresponda con una situación real, el algoritmo de combinación de nodos lo permite, ya que muy probablemente en el siguiente intento se puedan combinar los nodos unidos por el doble enlace, de forma que éste desaparecería.

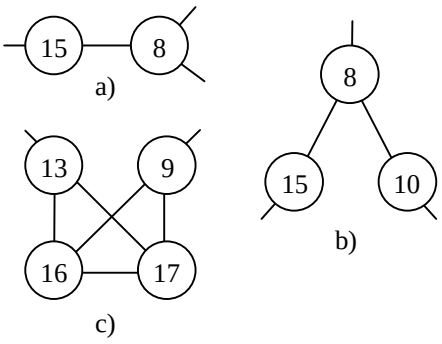


Fig. 5: Nodos vecinos.

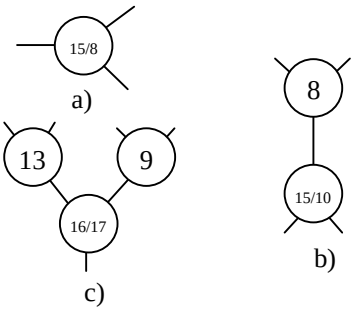


Fig. 6: Unión de nodos vecinos.

Tal como se ha mencionado anteriormente, el tráfico que circulará por un enlace obtenido a partir de la unión de otros dos, será la agregación de los tráficos de los enlaces originales. A la hora de calcular este nuevo tráfico hay que evitar que se formen bucles cerrados de flujos de información. Un ejemplo de ello se muestra en la figura 7. En este caso el tráfico generado por la estación 1 y destinado a la estación 6 sigue la ruta 1→2→3→4→5→6. Al combinar los nodos 2 y 4, si el nuevo tráfico se calcula simplemente por agregación de los anteriores, se produce una situación en la que un mismo flujo de información circula en ambos sentidos por el mismo enlace. Es evidente que la ruta que seguirá este tráfico será 1→2/4→5→6, evitando su paso por el nodo 3. Así pues, si se da una situación en la que después de la agregación de tráficos se obtiene un bucle cerrado, éste debe ser eliminado.

El algoritmo parte de una topología inicial, en este punto calcula para cada nodo cuáles son sus vecinos y las posibles combinaciones con ellos. Además calcula cuál es el número de enlaces ahorrados en cada caso, seleccionando aquellas combinaciones que ahorren el mayor número de enlaces. En el ahorro de cada enlace quedarán dos puertos libres, de manera que cuanto mayor sea el número de enlaces ahorrados, mayor será el número de puertos liberados. El hecho de liberar el máximo número de puertos facilitará que se puedan producir nuevas combinaciones de nodos en las siguientes iteraciones. Una vez calculados los pares de nodos que ahorran el mayor número de enlaces, se elige un par al azar y se combina. Las simplificaciones de la topología se repiten de forma iterativa hasta que no son posibles más combinaciones entre nodos vecinos. Llegado este punto, se guarda la topología obtenida y se vuelve a repetir el algoritmo para la misma topología inicial. Dado que el algoritmo debe elegir pares de nodos de manera aleatoria, cada vez se obtiene una solución distinta. La bondad de cada solución depende del número de nodos que forman la topología final, eligiendo aquellas soluciones para las cuales dicho número sea mínimo. En caso de empate, también se contempla el número de enlaces, eligiendo los casos en los cuales éste sea mínimo. El algoritmo finaliza cuando después de un número razonable de iteraciones la solución final no mejora.

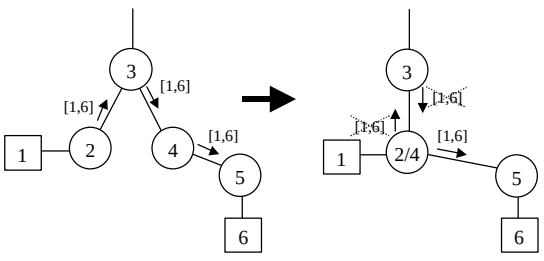


Fig. 7: Bucles cerrados de tráfico.

Debido a que la topología de partida no es única, depende de cómo se ordenan los puntos de conexión en las extremidades de las subtopologías, es posible generar una nueva topología inicial y reiniciar el algoritmo de combinación de nodos. Este proceso también se repite de forma iterativa hasta que después de un número razonable de intentos no se obtienen mejoras en la solución final.

4 Ejemplo de diseño

Para este ejemplo se han definido 4 clases de tráfico, las cuales se muestran en la tabla 1. Los valores de t_{on}, t_{off} y la intensidad de fluido r se han obtenido de la referencia [11]. Se supone una red con seis estaciones y los siguientes servicios:

- Un servicio unidireccional de transferencia de ficheros entre la estación 1 y el resto de estaciones.
- Un servicio unidireccional de fax de color entre todas las estaciones y la estación 1.
- Un servicio bidireccional de video-telefonía entre las estaciones 2 y 3.
- Un servicio bidireccional de voz entre las estaciones 2, 3, 4 y 5.

Suponiendo una probabilidad de pérdida de la información de 10^{-13} se obtiene que el caso más restrictivo de capacidad se produce en los enlaces de acceso de las estaciones 2 y 3. En ambos casos, la capacidad mínima debe valer 11,37 Mbps. Imponiendo que las capacidades de los enlaces sea de 12 Mbps. Y considerando nodos con cuatro puertos cada uno, se obtiene la topología inicial mostrada en la figura 8, con 6 estaciones, 11 nodos y 22 enlaces. Aplicando el algoritmo de combinación de nodos se obtiene la topología simplificada mostrada en la figura 9 la cual sólo está formada por dos nodos y siete enlaces.

En la figura se observa como las estaciones 2 y 3 comparten el mismo nodo de acceso, de manera que el tráfico de video-telefonía, el cual necesita más ancho de banda que el resto de servicios, sólo

Tabla 1. Clases de servicio.

Clase	Servicio	t_{off} (s)	t_{on} (s)	r (Mbps)
1	Voz	260	100	0.064
2	Video-telefonía	5000	100	10
3	fax de color	300	3	2
4	Transferencia de ficheros	333	1	2

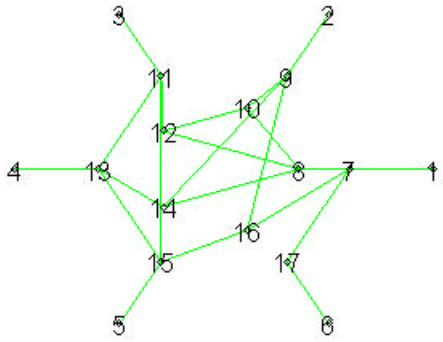


Fig. 8: Topología inicial

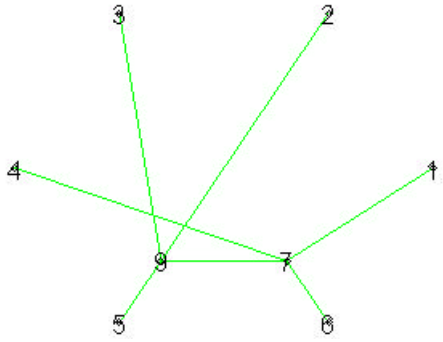


Fig. 9: Topología final

transita por los enlaces de acceso a dicho nodo. Ello evita que dicho servicio congestione otros enlaces de la red, así pues, la topología diseñada cumple con las restricciones de calidad de servicio impuestas a la red.

5 Conclusiones

En este artículo se ha presentado una metodología heurística para el diseño de topologías de redes conmutadas en entornos locales. En dicho diseño se contemplan los requisitos de calidad de servicio impuestos a la red, en este caso que la probabilidad de pérdida de información esté acotada. Los servicios soportados por la red son modelados mediante fuentes ON-OFF y para el tráfico se utiliza un modelo de fluidos. Ello permite independizar el diseño de la red de los protocolos utilizados en la transmisión de la información. Todos los nodos y puertos se suponen iguales. Los parámetros de diseño de la red son: la capacidad de los enlaces, el número de puertos de cada nodo, el tamaño de los *buffers* de los puertos y la máxima probabilidad de pérdida de información admitida en cada enlace. El método heurístico se basa en la mejora de una topología inicial que cumple los requisitos de conexión y de calidad de servicio de la red. Dicha mejora se consigue mediante la combinación de nodos vecinos con el consecuente ahorro de nodos y enlaces. Se eligen aquellas

combinaciones que ahorran el máximo número de enlaces. El algoritmo se repite de manera iterativa hasta que tras un número razonable de intentos no se consiguen mejoras en las topologías resultantes obtenidas. Finalmente, también se ha presentado un ejemplo de funcionamiento del algoritmo.

Agradecimientos

Este trabajo ha sido financiado por los proyectos de investigación SSADE (CICYT TEL99-0822), PRIME-IP (TIC2000-1734-C03-01) y FAR-IP (TIC2000-1734-C03-03).

Referencias

[1] E. Sanvicente, "Análisis y Diseño de Redes de Comunicación de Ordenadores", *NOVATICA*, vol. VII, num. 37, pp. 43-54, (1981).

[2] L. Kleinrock, "Queueing Systems Volume II: Computer Applications", John Wiley & Sons, (1976).

[3] I. Nikolaidis and I. Akyildiz, "Source Characterization and Statistical Multiplexing in ATM Networks", *College of Computing, Georgia Institute of Technology*, (1992).

[4] E. Pallarès, L.J. De la Cruz y Joan García, "Estudio de la Probabilidad de Pérdida en un nodo de Conmutación Mediante modelos de Fluidos", *Actas del XV Simposium Nacional de la Unión Científica Internacional de Radio*, pp. 407-408, (2000).

[5] R.C.F. Tucker. "Accurate Method for Analysis of a Packet-Speech Multiplexer with Limited Delay". *IEEE Transactions on Communications*, vol. 36, no. 4, pp. 479-483 (April 1988).

[6] D. Anick, D. Mitra and M.M. Sondhi, "Stochastic Theory of a Data-Handling System with Multiple Sources", *Bell Sys. Tech. J.*, pp. 1871-1894, (1982).

[7] C-K. Lim and J. Harms, "A Methodology for Designing Local Area Networks", *Proceedings of the ICC'95*, pp. 1925-1929, (1995).

[8] L. Kosten, "Stochastic Theory of a Data-Handling Systems with Groups of Multiple Sources".*Proc. Performance of Computer Communications Systems*, IFIP, pp. 321-331, (1984).

[9] E. Pallares and J. Garcia-Haro, "Mathematical Approach to Designing Switched LAN's. An Alternative Solution to Compute the Loss Probability in an Heterogeneous Traffic Environment", *Proceedings of the MELECON'2000*, vol. 1, pp. 11-14, (May 2000).

[10] E. Pallares and J. Garcia-Haro, "Fluid-Flow Approximation of the Information Loss Probability for a Switching System with Finite Buffering under Heterogeneous ON/OFF Input Traffic Sources", *Proc. QNETs 2000, Fourth International Workshop on Queuing Networks with Finite Capacity*, pp. 35/1-35/10, (July 2000).

[11] R. Onvural, "Asynchronous Transfer Mode Networks: Performance Issues", Second Edition, Artech House (1995).

Aplicación para la gestión electrónica de documentos XML

Pablo Hernández, Marta Cárdenes
Departamento de Ingeniería Telemática. Universidad de Las Palmas de Gran Canaria
Campus Universitario de Tafira, Pabellón C - 35017 - Las Palmas de Gran Canaria
Telf: 928-451250, Fax: 928-451243
E-mail: pablo@cma.ulpgc.es, marta@cocoon.ulpgc.es

Abstract. *Electronic document management enables you to manage and file documents efficiently via PC. Without leaving your desk, you can access any filed documents. Either on your corporate business or around the world: document access can be made available via the Internet, an Intranet or a LAN. In this paper we introduce an application for the electronic management of documents of an specific department of Binter Canarias Airlines that lays the foundations for a future implementation of a Documental Management System for the whole company. The solution combines the new 'language' XML that is being used in the Web for data exchange and the Java programming language as they complement each other.*

Keywords: *electronic document management, XML, Java, applet.*

1 Introducción

La gestión documental informatizada [1] tiene como objetivo proporcionar apoyo a la utilización de los documentos para cualquier tipo de proceso en cualquier entorno de una organización, entendiendo por documento cualquier objeto que contenga información en forma de datos, texto, audio o imagen (fija y animada).

Los procesos básicos que se siguen en un sistema de gestión documental (SGD) son:

- *Obtención de documentos*: con la finalidad de nutrir al sistema de contenidos de forma automática y masiva. Puede servirse de servidores de fax, correo electrónico, así como de aplicaciones software que permitan la creación de documentos dentro del propio sistema.

- *Análisis de documentos*: donde se identifica el tipo de documento (imagen, texto, voz, ...), para determinar el visualizador más idóneo. Otro tipo de análisis dentro de este proceso sería el control de versiones, comparación de documentos, generación de esquemas y resúmenes, etc. El resultado del proceso debe producir mejoras en su distribución y posteriores referencias.

- *Gestión de documentos*: donde se realizan los trámites que se hayan determinado durante el análisis (controlar los estados del documento, remitirlo a otras personas, etc.).

- *Almacenamiento de documentos*: con objeto de poderlo recuperar posteriormente.

El SGD se basa en una serie de módulos o subsistemas que realizan los procesos descritos anteriormente, y que al mismo tiempo dan la flexibilidad y potencia necesarias para hacerlo distribuido y escalable.

Los SGD comerciales aportan soluciones propietarias, son aplicables a entornos concretos, los métodos de búsqueda se basan generalmente en palabras clave, limitando su escalabilidad e integración con otros servicios (*business to business*) [2][3]. Últimamente ya están apareciendo los primeros productos software que soportan la tecnología XML (Extensible Markup Language) [4][5], aunque en realidad lo que presentan son soluciones propietarias basadas en XML.

En este artículo se presenta una aplicación realizada en la empresa de transporte aéreo Binter Canarias S.A. ubicada en Las Palmas de Gran Canaria, con el objeto de sentar las bases de una futura implantación de un SGD en la empresa. Concretamente se diseñó el módulo que gestiona las Circulares del Departamento de Dirección de Operaciones. Este documento recoge la normativa vigente para la Dirección de Operaciones de las compañías aéreas, el cual debe estar físicamente (impreso) en el libro de vuelo de cada uno de los tripulantes técnicos.

El gran volumen de información que poseen este tipo de organizaciones, ligado a la necesidad de contar con un acceso rápido y fácil a esta información fueron las razones que motivaron que Binter Canarias iniciara esta línea de trabajo.

2 Marco de trabajo

La primera tarea fue la elección del sistema más idóneo para almacenar la información que la empresa poseía. Se adoptó el empleo del estándar XML [6] por las siguientes razones:

- Permite definir la estructura lógica de almacenamiento de la información dentro de un archivo de texto, siendo su principal aplicación el manejo de datos estructurados.

- Es un lenguaje orientado a la descripción del contenido de un documento y no al aspecto como ha ter-

minado siendo el lenguaje HTML.

- Es un lenguaje de marcas flexible y extensible que permite personalizar las marcas, asociando a cada documento un DTD (Document Type Definition) donde se definen las marcas que se usarán para describir los contenidos de un documento.

- Cuenta con un sistema de vinculación tan especializado y exhaustivo que posee dos especificaciones XLL (Extensible Linking Language) [7] y Xpointer [8], que describen el sistema de vinculación con otros documentos. XLL permite describir enlaces unidireccionales como los utilizados con el lenguaje HTML, así como enlaces más sofisticados, escalables y flexibles. Xpointer permite especificar lugares precisos dentro de un documento XML (ambas especificaciones no forman parte de la especificación de XML).

- No sólo puede describir documentos, sino también metadatos, esto es, información que describe otra información, lo que facilita una localización más organizada de los recursos de la Red.

- Permite la realización de potentes búsquedas con capacidad para personalizarlas.

- Soporta dos lenguajes para la elaboración de hojas de estilo, el tradicional CSS (Cascading Style Sheets) [9] creado para HTML, y el XSL (Extensible StyleSheet Language) [10] diseñado específicamente para XML.

- Es independiente de la plataforma y de libre distribución.

El siguiente paso fue el desarrollo de una aplicación para la gestión electrónica de documentos en formato XML.

Hasta la realización e implantación de esta nueva aplicación, el proceso consistía en escribir el documento en formato Word e imprimirlo para ser entregado a cada uno de los destinatarios, puesto que éste debía estar físicamente en el libro de vuelo de cada uno de los tripulantes técnicos. Su principal inconveniente era la gestión de los mismos, no existiendo la posibilidad de: anular un documento al generar uno nuevo, buscar documentos según criterios de selección, acceso fácil y rápido al documento (era necesaria la presencia física del administrativo), etc.

Con la nueva aplicación el documento se crea electrónicamente accediendo a la intranet de la compañía previa comprobación del permiso de acceso. Además permitirá realizar modificaciones de los documentos y consultas de los mismos. El programa comprueba si el usuario está autorizado para acceder a los servicios que solicita. Se distinguen dos tipos de permisos de acceso: creación y edición de documentos (habilitado a unos pocos usuarios), y consulta de documentos, la cual queda abierta a los integrantes del Departamento de Dirección de Operaciones.

En la Fig. 1 se muestra el tipo de documento con el que se va a trabajar.

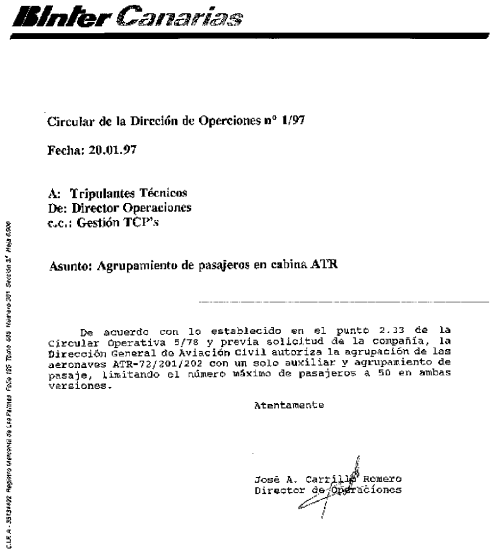
Java ha sido el lenguaje evaluado y aprobado por el departamento de Desarrollo de Sistemas de Binter Canarias para el desarrollo de programas propios y de la intranet corporativa, por lo que fue el lenguaje utilizado para la elaboración de esta aplicación. Java es un lenguaje independiente de la plataforma con el que se supera la dificultad existente en la empresa Binter Canarias motivada por la heterogeneidad de su red. Además, la sintonía, la documentación y el material existente de la pareja Java-XML potenciaron esta elección [11].

Otra de las justificaciones para el uso de Java es la posibilidad de conexión con bases de datos a través de los controladores JDBC (Java Database Connectivity), que ofrecen una API de programación de bases de datos para programas Java.

3 Componentes del sistema

El sistema diseñado para la gestión electrónica de documentos forma parte de la intranet de Binter Canarias con un modelo de programación Cliente/ Servidor.

El lado servidor lo constituye un equipo con sistema operativo *Linux* donde esta configurado el servidor *Apache Web Server*. La aplicación diseñada se encuentra ejecutándose en la misma máquina que el servidor Web, por la restricción que presentan los applets de Java de comunicarse con el servidor de donde fue requerido (el envío de los applets al cliente se realiza mediante su inclusión en páginas



Binter Canarias

Circular de la Dirección de Operciones nº 1/97

Fecha: 20.01.97

A: Tripulantes Técnicos
De: Director Operaciones
c.c.: Gestión TCP's

Asunto: Agrupamiento de pasajeros en cabina ATR

De acuerdo con lo establecido en el punto 2.13 de la Circular Operativa 5/78 y previa solicitud de la compañía, la Dirección General de Aviación Civil autoriza la agrupación de las aeronaves ATR-72/201/202 con un solo auxiliar y agrupamiento de pasaje, limitando el número máximo de pasajeros a 50 en ambas versiones.

Atentamente

José A. Carrillo Romero
Director de Operaciones

Figura 1: Circular de la Dirección de Operaciones

HTML). La aplicación se encuentra siempre escuchando, por un puerto TCP/IP determinado, las posibles peticiones de trabajo solicitadas por un cliente.

XML permite la creación de DTDs personalizadas para definir específicamente cierto tipo de información. En realidad la DTD es parte del propio documento aunque tienen una funcionalidad distinta, de manera que tanto la DTD como el documento XML pueden almacenarse bajo un mismo archivo (DTD incluida en el XML, DTD interna) o en dos archivos distintos físicamente (DTD externa única). Se escogió esta última solución por su mayor eficiencia (una sola modificación afecta a todos los documentos asociados a esta DTD, mientras que con DTD interna es necesario modificar documento por documento).

En el servidor donde reside la aplicación de gestión documental se encuentra almacenado el archivo DTD. Este DTD fue creado específicamente para las circulares de la Dirección de Operaciones. A continuación se presenta un extracto de la DTD.

```
<!-- edited with xml spy v2.5 -->
<!-- dtd del tipo circular -->
<!ELEMENT circular
(numerocircular,fechaemision,destinatario,remi-
tente,asunto,contenido,anular,saludo,usua-
rio,estado)>

<!ELEMENT numerocircular (#PCDATA)>
<!ELEMENT fechaemision (dia,mes,ano)>
<!ELEMENT dia (#PCDATA)>
<!ELEMENT mes (#PCDATA)>
<!ELEMENT ano (#PCDATA)>
<!ELEMENT destinatario (nombre_destinatario)>
<!ELEMENT nombre_destinatario (#PCDATA)>
     . . .
     . . .
<!ELEMENT saludo (#PCDATA)>
<!ELEMENT usuario (#PCDATA)>
<!ELEMENT estado (#PCDATA)>
```

En el lado cliente se encuentra un equipo informático equipado con un navegador, y que además debe tener instalado el *plug-in* para visualizar applets de Java con componentes *Swing*. El utilizar una interfaz web permite realizar las tareas de gestión de documentos desde múltiples plataformas hardware y diferentes sistemas operativos.

La comunicación de datos entre las entidades cliente y servidor se realiza dentro de la intranet de la empresa mediante sockets orientados a la conexión con el fin de aumentar la fiabilidad de la comunicación y garantizar la recepción en secuencia de los datos. El servidor es el encargado de atender y mantener las peticiones de conexión solicitadas por el cliente.

En la Fig. 2 se representa de forma gráfica la situación de los diversos componentes dentro del sistema completo.

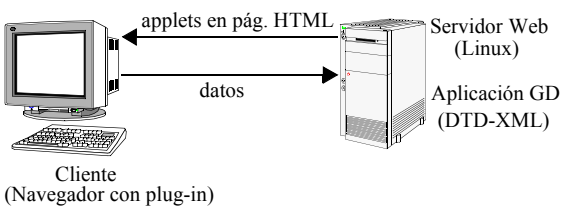


Figura 2: Componentes del sistema

4 Descripción de la aplicación

El empleo de interfaces web para la gestión de los documentos permite el acceso a la aplicación de gestión documental desde cualquier plataforma o entorno que disponga de un navegador. Estas interfaces que se presentan al usuario son applets de Java formados por componentes *Swing* proporcionados por el JDK 1.2 (Java Development Kit) e incrustados dentro de páginas web.

Para iniciar la aplicación es necesario acceder a la página principal de la intranet de la empresa, y dentro de ella, seleccionar el enlace destinado a la Gestión Electrónica de Documentos. A continuación, aparece el formulario mostrado en la Fig. 3, donde se puede seleccionar el tipo de documento y la tarea a realizar.

Las tareas habilitadas para la gestión de documentos son:

- *Creación*: con los datos introducidos por el usuario se generará un fichero escrito en XML, partiendo del fichero DTD asociado.

- *Consulta*: presenta una lista de aquellos documentos que cumplan los criterios de selección indicados por el usuario.

- *Edición*: edita un documento para su modificación.

A continuación se describirán cada una de las tareas citadas.

Figura 3: Página principal de la aplicación

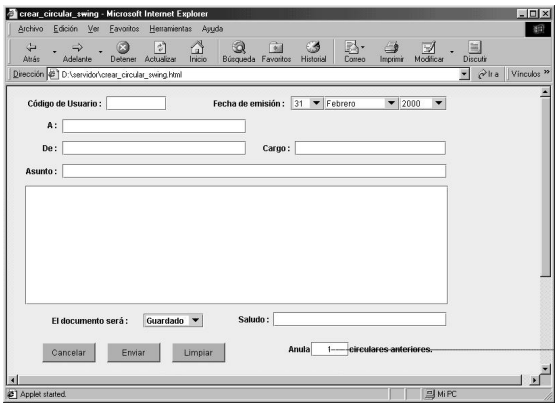


Figura 4: Interfaz para la creación de documentos

4.1 Creación de documentos

Al seleccionar esta tarea desde la página principal, el servidor Web envía al cliente un applet Java incluido en la página HTML que se muestra en la Fig. 4. Esta página servirá de interfaz para la creación de documentos, donde el usuario rellenará los campos con los datos que forman el documento.

La interfaz está formada por campos que mantienen una estrecha relación con los elementos definidos en el fichero DTD, pudiendo hacer corresponder cada dato con su elemento asociado en la DTD en el momento de generar el documento XML.

El applet enviado al cliente soporta el uso de estilos propios de escritura (negrita, cursiva, etc.) en la redacción del contenido del documento.

Antes de enviar los datos al servidor, el propio applet realiza una validación de los mismos para comprobar cualquier posible error y la existencia de datos en los campos de obligada cumplimentación. En caso de detectar cualquier anomalía el envío es cancelado y se informa al usuario del error detectado.

Una vez validados los datos, estos se envían al servidor junto con la indicación de la tarea a realizar y el tipo de documento que se pretende generar (para localizar la DTD correspondiente) a través de un socket que abre el propio applet.

La aplicación de gestión documental identifica la tarea a realizar en la cadena de datos recibida del cliente, que en este caso será la de creación de documento de tipo Circular. La aplicación busca el fichero DTD asociado a este tipo de documento y construye el archivo XML vacío correspondiente al DTD hallado. Con esta intención se analizan cada una de las declaraciones contenidas en el DTD. Éstas pueden incluir tanto la definición de un contenido de datos como el anidamiento de otros elementos, en cuyo caso se analizan las declaraciones de estos últimos hasta llegar a un elemento con un contenido de datos. El proceso se indica en la Fig. 5, donde # representa la definición de un contenido de datos y A, B, ... son elementos de la DTD.

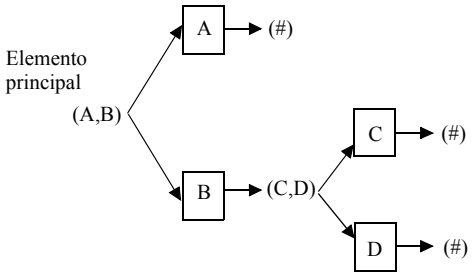


Figura 5: Recorrido del DTD para la construcción del archivo XML vacío

A modo de ejemplo se muestra a continuación el fichero XML vacío correspondiente al DTD hallado.

```
<circular>
   <numerocircular></numerocircular>
   <fechaemision>
     <dia></dia>
     <mes></mes>
     <ano></ano>
   </fechaemision>
     . . .
     . . .
   <anular></anular>
   <saludo></saludo>
   <usuario></usuario>
   <estado></estado>
</circular>
```

A continuación, la aplicación inserta los datos introducidos por el usuario dentro de los pares de etiquetas correspondientes del fichero XML. En el caso de que el documento anule alguno anterior, se deberá buscar el documento XML que se pretende anular para modificar su estado (`<estado>` anulado`</estado>`) e indicar el documento que originó la anulación (contenido de las etiquetas `<anular>` `</anular>`).

El hecho de contemplar como independientes la tarea de construcción del archivo XML vacío y la tarea de inserción de los datos dentro del fichero construido posteriormente, permite que la aplicación de gestión documental no esté ligada a un DTD determinado, de manera que sea capaz de generar ficheros XML bajo cualquier DTD definido.

El proceso siguiente es construir un fichero HTML con los contenidos del documento para su posterior visualización en el navegador del cliente. Esta no es la mejor forma para presentar el documento final, pero fue adoptada por no existir una estandarización de las hojas de estilo XSL en los navegadores actuales.

Para concluir el proceso, la aplicación confirma la creación del documento al cliente mediante una página HTML indicando el código asignado al docu-

mento generado. Esta página será presentada por el propio applet en el navegador del cliente, gracias a la capacidad que tienen los applets de devolver una página web como respuesta a un evento acontecido en el mismo (presionar el botón *Enviar*), evitando crear hiperenlaces a otras direcciones. El código del documento constituye un vínculo al documento que presentará el aspecto mostrado en la Fig. 6.

El proceso descrito en esta sección se muestra de forma gráfica en la Fig. 7, donde se pueden identificar los bloques que componen esta tarea.

4.2 Consulta de documentos

Esta tarea permite realizar consultas de documentos con criterios específicos de selección. Los datos introducidos por el usuario como criterios de selección pueden corresponder a algún dato almacenado en el documento o el código de documento.

La interfaz utilizada para esta tarea está formada por campos que mantienen una estrecha relación con los elementos definidos en el fichero DTD y, por tanto, con las etiquetas utilizadas en los ficheros XML correspondientes. Al tener el lenguaje XML la facilidad de organizar datos, es en esta tarea donde se observan las ventajas que proporciona este lenguaje en cuanto a rapidez y delimitación de la búsqueda.

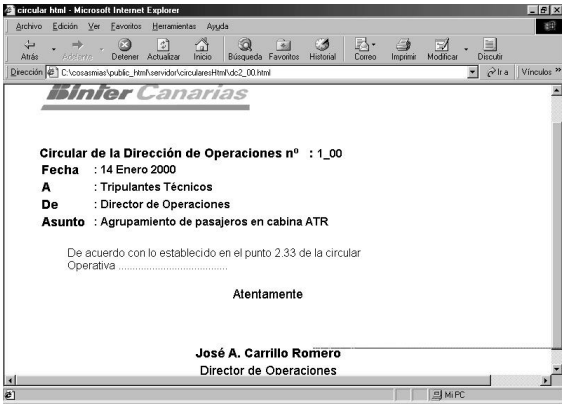


Figura 6: Presentación final del documento

De igual forma que con la tarea de *Creación de documentos*, el applet correspondiente verifica los criterios de selección solicitados por el usuario, para posteriormente enviar la solicitud al servidor. Éste localiza los documentos y presenta en una página HTML una lista con todos los documentos encontrados para acceder a ellos directamente a través de un hiperenlace.

El proceso descrito se muestra de forma gráfica en la Fig. 8, donde se pueden identificar los bloques que componen esta tarea.

4.3 Edición de documentos

Cualquier documento creado puede ser modificado mediante la edición del mismo. Con este fin, el usuario debe proporcionar el código del documento que se desea editar.

Debido a que el lenguaje Java no es capaz de recuperar directamente las variables de entorno de una página web, esta labor se ha realizado mediante un Servlet Java con el objeto de remitir posteriormente el documento a editar al cliente.

Una vez el servidor ha localizado el documento solicitado, se envía un applet dentro de una página HTML que muestra el contenido del documento para su modificación. En la interfaz existen algunos campos que no pueden ser modificados como es el caso del `código de usuario` , de forma que siempre va a quedar identificado el creador original del documento. También está deshabilitada la posibilidad de modificar los códigos de los documentos que perdieron su vigencia con la creación del documento editado en el momento de su generación.

Cuando el usuario ha rellenado el formulario

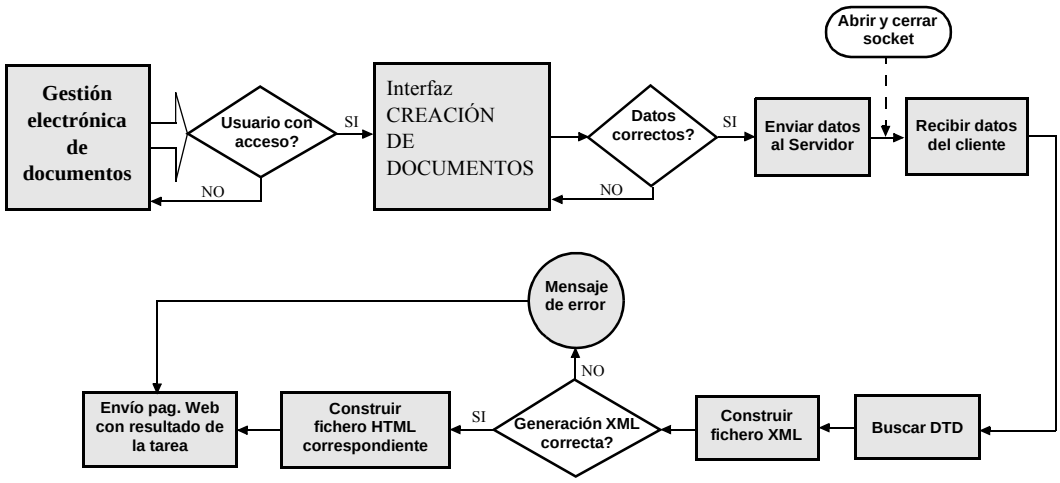


Figura 7: Proceso de creación de documentos

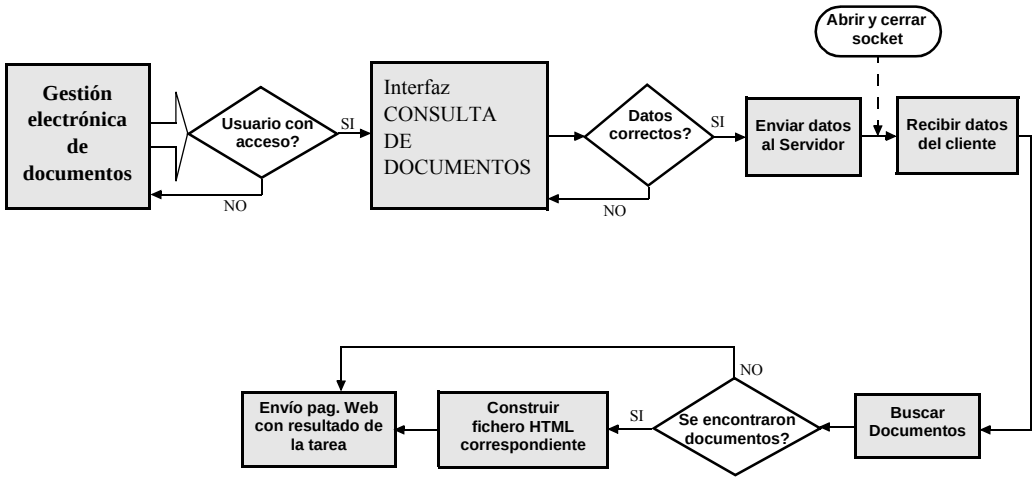


Figura 8: Proceso de consulta de documentos

comienza el proceso para la generación del documento modificado sobrescribiendo el fichero XML inicial con los nuevos datos introducidos por el usuario, de forma similar a la descrita para la tarea de *Creación de documentos*.

5 Conclusiones y líneas futuras

Se ha presentado una aplicación para la gestión electrónica de documentos que aunque se ha particularizado a un tipo de documento, un departamento y una serie de tareas, ha servido para sentar las bases para una futura ampliación de la aplicación que abarque otras tareas y otro tipo de documentos.

Su aplicación a otros documentos conlleva necesariamente la construcción del DTD correspondiente y las interfaces de usuarios necesarias, pero no es necesario modificar la aplicación de gestión documental.

Algunas de las soluciones adoptadas han venido determinadas por requisitos exigidos por la empresa Binter Canarias, por lo que su idoneidad debe evaluarse en el contexto donde se han aplicado.

Desde el punto de vista de los usuarios es importante destacar la buena aceptación que ha tenido entre los equipos de tripulantes técnicos debido a la facilidad de uso de la aplicación, a la rapidez de acceso a los documentos y al sistema de organización de documentos implementado.

Como líneas de trabajo futuro hay que resaltar:

- Actualización de la aplicación, adaptándola a las especificaciones que finalmente se aprueben de los estándares adoptados en esta aplicación.

- Almacenamiento de los datos de los documentos generados en una base de datos. Esta tarea es relativamente sencilla con el empleo del estándar XML, gracias a que se puede presumir una correspondencia entre etiquetas del documento XML y campos de las tablas de la base de datos. Esta tarea, que en principio se contempló, quedó suspendida hasta que el departamento encargado de la configuración de los sistemas informáticos decidiera cuál iba a ser el motor de la base de datos para la intranet corporativa: DB2 [12] u ORACLE [13].

  En cualquier caso, la aplicación de Gestión Documental generaría sentencias SQL (Structured Query Language) que serían trasladadas al DB2 u ORACLE mediante los JDBC correspondientes. La aplicación no generaría ficheros planos XML, sino que almacenaría su contenido en la base de datos. La generación del documento XML se realizaría cuando el cliente así lo solicitara.

Referencias

[1] M. Siminiani. “Intranets, empresa y gestión documental: Cómo enfocar en la práctica la tecnología desde la necesidad de eficiencia en todo tipo de empresas”. Ed. Mc Graw-Hill, Madrid, 1997. ISBN: 8448110625.

[2] DocuWare. http://www.docuware.com/uk.

[3] DocuTrack. http://www.docu-track.com.

[4] Cardiff Software. http://www.cardiff.com/cardiff.

[5] IcomXpress. http://www.icomxpress.com.

[6] XML version 1.0. http://www.w3.org/TR/REC-xml.

[7] XLink version 1.0. http://www.w3.org/TR/2000/PR-xlink-20011220.

[8] XPointer version 1.0. http://www.w3.org/TR/2000/PR-xlink-20011220.

[9] CSS3. http://www.w3.org/TR/2001/WD-CSS3-roadmap-20010119. Las especificaciones anteriores CCS1 y CSS2 también pueden localizarse en esta dirección.

[10] XSL version 1.0. http://www.w3.org/TR/2000/CR-xsl-20001121.

[11] Java y XML: una buena pareja. InformationWeek. nº 23.

[12] DB2 XML Extender. http://www-4.ibm.com/software/data/db2/extenders/xmlext.

[13] ORACLE9i New Features Summary April 2001. http://technet.oracle.com/products/oracle9i/pdf/9i_new_features.pdf.

Mediador Inteligente para Comercio Electrónico en Entornos Móviles.[1]

Diego Ponce, Miguel Soriano
Departament d'Enginyeria Telemàtica. Universitat Politècnica de Catalunya.
Jordi Girona 1 y 3. Campus Nord, Mód C3, UPC. 08034 Barcelona
Teléfono: 934 015 982 Fax: 934 015 981
E-mail: {dponce, soriano}@mat.upc.es

Abstract. *The great acceptance of mobile telephony in the world market presents an excellent opportunity for electronic commerce. The characteristics of mobile devices with capabilities of WAP (Wireless Application Protocol) and intrinsic properties of wireless environments impose a set of difficulties for mobile commerce development presented in this article. At the same time an intelligent broker between mobile clients and servers on Internet is proposed to profit standards and existing technologies to fulfill client needs, manage and personalize client information.*

Keywords: *WAP, m-commerce, intelligent agents.*

1 Introducción

La gestión de la empresa en la nueva economía se construye en red, internamente y en relación con el mercado y los proveedores. El concepto que en los años 80 fue conocido como "nuevas tecnologías" ha acabado originando una "nueva economía". A finales de los 90 apareció el concepto de "e-business" como la forma de entender la adaptación de la empresa al mundo digital.

El comercio electrónico (e-commerce) se puede definir, en un sentido amplio, como cualquier forma de transacción financiera o intercambio de información comercial basada en la transmisión de datos sobre redes de comunicación. Analistas económicos señalan que el éxito del comercio electrónico será proporcional al valor que agregue a la cadena de suministro del producto. La utilización de las nuevas tecnologías facilita al proveedor brindar el servicio de postventa y obtener una realimentación útil para el seguimiento del producto a partir de información suministrada por el cliente. Actualmente existen alternativas tecnológicas para los procesos mercantiles en aspectos tan importantes como procesos publicitarios, estudio y segmentación de mercado, negociación, mecanismos de pago, servicio al cliente, ... El éxito de su implantación masiva, sin embargo, debe aún superar barreras culturales, mejorar la facilidad de uso y garantizar la confianza de los usuarios. Una de las novedades del comercio electrónico es la posibilidad de atraer a los clientes que se encuentran en la vecindad de un centro de negocios y/o servicios aportándoles la información adecuada.

En este artículo se presenta en primer lugar una serie de aspectos a tener en cuenta en el diseño de un esquema de comercio electrónico móvil (m-commerce). En la sección 3 se propone la utilización de un intermediario basado en inteligencia artificial, y en el siguiente apartado se presentan sus prestaciones y limitaciones. Finalmente, se enumeran las principales conclusiones del trabajo presentado.

2 El m-commerce

El m-commerce involucra tres aspectos básicos; i) oferta de los negocios y de servicios en un área circundante al usuario, ii) información oportuna georeferenciada mientras el usuario está en movimiento, y iii) posibilidad de completar la transacción en forma inmediata. Por ello, debe ofrecer al usuario las siguientes prestaciones: a) negociación y entrega inmediata, b) métodos de micro y macro pagos, y c) facilidades de uso en este contexto móvil [6].

Existen, sin embargo, una serie de factores que dificultan la implantación y desarrollo del m-commerce frente al e-commerce. Estos inconvenientes están relacionados con las características del entorno inalámbrico (habitualmente menor ancho de banda, mayor latencia, conexiones menos estables y disponibilidad menos predecible) y limitaciones de los teléfonos móviles (procesadores menos potentes, menor memoria, limitaciones en el consumo de potencia, dimensiones de las pantallas, ...).

La figura 1 muestra el entorno típico de WAP, normalmente el equipo móvil se conecta a Internet a través de una pasarela (WAP proxy) que realiza la traducción entre los protocolos de Internet y WAP.

En el marco del comercio electrónico, existen propuestas interesantes que integran las tecnologías emergentes en una infraestructura de tal naturaleza que se redefine la manera de hacer negocios y acceder a la información en línea. Se pretende implantar una verdadera infraestructura conceptualmente superior a lo que actualmente disponemos en la Web, conocida como "The Grid" [10].

[1] Este trabajo ha sido desarrollado dentro del proyecto ACIMUT CICYT TIC2000-1120-C03-03.

El mercado de las comunicaciones a móviles puede utilizarse como la extensión del B2B (business to business) para entornos corporativos móviles con clientes internos. En el entorno del consumidor final la situación es del B2C (business to consumer), el acceso libre y sin limitaciones a los contenidos existentes en el Internet. Una manera de comunicar las plataformas de tecnología B2B con las B2C puede ser mediante la utilización de intermediarios que gestionen y faciliten la búsqueda de información al usuario. La negociación siempre requerirá protección de los datos.

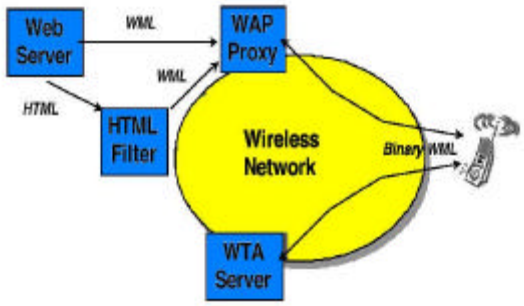


Figura 1 El entorno típico de WAP.

En el comercio electrónico es de vital importancia dar al usuario varias alternativas y modalidades de negociación, que permitan a cada usuario o empresa adaptar el comercio electrónico a sus propias necesidades personales o empresariales de forma segura. Existen el estándar TLS (Transport Layer Security) para proveer seguridad a nivel de la capa de transporte. TLS [4] es el estándar de Internet de SSL (Secure Socket Layer). A nivel de red privada virtual se dispone de IP-Sec [24] y Wireless Virtual Private Network.

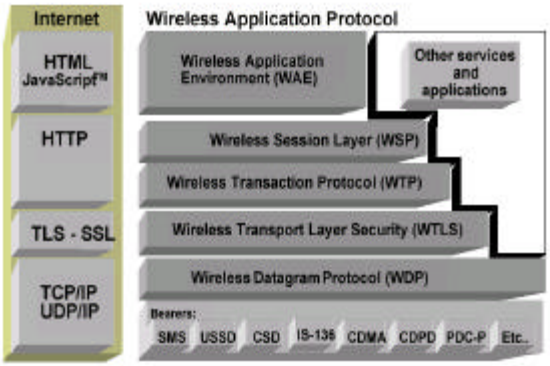


Figura 2 Pilas de protocolos Internet y WAP

En el m-commerce tanto el proveedor como el consumidor se conectan indirectamente a través de entidades de software e Internet, de modo que se deben establecer relaciones de confianza entre las partes y garantizar autentificación, confidencialidad e integridad [7]:

La autentificación de los mensajes se alcanza cuando el usuario firma un mensaje con su clave privada y el receptor la descifra con la clave pública. Con la finalidad de garantizar la autenticidad de las claves se requiere de una infraestructura de clave pública, con elementos tales como las autoridades de certificación y de registro. Las autoridades de certificación emiten un certificado digital que permite la autentificación de las partes y de los mensajes que intercambian. Desafortunadamente aún no se implanta una infraestructura de alcance global y por este motivo los comerciantes han optado por otras soluciones.

La figura 2 presenta las pilas de protocolos de Internet y WAP. La seguridad a nivel de la capa de transporte en Internet la provee TLS. En los entornos móviles WAP la seguridad la proveen diferentes capas: en la capa WTLS (Wireless Transport Layer Security), en la capa de aplicación el módulo de identidad WIM (Wireless Identity Module) y en la portadora, por ejemplo: en GSM se dispone del algoritmo de seguridad A5 [1].

La figura 3 representa el modelo de seguridad en el entorno WAP. Se puede observar que los canales seguros en entornos inalámbricos se establecen entre la pasarela WAP y el cliente WAP mediante WTLS, mientras la seguridad entre el servidor en Internet y la pasarela WAP la provee TLS. La traducción entre WTLS y TLS se ejecuta en la pasarela WAP. Para garantizar la privacidad e integridad de los datos en la pasarela es preciso:

1. Asegurar que la pasarela nunca almacena contenido en claro en un medio secundario.
2. Utilizar procesos de cifrado y descifrado veloces y seguros con borrado de contenidos en claro de la memoria volátil interna de la pasarela WAP inmediatamente después de la transacción.
3. Asegurar fisicamente el acceso a la consola de la pasarela WAP solamente a administradores autorizados.
4. Aplicar todos los mecanismos de seguridad para proteger los sistemas de facturación y el registro de localización de la pasarela WAP.

Sin embargo, la seguridad en los entornos WAP aún presenta las siguientes debilidades [5]:

- No provee autenticación extremo a extremo.
- Los contenidos quedan en claro en la pasarela WAP.

En [20] se presenta una alternativa en la arquitectura WAP que solventa los problemas mencionados anteriormente. Actualmente en los entornos de Internet se dispone adicionalmente de protocolos de seguridad a nivel de la capa de IP, tales mecanismos se agrupan bajo la denominación IP-Sec, y permiten la implantación de redes privadas virtuales dentro del Internet. Análogamente se dispone de las WVPN (Wireless Virtual Private Networks) en entornos inalámbricos.

3 El intermediario

La figura 4 presenta el esquema general de los componentes del Intermediario (Broker) que gestiona y facilita el intercambio de contenidos entre cliente y servidor. El intermediario realiza búsquedas de información utilizando agentes inteligentes, filtra los contenidos mediante el motor de inferencia, gestiona la relación con el cliente utilizando herramientas e-CRM (electronic customer relationship management), y mantiene una autoridad de registro con los certificados de los clientes. Una vez procesada la información se la prepara en el formato de entrega adecuado.

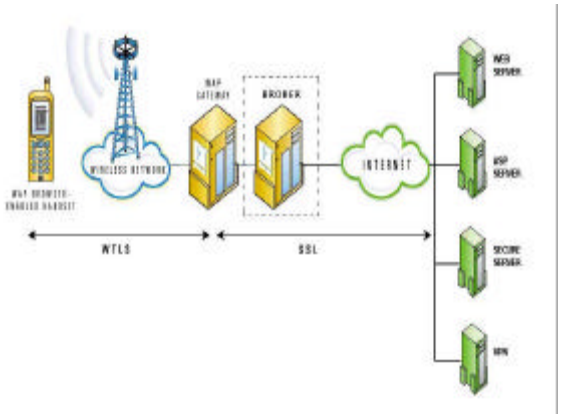


Figura 3 El modelo de seguridad WAP con el Intermediario.

A grandes rasgos podríamos considerar dos modos de funcionamiento. En primer lugar, cuando un usuario solicita una determinada información, se realiza un procesamiento de la solicitud que comienza con la búsqueda de la información que luego será clasificada, organizada, filtrada y entregada al terminal WAP a través del intermediario, tomando las medidas de seguridad necesarias. La figura 5 presenta el diagrama de flujo de una solicitud realizada por el cliente WAP. En segundo lugar, la gestión y entrega automática de contenidos utilizando la información personalizada del perfil del usuario. La figura 6 presenta brevemente el esquema la distribución de contenidos para clientes WAP en modalidad Push.

El canal inalámbrico entre el usuario y la pasarela WAP constituye el "cuello de botella" en cuanto al ancho de banda, por lo tanto el Intermediario deberá ubicarse lo más próximo a la pasarela WAP. Para gestionar la información existen esquemas que han tenido éxito en el manejo y replicación de los datos contenidos en sistemas distribuidos Web mediante el uso de caches y proxies [12]. Esquemas similares pueden ser útiles para gestionar contenidos procedentes de WWW, aunque no se han obtenido tan buenos resultados al utilizar estos mecanismos con contenidos dinámicos (D-HTML) como con contenidos estáticos.

El e-marketing posibilita segmentar el mercado y personalizar los servicios con la información suministrada por los mismos usuarios. Actualmente está muy ligada a las ventas y se pone especial cuidado en la fidelización y el servicio personalizado al cliente, se habla de la gestión de la relación del cliente (CRM), basándose en un perfil del usuario, su actualización dinámica y el análisis de sus necesidades. La integración de tecnología e-CRM posibilita extender las capacidades del equipo móvil, y facilita su utilización, la gestión se realiza en forma predictiva y dinámica en lugar de bajo pedido y explota la facilidad Push del protocolo WAP para la entrega contenidos de posible utilidad para el destinatario en el formato adecuado, por ejemplo: mediante SMS. Utilizar Push implica una menor utilización del canal inalámbrico que los mediante peticiones del usuario, que se gestionan en modalidad Pull (viaje de ida y vuelta).

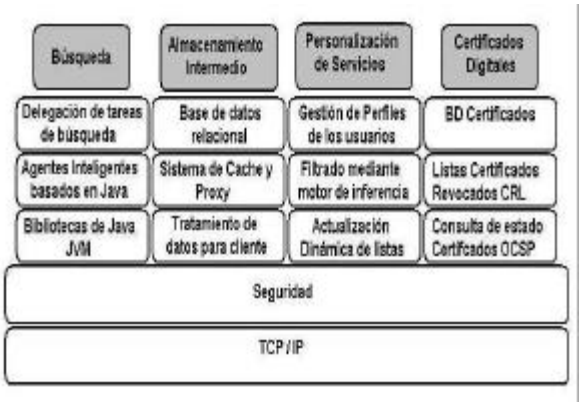


Figura 4 Modelo general del Intermediario

La gestión de la relación con el cliente desde el punto de vista de integración del sistema CRM en el Intermediario requiere una estrategia para el arranque, crecimiento y establecimiento del sistema. Se debe garantizar el cumplimiento de los siguientes requisitos:

- Disponer de las capacidades de inteligencia y análisis para capturar y convertir los datos del cliente en información útil.
- Poder procesar la información proveniente de diferentes fuentes de interacción del cliente por ejemplo: e.mail, web, sms….
- Manejar las transacciones desde el Web de la compañía.
- Proveer un repositorio centralizado de la información de los clientes.
- Integrar en su flujo de trabajo toda la información de manera que esté disponible en cada paso del proceso.
- Funcionar con su sistema operativo y con otras aplicaciones, ser transportable.

La entrega de información en modalidad Push, requiere la generación de listas de distribución. La lista de distribución se forma a partir del perfil del cliente. Los contenidos más recientes del almacén intermedio se entregan a cada lista mediante un

proceso por lotes. Un cliente puede pertenecer a varias listas.

Se necesita utilizar componentes de software especializados para mejorar la eficacia de la búsqueda de información. El intermediario utiliza agentes inteligentes, mediante una infraestructura orientada a objetos basada en Java que posibilita el trabajo cooperativo de múltiples agentes para cumplir con la búsqueda encomendada [16]. En la actualidad se investigan los mecanismos para garantizar el intercambio tanto de información como de código móvil en forma segura en un entorno distribuido de intermediarios.

Un sistema intermediario con agentes inteligentes puede realizar tareas tales como: buscar, aconsejar, contactar, comparar, filtrar, facilitar…. Se utiliza un entorno multiagente para la búsqueda de información, y un sistema experto basado en reglas y lógica difusa, para manejar la personalización del cliente WAP. La información de los servidores se almacena y organiza en mecanismos de bases de datos distribuidas que actualizan su contenido en forma dinámica.

Los agentes inteligentes en el comercio electrónico se utilizan para buscar contenidos en Internet y facilitar al usuario la interacción desde su equipo móvil. Sin embargo, esta tecnología requiere del entorno lo siguiente: accesibilidad, determinismo, entornos predecibles estáticos o dinámicos, y reglas de acceso. Desafortunadamente Internet no es, un entorno suficientemente amigable para los agentes inteligentes [23].

Una vez que el intermediario ha realizado el contacto entre el cliente y el servidor es posible que

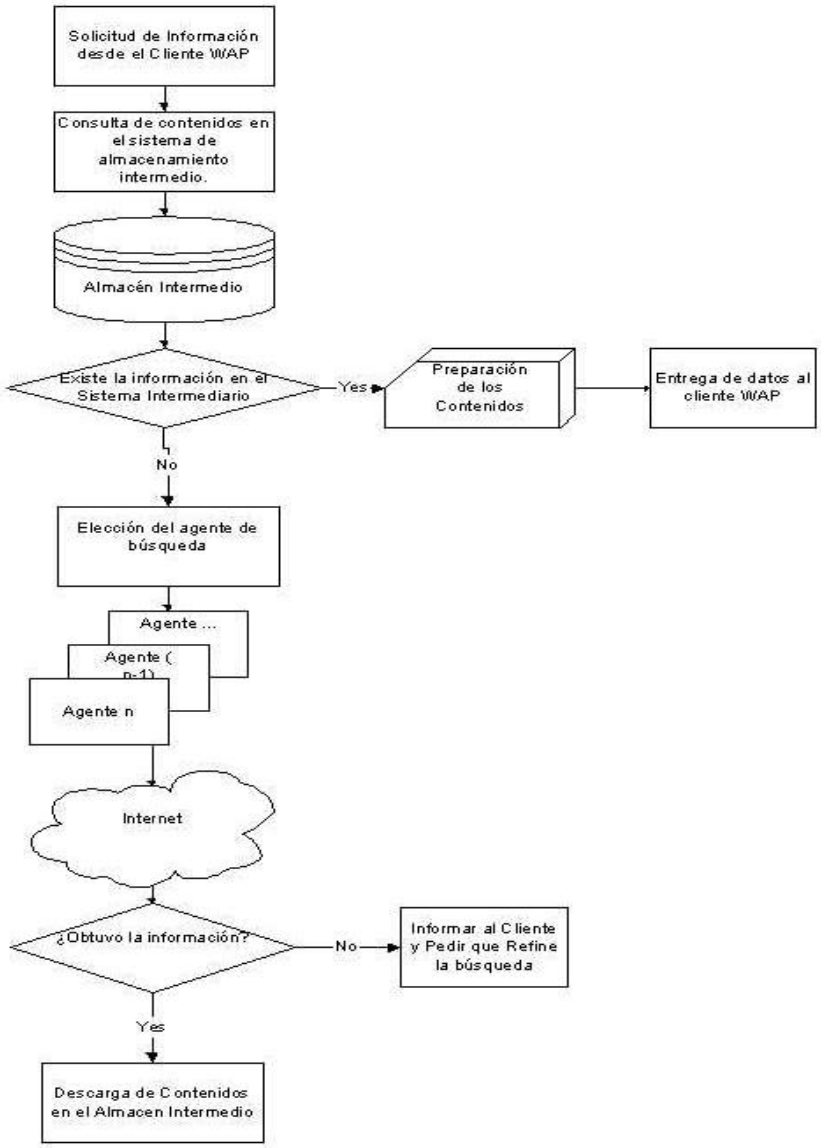


Figura 5. Flujograma de una solicitud (Pull) en entorno WAP.

se requiera un canal seguro para la negociación y el pago, en este caso el intermediario deja de intervenir, y se utiliza la transmisión de datos en modalidad segura extremo a extremo. Se debe considerar la implantación de sistemas de marcas de tiempo o relojes vectoriales así como la utilización de

infraestructura de clave pública para garantizar la confianza de las partes y del mensaje [21].

Figura 6 Entrega de información en modalidad Push utilizando el perfil de los usuarios.

Entre las propuestas para extender el estándar TLS, en "Wireless Extension for TLS" [26] se propone entre otras cosas, referenciar un sitio mediante su URL en donde encontrar los certificados del cliente, esto descarga al equipo móvil de la tarea de mantener los certificados del cliente. La autoridad de certificación del intermediario, se puede utilizar para mantener los certifcados de cada cliente, actualizar la lista de certificados revocados (CRL) y gestionar la información del estado de los certificados, por ejemplo, con OCSP (On-line Certificate Status Protocol).

Es necesario esconder al usuario la complejidad de los mecanismos involucrados en el comercio móvil. El uso de agentes inteligentes y sistemas expertos facilita tareas puntuales como por ejemplo; detección de proximidad geográfica con un centro de servicios, zonificación de la información, filtrado y tratamiento previo de la información que llega al equipo móvil. De forma similar; conocer el perfil del usuario y personalizar la entrega de información e incluso anticipar la entrega de información de interés. Se consigue extender las capacidades y mejorar las facilidades de uso de los equipos móviles utilizando un intermediario que gestione la relación entre el cliente e Internet.

5 Conclusiones

Se han analizado los aspectos que condicionan el entorno de telefonía móvil con capacidades de WAP, sus características, ventajas y limitaciones. Se propone la utilización de intermediarios destinados a ampliar las capacidades de los equipos móviles y actuar como integradores de las diferentes herramientas tecnológicas existentes. La propuesta se enfoca desde el punto de vista de la conveniencia del cliente.

Los canales seguros basados en TLS, proveen un medio eficaz de transmisión segura de contenidos a móviles y de contenidos entre el Intermediario y el proveedor en el Web. Sin embargo, la seguridad implica no solamente a los mecanismos de transmisión sino a los mecanismos de bases de datos, negociación, autentificación de cliente, sistema operativo, integridad del código móvil y de los agentes inteligentes móviles, así también las políticas de uso.

La utilización de tecnologías basadas en objetos tales como Java y CORBA permiten la transportabilidad de aplicaciones y reutilización de componentes. Permiten además racionalizar y simplificar el código y las funciones para entornos con restricciones tal como WAP.

La integración entre WWW y WAP permite el intercambio de transacciones, desarrollo de componentes, entre proveedores y clientes y una adaptación flexible a los entornos móviles.

Es conveniente que el intermediario almacene los certificados de cada cliente, constituyendo una autoridad de registro RA o al menos un repositorio de certificados de clientes. Los usuarios pueden consultar el estado de los certificados almacenados en el intermediario mediante un URL. La evolución de los estándares debe tender a compatibilizar TLS y WTLS.

Los mecanismos de Push y Pull de WAP permiten acceso a los datos en dos modalidades. Una manera de aprovechar el Push para entregar información en forma predictiva puede ser la personalización del servicio, sin embargo, es necesario filtrar la información de modo que se evite un flujo excesivo de mensajes hacia el móvil.

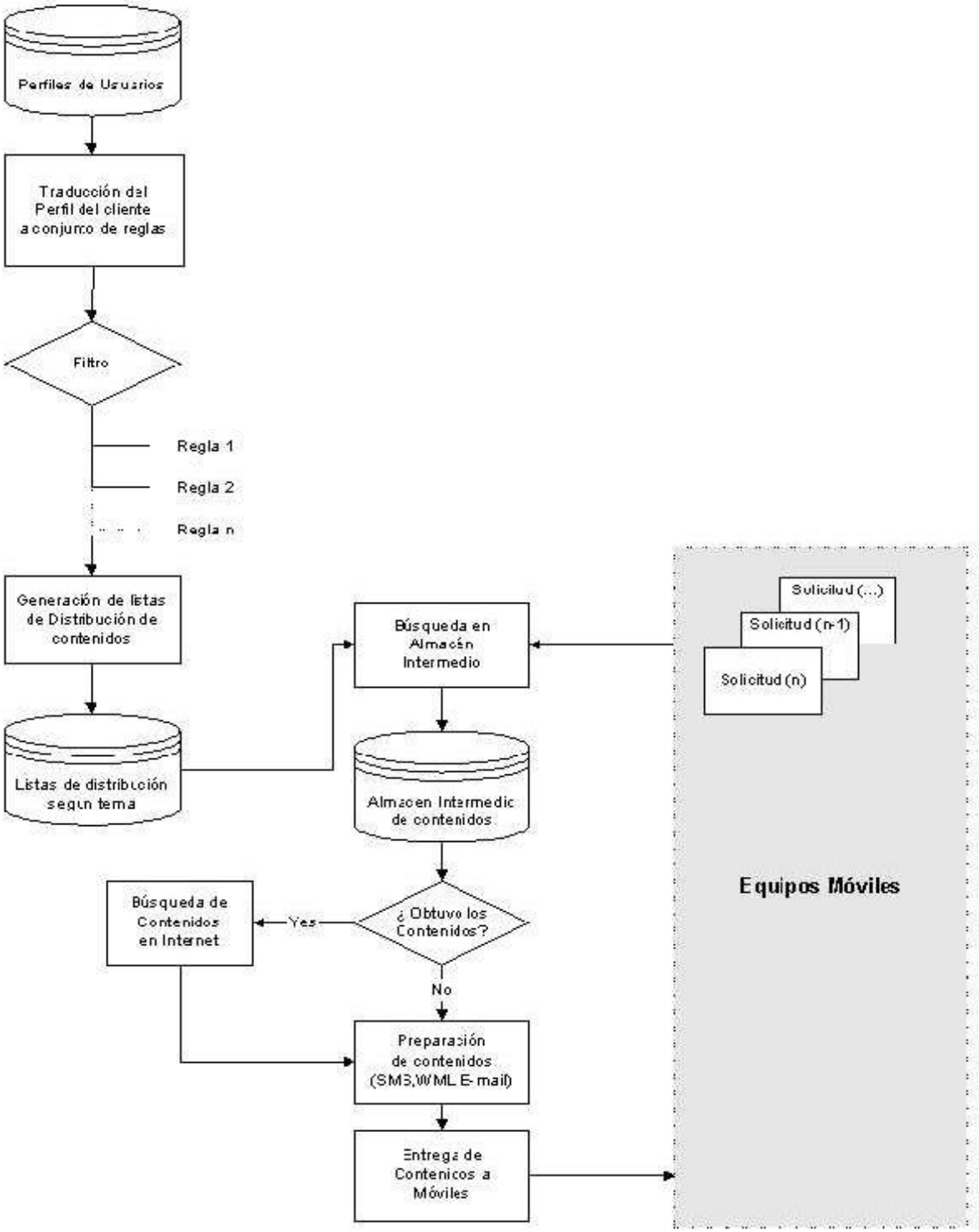


Figura 7 Entrega de contenidos al cliente WAP.

6 Referencias

[1] WapForum, *Wap 1.2 Specification*, (Jun. 2000), http://www.wapforum.org.

[2] M. Rader, U. Riehm, *Payments by mobile phone more convinient than e-wallets?*, ITAS, Karlruhe, Germany.

[3] J.H. Saltzer, D.P. Reed, D.D. Clark, *End to end Arguments in System Design*, ACM Transactions in Computer Systems 2.4, (Nov. 1984), pp. 277-288, http://web.mit.edu/Saltzer/www/publications/.

[4] T.Dierks, C..Allen, *The TLS Protocol version 1.0*, (Jan. 1999), Internet Standard RFC 2246.

[5] R. Khare, *W* Effect Considered Harmful*, IEEE Internet Computing, (Jul.-Aug. 1999), http://computer.org/internet/.

[6] A. Fasbender, F. Reichert, *Any Network, Any Terminal, Anywhere*, IEEE Personal Communications. (Apr. 1999), pp. 22-30.

[7] D. Van Thanh, *Security issues in Mobile eCommerce*, IEEE, ISBN 0-7695-0680-1, (Aug 2000).

[8] S. Halevi, H. Krawczyk, *Public-Key Cryptography and Password Protocols*, ACM Transactions on Information and System Security, Vol 2, No 3, (Aug 1999), pp. 230-268.

[9] Durlacher Research Ltd., *Mobile Commerce Report*, http://www.durlacher.com.

[10] K. Westhead, et al. *New Economy, Forget the Web make way for the grid.* Deustche bank, (Jun. 2000).

[11] Y. Malhotra, *Knowledge Management for e-Business performance: Advancing Information Strategy to "Internet Time"*, CRC Press, (2000).

[12] T. Sau Loon, V. Barghavan, *Alleviating the latency and bandwidth problems in WWW Browsing*, (Dec 1997), http://www.usenix.org.

[13] B. Pfitzmann, M. Waidner, *Properties of Payment Systems: General Definition Sketch and Classification*, IBM Research Division, (May. 1996).

[14] Sun microsystems, *The Java 2 Platform Micro Edition J2ME for Linux*, (Ene. 2001), http://java.sun.com

[15] Qualcomm, *Binary Run-time Environment for Wireless BREW*, (Feb 2001) http://www.qualcomm.com/brew.

[16] R. H. Guttman, et al., *Agent Mediated Electronic Commerce: A Survey*, Software Agents Group MIT Media Laboratory, http://ecommerce.media.mit.edu

[17] B. Venners, *Under the hood: The architecture of aglets*, Javaworld, (Apr. 1997), http://www.javaworld.com/javaworld/jw-04-1997/jw-04-hood.html.

[18] M. B. Blake, *KOJAC: Implementing KQML with Jini to Support Agent-Based Communication in Emarkets*, American Association for Artificial Intelligence. (2000). http://www.aaai.org.

[19] M. Cannataro, D. Pascuzzi, *An Object-Based Architecture for WAP-Compliant Applications*, IEEE, ISBN 0-7695-0680-1, (Aug. 2000).

[20] D. Ponce, M. Soriano, P. Mur, *A proposal for B2C Electronic Commerce Scheme based on WAP,* World Multiconference on Systemics, Cybernetics and Informatics, SCI 2000. (Jul 2000).

[21] M. Raynal, M. Singhal, *Logical Time:Capturing Causality in Distributed Systems*, IEEE Computer Magazine, (Feb 1996), pp. 49-56.

[22] Phone.com, *Understanding Security on the Wireless Internet.* (Ene. 2000). http://www.phone.com

[23] M. Wooldridge, N. R. Jennings. *Pitfalls of Agent-Oriented Development*. University of London.

[24] S. Kent, R. Atkinson, *Security Architecture for the Internet Protocol*. Request for Comments 2401. (Nov 1998).

[25] Nielsen Norman Group, *WAP Usability Report: Field Study Fall 2000*. (Dec. 2000). http://www.nngroup.com/reports/wap/.

[26] S. Blake, Magnus Nystrom, *Wireless Extensions to TLS,* draft-ietf-tls-wireless-00.txt, (Nov. 2000).

SISTEMA DE COMERCIO ELECTRÓNICO DE CONTENIDOS CON ANONIMATO MEDIANTE DINERO NO TRAZABLE

Juanjo Unzilla, Alejandro Muñoz, Javier Eguíluz, Cristina Perfecto
Departamento de Electrónica y Telecomunicaciones.
Universidad del País Vasco/Euskal Herriko Unibertsitatea
Alda. Urquijo s/n . 48013 – Bilbao
Teléfono: 94 601 42 06. Fax: 94 601 42 59
Email: jtpungaj@bi.ehu.es, jtpmumaa@bi.ehu.es, jtbegpej@aintel.bi.ehu.es, jtppeamc@bi.ehu.es

***Abstract**: Largely seduced by the emergence of the World Wide Web, the international business community is finally beginning to accept the Internet as a viable medium for doing business despite the flimsy security mechanisms. However the new cryptographic techniques such as "blind signature" are bringing security and anonymity to the electronic transactions. This cryptographic technique permits numbers to serve as electronic cash and to develop fully-anonymous electronic payment systems.*

1 Introducción

El gran desarrollo alcanzado por el comercio electrónico en los últimos tiempos, ha llevado a numerosas empresas privadas y organismos nacionales e internacionales a la publicación de multitud de esquemas y protocolos de pago electrónico. Hasta ahora, el gran objetivo (y casi único) de estos sistemas, era la seguridad en las transacciones electrónicas, de modo que los datos que se intercambian a través de una red insegura como Internet, viajaran de forma segura e indescifrable.

Sin embargo, las nuevas técnicas criptográficas han permitido desarrollar nuevas arquitecturas y protocolos que además de la seguridad, proporcionan el anonimato del usuario. Al realizar una compra en estos nuevos sistemas, los datos del usuario viajan de forma segura y además anónima para todas las demás partes del sistema.

El conjunto de los sistemas desarrollados hasta el momento, puede dividirse en dos grandes bloques atendiendo a su característica de anonimicidad. El primer grupo es el de los sistemas tradicionales en los que el anonimato del usuario no está contemplado. El segundo grupo es el de los sistemas más modernos que aplican técnicas de firma ciega y consiguen el pleno anonimato de los usuarios, como se verá posteriormente.

En este artículo, se tratarán los sistemas de pago electrónico aparecidos hasta la fecha y las nuevas propuestas realizadas en el contexto del comercio de contenidos. En primer lugar, se hará un breve repaso a los sistemas de pago existentes en Internet destacando sus características principales. A continuación, se presentará la nueva arquitectura propuesta que resulta de un diseño mixto de los sistemas existentes. Por último, se presentarán las conclusiones más relevantes de la implementación del sistema.

2 Comercio de contenidos

Por comercio de contenidos se entiende la compraventa de información en formato electrónico. Por tanto, no se produce intercambio de elementos susceptibles de entrega física. Uno de los problemas más importantes del comercio de contenidos es el de la protección de la propiedad intelectual debido a la facilidad de copia y distribución ilegal de la información entre usuarios [1].

Otro de los aspectos fundamentales de este tipo de comercio es el de la tarificación de la información entregada. Aunque en principio se podría pensar en el cobro de una pequeña cantidad por cada byte enviado de la página HTML, es más razonable pensar en el cobro de una determinada cantidad por el tipo de información entregada independientemente de la cantidad de bytes que ocupe. Con la ayuda del lenguaje XML, es posible clasificar la información según su contenido y de esta forma, facilitar la implantación de este modelo.

A continuación, se explica el funcionamiento del protocolo de firma ciega para poder entender mejor cómo funcionan el sistema de pago.

2.1 Firma ciega

El protocolo de firma ciega constituye una variante interesante de la firma digital tradicional y está estrechamente relacionada con las técnicas de encriptación de clave pública. Con las técnicas tradicionales de firma digital se consigue proporcionar a las comunicaciones la integridad (asegurar que el mensaje no sufre ninguna modificación), la autenticación (asegurar que el autor del mensaje es quien dice ser) y el no-repudio (asegurar que el originador no pueda decir que él no ha enviado el mensaje). Con las técnicas de firma ciega, se añade una característica adicional muy notable: el anonimato del usuario.

El objetivo principal de la firma ciega es conseguir una firma válida de una determinada entidad sobre un documento nuestro sin que esa entidad llegue a ver el contenido de dicho documento. De esta forma, el usuario puede obtener una firma válida manteniendo en secreto su identidad [2].

En la figura 1 se muestra el esquema básico de utilización de criptografía de clave pública para el intercambio seguro de información. Tomando como base este esquema, las técnicas de firma ciega introducen un elemento más en el sistema para obtener el anonimato del usuario. En concreto, los sistemas de firma ciega utilizan una técnica de aleatorización de la información antes de ser cifrada. Con este mecanismo, podemos conseguir que alguien nos firme de forma anónima esa información. En la figura 2 puede verse este proceso.

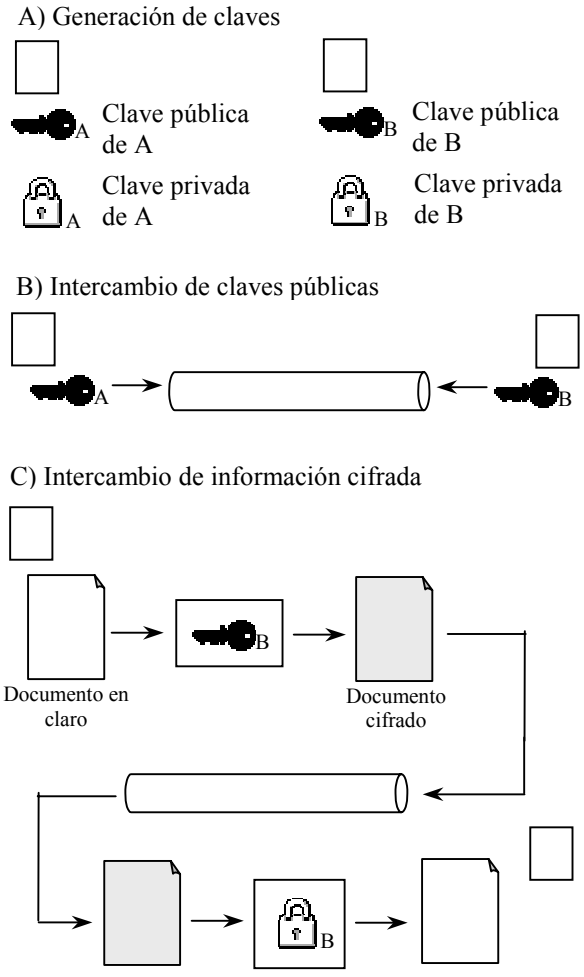


Figura 1.- Esquema básico de cifrado asimétrico para el intercambio seguro de información.

El protocolo de firma ciega se basa matemáticamente en el uso de funciones hash o de resumen, funciones de aleatorización de información y la exponenciación en el campo discreto de los números *módulo n*.

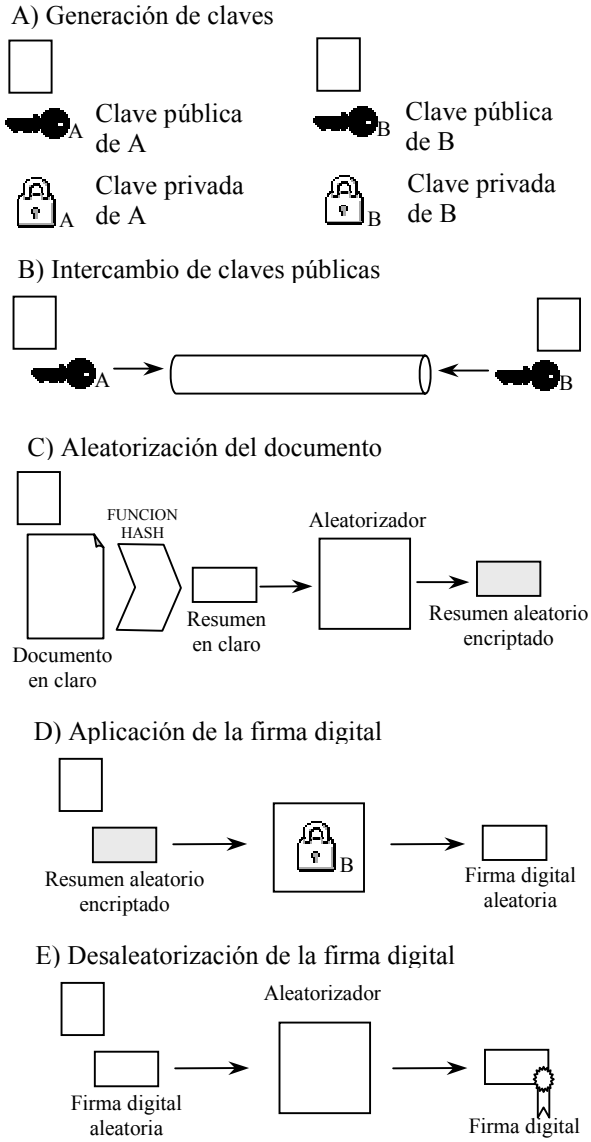


Figura 2.- Protocolo de firma ciega aplicado a un documento electrónico.

Su robustez se basa no solo en la dificultad de la factorización de números primos sino también en la dificultad de *adivinar* un número aleatorio de gran longitud. Los fundamentos matemáticos que posibilitan el entendimiento del protocolo de firma ciega para el caso en el que A quiera que B le firme un documento sin ver su contenido y sin revelar la identidad de A son los siguientes:

1.- Cada parte dispone de una clave pública formada por el par *(e, n)* y por una clave privada formada por el trío *(d, p, q)*. En este caso, cuando se hace referencia a *e, n, d, p* o *q*, éstos se refieren a los de B.

2.- A comunica a B que desea que le aplique la firma ciega a uno de sus documentos (que llamaremos *M*).

3.- A envía a B el documento aleatorizado de la siguiente forma:

$$M_{aleatorio} = H(M) \cdot r^e \ (mod\ n)$$

Donde *H(M)* representa la aplicación de una función hash al mensaje *M* original, *r* es el número aleatorio escogido por A para aleatorizar el mensaje y el par *(e, n)* es la clave pública de B.

4.- B aplica la firma digital sobre el documento que le entrega A (nótese que B no puede acceder al contenido del documento):

$$\sigma_{aleatoria} = (M_{aleatorio})^d \ (mod\ n)$$

donde *d* y $n=p \cdot q$ son la clave privada de B y σ es la firma digital sobre el mensaje. Una vez obtenida, B envía la firma aleatoria a A.

5.- Cuando A recibe la firma aleatoria, la puede desaleatorizar con la ayuda del número aleatorio que utilizó anteriormente para aleatorizar el mensaje:

$$\sigma = (\sigma_{aleatoria}) / r$$

Esta última propiedad se cumple ya que:

$\sigma_{aleatoria} = (M_{aleatorio})^d \ (mod\ n) = (H(M) \cdot r^e)^d \ (mod\ n) = (H(M)^d \cdot r^{e \cdot d}) \ (mod\ n) = H(M)^d \cdot r^1 \ (mod\ n) = H(M)^d \cdot r \ (mod\ n)$, donde $H(M)^d \ (mod\ n)$ es la firma digital de B sobre el documento en claro de A.

De este modo, el par *(M,* σ*)* constituye un par mensaje-firma válido sobre la clave pública de B. Como se ha visto, la entidad B ha realizado su firma sin ver en ningún momento el contenido del mensaje real. Por tanto, si posteriormente la entidad B ve el mensaje original de A, no sabrá a quién pertenece aunque sí que podrá verificar su firma sobre el documento.

2.2 Situación actual

El conjunto de los sistemas de pago electrónicos propuestos hasta ahora, puede dividirse en tres grandes grupos: los estándares de transacciones seguras, los sistemas de pago no anónimos y los sistemas de pago totalmente anónimos.

Los estándares de transacciones seguras (como *SSL* o *iKP*) son los mecanismos que se emplean hoy en día para llevar a cabo las transacciones seguras necesarias para realizar un determinado pago electrónico.

El segundo grupo lo forman los sistemas de pago no anónimos (como *CheckFree* o *SET*). Son sistemas específicamente desarrollados para realizar pagos electrónicos.

El último grupo de sistemas de pago está constituido por los sistemas de pago que permiten al usuario mantener su identidad totalmente oculta para el resto de elementos del sistema.

2.2.1 Estándares de transacciones seguras

No se trata de sistemas de pago electrónico seguros, sino que son estándares diseñados para conseguir una comunicación segura sobre canales de comunicación no seguros. A pesar de no ser sistemas de pago, son los mecanismos más empleados hoy en día en las transacciones llevadas a cabo en el comercio electrónico.

El más conocido de estos estándares es el SSL (*Secure Sockets Layer)*, desarrollado originalmente por *Netscape Communications* y actualmente estándar internacional bajo la denominación *TLS*. SSL se encuentra por debajo del nivel de aplicación (para el que es transparente) y por encima del nivel de transporte (sobre el que proporciona la seguridad necesaria).

Las comunicaciones tienen lugar en dos fases. En la primera de las fases, se negocia la clave simétrica que se usarán el cliente y el servidor y que será válida sólo para esa sesión. En la segunda fase, se pueden ya transmitir datos cifrados con esa clave simétrica de forma segura. Para que este protocolo funcione de la manera indicada, el cliente debe conocer de antemano las claves públicas de ciertas entidades que actúan como notarios digitales (la entidades de certificación o CAs). De esta forma, el cliente puede verificar la validez del certificado que le envía el servidor y que contiene entre cosas la clave publica del propio servidor.

2.2.2 Sistemas de pago no anónimo

Como ejemplo de sistemas de pago no anónimo se pueden citar a SET, CyberCash, CheckFree y NetBill. Estos sistemas comparten muchas de sus características con los sistemas de pago electrónico anónimo que se verán a continuación.

Prácticamente todos los sistemas propuestos utilizan soluciones propietarias, incluyendo protocolos, clientes y servidores propios [3]. En algunos de los sistemas, se deben tener cuentas especiales en el banco y en otros sistemas, se utilizan las cuentas tradicionales.

En algunos de los sistemas desarrollados (como CheckFree), las tiendas que deseen ofrecer este servicio a sus clientes tienen que estar registradas en el sistema. Algunas propuestas (como NetBill) utilizan Kerberos como sistema de autenticación.

El protocolo más conocido de este grupo es sin duda el protocolo SET (*Secure Electronic Transactions Protocol*). Este protocolo surge como resultado de la unión de los trabajos de VISA, MasterCard, IBM, Microsoft y Verisign principalmente. SET es un protocolo estándar para transacciones electrónicas en Internet con tarjetas de crédito. Proporciona características de autenticación, privacidad e integridad con la ayuda de criptografía de clave pública.

El protocolo SET puede operar en tiempo real o en entornos de *"almacenar y reenviar"*. Su objetivo no es el de ocultar la información relativa a la compra (como la lista de los productos adquiridos), sino evitar que la tienda descubra el número de la tarjeta de crédito del consumidor. Al utilizar SET, los consumidores ocultan sus números de tarjeta en una estructura determinada que sólo puede ser accedida por unos centros de procesamiento de tarjetas de crédito y no por las tiendas electrónicas. Para que este procedimiento pueda ser llevado a cabo, la tienda debe confiar en el centro procesador de tarjetas para que pueda validar la tarjeta del usuario.

Las características principales de SET incluyen la encriptación de clave pública y los certificados digitales. Soporta autenticación robusta, verificación on-line/off-line y no-repudio. Sin embargo, a las transacciones realizadas con SET se les puede seguir el rastro y por tanto la anonimicidad del usuario no está contemplada. Además, SET solamente soporta transmisiones seguras pero no ofrece privacidad a los usuarios.

El sistema NetBill se desarrolló en la Universidad de Carnegie Mellon en colaboración con el Mellon Bank. En este caso, también se trata de un sistema de criptografía de clave pública. Antes de llevar a cabo una transacción, el usuario y la tienda se autentican mutuamente usando sus certificados de clave pública y establecen una clave de sesión simétrica para conseguir una transacción segura. El sistema NetBill soporta la verificación en tiempo real de la base de datos de las cuentas de los usuarios además del no-repudio.

Otro de los sistemas propuestos ha sido el de la compañía CyberCash Inc. y que ha denominado CyberCash. Este sistema, ha sido desarrollado a partir de protocolos de Internet y de software propietario de la propia compañía. Básicamente, consiste en un gateway que se comunica con la tienda electrónica y con el banco del usuario y de la tienda.

El sistema completo comprende un software de usuario, un software para la tienda electrónica y un servidor que soporte las comunicaciones seguras necesarias para las transacciones de las tarjetas de crédito. La clave pública del gateway CyberCash se encuentra embebida en el código fuente de los programas de usuario y de la tienda. Los usuarios y las tiendas deben generar su propio par de claves pública-privada y deben entregar su clave pública al gateway.

Así, el gateway CyberCash mantiene una base de datos con la información de los usuarios y de las tiendas (incluyendo sus claves públicas). De esta forma, el gateway CyberCash puede autenticar las firmas tanto de clientes como de tiendas ya que conoce la clave pública de cada uno. En este caso, las tiendas electrónicas tampoco pueden acceder al número de tarjeta de los clientes, ya que la información va encriptada de una forma similar a SET.

2.2.3 Sistemas de pago anónimo

Los sistemas de pago anónimo constituyen una interesante variante surgida en los sistemas de pago electrónico. La característica más destacada que presentan respecto de los sistemas citados anteriormente es la consecución del anonimato absoluto de los usuarios en las transacciones realizadas.

Todos los sistemas anónimos desarrollados hasta ahora hacen un uso intensivo del protocolo de firma ciega, ya que es el medio disponible de mayor difusión en la actualidad para conseguir el anonimato en las transacciones, como ya se ha visto.

El sistema más conocido es el sistema Ecash desarrollado por la empresa holandesa DigiCash fundada por David Chaum, el creador de la firma ciega [4].

Además del sistema Ecash, la alternativa comercial más conocida es PayCash de la empresa Alkorsoft. En ambos casos, se trata de sistemas cerrados y propietarios que requieren la utilización de un software desarrollado por esas empresas.

En concreto, Ecash requiere la utilización de un software de usuario que debe instalarse en la máquina de cada usuario y que actúa como monedero electrónico. Los bancos también requieren de software específico y de una firma distinta por cada tipo de moneda que exista. Contiene un mecanismo de recuperación que permite al usuario recuperar el dinero que contenían las monedas electrónicas que ha perdido.

El sistema PayCash es muy similar al planteado por Ecash, aunque utiliza una variante de firma ciega desarrollada por su empresa.

El sistema que se presenta en este artículo constituye una variante del propuesto por Ecash e introduce una novedad en la arquitectura básica que lo hace más flexible.

3 Diseño del sistema

3.1 Introducción

El sistema propuesto se fundamenta en los desarrollos realizados fundamentalmente por Ecash y se aprovecha de las técnicas de firma ciega tradicionales. Es un sistema en el que la complejidad de la parte de usuario ha sido reducida al mínimo, manteniendo los mismos niveles de seguridad y anonimicidad que los desarrollos anteriores en los que se basa.

3.2 Elementos del sistema

El sistema de comercio de contenidos se compone de tres elementos principales: el usuario, la tienda electrónica y el banco electrónico. Además, y como funcionalidad añadida, se considera la inclusión de una infraestructura de certificación (PKI) para la generación y gestión de los certificados de los integrantes del sistema. Además de estos elementos principales, se encuentran otros menos importantes pero también imprescindibles, como son el dinero electrónico, los certificados y los contenidos. A continuación, se describen los diferentes elementos y sus requerimientos básicos.

3.2.1 Usuario

El usuario que desee acceder al sistema debe realizar muy pocos cambios a su forma actual de realizar las compras por Internet. Además de disponer de un PC con la conexión adecuada, debe disponer de un navegador con soporte completo de Java. De este modo, la funcionalidad básica del sistema se puede soportar sin realizar ningún cambio en la infraestructura del cliente.

3.2.2 Tienda

En este caso, al hablar de tienda electrónica nos estamos refiriendo al proveedor de contenidos al que el cliente quiere acceder. En este caso, la tienda sí que requiere del uso de un software especial que interactúe con el resto de elementos del sistema.

3.2.3 Banco

El tercer elemento imprescindible del sistema es el banco electrónico. Cuando se habla del banco, se hace referencia al banco del cliente que puede ser distinto al de la tienda y distinto al del resto de clientes.

En este caso, los cambios que debe realizar el banco en su infraestructura son mayores ya que además de disponer de un pequeño programa que se relacione con el resto de elementos del sistema debe disponer de una base de datos para almacenar información de las transacciones electrónicas realizadas.

3.2.4 Infraestructura de clave pública (PKI)

El último elemento que forma el sistema es la PKI. Su función principal es la de expedir los certificados de las distintas partes, además de entregarlos a los elementos del sistema que requieran comprobar la identidad de las otras partes del sistema.

3.2.5 Dinero electrónico

Se hace referencia a él indistintamente como billete o moneda electrónica, aunque en la literatura especializada se prefiere la palabra moneda electrónica [5]. Este elemento será una de las claves de funcionamiento del sistema y el principal elemento que conseguirá mantener el anonimato del usuario.

3.2.6 Certificados

Son proporcionados por la PKI y contienen, entre otros datos, la información de identidad y clave pública necesaria para la interacción segura de los distintos elementos del sistema. Básicamente se trata de un documento electrónico en el cual un elemento confiable para todas las partes (la PKI) asegura que cierta clave publica pertenece a cierta entidad, o lo que es lo mismo, asegura que la identidad de cierto elemento es veraz [6].

3.2.7 Contenidos

En este caso, la mercancía que se va a intercambiar entre el cliente y la tienda electrónica consiste en imágenes digitales y documentos de texto que la tienda pone a disposición de los usuarios. Se trata pues de un comercio de *bytes* y no de productos físicos.

3.3 Funcionamiento básico

El funcionamiento básico del sistema se recoge en la figura 3. El usuario desea acceder a unos determinados contenidos de pago que proporciona un proveedor de contenidos o tienda electrónica. Además, desea que su identidad no sea revelada ni

a la tienda ni al banco de donde obtendrá el dinero para realizar el pago.

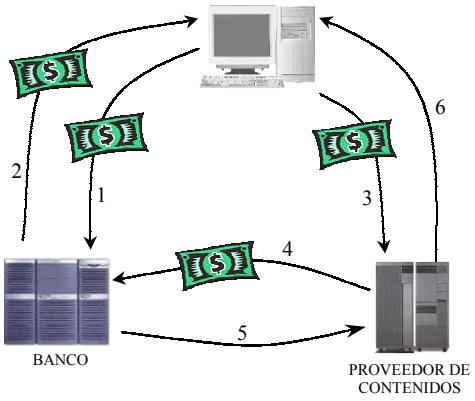


Figura 3.- Funcionamiento básico del sistema de comercio de contenidos.

Cuando ha decidido qué contenidos desea comprar, la tienda enviará al cliente la solicitud del pago necesario para obtener esos contenidos. A partir de ese momento, comienza el mecanismo necesario para realizar el pago de forma anónima. El proceso que se realiza es el siguiente:

① El cliente genera las monedas necesarias para satisfacer el pago, las oculta aplicando el protocolo de firma ciega y se las envía al banco.

② El banco comprueba la identidad del cliente, descuenta la cantidad necesaria de su cuenta bancaria y *legaliza* las monedas con un procedimiento determinado.

③ Una vez que el cliente ha obtenido las monedas ocultas y legales del banco, las *desoculta* aplicando de nuevo el protocolo de firma ciega y obtiene las monedas *normales* y legales. De esta manera, el banco ha podido descontar el dinero de la cuenta adecuada del cliente y ha legalizado las monedas, pero cuando el banco vuelva a ver las monedas que firmó al cliente, no podrá relacionarlas con el cliente ya que este último las ha *desocultado*. A continuación, el cliente enviará las monedas necesarias a la tienda electrónica para satisfacer el pago.

④ Una vez que la tienda ha recibido las monedas, verificará su validez y posteriormente las enviará al banco electrónico del usuario para comprobar que esas monedas no han sido gastadas anteriormente.

⑤ El banco coteja las monedas recibidas con las monedas ya gastadas que almacena en una base de datos. Si las monedas no han sido gastadas se lo comunicará a la tienda electrónica.

⑥ Si las monedas no han sido gastadas, se enviarán los contenidos y el recibo electrónico al cliente y en caso de que hayan sido gastadas se informará al cliente y se finalizará la transacción.

Como se ha visto, el sistema propuesto no presenta una gran complejidad en la parte de usuario, ya que la mayor parte de las tareas se realizan de forma automática y transparente al mismo.

3.4 Arquitectura funcional

La figura 4 muestra la arquitectura funcional del sistema en forma de diagrama de bloques. Como se ve en el diagrama, tanto la tienda como el banco deben instalar un pequeño módulo software que les permita interactuar con el resto de los elementos que componen el sistema.

Sin embargo, en el caso del usuario, no es necesario que instale ni configure ni mantenga ningún tipo de software para acceder al sistema. La razón es que el módulo de usuario presenta una novedosa arquitectura mixta que posibilita combinar las virtudes de las dos arquitecturas clásicas: centralizada y distribuida. Esta es la principal novedad del sistema.

El módulo del cliente (denominado "monedero") constituye el método de acceso del usuario al sistema, por lo que ha sido diseñado buscando la máxima sencillez y claridad. Así, el sistema propuesto abandona los tradicionales esquemas de software de usuario específico instalado en la máquina del usuario.

Este módulo presenta una arquitectura mixta: el software se almacena de forma centralizada, pero la ejecución es local en la máquina del usuario. De esta forma, la distribución del software de usuario es centralizada asegurándonos que todos los usuarios utilizan la misma versión del software y que se ejecuta la última versión.

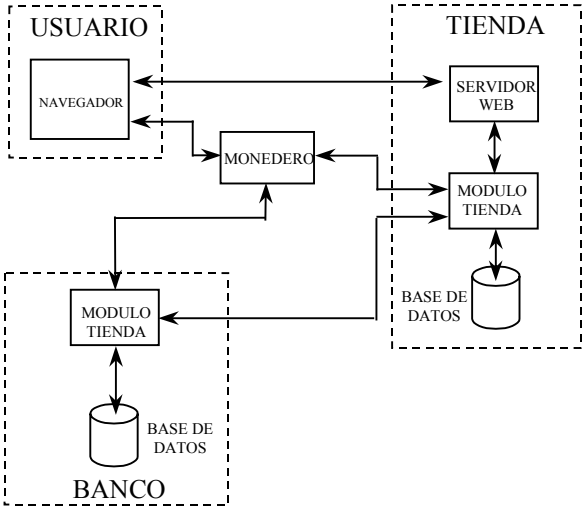


Figura 4.- Esquema funcional del sistema propuesto.

Sin embargo, la ejecución del programa es local y el almacenamiento de la información de usuario también se realiza de forma local, con lo que se mantienen las ventajas del software de usuario local en lo que concierne a los aspectos de seguridad.

Por otra parte, entre las funciones más importantes de este módulo se encuentran, además de la interacción con el usuario, la recepción de las solicitudes de pago, la generación del dinero electrónico y la realización de los pagos. Entre ellas, la función más importante es la de la generación de dinero, que efectúa siguiendo las técnicas de firma ciega. El usuario construye un archivo por cada moneda electrónica que necesita y que contiene los atributos necesarios para la moneda como son el número de serie, la divisa y el importe.

Después de construir las monedas necesarias, el monedero las envía al banco para que este último les aplique la firma ciega y se obtengan así las monedas electrónicas válidas. Con esta técnica, el banco no puede seguir el rastro de las monedas que firma, aunque si que puede verificar la validez de todas las firmas.

El módulo de la tienda consiste en un software local y cuyas funciones principales son las de interactuar con el usuario en el envío de solicitudes de pago y la comprobación de las monedas electrónicas recibidas de los clientes. La comprobación de las monedas recibidas es una operación que la tienda realiza on-line antes de la entrega del producto. Al ser la compra totalmente anónima, si la tienda no comprobase la validez de las monedas y el banco le indicara posteriormente que las monedas ya habían sido gastadas, la tienda no será capaz de recuperar el dinero que ha perdido al no poder descubrir la identidad del estafador.

Además de estas tareas principales, la tienda también se encarga de enviar los recibos o justificantes de pago y de ejecutar el mecanismo de resolución de contiendas en caso de que surja algún problema durante la ejecución de la transacción.

El último de los módulos es el del banco electrónico y es el que más necesidades de procesamiento requiere ya que tiene que interactuar con gran cantidad de usuarios y de tiendas. Además, se encarga de las tareas de legalización del dinero electrónico y la posterior comprobación de la validez de las monedas. La tarea más importante es la de la legalización del dinero generado por el usuario. Como se ha comentado, para que el dinero que acuña el usuario tenga validez, debe estar firmado por el banco, para lo cual se emplea el protocolo de firma ciega. Esto permite al banco legalizar el dinero de los usuarios sin que pueda posteriormente seguir el rastro de las monedas electrónicas (al igual que sucede en el mundo real con las monedas físicas).

Además de la legalización, el banco lleva a cabo la tarea de validación de las monedas electrónicas. La tienda recibe una determinada cantidad de monedas para satisfacer un pago electrónico. Estas monedas, son enviadas al banco para que este compruebe si ya han sido gastadas anteriormente o no.

Para llevar a cabo esta comprobación, el banco dispone de una base de datos con los números de serie de todas las monedas que han sido gastadas en el sistema, por lo que para comprobar si una moneda ya ha sido gastada, debe verificar si su número de serie está almacenado en la base de datos de las monedas gastadas.

5 Conclusiones

El comercio electrónico se está convirtiendo en un elemento imprescindible en nuestras actividades cotidianas. Las compras electrónicas son cada vez más habituales y por tanto, también lo son los pagos electrónicos. En los sistemas de pago tradicionales (que se utilizan hasta nuestros días) la única condición de diseño que se ha impuesto es la confidencialidad en las transacciones, olvidando por completo la característica de anonimicidad.

Sin embargo, en las compras físicas realizadas en el mundo real, además de la seguridad y de la confidencialidad, es claro que se ofrece la anonimicidad en los pagos, ya que en las compras realizadas con dinero en efectivo no es posible seguir el rastro de la transacción efectuada.

Por ello, surge la necesidad de que los pagos electrónicos posean la característica de anonimicidad de la que adolecen todos los sistemas actuales. De esta forma, en los últimos años han surgido varios sistemas de pago en los que la identidad del usuario permanece secreta tanto para la tienda como para el banco y la posibilidad de asociar pagos con pagadores es nula.

Para llevar a cabo estos sistemas, se han propuesto las técnicas de firma ciega. Los sistemas desarrollados hasta ahora, han sido diseñados siguiendo una estructura similar, en la que cada elemento posee un software específico para acceder al sistema. Este hecho no es importante en el caso de la tienda o en el banco, pero sí en el caso del usuario, que puede no tener los conocimientos técnicos suficientes para el manejo, configuración y mantenimiento del software necesario.

Por ello, el sistema propuesto combina las ventajas de las arquitecturas tradicionales con la posibilidad de eliminar gran parte de la complejidad soportada hasta ahora por el usuario. El software de usuario se distribuye de forma remota desde la tienda electrónica y se ejecuta en forma local en la máquina del usuario, por lo que se mantienen los

mismos niveles de seguridad que en el caso del software local.

Agradecimientos

El trabajo presentado ha sido realizado gracias a la ayuda recibida del Gobierno Vasco y de la empresa SARENET S.A., cofinanciadores del proyecto OD00UN57.

Referencias

[1] Unzilla, JJ.; Goirizelaia, I.; Jacob, E.; Ferro, A. Visión general de las técnicas de marcas de agua (watermarking) y sus aplicaciones. Proceedings de JITEL'99. II Jornadas de Ingeniería Telemática. Madrid. 15 – 17 Septiembre 1999.

[2] Zulfikar Amin Ramzan. *Group Blind Digital Signatures: Theory and Applications* Massachusetts Institute of Technology, Mayo 1.999

[3] Yinan Yang. *Security Mechanisms in Electronic Commerce.* http://www.cs.adfa.oz.au/~yany97/payments.html

[4] David Chaum. *Achieving Electronic Privacy*. Scientific American, Agosto 1.992, pp. 96-101

[5] N. Asokan, Phil Janson, Michael Steiner, Michael Waidner. *Electronic Payment Systems*. IBM Research Division, Zurich Research Laboratory.

[6] J. Feghhi, J. Feghhi, P. Williams. "Digital certificates". Addison-Wesley, 1999.

DelfosnetX: Sistema de Recuperación de Información basado en Metadatos*

P. Pavon, J. Rodriguez[1], M.J. Fernandez[1], M. Llamas[1], J. Santos[1], M. Caeiro[1], L. Anido[1]
Departamento de Tecnología de la Información y las Comunicaciones
Universidad Politécnica de Cartagena
Teléfono: +34 968325952 Fax: +34 968325338
E-mail: Pablo.Pavon@upct.es

Abstract*. In this paper we present DelfosnetX, a Java-based Information Retrieval (IR) system intended to evaluate different relevance analysis and ranking techniques for metadata-enabled IR, and more specifically, XML-based IR. A theoretical background supporting metadata IR is also proposed: the matrix model. Under this model, queries and documents are viewed as (tag , content) sets, and mapped as matrixes. XML files and other semistructured data formats can be expressed in this way, which is adopted as our metadata representation framework. Benefits of mapping documents and queries as matrixes are discussed. Transformation matrixes are also presented, which are simple mathematical ways to express many of the operations involved in IR systems. New operations on tags space, associated to metadata IR are presented as different transformation matrixes. Software design of DelfosnetX and the underlying storage techniques are fully described.*

1 Introducción

Los sistemas de recuperación de información (RI) se encargan de la representación, almacenamiento, organización y acceso a los elementos de información. La expansión del WWW y la mayor capacidad de procesamiento y almacenamiento de los equipos han llevado a un 'boom' de este tipo de sistemas. Al mismo tiempo, nuevos estándares para la representación de la información permiten un enriquecimiento y creciente complejidad de las búsquedas, siendo ésta una línea de investigación abierta, dentro de la que se enmarca el proyecto DelfosnetX.

En este documento se estudia la introducción de metadatos [1], tomando XML [2] como formato de representación, dentro del dominio de la recuperación de información (RI). La introducción de XML proporciona un entorno abierto dentro del cual es posible definir cualquier esquema de metadatos existente como Dublín Core [3] o crear esquemas de metadatos particulares para los nuevos escenarios que puedan emerger.

El proyecto DelfosnetX nació para la evaluación y análisis de este tipo de sistemas, frente a los sistemas RI tradicionales. El esfuerzo realizado ha permitido una transferencia desde la investigación hacia un prototipo comercial, en colaboración con la empresa Telémaco [4], situada en el Parque Tecnolóxico de Galicia [4]. Las aplicaciones de un sistema de este tipo van desde las bases de datos de documentos legales, documentos médicos, ofertas de empleo y cualquier tipo de información que se desee catalogar.

El resto del documento está organizado de la siguiente manera. En primer lugar, se profundizará en la motivación y objetivos del proyecto DelfosnetX. A continuación, se expondrán unos breves ejemplos de máquinas de búsqueda y sistemas RI que trabajen con ficheros XML. La sección 4 presenta la base teórica clásica que soporta la mayoría de los sistemas RI. La sección 5 presenta el modelo matricial desarrollado íntegramente dentro del proyecto DelfosnetX. En la sección 6 y 7 se presentará la arquitectura DelfosnetX y los servicios de búsqueda y evaluación de prestaciones que proporciona. Finalmente se expondrán las conclusiones y líneas futuras dentro del proyecto.

2 Motivación y objetivos

El concepto de metadatos ha sido clásicamente definido como "datos sobre datos". Los metadatos asociados a un documento, describen el contenido del mismo, dotándolo de una estructura, aportando un más alto grado de conocimiento.

Una ficha de un libro que además de su contenido especifique el autor, año, editorial, palabras clave,

* Este trabajo está amparado por la Comunidad Europea y el Ministerio de Educación bajo la concesión 1FD97 0282.

[1] Departamento de Tecnologías de las Comunicaciones, Universidad de Vigo.

resumen, etc. sería el ejemplo claro de metadatos (en contraposición a un libro como un documento plano).

Para los sistemas RI, la gestión de metadatos se traduce en una mayor expresividad en las consultas, y la necesidad de una base teórica donde las mismas tengan cabida, y poder evaluar así este tipo de sistemas. El trabajo realizado dentro del proyecto DelfosnetX ha incluido la propuesta de un modelo matemático donde las consultas y documentos son asociados a una matriz, frente al tradicional modelo vectorial [6], base de los sistemas RI no basados en metadatos. Esta propuesta ha sido publicada en [7], y será brevemente introducida en este documento, junto con las posibilidades que ofrece en áreas como sistemas RI distribuidos y sistemas de intermediación *(brokerage)*.

Como se ha argumentado previamente, XML es el formato predominante para la representación de metadatos, y ha sido adoptado por el sistema DelfosnetX. Mediante la definición del tipo de documento (DTD, *Document Type Definition*) es posible la descripción de las etiquetas, que conformarán un esquema de metadatos concreto. DelfosnetX no impone ningún DTD a los documentos XML, lo que lo convierte en un motor de búsqueda no específico, que se adapta a cualquier tipo de información que se desee catalogar.

3 Sistemas RI y XML

En los últimos años han surgido diferentes sistemas RI basados en ficheros XML, y un cierto número de lenguajes de consulta para este tipo de datos: XQL[8], XML-QL[9] o Lorel[10]. Dentro de los motores de búsqueda pueden destacarse:

- XRS[11], como una máquina de búsqueda sobre ficheros XML, basado en el sistema BUS[12] *(Bottom Up Scheme)*.

- Sistema XSet[13], donde las consultas son expresadas como documentos XML, cuyas etiquetas reflejan los parámetros de la búsqueda.

Estos sistemas resuelven de forma diversa el problema de (eficientemente) almacenar ficheros XML y procesar las consultas. En XRS, la gestión de la información está basada en tecnologías clásicas de ficheros invertidos [6]. XSet se basa en tablas *hash* que reflejan la estructura y organización de los documentos XML.

4 Sistemas RI clásicos

En lo sucesivo se calificarán como sistemas RI clásicos aquellos que no manejan información de metadatos. Numerosos modelos han surgido para representar matemáticamente los documentos y las consultas de esta manera: modelo vectorial, modelos basados en conjuntos difusos, o en redes neuronales son ejemplos. De hecho, este escenario se enriquece con variaciones y combinaciones de estos modelos. El lector puede encontrar en [6] [14] [15] explicaciones detalladas de los modelos citados anteriormente.

Nuestro análisis se centrará en el modelo vectorial, no sólo por ser el más arraigado y difundido, sino porque es en el que se basa el modelo matricial que propondremos a continuación. Una de las virtudes del modelo matricial es que incluye y extiende al modelo vectorial, lo que define un marco dentro del que adaptar técnicas clásicas a los nuevos requerimientos de metadatos.

Los siguientes conceptos y notaciones serán usados para describir nuestra propuesta:

- Las consultas y documentos se representan como vectores de dimensión *I*, donde *I* es el tamaño del diccionario (i.e. número de términos distintos en todos los documentos almacenados en el sistema RI).

- Una consulta *q* se representa $\vec{q} = (n_{1q}, \ldots, n_{Iq})$, y un documento *j* se representa $\vec{d}_j = (n_{1j}, \ldots, n_{Ij})$. Las coordenadas n_{ij} (n_{iq}) ponderan la importancia que el término *i* tiene dentro del documento *j* (consulta *q*). Se observa que este modelo descarta la información sobre la posición relativa de los términos dentro del documento. La colección de *J* documentos de la base de datos se representa por $B_{clásica} = \{n_{ij}, i=1...I, j=1...J\}$

- Las consultas inducen el cálculo de una función de similaridad $sim(\vec{q}, \vec{d}_j)$ que estima el parecido entre la consulta y los documentos de la colección: es en muchos casos una medida con las propiedades de distancia entre vectores. Para el cálculo de esta función se necesitan los coeficientes de la consulta y el documento, además de una serie de estadísticos extraídos de $B_{clásica}$, que se pueden catalogar en:

  - *G*: Estadísticos globales, asociados a la colección. Por ejemplo, *N* (número de documentos de la colección) es un estadístico G.

  - *T*: Se calculan para cada término del diccionario. Por ejemplo, $idf_i = \log(N/n_i)$ se usa comúnmente para calibrar el poder discriminador del término *i* (n_i es el número de veces que el término *i* aparece en toda la colección).

- *D*: Se calculan para cada documento. Por ejemplo, u_j, que representa el número de términos distintos que hay en ese documento.

5 Modelo matricial

El modelo matricial está pensado para un modelo de documento etiquetado, formado por pares (etiqueta,contenido). Un documento sin metadatos podría verse de esta manera como un documento con una sola etiqueta.

Sea B_{ext} una colección de J documentos etiquetados, representado por $B_{ext} = \{n_{ijm},\ i=1\ldots I,\ j=1\ldots J,\ m=1\ldots M\}$, donde I es el número de términos distintos, M el número de etiquetas distintas, y n_{ijm} representa el número de veces que el término *i* aparece dentro del contenido asociado a la etiqueta *m* en el documento *j*.

Las consultas y documentos serán representados ahora como matrices *MxI*. Para un documento $[d_j]$, cada fila *m* será un vector $\vec{d}_j^{\,m}$ representando el contenido asociado a esa etiqueta.

$$[d_j] = (\vec{d}_j^{\,1}, \ldots, \vec{d}_j^{\,M});\ [q] = (\vec{q}^{\,1}, \ldots, \vec{q}^{\,M})$$

donde $\vec{q}^{\,m} = (q_1^m, \ldots, q_I^m)$, y q_i^m indica el interés del usuario en documentos que contengan el término *i* asociado a la etiqueta *m*.

Las funciones de similaridad deben medir ahora el parecido entre matrices *MxI*, y los documentos recuperados son ordenados según ese resultado. El enriquecimiento de las consultas lleva aparejado una ampliación de los estadísticos de interés:

- *M*: Son calculados para cada etiqueta. Por ejemplo N_m, representa el número de documentos que tienen algún contenido asociado a la etiqueta *m*.

- *DM*: Calculados para cada etiqueta de cada documento. Por ejemplo, n_{jm} representa la suma de términos en esa etiqueta de ese documento.

- *TM*: Calculados para cada etiqueta y cada término. Por ejemplo $idf_{im} = \log(N_m / n_{im})$ representa el poder discriminador del término *i* en los contenidos asociados a la etiqueta *m*.

La utilidad de estos y otros estadísticos debe ser estudiada. Hasta nuestro conocimiento, la evaluación de este tipo de parámetros es un problema abierto.

5.1 Traducción de ficheros XML

Los ficheros XML tienen una estructura de etiquetas jerárquicas en forma de árbol (árbol DOM[16]). El modelo matricial no impone ninguna jerarquía a las etiquetas, con lo que es posible modelar ficheros XML como matrices documento, siendo este el mecanismo de trabajo del sistema DelfosnetX. La Fig. 1 muestra un ejemplo de cómo DelfosnetX traduce ficheros XML en matrices documento. En un primer paso el documento XML (parte superior) es procesado para extraer las etiquetas y los términos asociados a éstas (parte intermedia). Finalmente se construye la matriz (parte inferior) en la que se refleja el número de veces que aparece un término ligado a una etiqueta.

5.2 Matrices transformación

Dentro de una matriz documento (o consulta) D_{MxI}, las filas contienen la información asociada a una etiqueta, y las columnas la información asociada a un término.

Multiplicando por la izquierda la matriz documento por la matriz $T^e_{M'xM}$, se transforma separadamente el espacio de etiquetas, donde las nuevas etiquetas serán combinación lineal de las antiguas.

Multiplicando por la derecha la matriz documento por la matriz $T^t_{IxI'}$, se transforma únicamente el espacio de términos, donde los nuevos términos serán combinación lineal de los antiguos.

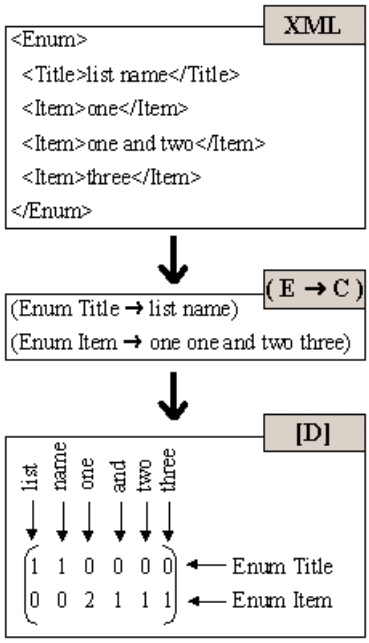


Figura 1: Modelado matricial de un documento XML

La aplicación de este tipo de transformaciones es directa dentro de los sistemas RI, como será mostrado más adelante. Por otro lado, una de las mayores ventajas del modelo matricial es la posibilidad de definir una función de similaridad con las propiedades de producto escalar. En efecto, teniendo en cuenta que :

$$traza(A_{mxm}) = a_{11} + a_{22} + ... + a_{mm}$$

se comprueba que:

$$\langle Q_{MxI}, D_{MxI} \rangle = traza(Q^T D) = traza(QD^T) =$$
$$= traza(D^T Q) = traza(DQ^T) =$$
$$= q_{11}d_{11} + q_{12}d_{12} + ... + q_{MI}d_{MI}$$

Esta función es una extensión para las matrices, del producto escalar de vectores, utilizado tradicionalmente en los sistemas vectoriales.

$$sim(\vec{q}, \vec{d}) = \frac{\vec{q} \cdot \vec{d}}{|\vec{q}| \cdot |\vec{d}|} = \frac{q_1 d_1 + ... + q_I d_I}{|\vec{q}| \cdot |\vec{d}|}$$

La ventaja en nuestro caso radica en las propiedades que tiene este producto escalar en combinación con las transformaciones por la derecha y por la izquierda. Se comprueba fácilmente que:

$$\langle Q, T_E D \rangle = \langle T_E{}^T Q, D \rangle$$
$$\langle Q, D T_T \rangle = \langle Q T_T{}^T, D \rangle$$
$$\langle Q, T_{E1} T_{E2} D T_{T2} T_{T1} \rangle = \langle T_{E2}{}^T T_{E1}{}^T Q T_{T1}{}^T T_{T2}{}^T, D \rangle$$

Gran parte de las operaciones realizadas comúnmente dentro de los sistemas RI se pueden expresar mediante matrices de transformación. En el caso de las transformaciones en el espacio de términos, es la matriz T_T la implicada:

1. La aplicación de *stoplist* (eliminación de palabras no semánticas, como artículos, preposiciones...) es modelado con una matriz T_T en la que desaparecen las columnas correspondientes a los términos descartados.

2. *Stemming* o extracción de raíces, donde varios términos con la misma raíz confluyen en un único nuevo término. Esto no es más que una simple combinación lineal de términos.

3. Traducción de diccionarios, donde la polisemia y sinonimia se expresan a través de los coeficientes no diagonales de la matriz de transformación de términos.

La matriz T_E, modela diversas operaciones que aparecen en los nuevos sistemas RI basados en metadatos:

- Una matriz de transformación T^e_{1xM} produce documentos sin etiquetar (con una sola etiqueta), permitiendo ponderar la importancia que cada etiqueta original tiene en el documento resultado.

- Las matrices T^e_{MxM} diagonales (con los elemento $t_{ij}=0$ para $i \neq j$) reponderan las etiquetas del documento, dando por ejemplo más importancia a los contenidos que aparecen en el campo “título” frente a “descripción”.

- Matrices genéricas $T^e_{M'xM}$ pueden ser utilizadas para traducir documentos entre dos esquemas de etiquetas, de dimensiones M y M' respectivamente. En el ámbito de la tele-educación por ejemplo, sería posible definir una matriz de transformación entre los esquemas LOM[17] e IMS[18] para almacenar información de recursos educativos. Así sería posible por ejemplo hacer consultas LOM a un sistema RI que almacene documentos en el formato IMS. Otra aplicación directa es el campo de los sistemas RI distribuidos, donde es necesario hacer consultas a distintos servidores que, potencialmente, pueden almacenar documentos según esquemas distintos (que llevarán asociadas distintas $T^e_{M'xM}$). En este caso además, debido a las propiedades 1 a 3, es posible hacer en el cliente (matriz consulta) las transformaciones que de otra manera solo podría hacer el servidor (matriz documento). Por otro lado, esta misma equivalencia, modela el problema de unir en un solo *ranking* final los diversos *rankings* obtenidos de cada servidor.

La posibilidad de modelar matemáticamente y de forma conjunta estas operaciones, abre la puerta al estudio sistemático de las mismas. Asimismo, el hecho de que el modelo vectorial sea reducible a un caso particular del modelo matricial, permite aprovechar y adaptar gran parte del trabajo realizado en este campo.

6. Evaluación de sistemas basados en metadatos

Una vez definido un modelo sencillo, donde es posible englobar sistemas basados en metadatos, y donde formatos establecidos como XML tienen cabida, DelfosnetX evolucionó para permitir la evaluación y estudio de distintas funciones similaridad matricial. El rendimiento de un sistema RI se mide mediante su respuesta a determinadas consultas estandarizadas sobre colecciones de documentos estandarizados (p.e. base de datos de la Fibrosis Cística [19] o colecciones TREC [20]). En

la actualidad, el sistema permite a investigadores en este campo, a través de una simple conexión a Internet, la evaluación de sus propias funciones de similaridad, listas de *stoplist* y funciones de *stemming* sobre documentos de la base de datos de referencia *Cystic Fibrosis*, indexados según el modelo matricial en el servidor.

7 DelfosnetX

Las características principales ofrecidas por el sistema DelfosnetX son las siguientes:

- Capacidad para indexar y recuperar atendiendo a medidas de relevancia, documentos XML con cualquier formato DTD, que serán mapeados según el modelo matricial. Se precalcularán automáticamente diversos estadísticos sobre los documentos indexados.

- Posibilidad de configurar el tipo similaridad a aplicar, tipo de extractor de raíces *(stemmer)* y lista de palabras prohibidas *(stoplist)* que se desee.

- Investigadores en este campo pueden programar sus propias medidas de similaridad según se indica en la documentación del sistema, y probarlas sobre las bases de datos de referencia almacenadas. Se proporcionan puntos *precision-recall* [6] para la automatización de pruebas.

- Control de acceso a los usuarios del sistema, permitiendo otorgar individualmente a cada usuario permisos para ejecutar cada comando de la API.

- Por supuesto, la realización de consultas siguiendo el modelo clásico, o el modelo etiqueta-contenido, son también parte de la funcionalidad del sistema. Junto con los documentos, es posible recuperar un objeto asociado, con los contenidos que se deseen (imágenes, referencias externas, datos...). Esto permite que el documento XML indexado contenga únicamente la información influyente en la búsqueda, para llegar a esos datos asociados.

7.1 Arquitectura del sistema

DelfosnetX ha sido programado en lenguaje Java, siguiendo el paradigma de programación en tres capas: acceso cliente, capa intermedia, y base de datos de almacenamiento. La relación entre los componentes del sistema se muestra en la Fig. 2:

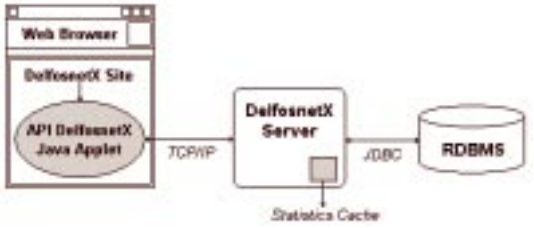


Figura 2: Arquitectura de DelfosnetX

- **Módulo cliente**: El acceso a toda la funcionalidad del sistema se proporciona a través de un *applet* Java que implementa la API *(Application Program Interface)* DelfosnetX. Esto permite a usuarios interesados crear sus propios programas Java y páginas Web que accedan al sistema a través de este API, bien desde Java o desde JavaScript. El *site* DefosnetX son las páginas creadas como parte del proyecto, que cualquier usuario puede utilizar para acceder a los servicios de búsqueda y administración.

- **Módulo servidor**: Este módulo gobierna el funcionamiento del sistema. Ejecuta las peticiones enviadas por los usuario a través del módulo cliente, accediendo a los datos de la capa de almacenamiento.

- **Módulo de almacenamiento**: Gestiona el almacenamiento de los datos sobre los que trabaja el sistema: los propios documentos y la información de control y administración necesaria. El soporte elegido ha sido una base de datos relacional, que el módulo servidor accede vía conexiones JDBC [21].

7.2 El modelo relacional

La utilización de una base de datos relacional como soporte para nuestro sistema RI está fundado en el modelo de metadatos (etiqueta,contenido), al que se traducen los documentos XML. Esta traducción supone una validación y procesado del fichero XML, una extracción de las etiquetas y términos asociados a cada una de ellas.

En la Fig. 3 se muestra la organización interna de la base de datos. Como se puede apreciar en dicha figura, diferentes tablas relacionan términos con etiquetas, términos con documentos y términos con etiquetas y documentos. Además se almacenan tablas con diccionarios de términos, etiquetas y documentos.

Cada una de las tablas es indexada, pudiéndose ver cada uno de estos índices como un fichero invertido, empleado clásicamente en las bases de datos documentales. La elección del modelo relacional ha permitido una solución rápida y limpia del problema de almacenamiento para grandes volúmenes de datos.

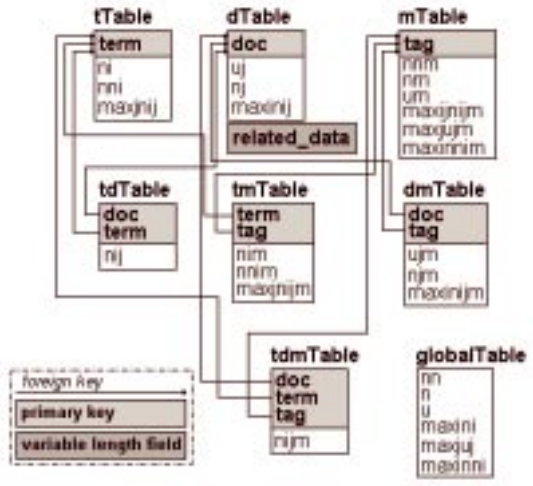


Figura 3: Organización de la base de datos

Además de términos, etiquetas y documentos, en la base de datos se almacenan estadísticos relacionados con cada uno de ellos, que son recalculados cuando nuevos documentos son introducidos en la base de datos. Estos estadísticos podrán ser utilizados por las funciones de similaridad registradas en el sistema, con lo que la programación de las mismas se simplifica enormemente.

7.3 Configuración de pruebas sobre DelfosnetX

DelfosnetX permite la definición y realización de pruebas sobre los siguientes aspectos de los sistemas IR:

- Funciones de similaridad, que se adapten al modelo matricial (que recordamos que incluye el modelo vectorial). Estas funciones deben estar programadas en Java, cumpliendo un determinado interfaz documentado.

- Ficheros de *stoplist* para el filtrado de documentos y consultas.

- Extracción de raíces *(stemming)* para el filtrado de documentos y consultas. La función *stemmer* a probar debe ser programada en Java, cumpliendo un determinado interfaz documentado.

- Para realizar estas pruebas, existe una base de datos de referencia almacenada en el sistema. Los usuarios con los permisos adecuados pueden sin embargo añadir y eliminar documentos XML de la base de datos.

8 Conclusiones y líneas futuras

Los sistemas RI surgieron por la necesidad de manejar volúmenes de información cada vez mayores, información que debe ser catalogada y clasificada para poder ser útil. La mejora en la capacidad de procesamiento y almacenamiento de los equipos ha permitido incorporar esquemas más complicados de representación de los datos, basados en metadatos. El lenguaje XML destaca en este campo por su flexibilidad, al ser posible expresar con él cualquier esquema de metadatos.

Los modelos matemáticos pensados para sistemas RI no basados en metadatos son incapaces de evaluar la ganancia en expresividad y rendimiento de cara al usuario que los metadatos suponen.

En este documento se presenta brevemente el modelo matricial y el concepto de matriz de transformación, surgido del trabajo realizado dentro del proyecto DelfosnetX. Se describe su capacidad para modelar sistemas RI basados en metadatos, mostrándose que numerosas operaciones comunes dentro de estos sistemas pueden ser modeladas fácilmente. Esto abre la puerta a un estudio sistemático de los sistemas RI basados en metadatos.

El sistema DelfosnetX ha sido implementado siguiendo el modelo matricial de representación de documentos y consultas. Incluye una serie de servicios, a través de una interfaz Java, que permite a investigadores conectados a través de Internet, probar sus propuestas sobre una base de datos de referencia almacenada en el servidor.

Actualmente nos encontramos en un proceso de evaluación de la herramienta, cuyo objetivo fundamental es conseguir un producto comercial. Nuestro colaborador empresarial está realizando un estudio pormenorizado de la plataforma y de sus posibilidades para adaptarla a las necesidades de los clientes potenciales. Como indicábamos anteriormente, DelfosnetX es una plataforma abierta sobre la que desarrollar sistemas de gestión documental, basados en XML, para un entorno de red. En este sentido, parece que los sectores donde tendría más aceptación una aplicación concreta basada en DelfosnetX son el médico y el jurídico.

Por otra parte, estamos estudiando la adaptación del soporte teórico y de la metodología de recuperación de información y gestión documental desarrollados a entornos de aprendizaje electrónico. Más en concreto, pretendemos utilizar nuestros resultados para dar soporte a la localización de recursos educativos dentro un servicio de intermediación (*brokerage*). Esto supone, entre otras cosas, la adaptación de DelfosnetX para soportar metadatos educativos, o la modificación del modelo de comunicación para permitir la integración en un entorno de intermediación. El resultado de este proceso se puede ver también como una nueva versión de DelfosnetX, adaptada a las necesidades particulares (lógica de negocio) de la intermediación de recursos educativos.

Referencias

[1] What is Metadata? Información disponible en http://luna.cas.usf.edu/~gregory/metadata/, accedido 29 Marzo 2001.

[2] T. Bray, J. Paoli, E. Maler (2000). "Extensible Markup Language". W3C Recommendation. Versión electrónica disponible en http://www.w3.org/TR/2000/REC-xml-20001006.pdf.

[3] "DublinCore Working Group. DublinCore Metadata for Resource Discovery". Dublin Core 1998. Versión electrónica disponible en ftp://ftp.isi.edu/in-notes/rfc2413.txt.

[4] Parque Tecnolóxico de Galicia, Consellería de Industria y Comercio, Xunta de Galicia. Información disponible en http://www.ptg.es, accedido 29 Marzo 2001.

[5] Telémaco, Parque Tecnolóxico de Galicia. Información disponible en http://www.telemaco.com, accedido 29 Marzo 2001.

[6] R. Baeza-Yates, B. Ribeiro-Neto. "Modern Information Retrieval". Addison-Wesley, 1999.

[7] M. J. Fernández, P. Pavón, J. Rodríguez, L. Anido, M. Llamas. "DelfosnetX: A Workbench for XML-based Information". Retrieval Systems Procs. of the Seventh International Symposium on String Processing and Information Retrieval. SPIRE 2000. IEEE Computer Society. Los Alamitos, CA. September. ISBN 0-7695-0746-8, pp. 77-85.

[8] J. Lapp, J. Robie, D. Schach. "XML Query Language (XQL)". W3C, 1998. Versión electrónica disponible en http://www.w3.org/TandS/QL/QL98/pp/xql.html.

[9] A. Deutsch, M. Fernández, D. Florescu, A. Levy, D. Suciu. "XML-QL: A query language for XML". W3C, 1998. Versión electrónica disponible en http://www.w3.org/TR/1998/NOTE-xml-ql-19980819.

[10] J. S. Abiteboul, D. Quass, J. McHugh, J. Windom, J. Wiener. "The Lorel query language for semistructured data". International Journal on Digital Libraries, Abril 1997, 1(1):68-88.

[11] XRS. Información disponible en http://dlb2.nml.nih.gov/~dwshin/xrs.html, accedido 29 Marzo 2001.

[12] D. Shin, "BUS: An effective indexing and retrieval scheme in structured documents", Procs. of Digital Libraries 98, 1998.

[13] Xset. Información disponible en http://www.cs.berkeley.edu/~ravenben/xset, accedido 29 Marzo 2001.

[14] A. Salton, M. J. McGill. "Introduction to Modern Information Retrieval". McGraw-Hill, 1983.

[15] W.F. Frakes, R. Baeza-Yates. "Information Retrieval. Data Structures and Algorithms". Prentice-Hall, 1992.

[16] W3C DOM Working Group. "Document Object Model". W3C, 1998. Versión electrónica disponible en http://www.w3.org/DOM.

[17] W. Hodgins. "Draft Standard for Learning Object Metadata (LOM) Specification. Proposed Draft 5.0". Institute of Electrical and Electronics Engineers, Inc, 2000. Versión electrónica disponible en http://ltsc.ieee.org/doc/wg12/LOM_WD5.doc

[18] T. Anderson, T. Wason. "IMS Learning Resource Metad-data Information Model". IMS Global Learning Consortium, Inc., 2000. Versión electrónica disponible en http://www.imsproject.org/metadata/mdinfov1p1.html

[19] W. M. Shaw, J. B. Wood, R. E. Wood, H. R. Tibbo, "The cystic fibrosis database: content and research opportunities". Library and Information Science Research, 13, 1991, pp. 247-366.

[20] Text Retrieval Conference (TREC). Información disponible en http://trec.nist.gov/, accedido 29 Marzo 2001.

[21] S. White, M.Fisher, R. Cattell, G. Hamilton, M. Happer. "JDBC™ API Tutorial and Reference, Secon Edition: Universal Data Access for the JAVA™ 2 Platform (Java Series)". Addison-Wesley, 1999. ISBN: 0201433281.

Desarrollo de un sistema remoto de consultas y reservas mediante interfaz oral.

Jesús E. Díaz-Verdejo, Pedro García, Ramón López-Cózar, Juan M. López-Soler,
Juan M. Estévez, A. Rubio Ayuso
Dpto. Electrónica y Tecnología de Computadores. Universidad de Granada.
Facultad de Ciencias. 18071 - Granada
Teléfono: 958 244 011 Fax: 958 243 230
E-mail: jedv@ugr.es

Abstract. *This paper describes the architecture, functioning and development of an automated system for information retrieval in a bus company environment. The access to the service is done orally by using a telephone. The system has been developed with the main objective of providing a natural and comfortable service. Therefore, the user should speak in a natural way, without constraints nor a predefined flow of query-answers. For this purpose, a dialog system is included in order to guide the interaction between the system and the user. Other main components are a continuous speech recognition system and a text-to-speech synthesizer. A keypoint for the proper operation of the system is the modeling of real dialogs that have been obtained by monitoring a man-operated information system.*

1 Introducción

La posibilidad de acceso remoto a información de diversa naturaleza constituye uno de los objetivos prioritarios de la denominada "Sociedad de la Información". Habitualmente, el acceso se realiza mediante "terminales" de usuario, como pueden ser, p.e., ordenadores o teléfonos móviles dotados de navegadores, en los que la comunicación se basa en técnicas totalmente diferentes de la comunicación habitual en el hombre, esto es, la comunicación por vía oral. Sin embargo, la tecnología de reconocimiento del habla posibilita el desarrollo de sistemas automáticos que permitan la comunicación con los usuarios mediante la voz [1], [2], [3]. El sistema en desarrollo que se describirá a continuación pretende permitir una comunicación oral para la adquisición de los datos de interés, así como para ordenar ciertas acciones o actuaciones relacionadas con los mismos. De esta forma se consigue que la comunicación resulte más natural al usuario al tiempo que se sustituyen los costosos terminales informáticos por un sencillo terminal telefónico sin ningún tipo de función adicional.

Evidentemente, la disminución de los requerimientos relativos a uno de los extremos de la comunicación se traducirá en el incremento de la complejidad del sistema final a desarrollar.

El presente artículo se desarrollará realizando, en primer lugar, una descripción funcional del sistema en la que se señalarán tanto los objetivos como los problemas a abordar. También se presentarán los antecedentes del mencionado sistema, indicando los aspectos en los que se pretenden introducir mejoras, así como la arquitectura propuesta para la consecución de los objetivos marcados. A continuación se describirán los tres módulos principales del sistema, esto es, el módulo de reconocimiento de voz, el módulo de diálogo y el módulo de conversión texto-a-voz. Para finalizar, se presentarán algunos aspectos relativos a la implementación práctica del sistema y su evaluación.

2 Descripción del sistema

2.1 Descripción funcional

El sistema en desarrollo proporciona un servicio de consulta oral automática para una compañía de autobuses, contemplando la posibilidad de realizar la reserva de billetes.

El objetivo del sistema es, por tanto, proporcionar la información solicitada por los locutores mediante la interacción oral con los mismos. Para ello, obviamente, debe ser capaz de determinar la información objeto de consulta a partir de las preguntas que se le realicen. Este objetivo aparentemente simple presenta enormes dificultades en varios aspectos.

En primer lugar, es necesario "transcribir" las preguntas realizadas, para lo que se utilizarán sistemas de reconocimiento de voz.

Por otra parte, las consultas suelen contener ambigüedades e imprecisiones que imposibilitan la obtención de la información mediante consultas directas a bases de datos, siendo necesario acotar y delimitar claramente el objeto de las mismas. Para ello es necesario interaccionar con el hablante a fin de obtener la información adicional que se estime

A: Alsina, buenos días, le atiende Noelia, dígame.
B: Buenos días. Para saber los horarios a Carchuna.
A: ¿Para hoy?
B: No para ... para mañana
A: A Carchuna tiene a las 6 45 de la mañana, 9 30 de la mañana, 11 30, 17, 17 30 y 19 30
B: Vale muchas gracias.
A: Nada, a usted.
B: Hasta luego.

Fig. 1: Ejemplo de conversación real.

oportuna en cada caso y que permita establecer la consulta exacta a realizar a la base de datos. Es necesario, por tanto, incluir un sistema de diálogo que mantenga una "conversación" con el locutor que permita determinar exactamente la información requerida (véase la Fig. 1).

Finalmente, será necesario proporcionar los mensajes salientes en forma oral, para lo que se utilizará un conversor texto-a-voz, dada la naturaleza dinámica de los mensajes que hay que proporcionar en cada fase de la interacción con los locutores.

Cada uno de los tres aspectos mencionados presentan una problemática específica, que se detallará posteriormente, si bien la principal dificultad del sistema reside en la capacidad del mismo para desarrollar una "conversación" de la forma más natural posible. Idealmente, el locutor no debería ser capaz de determinar si le está atendiendo un operador humano o un sistema automático, ya que de ello depende en gran parte la aceptación del sistema por parte de los potenciales usuarios del servicio.

2.2 Antecedentes

El sistema descrito presenta un antecedente inmediato en el denominado Sistema Telefónico Automático de Consulta de Calificaciones (STACC) [4] desarrollado con tecnología propia por el Grupo de Investigación en Señales y Comunicaciones de la Universidad de Granada. Dicho sistema (Fig. 2) permite a los alumnos de algunas titulaciones acceder a sus calificaciones de forma automática a través de una consulta telefónica, sin más que ir respondiendo a las preguntas que se le realizan. A este fin, el sistema utiliza un módulo de control basado en un autómata de estados finitos (Fig. 3). Este autómata permite determinar el mensaje o pregunta saliente que se le realiza al usuario, así como delimitar, en función de los datos disponibles, las posibles respuestas que puede proporcionar (modelo de lenguaje). El modelo de interacción es, por tanto, rígido, no pudiendo hablarse de diálogo real entre la máquina y el usuario, ya que todo el proceso está especificado a priori.

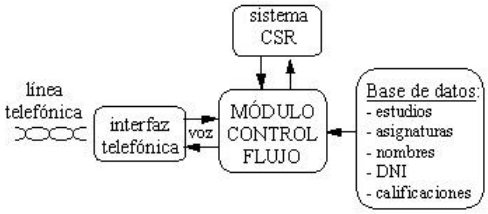


Fig. 2: Diagrama de bloques del sistema STACC.

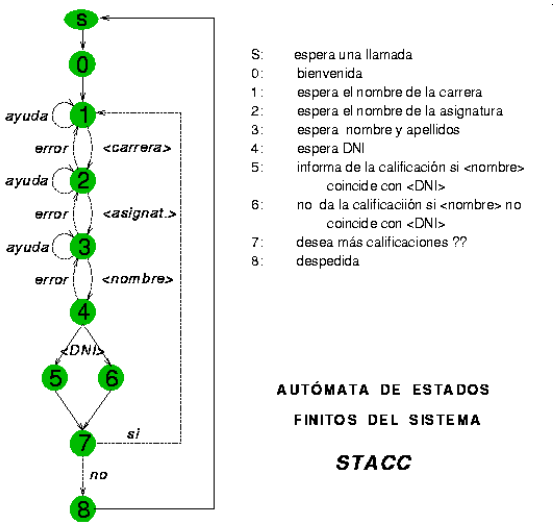


Fig. 3: Autómata de estados finitos de STACC.

El diseño aplicado para el sistema descrito permite cambiar fácilmente la aplicación concreta a la que está destinado sin más que modificar los archivos de descripción del autómata y los mensajes salientes. A modo de ejemplo, esta característica ha posibilitado su utilización para proporcionar información personalizada sobre la cita previa de los alumnos de la Facultad de Ciencias para la realización de la matrícula durante el curso 2000/2001. Sin embargo, el propio diseño del sistema impone limitaciones a su uso, ya que es necesario estructurar, de acuerdo a un autómata de estados finitos, toda la interacción con el locutor. Por otra parte, las posibles respuestas del locutor deben encontrarse en un grupo finito, que también se establece previamente a partir de los datos disponibles. La naturalidad de la interacción es, por tanto, reducida, ya que se exigen respuestas concisas y precisas.

2.3 Arquitectura del sistema

La arquitectura del sistema en desarrollo se muestra en la Fig. 4. En ella se pueden diferenciar varios grupos funcionales que, a su vez, se encuentran implementados en distintos módulos que se relacionan entre sí de forma específica a fin de conseguir los objetivos globales.

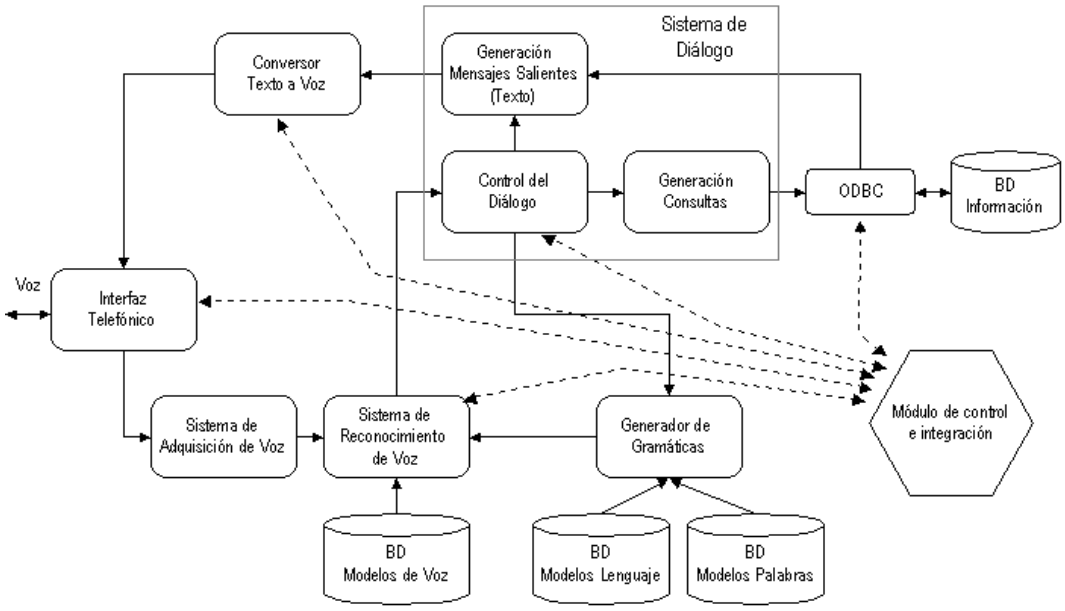


Fig. 4: Diagrama de bloques del sistema de consulta

El sistema se encuentra organizado en 4 bloques principales: sistema de control de diálogo, sistema de reconocimiento, conversor texto-a-voz e interfaz de E/S de voz. El intercambio de información entre los diferentes sistemas así como la activación de los mismos se gestiona mediante un módulo de control global e integración, que, en consecuencia, será el encargado de ir activando los módulos necesarios en cada estado del sistema global.

A diferencia del sistema STACC, en éste existe un módulo dedicado específicamente a la gestión del diálogo con éste que, por tanto, irá guiando la interacción con el usuario y evaluando las respuestas probables a partir de un modelo de diálogo. Este módulo constituye, en cierta medida, el núcleo del sistema, ya que debe interaccionar directamente tanto con el conversor texto a voz, indicándole la frase que debe ser emitida, como con el sistema de reconocimiento, al que debe proporcionar un modelo de lenguaje que corresponda a la estimación realizada sobre las posibles respuestas del usuario.

Evidentemente, esta nueva arquitectura, si bien responde al objetivo inicial de mejorar la naturalidad de la interacción con el usuario, introduce un mayor grado de complejidad en la actuación, tanto del sistema global, como de cada uno de los módulos individuales. Así, por ejemplo, el sistema de reconocimiento debe pasar de ser capaz de reconocer una opción concreta de entre un conjunto preestablecido a reconocer una frase a partir de un modelo de gramática que únicamente contempla probabilidades de aparición de palabras.

3 Módulo de reconocimiento

El módulo de reconocimiento de voz utiliza una librería de funciones desarrollada por el Grupo de Investigación en Procesamiento de Señales y Comunicaciones de la Universidad de Granada [5] [6]. Esta librería permite implementar sistemas de reconocimiento de voz continua (CSR, *Continuous Speech Recognition*) mediante la utilización del modelado oculto de Markov [7] combinado con modelos de lenguaje que permiten establecer y comprobar hipótesis acerca de las frases a reconocer.

En esta aproximación, el proceso de reconocimiento puede ser descrito como el cálculo de la probabilidad P(W|A) de que una frase W corresponda a la señal acústica A, sobre todo el conjunto de frases posibles, a fin de encontrar la que proporciona el valor máximo. De esta forma, el reconocimiento de una frase consistirá en la selección de la secuencia de palabras $\hat{W}$ que cumpla

$$P(\hat{W} \mid A) = \max_{W} P(W \mid A)$$

La probabilidad P(W|A) no puede ser evaluada directamente, aunque, utilizando la regla de Bayes, puede ser calculada como

$$P(W \mid A) = \frac{P(W)P(A \mid W)}{P(A)}$$

donde P(W) es la probabilidad de la frase W, P(A|W) es la probabilidad de la señal acústica, A, dada la frase W, y P(A) es la probabilidad de la señal acústica. El reconocimiento de voz puede ser dividido, por tanto, en dos fases: evaluación de la evidencia acústica y evaluación de la probabilidad de emisión de la frase. De esta forma, es necesario considerar dos modelos: el modelo acústico, determinado por P(A|W), y el modelo de lenguaje, descrito por P(W). Para el modelado acústico se utilizan los modelos ocultos de Markov (HMM), mientras que para el modelado de lenguaje es habitual considerar modelos denominados bigramáticas. Una bigramática es una gramática estocástica de estados finitos en la que se considera que la probabilidad de producción de una palabra depende únicamente de la palabra emitida anteriormente. Por tanto, la bigramática queda determinada por el vocabulario (conjunto de palabras aceptadas) y las probabilidades de producción de cada una de las palabras del vocabulario tras cualquiera otra de ellas, $P(r_i \mid r_j)$. La probabilidad de emisión de una frase compuesta por m palabras será, por tanto

$$P(W) = \prod_{i=1}^{m} P(w_i \mid w_{i-1})$$

siendo w_i la palabra i-ésima de la frase.

Resulta evidente que el módulo de reconocimiento debe combinar la información procedente de los modelos acústicos, en forma de modelos HMM, con los modelos de gramática. Por otra parte, dada la gran cantidad de palabras existentes en cualquier idioma, los modelos HMM utilizados corresponden habitualmente a unidades inferiores a la palabra. En nuestro caso, se utilizan fonemas. Para componer los modelos de palabras simplemente se concatenan los modelos fonemáticos correspondientes de acuerdo al vocabulario considerado. Así, una vez entrenados los modelos HMM, éstos permanecen fijos en el sistema. Sin embargo, el modelo de lenguaje, si bien puede ser estimado a partir de un conjunto suficientemente amplio de frases observadas, debería ser variable a fin de adaptarse al flujo de la conversación entablada entre el sistema y el usuario. En consecuencia, el modelo de lenguaje será implementado a partir de un modelo genérico que debe ser modificado por el módulo de diálogo (fig. 4).

4 Módulo de diálogo

Como ya se ha mencionado anteriormente, el módulo de diálogo constituye, en cierta medida, el núcleo del sistema, ya que determina el flujo de la conversación e intenta adaptarse a los requerimientos tanto del sistema como del locutor.

La finalidad básica del módulo de diálogo es la gestión directa de la interacción con el locutor, para lo que necesita tanto las transcripciones realizadas por el módulo de reconocimiento como las funciones del conversor texto a voz. Adicionalmente, dado que el conjunto de respuestas probables depende del estado de diálogo, debe proporcionar información al sistema de reconocimiento, a través de los modelos de gramáticas, acerca de las frases que pudieran ser emitidas. La finalidad de este mecanismo es aumentar la fiabilidad del proceso de reconocimiento al limitar en cierta medida el espacio de búsqueda de la cadena a reconocer.

También es misión del módulo de diálogo la obtención de los datos solicitados por el locutor, tanto directa como indirectamente, así como guiar al sistema y al locutor hacia la consecución de un objetivo concreto, como podría ser la obtención de un dato o la realización de una reserva.

La operación del módulo de diálogo se estructura en torno al establecimiento de un conjunto de acciones [8], cada una de las cuales puede requerir el conocimiento de un conjunto de datos previos. Es misión del sistema de diálogo determinar la acción concreta, p.e. consultar la hora de llegada de un autobús, y requerir del locutor los datos necesarios para poder realizar dicha acción de forma satisfactoria.

La implementación del módulo de diálogo se ha realizado siguiendo una estrategia de iniciativa mixta [9], en la que se utiliza un analizador semántico basado en tramas que permite el procesamiento del mensaje contenido en las frases de los usuarios [10]. Cada trama está compuesta por un conjunto de *slots* (ranuras) que almacenan los datos relevantes de las frases en relación a proporcionar información suficiente al sistema para que realice la acción requerida. Si una frase o interacción con el usuario no proporciona toda la información necesaria, se generarán *slots* vacíos en la trama asociada. Es misión del sistema de diálogo la obtención de los datos adicionales que permitan rellenar estos *slots* y, por tanto, completar la trama. Obviamente, se hace necesario establecer mecanismos de unificación de tramas, que permitan combinar informaciones parciales.

5 Conversión texto a voz

El conversor texto-a-voz tiene, obviamente, la misión de convertir las cadenas de texto proporcionadas por el módulo de diálogo a su forma oral a fin de que puedan ser presentadas al locutor por vía telefónica. Para ello debe disponer de los correspondientes modelos acústicos, que son concatenados de acuerdo a la cadena indicada, y de un modelo de prosodia que permita aplicar la entonación y cadencia correcta a la frase final. El modelo prosódico es especialmente importante, ya que es el que confiere en mayor grado la sensación de naturalidad a la voz sintetizada.

Dado que el estudio de los modelos prosódicos queda fuera de los objetivos del presente trabajo, se ha optado por incluir en el sistema a implementar el

Datos estadísticos	
N. de conversaciones	489
N. de frases	3677
N. de palabras totales	17839
N. de palabras diferentes	1341
Nombres de destinos	1004
N. de elisiones	434

Categoría	Subcategoría	%
Consulta	*Horarios + Destinos* *Precios* *Duración del trayecto* *Existencia de plazas* *Número de teléfono* *Otras*	70
Reserva	Realización Anulación	30
Compra		<1
Quejas		<1
Objetos perdidos		<1
Entrega a domicilio		<1
Otros		<1

Tabla 1: Datos del corpus adquirido.

sintetizador multilingüe de dominio público *Festival,* desarrollado en la Universidad de Edimburgo [11], por proporcionar una calidad aceptable.

6 Implementación del sistema

La implementación práctica del sistema requiere la realización de un conjunto de tareas previas relacionadas con la obtención de los diferentes modelos a utilizar. Las más destacables son:

- **Estudio de conversaciones**. En primer lugar, se ha realizado un proceso de grabación y transcripción de numerosas conversaciones mantenidas por el servicio de atención telefónica de la compañía Alsina Graells, con la finalidad doble de extraer modelos de lenguaje que se adapten a la situación real y de determinar las acciones y datos asociados necesarios (diferentes tipos y plantillas de tramas para el sistema de diálogo). El análisis de dichas transcripciones indica que un gran porcentaje de las conversaciones (en torno al 90%) responden a un modelo simple para la consulta del horario de salida o llegada de autobuses (Fig. 1). Los datos más relevantes se muestran en la tabla 1.

- **Modelado acústico**. En segundo lugar, es necesario establecer los modelos acústicos a utilizar en el reconocimiento (modelos semicontinuos de Markov fonemáticos). Para ello se está procediendo a la adaptación de los modelos empleados en el sistema STACC, ya que no responden adecuadamente a las nuevas condiciones de utilización debido a factores como diferencias en el sistema de adquisición, niveles de ruido, etc. La obtención de los modelos se ha realizado a partir de las señales de voz preenfatizadas y segmentadas en tramas de 30 ms. con solapamientos de 10 ms., que son parametrizadas mediante el Cepstrum, la energía y sus respectivas derivadas [7]. El número de componentes empleados en la mezcla es de 1024.

- **Acceso a la base de datos**. Se están desarrollando las herramientas que permitan realizar consultas a la base de datos de la compañía (actualmente en formato dBase3) a través de la red, así como la traducción de las peticiones del sistema de diálogo a un lenguaje de consulta adecuado.

Otro de los aspectos fundamentales del desarrollo del sistema es la integración y adaptación de los diferentes módulos que la componen. Aunque algunos de dichos módulos estaban ya presentes en el sistema STACC, deben ser adaptados al nuevo sistema e integrados adecuadamente para que operen de forma conjunta con los restantes módulos. Además, es necesario dotarlos de nuevas funcionalidades y mayor grado de fiabilidad, dada la mayor complejidad de la tarea a abordar.

A modo de ejemplo, el sistema de reconocimiento de voz continua operaba en el sistema STACC con modelos de lenguaje que respondían a gramáticas de estados finitos no estocásticas. En el nuevo sistema ha de utilizar bigramáticas, por lo que ha sido necesario adaptar los mecanismos de gestión del modelo de lenguaje.

También se esta trabajando en el desarrollo e integración de medidas de confianza de las palabras reconocidas a fin de proporcionar al sistema de diálogo de mayor robustez y tolerancia a fallos en el reconocimiento [12]. Para ello se estima la probabilidad de que cada palabra sea la correcta en cada caso, incluyéndose esta información en las tramas utilizadas por el sistema de diálogo, que actuará en consecuencia.

7 Evaluación

La evaluación del sistema se realizará en tres fases: en el laboratorio, evaluación real con diálogo reducido y evaluación real con diálogo.

La primera de las fases, actualmente en desarrollo, corresponde a la evaluación en condiciones de laboratorio del sistema, fundamentalmente, de los modelos acústicos y de lenguaje obtenidos. Para ello se está utilizando un módulo de diálogo que responde a una gramática de estados finitos, análogo en su operación al empleado en el sistema STACC.

Una vez se de por finalizada esta fase, se procederá a la evaluación del mismo sistema en un entorno real. Se usará el mismo módulo de diálogo que en la fase

anterior, es decir, el sistema obligará al locutor a la elección de una de entre varias opciones. La finalidad de esta fase es comprobar la robustez del sistema de reconocimiento así como evaluar la respuesta de los usuarios ante el uso de un sistema automático.

Finalmente, si las fases anteriores resultan satisfactorias, se procederá a evaluar el sistema completo, incluido el módulo de diálogo.

8. Conclusiones y trabajo futuro

En este artículo se ha descrito la funcionalidad y arquitectura de un sistema de información telefónica mediante comunicación oral que está siendo desarrollado por el Grupo de Investigación en Procesamiento de Señales y Comunicaciones de la Universidad de Granada. Se han señalado los principales problemas a abordar así como las técnicas y sistemas que se están utilizando. Los elementos básicos del sistema son un reconocedor de voz continua, un sistema de diálogo y un conversor texto-a-voz, que deben ser integrados adecuadamente para que trabajen de forma cooperativa.

El desarrollo adecuado del sistema ha requerido del estudio de un número suficiente de conversaciones reales con usuarios a fin de extraer las características relevantes. Una vez modelado el diálogo y evaluado el sistema en laboratorio será necesario realizar la prueba real del sistema en las condiciones normales de operación.

Agradecimientos

Este trabajo está subvencionado por la Comisión Interministerial de Ciencia y Tecnología bajo el proyecto TEL1999-0619.

Referencias

[1] Asoh H., Matsui T., Fry J., Asano F. Hayamizu S. “A Spoken Dialog System for a Mobile Office Robot”, Eurospeech ’99, pp. 1139-1142.

[2] L. Bell, J. Gustafson, “Interaction with an Animated Agent in a Spoken Dialogue System”, Eurospeech ’99, pp. 1143-1146.

[3] O. Grisvard, B. Gaiffe, “An Event-Based Dialogue Model and its Implementation in MultiDial2”, Eurospeech ’99, pp- 1155-1158.

[4] A. Rubio, P. García, A. De la Torre, J. C. Segura, J. Díaz-Verdejo, M. C. Benítez, V. Sánchez, A. M. Peinado, J.M. López-Soler, J.L. Pérez-Córdoba. “STACC: an automatic service for information access using continuous speech recognition through telephone line”. Proc. Of EUROSPEECH-97, vol. 4. pp. 1779-1782. Septiembre, 1997.

[5] J. Díaz. “Reconocimiento de voz continua mediante una aproximación híbrida basada en SLMM”. Tesis doctoral. Universidad de Granada. Noviembre, 1995.

[6] P. García. “Reconcomiento de voz continua basada en técnicas MVQHMM”. Tesis doctoral. Universidad de Granada. Febrero, 1996.

[7] L. Rabiner, B. Juang. “Fundamentals of Speech Recognition”. Signal Processing Series. Prentice Hall, 1995.

[8] R. López-Cozar, P. García, J. Díaz, A. Rubio. “A voice activated dialog system for fast-food restaurant applications”. EUROSPEECH-97, vol. 4, pp- 1783-86. Septiembre, 1997.

[9] Rosset S., Bennacef S., Lamel L., “Design Strategies for Spoken Language Dialog Systems”, Eurospeech ’99, pp. 1535-1538.

[10] J. Allen, “Natural Language Understanding”, Benjamin/Cummings Publishing Company Inc. 1995.

[11] A. Black, P. Taylor, R. Caley. “The Festival Speech Synthesis System”.

[12] Mazin Rahim, “Utterance verification for the numeric language in a natural spoken dialogue”, Eurospeech ’99, pp. 57-60.

Modelado de Servicios Complejos en una Plataforma de Intermediación para Comercio Electrónico

Enrique Vázquez[1], Francisco Valera[2], Luis Bellido[1]
[1]Universidad Politécnica de Madrid, [2]Universidad Carlos III de Madrid
ETS Ing. Telecomunicación, Ciudad Universitaria, 28040 Madrid
Teléfono: 913 367 330 Fax: 913 367 333
E-mail: enrique@dit.upm.es

Abstract. *This paper describes a brokering platform for the provision of complex services over the Internet. The platform is able to analyse a request for a complex service, divide it into simple components, and combine several services and products, offered by different providers, into a solution that satisfies the complex request. The platform helps the user to refine or change the request until an suitable solution is found. After the user agreement, it confirms the transaction to the selected providers. Overall, the client can buy a complex service, involving several suppliers, in the same way as a simple one. The main innovations of the platform are the use of an e-commerce ontology to handle the static and dynamic aspects of complex services in different business areas, and the control of the transactional properties of complex services. The system exploits other advanced technologies, such as multi-device publishing based on XML, mobile agents, the Enterprise JavaBeans architecture, and the Java 2 Platform, Enterprise Edition (J2EE).*

1 Introducción

La prestación de servicios complejos que requieren la coordinación de varios proveedores es una tarea difícil, especialmente si se quiere realizar en un sistema de intermediación que negocie automáticamente con servidores de comercio electrónico a través de Internet. Este es el objetivo del proyecto Smart-EC* (Support for Mediation And bRokering for Electronic Commerce) [1], perteneciente al programa europeo de Tecnologías de la Sociedad de la Información IST.

En Abril de 2001, el proyecto presentó un primer prototipo del sistema que muestra sus funciones principales en un escenario de servicios avanzados en el sector editorial, por ejemplo imprenta electrónica, creación de libros con contenido a la carta, publicaciones *online*, etc. Un segundo prototipo que implementará todas las funciones del sistema y otras adicionales como el acceso desde terminales móviles con WAP está previsto para Marzo de 2002.

Uno de los elementos clave que hacen posible el manejo inteligente de servicios complejos en Smart-EC es el uso de una ontología de comercio electrónico [2]. La ontología describe conceptos y relaciones lógicas que permiten al sistema ayudar al usuario en todas las fases de la prestación del servicio: a) definición de la petición del usuario, b) descomposición del servicio complejo en otros más simples, c) búsqueda de proveedores adecuados para cada uno, y d) ejecución de una transacción en la que se confirma el servicio solicitado a cada uno de los proveedores elegidos, de forma que se asegura al usuario la prestación del servicio complejo que necesita.

La ontología que está desarrollando el proyecto tiene una parte genérica que describe las propiedades y las relaciones lógicas entre conceptos básicos como servicio, atributos, usuario, proveedor, petición, oferta, transacción, etc. y otra parte que describe conceptos específicos de los servicios de sectores determinados. La ontología demostrada en el primer prototipo del sistema describe conceptos relacionados con publicaciones, por ejemplo libro, autor, traductor, diseño de página, tipo de encuadernación, número de páginas, tecnología de impresión, etc.

El presente artículo se centra en el modelo genérico de servicios utilizado en Smart-EC y en su uso para dos de las funciones principales (y más complicadas) del sistema: la descomposición de servicios complejos en simples y la evaluación de cuáles son las ofertas más adecuadas para una determinada petición de usuario.

La sección 2 presenta la arquitectura del sistema Smart-EC. La sección 3 define el modelo genérico de servicios utilizado y la sección 4 describe el escenario de servicios de publicaciones elegido para el primer prototipo. Por último, la sección 5 resume las conclusiones del trabajo realizado y las tareas a realizar en la segunda parte del proyecto.

2 Arquitectura

La Fig. 1 muestra la arquitectura general del sistema Smart-EC [3].

* http://www.telecom.ntua.gr/smartec/

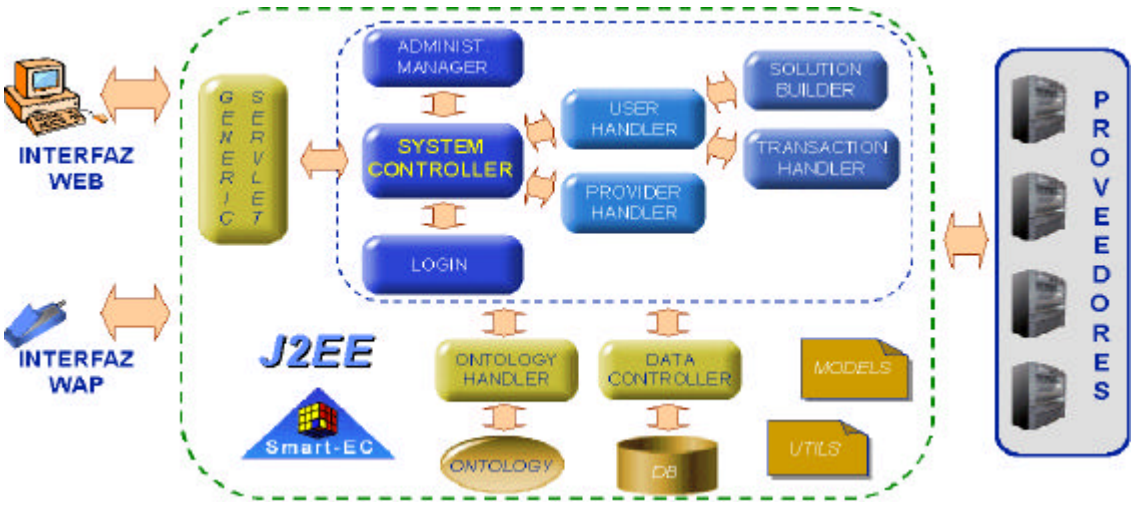


Figura 1: Arquitectura del Sistema Smart-EC

Esta arquitectura está soportada por la plataforma de distribución de aplicaciones Java 2, Enterprise Edition (J2EE) y por las Enterprise Java Beans (EJB), parte fundamental de J2EE [4], que facilitan el desarrollo, integración, implantación, ejecución y posterior reutilización de componentes software [5].

El módulo *Generic Servlet* se encarga de las interfaces de acceso al sistema y se comunica con el módulo *System Controller* mediante mensajes en formato XML (eXtensible Markup Language) [6]. El uso de XML para representar la información, junto con hojas de estilo en XSL (eXtensible Stylesheet Language) [7] para definir el formato con el que la información debe presentarse al usuario, permite que la interfaz del sistema sea genérica y fácilmente configurable para diferentes tipos de terminales, tanto fijos como móviles. La adaptación de la interfaz a un nuevo tipo de terminal se reduce básicamente al diseño de una nueva hoja de estilo en XSL.

Mientras que el módulo *Generic Servlet* se ocupa únicamente de la presentación de la información, el *System Controller* controla los aspectos lógicos del acceso al sistema, esto es, qué opciones se ofrecen al usuario en cada momento en función de las opciones que haya seleccionado anteriormente. La Fig. 2 muestra un diagrama de estados simplificado en UML [8] que ilustra las opciones disponibles.

El módulo *User Handler* guía al usuario en el proceso de petición de servicios. El usuario puede navegar por el catálogo de servicios disponibles, que se le presenta en forma de árbol. A continuación, puede seleccionar un determinado servicio complejo y descender por el árbol para ver sus componentes. Alternativamente, el usuario puede definir su *petición* para el servicio complejo y dejar que el sistema lo descomponga en otros más simples de forma transparente. Para ello el *User Handler* utiliza las funciones del módulo *Ontology Manager*, que da también soporte a otros módulos de la arquitectura, tal como indica la Fig. 1.

El *Provider Handler*, encargado de la comunicación con los proveedores, tiene algunas funciones similares a las del *User Handler*. Por ejemplo, un proveedor puede navegar por el árbol de servicios y registrar *ofertas* para aquellos que le interesen de forma similar a como un usuario navega y selecciona servicios para hacer peticiones. Normalmente el registro de ofertas se hará por procedimientos automáticos entre los sistemas de los proveedores y Smart-EC. En el proyecto se han considerado varios tipos de proveedores con los que Smart-EC será capaz de comunicarse [9].

El módulo *Solution Builder* recibe las peticiones de servicio que ha construido el usuario, ayudado por el *User Handler*, y busca la oferta u ofertas registradas que se ajustan mejor a cada petición. En esencia este proceso se basa en comparar los valores de atributos del servicio que ha definido el usuario en su petición con los valores de atributos que figuran en las diferentes ofertas disponibles para el servicio pedido. Algunos atributos son generales (por ejemplo precio, duración) y otros específicos de cada servicio (por ejemplo, el idioma origen y el destino, el tema del texto, etc. en un servicio de traducción de una publicación).

El proceso de comparación de peticiones y ofertas es bastante flexible, permitiendo por ejemplo la definición de rangos de valores y condiciones para los atributos (por ejemplo valor menor que un máximo dado, valor dentro de un determinado rango, o cualquier valor de una lista dada). Si el usuario lo desea, el sistema puede considerar también ofertas que no cumplen todas las condiciones especificadas en la petición. En este caso, cada oferta se puntúa según su mayor o menor cumplimiento de las condiciones de la petición. También se considera la posibilidad (no implementada en el primer prototipo) de, a la vista de las ofertas disponibles, sugerir al usuario modificaciones en su petición que le permitirían aprovechar ofertas más ventajosas que las que obtendría con su petición original.

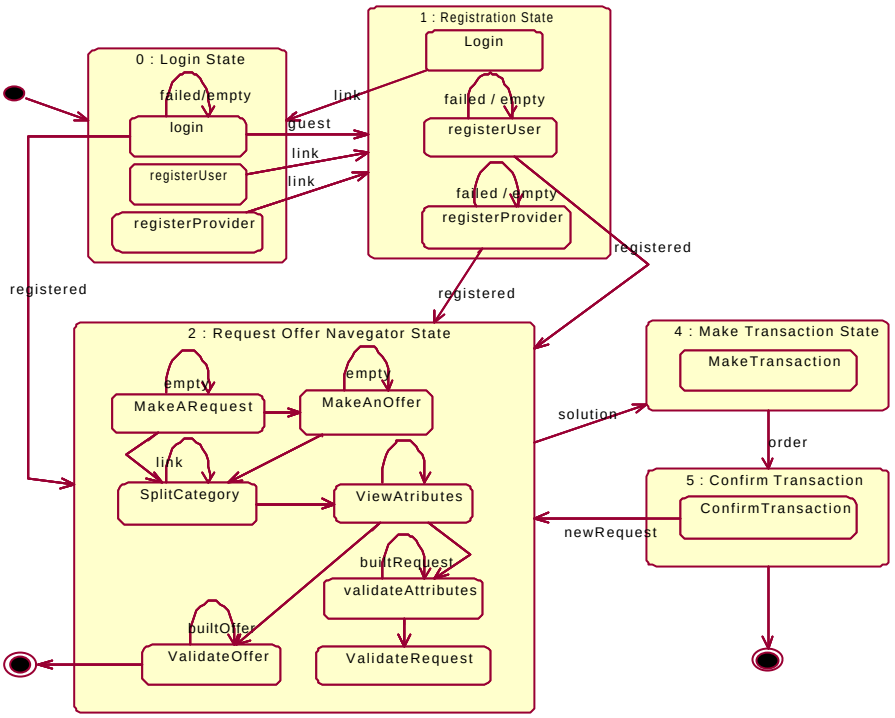


Figura 2: Estados del Controlador del Sistema

Por último, el proceso de selección y valoración de ofertas tiene también en cuenta las preferencias del usuario, bien indicadas explícitamente en su perfil (por ejemplo proveedores preferidos) o aprendidas por el sistema a partir de interacciones anteriores con el mismo usuario.

Una vez seleccionadas las ofertas de servicio más adecuadas, se combinan para formar una o varias *soluciones* que se proponen al usuario. La presentación de las soluciones dependerá de las preferencias del usuario, por ejemplo si se desea una única solución o varias ordenadas de mejor a peor, si se quiere información resumida o detallada de cada solución, etc.

Cuando la solución o soluciones encontradas no son satisfactorias, el usuario puede modificar las condiciones de su petición y repetir el proceso de búsqueda de ofertas. Cuando se obtiene una solución aceptable, si el usuario lo desea, el sistema ejecuta una transacción en la que establece contacto con todos los proveedores involucrados en la prestación del servicio complejo. El módulo *Transaction Handler* se encarga de reservar el servicio de cada proveedor. Si alguna reserva falla, el resto de reservas se cancelan.

La arquitectura del sistema se completa con los módulos *Ontology Manager*, encargado del acceso a la ontología, *Data Controller*, encargado del acceso a la base de datos del sistema (Oracle), *Login*, encargado de verificar la identidad de los usuarios, y *Utils*, que implementa funciones generales. Por último, el módulo *Models* define las estructuras de datos tratadas en la sección siguiente.

3 Modelo de Servicios Complejos

A lo largo de la descripción del sistema Smart-EC presentada en la sección 2 se han introducido informalmente los conceptos básicos que se han utilizado para modelar servicios complejos en Smart-EC. En resumen, se tiene un catálogo de servicios, cada uno de los cuales se define por atributos con un dominio de valores posibles. Un servicio puede descomponerse en servicios más simples recursivamente. Los usuarios definen peticiones de servicio por medio de condiciones que deben cumplir los valores de sus atributos. Por su parte, los proveedores definen ofertas de servicio de la misma forma, es decir, mediante condiciones sobre los valores de los atributos. La petición de un servicio complejo se descompone en peticiones de servicios simples y se buscan ofertas adecuadas para cada una. Las ofertas seleccionadas se puntúan y se combinan para formar una o varias soluciones al servicio complejo demandado. Si se acepta una solución, se ejecuta una transacción entre el sistema y los proveedores de las diferentes ofertas incluidas en la solución que preserva la atomicidad del servicio complejo.

La Fig. 3 representa un conjunto de servicios en forma de árbol. Esto no implica que el catálogo de servicios conocidos por el sistema esté organizado internamente en una estructura de árbol estática. La figura representa solamente la visión de un grupo de servicios que se construye dinámicamente y se va presentando al usuario a medida que éste selecciona servicios. (En el prototipo este árbol se presenta en forma de menús. Ver Fig. 5.)

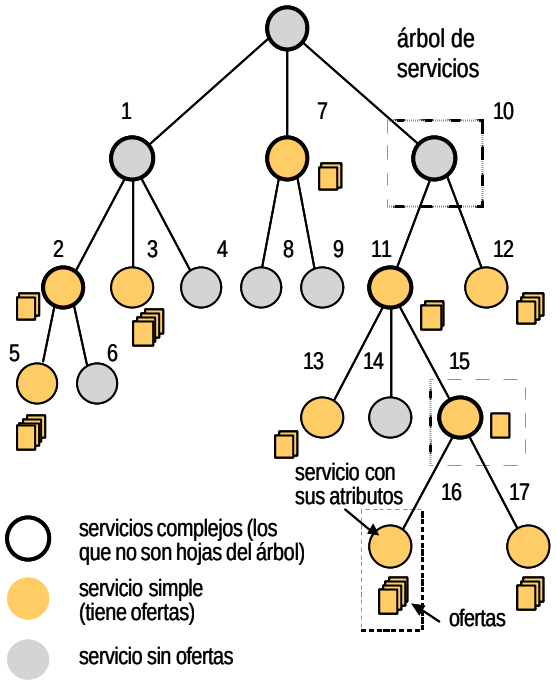


Figura 3. Servicios Complejos y Simples

Todo nodo del árbol, excepto los que son hojas, representa un servicio complejo, ya que puede descomponerse en servicios más simples (los nodos hijos del considerado). Por otra parte todo nodo para el cual existe alguna oferta registrada representa un servicio simple, ya que hay proveedores que ofrecen el servicio como un todo.

Esta manera de definir los tipos de servicios en el modelo de Smart-EC implica que un servicio puede ser simple y complejo a la vez: simple porque alguien lo ofrece como un todo y complejo porque puede descomponerse, si se desea, en otros más simples ofrecidos por diferentes proveedores. Por ejemplo, los servicios 2, 7, 11 y 15 en la figura anterior pueden ser tratados como simples o como complejos. En cambio, los servicios 1 y 10 sólo pueden ser tratados como complejos. El resto de los nodos son servicios simples únicamente (no pueden descomponerse). El nodo raíz del árbol no representa un servicio individual sino una familia de servicios, por ejemplo publicación electrónica, viajes, etc.

3.1 Peticiones

Una petición define un servicio (simple o complejo) deseado por el usuario especificando para cada uno de sus atributos:

- cero o más valores
- una condición en la que intervienen los valores anteriores
- un peso o prioridad

Además, la petición indica:

- cuándo debe prestarse el servicio
- una lista de proveedores preferidos
- el número y tipo de ofertas deseado

Todos los componentes de la petición son opcionales. Si no se da ningún valor para un atributo, significa que cualquier valor es aceptable para el usuario. Las posibles condiciones sobre el valor o valores indicados incluyen “igual”, “distinto”, “mayor”, “menor”, “en rango”, “fuera de rango”, “cualquiera de”, “ninguno de”. Combinando uno o más valores con la condición adecuada, el usuario puede expresar de forma muy flexible las propiedades del servicio que desea. Por ejemplo, para un atributo numérico x el usuario puede especificar

$x = 10$ (valor 10, condición “igual”)
$x \neq 25$ (valor 25, condición “distinto”)
$x = 1$ ó 2 (valores 1 y 2, condición “cualquiera de”)
$x < 50$ (valor 50, condición “menor”)
$1 \leq x \leq 5$ (valores 1 y 5, condición “en rango”)
etc.

De igual forma se pueden definir conjuntos de valores para atributos de tipo enumerado.

Al igual que las peticiones, una oferta para un servicio (ver sección 3.2) especifica valores para los atributos del mismo. Si el conjunto de valores incluidos en una oferta y en una petición tienen intersección no nula, esa oferta *cumple* con la petición para el atributo considerado. Ver ejemplos en Fig. 5 y 6.

En su petición, el usuario dice cuántas ofertas debe buscar el sistema y de qué tipo:

- sólo ofertas *válidas*, esto es aquellas que cumplen con la petición para todos los atributos

o bien

- las ofertas *mejores* que estén disponibles, aunque no cumplan con todos los atributos

El peso de cada atributo indica la importancia relativa que tiene para el usuario el que una oferta cumpla o no con ese atributo. Los pesos son utilizados por el sistema para puntuar las ofertas de mejor a peor para la petición considerada [3]. También se tiene en cuenta la posibilidad de que una petición indique uno o varios proveedores preferidos, por ejemplo un usuario que prefiere volar con una determinada línea aérea.

3.2 Ofertas

Una oferta define un servicio prestado por un proveedor. Como se ha dicho, tanto las ofertas como las peticiones se definen dando valores y condiciones para los atributos del servicio. Por ello, las dos estructuras, oferta y petición, son bastante parecidas en el modelo de servicios de Smart-EC. Una oferta incluye para cada atributo del servicio

- cero o más valores
- una condición sobre los valores anteriores

junto con:

- el periodo de validez de la oferta
- la disponibilidad del servicio ofertado

La posibilidad de usar valores y condiciones para los atributos también en las ofertas permite que un proveedor pueda registrar con una sola oferta varios servicios que se diferencian sólo en los valores de algún atributo, por ejemplo un servicio para el que el proveedor quiere ofrecer varias opciones con el mismo precio, periodo de validez, etc.

Para que una oferta sea válida para una determinada petición, además de cumplir con los valores de atributos pedidos debe verificarse que el momento de prestación del servicio indicado en la petición está dentro del periodo de validez de la oferta.

El campo de disponibilidad indica si, además de las comprobaciones anteriores, es necesario establecer contacto con el proveedor que ha registrado una oferta para comprobar la disponibilidad del servicio. Por ejemplo, una oferta de alojamiento en un hotel incluiría en sus atributos la descripción del hotel, tipos de habitaciones disponibles, precio, etc., lo que permite saber si satisface una determinada petición. Si es así, se necesita además comprobar la disponibilidad de habitaciones en las fechas pedidas por el usuario.

La duración de un servicio es uno más de sus atributos y se comprueba de la misma forma que los demás. Por ejemplo, una oferta de un servicio de encuadernación que especifique

Duración = 3 días

sería válida para una petición que indicara

Duración ≤ 5 días

El sistema permite definir varias unidades diferentes para los valores de un atributo y las funciones de conversión de unas unidades a otras. Así, la oferta anterior también se consideraría válida para otra petición con

Duración ≤ 1 semana

Al igual que la duración de un servicio, su precio es un atributo más: una oferta será válida (por lo que respecta al precio) si indica una cantidad dentro del rango admitido por el usuario en su petición.

Como otros atributos, la duración es opcional y puede no ser necesaria en algunos servicios. En esos casos, basta comprobar que se está dentro del periodo de validez de la oferta como ya se ha mencionado. Si un servicio tiene duración, pero el usuario no incluye una condición para ella en su petición, significa simplemente que acepta ofertas con cualquier duración.

El modelo de Smart-EC está diseñado para servicios, pero la compra de productos puede representarse con él sin dificultad. Un producto se describe con una lista de atributos y pueden registrarse peticiones y ofertas para el mismo. En el caso de un producto no hace falta un atributo "duración" aunque puede existir un "plazo de entrega".

El ejemplo de los 3 y 5 días propuesto antes sería igualmente aplicable al plazo de entrega o al precio (cambiando de unidades) o, de hecho, a cualquier otro atributo.

El modelo de servicios de Smart-EC y el prototipo desarrollado en el proyecto permiten manejar cualquier atributo *sin necesidad de tener en cuenta su significado*. Si una petición indica un conjunto de valores (numéricos o de otro tipo) para un atributo X, el sistema se limita a buscar ofertas en las que figure el mismo atributo X con un conjunto de valores que tenga intersección no nula con el conjunto de la petición (haciendo conversión de unidades si es necesario). La posible mayor importancia para el usuario de unos atributos frente a otros se indica explícitamente con los valores del campo peso o prioridad ya citado.

En resumen, este enfoque permite definir fácilmente nuevos servicios o productos con nuevos atributos e incluirlos en la base de datos del sistema sin necesidad de reprogramar los métodos que comparan peticiones y ofertas. (En la práctica, previsiblemente la facultad de definir nuevos servicios e integrarlos de forma consistente en el catálogo de Smart-EC la tendrá sólo el administrador del sistema. Los usuarios y proveedores deberán limitarse a pedir u ofertar los servicios existentes).

3.3 Soluciones

Una solución es simplemente una combinación de ofertas, en general de diferentes proveedores, que el sistema da como respuesta a la petición de un servicio complejo.

Como ya se ha dicho, un usuario puede pedir al sistema varias soluciones para una misma petición. Si una de ellas es aceptada el sistema inicia una transacción que provee el servicio complejo de forma atómica, esto es, o todos los servicios incluidos en él o ninguno. Los detalles del manejo de transacciones no se tratan aquí pero pueden consultarse en [3,9].

4 Escenario de Demostración

El primer prototipo del sistema Smart-EC se ha demostrado en un escenario de servicios de publicación simples y complejos, definido por una empresa editorial que participa en el proyecto. La jerarquía de servicios considerada se representa esquemáticamente en la Fig. 4. Los atributos de cada servicio, sus posibles valores, unidades, etc. están definidos con detalle en [10].

4.1 La Ontología

La ontología incluida en el prototipo del sistema describe de manera más formal los elementos que intervienen en el escenario seleccionado y las relaciones lógicas que existen entre ellos, por ejemplo qué servicios más simples incluye uno complejo.

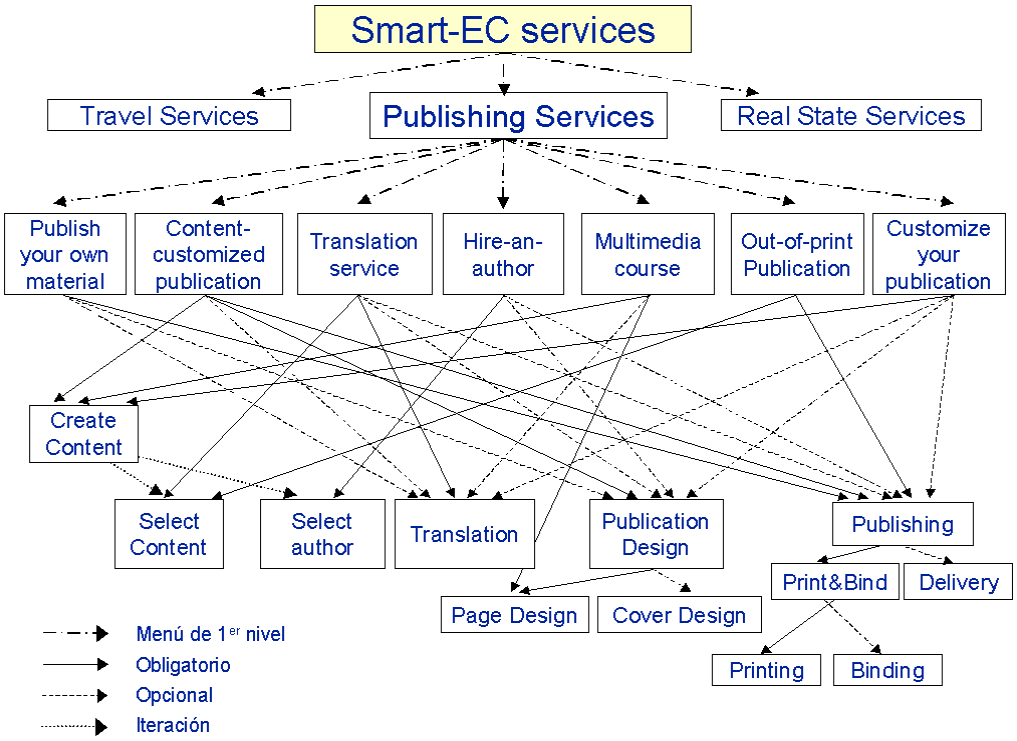


Figura 4: Escenario de Servicios en el Sector Editorial

Las ontologías pueden utilizarse en los sistemas de comercio electrónico para facilitar el entendimiento entre compradores y vendedores o para facilitar la reutilización de información. Algunos buscadores inteligentes usan ontologías para encontrar documentos que contienen otros términos lógicamente relacionados con el término buscado por el usuario (incluso términos y relaciones inicialmente desconocidos por el propio usuario si no es experto en el tema sobre el que busca información) [11].

En Smart-EC, como se ha dicho, el propósito principal de la ontología es poder prestar de manera global servicios complejos construidos a partir de una combinación de servicios más simples obtenidos de diferentes proveedores.

El ejemplo siguiente muestra una pequeña parte de la ontología de Smart-EC, escrita en lenguaje Classic [12]. La parte seleccionada describe los componentes del servicio complejo *Publish your own material* (ver Fig. 4). Este servicio ofrece al usuario la posibilidad de editar sus propios libros a partir del material proporcionado por él, incluyendo servicios de traducción, diseño, imprenta y encuadernación.

```
…
(cl-define-concept 'PUBLISH-YOUR-OWN-MATERIAL
  '(and PUBLISHING-SERVICE
   (at-most 1 translateMaterial)
   (all translateMaterial TRANSLATION)
   (at-most 1 publiDesignMaterial)
   (all publiDesignMaterial PUBLICATION-DESIGN)
   (at-least 1 publishMaterial)(at-most 1 PublishMaterial)
   (all publishMaterial PUBLISHING)))
```

```
(cl-define-concept 'PUBLICATION-DESIGN
   '(and PUBLISHING-SERVICE
    (at-least 1 pageDesignMaterial)
    (at-most 1 pageDesignMaterial)
    (all pageDesignMaterial PAGE-DESIGN)
    (at-most 1 coverDesignMaterial)
    (all coverDesignMaterial COVER-DESIGN)))

(cl-define-concept 'TRANSLATION
  '(and PUBLISHING-SERVICE
   (at-least 1 topic)(at-most 1 topic)(all topic string)
   (at-least 1 publicatCategory)(at-most 1 publicatCategory)
   (all publicatCategory  PUBLICATION)
   (at-least 1 pricePerPage)(at-most 1 pricePerPage)
   (all pricePerPage number)
   (at-least 1 sourceLanguage)(at-most 1 sourceLanguage)
   (all sourceLanguage LANGUAGE)
   (at-least 1 targetLanguage)(at-most 1 targetLanguage)
   (all targetLanguage LANGUAGE)
   (at-least 1 timeToDeliver)(at-most 1 timeToDeliver)
   (all timeToDeliver string )))
...
...
```

Sin entrar en detalles, la primera sección del ejemplo define los componentes de *Publish your own material* y la segunda, los componentes del servicio *Publication Design* que está incluido en el anterior. La tercera sección muestra los atributos del servicio *Translation*: tema del texto a traducir, tipo de publicación, precio por página, idiomas origen y destino y plazo de entrega. Las Fig. 5 y 6 muestran un ejemplo de petición y soluciones para este servicio.

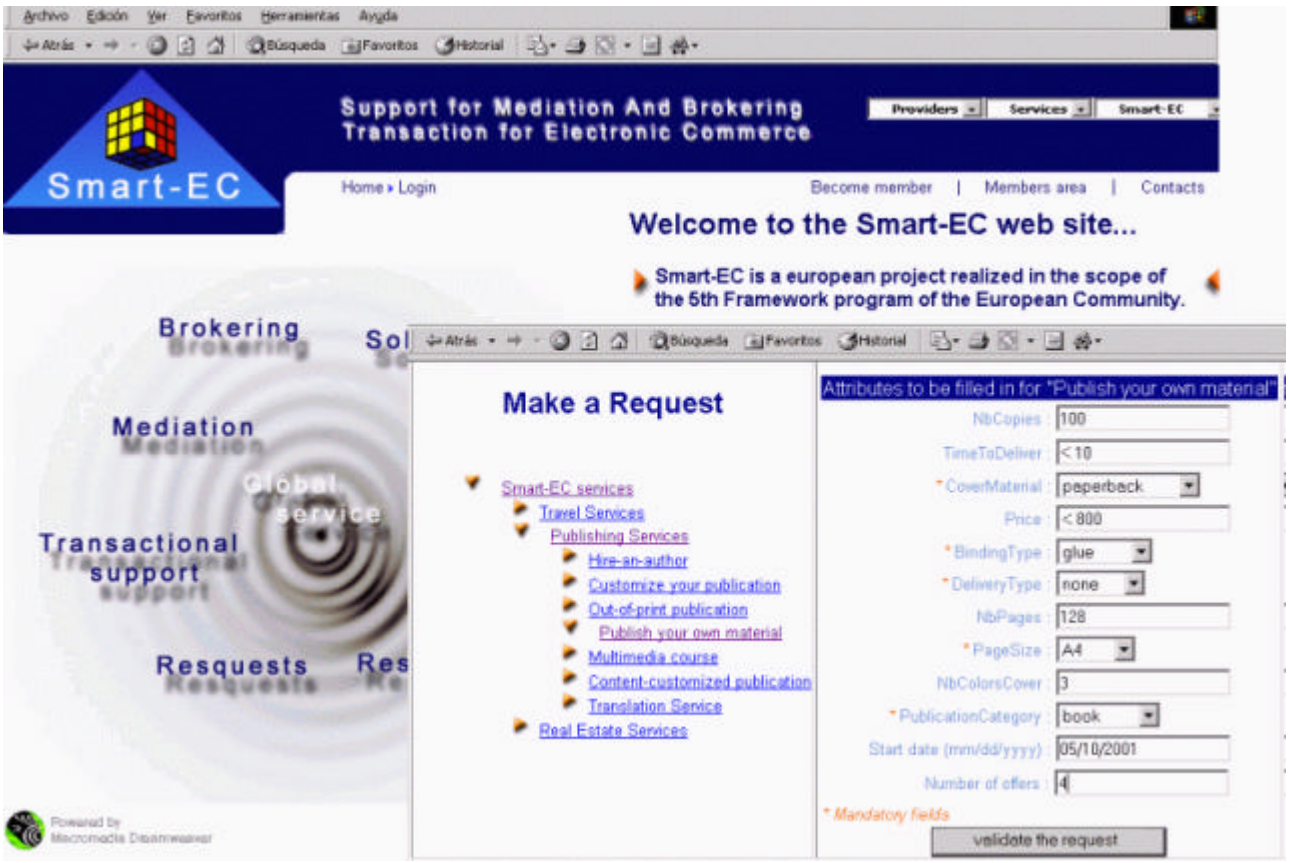


Figura 5: Prototipo de Smart-EC. Ejemplo de Petición de Servicio Complejo

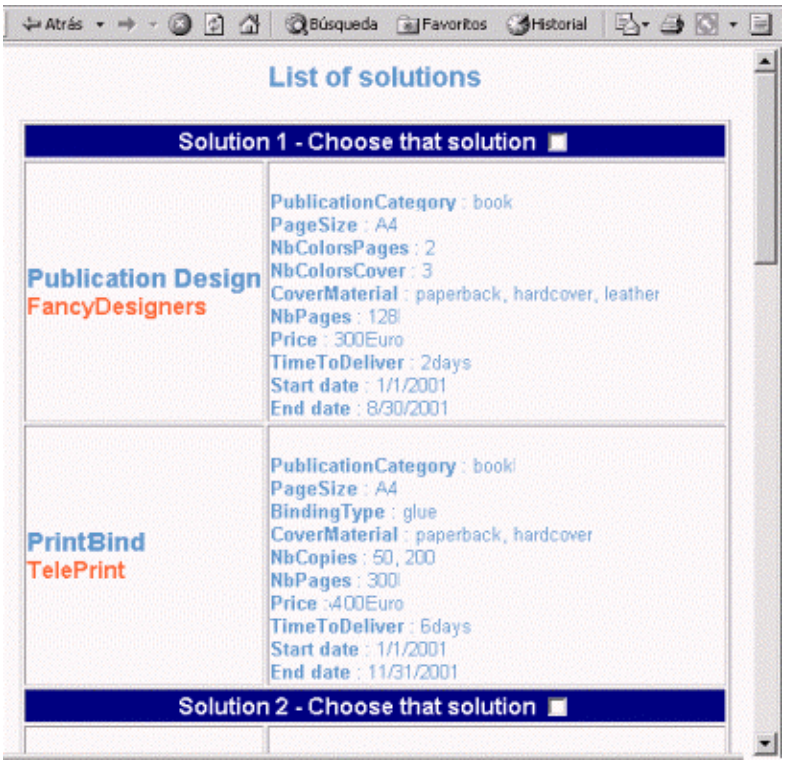


Figura 6: Prototipo de Smart-EC. Ejemplo de Solución Propuesta

5 Conclusiones

Este artículo ha descrito los objetivos y el diseño de una plataforma de intermediación de comercio electrónico capaz de prestar servicios complejos. Los intermediarios que existen en Internet se limitan en general a la búsqueda de productos o de servicios simples como, por ejemplo, un viaje de avión. Smart-EC es capaz también de buscar proveedores de diferentes servicios y combinarlos para satisfacer de forma global un servicio complejo, por ejemplo un viaje de negocios que incluya avión, hotel, etc. Por ejemplo, en lugar de limitarse a contestar "hay plazas para volar a Barcelona el día ...", Smart-EC puede guiar al usuario en su petición de servicio e incluso proponer alternativas mejores, por ejemplo "hay plazas en el día pedido, pero volando un día antes tendría una tarifa más barata".

Estas tareas tienen que hacerse normalmente a mano por una persona con suficientes conocimientos sobre el tipo de servicios de que se trate. El proyecto Smart-EC pretende automatizarlas, representando el conocimiento necesario para ello en una ontología incluida en la plataforma de intermediación.

Recientemente se ha probado un primer prototipo con éxito, aunque con funciones todavía muy limitadas. En la segunda parte del proyecto se ampliará la ontología y se mejorará el prototipo en cuanto a acceso desde diferentes tipos de terminales fijos y móviles, prestaciones y fiabilidad.

Por sus objetivos y enfoque, el proyecto Smart-EC se enmarca en las tendencias más recientes hacia una *Web semántica*[1] en la que la información tenga un significado bien definido con el fin de facilitar su tratamiento automático y mejorar la interacción entre personas y ordenadores.

Agradecimientos

Los miembros que forman el proyecto Smart-EC son: SEMA Group (España), Universidad Politécnica de Madrid (España), National Technical University of Athens (Grecia), Laboratoire d'Informatique, Robotique, et Microélectronique de Montpellier (Francia), France Télécom R&D (Francia), Tradezone International Ltd. (Reino Unido), Anaya Multimedia (España) y AQL (Francia).

Referencias

[1] P. Le Louët, E. Vázquez, F. Valera. "Smart-EC: Making Complex e-Commerce Services Simple". Enviado a la eBusiness and eWork Conference e-2001, Venecia, Italia, 17-19 Octubre 2001. http://www.ebew.net/

[2] M. Uschold, M. Gruninger. "Ontologies: Principles, Methods and Applications". Knowledge Engineering Review, Vol.11, Num.2, Junio 1996. http://www.aiai.ed.ac.uk/publications/tr96.html

[3] Smart-EC Project Deliverable 5.5.1. "Smart-EC Manual version 1". Abril 2001. http://www.telecom.ntua.gr/smartec/

[4] N. Kassem. "Designing Enterprise Applications with the Java™ 2 Platform, Enterprise Edition". Addison-Wesley. 2000. ISBN 0-201-70277-0. http://cseng.aw.com/book/0,,0201702770,00.html

[5] F. Valera, E. Vázquez, L. Bellido. "Design and Implementation Experiences with Distributed e-Commerce Brokering Systems". Enviado a la eBusiness and eWork Conference e-2001, Venecia, Italia, 17-19 Octubre 2001. http://www.ebew.net/

[6] A. Bergholz. "An XML Tutorial". IEEE Internet Computing, Vol. 4, Num. 4, Julio-Agosto 2000. http://computer.org/internet/ Ver también http://www.w3.org/XML/

[7] "Extensible Stylesheet Language (XSL)". http://www.w3.org/Style/XSL/

[8] G. Booch, J. Rumbaugh, I. Jacobson. "The Unified Modeling Language User Guide". Addison-Wesley. 1999. ISBN 0-201-57168-4. http://cseng.awl.com/book/0,,0201571684,00.html

[9] Smart-EC Project Deliverable 4.1. "Smart-EC Architecture version 1". Abril 2001. http://www.telecom.ntua.gr/smartec/

[10] A. Ruiz-Andino, E. Vázquez. "Publishing Services Demonstration". Marzo 2001. http://www.telecom.ntua.gr/smartec/

[11] S. Debnath, S. Sen, B. Blackstock. "LawBot: A Multiagent Assistant for Legal Research". IEEE Internet Computing, Vol. 4, Num. 6,Noviembre-Diciembre 2000. http://computer.org/internet/

[12] A. Borgida. "CLASSIC: A Structural Data Model for Objects". Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data, Portland, Oregon, 1989.

[1] http://www.w3.org/2001/sw/

Análisis y Dimensionado de Sistemas de Pérdidas Multiservicio

Jorge Martínez y Vicente Casares
Dpto. de Comunicaciones. Universidad Politécnica de Valencia.
Camino de Vera s/n. 46022 – Valencia.
Teléfono: 96 387 7767 Fax: 96 387 7309
E-mail: jmartinez,vcasares@upvnet.upv.es

Abstract*. The analysis of multiservice lost-call-cleared systems are attracting new interest because they can be applied to evaluate the performance of multiservice systems like ATM, UMTS, LMDS, etc. This paper describes an extension of a previous analysis technique proposed by Iversen [3], by which the exact solution of systems with access control can be obtained. The analysis technique can be applied to systems in which the arrival processes must be state dependent Poisson processes. The extension allows to analyse systems in which the different services have dedicated resources allocated to them and, when exhausted, can compete for a pool of common resources. The technique is simple to understand and the performance parameters of interest can be obtain by programs of low computational complexity.*

1 Introducción.

El sistema estudiado es un caso particular de un sistema de pérdidas (*lost-call-cleared system*) más general, que se denomina multidimensional, heterogéneo y multitasa, o simplemente multiservicio. Se denomina multidimensional puesto que varias fuentes o servicios compiten por un conjunto limitado de canales, heterogéneo puesto que las distribuciones de las llamadas generadas por los diferentes servicios pueden ser diferentes y multitasa puesto que las llamadas de cada servicio requieren la asignación de uno o varios canales para ser cursadas.

El objetivo final es la definición de una técnica de análisis y dimensionado de sistemas de pérdidas multiservicio con población finita que incorporen control de acceso.

Por análisis se entiende la determinación del valor exacto de los parámetros típicos que definen las prestaciones del sistema, como son: la Probabilidad de Bloqueo (PB) (*time congestion*), la Probabilidad de Pérdidas (PP) (*call congestion*), la Intensidad de Tráfico Ofrecido (TO), la Intensidad de Tráfico Cursado (TC), la Intensidad de Tráfico Perdido (TP) y la Congestión de Tráfico (CT) (*traffic congestion*). Todos estos parámetros se definen posteriormente.

Por dimensionado se entiende la determinación del número máximo de individuos que pueden estar asociados a cada uno de los servicios, para que el sistema en su conjunto cumpla unos objetivos de grado de servicio (GOS, *Grade of Service*) dados.

La técnica de análisis en la que se basa este estudio fue propuesta por Iversen en [3] y puede ser aplicada a un conjunto muy extenso de sistemas reales. La única hipótesis restrictiva que se impone es que el proceso de llegadas debe ser de Poisson, pudiendo su tasa variar con el estado del sistema. En particular, se ha utilizado con éxito para el análisis de sistemas multiservicio BPP (*Bernouilli-Poisson-Pascal*), aunque en este documento sólo se describe su aplicación para servicios de población finita o de Bernouilli.

2 Trabajos Previos

Son bien conocidos los trabajos pioneros de Kaufman [5] y Roberts [7]. Desafortunadamente sus modelos sólo permiten abordar el estudio de sistemas multiservicio con fuentes de Poisson exclusivamente.

Delbrouch en [1] propone una técnica de análisis de sistemas multiservicio BPP que permite determinar la PB de forma exacta, aunque la PP sólo puede determinarse de forma aproximada. Es interesante resaltar que con la aproximación propuesta en [1] para el cálculo de la PP se obtiene un valor de ésta ligeramente superior al real, aunque es una buena aproximación para la Congestión de Tráfico [3].

Diferentes trabajos como los de Nilsson, Perry, Gersht e Iversen en [6] proponen métodos de cálculo iterativos que eliminan las potenciales inestabilidades de los métodos de cálculo propuestos anteriormente.

3 Descripción del Sistema Bajo Estudio

La Fig. 1a describe el escenario general del sistema bajo estudio. Como se observa, un conjunto de R fuentes o servicios compiten por un conjunto

limitado de canales C. Cada una de las fuentes representa el proceso de generación de llamadas asociado a un servicio determinado. La fuente r se caracteriza por los siguientes parámetros:

1. $c_r \;;\; 0 \le r \le R$

   El número de canales necesarios para cursar una llamada.

2. $n_r \;;\; 0 \le r \le R$

   El número máximo de llamadas simultáneas. Se debe cumplir que $n_r \cdot c_r \le C$.

3. $m_r \;;\; 0 \le r \le R$

   La duración media de una llamada del servicio r, siendo $m_r = 1/\mu_r$. Aunque se asumirá que la duración de las llamadas está distribuida exponencialmente, los modelos considerados son insensibles al tipo de distribución, siendo sólo relevante su valor medio [2].

4. $\lambda_r(i) \;\; 0 \le r \le R \;;\; 0 \le i \le n_r \cdot c_r$

   La tasa de generación de llamadas definidas sobre el sistema de la Fig. 1b. En general será función del estado del sistema.

5. $\{p_r(i)\} \;\; 0 \le r \le R \;;\; 0 \le i \le n_r \cdot c_r$

   Las probabilidades de estado en régimen permanente definidas sobre el sistema de la Fig. 1b.

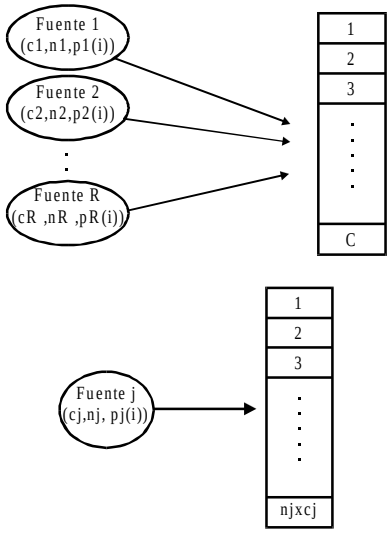


Figura 1. a) Sistema de pérdidas multiservicio. b) Sistema sobre el que se define el tráfico ofrecido por el servicio j.

Como se observa en la Fig. 1b, las probabilidades de estado asociadas a cada fuente se definen sobre un sistema en el que únicamente esa fuente accede a un conjunto de canales limitado a $n_r \cdot c_r \le C$. Como es inmediato observar, las $\{p_r(i)\}$ sólo están definidas para i múltiplo de $\cdot c_r$ y se determinan mediante la siguiente expresión:

$$p_r(k \cdot c_r) = p_r(0) \frac{\prod_{j=0}^{k-1} \lambda_r(jc_r)}{k! \cdot \mu_r^k} \qquad 0 \le k \le n_r \tag{1}$$

Sólo los valores relativos de las $\{p_r(i)\}$ son relevantes para obtener los parámetros de tráfico que definen las prestaciones del sistema, por lo que se puede fijar $p_r(0) = 1$. No obstante, es conveniente normalizar las $\{p_r(i)\}$ de forma que

$$\sum_{k=0}^{n_i} p_r(k \cdot c_r) = 1$$

4 Método de Análisis

Dados dos vectores de probabilidades de estado $x = \{x(0), \ldots x(a_x)\}$ e $y = \{y(0), \ldots y(a_y)\}$, se define el operador convolución (*) como aquel que aplicado a estos dos vectores da como resultado el vector z con los siguientes elementos:

$$z(i) = \sum_{j=u(i)}^{v(i)} x(i-j) \cdot y(j) \;\; ; \;\; 0 \le i \le a_z$$

estando las funciones $u(i)$ y $v(i)$ definidas de la siguiente forma:

$$u(i) = \begin{cases} 0, & 0 \le i < a_x \\ i - a_x & a_x \le i \le a_z \end{cases}$$

$$v(i) = \begin{cases} i, & 0 \le i < a_y \\ a_y & a_y \le i \le a_z \end{cases}$$

Para determinar el valor de los diferentes parámetros de tráfico que definen las prestaciones del sistema se sigue el siguiente proceso:

1. Por sucesivas convoluciones de los vectores probabilidad de estado normalizados $\{p_r(i)\}$ de todas las fuentes, en cualquier orden, se obtendrá el vector
   $$q_N = \{q_N(0), \ldots, q_N(C)\}$$
   que representa las probabilidades de estado del sistema de la Fig. 1a.
2. Para cada una de las fuentes, por ejemplo la r, se de-convoluciona q_N en dos vectores $q_{N/r}$ y p_r, de forma que
   $$q_N = q_{N/r} * p_r$$

La complejidad computacional asociada a la realización de las diferentes convoluciones puede reducirse utilizando el algoritmo propuesto en [4].

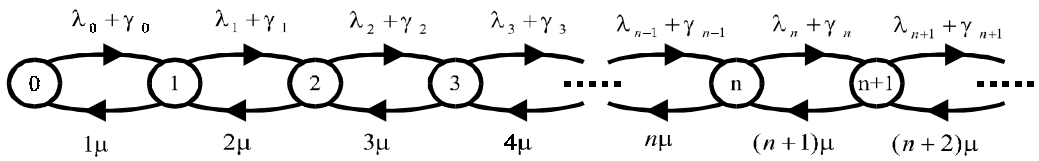


Figura 2. Diagrama de estados para los sistemas BPP.

Los diferentes parámetros que definen las prestaciones del sistema pueden ser ahora obtenidos. La PB para la fuente r se determina mediante la siguiente expresión:

$$PB_r = \frac{1}{N_z}\left\{ p_r(n_r c_r) \sum_{i=0}^{C-n_r c_r} q_{N/r}(i) + \sum_{j=C-c_r+1}^{C} \sum_{i=0}^{(n_r-1)c_r} q_{N/r}(j-i)\, p_r(i) \right\} \quad (2)$$

siendo $N_z = \sum_{i=0}^{C} q_N(i)$.

Como se observa en la expresión (2), una nueva llamada del servicio r quedará bloqueada en los siguientes casos: i) cuando el número de llamadas en curso del servicio r sea igual a n_r; ii) cuando el número de canales libres no es suficiente para cursar una nueva llamada del servicio r.

La PP para la fuente r se determina mediante la siguiente expresión:

$$PP_r = \frac{1}{\sum_{j=0}^{C} \sum_{i=0}^{j} q_{N/r}(j-i)\, p_r(i) \lambda_r(i)} \left\{ p_r(n_r c_r) \lambda_r(n_r c_r) \sum_{i=0}^{C-n_r c_r} q_{N/r}(i) + \sum_{j=C-c_r+1}^{C} \sum_{i=0}^{(n_r-1)c_r} q_{N/r}(j-i)\, p_r(i) \lambda_r(i) \right\} \quad (3)$$

Como se observa en la expresión (3), la PP del servicio r se obtiene como la relación entre las llamadas perdidas y las ofrecidas.

El TO por una fuente se define como el tráfico que cursaría un sistema con infinitos recursos al ser atacado por dicha fuente. Por tanto, el TO por la fuente r, expresado en canales, vale:

$$TO_r = \sum_{i=0}^{n_r c_r} i \cdot p_r(i) \quad (4)$$

El TC por la fuente r, expresado en canales, se obtiene mediante la siguiente expresión:

$$TC_r = \frac{1}{N_z}\left[\sum_{j=0}^{C} \sum_{i=0}^{j} i \cdot q_{N/r}(j-i) \cdot p_r(i) \right] \quad (5)$$

La congestión de tráfico de la fuente r se obtiene mediante la siguiente expresión:

$$CT_r = \frac{TO_r - TC_r}{TO_r} \quad (6)$$

5 Fuentes Bernouilli-Poisson-Pascal (BPP)

Las fuentes BPP son sistemas de nacimiento y muerte con un diagrama de estados como el de la Fig. 2. En las siguientes secciones se definen las tasas de nacimiento para los diferentes procesos de llegada.

5.1 Poisson

$$\lambda_n = \lambda \;; \quad \gamma_n = 0$$

Cuando el número de canales C es finito, la distribución de Poisson truncada se denomina de Erlang-B.

5.2 Bernouilli (binomial positiva)

$$\lambda_n = M\lambda' \;; \quad \gamma_n = -n\lambda'$$

siendo M el tamaño de la población. En este caso, es importante destacar que, aunque los recursos sean infinitos, el número de estados es finito e igual a M+1. Cuando se verifica que C<M, la distribución de Bernouilli truncada se denomina de Engset-B.

Esta distribución se utiliza para modelar los sistemas de población finita. Una de las fuentes de la Fig. 1a es finita cuando se concibe como un

conjunto finito de individuos M, que cuando están en reposo (*idle*), es decir, no están conversando, generan llamadas de forma individual a tasa constante λ'. Por tanto, cada uno de estos individuos puede representarse mediante el diagrama de estados de la Fig. 3.

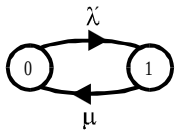


Figura 3. Diagrama de estados de un individuo en una fuente finita.

Las probabilidades de estado en régimen permanente para el sistema de la Fig. 3 vienen dadas por $p(0) = \mu/(\lambda'+\mu)$ y $p(1) = \lambda'/(\lambda'+\mu)$. Se denotará por $p = p(1)$ y $q = p(0) = (1-p)$.

Se considerará que p es el **tráfico ofrecido por individuo** expresado en llamadas. Las probabilidades de estado de la fuente r definidas en la expresión (1) se pueden expresar en función del tráfico ofrecido por individuo de la siguiente forma:

$$p_r(k \cdot c_r) = p_r(0) \frac{\prod_{j=0}^{k-1} \lambda_r(jc_r)}{k! \cdot \mu_r^k} =$$

$$p_r(0) \left(\frac{p_r}{1-p_r} \right)^k \frac{\prod_{j=0}^{k-1} (M_r - j)}{k!} \qquad 0 \le k \le n_r$$

Para este sistema, denotaremos por X a la variable aleatoria que define el número de canales simultáneamente ocupados e Y a la variable aleatoria que define el número llamadas en curso. Entonces la probabilidad del evento $\{Y = n\}$ viene dada por

$$p(n) = \binom{M}{n} p^n (1-p)^{M-n} \qquad n = 1,2,\ldots M$$

es decir, una distribución binomial. La función generadora de probabilidades (*probability generating function*, pgf) de esta distribución discreta de probabilidades viene dada por

$$P_Y(s) = \sum_{n=0}^{M} p(n) s^n = (ps+q)^M$$

Su momento de primer orden $E[Y]$ se pueden obtener mediante la siguiente expresión:

$$E[Y] = \left. \frac{dP(s)}{ds} \right|_{s=1} = P'(s=1) = Mp$$

La variable aleatoria X sólo puede tomar valores $n \cdot c$ siendo n un número natural y c el número de canales por llamada. Además, X está directamente relacionada con Y, puesto que las probabilidades de los eventos $\{Y = n\}$ e $\{X = n \cdot c\}$ son iguales.

La pgf de X se puede obtener mediante la siguiente expresión

$$P_X(s) = \sum_{n=0}^{M} p(nc) s^{nc} = (ps^c + q)^M$$

Por tanto, el tráfico ofrecido por la fuente en su conjunto TO, viene expresado por:

$$TO = \left. \frac{dP_X(s)}{ds} \right|_{s=1} = P'(s=1) = Mpc$$

5.3 Pascal (binomial negativa)

$$\lambda_n = \lambda(1-\theta) \; ; \quad \gamma_n = n\theta\mu$$

En los estudios clásicos de telefonía, la distribución de Pascal se ha utilizado para modelar el tráfico que desborda de un enlace de N canales sobre una ruta secundaria de capacidad infinita, tal y como se observa en la Fig. 4. El efecto de desbordamiento buscado se consigue haciendo que la tasa de nacimiento aumente con el estado del sistema.

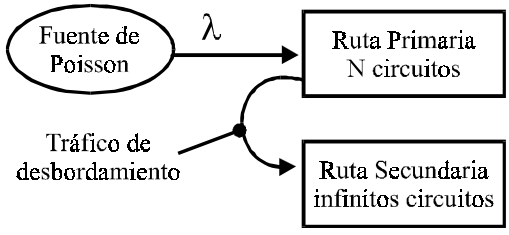


Figura 4. Concepto de tráfico de desbordamiento.

6. Dimensionado del Sistema

El objetivo del dimensionado es obtener la máxima intensidad de tráfico que puede ofrecerse (TO) por servicio para conseguir un GOS determinado.

El GOS se puede definir de diferentes formas, en este trabajo se han considerado sólamente dos: i) el GOS medio, definido como la media aritmética de las PP_r; ii) el GOS del servicio más prioritario, definido como la PP_r del servicio más prioritario. En algunos casos, el dimensionado consiste en

conseguir que simultáneamente el GOS medio y el GOS del servicio más prioritario no superen ciertas cotas.

Los parámetros de entrada al algoritmo de dimensionado son los siguientes: i) C, el número total de canales disponibles; ii) R, el número de servicios; iii) c_r, el número de canales necesarios para cursar una llamada de cada uno de los servicios; iv) p_r, el tráfico ofrecido por un usuario de cada servicio; v) el vector de penetración de los servicios, el cual define el porcentaje de usuarios asociados con cada servicio; vi) n_r, las restricciones impuestas por el control de acceso; vii) el proceso de llegada de llamadas (población finita o infinita); viii) El GOS deseado. Los valores de los parámetros deben ser conocidos por el proveedor del servicio, bien debido a experiencia previa o bien debido a predicciones de servicio.

El algoritmo de dimensionado es sencillo y comienza por definir el intervalo en el que se realizará la búsqueda del valor óptimo para el TO agregado. Un intervalo posible podría ser [0,C]. A continuación se elige el punto medio del intervalo C/2, se reparte el TO agregado entre los diferentes servicios según el vector penetración, se obtiene el PP_r de cada servicio y se verifica si se ha conseguido el GOS deseado. Si el GOS obtenido es superior al deseado, el nuevo intervalo de búsqueda será [0,C/2]. En caso contrario, el nuevo intervalo de búsqueda será [C/2,C].

El algoritmo se repite hasta que se consigue el GOS deseado o hasta que la diferencia entre el GOS deseado y el obtenido es inferior a un valor dado, siendo el GOS obtenido inferior y al GOS deseado.

Un ejemplo de dimensionado es el siguiente:

- Número total de canales C=122.
- Número de servicios R=3. El servicio 0 son llamadas que ocupan 1 canal, el servicio 1 son llamadas que ocupan 1 canal y el servicio 2 son llamadas que ocupan 2 canales.
- Penetración de los servicios PN={0.1, 0.6, 0.3}.
- Tráfico ofrecido por individuo expresado en llamadas p=0.22 Erlangs.
- Control de acceso desactivado. Es decir, cada servicio tiene acceso a la totalidad de los 122 canales.
- GOS medio deseado menor o igual a 1%.

Cuando la penetración se interpreta como porcentaje del TC, el resultado se detalla en la Tabla 1. Cuando la penetración se interpreta como porcentaje del TO, el resultado se detalla en la Tabla 2.

Tabla 1.

r	M	c	CA	PB	PP	TC
0	47	1	122	0.007173	0.006879	10.284434
1	281	1	122	0.007173	0.006879	61.487788
2	70	2	122	0.015670	0.0143981	30.453425
Total	398					102.22564
Media				0.010005	0.009380	

Tabla 2.

r	M	c	CA	PB	PP	TC
0	36	1	122	0.007018	0.006748	7.878248
1	214	1	122	0.007018	0.006748	46.831810
2	107	2	122	0.015243	0.014070	46.561726
Total	357					101.27178
Media				0.009759	0.009189	

En general, hubiese sido deseable que el GOS medio obtenido en las Tablas 1 y 2 estuviese más próximo a 1%, pero, desafortunadamente, la sensibilidad del sistema a pequeños cambios en la población, hace difícil alcanzar el óptimo. No obstante, para el sistema estudiado, se ha demostrado que el tamaño óptimo de la población nunca es mayor a 398+2 o 357+2, según la interpretación que se haga del TC.

7 Impacto del Control de Acceso en el Dimensionado

Como se observa en las Tablas 1 y 2, en general, el servicio que solicita más canales por llamada es el que obtiene substancialmente peor PP. A este servicio se le denominará el servicio más exigente.

Es interesante estudiar un escenario en el que el resto de servicios tengan limitado el acceso a los recursos y observar en que medida esta limitación favorece al servicio más exigente. Dicho de otra forma, la intuición sugiere explorar la posibilidad de "ecualizar" los servicios. Es decir, utilizar diferentes mecanismos de control de acceso para conseguir que las PP de todos los servicios sean lo más parecidos posible, consiguiendo, al mismo tiempo, el objetivo de GOS. Lógicamente, el interés de esta ecualización es cursar más tráfico.

En esta sección se estudia el impacto que diferentes técnicas de control de acceso tienen sobre el dimensionado del sistema. En particular, se han estudiado las siguientes alternativas:

1. Ninguno de los servicios reserva recursos para su uso exclusivo, pero se limita el número máximo de recursos que cada uno los servicios puede ocupar.
2. La totalidad de los recursos se divide en R porciones independientes, siendo R el número de servicios. Cada uno de los servicios usa de forma exclusiva su porción de recursos.
3. La totalidad de los recursos se divide en R+1 porciones, siendo R el número de servicios. Cada uno de los servicios tiene asignada una porción de recursos para su uso exclusivo y además, al agotarlos, compite con el resto de

servicios por las asignación de recursos de una porción común.

La técnica de análisis propuesta por Iversen en [3] sólo permite el estudio de las dos primeras alternativas de control de acceso, pero puede ser extendida de forma sencilla para permitir el estudio de la tercera alternativa.

El siguiente ejemplo ilustra la forma en que la técnica de análisis de Iversen puede ser extendida. Supóngase un sistema en el que tres servicios compiten por C=122 canales. Los diferentes servicios han reservado los siguientes canales [10,40,50]. Es decir, hay un total de 100 canales reservados y, por tanto, 22 canales son de uso compartido.

En este escenario, el número máximo de canales que pueden ser ocupados por cada uno de los servicios son [10+22,40+22,50+22]= [32,62,72]. Por tanto, las probabilidades de estado $\{p_r(i)\}$ de cada servicio deberán ser nulas para estados superiores a éstos. Además, este hecho deberá también ser tenido en cuanta al realizar las diferentes convoluciones. Por ejemplo, si no se impone ningún tipo de limitaciones adicionales, el vector resultante de la convolución de las probabilidades de estado de los servicios 0 y 1, tendrá nulos los elementos del 32+62+1=95 a 122. Es obvio que este resultado sería erróneo, puesto que, en realidad, los elementos nulos deberían ser del 10+40+22+1=73 a 122.

Por tanto, el método de Iversen puede extenderse de forma sencilla si en cada convolución se verifican los servicios implicados y se anulan las probabilidades de estado agregadas para los estados no definidos.

En las siguientes secciones se estudian las diferentes alternativas de control de acceso en un escenario común. Para simplificar la complejidad del problema, sólo se estudia la ecualización de los servicios 1 y 2. Las características del escenario común son:

- Número total de canales C=122.
- Número de servicios R=3. El servicio 0 son llamadas que ocupan 1 canal, el servicio 1 son llamadas que ocupan 1 canal y el servicio 2 son llamadas que ocupan 2 canales.
- Penetración de los servicios PN={0.1, 0.6, 0.3}.
- Tráfico ofrecido por individuo expresado en llamadas p=0.22 Erlangs.
- GOS medio deseado menor o igual a 1%.

7.1 Cada Servicio Accede a la Totalidad de Recursos pero con un Límite Máximo de Llamadas Simultáneas

En este caso, sólo el servicio 1 tiene el control de acceso activado. Es decir, se va a limitar el número máximo de canales que pueden ocupar las llamadas del servicio 1, mientras que el resto de servicios tienen acceso a la totalidad de los 122 canales.

Los resultados se presentan en la Fig. 5 y Fig. 6. Como se observa, los servicios 0 y 2 son insensibles a la disminución del número máximo de canales que pueden ocupar las llamadas del servicio 1, hasta un límite que se encuentra entre los 67 y 62 canales. A partir de ese punto, la PP1 aumenta muy rápidamente, por lo que para mantener el objetivo de GOS es preciso disminuir el tamaño de las poblaciones de cada servicio.

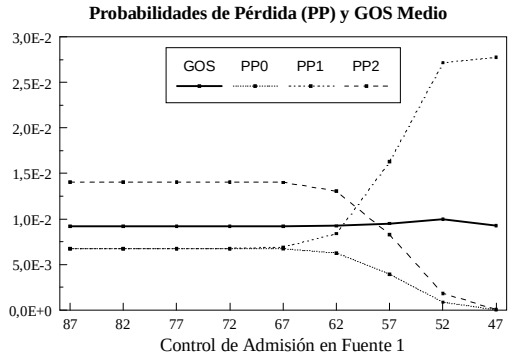


Figura 5. Probabilidades de Pérdida individuales de los diferentes servicios y GOS medio en función del número máximo de canales simultáneos que pueden ocupar las llamadas del servicio 1.

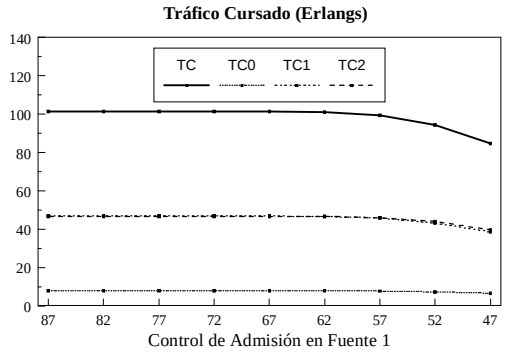


Figura 6. Tráficos Cursados individuales de los diferentes servicios y Tráfico Cursado total en función del número máximo de canales simultáneos que pueden ocupar las llamadas del servicio 1.

Por tanto, la conclusión es que el servicio 2 sólo consigue beneficiarse de la limitación impuesta sobre el servicio 1 cuando éste último es severamente penalizado. Además, el servicio 2 no consigue cursar más tráfico. Ello es debido a que al aumentar tan rápidamente la PP1, el tamaño de las

poblaciones debe ser reducido y, como consecuencia, el tráfico cursado total disminuye.

7.2 Cada Servicio Accede Sólo a una Porción de Recursos para Uso Exclusivo

En este caso, todos los servicios tienen el control de acceso activado, imponiéndose un límite al número máximo de llamadas simultáneas que cada servicio puede cursar. De forma que, la totalidad de los canales quede divida en tres partes, cada una de ellas de uso exclusivo para uno de los servicios.

Los recursos se reparten a priori en función del porcentaje de tráfico cursado por cada uno de los servicios. Se espera que los servicios 1 y 2 cursen un tráfico similar y que éste sea unas 7 veces mayor al cursado por el servicio 0.

Los resultados se presentan en la Tabla 3 y Tabla 4. Como se observa en los resultados de la Tabla 3, el servicio 0 resulta severamente penalizado, por lo que es necesario asignarle más recursos. En la Tabla 4 se observa como sólo asignando 5 canales adicionales al servicio 0 éste mejora su PP considerablemente, en cambio, el resto de servicios la empeoran también considerablemente.

Tabla 3.

r	M	c	CA	PB	PP	TC
0	27	1	10	0.033549	0.026921	5.814526
1	165	1	52	0.001237	0.001086	36.269236
2	82	2	60	0.001020	0.000829	36.056658
Total	274		122			78.140420
Media				0.011936	0.009612	

Tabla 4.

r	M	c	CA	PB	PP	TC
0	31	1	15	0.000772	0.000511	6.817281
1	185	1	51	0.013885	0.012858	40.290657
2	92	2	56	0.015803	0.014050	40.034996
Total	308		122			87.142934
Media				0.010154	0.009140	

Por tanto, se puede concluir que:

1. Con este tipo de control de acceso, las PP de los diferentes servicios son muy sensibles a las asignaciones de recursos. Esta característica limita considerablemente la aplicabilidad práctica de este método de ecualización.
2. Además, como era de esperar debido al fenómeno de la ganancia estadística, es más ventajoso disponer de un conjunto de recursos compartidos por todos los servicios que asignar porciones de estos recursos a cada servicio para su uso exclusivo.

7.3 Cada Servicio Accede a una Porción de Recursos para Uso Exclusivo y a una Porción Compartida por Todos los Servicios

En esta sección se estudia el impacto que un aumento progresivo en la porción de canales reservados para el servicio más exigente tiene sobre las prestaciones del sistema.

En este caso, todos los servicios tienen el control de acceso activado, imponiéndose un límite al número máximo de llamadas simultáneas que cada servicio puede cursar. De forma que, la totalidad de los canales se divide en cuatro partes, tres de ellas de uso exclusivo para cada uno de los servicios y la cuarta es compartida por todos los servicios. Las porciones de canales reservadas para cada uno de los servicios van desde [10,40,40] con 32 canales de uso compartido a [10,40,58] con 14 canales de uso compartido.

Los resultados se presentan en la Fig. 7 y Fig. 8. Como se observa, los servicios 0 y 1 son insensibles al aumento de la porción de canales reservados para el servicio 2 hasta un límite que se encuentra en los 48 canales. A partir de ese punto, la PP1 aumenta muy rápidamente, por lo que para mantener el objetivo de GOS es preciso disminuir el tamaño de las poblaciones de cada servicio.

En la Fig. 8 se observa como el tráfico cursado agregado presenta un pequeño máximo en 50 canales, cuyo valor es de 101,941 Erlangs y que no puede considerarse significativo.

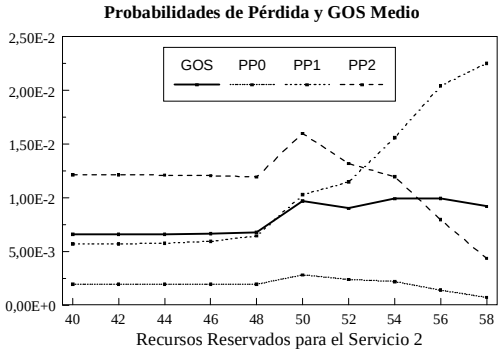


Figura 7. Probabilidades de Pérdida individuales de los diferentes servicios y GOS medio en función del número máximo de canales reservados para el servicio 2.

Por tanto, la conclusión es que el servicio 2 sólo consigue beneficiarse de un aumento de recursos para su uso exclusivo cuando el servicio 1 es severamente penalizado. Además, el servicio 2 no consigue cursar más tráfico. Ello es debido a que al aumentar tan rápidamente la PP1, el tamaño de las poblaciones debe ser reducido y, como consecuencia, el tráfico cursado agregado disminuye.

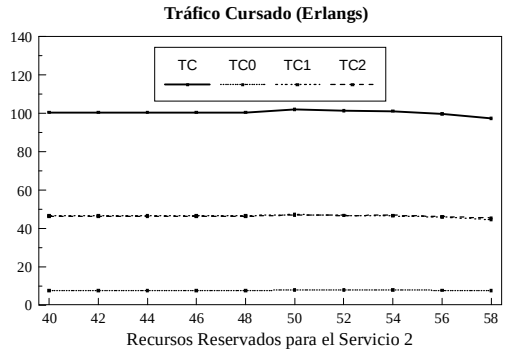


Figura 8. Tráficos Cursados individuales de los diferentes servicios y Tráfico Cursado total en función del número máximo de canales reservados para el servicio 2.

8 Conclusiones

Se ha presentado una técnica de análisis de sistemas multiservicio BPP que incorporan control de acceso, es de sencilla implementación y permite determinar los valores exactos de la PB, PP y CT.

Esta técnica puede ser extendida de forma sencilla para estudiar sistemas que, además de incorporar mecanismos de control de acceso, permiten que los servicios puedan reservar recursos de uso exclusivo y, al agotarlos, puedan competir por recursos comunes.

Los resultados demuestran que en general el servicio que requiere más canales para cursar una llamada es el que obtiene, substancialmente peor PP. Por tanto, la intuición sugiere explorar la posibilidad de "ecualizar" los servicios. Es decir, utilizar diferentes mecanismos de control de acceso para conseguir que las PP de todos los servicios sean iguales y que, al mismo tiempo, se cumpla el objetivo de GOS.

La ecualización, mediante los mecanismos de control de acceso explorados, presenta los siguientes problemas. Primero, es muy inestable. Es decir, que una vez dimensionado el sistema, pequeñas variaciones en los parámetros del mismo, como las características de tráfico, pueden alterar considerablemente los resultados. Y segundo, incluso en el caso de una ecualización perfecta, el sistema no siempre consigue cursar más tráfico.

Agradecimientos

Este trabajo ha sido financiado en parte por Alcatel España y en parte por la Comisión Interministerial de Ciencia y Tecnología (CICYT) mediante los proyectos TIC-98-0495-C02 y TIC2000-1041-C03-02.

Referencias

[1] L.E.N. Delbrouck, "On the Steady-State Distribution in a Service Facility Carrying Mixtures of Traffic with Different Peakedness Factor and Capacity Requirements," IEEE Trans. on Communications, vol. COM-31, no.11, pp.1209-1211, 1983.

[2] V.B. Iversen, "A Generalization of the Classical Teletraffic Theory," Proc. ITC11, paper 1.4.7, pp.1-7, Kyoto 1985.

[3] Villy B. Iversen, "The Exact Evaluation of Multi-Service Loss Systems with Access Control," Seventh Nordic Teletraffic Seminar (NTS-7) in Lund, Sweeden, August, 1987.

[4] V.B Iversen and S.N. Stepanov, "The Usage of Convolution Algorithm with Truncation for Estimation of Individual Blocking Probabilities in Circuit-Switched Telecommunication Networks," Proc. ITC15, pp. 1327-1336, 1997.

[5] J.S. Kauffman, "Blocking in a Shared Resource Environment," IEEE: Trans. Commun. vol.29, no.10, pp.1474-1481, October 1981.

[6] A.A. Nilsson, M. Perry, A. Gersht and V.B. Iversen, "On Multi-rate Erlang-B Computations," Proc. ITC16, vol.3b, pp. 1051-1060, Edinburgh, England, June 1999.

[7] J.W. Roberts, "A Service System with Heterogeneous User Requirements," Performance of Data Communication Systems and Their Applications, G. Pujolle Ed., North Holland, Amsterdam, pp.432-431, 1981.

Análisis y estudio comparativo de la capacidad de tráfico de un sistema LMDS multiservicio con diferentes procedimientos de reserva de canales

Vicent Plà Boscà
vpla@dcom.upv.es

Vicente Casares Giner
vcasares@dcom.upv.es

Dept. Comunicaciones, U.P. de Valencia (UPV)
Carretera Nazaret-Oliva *Camino de Vera, s/n*
46730 Grau de Gandia *46022 Valencia*

Abstract *In this paper we analyse a multiservice LMDS system fed by finite sources. Different policies for channel reservation are considered. The main goal of channel reservation is to increase capacity with respect to a completely shared environment. Efficient analytical tools are provided for the performance evaluation of such systems. We conclude that for the particular type of services under study, channel reservation does not lead to a significant improvement in terms of capacity.*

1 Introducción

El presente documento describe la solución analítica de un sistema multiservicio con pérdidas , particularizado para las necesidades y características de tráfico que ha de atender un equipo comercial [1] en un entorno LMDS.

En un sistema multiservicio con pérdidas se dispone de unos recursos limitados que han de atender peticiones de servicio de distintos tipos en cuanto a tasa de llegadas, duración del servicio y cantidad de recursos necesarios para atender la petición. Al ser los recursos limitados, la peticiones que no pueden ser atendidas se pierden (sistema de pérdidas). Por tanto, la probabilidad de que una petición se pierda (probabilidad de pérdidas) es uno de los principales indicadores de la bondad del servicio ofrecido (GoS). Por otro lado, dado que en el entorno que se está considerando el número de usuarios [1] a los que se va a prestar servicio no es muy elevado, parece razonable utilizar un modelo de población finita para las fuentes [1]. Si se fija una calidad de servicio mínima (GoS objetivo), para una cantidad de recursos y unas características determinadas de las fuentes, habrá un número máximo de usuarios que podrá atenderse y al mismo tiempo satisfacer el GoS especificado. Al número máximo de usuarios así definido es lo que se denomina capacidad del sistema.

La existencia de fuentes de distintas características y necesidades sugiere que un mecanismo en el que se reservase una serie de recursos, para todos o algunos de los servicios, podría aumentar la capacidad del sistema. En nuestro trabajo hemos hecho un análisis que permite evaluar y comparar algunos de estos esquemas de reserva.

El resto del documento está estructurado del siguiente modo: en la sección 2 se describe el sistema y se introduce la notación utilizada para proceder posteriormente a su análisis en la sección 3. En la sección 4 se utilizan las expresiones obtenidas para evaluar las distintas alternativas y, por último, extraer conclusiones en la sección 5.

2 Descripción del sistema

El sistema que se estudia en este trabajo es básicamente el mismo que el analizado en [2] donde puede encontrarse una descripción detallada del mismo. De todos modos, con el fin de que este artículo sea autocontenido a continuación se da una descripción esquemática del sistema.

2.1 Fuentes

Se tiene un total de R fuentes — cada fuente genera el tráfico de un tipo de servicio — y cada fuente se caracteriza mediante los siguiente parámetros:

- c_r número de canales necesarios para cursar una llamada.
- La duración de las llamadas se supone que está distribuida exponencialmente y su valor medio es μ_r^{-1} segundos.
- El tamaño de la población es N_r y cada individuo de los que componen la fuente, cuando está inactivo, genera llamadas a una tasa constante de λ_r.

[1] En este contexto *usuarios* son los potenciales generadores de llamadas

2.2 Recursos

- El sistema dispone C canales.
- Para cada servicio se reservan n_r canales ($n_1 + \cdots + n_R \leq C$).
- Los canales no reservados se comparten por todos los servicios ($n_c = C - \sum_{r=1}^{R} n_r$).

Basándonos en el escenario descrito en [1] algunos de los parámetros anteriores se particularizan de la siguiente forma: $R = 3$, es decir, tres tipos de servicio

1. Llamadas a Internet, $c_1 = 1$
2. Llamadas que ocupan un canal, $c_2 = 1$
3. Llamadas que ocupan dos canales, $c_3 = 2$

No obstante, la metodología aplicada en el análisis es perfectamente generalizable para valores cualesquiera de R y c_i. Lógicamente la complejidad computacional sí se verá afectada pues, entre otros factores, depende fuertemente de R. El coste temporal para el cálculo de las probabilidades de pérdida es

$$O(R\frac{C^R}{c_1 \cdots c_R})$$

2.3 Política de asignación de recursos

La reserva de canales entendemos que funciona de la forma siguiente: si (n_1, n_2, n_3) son los canales reservados para cada servicio y en un momento dado los canales ocupados para cada servicio son (b_1, b_2, b_3) cuando llega una llamada del tipo i se admitirá si [2]:

$$\begin{cases} b_i + c_i \leq n_i & \text{o bien} \\ \sum_{j \neq i} [b_j - n_j]^+ + c_i < n_c & \text{si} \quad b_i + c_i > n_i \end{cases}$$

esto es, si quedan suficientes canales reservados por utilizar en el grupo al que pertenece la llamada entrante, o si no es así, si los hay considerando también los de acceso libre.

Los distintos procedimientos de reserva de canales que se contemplan son los siguientes:

Acceso libre: si hay canales libres, cualquier servicio puede ocuparlos.

Frontera fija: existe un número de canales para cada servicio.

Reserva mínima: se reserva un mínimo de canales para cada servicio; el resto son de acceso libre.

Reserva prioritaria: como el anterior, pero sólo el servicio más prioritario tiene un mínimo de canales reservado.

Es importante destacar que si el análisis permite cualquier valor para n_1, n_2 y n_3[3] todos lo procedimientos de reserva antes mencionados pueden considerarse como caso particulares de dicho análisis. Así tendríamos que:

Acceso libre: $n_1 = n_2 = n_3 = 0$ y $n_c = C$

Frontera fija: $n_1 + n_2 + n_3 = C$ y $n_c = 0$

Reserva mínima: $n_1 + n_2 + n_3 \leq C$ y $n_c = C - (n_1 + n_2 + n_3)$

Reserva prioritaria: si p es el servicio prioritario

$$\begin{cases} 0 < n_p < C & \text{y} \\ n_j = 0 & \text{si} \quad j \neq p \end{cases}$$

y por tanto, $n_c = C - n_p$

3 Desarrollo analítico

En este apartado vamos a calcular las probabilidades de estado estacionarias para el proceso que modela nuestro sistema. A partir de éstas se obtiene la probabilidad de pérdidas global para cada servicio.

Si la terna (k_1, k_2, k_3) representa el estado en el que hay k_i llamadas activas del servicio i, y definimos S como el conjunto de posibles estados del sistema, tendremos que

$$S := \left\{ (k_1, k_2, k_3) : \sum_{i=1,2,3} [c_i k_i - n_i]^+ \leq n_c; \quad k_i \leq N_i \right\}$$

A modo de ejemplo, si suponemos que existen los estados (i, j, k) y $(i+1, j, k)$ las transiciones entre ambos serían las representadas en la Figura 1

Sea $(i, j, k) \in S$, definimos $p(i, j, k)$ como la probabilidad estacionaria del estado (i, j, k). El diagrama de transiciones de estado de este proceso puede verse como una versión truncada — resultado de eliminar algunos estado y la transiciones correspondientes — de la combinación de los diagramas de 3 sistemas independientes de colas , los cuales son a su vez procesos reversibles por tener diagramas que forman un grafo sin ciclos. En consecuencia el proceso es reversible y por tanto, existe una expresión sencilla en forma cerrada para sus probabilidades estacionarias [3, pp 440–445]

$$p(i_1, i_2, i_3) = \frac{p_1(i_1)p_2(i_2)p_3(i_3)}{\sum_{(j_1, j_2, j_3) \in S} p_1(j_1)p_2(j_2)p_3(j_3)}$$

donde $p_r(\cdot)$ es la distribución *Engset* para una población de N_r usuarios, $n_r + n_c$ servidores y parámetros de nacimiento y muerte λ_r y μ_r, respectivamente.

[2] $[x]^+ := max(0, x)$

[3] naturalmente se tendrá que cumplir que $n_1, n_2, n_2 \geq 0$ y $n_1 + n_2 + n_3 \leq C$

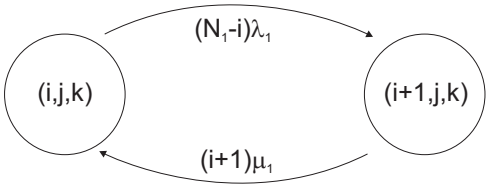


Figura 1: Transiciones entre los estados (i,j,k) y $(i+1,j,k)$

Si definimos F_r como el conjunto de estados de S a los que si llegase una llamada del tipo r sería bloqueada

$$F_1 := \{(i,j,k) \in S : (i+1,j,k) \notin S))\}$$
$$F_2 := \{(i,j,k) \in S : (i,j+1,k) \notin S))\}$$
$$F_3 := \{(i,j,k) \in S : (i,j,k+1) \notin S))\}$$

y definimos las tasas ofrecidas

$$\lambda_r^o = \sum_{(i_1,i_2,i_3)\in S} (N_r - i_r)\lambda_r p(i_1,i_2,i_3)$$

podemos expresar PP_r como

$$PP_r = \frac{\sum_{(i_1,i_2,i_3)\in F_r} (N_r - i_r)\lambda_r p(i_1,i_2,i_3)}{\lambda_r^o}$$

y finalmente obtenemos la probabilidad de pérdidas global PP

$$PP = \frac{\lambda_1^o PP_1 + \lambda_2^o PP_2 + \lambda_3^o PP_3}{\lambda_1^o + \lambda_2^o + \lambda_3^o}$$

El cálculo de estás expresiones se ha programado en $MATLAB$. Para hacer más rápida la ejecución se han generado funciones MEX compilando las funciones M. En la compilación se ha utilizado la prestaciones de optimización de código del compilador de $MATLAB$ versión 1.2. Con todo, se obtiene que para un sistema con $C = 122$ el tiempo de ejecución para el cálculo de las probabilidades de pérdida en un ordenador *Pentium III* a $500MHz$, es inferior a $3\,s$.

4 Resultados y conclusiones

En esta sección se utilizan los resultados obtenidos previamente para evaluar la capacidad del sistema en distintas situaciones de carga y estudiar en qué medida ésta puede mejorarse utilizando un mecanismo de reserva. La capacidad, y el efecto que sobre ésta pueda tener un mecanismo de reserva, depende del objetivo de calidad que se busque, o de otro modo, de qué parámetros compongan la especificación del GoS objetivo. Por ello se ha considerado distintos casos que podrían ser de interés en diversas situaciones.

Salvo que se indique lo contrario, todos los resultados presentados en esta sección corresponden a un escenario con $C = 122$ canales, un tráfico medio por usuario de 0.22 Erlangs y duraciones medias de las llamadas 45 minutos, 90 segundos y 160 segundos según sean del tipo 1, 2 ó 3, respectivamente.

4.1 CASO 1: $GoS \equiv PP_i \leq 1\% \quad i = 1,2,3$

Cuando los recursos están totalmente compartidos, el servicio limitante (peor PP_i) es el que ocupa 2 canales (servicio 3). Cosa que por otra parte podía esperarse.

Para el caso representado en la Figura 2 se tiene que, para que se cumpla que $PP_i \leq 1\% \quad i = 1,2,3$ el número total de usarios N no puede ser mayor que 456, es decir, la capacidad es igual a 456. De la comparación de las figuras 2, 3 y 4 se observa que, cualitativamente, este comportamiento es independiente de la penetración de los distintos servicios.

En esta situación, para aumentar la capacidad se reservan una cantidad de canales para uso exclusivo del servicio 3. En la Figura 5 se observa que hay un valor de n_3 a partir del cual PP_3 comienza a decrecer, pero PP_1 y PP_2 empeoran más rápidamente. Para $N = 460$, aumentando n_3 hasta 36 se puede conseguir el GoS deseado. Sin embargo, para $N = 461$ esto ya no es posible. Por tanto, mediante la reserva de canales se ha conseguido aumentar la capacidad de 456 a 461, es decir, un 1.1%.

Puede comprobarse también que a menor penetración del servicio 3 mayor es la mejora, pero en cualquier caso es bastante limitada. Por ejemplo, para unas penetraciones $(70\%, 20\%, 10\%)$, se consigue pasar de una capacidad de 462 usuarios con acceso libre, a una capacidad de 470 usuarios reservando 13 canales para el servicio 3.

4.2 CASO 2: $GoS \equiv PP < 1$

Puesto que PP (probabilidad de pérdidas global) es una media ponderada de las PP_i

$$PP = \alpha_1 PP_1 + \alpha_2 PP_2 + \alpha_3 PP_3$$

con factores de ponderación

$$\alpha_i = \frac{\lambda_i^o}{\lambda_1^o + \lambda_2^o + \lambda_3^o}$$

N	n_2	**PP**(%)	PP_1(%)	PP_2(%)	PP_3(%)
480	0	**1.02**	0.97	0.97	2.12
483	72	**1.00**	3.82	0.65	7.61

Tabla 1: Efecto sobre PP de la reserva de canales para el servicio con más peso. Penetración servicios $(30\%, 60\%, 10\%)$

N	n_3	a_1	PP(%)	PP_1(%)	PP_2(%)	PP_3(%)
456	0	0.22	0.51	0.45	0.45	0.98
423	0	0.4	0.50	0.42	0.42	0.96
443	40	0.22	0.91	0.99	0.99	0.36
443	40	0.4	4.30	4.63	4.76	1.00

Tabla 2: Reserva de canales para el servicio prioritario. Penetración servicios $(10\%, 60\%, 30\%)$

si alguna de las probabilidades de pérdida (PP_d) domina sobre las otras (α_d relativamente alta) podría pensarse que reservando canales el tipo de servicio en cuestión (i_d), puede hacerse que PP disminuya gracias a la mejora de PP_d, aunque el resto de PP_i aumenten. En la Tabla 1 se puede ver un ejemplo en el que puede apreciarse este efecto, aunque el aumento de capacidad es también en este caso muy pequeña.

Por otra parte, para las estadísticas de tráfico con que se trabaja tenemos que para penetraciones más o menos homogéneas el peso específico del servicio 2 en PP es considerablemente mayor. Por ejemplo, si las penetraciones son $(33.3\%, 33.3\%, 33.3\%)$ y el resto de parámetros toman los valores habituales se obtiene que $(\alpha_1, \alpha_2, \alpha_3) = (0.025, 0.761, 0.214)$. Incluso aunque el porcentaje de tráfico del tipo 2 sea pequeño comparado con el de otro tipo el α_2 continua siendo el peso dominante: las penetraciones $(95\%, 5\%, 5\%)$ dan lugar a unos pesos $(\alpha_1, \alpha_2, \alpha_3) = (0.331, 0.522, 0.147)$.

De los ejemplos anteriores se desprende que, cuando el GoS se mide por la probabilidad de pérdidas global, la única forma en la que se puede conseguir un pequeño incremento de la capacidad, es reservando canales para el servicio 2 y esto si el porcentaje de tráfico ofrecido de este servicio es lo suficientemente alto. Además, debido a las características de la fuente no se puede conseguir el mismo efecto con los otros servicios.

4.3 CASO 3: $GoS \equiv$ CASO 1 y $PP_{prioritario} \leq 1\%$ de forma robusta

En los casos analizados hasta ahora el hecho de reservar canales para algún servicio sólo servía para que, en casos bastante concretos, mejorase ligeramente la capacidad. Realmente esto no resulta sorprendente pues se sabe que como criterio general, compartir recursos es más eficiente que reservarlos. Sin embargo, también es conocido que la reserva de recursos sirve para aislar unos servicios de otros y así evitar que un aumento de la carga en uno de ellos degrade la calidad del servicio percibido por el resto.

Para el sistema que estamos analizando se podría tener que unos de los tres servicios es prioritario y se quiera asegurar un GoS mínimo para él aunque los otros dos experimenten picos de carga. En este contexto, reservar de canales para el servicio prioritario es probablemente la alternativa más sencilla. Para ilustrar esta casuística consideremos el siguiente ejemplo: partimos de una situación como la de la Figura 2 en la que el tráfico medio por usuario es $a_i = 0.22$ Erlangs (para todos los servicios $i = 1, 2, 3$). Para este caso habíamos visto que la capacidad es 456 usuarios. Si ahora suponemos que el tráfico del servicio 1 puede sufrir picos de carga que alcanzan los $a_1 = 0.4$ Erlangs, y queremos que el servicio 3, que consideramos prioritario, mantenga $PP_3 \leq 1\%$ tenemos dos alternativas:

1. Reducir la capacidad del sistema hasta que se cumpla que $PP_3 \leq 1\%$ aun cuando $a_3 = 0.4$ Erlangs. Así obtenemos que la capacidad es de 423 usuarios

2. Reservar canales para el servicio 3 de forma que cuando $a_1 = 0.22$ Erlangs tengamos que $PP_i \leq 1\%$ $(i = 1, 2, 3)$ y cuando $a_1 = 0.4$ Erlangs todavia se cumpla que $PP_3 \leq 1\%$. De esta forma obtenemos que con una reserva de 40 canales se puede atender a 443 usuarios y cumplir el GoS fijado.

En la Tabla 2 se puede ver un resumen de los resultado del ejemplo anterior.

5 Conclusiones

En este trabajo se ha hecho un estudio analítico de la capacidad de un sistema multiservicio con pérdidas atacado por fuentes de población finita. En el análisis se contempla la posibilidad de reservar canales por tipo de servicio. Las expresiones que se han obtenido del análisis se pueden utilizar

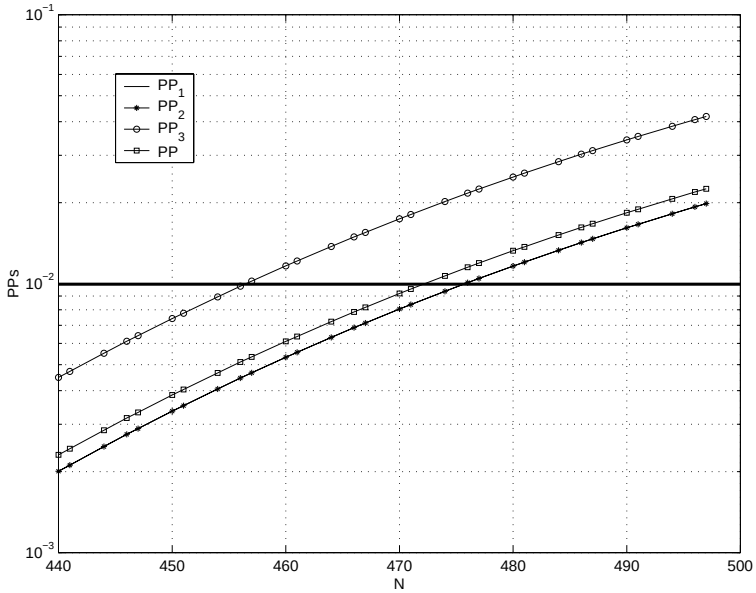


Figura 2: Acceso Libre.Penetración $(10\%, 60\%, 30\%)$

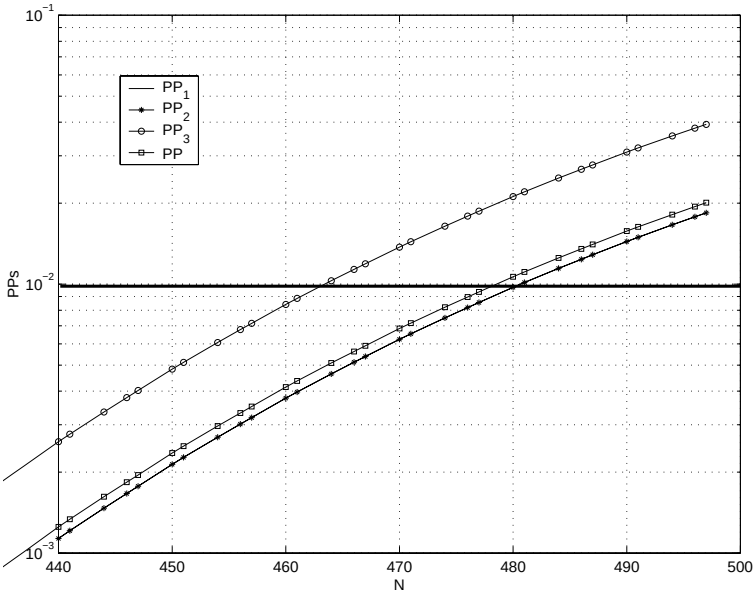


Figura 3: Acceso Libre.Penetración $(60\%, 30\%, 10\%)$

para el dimensionado de sistemas de conmutación de circuitos, multiservicio y que atienden a un número de usuarios que no es lo suficientemente alto como para suponer una población infinita. La herramienta de análisis se ha aplicado para evaluar el impacto de la reserva sobre la capacidad, en un escenario con unos tipos de servicio y unos objetivos de *GoS* concretos. En este caso, se concluye que la reserva de canales no es útil para aumentar la capacidad de forma significativa.

Agradecimientos

Este trabajo ha sido financiado en parte por *Alcatel España* y en parte por la *Comisión Interministerial de Ciencia y Tecnología (CICYT)* mediante los proyectos TIC-98-0495-C02 y TIC2000-1041-C03-02.

Referencias

[1] Carrier Interworking Division (CID) Fixed Wireless R&D Systems. Modelos de tráfico para sistemas de conmutación de circuitos. Escenarios. Technical report, Alcatel, January 2001.

[2] Jorge Martínez y Vicente Casares. Análisis y dimensionado de sistemas de pérdidas multiservicio. In *Actas de JITEL'01*, Septiembre 2001.

[3] Randolph Nelson. *Probability, stochastic processes, and queueing theory: The mathematics of computer performance modeling.* Springer-Verlag, 1995.

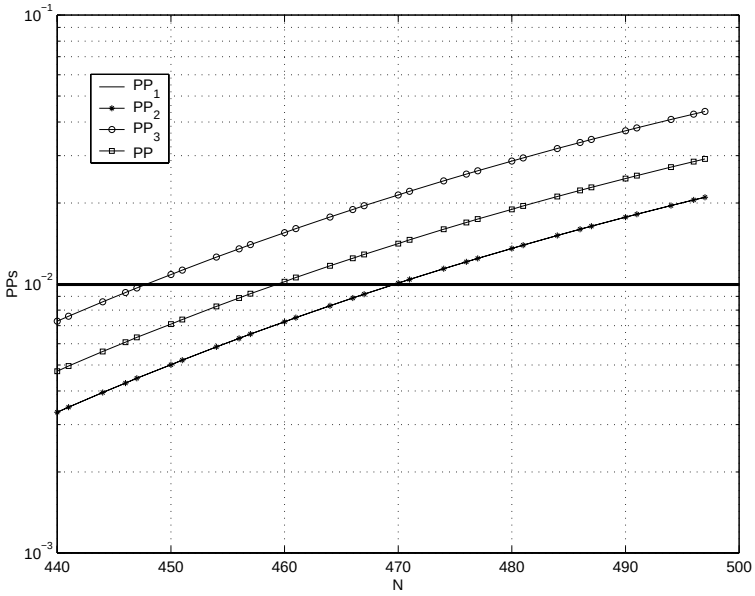


Figura 4: Acceso Libre.Penetración $(10\%, 30\%, 60\%)$

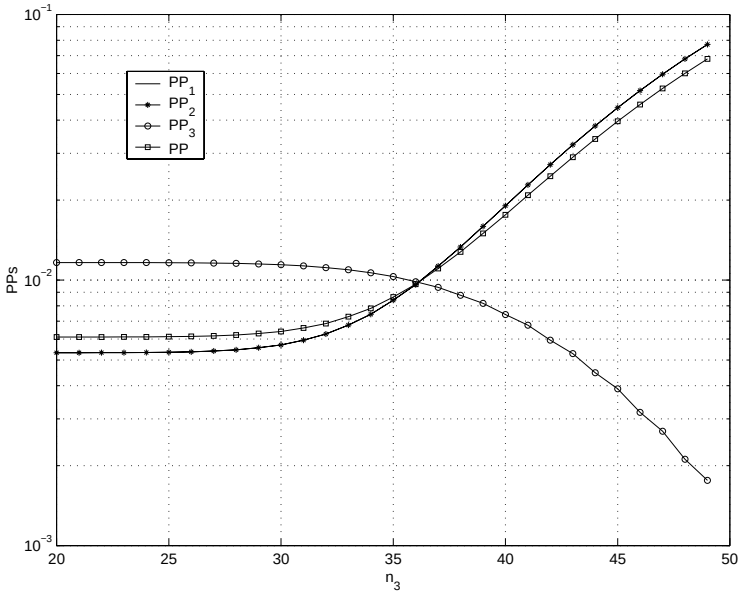


Figura 5: Reserva de canales para el servicio 3. $N = 460$, penetración $(10\%, 60\%, 30\%)$

Garantía de calidad de servicio basada en la predicción del ancho de banda

Jesús Villadangos, Eduardo Magaña
Dpt. Automática y Computación, Universidad Pública de Navarra
Campus Arrosadía, 31006 - Pamplona - SPAIN
Teléfono. : +34 948 169645 Fax: +34 948 168924
E-mail: jesusv@unavarra.es

Abstract*. This paper presents the architecture and performance evaluation of a neuronal estimator to predict network load in communication networks. System benchmarks are tested with real network traffic picked up from the 155 Mbps ATM Internet connection of the Universidad Pública de Navarra. The system shows good benefits in traffic prediction with 3 and 5 hours of advance. So the system shows characteristic of great interest to carry out the dynamic assignment of bandwidth in Internet Service Providers (ISPs), guaranteeing quality of service hired by the users.*

1 Introducción

Los usuarios residenciales acceden a las redes de datos a través de un proveedor de servicios (ISP). Actualmente, cada ISP tiene contratado un acceso a una operadora de red de datos y sobre dicho acceso multiplexa a sus clientes, por lo que el ancho de banda se comparte por los usuarios.

Un parámetro de diseño para los operadores de un ISP es, por tanto, el número de usuarios a los que se va a proporcionar servicios. Sin embargo, más importante es conocer qué tipo de servicios van a ser usados por sus clientes y, en concreto, el consumo de ancho de banda que requiere cada usuario. El estudio del ancho de banda requerido por el ISP para dar servicio a sus clientes permite determinar la tasa que éste contratará a la operadora de red. La selección de un ancho de banda es un compromiso entre el coste del acceso y el grado de servicio que se quiere proporcionar a los usuarios. A los ISPs les interesa maximizar el uso del ancho de banda contratado.

El objeto principal de este trabajo es presentar una arquitectura de estimador del ancho de banda basada en una red neuronal que permite predecir el ancho de banda requerido para asegurar el servicio a sus clientes. Se presenta los resultados de su aplicación para el caso del enlace de entrada y salida de la Universidad Pública de Navarra.

El sistema se basa en el uso de un estimador de la carga de la red que permite predecir con horas de anticipación los bytes totales transferidos por los usuarios. El estimador consiste en un filtro FIR realizado mediante una red neuronal [1]. Las redes neuronales se han aplicado en muchos casos para la identificación y control de sistemas [2, 3] y de sistemas de telecomunicación (codificación de datos, control de errores, control de admisión de llamadas, control de congestión, etc.) [4, 5, 6]. Sin embargo, en todos los casos se han utilizado siempre modelos de tráfico sintético y no datos reales de tráfico.

Los datos para entrenar la red neuronal se obtienen de la captura de todos los paquetes que circulan por el punto de acceso (TR1) de la Universidad Pública de Navarra con RedIris, la troncal que une a las universidades españolas con el resto de Internet. El tráfico analizado se puede considerar representativo del tráfico que maneja un ISP, ya que la universidad tiene un alto número de usuarios que hacen uso de muy diversos servicios de Internet.

En este trabajo se presenta, por tanto, la adecuación de un estimador neuronal para la predicción de la carga de la red de la Universidad. Se analiza tanto el canal de bajada como el de subida haciéndose la predicción con una anticipación entre una y cinco horas.

En primer lugar se presentará la arquitectura de red neuronal que permite realizar predicciones sobre series temporales. Esta propuesta se centra en los resultados que se han obtenido al hacer uso de una red neuronal para estimar el tráfico real de la Universidad Pública de Navarra. A continuación se analiza la carga de la red para diferentes niveles de predicción. Finalmente se presentan las conclusiones y referencias del trabajo.

2 Arquitectura del estimador neuronal

Es bien conocido que las redes neuronales son capaces de realizar mapeos no lineales entre conjuntos de entradas y salidas. Una red neuronal feedforward de tres capas con neuronas, cuya función de activación es de tipo sigmoidal es capaz de aproximar una función no lineal con cualquier

grado de precisión [1]. Sin embargo, este tipo de redes no están diseñadas para tener en cuenta la dinámica de las señales variables en el tiempo.

Un método para representar el tiempo en las redes neuronales es utilizar una red de tipo Time Delay Neural Network (TDNN), la cual es una red multicapa feedforward en la que las salidas de las neuronas se almacenan durante un número finito de intervalos de tiempo y sirven de entrada para las neuronas de la siguiente capa.

La topología de las redes TDNN está incluida en las redes perceptron multicapa considerando que cada una de las salidas en realidad es la respuesta de un filtro FIR (*Finite Impulse Response*), donde FIR indica que para una entrada de duración finita, la salida del filtro tiene una duración finita. Este tipo de redes se denominan perceptrones multicapa FIR. Ambas redes TDNN y FIR son funcionalmente equivalentes. Sin embargo, la red FIR está directamente relacionada con las redes multicapa. Las redes FIR, además, permiten derivar esquemas de adaptación de modo sencillo. Por tanto, en este trabajo se hará uso de una red FIR como sistema de predicción del tráfico.

2.1 Modelo de red FIR

Como se ha indicado anteriormente, las redes multicapa permiten realizar un mapeo estático. A estas redes se les aplica una modificación transformando el peso de cada una de las conexiones por un filtro FIR lineal. Para este filtro, la salida *y(k)* se corresponde con la suma ponderada de los valores de entradas de instantes anteriores:

$$y(k)=\sum_{n=0}^{T} w(n)x(k-n) \tag{1}$$

A partir de la ecuación (1) se puede formular el modelo de una neurona FIR. Sea $w_{ij}(l)$ el peso que corresponde al filtro FIR de la conexión que une la neurona *i* con la neurona *j* (*i*=1,2…,p). El parámetro *l* toma valor en el intervalo [0, M], donde M es el número total de unidades de retardo que se consideran al diseñar el filtro FIR. Finalmente, sea $y_j(n)$ el valor de la función de salida de la neurona *j* y $x_i(n)$ la señal de entrada. Entonces, se tiene que

$$v_j(n)=\sum_{i=1}^{p}\sum_{l=0}^{M} w_{ji}(l)\, x_i(n-l)-\theta_j \tag{2}$$

$$y_j(n)=\varphi(v_j(n)) \tag{3}$$

donde $v_j(n)$ es el potencial de activación de la neurona *j*, θ_j es el umbral externo para la neurona *j* y $\varphi(.)$ es la función no lineal de activación de la neurona.

Las ecuaciones 2 y 3 se pueden reformular en forma matricial, donde se va a hacer uso de las siguientes definiciones para el vector de estado y el vector de pesos para la conexión *i*, respectivamente:

$$\chi_i(n)=[x_i(n), x_i(n-1),..., x_i(n-M)]^T \tag{4}$$

$$\varpi_{ij}=[w_{ji}(0), w_{ji}(1),..., w_{ji}(M),]^T \tag{5}$$

El valor de salida $y_j(n)$ de la neurona *j* se expresa del siguiente modo:

$$y_j(n)=\varphi\left(\sum_{i=1}^{p}\varpi_{ji}^T\,\chi_i(n)-\theta_j\right) \tag{6}$$

Además, el modelo de la red FIR se basa en neuronas cuya estructura considera un peso w_{0j} conectado a la entrada fija x_0 = -1 para representar el umbral externo θ_j.

A partir del modelo de neurona anterior se puede construir una red de tipo perceptrón multicapa cuyas neuronas ocultas y de salida se basan en el modelo de filtro FIR. Esta estructura de red se denomina red FIR. La diferencia entre las redes FIR y las redes multicapa tradicionales reside en que las conexiones estáticas de las redes multicapa se cambian por versiones dinámicas. En redes multicapa tradicionales la conexión entre dos neuronas está ponderada por un peso, mientras que en las redes FIR este peso se transforma en un conjunto de pesos asociados cada uno de ellos a la entrada en instantes anteriores.

2.2 Aprendizaje backpropagation temporal

Dada una secuencia de entrada *x(k)*, la red produce una secuencia de salida *y(k) = N(W, x(k))*, donde *W* representa el conjunto de todos los coeficientes de los filtros presentes en la red. Se define el error instantáneo $e^2(k) = ||d(k) - y(k)||^2$ como la distancia euclídea entre la salida de la red y la salida deseada. Por tanto, el objetivo del entrenamiento se corresponde con ajustar el valor de los coeficientes de *W* para minimizar la siguiente función de coste:

$$C=\frac{1}{2}\sum_{k=1}^{K} e^2(k) \tag{7}$$

donde la suma se realiza sobre el conjunto total *K* de muestras de aprendizaje. El algoritmo para minimizar el error de la función *C* se presenta en [7] y se denomina backpropagation con tiempo. La función de adaptación de los pesos se presenta a continuación:

$$\varpi_{ji}(k+1)=\varpi_{ji}(k+1)-\eta\frac{\partial C}{\partial v_j(k)}\frac{\partial v_j(k)}{\partial \varpi_{ji}(k)}=\varpi_{ji}(k+1)-\eta\delta_j(k)\chi_i(k) \tag{8}$$

$$\delta_j(k)=\begin{cases} e_j(k)\varphi'(v_j(k)), & capa\,de\,salida \\ \varphi'(v_j(k))\sum_{m\in A}\Delta_m^T(k)\varpi_{mj}, & capa\,oculta \end{cases} \tag{9}$$

donde η es el parámetro de velocidad de aprendizaje, A se define como el conjunto de todas las neuronas cuyas entradas se ven afectadas por el valor de salida del nodo *j* y $\Delta_m(k)$ se define del siguiente modo:

$$\Delta_m(k)=[\delta_m(k),\delta_m(k+1),...,\delta_m(k+M)]^T \quad (10)$$

El conjunto de ecuaciones anteriores representa una generalización del algoritmo de aprendizaje clásico de backpropagation. De hecho, se puede reemplazar el vector de entrada $\chi_i(n)$, el vector de pesos $\varpi_{mj}(n)$, y el vector gradiente Δ_m por sus correspondientes valores escalares y se obtendría el algoritmo backpropagation para redes estáticas. Para calcular el valor $\delta_j(k)$ para una neurona *j* de una capa oculta, se filtra el resultado de las δ de las siguientes capas hacia atrás a partir del conjunto de nodos que se ven afectados por el valor de salida de la neurona *j*. Por tanto, el valor de las δ no está afectado sólo por los valores de los pesos, sino que se adapta en función del resultado del filtrado de la señal hacia atrás. Para cada nueva entrada y respuesta deseada, los filtros hacia delante y hacia atrás se incrementan una unidad temporal. Entonces, los pesos se adaptan on-line para cada intervalo de tiempo.

Utilizar el algoritmo backpropagation con tiempo hace que se preserve la simetría entre la propagación hacia delante de los estados y la propagación hacia atrás de los errores. Se mantiene por tanto el procesamiento paralelo en el sistema. Además, cada peso se utiliza de forma individual y una sola vez para calcular el valor de las δ; es decir, no hay redundancia en el instante de aplicar el modelo del gradiente.

Sin embargo, un análisis detallado de las ecuaciones anteriores permite mostrar que no se tiene un orden causal a la hora de determinar los valores de $\delta_j(k)$. Este cálculo se puede expresar de forma causal el algoritmo backpropagation con tiempo del siguiente modo:

Para cada neurona *j* de la capa de salida, calcular

$$\varpi_{ji}(k+1)=\varpi_{ji}(k)+\eta\,\delta_j(k)\,\chi_i(k) \quad (11)$$

$$\delta_j(k)=e_j(k)\varphi'_i(k) \quad (12)$$

Para cada neurona *j* de una capa oculta, calcular

$$\varpi_{ji}(k+1)=\varpi_{ji}(k)+\eta\,\delta_j(k-lM)\chi_i(k-lM) \quad (13)$$

$$\delta_j(k-lM)=\varphi'(v_j(k-lM))\sum_{m\in\Lambda}\Delta_m^T(k-lM)\,w_{mj} \quad (14)$$

donde *M* es la longitud del filtro y se usa el índice *l* para identificar la capa oculta. Es decir, *l* = 1 se refiere a la primera capa oculta anterior a la capa de salida.

3 Estimación de la carga usando redes FIR

Las redes neuronales tienen capacidad de adaptación y permiten modelar la ausencia de estacionariedad de los sistemas. Sus capacidades de generalización las hacen herramientas flexibles y robustas cuando se está tratando con datos que incluyen patrones ruidosos. En este trabajo, el papel de la red neuronal es capturar la compleja relación entre valores de carga pasados y futuros. El objetivo es predecir cual va a ser la carga del enlace en instantes futuros con el fin de proponer un sistema que permita prever los recursos de comunicaciones necesarios para el sistema.

3.1 Configuración del estimador durante el aprendizaje

Sea *x(k)* una serie temporal escalar, la cual se describe por un modelo de regresión no lineal de orden q como sigue:

$$x(n)=f(x(n-1), x(n-2), \ldots, x(n-q)) + \varepsilon(n) \quad (15)$$

donde *f* es una función no lineal de sus argumentos y $\varepsilon(n)$ es un residuo. Se asume que $\varepsilon(n)$ se puede representar por un ruido blanco gausiano. La funcion no lineal *f* se desconoce a priori, y la única información que se posee es la observado y representada por la serie *x(1)*, … , *x(N)*. *N* determina el número total de muestras de la serie. Se puede usar una red FIR como estimador de un paso y orden q para modelar la serie temporal. De hecho la red se diseña para realizar la estimación del valor *x(n)* teniendo en cuenta las q entradas pasadas. Es decir, *x(n-1)*, …, *x(n-q)*, como se indica en la expresión del estimador:

$$\hat{x}(n)=F(x(n-1),...,x(n-q))+e(n) \quad (16)$$

En este trabajo se utiliza el estimador para predecir con diferentes pasos.

La función no lineal *F* es una aproximación de la función f, la cual se calcula mediante la red FIR. El valor *x(n)* actúa como valor deseado. Así, la red FIR se entrena con el objetivo de minimizar el error de predicción:

$$e(n)=x(n)-\hat{x}(n) \quad q+1\le n\le N \quad (17)$$

En nuestro caso, la red FIR se diseña como una red totalmente conectada con tres capas de 1, 8 y 1 neuronas en cada capa a partir de la de entrada y con 3 elementos para llevar a cabo los filtros FIR. La selección se ha realizado mediante el método de prueba/error, ya que no se conocen en la literatura métodos que determinen la configuración de las redes neuronales. La red FIR se entrena utilizando la forma causal del algoritmo backpropagation temporal y se usa el error cuadrático medio como la

medida del error. El parámetro de aprendizaje se establece a valor 0.1 y se utiliza en cada neurona como función de activación la sigmoidal.

3.2 Modelo de tráfico

El entrenamiento y validación del estimador se realiza mediante el uso de una traza de tráfico capturada en la Universidad Pública de Navarra desde el 27 de noviembre de 1998 al 11 de enero de 1999 de la que se obtienen los bytes por hora en el enlace de la universidad tanto en la bajada (downstream), como en la subida (upstream). Se distinguen los dos casos debido a la asimetría que presenta el tráfico en la red, siendo el tráfico de bajada mayor que el de subida debido a que los usuarios demandan del exterior más información que la proporcionada por servidores de la universidad a usuarios externos. La traza de tráfico contiene información sobre 1095 horas (46 días) siendo los valores de cada hora los bytes transferidos en cada sentido durante cada hora.

4 Análisis de prestaciones del estimador de la carga de tráfico

Este apartado muestra las prestaciones del estimador en dos casos: (i) estimación del ancho de banda requerido por los usuarios en la hora siguiente y (ii) la predicción para dentro de tres y cinco horas a partir de la actual. En el primer caso se estudia las posibilidades de estimación usando como información de partida el día de la semana y la hora del día. A continuación se basa la predicción en el uso del valor de la carga para horas anteriores junto con el día de la semana. En el segundo caso se utilizan las entradas retrasadas y el día de la semana para determinar el ancho de banda requerido por los usuarios con tres y cinco horas de antelación.

4.1 Estimación del ancho de banda para la hora siguiente

En este apartado se analiza la estimación obtenida por la red neuronal para los enlaces de bajada y subida considerando que se tiene como información de partida (i) en primer lugar, el día de la semana y la hora del día y (ii) en segundo lugar, el día de la semana y el valor del ancho de banda requerido por los usuarios en X horas anteriores.

En el primer caso, en cuanto al canal de bajada, la estimación de la carga del enlace se muestra en la Fig.1, mientras que la Fig.2 muestra dicha estimación para el enlace de subida. En ambos casos se puede comprobar que la red neuronal determina el comportamiento del sistema como periódico.

En este caso se comprueba que las entradas no son suficientemente generales para poder modelar el comportamiento de las necesidades de ancho de banda de los usuarios. El estimador, sin embargo, ha generalizado el comportamiento de los usuarios a un esquema de alta demanda durante los días laborables y de baja demanda durante los días no laborables.

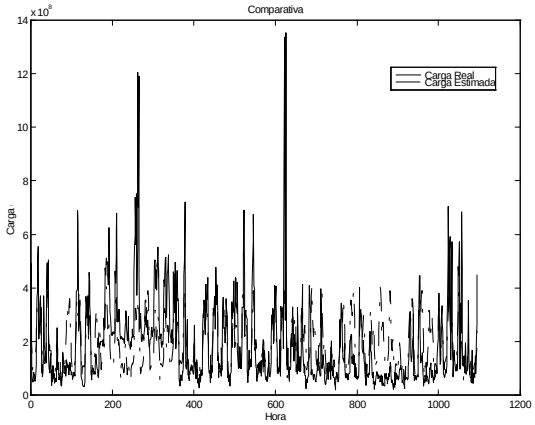


Figura 1 : Tráfico en el canal de bajada considerando día de la semana y hora del día.

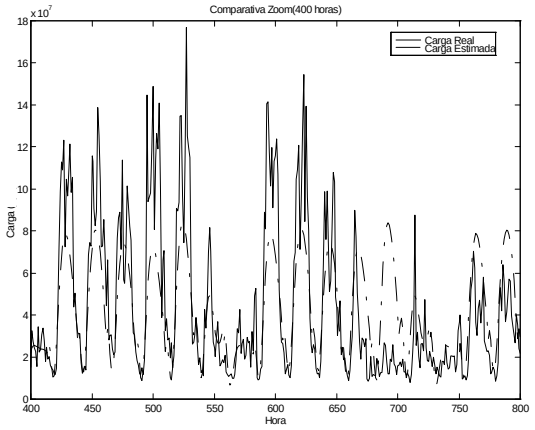


Figura 2 : Tráfico en el canal de subida considerando día de la semana y hora del día.

En segundo lugar se estima el ancho de banda requerido por los usuarios en la siguiente hora pero usando como información de entrada el día de la semana, la hora del día y el ancho de banda requerido en X = 5 horas anteriores.

La Fig. 3 muestra la predicción para el canal de bajada. En este caso se puede comprobar como el estimador ha sido capaz de generalizar los datos de aprendizaje y permite una predicción bastante buena para la hora siguiente.

A continuación se muestra en la Fig. 4 un detalle de la estimación superponiendo la carga real y la estimada. En este caso se puede comprobar que la realización del estimador es bastante aproximada a la carga real.

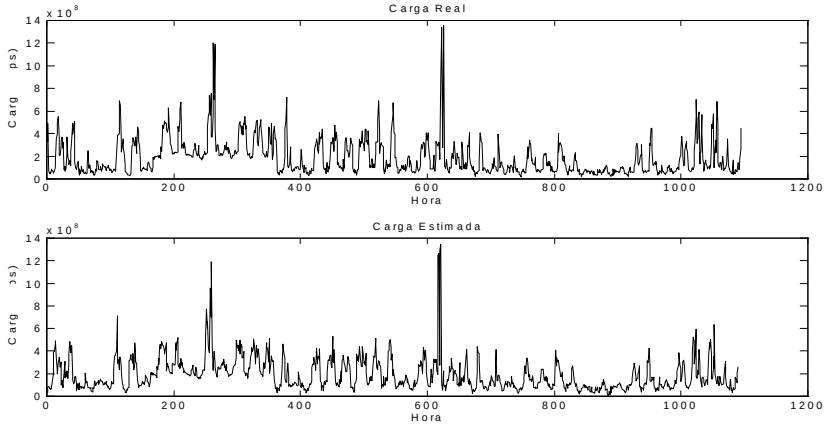


Figura 3: Estimación del ancho de banda del canal de bajada considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

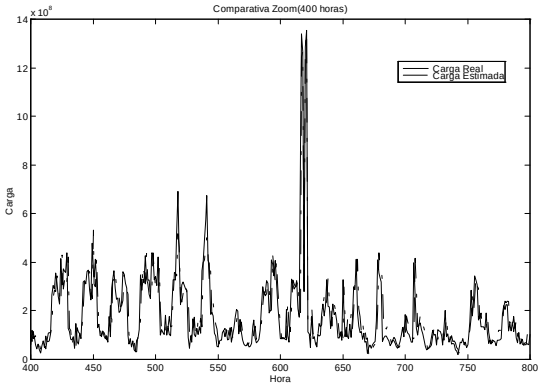


Figura 4 : Detalle de la estimación del ancho de banda en el canal de bajada considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

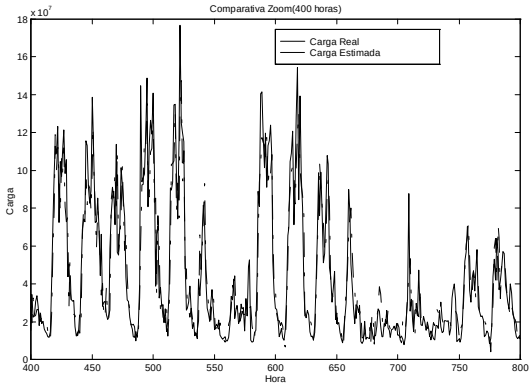


Figura 6 : Detalle de la estimación del ancho de banda en el canal de bajada considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

La Fig. 5 muestra la predicción para el canal de subida, así como la Fig. 6 muestra un detalle para dicha predicción. En este caso, como en el anterior para el canal de bajada se puede comprobar que el estimador puede ser de gran utilidad para un proveedor de servicios de Internet para determinar el ancho de banda que van a requerir los usuarios en la siguiente hora.

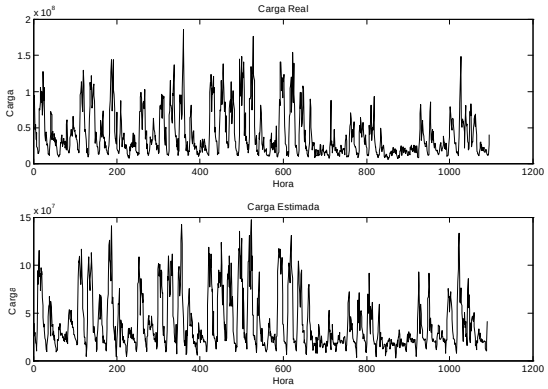


Figura 5 : Estimación del ancho de banda en el canal de subida considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

En este apartado se ha mostrado que el uso de variables como el día de la semana, la hora del día y entradas anteriores permiten determinar el ancho de banda requerido por los usuarios en la hora siguiente. Sin embargo, para un proveedor de servicios resulta de mayor interés estimar con precisión el ancho de banda requerido con antelación y cuanta mayor antelación mejor. Esto permitiría tomar acciones preventivas al proveedor de servicios para disponer del ancho de banda en caso de ser necesario aumentar el contratado o bien reducir el ancho de banda contratado si no se va a utilizar.

4.2 Predicción del ancho de banda requerido con horas de antelación

El estimador trabaja en este caso con los siguientes datos: día de la semana, hora del día y valor del ancho de banda en X=5 horas anteriores. El resultado obtenido es el valor de ancho de banda que se predice con Y= 2 horas de antelación.

La Fig.7 muestra la capacidad de predicción del sistema para el caso del canal de bajada y la Fig. 8 muestra la capacidad de predicción del sistema para el canal de subida. En ambos casos se puede comprobar que la red neuronal realiza una predicción que no se ajusta exactamente a la carga real pero que puede ser una buena aproximación para determinar el ancho de banda que requiere el proveedor de servicios para dar servicio a sus usuarios.

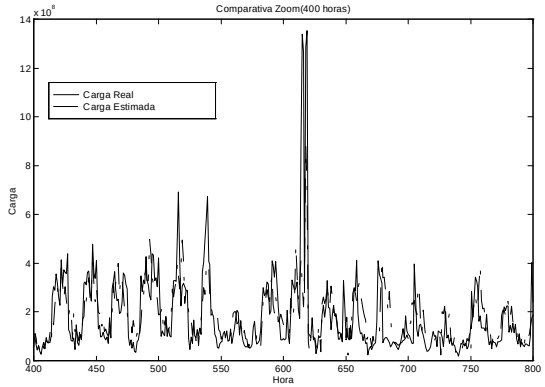


Figura 7 : Detalle de la estimación del ancho de banda en el canal de bajada con dos horas de antelación considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

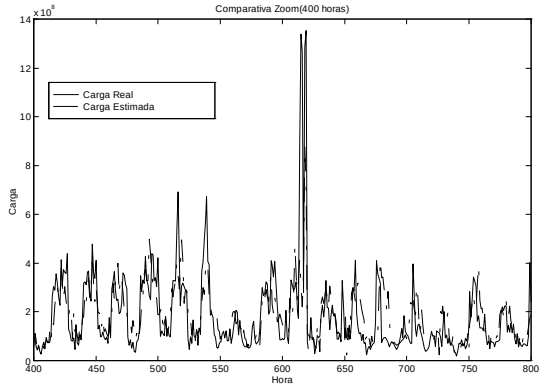


Figura 8 : Detalle de la estimación del ancho de banda en el canal de subida con dos horas de antelación considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

Conclusiones

El presente trabajo muestra la utilidad de las redes neuronales como estimadores de tráfico en redes de datos, las cuales se caracterizan por tener una alta variabilidad en el patrón del tráfico que circula por ellas. La estimación del ancho de banda permite prever con suficiente antelación el ancho de banda que debe contratar el proveedor de servicios para dar servicio a sus usuarios. Esto permite que los proveedores puedan contratar un ancho de banda básico y activar líneas de respaldo cuando la demanda de ancho de banda se prevea muy alta. Esto permitiría reducir costes de contratación a los proveedores de servicios.

En este trabajo, además, se muestra la posibilidad de integrar de manera natural en el estimador entradas de gran contenido semántico como son el día de la semana y la hora del día además de los valores del ancho de banda requerido por los usuarios en horas anteriores. Estas entradas no se pueden integrar en la realización de estimadores clásicos y, sin embargo, son de gran utilidad para la realización de las previsiones de un operador de red porque tienen en cuenta datos importantes como la estacionalidad del tráfico.

La estimaciones realizadas con el sistema propuesto se han realizado sobre tráfico real mostrando la bondad del sistema y las posibilidades que proporciona para un proveedor de servicios. Las redes neuronales muestran su capacidad para predecir el tráfico que circula por un enlace de comunicaciones, aunque se debe buscar una arquitectura de red u otro tipo de sistemas para ajustar la estimación a la carga real del sistema e intentar la predicción con mayor antelación.

Referencias

[1] Z. Fan, P. Mars; *ATM traffic prediction using FIR neural networks.* ATM Networks: Performance modelling and evaluation, vol II. Chapman & Hall, 1996

[2] B. Kosko; *Neural networks and fuzzy systems: A dynamical systems approach to machine intelligence*. Prentice Hall, 1991.

[3] J. E. Neves, M. J. Leitao, L. B. Almeida; *Neural networks in B-ISDN flow control: ATM traffic prediction or Network modeling?*. IEEE Communications Magazine, Oct. 1995.

[4] A. Hiramatsu; *ATM communications network control by neural networks.* IEEE Transactions on Neural Networks, v. 1, n. 1, March 1990.

[5] Y-K. Park, G. Lee; *Applications of neural networks in high-speed communication networks.* IEEE Communications Magazine, Oct. 1995.

[6] R-G. Cheng, C-J. Chang, L-F. Lin*; A QoS-provisioning neural fuzzy connection admission controller for multimedia high-speed networks*. IEEE/ACM Transactions on Networking, v. 7, n. 1, Feb. 1999.

[7] E. A. Wan; *Time series prediction by using a connectionist network with internal delay lines*. In Time Series Prediction, 195-217, Addison-Wesley.

La formación del Ingeniero de Telecomunicación para la Sociedad de la Información

Luis Guijarro Coloma, Antonio Alabau Muñoz
Departamento de Comunicaciones. Universidad Politécnica de Valencia.
Camino de Vera, s/n. 46022 Valencia
Teléfono: 96 3879303 Fax: 96 3877309
E-mail: lguijar@dcom.upv.es, aalabau@dcom.upv.es

Abstract. *The Information Society has been identified as one of the main driving forces of the development of the European economy. Furthermore, the deployment of the Information Society is requiring an increasingly complex level of information and communication technologies and has caused a dramatic lack of engineers in the field of the Telecommunications. The University, as the institution committed to provide the labour market with these sort of professionals, should find new ways of matching the dinamic requirements of the Information Society and design the curriculum of the Telecommunications Engineering accordingly. Some experiences carried out at the Universidad Politécnica de Valencia and some proposals are described in this paper.*

1 Introducción.

La Sociedad de la Información remonta su gestación a los años 70, si bien el término, según es aceptado en el sector, fue acuñado por el expresidente de la Comisión Europea Jacques Delors en el Libro Blanco de 1993 [1].

Entre las acepciones diversas que aparecen en la literatura, citaremos aquí una de las más recientes y, en nuestra opinión, canónica:

«La Sociedad de la Información es un estadio de desarrollo social caracterizado por la capacidad de sus miembros (ciudadanos, empresas y administración pública) para obtener y compartir cualquier información, instantáneamente, desde cualquier lugar y en la forma que se prefiera» [2]

Este nuevo modelo de sociedad está emergiendo impulsado por dos factores que le son esenciales. Por un lado, la constante penetración y la consecuente omnipresencia de las tecnologías de la información y las comunicaciones (TIC) en los procesos de la economía, que indefectiblemente modificará las actitudes y los valores de la sociedad. Por otro lado, la instalación de nuevos modelos organizativos en las empresas, los cuales son cada vez más dependientes de las posibilidades que ofrecen las TIC para distribuir sus procesos productivos y de decisión [3].

Dentro de este marco, las acciones estratégicas de algunos gobiernos están persiguiendo conseguir las condiciones óptimas para precipitar las condiciones apuntadas que caracterizan el acceso general, ubicuo e ilimitado a la información en la Sociedad de la Información.

Por nuesta parte, centraremos la atención en la repercusión que este nuevo escenario socioeconómico tiene sobre la profesión del Ingeniero de Telecomunicación y, especialmente, en cuáles son las oportunidades y las amenazas que este escenario plantea a la Universidad como institución responsable de la formación de nuevos profesionales que lideren la implantación de la Sociedad de la Información.

Las reflexiones que presentamos aquí fruto del trabajo desarrollado en la Escuela de Ingenieros de Telecomunicación de Valencia, cuyos primeros resultados fueron publicados en las II Jornadas de Ingeniería Telemática, celebradas en 1999 [4]. En aquel momento, intuímos que las condiciones que hace más de una década determinaron la aparición del Ingeniero Telemático en el currículum de los estudios de Ingeniero de Telecomunicación, se estaban dando como consecuencia del desarrollo de la Sociedad de la Información. Efectivamente, se trató entonces de la demanda de un nuevo perfil profesional que inicialmente fue cubierto por profesionales de procedencia diversa y que determinó la incorporación del perfil del Ingeniero Telemático en los estudios de Ingeniero de Telecomunicación. Concluíamos entonces que era necesario reaccionar con más decisión ante el nuevo fenómeno de la Sociedad de la Información.

2 Acciones estratégicas en Sociedad de la Información.

El Libro Blanco *Crecimiento, competitividad y empleo. Retos y pistas para entrar en el siglo XXI* (1993), supone la identificación de la Sociedad de la Información como elemento clave para el desarrollo económico de la Unión Europea. A partir de este momento, desde la Comisión Europea se toman las medidas oportunas para imbricar este

objetivo estratégico en la política europea. A continuación describimos las medidas de este tipo más destacables.

2.1 Quinto Programa Marco

El Quinto Programa Marco establece las prioridades en materia de actividades de Investigación, Desarrollo Tecnológico y Demostración de la Unión Europea para el periodo 1998-2002. Estas prioridades se han formulado con el objetivo de lograr aumentar la competitividad de las empresas europeas y la calidad de vida de los ciudadanos europeos.

El Quinto Programa Marco, a diferencia de sus predecesores, ha decidido limitarse a una serie de áreas de investigación, con el fin de maximizar su impacto. Así pues, su estructura define siete Programas Específicos, de los cuales cuatro de ellos son Programas Temáticos. Uno de ellos es el Programa de las Tecnologías de la Sociedad de la Información (*Information Society Technologies*, IST), el cual concentra 3600 de los 13700 millones de Euros dedicados al Programa Marco, lo cual refleja la importancia que empieza a adquirir este tipo de tecnologías.

El objetivo estratégico de este Programa Temático es materializar los beneficios de la Sociedad de la Información para Europa a través de la aceleración en su implantación y de la satisfacción de las necesidades de los ciudadanos y de las empresas. Se trata de un Programa que hereda las iniciativas de los programas ESPRIT, ACTS y Telematics Applications del IV Programa Marco, pero se basa en una nueva aproximación más integradora de las TIC.

2.2 Iniciativa *eEurope*

En diciembre de 1999, la Comisión Europea puso en marcha la iniciativa *eEurope*, con los siguientes objetivos clave:

- Conseguir que todos los europeos entren en la era digital y estén conectados a la red.
- Crear en Europa una cultura y un espíritu empresarial abiertos a la cultura digital.
- Garantizar que el proceso no se traduzca en exclusión social y se gane la confianza del consumidor.

Esta iniciativa supuso el relanzamiento del proyecto de creación de la Sociedad de la Información por parte de la Comisión Europea. En junio de 2000, el Consejo Europeo de Feira adoptó el plan de acción *eEurope* 2002.

Las actividades de *eEurope* han comenzado ya a arrojar resultados positivos a nivel sectorial. Las principales áreas en las que las actividades políticas y a nivel de programa han actuado en paralelo son las siguientes:

- *eContent*, área en la que la Comisión ha propuesto un nuevo programa de 150 millones de euros con el propósito de estimular la creación y el uso de contenidos digitales europeos en Internet y promover la diversidad lingüística en las páginas web europeas.
- Educación, área en la que la iniciativa *eLearning* y el reforzamiento de las actividades correspondientes del programa IST contribuirán a adaptar el sistema educativo a la nueva economía.

2.3 Iniciativa InfoXXI

En el ámbito de actuación español, el Plan de Acción Info XXI, para el período 2001-2003, está compuesto por un conjunto de iniciativas (más de trescientas acciones y proyectos) que representan un importante impulso para el desarrollo de la Sociedad de la Información en España.

Es la concreción (con objetivos, plazos, responsables, colaboradores y financiación) de las líneas maestras recogidas en la iniciativa del Gobierno Info XXI: "La Sociedad de la Información para todos", que se presentó un año antes.

El Plan Info XXI responde a los objetivos establecidos en la iniciativa *eEurope*. Este Plan de Acción se articula en tres grandes líneas:

- El impulso del sector de las Telecomunicaciones y las Tecnologías de la Información, completando la liberalización y favoreciendo la competencia
- La potenciación de la Administración electrónica
- El acceso de todos a la Sociedad de la Información.

Éste último apartado contempla, a su vez, iniciativas destinadas tanto a los ciudadanos (acceso y formación de usuarios y profesionales) como a las empresas (incorporación a las nuevas tecnologías y al comercio electrónico) y al conjunto de la Sociedad (España en la Red, a través de contenidos digitales: las lenguas de España, el patrimonio histórico y natural, el turismo y la creación).

El Plan recoge desde medidas de tipo normativo (regulación) hasta actuaciones y proyectos concretos de promoción (el núcleo central del Plan de Acción Info XXI, 126000 millones de pesetas), pasando por la necesaria inversión que ha de

realizar la Administración para convertirse en una verdadera Administración electrónica (600000 millones de pesetas).

A este gasto hay que añadir las partidas específicas que, para el desarrollo del Sector TIC y de la Sociedad de la Información, destinará el Ministerio de Ciencia y Tecnología en ese periodo y que alcanzarán los 225.000 millones de pesetas.

3 Profesionales de la Sociedad de la Información

La anterior enumeración de iniciativas encaminadas a la implantación de la Sociedad de la Información en Europa y en España es fruto del convencimiento de los gobiernos de la necesidad de crear las condiciones óptimas para que tal implantación tenga lugar dentro de un marco temporal cercano.

Paralelamente al convencimiento del carácter estratégico que aporta la Sociedad de la Información, se constata la necesidad perentoria de formar a los profesionales responsables de su desarrollo. Si la Sociedad de la Información es aquella en la que una parte más importante del bienestar y la riqueza generada dependen de la producción, la transformación, la difusión y el consumo de la información mediante dispositivos informáticos y redes de telecomunicaciones, el desarrollo de la Sociedad de la Información implicará una necesidad de un nivel cada vez mayor de conocimientos tecnológicos por parte de las personas que produzcan, transformen, difundan y utilicen dichas informaciones.

Como botón de muestra de la mencionada necesidad, baste citar las cifras que arrojaba un estudio de IDC, según el cual medio millón de puestos de trabajo con perfil profesional de las TIC quedaron vacantes en Europa en 1998 y para el 2002 se preveía que esta cifra superaría los 2,3 millones de especialistas [5].

Efectivamente, el crecimiento progresivo de la Sociedad de la Información nos está conduciendo a una sociedad cada vez más necesitada de conocimientos relacionados con las TIC. Esta necesidad afectará directamente al ciudadano, en su calidad de ususario de la información. Igualmente, afectará al ciudadano en su actividad laboral como consecuencia de la introducción de equipos informáticos y de telecomunicaciones en su puesto de trabajo. Y, por último, esta nueva situación afectará también a los profesionales que tengan que participar directamente en los procesos de desarrollo, implantación y gestión de todos aquellos productos y procesos específicos de la Sociedad de la Información. Una buena parte de estos profesionales serán titulados universitarios y la formación que reciban será decisiva en el proceso

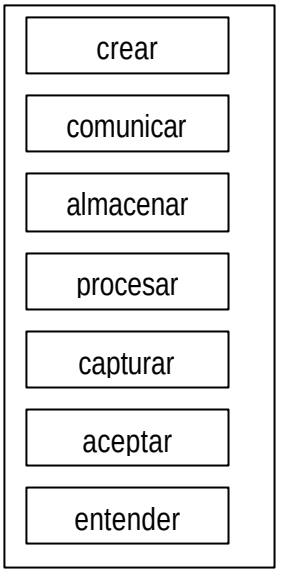


Figura 1: Procesos sobre la información

de desarrollo, implantación y gestión de la Sociedad de la Información.

A la hora de identificar cuáles son los profesionales de la Sociedad de la Información, consideramos útil partir de la identificación de los procesos a los que se ve sometida de forma genérica toda información, los cuales entendemos que lógicamente en el marco de la emergente Sociedad de la Información se verán potenciados y, por tanto, también lo serán las profesiones más vinculadas con tales procesos. Tales procesos se muestran en la Fig. 1. Las profesiones actuales más vinculadas con los procesos que se muestran serían:

- El Ingeniero de Telecomunicación, profesional de la comunicación de la información.
- El Ingeniero Informático, del almacenamiento, procesado y captura de la información.
- El profesional del audiovisual, de la creación, la aceptación y el entendimiento de la información.

Así pues, el debate que aquí abordamos no es específico de la formación del Ingeniero de Telecomunicación pero lo centraremos en él.

La enumeración de profesionales de la Sociedad de la Información anterior es muestra del carácter intrínsecamente multidisciplinar de la Sociedad de la Información, el cual dará pie, como ya lo está dando, a que concurran profesionales con muy diversos perfiles curriculares para desempeñar las funciones de los "Ingenieros de la Sociedad de la Información", a saber, el desarrollo, la implantación y la gestión de la Sociedad de la Información [4].

Sin embargo, en nuestra opinión, los Ingenieros de Telecomunicación disponen de una considerable ventaja competitiva respecto de otros profesionales para llegar a convertirse en los futuros Ingenieros de la Sociedad de la Información. Esta ventaja consiste básicamente en que poseen un conocimiento sólido de las tecnologías de soporte de los dispositivos y los sistemas de la Sociedad de la Información.

Por el contrario, estimamos que, como se desprende de los objetivos de las acciones estratégicas estudiadas en la sección anterior, en el contexto de la Sociedad de la Información, el interés se encuentra gravitando en torno a las Aplicaciones, mucho más que sobre las tecnologías concretas que las soportan. En otras palabras, la tecnología sólo tiene interés en la medida en que se traduce en aplicaciones útiles para los usuarios.

4 La formación universitaria para la Sociedad de la Información

Después de haber dibujado el escenario en el que se mueve el papel del Ingeniero de Telecomunicación dentro de la Sociedad de la Información, nos proponemos abordar algunos aspectos que puedan reflejar la situación de la Universidad en el proceso de la formación del Ingeniero de la Sociedad de la Información [6].

En primer lugar, cabe preguntarse si la Universidad española está realmente en condiciones de formar, de manera adecuada, al Ingeniero de la Sociedad de la Información.

En este sentido, es patente la velocidad inusitada a la que está evolucionando la tecnología en la actualidad. Este hecho reviste una especial relevancia para la formación de los Ingenierios de la Sociedad de la Información. La razón es la siguiente: los procesos tradicionales de aprendizaje del profesorado universitario —a saber, estudio en los libros, actividades de investigación— aparecen como absolutamente inadecuados cuando se trata de impartir materias de carácter aplicado y en continua evolución, como es el caso de algunos aspectos de la Sociedad de la Información.

Una solución a este problema pasa, en nuestra opinión, por romper la situación actual de estanqueidad del profesorado universitario, de manera que sea posible fomentar, sin traumas, la permeabilidad entre el personal de dentro y de fuera de la Universidad. Pero esta solución choca de plano con el actual blindaje universitario frente a su entorno empresarial y social.

En segundo lugar, cabe preguntarse cómo adecuar las actuales titulaciones universitarias al perfil del Ingeniero de la Sociedad de la Información.

Es cierto que la creación de la Sociedad de la Información conllevará la creación de nuevos puestos de trabajo, pero abusar del término de nuevas profesiones como la solución al problema laboral en la Sociedad de la Información puede conducir a consecuencias negativas y, quizás, dar lugar a actitudes oportunistas. En lugar de ello, consideramos más correcto utilizar el término de nuevos perfiles profesionales, que a nuestro juicio refleja mejor la necesidad de adaptar las actuales titulaciones a las necesidades de la Sociedad de la Información.

En este sentido, debemos ser conscientes de que la enseñanza universitaria se rige por Planes de Estudios, los cuales imponen un retraso desde que se diseñan hasta que la primera promoción de estudiantes llega al mercado laboral de ocho años, que son de todo punto demasiados en el proceso de desarrollo de la Sociedad de la Información.

La solución tendría que pasar por flexibilizar el proceso de introducción de nuevos conocimientos en los Planes de Estudios, sin necesidad de modificarlo, de una forma más dinámica que en la actualidad. Este procedimiento es posible dado que descansa exclusivamente en la iniciativa particular de cada Escuela de Ingenieros de Telecomunicación. No obstante, somos conscientes de la dificultad que un proceso de las características encontraría en la Universidad actual, pues choca con el generalizado carácter patrimonial de las asignaturas que históricamente ha venido asumiendo el profesorado.

Este diagnóstico impone, en nuestra opinión, la necesidad urgente de vencer las inercias tradicionales de la Universidad y de su profesorado.

5 Plan de Acción

Habiendo constatado en este punto la necesidad de orientar la formación del Ingeniero de Telecomunicación hacia la Sociedad de la Información y a pesar de la magnitud de los impedimentos que hemos identificado dentro de la Universidad para su satisfacción, somos conscientes, no obstante, de que son posibles actuaciones puntuales y menos ambiciosas, como las que a continuación detallamos.

5.1 La Escuela de Ingenieros de Telecomunicación de Valencia

En los estudios de Ingeniero de Telecomunicación en la ETSIT de la Universidad Politécnica de Valencia se ha aprovechado la implantación del Plan de Estudios B.O.E. 15/5/1996 para introducir

formación específica, actualizada y dinámica en el ámbito de la Sociedad de la Información.

Se trata de un Plan de Estudios que estructura la optatividad del segundo ciclo en bloques de intensificación, siendo la Intensificación de Telemática el elemento novedoso al respecto. En el Plan de Estudios cada bloque de intensificación queda subdividido en un primer bloque de asignaturas que le son características y en un segundo bloque de materias cuyo desdoblamiento en asignaturas se deja al criterio de la Escuela. La previsión de este procedimiento más dinámico y efectivo para configurar la parte final del currículum universitario de nuestros estudiantes, nos ha permitido reducir el tiempo transcurrido desde que se diseña éste hasta que se implanta (nuestro *time-to-market*) a tres años.

La característica más específica del Plan de Estudios, desde la perspectiva de la Ingeniería Telemática, es la articulación del mencionado segundo bloque de la Intensificación Telemática en los denominados Perfiles [7], que son tres:

- Perfil "Redes Públicas"
- Perfil "Redes Corporativas"
- Perfil "Sociedad de la Información"

El propósito de esta estructuración ha sido doble:

- Constituir grupos de asignaturas orientados a la preparación del estudiante para el ejercicio profesional en un ámbito específico e identificado del sector de la Telemática
- Servir de elementos vertebradores de los contenidos del resto de las asignaturas vinculadas con la Ingeniería Telemática, en una lectura del Plan de Estudios "de atrás hacia delante"

Al respecto del primer propósito, cada Perfil se planteó el objetivo de proporcionar una visión de la Ingeniería Telemática orientada al desempeño de diferentes puestos de trabajo:

- Redes Públicas: Puestos de trabajo en los Operadores de Telecomunicación con licencias y autorizaciones para la implantación de redes y prestación de servicios de comunicaciones de voz, datos e imagen, a través de redes fijas, móviles y de cable.
- Redes Corporativas: Puestos de trabajo de planificación, diseño y explotación de sistemas corporativos de comunicaciones de voz, datos e imagen en empresa.

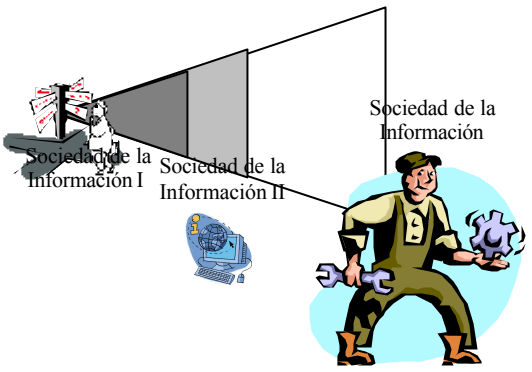


Figura 2: Visión formativa en Sociedad de la Información

- Sociedad de la Información: Puestos de trabajo de planificación, desarrollo y gestión de aplicaciones telemáticas destinadas a usuarios finales, denominadas Aplicaciones de la Sociedad de la Información.

El Perfil "Sociedad de la Información" forma parte de las actividades de la Cátedra Jean Monnet de Política de Telecomunicaciones y Sociedad de la Información de la Universidad Politécnica de Valencia, de la cual el segundo de los autores es titular. Este perfil partió de una visión de la formación necesaria en el último curso orientada fundamentalmente a satisfacer dos objetivos:

- Recopilar, integrar y complementar los conocimientos y habilidades adquiridos a lo largo de la carrera, de forma deslabazada, por parte del estudiante, desde la perspectiva del perfil profesional
- Mostrar la situación del entorno socioprofesional y de mercado en el que se va a desempeñar el perfil profesional, abordando el estado del arte de productos y servicios, así como las características del mercado.

A partir de esta visión, que se intenta plasmar en la Fig. 2, el Perfil se ha materializado en dos asignaturas secuenciales, a saber, "Sociedad de la Información I" y "Sociedad de la Información II".

Los objetivos de la asignatura "Sociedad de la Información I" son:

- Conocer las tecnologías sobre las que se fundamentan las Aplicaciones de la Sociedad de la Información: tecnologías middleware, seguridad y firma digital, sistemas de pago, etc.
- Analizar las Aplicaciones de la Sociedad de la Información desde los puntos de vista estructural, tecnológico y socioeconómico.

Por su parte, los objetivos de la asignatura "Sociedad de la Información II" son:

- Conocer el estado del arte de los productos y de los servicios que componen las Aplicaciones de la Sociedad de la Información.

- Analizar el panorama de la utilización de las Aplicaciones de la Sociedad de la Información así como su incidencia sobre las empresas y sobre los usuarios

Tanto el carácter terminal de las asignaturas de los Perfiles de la Intensificación Telemática como el carácter dinámico y actual que intrínseca y específicamente tiene el Perfil "Sociedad de la Información", determina una metodología docente caracterizada por el uso intensivo de fuentes de información principalmente distribuidas en la Red y por el recurso frecuente a profesionales del sector. De esta manera, se es fiel a la visión de los perfiles como preparatorios para el ingreso en un campo profesional en definición, así como al convencimiento de que sólo promoviendo la transferencia de conocimiento desde el sector hacia la Universidad conseguiremos adaptar eficazmente la formación del Ingeniero de Telecomunicación a los nuevos perfiles profesionales que requiere la Socieda de la Información[1].

5.2 La formación integral en Sociedad de la Información

La experiencia anterior se inserta en un escenario muy concreto, con un plan de actuación orientado a mejorar la formación de aquel grupo reducido de estudiantes que configuran su currículum universitario hacia la Ingeniería Telemática y, en particular, hacia las Aplicaciones de la Sociedad de la Información.

Siendo consecuentes con la argumentación que hemos desarrollado hasta este punto, entendemos que son posibles actuaciones de mayor alcance que la que los autores han llevado a cabo en la Universidad Politécnica de Valencia.

Se trataría de elevar el nivel de compromiso que en este documento proponemos, desde un grupo de profesores hasta una Escuela de Ingenieros de Telecomunicación.

Efectivamente, en un mapa universitario estatal constituido por 17 ETSITs [8], empieza a ser sensato plantearse la oportunidad de la diferenciación formativa, en aras de captar a los mejores estudiantes y de atraer la atención de las empresas más potentes del sector. Este planteamiento no hubiera tenido sentido ni aceptación hace una década, con un número reducido de Escuelas en nuestro país y con un régimen de explotación de las telecomunicaciones poco diverso.

Pues bien, apostar por la formación del Ingeniero de Telecomunicación en Sociedad de la Información pasaría, en nuestra opinión, por ofrecer un Plan de Estudios que configurara su optatividad en orden a conseguir un Ingeniero de la Sociedad de la Información, una vez garantizada la formación básica del Ingeniero de Telecomunicación por la troncalidad según establece el R.D. 1421/1991. Este tipo de iniciativas sólo puede acometerse desde Universidades que no se vean lastradas por las rigideces a las que hemos hecho mención anteriormente, situación que únicamente podría darse en aquellas sin estudios de Ingeniero de Telecomunicación en la actualidad.

Quede constancia de nuestro apoyo y colaboración con una iniciativa de esta índole.

6 Conclusiones

En nuestra opinión, la Sociedad de la Información supone para la Universidad, como también lo está suponiendo en otros ámbitos de nuestra sociedad, un desafío estratégico de primera magnitud. El Ingeniero de Telecomunicación está en condiciones de jugar un papel fundamental en el desarrollo de la Sociedad de la Información, y las Escuelas de Ingenieros de Telecomunicación deben desempeñar un papel decisivo para la incorporación no traumática de los recién titulados a este nuevo sector.

Entendemos que el camino para que la Universidad española acabe, de verdad, estando en condiciones de dar una respuesta válida a las exigencias de formación del proceso de implantación de la Sociedad de la Información deberá pasar, entre otras cosas, por fomentar la permeabilidad de su personal docente con el de la empresa y por flexibilizar el procedimiento para la fijación de los contenidos de los planes de estudios.

Confiamos, asimismo, en el acierto de nuestra visión estratégica de la formación del Ingeniero de Telecomunicación, por cuanto las iniciativas que hemos llevado a cabo y las propuestas que hemos planteado hasta la fecha han recibido los parabienes de aquellas instituciones y personas a las que las hemos hecho llegar.

Reiteramos nuestro ofrecimiento a debatir nuestra propuesta así como transferir nuestra experiencia a quien esté interesado.

Agradecimientos

Este trabajo ha sido posible gracias a la Cátedra Jean Monnet de Política de Telecomunicaciones y Sociedad de la Información.

[1] Para más información sobre el perfil, visite: http://www.upv.es/~lguijar/socinfo

Referencias

[1] COM (93) 700. Crecimiento, Competitividad y Empleo. Retos y Pistas para entrar en el siglo XXI. Libro Blanco. Bruselas. 5 de diciembre de 1993.

[2] Telefonica. La Sociedad de la Información en España. Presente y perspectivas 2000. Julio 2000. ISBN: 84-89900-24-8.

[3] M. Castells. La Era de la Información. Economía, Sociedad y Cultura. Vol. 1: La Sociedad Red. Alianza Editorial. Mayoy 1998. ISBN: 84-206-4247-9.

[4] M. Tomás y A. Alabau. "De la Ingeniería Telemática a la Ingeniería de la Sociedad de la Información". II Jornadas de Ingeniería Telemática JITEL'99. Leganés, 93-96 Septiembre 1999. ISBN: 84-89315-14-0.

[5] J.C. Ambrojo. "La industria se une para paliar el déficit de técnicos en Europa". Ciberp@ís. El País. 9 de diciembre de 1999.

[6] A. Alabau. "La formación universitaria para la Sociedad de la Información". BIT nº 121. Madrid, 6-8 Mayo-Junio 2000.

[7] A. Alabau. "Propuesta sobre el desarrollo de la intensificación Telemática del nuevo plan de estudios". Documento de trabajo, nº 16. Valencia, Febrero 1999.

[8] Varios autores. "En el umbral del siglo XXI". BIT nº 124. Madrid, 6-8 Noviembre-Diciembre 2000.

INTECA: Infraestructura para Tele-Educación y su implementación mediante un portal vertical.

KLAUS D. HACKBARTH, JOSÉ ANTONIO PORTILLA, ROBERTO ORTIZ
Grupo de Ingeniería Telemática / Departamento de Ingeniería de Comunicación
Universidad de Cantabria
Avda. Los Castros s/n, 39005 Santander (CANTABRIA)
Telf: 942-201392 Ext.22, Fax: 942-201488
E-mail: klaus@tlmat.unican.es, jantonio@tlmat.unican.es , rortiz@tlmat.unican.es

Abstract. *This paper shows an overview about the INTECA (Tele-Education Infrastructure) project developed in the spanish national framework program PISTACABLE. The aim of INTECA is to develop an infrastructure and basic contents for multimedia tele-education, with open access regardless of the type of connection (modem, cable modem, ADSL, etc). The paper describes the basic applications running currently on the server, like videoconference, audio and video streaming, SMIL-based slideshows, on-line help. Additionally the paper indicates the applications currently under implementation, wich are: voice and text chats, video broadcast, document-exchange system and a corresponding basic course management system. Finally the paper propose a generic structure for multimedia course design.*

1 Introducción

Uno de los tópicos que mas están resonando actualmente en el mundo de las telecomunicaciones es el de la tele-educación. Este concepto se entiende como enseñanza a distancia, reglada o no reglada. El objeto de la tele-educación es, por lo tanto, el proporcionar de manera remota contenidos que precisan de un cierto grado de interactividad entre el instructor y el alumno [1] .

Existen gran cantidad de información respecto a como debe implementarse la tele-educación, principalmente en estándares como el "IMS project" [2]. Aunque ya existen algunas implmementaciones, éstas no cumplen las expectativas. Por un lado, existen muchas propuestas en forma de cursos multimedia en cd, no destinados a la educación "en línea" a través de Internet. Por otro lado, los desarrollos comerciales realizados en el campo de la tele-educación a través de internet presentan un esquema demasiado rígido a la hora de incluir contenidos, especialmente cuando éstos son de tipo multimedia.

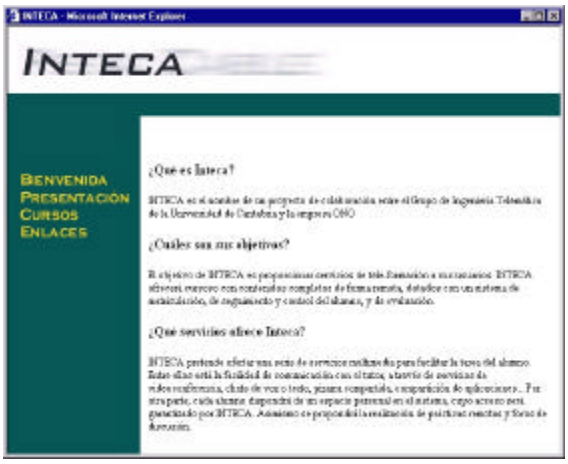
Tomando como punto de partida estas implementaciones prácticas de tele-educación, y con el objetivo de construir un servicio completo que solvente las carencias de los mismos, el Grupo de Ingeniería Telemática de la Universidad de Cantabria, en colaboración con el operador de cable "ONO", está desarrollando el proyecto INTECA (INfraestructura para TEle-eduCAción) que se engloba dentro de los proyectos PISTACABLE de la Secretaría General de Comunicaciones. El objetivo es proporcionar en primera instancia infraestructuras y servicios de tele-formación al propio Departamento de Ingeniería de Comunicaciones así como para docencia y para la creación de un laboratorio virtual.

En una segunda etapa se pretende extender este servicio a todos los usuarios y todos los tipos de accesos. Obviamente un mejor aprovechamiento de los servicios lo realizarán los usuarios que dispongan de un acceso de banda ancha (Cable-modem, LAN, ...). Con ello se entraría en un sector del mercado que no solo accedería a la tele-educación como alumno, sino que en el caso de academias, como instructores, proporcionando a su vez cursos que implementar en el servicio de tele-educación.

2 Implementación e infraestructura

El servicio de tele-educación de INTECA se encuentra implementado en forma de un portal vertical [3]. Este portal se encuentra instalado en dos ordenadores Pentium III con memoria RAM mínima de 256Mb y dos discos duros SCSI de 10 Gbytes cada uno. Ambos servidores se encuentran protegidos mediante sistemas de alimentación ininterrumpida S.A.I. El acceso a los mismos se realiza mediante sendas tarjetas de red Ethernet a 100Mbps, conectadas a un switch a la misma velocidad.

[Figura 1]: Entrada al proyecto PISTACABLE



[Figura 2]: Entrada a la sección de tele-educación

Con el objeto de realizar una comparación entre los sistemas operativos, se ha instalado en uno de ellos (Metis) el sistema operativo LINUX [4], SUSE con el servidor Web APACHE [5]versión 1.3. El otro servidor (Amaltea) tiene un sistema operativo Windows NT Server 4.0 con servidor Web IIS. Todos los desarrollos realizados han tenido en cuenta que debían mantener la compatibilidad con estos dos sistemas operativos.

3 Aplicaciones Básicas

Se definen como aplicaciones básicas a aquellas funcionalidades que son imprescindibles para ofrecer el servicio de tele-educación. Estos servicios se encuentran actualmente instalados en los servidores, insertados tanto de manera genérica en las páginas web correspondientes a un demostrador del servicio de tele-educación, como en dos cursos actualmente en desarrollo. Estas aplicaciones se pueden dividir en dos categorías: Aplicaciones que proporcionan contenidos y aplicaciones de comunicación.

Las aplicaciones básicas que proporcionan contenidos son las siguientes:

3.1 Video y audio almacenado : Se han instalado fragmentos de vídeo y audio de corta duración para ser descargados por los usuarios o alumnos de un determinado curso. Su objetivo es ahondar de manera más profunda en aspectos concretos de una lección determinada. El formato utilizado es Real Media, por su relación calidad / tamaño de los ficheros.

3.2 Video y audio en formato streaming: Destinado a fragmentos de audio y vídeo de larga duración que el usuario puede visualizar o escuchar sin tener que descargarlo ni esperar a realizar una carga remota total. Su objetivo son explicaciones de larga duración. Al igual que antes, el formato elegido es Real Media.

3.3 Sistema de ayuda on-line: Especializado para cada sección de cada curso. Su cometido es solucionar problemas técnicos y de servir como guía en la utilización del servicio. Este sistema de ayuda está complementado mediante audio sincronizado (utilizando ficheros en formato Real Media)

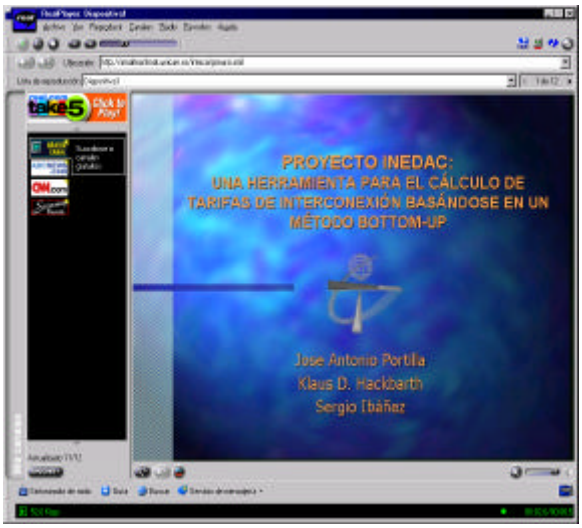
[Figura 3]: Representación del video en formato almacenado y streaming

[Figura 4]: Sistema de ayuda on-line

3.4 Diapositivas sincronizadas con audio: Mediante el lenguaje SMIL se permite realizar la sincronización de imágenes con audio en modo streaming [6].

3.5 Sistema de protección basado en scripts PHP: Como primera instancia en un sistema de gestión de alumnos se ha implementado una protección de acceso basada en una pareja de login / password [7].

3.6 Sistema de video-cámara con control de movimiento para la transmisión de clases no presenciales: Como primer estudio para un sistema multicast de transmisión de video y audio.



[Figura 5]: Diapositivas en formato SMIL

[Figura 6]: Sistema de Autenticación

[Figura 7]: Videoconferencia con un desarrollador del proyecto

En cuanto a los servicios de comunicación se encuentran actualmente implementados los siguientes.

3.7 Videoconferencia: Se ha implementado un servicio de videoconferencia basado en NetMeeting con uno de los desarrolladores del proyecto. Este servicio proporciona un alto nivel de interactividad entre los alumnos y los instructores puesto que permite comunicaciones en tiempo real.

3.8 Correo Electrónico: Método de comunicación básica para cuando no se precisa instantaneidad en las contestaciones por parte del instructor del curso a las preguntas o requerimientos de los alumnos.

El correcto funcionamiento de estos servicios ha sido verificado mediante un completo patrón de test. Este patrón esta basado en un cuestionario que ha sido rellenado por los usuarios del servicio, que actualmente son los propios desarrolladores. Con el fin de probar la fiabilidad de las aplicaciones y su versatilidad la batería de pruebas se ha realizado desde los diferentes tipos de acceso que se detallan a continuación

- Acceso LAN: Realizado con ordenadores colocados en el mismo y diferentes segmentos al que se encuentran los servidores de la red departamental

- Acceso ADSL: Desde un cliente ajeno a la Universidad de Cantabria.

- Acceso RTC: Desde diversos clientes con diversos ISP's.

- Acceso Cable: Mediante um MODEM de Cable de un cliente del Operador de Cable ONO.

4 Portal avanzado

Dentro del proyecto INTECA se ha quedado en definir como portal avanzado al portal formado por las aplicaciones que o bien necesitan un mayor trabajo para su instalación y funcionamiento o bien todavía se encuentran en proceso de investigación. Nuevamente estas aplicaciones se dividen en las dos categorías de servicios que proporcionan contenidos y servicios de comunicación.

Estos nuevos servicios se están implementando en un portal que ya tiene su estructura definitiva, junto a dos cursos, el primero de ellos sobre "cómo crear un curso para el portal Inteca", y el segundo sobre "planificación de redes de telecomunicación". Los nuevos servicios son los siguientes.

4.1 Sistema de broadcast de video: Con la finalidad de proporcionar un sistema para la impartición de clases remotas en tiempo real a varios usuarios se está estudiando la creación de un sistema de multicast de vídeo. Las primeras líneas de trabajo están orientadas a la utilización de servidores de streaming comerciales del tipo Real Server que funcionan basándose en el protocolos IP multicast.

4.2 Establecimiento de un método para el intercambio de material docente: Para cada uno de los cursos establecidos dentro del portal se instalará un área para el intercambio de material docente. Este área estará basado en un script de PHP y ofrecerá un servicio de FTP integrado en las páginas Web.

4.3 Desarrollo de herramientas para la creación de contenidos multimedia: Para facilitar la creación de contenidos a los desarrolladores de cursos que no tengan experiencia en la edición de audio, video y sesiones de diapositivas se facilitará un conjunto de programas que hagan más sencilla estas tareas. En la actualidad, se ha desarrollado ya un editor (SmilGen) que permite generar fácilmente presentaciones con diapositivas y audio sincronizado.

4.4 Herramientas para el cálculo remoto: Una de las aplicaciones más interesantes es la posibilidad de realizar un laboratorio virtual donde se puedan ejecutar de forma remota cierto tipo de aplicaciones. El objetivo sería poder dotar de flexibilidad a las sesiones de prácticas de cada curso en concreto. Actualmente se dispone ya de varios applets de Java desarrollados para la creación de un laboratorio virtual de Planificaión de Redes.

En cuanto a las nuevas formas de comunicación se están contemplando las siguientes:

4.5 Creación de foros de discusión: Se establecen foros de discusión basados en correo electrónico a los que se accederá en dos modos: Mediante el cliente de correo electrónico, típico ejemplo de los servidores de noticias ("news"), o mediante el acceso vía web. Estos foros de discusión serán individuales por cada curso, aunque también existirán foros generales donde debatir aspectos globales del portal.

4.6 Creación de salas de "chat": En versiones de voz y texto, permitirán a los participantes en los cursos intercambiar opiniones. Se crearán salas de chat restringidas para alumnos, profesores y general por cada curso. En todas ellas existirá un participante moderador que en general será el responsable del curso en concreto. Actualmente se encuentra en fase de implementación, utilizando scripts PHP y la base de datos MySQL para su funcionamiento.

La inclusión de aplicaciones avanzadas y de nuevas formas de comunicación provocará un alto tráfico que por otra parte no se encuentra caracterizado convenientemente. Por esta razón paralelamente al desarrollo de estas aplicaciones se estudiarán modelos y métodos para la caracterización de tráfico multimedia con el objetivo de proporcionar a un posible operador de red que ofreciese el servicio de tele-educación, una serie de herramientas que le permitiesen el correcto dimensionado de la red de acceso a los servicios y de la granja de servidores que se necesita

Concretamente se estudiarán:

Trafico Web clásico: Cuyas características principales son su gran dependencia estocástica y el ser un tráfico de ráfagas con largas pausas.

Nuevas formas de tráfico Web: De tipo streamnig de audio y/o video y multimedia.

Para estos estudios se utilizarán métodos analíticos, mediante modelos de procesos estocásticos, y simulación mediante herramientas como OPNET, COMNET o simuladores propios basados en XjChart.

5 Estructura y Contenidos

En la introducción de este artículo se ha comentado que una de las posibles aplicaciones clave para la tele-educación es la participación de academias privadas que ofrezcan cursos remotos. Estos cursos tendrán, por lo general, una temática variada y su estructura, a priori, diferirá mucho entre uno y otro.

Es, por lo tanto, de prioritaria necesidad el establecer una estructura fija para la disposición de los contenidos dentro del portal, así como de un árbol de exploración en el que se distribuyan por segmentos todas los servicios que en el se ofrecen. Por otra parte, y con el fin de facilitar la inclusión de los contenidos que proporcionen los creadores de los cursos, estos se deben proporcionar con unos formatos adecuados.

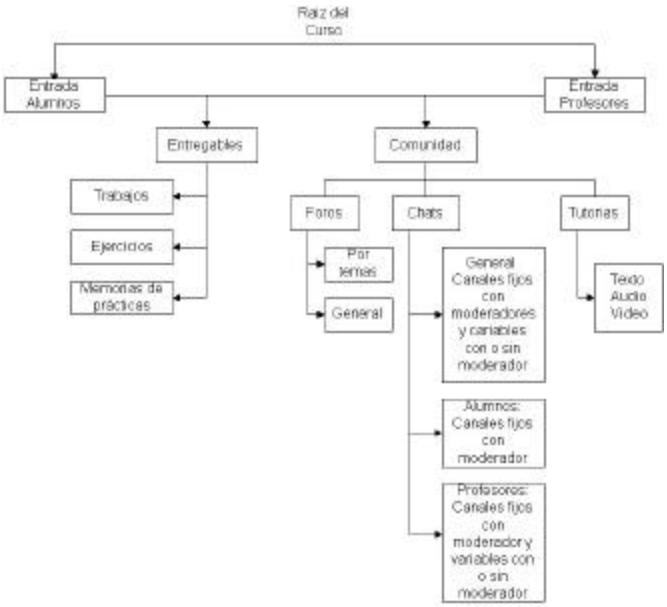
El árbol de exploración del portal, su estructura, debe de cumplir tres características :

Generalidad : Debe contemplar la mayor cantidad de áreas posibles

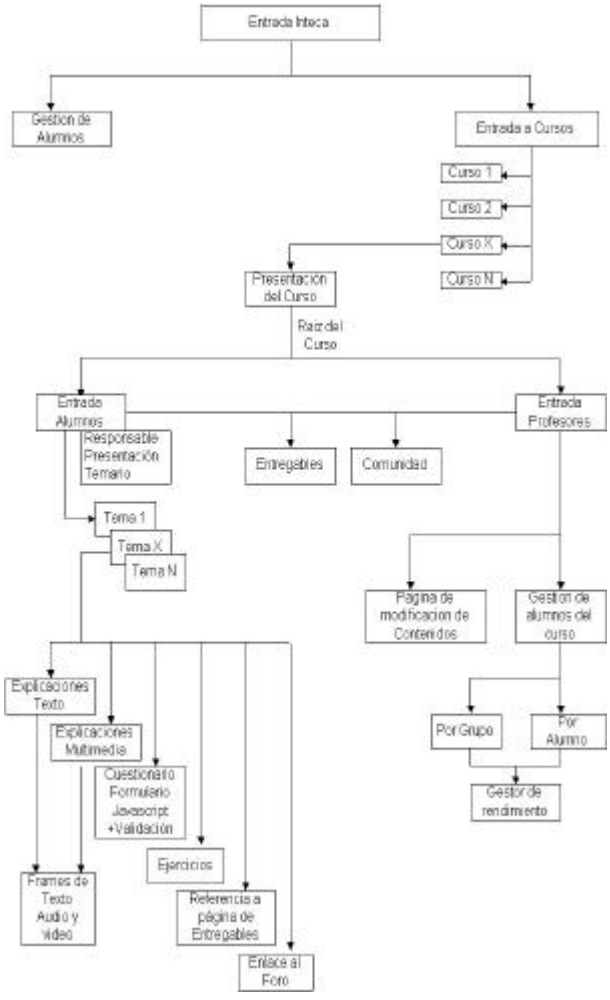
Precisión **:** Definir de manera unívoca la posición de servicios y contenidos

Flexibilidad **:** Adaptable a cambios y nuevos servicios.

Conforme a estas premisas el G.I.T ha desarrollado la estructura del portal que se muestra a continuación. [1]



[Figura 8]:Estructura del curso



1 Debido a su gran tamaño no es posible disponer toda la estructura en una sola figura, con lo cual se encuentra dividida en dos secciones

6 Gestión de Alumnos

Un aspecto fundamental del servicio de tele-educación es el mecanismo de gestión de usuarios. Este debe ser lo suficientemente flexible para incorporar las siguientes tareas.

6.1 Mecanismos de acceso seguro por cada usuario

6.2 Mecanismos de acceso seguro por grupo de usuarios

6.3 Mantenimiento de perfil de usuario

6.4 Mantenimiento de perfil de grupo de usuarios.

A la hora de implementar este mecanismo surgen dos cuestiones principales, la base de datos que se va a utilizar, y que mecanismo de acceso a la misma se debe usar.

La base de datos con la que se está trabajando es MySQL. Se ha escogido esta base de datos debido a su flexibilidad, su generalidad, y que se encuentra disponible para versiones LINUX y WINDOWS.

En cuanto al protocolo de acceso a la base de datos actualmente se están desarrollando scripts en PHP

con los que poder hacer llamadas en lenguaje SQL al servidor de la base de datos MySQL.

7 Requerimientos de los clientes.

Para poder obtener acceso a todas las aplicaciones se necesitan unos requerimientos en cuanto al conjunto de programas clientes que precisa el ordenador desde el que se acceda.

Concretamente se precisa que el ordenador cliente disponga de los siguientes programas

7.1 Internet Explorer versión 5.0 o superior: Se precisa además que disponga de los plug-in necesarios para la visualización de imágenes creadas con Flash 4.0. El motivo de que se precise el navegador Explorer en detrimento de otros es debido a que la aplicación de Videoconferencia no puede ser lanzada desde otro navegador.

7.2 Real Player Basic Versión 8.0: Necesario para la visualización del vídeo y audio almacenado o en streaming. [8]

7.3 Cliente de correo electrónico: Con el fin de poder realizar comunicaciones de correo electrónico.

7.4 NetMeeting v3.01: Para poder realizar videoconferencia con uno de los desarrolladores del proyecto.

8 Conclusiones

INTECA pretende ser una alternativa abierta y de coste bajo frente a los sistemas de tele-educación comerciales, no abiertos y de altos costes Aunque aún se encuentra en fase de desarrollo, las pruebas ya realizadas en diferentes entornos de redes indican la viabilidad de la solución tanto en un servidor Windows como Linux. A partir del curso 2001/2002 esta previsto la aplicación real para las partes de asignaturas virtuales ofrecido por el GIT-UC y el laboratorio virtual. El acceso para el alumno se proporcionará tanto desde las aulas informáticas correspondientes de la UC como desde su casa vía un cable módem y la red de cable ONO. Para el último el operador ONO está planificando una conexión directa con la red de campos de UC.

9 Referencias

[1] "Creating educational Web Sites", Junichi Azuma, University of Marketing and Distribution Sciences. IEEE Comunications Magazine March 1999.

[2] http://www.imsproject.org Información sobre el estándar IMS.

[3] Dirección del portal de tele-educación: http://amaltea.tlmat.unican.es . Servidor alternativo de respaldo: http://metis.tlmat.unican.es .

[4] http://www.linux.box.sk

[5] The Apache Software Foundation, - http Server Poject. http://httpd.apache.org

[6] W3C Synchronize Multimedia http://www.w3.org/AudioVideo

[7] "Authenticate and Track Users with PHP", Julie Meloni, http://hotwired.lycos.com/webmonkey/00/05/index2a.html?tw=programming

[8] Distribuciones de software de Realnetworks. http://www.real.com , http://www.realnetworks.com

Implantación de un Laboratorio Docente para Redes de Comunicaciones

Francisco Javier Ruiz[1], David Fernández[1], Ana B. García[1], Fernando Muñoz[1], Luis Bellido[1], José I. Moreno[2]

[1]Departamento de Ingeniería de Sistemas Telemáticos. Universidad Politécnica de Madrid.
ETSI Telecomunicación. Ciudad Universitaria s/n. 28040 Madrid.
E-mail: {fruiz, david, abgarcia, fmunoz, lbt}@dit.upm.es

[2]Departamento de Ingeniería Telemática. Universidad Carlos III de Madrid.
Avda. de la Universidad 30. 28911 Leganés (Madrid).
E-mail: jmoreno@it.uc3m.es

Abstract. *This paper describes some experiences in setting up and running a communications network teaching laboratory. The laboratory is used both for teaching graduate courses in networking and for post-graduate courses in ETSI Telecomunicación of Universidad Politécnica de Madrid. The main objectives of the laboratory are presented, and then its structure is reviewed to show the diverse scenarios available. The laboratory can support exercises of configuration, usage, monitoring and performance evaluation of real network scenarios. In addition, simulation tools are also supported. Finally, some guidelines for evolution of this laboratory in the future are presented in the paper.*

1. Introducción

El artículo describe la implantación de un laboratorio docente para redes de comunicaciones en el Dpto. de Ingeniería de Sistemas Telemáticos (DIT) de la E.T.S.I. Telecomunicación de la Universidad Politécnica de Madrid. El ámbito de uso de este laboratorio es la docencia en redes de comunicaciones, tanto en asignaturas de grado como para formación de postgrado.

El laboratorio de redes está soportado por la infraestructura general de laboratorios del DIT. Los puestos de laboratorio son ordenadores tipo PC, algunos de los cuales tienen hardware adicional instalado para su uso en el laboratorio de redes de comunicaciones. Además, varios laboratorios docentes, no solamente de comunicaciones, hacen uso de esta infraestructura, por lo que se ha implantado con los criterios de proporcionar la máxima flexibilidad posible con la mayor facilidad de mantenimiento. Concretamente, en el entorno actual es posible disponer de las siguientes facilidades:

- Soporte de red local Ethernet conmutada a 100 Mbps.
- Arranque de varios sistemas operativos (Linux / Windows 95/98/NT).
- Arranque de los sistemas directamente de una imagen disponible en la red, o bien tener la posibilidad de regenerar el sistema operativo de la máquina a partir de una imagen accesible por la red.
- Servicios generales de red: acceso a Internet a través de un proxy, correo electrónico, impresión, etc.
- Reserva de puestos en línea por parte de los alumnos.
- Acceso a la documentación de los diferentes laboratorios vía red.

En el área dedicada a la impartición de los laboratorios de comunicaciones se dispone en la actualidad de un total de 56 puestos.

2. Objetivos del Laboratorio de Redes de Comunicaciones

El laboratorio de redes de comunicaciones objeto de este artículo ha sido implantado sobre el entorno descrito previamente, y pretende cubrir las siguientes áreas:

- Estudio, planificación y diseño de redes de comunicaciones basado en herramientas de simulación. Se dispone de las herramientas de simulación Network Simulator (NS) [3] y COMNET III. Ambos entornos de simulación ofrecen bibliotecas con funciones de simulación de protocolos y redes de comunicaciones.
- Configuración de equipos de comunicaciones. Se dispone de una serie de entornos de red con infraestructura física instalada en el laboratorio, que abarcan los siguientes ámbitos: LAN, Frame Relay, RDSI, ATM. Estos entornos están soportados por equipos de comunicaciones de diversos fabricantes: TELDAT[6], Cisco[7], RAD[8], Fore[9], 3Com[10], etc. Asimismo en el

laboratorio se dispone de puestos equipados con hardware que permite la conexión a dichos entornos de red. Los alumnos pueden de esta forma realizar prácticas de configuración tanto de los puestos de usuario como de los equipos de red.

- Gestión y monitorización de red. Se dispone de una herramienta de gestión SNMP comercial (HP OpenView [4]), de analizadores de protocolo comerciales (Agilent[11]) y de libre distribución (Ethereal[12]), así como de diversas herramientas de monitorización.

Esta disponibilidad de entornos, combinada con la disponibilidad de puestos y la posibilidad de realización de reservas de puestos vía red, permite la posibilidad de realizar dos tipos de prácticas, en función de cómo se organiza la asistencia al laboratorio:

- Prácticas de asistencia abierta al laboratorio, en la que los alumnos disponen de un plazo amplio para la realización de las prácticas, y pueden acudir al laboratorio durante ese plazo, disponiendo de la facilidad de reservas de puesto para la utilización de equipos concretos.
- Prácticas de asistencia controlada al laboratorio en horario concreto y limitado, en las que hay presencia del profesorado para la realización de prácticas monitorizadas o guiadas con asistencia de todo el grupo de alumnos.

El primer tipo de prácticas se adapta más a las asignaturas de grado, permitiendo a los alumnos planificar la asistencia al laboratorio de acuerdo con sus necesidades. El segundo tipo se adapta más a los cursos de postgrado, que tienen un horario mucho más limitado.

Con esta capacidad se da soporte a la docencia de las siguientes materias:

- Laboratorio de Ingeniería de Redes y Servicios Telemáticos[1]. Se trata de una asignatura de grado, de quinto curso, optativa de la especialidad de Telemática del Plan 94 de los estudios de Ingeniería de Telecomunicación. Sus objetivos son la realización de prácticas orientadas a la conexión de estaciones a diversas redes para acceder a servicios, a la interconexión de distintas redes, y al análisis de sus características, capacidades y prestaciones.

- Asignaturas de postgrado de los distintos cursos de maestría organizados por la ETSI Telecomunicación[2], entre ellos, el Máster en Sistemas y Redes de Comunicaciones y el Máster en Comunicaciones Móviles Airtel-UPM.

3. Estructura del laboratorio

Como se ha indicado, en el laboratorio se dispone de una gran variedad de entornos de red para la realización de prácticas. En la Figura 1 se presenta la infraestructura fija que se utiliza en cada uno de los entornos de red utilizables actualmente.

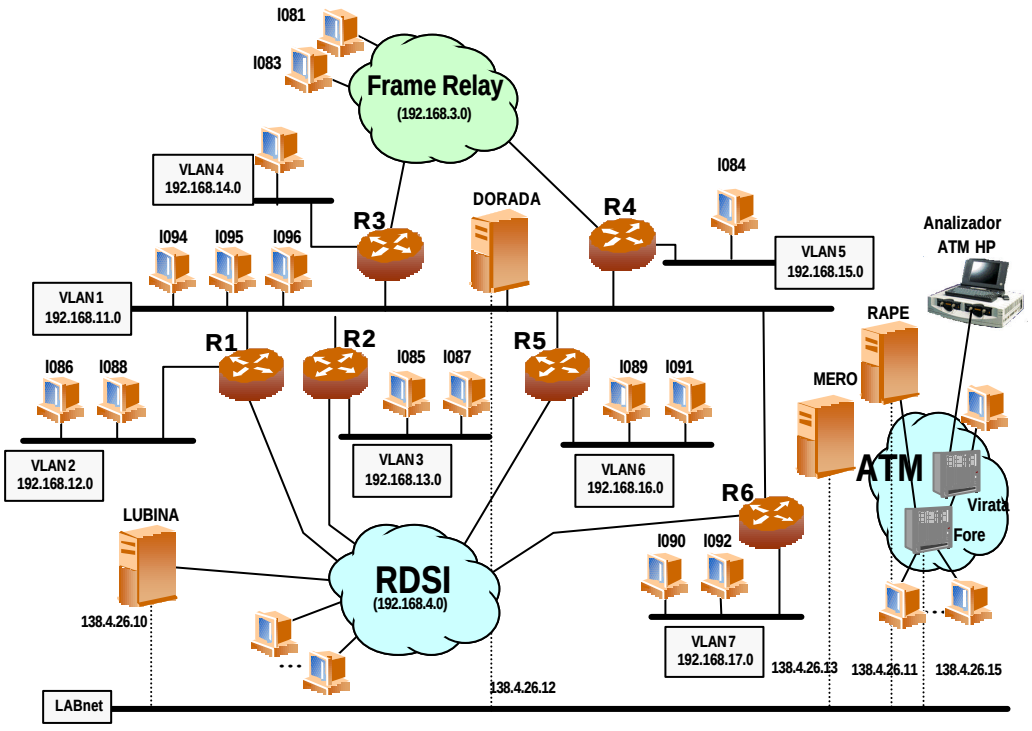


Figura 1: Infraestructura de red del laboratorio

Más en detalle, los distintos escenarios de prácticas que aparecen en la figura están soportados por los siguientes equipos:

- Redes Locales. Los distintos segmentos de LAN que forman la red se han creado mediante varios conmutadores Ethernet con soporte de redes virtuales (VLAN), interconectados mediante routers. La utilización este tipo de conmutadores permite crear escenarios de red complejos sin la necesidad de involucrar un número grande de equipos. Además, añaden una gran flexibilidad a la hora de gestionar y reconfigurar los escenarios de prácticas.
- RDSI y RTB. Basada en dos centralitas con capacidad para extensiones de accesos básicos RDSI y extensiones analógicas.
- ATM. Basado en dos conmutadores ATM con líneas de 25 y 155 Mbps, y que soportan protocolos de señalización estándar y funciones para la configuración de LANE y CLIP.
- Frame Relay. Basado en un conmutador Frame Relay y dos multiplexores de acceso, con soporte para LAN, líneas serie e incluso interfaces de voz.

Además del equipamiento citado, y adicionalmente a la tarjeta Ethernet que poseen los puestos del laboratorio para su conexión a la red de producción, algunos de ellos disponen de otras tarjetas (Ethernet, ATM, RDSI, Frame Relay, etc) para conectarse a los distintos escenarios. La realización de cada práctica exige pues la utilización de unos puestos de prácticas concretos. En el diseño del laboratorio se ha tratado de limitar al máximo esta dependencia, dotando al máximo número de puestos de placas de acceso a cada escenario de red, con el objeto de maximizar la utilización de los equipos más escasos (routers, conmutadores, etc). Sin embargo, algunos factores como la limitación del número de placas soportadas por cada PC (actualmente se soportan como máximo 3 en nuestro entorno: la Ethernet de producción y dos tarjetas adicionales) dificultan la consecución de este objetivo.

Además de la infraestructura presentada, que se utiliza en las prácticas básicas de interconexión y que no exige que los alumnos tengan que realizar conexiones físicas o manipular físicamente equipos, es posible desplegar otros escenarios de red pensados para la realización de prácticas de interconexión más avanzadas, tal como se describe en la siguiente sección.

La Figura 2 presenta la organización de los equipos fijos de comunicaciones del laboratorio. Como puede apreciarse, todo el equipamiento, tanto el perteneciente a la red de producción como el dedicado a entornos de red experimentales, se concentra en dos armarios dotados de "racks" estándar de 19". A la izquierda de la imagen, o con más detalle en la Figura 3, puede apreciarse uno de los aspectos clave de la instalación: el gran número de cables que llegan a los armarios procedentes principalmente de los PCs de prácticas.

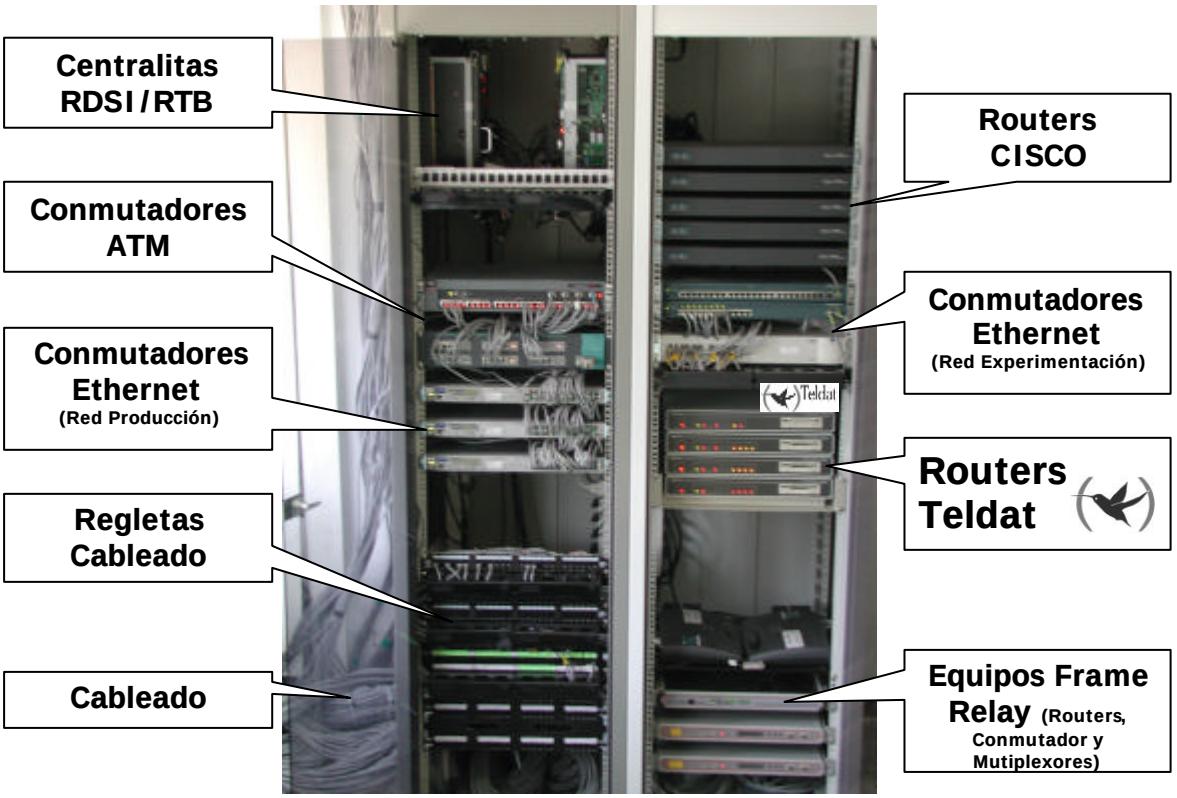


Figura 2: Armarios de Equipos del Laboratorio

Figura 3: Detalle del cableado

Con el objeto de dotar al laboratorio de la mayor flexibilidad y versatilidad posible, cada puesto de prácticas tiene instalado tres cables UTP. Dado que, además, la infraestructura se utiliza también desde otros laboratorios de postgrado (unos 40 puestos adicionales), y que, por falta de espacio algunos equipos de comunicaciones están situados en el otros lugares, el número de cables a manejar supera los 300. Este hecho exige un diseño muy cuidadoso a la hora de situar los distintos equipos dentro de los armarios.

Finalmente mencionar que las herramientas utilizadas en las prácticas de simulación, dado que no exigen la utilización de ningún equipamiento adicional, pueden ser ejecutadas desde cualquier puesto del laboratorio.

4. Descripción de las Prácticas

A continuación se clasifican y describen brevemente los aspectos y los objetivos más importantes de cada una de las prácticas que se plantean en nuestro laboratorio de redes de comunicaciones.

Las prácticas realizadas hasta la fecha son de dos tipos: prácticas de simulación y prácticas sobre equipos. Respecto a las prácticas sobre equipos, éstas se pueden clasificar a su vez en prácticas de configuración, prácticas de interconexión y prácticas de operación y monitorización.

4.1. Prácticas de Simulación

Desde un punto de vista genérico, las prácticas de simulación se presentan como un complemento importante sobre las prácticas con equipos, proporcionando al alumno la oportunidad de experimentar tanto con escenarios de red como con tecnologías que o bien son difíciles de implantar en un laboratorio o bien suponen un gasto económico difícil de abordar.

Para la realización de estas prácticas se utilizan dos herramientas de simulación diferentes. Por un lado, se utiliza una herramienta de libre distribución denominada *ns* (*N*etwork *S*imulator). Se trata de un software de simulación basado en eventos discretos fundamentalmente utilizado en entornos de investigación sobre redes de comunicaciones y que puede funcionar tanto bajo plataformas *GNU-Linux* como plataformas *Windows*.

Por otro lado, también se trabaja con COMNET III, que es una herramienta comercial de análisis de las prestaciones de redes de comunicaciones. Basándose en la descripción de una red, sus protocolos y su carga, COMNET III simula el comportamiento de la red y proporciona medidas de sus prestaciones, ajustándose perfectamente al entorno educativo. Su interfaz es *amigable* y permite realizar modelos sofisticados muy rápidamente.

En nuestro laboratorio se plantea el estudio mediante simulaciones de varios escenarios de red diferentes, tales como redes de conmutación de circuitos, redes Frame Relay, redes de área local, etc. En el presente curso se ha propuesto un escenario en el que se plantea estudiar el efecto que tiene el uso del protocolo RSVP por parte de los nodos de una subred. Se pretende examinar el efecto que tiene la reserva de recursos sobre los flujos que atraviesan la subred. Para ello, se realiza el estudio de las prestaciones de la red IP antes y después de que los nodos de la red utilicen dicho protocolo, así como el efecto que distintas demandas de QoS para determinados flujos tienen sobre el resto de flujos que son cursados por la red.

4.2. Prácticas sobre Equipos

Prácticas de Configuración

Las prácticas de configuración se plantean como unos ejercicios guiados que permiten al alumno configurar tanto los equipos terminales (sus tarjetas adaptadoras) como los nodos de red en distintos escenarios basados en tecnologías como ATM, RDSI, Frame Relay, etc.

En el presente curso, esta práctica se centra en entornos de red ATM, con el objeto de que el alumno se familiarice con los escenarios de red ATM y las distintas soluciones para su interoperatividad con redes IP.

Sobre la infraestructura ATM que se muestra en la Figura 4, se configuran dos entornos diferenciados, a saber, un entorno de LAN Emulada según las especificaciones del ATM Fórum y un entorno de IP clásico sobre ATM, también denominado CLIP o modo nativo.

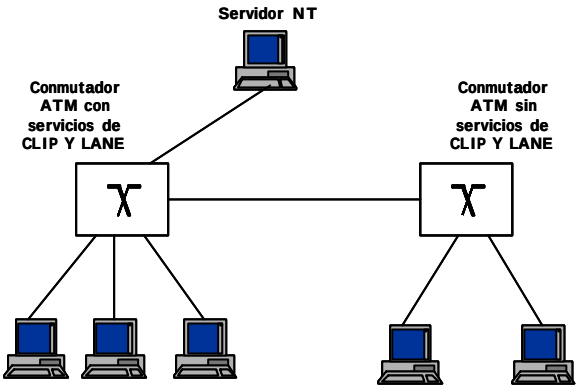


Figura 4: Escenario de configuración ATM

En esta práctica el alumno consigue configurar adecuadamente PCs cliente para conectarse a una red ATM y acceder a distintos servicios a través de ella. Concretamente, el alumno instalará y configurará los *drivers* ATM de la tarjeta ATM, y posteriormente configurará tanto clientes CLIP como clientes LANE, tanto bajo plataforma Windows como GNU-LINUX.

Respecto a la configuración de los conmutadores ATM, los alumnos realizarán dos tareas bien diferenciadas: configuración básica de un conmutador ATM para crear circuitos permanentes (PVC), y configuración de un servidor LINUX con acceso a la red ATM para que proporcione los servicios de soporte a un entorno CLIP y LANE.

Por último, el alumno realiza la medición de las prestaciones de red que se pueden obtener a través de una interfaz ATM, utilizando para ello un programa de libre distribución denominado *netpipe* [5] y otro desarrollado en el DIT denominado *WinNetTest* [13]. Se trata de aplicaciones cliente/servidor que permiten configurar un flujo de paquetes entre dos PCs y ofrecen resultados sobre las prestaciones de transferencia de datos entre dos sistemas.

Prácticas de Interconexión

Las prácticas de interconexión proporcionan al alumno la oportunidad de configurar escenarios completos de comunicaciones en los que entran en juego varias tecnologías de subred y diversos tipos de equipos (PCs, *routers*, conmutadores, servidores, etc.). Esto permite obtener una visión general de los principales protocolos y mecanismos involucrados en un escenario TCP/IP sencillo pero completo, asentando los conocimientos que sobre los mismos ya se han obtenido en otras asignaturas.

A este respecto se dispone de un conjunto de escenarios posibles, de modo que cada uno de los cuales permite realizar una o varias prácticas de interconexión que inciden en aspectos y tecnologías diversos. Se presentan a continuación los escenarios disponibles, junto con una breve descripción de algunas de las posibilidades que cada uno ofrece a la hora de plantear prácticas de las asignaturas que hacen uso del laboratorio.

Escenario genérico de interconexión

El primero de los escenarios se muestra en la Figura 5.

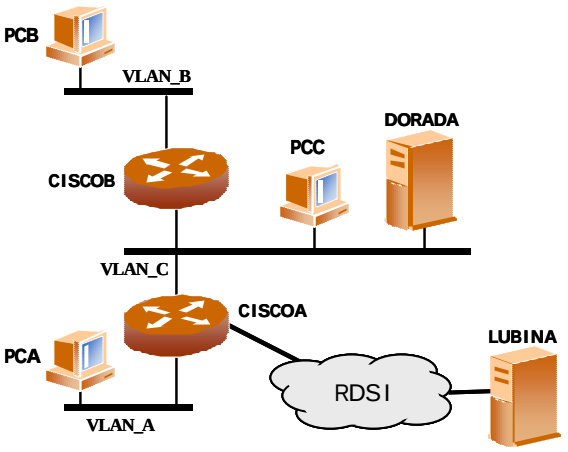


Figura 5: Escenario genérico de interconexión

Como se puede observar, en el mismo se incluyen dos *routers* Cisco modelo 1605, tres segmentos de LAN, una subred RDSI, y varios PCs y servidores. Sobre él se han planteado tres tipos de prácticas: En un primer tipo se aborda la interconexión IP entre los distintos elementos del escenario y cubre los conocimientos básicos de TCP/IP que serán necesarios para abordar prácticas más especializadas. Partiendo de esta práctica, que se puede denominar de interconexión básica, se han desarrollado otras dos prácticas que se centran en aspectos más avanzados o complementarios a los vistos; en concreto, se dispone de una dedicada a la nueva versión del protocolo IP (IPv6) y otra de introducción a las técnicas de filtrado de paquetes.

La práctica de interconexión básica comprende la realización de las configuraciones necesarias en los distintos equipos para lograr la interconexión completa a nivel IP entre ellos. Esto incluye desde la instalación de la tarjeta Ethernet en los PCs y configuración de TCP/IP sobre la misma, hasta la instalación de las tablas de encaminamiento necesarias en los *routers* y la activación del enlace RDSI utilizando PPP.

Además de lograr conectividad entre todos los elementos del escenario, el alumno, mediante varios experimentos guiados, observa el funcionamiento de protocolos como ARP o ICMP y se familiariza con distintos métodos de análisis y diagnóstico, entre los que se encuentran:

- La ejecución de utilidades de diagnóstico (ping, traceroute).

- La activación e interpretación del trazado de tráfico en los *routers*.

- La captura y análisis de tráfico mediante un analizador profesional de tráfico.

En cuanto a la práctica dedicada a la nueva versión del protocolo IP, el objetivo es ofrecer una visión de los fundamentos y principales características de IPv6 frente a las de la versión actual (IPv4). El escenario utilizado es el que se muestra en la Figura 5, salvo por la ausencia del entorno RDSI.

Al igual que en la práctica básica, se configura lo necesario para lograr conectividad total a nivel IPv6, para posteriormente incidir en algunos de los aspectos novedosos o particulares de la nueva versión de IP: autoconfiguración, resolución de direcciones mediante el protocolo ND (*Neighbour Discovery*), cabeceras opcionales y mecanismos de transición de IPv4 a IPv6.

Por último se ha planteado, tal y como se ha comentado, un tercer tipo de práctica sobre el escenario de la Figura 5 (ligeramente modificado), que permite introducir una de las técnicas básicas que forman parte de los sistemas de seguridad de acceso (denominados habitualmente cortafuegos o *firewalls*): las técnicas de filtrado de paquetes. Se trata de una práctica más orientada al diseño que las anteriores (menos guiada), puesto que el alumno parte de unas especificaciones y tiene libertad para llegar a una configuración que las cumpla. Se proporciona una política de seguridad sencilla pero realista, a partir de la cual se deben diseñar e implantar los filtros de paquetes en los interfaces adecuados de los *routers* del escenario para lograr su cumplimiento. Por último el alumno realiza diversas pruebas que le permiten comprobar la eficacia de los filtros diseñados y modificarlos si es necesario.

Escenario de interconexión para encaminamiento

Con el objeto de realizar prácticas sobre encaminamiento en redes IP se diseño el escenario de red que aparece en la Figura 6.

Dicho escenario trata de representar la red corporativa de una empresa de tamaño medio compuesta por:

- Una sede central formada por una red de área local que consta de múltiples segmentos interconectados mediante routers y varios servidores.
- Varias sucursales dotadas cada una de ellas de una LAN con uno o varios ordenadores y un router de acceso a la red troncal.
- Una red troncal que interconecta todas las sucursales entre sí y con la sede central, formada por varios routers enlazados mediante líneas punto a punto.
- Una red de acceso (RDSI) para respaldo de las conexiones

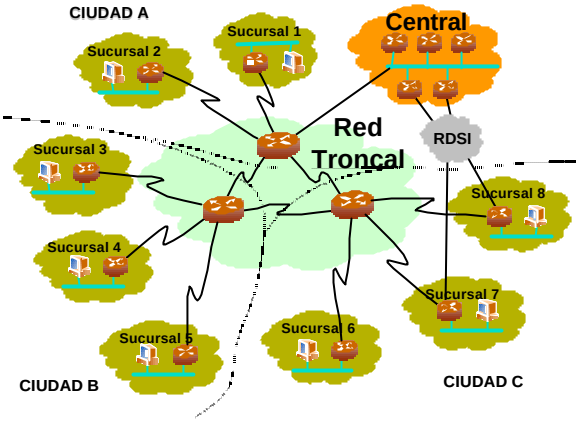


Figura 6: Escenario de interconexión para encaminamiento

Las prácticas que se realizan sobre este escenario inciden en el diseño de los planes de numeración y encaminamiento de una red IP. El trabajo a realizar por cada grupo de alumnos se centra en la puesta en marcha de una de las sucursales de la red, lo que implica, entre otras, las siguientes tareas:

- Configuración básica del router y clientes de la sucursal.
- Configuración de tablas de encaminamiento estático.
- Configuración de tablas de encaminamiento dinámico con protocolos como RIP u OSPF.
- Configuración de enlaces de respaldo (backup) a través de RDSI.
- Utilización de herramientas de diagnóstico (ping, traceroute, analizadores de protocolos, etc) y plataformas de gestión de red sobre el escenario configurado.

El escenario permite, además, experimentar otras técnicas de interés como la configuración de túneles seguros (VPN), traductores de direcciones (NAT) o filtrado de paquetes.

Cabe destacar que, a diferencia de los otros escenarios de interconexión descritos, este escenario implica que los alumnos manejen físicamente los equipos, realizando ellos mismos todas las conexiones necesarias. Además, esta práctica se adapta especialmente a los cursos de postgrado, en los cuales es necesario que múltiples grupos de alumnos realicen la misma práctica en paralelo. En la actualidad, la dotación del laboratorio permite que hasta 16 grupos realicen está práctica simultáneamente.

Finalmente mencionar que para la realización de esta práctica se utilizan routers de la empresa española TELDAT [6], que se adaptan especialmente a este

escenario dado su bajo coste y su especialización en el mercado de dispositivos de interconexión de acceso. Para la red troncal se utilizan 4 routers del modelo NUCLEOX-PLUS dotados de 6 líneas serie cada uno y un interfaz LAN. Para las sucursales se utiliza el modelo CBRA, dotado de un interfaz LAN, una línea serie y un interfaz RDSI.

Escenario de interconexión con Frame Relay

Existe un escenario de interconexión específicamente dedicado a la práctica sobre equipos FR (*Frame Relay*), que se describe en la Figura 7.

Este escenario permite la realización de la práctica a dos grupos de alumnos simultáneamente: cada uno de ellos utilizará un PC conectado a uno de los dos multiplexores de acceso FR, el cual a través del único conmutador FR se conecta a un *router* con un interfaz WAN (que en la práctica se configura para encapsular IP sobre FR). Se utiliza asimismo un segundo PC que se encuentra en la misma subred Ethernet que otro de los interfaces del *router*. En esta práctica los alumnos realizan dos conjuntos de tareas diferenciados:

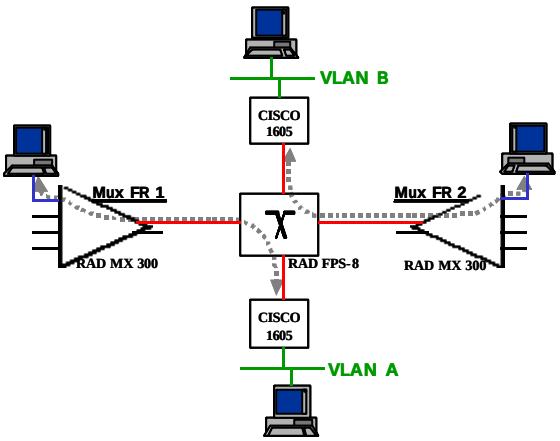


Figura 7: Escenario de interconexión para Frame Relay

- En una primera parte se pretende realizar la configuración de los parámetros FR, tanto en el PC como en el multiplexor asociado y en el *router*, que permitan una correcta encapsulación de IP sobre tramas FR utilizando PVCs (circuitos virtuales permanentes). Una vez lograda la comunicación IP sobre la red FR, se completa la conectividad IP entre los dos PCs involucrados.

- La segunda parte de la práctica está dedicada a la comprobación del efecto que sobre un flujo IP tienen los parámetros de contrato propios de un circuito FR, así como la presencia o ausencia de conformado de tráfico aplicado antes de la inserción del flujo en la red FR. Para ello se realizan diversas pruebas de prestaciones sobre flujos IP entre los dos PCs, utilizando distintos PVCs en la red FR y activando cuando es necesario la funcionalidad de conformado de tráfico en el *router*.

Prácticas de Gestión y Monitorización

Las prácticas de gestión y monitorización se plantean como un complemento práctico de la asignatura teórica "Gestión de Redes", de modo que el alumno puede poner en práctica sus conocimientos sobre gestión de redes basada en SNMP.

Estas prácticas proporcionan al alumno la oportunidad de conocer una plataforma comercial de gestión de redes, de modo que el alumno monitorice interactivamente los nodos de red utilizando dicha herramienta. Para ello se debe configurar la herramienta de gestión tanto para la activación de eventos y alarmas, como para la recogida de datos para su posterior análisis.

La herramienta utilizada por los alumnos en esta práctica es el *HP Openview Network Node Manager* que forma parte de la plataforma *Openview* de HP [4]. Las principales razones por las que se decidió utilizar esta herramienta frente a otras fueron su amplia difusión y sus buenas posibilidades.

5. Conclusiones y Evolución Futura

El laboratorio descrito en este artículo lleva en funcionamiento desde el curso 1998/1999, habiéndose aplicado tanto a prácticas de asistencia abierta adecuadas para asignaturas de grado, como a prácticas de asistencia controlada, más adecuadas para asignaturas de postgrado y cursos de formación.

La estructura del laboratorio permite la realización de prácticas sobre redes que constituyen un complemento necesario a la formación proporcionada en las asignaturas teóricas. Estas prácticas abarcan aspectos de configuración de equipos de usuario, configuración y pruebas de equipos de red, medidas de prestaciones sobre los entornos de red disponibles, y realización de simulaciones de redes de comunicaciones que permite el estudio de casos no fácilmente caracterizables de forma analítica o realizables con la infraestructura disponible. Los escenarios de red implantados en el laboratorio son muy ilustrativos de los entornos que posteriormente los alumnos se encuentran en su vida profesional.

Dadas las características de muchas de las prácticas realizadas en el laboratorio, que involucran configuración de equipos de usuario y de red, y teniendo en cuenta el número de alumnos que realizan las prácticas, tiene especial importancia para el buen funcionamiento del laboratorio el que se disponga de una infraestructura de mantenimiento muy flexible. En este sentido, la facilidad disponible

en todos los puestos de laboratorio de arrancar varios sistemas operativos, en algunos casos partiendo de una imagen del sistema a través de la red y en otros pudiendo regenerar la imagen del sistema presente en el disco duro, también a través de la red, permite que en caso de errores en la configuración, el alumno pueda, sin ayuda de monitores de laboratorio o del profesorado, volver a una situación estable.

En resumen, la instalación y mantenimiento de un laboratorio de estas características conlleva una carga de trabajo alta debido al gran número y heterogeneidad de los equipos disponibles en los distintos escenarios. Es necesario invertir un gran esfuerzo en mejorar los procedimientos de operación y mantenimiento, de manera que se simplifiquen y automaticen lo más posible. En este sentido, la utilización de herramientas que permitan emular escenarios complejos dentro de una misma máquina puede resultar de especial utilidad (por ejemplo, el entorno VMware[14] permite tener en una misma máquina física varias máquinas virtuales que pueden comunicarse entre sí a través de interfaces de red físicos o virtuales).

Cabe destacar también que la opinión de los alumnos hacia el laboratorio ha sido bastante positiva, como muestran las encuestas realizadas por el Gabinete de Estudios y Documentación de la ETSITM dentro de las actividades de evaluación de las distintas asignaturas por parte del alumnado, así como por una encuesta realizada por los responsables de la impartición del laboratorio. En estas encuestas la valoración general de la asignatura es muy positiva dentro del conjunto de los estudios de grado. Asimismo, los alumnos valoran especialmente la posibilidad que les ofrece el laboratorio de realizar prácticas sobre equipos reales, enfrentándose a problemas que no se aprecian suficientemente desde la perspectiva mayormente teórica de otras asignaturas.

Como líneas futuras de evolución del laboratorio, se plantea la instalación de nuevos entornos de red, concretamente en el plazo más inmediato la disponibilidad de un entorno de red local inalámbrica (WLAN). Asimismo, se plantea el introducir prácticas de instalación, configuración y pruebas de servicios telemáticos, tanto de soporte a red (DNS, DHCP, WINS), como de usuario (correo, WWW, etc.).

Agradecimientos

La creación del laboratorio de comunicaciones descrito en este artículo ha exigido la colaboración y el esfuerzo de múltiples personas y organizaciones. Queremos mostrar nuestro agradecimiento a todo el personal del DIT y de la ETSITM que nos ha dado su apoyo, en especial al personal del Centro de Cálculo del DIT, sin cuya actividad este laboratorio no sería posible. Asimismo queremos agradecer la colaboración de las empresas HP, Agilent, CISCO y, en especial, a TELDAT por su estrecha colaboración para la creación de uno de los escenarios de interconexión de redes.

Referencias

[1] Servidor de Web del Laboratorio. Disponible en: http://www.dit.upm.es/labrst

[2] Programa de Postgrado en Sistemas de Redes y Comunicaciones. Disponible en: http://www.master.etsit.upm.es/

[3] Página web del simulador NS. Disponible en: http://www.isi.edu/nsnam/ns/

[4] Página web de Openview. Disponible en: http://openview.hp.com

[5] Página web de Netpipe. Disponible en: http://www.scl.ameslab.gov/netpipe/

[6] Página web de Teldat. Disponible en: http://www.teldat.es

[7] Página web de Cisco. Disponible en: http://www.cisco.com

[8] Página web de RAD. Disponible en: http://www.rad.com/

[9] Página web de FORE. Disponible en: http://www.fore.com/

[10] Página web de 3COM. Disponible en: http://www.3com.com/

[11] Página web de Agilent. Disponible en: http://www.educatorscorner.com/products/promo

[12] Página web de Ethereal. Disponible en: http://www.ethereal.com

[13] Aplicación de medida de prestaciones WinNetTest. Disponible en: http://jungla.dit.upm.es/~bti/files/winnettest.zip

[14] Página web del entorno VMware. Disponible en: http://www.vmware.com

Sistema Automático de Administración Centralizada

Tomás P. de Miguel
Judith Álvarez
Omar Walid
Eduardo Gómez Melguizo
Telemática, E.T.S.I. Telecomunicación, Univ. Politécnica de Madrid
Antonio Beamud
Agora Systems S.A.
Teléfono: 91 3367366 Fax: 913367333
E-mail: tmiguel@dit.upm.es

Abstract. *In recent years networked workstations can provide users with access to a broad set of distributed resources. Unfortunately, the management overhead of maintaining workstations can become overwhelming, especially with a large installed base. Traditional manual management procedures do not provide system administrator with a set of efficient and flexible management tools that can take advantage of a networked environment. The paper describes the Centralized Automatic Administration System (ADC) developed to provide a tools set to store and rebuild in few minutes the complete system configuration. It is currently available for some Linux distributions and windows support is now under development.*

1 Introducción

El número de estaciones de uso personal y servidores en cualquier organización ha crecido en los últimos años a ritmo constante. Los sistemas modernos son muy potentes y capaces de actuar en red ejecutando gran número de aplicaciones sofisticadas. Sin embargo, no existe una solución clara al problema de la instalación y configuración de tantos equipos. Las redes se han complicado y los métodos manuales, que eran válidos hace unos años, ahora no los son. La administración manual tradicional también se ha complicado, provocando una demanda enorme de administradores expertos con conocimientos en áreas muy diversas: gestión, informática, comunicaciones, etc.

Para las pequeñas instalaciones (Small Business Netowork Environment) se siguen utilizando técnicas manuales de administración. Son instalaciones con pocas estaciones de sobremesa de uso personal y uno o muy pocos servidores. El sistema más popular es el Microsoft Enterprise Management [1] basado en NT.

Pero hay otro tipo de entornos con cientos o miles de máquinas, muy similares entre sí, que desempañan funciones similares en muchos casos. En estos entornos se aplican técnicas de Large Installation System Administration [2]. Ejemplos son las redes de las grandes compañías, las granjas de servidores, pero también grandes laboratorios docentes y de investigación.

Para abordar este problema han surgido gran número de herramientas que permiten automatizar los procesos de instalación y actualización. Desde la más simple copia de ficheros como *rdist* o *package* [3] (un sistema de actualización de utilidades de AFS) a sofisticados productos comerciales como *Tívoli* [4], pasando por un gran número de esfuerzos de desarrollo individuales como *Config* [5], *Cfengine* [6], *Alice* [7] o *GeNUAdmin* [8].

2 Modelos de Administración

Los sistemas desarrollados actualmente tratan de resolver el problema de la administración abordando dos aspectos fundamentales:

- La automatización
- La configuración centralizada

La automatización permite construir sistemas muy sencillos de usar, pero en general poco flexibles y normalmente adaptados a un sistema operativo particular. Otras tratan de resolver el problema de forma general para grandes instalaciones (con miles de máquinas), pero son sistemas muy complejos y costosos.

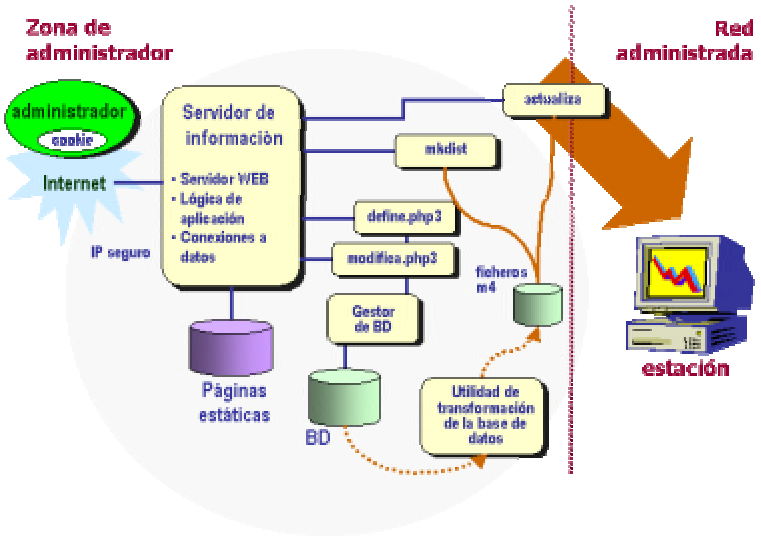


Figura 1. Arquitectura del Servidor Central de Supervisión

Cuando se gestionan configuraciones (tanto hardware como software) de estaciones en una gran red, es frecuente controlar la instalación desde un servidor (que puede ser la estación de sobremesa del administrador. En unos casos se hace utilizando herramientas estándar de registro de la actividad del sistema como *syslog*, en otros herramientas de gestión de red como las basadas en SNMP o directamente a través del sistema de ficheros.

Unos modelos de administración se basan en la idea de control que consiste en vigilar de la mejor manera posible la instalación. Ejemplo de ello son sistemas sofisticados como *Tivoli* [4], *HP OpenView* [9] o *SUN Solstice* [10]. En estos entornos, el administrador define las condiciones de error, estableciendo informes para cada tipo de error. La desventaja es que el administrador tiene que realizar los procedimientos de administración a mano, aunque es posible automatizar algunos procesos de gestión sencillos. Además se precisa de administradores muy expertos para conocer la herramienta y poder analizar y reparar los errores que surgen. Esto hace que la cantidad de trabajo requerida no se reduzca sustancialmente cuando aumenta el número de equipos de la red.

Otra tendencia que está teniendo bastante aceptación últimamente consiste en equipar cada equipo de la red con un sistema inmunológico simple. Al igual que en cuerpo humano, estos sistemas pretenden desplegar utilidades que se activan periódicamente haciendo limpiezas, preparando informes sobre problemas serios y que incluso pueden reaccionar ante problemas en la instalación. Por ejemplo, si un sistema falla, una utilidad lo detecta y pone en marcha inmediatamente un sistema de repuesto informando a todos los clientes que se ha cambiado la configuración. La herramienta *Cfengine* [11] está experimentando con este tipo de tecnología. El inconveniente de esta solución es que es parcial, no es fácil de aplicar en un entorno general y es poco efectiva con estaciones de sobremesa en las que los usuarios que pueden cambiar o incluso desactivar estos servicios.

Finalmente, otro planteamiento consiste en reiniciar en caso de problemas. En grandes instalaciones muchas estaciones pueden ser utilizadas por varios usuarios con conocimientos muy variados. Estos sistemas tienden a fallar con alarmante regularidad. Cuando se produce un error es difícil que pueda disponerse en ese momento de un administrador especializado que pueda resolverlo. La solución más sencilla es reiniciar. Esto se puede definir de varias formas. Si disponemos de la configuración de la estación y se detecta un problema podemos actualizar de nuevo la configuración. Si el problema es mas serio, se puede reinstalar completamente la estación.

En otros casos, como un laboratorio o un aula informática, cuando un usuario accede a un puesto espera encontrar siempre la estación correctamente instalada. Entonces, lo mejor es hacer una instalación completa cada vez que se arranca la estación. Se inician las particiones, se carga la imagen a través de la red y la máquina queda lista para su uso. Esto es posible siempre que los datos del usuario estén en un servidor y no en la estación y que el tiempo de instalación sea menos de un minuto.

Cuando hablamos de un servidor, el proceso de reinicio no es tan sencillo. Hay que salvaguardar los datos y la instalación completa es lenta y muchas veces difícil de repetir, porque el proceso se ha

realizado manualmente, está poco o mal documentado y los cambios de configuración realizados con posterioridad no se suelen documentar.

Una solución es generar una imagen del sistema inmediatamente después de la instalación. Esto garantiza que el tiempo necesario para efectuar el reinicio es pequeño. Si agrupamos los datos de usuario en una partición diferente de la del sistema (cosa muy frecuente), se puede rehacer la instalación sin tener que guardar y recuperar todos los datos de usuario.

Sin embargo, tiene dos inconvenientes. En primer lugar la imagen puede ocupar mucho espacio, en muchos casos mas que un CD-ROM. Si hay que administrar muchos servidores el proceso de reinicio es incomodo y difícil de automatizar, pues hay que buscar la cinta adecuada para poder vaciarla a través de la red y en muchos casos descomprimirla previamente.

Con todo, el principal inconveniente es que una imagen de la instalación casi nunca es completa. Es muy frecuente realizar pequeños cambios en la configuración después de la instalación. Para poder reiniciar una imagen es necesario registrar todos los cambios por otro procedimiento y poder repetir el proceso después del reinicio. El proceso es incómodo y la actualización es manual.

Si en vez de guardar la imagen completa se guardan solo las instrucciones de instalación y de actualización, el volumen de información que es necesario mantener es mucho menor y se puede almacenar en cualquier servidor sin problemas.

El objetivo de un sistema de administración automática y centralizada es poder reiniciar rápidamente gran número de estaciones y servidores, manteniendo la información mínima de la instalación y de cualquier actualización posterior.

3 La administración automática centralizada

La Administración Automática Centralizada (AC) se basa en la existencia de un Servidor Central de Supervisión (SCS), que consiste de un servidor WEB con acceso a una base de datos, donde se almacenan las configuración de las estaciones gestionadas y un conjunto de utilidades que realizan las tareas de administración. El sistema se utiliza para administrar varias distribuciones de Linux y se está preparando para Windows.

El sistema funciona conectándose con el SCS para configurar la nueva estación, utilizando una interfaz WEB. Si es una estación similar a otra ya existente, el proceso consiste en rellenar datos específicos como el nombre, la dirección IP o la localización.

Si la estación necesita una configuración especial para añadir nuevos manejadores, o instalar otras aplicaciones hay que realizar pruebas a mano, antes de establecer la configuración automática definitiva.

Si todas las estaciones de una red y todas sus configuraciones fueran diferentes, la ventaja de este sistema sería reducida. El sistema permite almacenar la configuración completa de una estación en unas pocas variables. Afortunadamente, incluso en una red de pequeño tamaño, las configuraciones de estaciones y servidores son todas muy similares y las ventajas del método se ven muy rápidamente.

En nuestro sistema se almacena solo un conjunto de variables para identificar unívocamente una estación.

Una vez definida la configuración de la estación es posible instalarla y actualizarla. El proceso de instalación se inicia siempre en la estación. Unas veces al encenderla y otras porque alguien invoca el procedimiento de instalación explícitamente.

Para que una estación se pueda instalar tiene que tener acceso al servidor SCS. Hay ocasiones en que la conexión entre estación y servidor es de baja capacidad. En otras ocasiones el servidor está protegido en una zona segura y la estación no tiene acceso directo a él.

En estos casos se puede configurar la red de administración con un Servidor Intermediario (SI). El SI funciona igual que el servidor central (SCS), administrando las estaciones instalando y actualizando, pero recibiendo las órdenes del SCS y no directamente del administrador.

El servidor central (SCS) tiene la base de datos de configuraciones. Cuando se define una estación en una red que se administra desde un SI, se actualiza este con la configuración de la nueva estación. De esta manera, cuando se pida la instalación, el SI estará preparado con la nueva configuración.

Las estaciones administradas con este sistema no necesitan atención personalizada del administrador. Si se detecta un fallo se actualiza la configuración desde el servidor. Esto es especialmente interesante en laboratorios de investigación donde hay usuarios con privilegios y se sospecha que han cambiado indebidamente la configuración.

Si se detecta un error grave se realiza un reinicio completo de la estación. Operación que dura unos minutos (dependiendo del tipo de configuración y las características de la red) y deja el sistema recién

instalado con la última configuración válida definida.

En estas condiciones un fallo hardware produce el mismo efecto. La pieza rota se sustituye y luego se reinicia de nuevo la estación.

El sistema de administración se completa con un sistema de monitorización también basado en WEB, que permite enviar al navegador del administrador los fallos de operación detectados en tiempo real.

4 Modelo de administración centralizada

El sistema AC consiste en un conjunto de utilidades que utilizan los datos de configuración almacenados en una base de datos. El sistema se puede manipular directamente desde línea de comandos o a través de un navegador WEB. Para operarlo hay que estar registrado como administrador de la red en la base de datos de administradores. Hay dos tipos de administradores:

- El administrador básico tiene capacidad para definir e instalar estaciones.

- El administrador experto tiene además capacidad para definir modelos de estaciones. El sistema arranca con un administrador experto que es propietario de la base de datos y del servidor Web y puede registrar a todos los demás.

Los administradores configuran estaciones definiendo patrones y asignando el modelo adecuado a cada estación.

4.1 Modelos de estación

Todas las estaciones y servidores de una red son diferentes, pero muchas comparten aspectos en común: tienen instalada la misma versión de sistema operativo, disponen del mismo hardware incluso cumplen la misma función, como los puestos de trabajo de un laboratorio. La configuración de una estación es una instancia de un modelo.

Un modelo es una lista ordenada de patrones. Un patrón es un conjunto de archivos y utilidades parametrizados con variables del sistema de administración, que permiten configurar un servicio o una aplicación.

Por ejemplo, si la estación dispone de una tarjeta controladora de RDSI, se definirá un patrón para configurar el manejador, instalar aplicaciones de monitorización y los archivos de configuración. En el patrón se definen variables para que se pueda cambiar la configuración de unas estaciones a otras.

Las variables de configuración de un patrón se introducen como propiedades en la base de datos de configuraciones. Las estaciones que utilicen un modelo en el que se incluya ese patrón deben dar un valor a esas variables. Algunos patrones son muy comunes y se ofrecen directamente con el sistema. El resto tienen que ser definidos por un administrador experto. En consecuencia, la base de datos no dispone de un conjunto de propiedades fijo definido de antemano.

Aunque en un primer momento se requiera cierto esfuerzo para definir los patrones, en general el número de patrones no es muy grande y en cuanto el número de estaciones empieza a crecer (diez o mas), el tiempo de instalación de nuevas estaciones se reduce notablemente.

En la figura 2 se puede ver la estructura del sistema de administración para una distribución Linux RedHat, en la que se pueden ver dos patrones. El primero, definido como BASICO, controla la lista de estaciones conocidas (`/etc/hosts`) y la configuración de los controladores (`/etc/modules.conf`). El segundo (RADIOTV) sirve para configurar una estación receptora de radio y actualiza la configuración de controladores y configura la estación receptora (`/var/rtv.conf`).

Un modelo de máquina incluye solo el patrón BASICO y otro puede ser la unión de BASICO y RADIOTV, como es el que corresponde en la figura a la estación `tokio`.

Como se puede observar en la figura, los patrones reflejan la estructura de ficheros de la estación final. La raiz (/) es un directorio con el nombre del patrón y los ficheros están en directorios con el mismo nombre y en la misma posición que deben ocupar en el sistema final.

La lista de patrones de una estación está ordenada. Cada patrón actúa como una transparencia que se superpone sobre el sistema ya existente. En caso de coincidencia, el fichero del último patrón prevalece sobre los anteriores. Así en el ejemplo, la versión del fichero `/etc/modules.conf` del patrón BASICO no se incluye porque prevalece la de RADIOTV.

4.2 La transformación de patrones

Como hemos mencionado anteriormente todas las estaciones de una red pueden clasificarse según su configuración. La estructura de patrones permite catalogar en modelos las configuraciones de forma extensible. Pero cada estación es diferente a las demás porque tiene un hardware diferente, diferentes utilidades o al menos porque su nombre es distinto.

En consecuencia, los patrones son modelos parametrizados de un componente o una utilidad. La configuración que finalmente se instala en la estación es el resultado de acumular todos los patrones del modelo elegido y sustituir los valores de las variables que aparecen en los ficheros de configuración.

Por ejemplo, el archivo `/etc/resolv.conf` permite configurar la forma en que se va a utilizar el servicio de nombres. Normalmente, el programa instalador deja configurado el fichero con el dominio al que pertenece la estación.

Pero si quisiéramos incluir algún otro dominio tendríamos que hacerlo a mano. La configuración automática consiste en definir un patrón que incluya ese fichero.

Incluyendo los nombres de dominio en el patrón se sustituirá el fichero al hacer la actualización y todas las estaciones de la red podrán incluir la lista de dominios deseada.

Pero no siempre se desea esto. En general, se desea que no todas las estaciones de una red incluyan la misma lista de dominios de búsqueda. La solución es definir una variable asociada al fichero del patrón.

Cuando se construye la configuración de cada estación se sustituyen las variables de cada fichero del patrón por los valores definitivos para esa estación.

```
nameserver AC_NAMESERVER
search AC_DNS_SEARCH
```

Para realizar esta operación se han desarrollado unos programas que están basados en la utilidad `m4` disponible en todas las distribuciones UNIX. Se invoca con dos parámetros: el archivo que contiene la definición de las variables y el archivo a transformar. El resultado es el archivo particularizado para una estación, con los valores de las variables sustituidos. En nuestro ejemplo, si queremos configurar la estación *tokio* elegiríamos el archivo de variables de tokio:

```
define(`AC_NAMESERVER',`138.4.2.10')
define(`AC_DNS_SEARCH',
       `dit.upm.es saba.rediris.es')
```

que al procesar el archivo `resolv.conf` produce el resultado esperado para la estación tokio:

```
nameserver 138.4.2.10
search dit.upm.es saba.rediris.es
```

4.3. Organización de la base de datos

Gestionar la base de datos de estaciones utilizando exclusivamente ficheros no es lo mas adecuado. Para cada operación de administración hay que entrar en el servidor y editar con cuidado los ficheros de configuración. Además, podemos cometer errores al escribir las propiedades de un modelo, o escribir modelos con propiedades que no han sido definidas antes en ningún otro modelo de ese tipo, o incluir tipos que no existen en una máquina.

En resumen, no es fácil comprobar la coherencia de los datos de configuración de todas las estaciones de una red. Para evitar estos problemas y ofrecer una interfaz más segura y sencilla, almacenando la información en un sistema de gestión de bases de datos con acceso desde Web.

La base de datos se basa en tres conceptos: máquina, componente y propiedad. Las máquinas se definen como la selección de un componente de cada uno de los tipos existentes. Por ejemplo, una estación que tenga una tarjeta gráfica ati selecciona este componente como valor del tipo de tarjeta gráfica.

Todos los componentes de un mismo tipo comparten las mismas propiedades, aunque cada componente tenga un valor distinto para esas propiedades. Por ejemplo, para todas las subredes de una organización se debe especificar una propiedad para definir la máscara de la red, otra para el router por defecto y otra para indicar cual es el servidor de DNS.

Cada propiedad tiene el mismo nombre que la variable que pretende sustituir en los patrones. Si hay una variable AC_NOMBRE en uno o mas ficheros de configuración, debe existir una propiedad definida en la base de datos con el mismo nombre.

A diferencia de otros sistemas de administración, este no establece ninguna propiedad por defecto. El administrador va definiendo la lista de propiedades a medida que define nuevos patrones según sus necesidades. Por esta razón, el esfuerzo de despliegue de la red es mucho mayor al principio que después cuando la red ha crecido suficientemente. En los sistemas de administración manual la situación es la opuesta, cuanto mayor es e número de estaciones, el trabajo se incrementa mucho más.

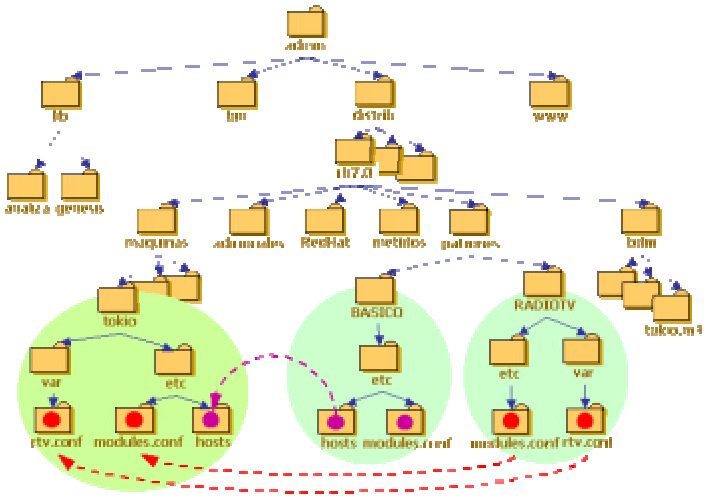


Figura 2. Estructura de ficheros del servidor central

4.4. Procedimiento de instalación

Todas las distribuciones UNIX son ligeramente diferentes. Cada una instala un conjunto diferente de paquetes, distribuye los ficheros de forma ligeramente diferente e incluso configura las utilidades de forma diferente. Por eso, cada distribución suele ofrecer un instalador que distribuye todo lo necesario de forma ordenada. Pero, al mismo tiempo, todos los sistemas necesitan ciertos datos para poder configurarse. En general son los mismos datos con independencia de la distribución elegida: el nombre de la estación, la dirección de red, el modelo de cada componente hardware, etc.

Por ejemplo, RedHat permite automatizar el proceso de instalación (sistema kickstart) generando un disco de instalación apartir de la configuración de la estación. AC construye los discos de instalación a partir de los datos de configuración de la estación almacenados en la base de datos.

No hay un único tipo de imagen de instalación. La estación puede estar aislada o conectada a la red mediante un enlace de poca capacidad en unos casos o enorme en otros. Tampoco el dispositivo de arranque del instalador es siempre el mismo:

Tipo de arranque	**Distribución**
Red (DHCP)	Servidor con NFS
Floppy	Servidor con NFS
CD-ROM	El mismo CD-ROM

La instalación desde CD-ROM solo es recomendable cuando la estación a instalar está aislada del servidor. La generación del CD-ROM lleva tiempo y puede que sea necesario mas de un CD para poder disponer de toda la información para la instalación. Otra posibilidad es utilizar el floppy para cargar el instalador y acceder a la red (vía NFS Network File System) para acceder a toda la información necesaria para la instalación.

Pero el sistema ideal es el basado en el arranque de red, si se dispone de esa facilidad en la tarjeta de red (sistema Preboot eXecution Environment PXE), que empieza a ser habitual en las BIOS modernas. La idea es conseguir que cualquier ordenador con una tarjeta de red instalada sea capaz de obtener la imagen de arranque de sistema operativo, partiendo únicamente de su dirección MAC (Medium Access Control Address) haciendo uso de un servidor DHCP (Dynamic Host Configuration Protocol), un servidor TFTP (Trivial File Transfer Protocol) y un servidor NFS.

El proceso de arranque por DHCP de una instalación automática de RedHat (`kickstart`) es el siguiente:

1) El ordenador da control a la BIOS que se encarga de detectar y hacer una primera configuración de los dispositivos del sistema.
2) La BIOS pasa control a la parte PXE (que bien puede estar dentro de la propia BIOS o en la ROM de la tarjeta de red).
3) PXE inicializa la tarjeta de red y se prepara para enviar una petición de DHCP a la dirección IP de broadcast de la red (`255.255.255.255`), indicando su dirección MAC y opcionalmente alguna de sus características.
4) El servidor DHCP de la red escucha la petición, identifica su dirección MAC y le envía de vuelta a la estación la configuración para poder

arrancar. La configuración incluye una dirección IP con la que poder operar y el nombre de la imagen de arranque que se utilizará para realizar la carga de la imagen con el instalador.

5) Después carga la imagen de arranque utilizando el protocolo TFTP (Trivial FTP). La imagen cargada se genera utilizando el sistema GRUB [12], que entre otras cosas permite dar control de arranque a particiones y sistemas operativos ya instalados en el disco duro local.

6) Cuando termina la carga se pasa control al programa de arranque. La ejecución de este programa provoca la descarga de otra imagen, también utilizando TFTP. Esta imagen contiene el programa de instalación (imagen: `initrd.img` y núcleo de sistema operativo: `vmlinuz`) y las opciones de configuración automática de la estación.

7) Una vez que los ficheros están cargados en memoria, se les pasa control y el ordenador arranca como si los hubiera obtenido de alguna de las particiones locales, pasándoles además las opciones de arranque obtenidas previamente.

8) Si la opción de arranque que se le pasa al núcleo (`vmlinuz`) es la de instalación automática de RedHat, se puede arrancar el instalador automático, buscando la configuración en la red (`ks=nfs:`IP_DEL_SERVIDOR`:/kickstart/`)

9) Una vez obtenida la configuración, el proceso de arranque recae en su totalidad en el instalador automático, que se encargará de particionar, formatear, instalar y configurar el sistema operativo en el disco local a través del servidor de red.

Este proceso que es muy sofisticado es sin embargo muy rápido y se puede instalar un servidor completo en menos de 10 minutos (dependiendo de las condiciones de la red y la velocidad de los dispositivos del servidor).

5. Organización de un laboratorio

Los grupos de investigación en sistemas multimedia distribuidos, protocolos y arquitecturas de comunicaciones utilizan laboratorios de investigación compartidos. Estos laboratorios requieren entornos distribuidos con gran número de estaciones con software muy variado: en unos casos son estaciones de desarrollo, mientras que en otros deben configurar aplicaciones con bibliotecas de comunicaciones especiales. En conjunto las maquetas necesarias para probar los desarrollos realizados necesitan numerosos recursos: equipos de comunicaciones, estaciones, espacio físico, etc.

Una forma de reducir costes consiste en aplicar métodos de compartición de recursos. Esto supone que varios grupos de investigación van a utilizar las mismas estaciones y se van a distribuir el tiempo de uso. El problema es que el tipo de instalación que utiliza cada grupo de trabajo es diferente de la de los demás. Además, cuando se prueba una aplicación distribuida es necesario instalar lo mismo en todas las estaciones de prueba.

La solución adoptada consiste en configurar cada estación del laboratorio con dos instalaciones diferentes y utilizar el Sistema de Administración Centralizado para asegurar la coherencia de todas ellas.

En cada estación se hacen dos instalaciones (ambas Linux o Linux y Windows). Cuando arranca la estación se elige una de las dos instalaciones. Si arranca la primera, la estación está en la red de pruebas principal. Si se selecciona la segunda, la estación se configura en una red diferente. Todas las estaciones se conectan a unos conmutadores ethernet que configuran las estaciones en redes virtuales diferentes de forma automática. Conectando el resto de los equipos de comunicaciones a los mismos conmutadores ethernet se puede configurar cualquier escenario de pruebas con las estaciones del laboratorio, sin necesidad de tocar el cableado.

Para gestionar el laboratorio se ha utilizado un servidor (SCS) que está dentro de la zona segura del departamento. Las estaciones se instalan desde un servidor intermediario (SI) que hay en la zona de pruebas, recibiendo las órdenes del servidor principal. Si se desea utilizar un único servidor se debe permitir las conexiones NFS de las estaciones de la zona de pruebas con el servidor de administración (SCS).

Cuando se introduce una nueva estación en el laboratorio de investigación, se analiza el hardware que tiene, se introduce la configuración en la base de datos o simplemente los datos de la máquina, si esta dispone de hardware conocido.

El sistema de Administración Centralizado se ha empezado a utilizar dentro del departamento para administrar los laboratorios de investigación y pronto se va a empezar a utilizar para gestionar los laboratorios docentes y la configuración de los servidores de la red troncal.

También se ha utilizado para configurar estaciones de uso personal. Esto permite repetir la instalación en caso de avería y la base de datos sirve de inventario. Esta es la mejor forma de conocer las características de cada estación de la red y permite analizar mejor planes de renovación.

Conclusiones

El Sistema de Administración Centralizado ha sido diseñado para facilitar la instalación y actualización de equipos informáticos en red de una forma

sencilla y rápida. Las características más relevantes son:

- Disponer de una organización centralizada con un servidor asignado, aunque no se utilice exclusivamente para administración.
- Mejorar sensiblemente la instalación y actualización de estaciones idénticas o muy similares.
- Facilitar a administradores sin experiencia, o incluso a los propios usuarios, la oportunidad de instalar y actualizar sus máquinas.
- Permitir identificar cualquier propiedad de una máquina y gestionar la configuración de diferentes distribuciones. El sistema no establece a priori ninguna definición.

Antes era necesario esperar varios días para poder disponer de una configuración determinada en un conjunto no muy elevado de estaciones y siempre había variaciones entre unas configuraciones y otras. Además, la mayor parte de las operaciones de configuración no estaban documentadas y solo podían ser realizadas por la misma persona. En consecuencia actualizaciones se retrasaban todo lo posible y las instalaciones nunca respondían a las necesidades reales de los usuarios. La mejora de productividad con este sistema ha sido enorme. El sistema se ha mostrado muy flexible en diferentes ámbitos de aplicación y se ha empezado a utilizar de forma regular en algunos laboratorios de investigación.

Se puede utilizar directamente desde línea de comandos, pero es mas sencillo utilizar la interfaz WEB. Se puede usar instalado las estaciones en red o a través de un CD-ROM si la estación estuviera aislada.

La definición de una nueva configuración con sus patrones correspondientes exige la intervención de un administrador experto con experiencia en la instalación de ese tipo de sistemas o aplicaciones. Sin embargo, la definición de una máquina y su instalación pueden ser realizadas por personal no especializado.

El sistema de gestión ha sido desarrollado integrando varias tecnologías: Shell, Perl, PHP y WEB, sobre sistema operativo Linux RedHat 7.0.

En la actualidad se está desarrollando la aplicación a estaciones Windows, aprovechando las nuevas facilidades de la Administración Zero [1] que ofrece Microsoft.

Bibliografía

[1] Zacker, C., "*Zero Administration for Windows*", O'Reilly & Associates, Inc., 1st Edition May 1999.

[2] Finke, J., "*Automation of Site Configuration Management*", USENIX LISA XI Proceedings, 1997.

[3] "AFS Administration Guide", IBM Press, release 3.6, April 2000.

[4] Tivoli Systems, "*Introducing NetView Automation*", (http://www.tivoli.com/products/index/netview_390/library/content/dqal2m05.htm) , 2000.

[5] Rouillard, J. P. and Martin, R. B., "*Config: A mechanism for installing and Tracking System Configurations*", USENIX LISA VIII Proceedings 1994.

[6] Burgess, M, "*cfengine automated administration system*", Edition 5.1 for version 1.5.4. Faculty of Engineering, Oslo College, Norway, April 2000 (http://www.sunsite.ualberta.ca/Documentation/Gnu/cfengine)

[7] Herschel, F., Hollants, P., Frank, A., "*Alice: Automatic Linux Installation and Configuration Environment*", SuSE Linux Solutions AG, 2000.

[8] Harlander, Dr. M., "*Central System Administration in a Heterogeneous Unix Environment: GeNUAdmin*", USENIX LISA VIII Proceedings, 1994.

[9] *"Introduction: HP OpenView Network Node Manager Special Edition 1.4 With Dell OpenManage™ HIP 3.4/3.4.1 User's Guide"* http://support.dell.com/docs/SOFTWARE/HIP141/intro.htm

[10] *"Solstice Network Client Documentation"* http://www.ssd.sscc.ru/misc/snc_docs/products/pcpro3/docs.htm

[11] Burgess, M., "*Evaluating cfengine´s immunity model of site maintenance*", Oslo College, 1998.

[12] *"GRUB: The GNU GRand Unified Bootloader"*, (http://www.gnu.org/software/grub/).

CDMAC, Un protocolo de acceso al medio para sistemas CDMA multimedia.

Loren Carrasco, Guillem Femenias
Universitat de les Illes Balears, Departement de Ciencies Matemàtiques i Informàtica.
Ctra Valldemossa km 7.5, 07071 Palma SPAIN
Teléfono: 971 172996 Fax: 971 173003
E-mail: loren@ipc4.uib.es, guillem@ipc4.uib.es.

Abstract

This paper describes the CDMAC, a new Medium Access Control (MAC) protocol for multimedia traf£c in CDMA wireless networks. The protocol intends to extract the maximum capacity and ¤exibility out of the CDMA scheme and at the same time guarantee the expected QoS of different type of services. CDMAC is able to maintain QoS requirements thanks to the shaping, policing and traf£c differentiation performed by the scheduler. Moreover, an iterative algorithm, applied at the beginning of each frame, is used to £nd the optimal vector of powers for all mobiles present in the system that maximize the system capacity. The basic constraint of the capacity maximization process is that the BER QoS of each connection should be ful£lled. Finally a distributed implementation feasible in a practical scenario is presented.

1 Introducción

En estos últimos años los nuevos sistemas móviles han empezado a tomar forma. Uno de los principales resultados de este proceso de de£nición ha sido la selección de la tecnología CDMA como la base para las nuevas interfaces radio. El interés en este esquema concreto de división de recursos se debe fundamentalmente a su ¤exibilidad para multiplexar servicios de voz y datos. Esta característica es muy importante en un momento en el que los móviles y Internet son las dos grandes fuerzas conductoras en el mercado de las telecomunicaciones y se espera una demanda generalizada de servicios móviles de datos en un futuro nada lejano.

Con el objetivo de soportar tipos de trá£co muy diferentes, los nuevos interfaces radio tienen en común una gran ¤exibilidad y capacidad. Formatos de transmisión diferentes (ganancias de procesado, transmisiones multicódigo, esquemas de codi£cación de canal etc.) estarán disponibles para cada tipo de trá£co permitiendo la operación del sistema en multitud de entornos y la óptima distribución de los recursos entre los diferentes servicios. Sin embargo, si se desea aprovechar al máximo las prestaciones de estos nuevos interfaces es necesaria una estrategia adecuada de control de acceso al medio.

Las primeras propuestas de protocolos MAC [**?**] consistían en protocolos distribuidos de control de acceso aleatorio como ALOHA, ALOHA ranurado, CSMA o ISMA. Con dos limitaciones básicas: su incapacidad de garantizar requerimientos de QoS de los diferentes tipos de trá£co y sus problemas de estabilidad en condiciones de carga elevada. Para resolver estos problemas, la mayor parte de las propuestas de protocolos MAC para W-CDMA incluyen algún tipo de control por parte de la estación de base y establecen reglas de diferenciación para diferentes tipos de trá£co.

A continuación, se describe el CDMAC, un nuevo protocolo de asignación bajo demanda . Este protocolo esta íntimamente relacionado con el esquema de control de potencia y mantiene el nivel de interferencia en unos valores aceptables. El análisis que se incluye a continuación corresponde al enlace ascendente. En la sección II se introduce una breve descripción de la capa física, las secciones III y IV van introduciendo gradualmente la operación del CDMAC y en la sección V las prestaciones del protocolo se evalúan a través de simulaciones.

2 La capa física W-CDMA

Las características de la capa física incluidas aquí se corresponden con las propuestas de interfaces radio para los nuevos sistemas de 3ª generación [**?**, **?**]. La transmisión es orientada a paquetes para permitir una multiplexación óptima de los diferentes tipos de trá£co. Transmisiones multicódigo y con ganancia de procesado variable son factibles gracias a la utilización de dos capas de códigos de ensanchamiento y un esquema ranurado en el que se permite variar los parámetros de transmisión cada trama de $10ms$.

Otra de las principales características de esta capa física es su estructura de canales. El canal físico dedicado para datos (DPDCH) tiene siempre asociado un canal físico dedicado de control (DPCCH). Este canal de señalización contiene señales piloto, comandos del control de potencia y opcionalmente indica el formato de transmisión utilizado en el DPDCH.

Un esquema de control de potencia básico regulado por medidas del SIR (relación señal interferencia) se utiliza tanto en el enlace ascendente como en el descendente. Este esquema se basa en la comparación del SIR ob-

tenido a la salida del receptor RAKE con el SIR objetivo requerido para asegurar la tasa de error (BER) que requiere ese servicio. Si el SIR medido es superior al SIR objetivo se pide al terminal que disminuya su potencia y al contrario, si el SIR medido es inferior se indica al terminal que incremente su potencia. La determinación de SIR objetivo se realiza a través de un bucle externo que ajusta dicho valor cada cierto tiempo de forma a garantizar el BER negociado para esa conexión. Este esquema de control de potencia es de gran importancia para la capa MAC como se comprobará en las siguientes secciones.

En el proceso de optimización de la capacidad del CDMAC no se ha considerado la multipropagación. Por lo tanto, el estudio siguiente es válido en un canal plano. Es decir entornos con un ancho de banda de coherencia superior al ancho de banda de la señal. En WCDMA este ancho de banda es de $W = 5Mhz$ con lo cual este estudio puede aplicarse a entornos con "delay Spreads"inferiores a $200ns$ es decir, escenarios urbanos e interiores. Futuras líneas de trabajo tratarán de incluir escenarios con multipropagación y receptores RAKE.

3 Una descripción global del CDMAC

En el protocolo CDMAC todos los usuarios que tienen nuevos paquetes para transmitir deben enviar una demanda. Existen diversas estrategias para minimizar el número de demandas necesarias, entre ellas el uso de demandas incluidas dentro de paquetes de datos (backlogged) siempre que sea posible. Las demandas indican el tipo de servicio, los valores de tiempo de expiración (timeout) de los paquetes a transmitir, el formato de transmisión utilizado en ese momento en el caso de demandas "backlogged"etc. La capa MAC lleva a cabo una lista de paquetes listos para transmisión ordenada según un esquema de prioridades. Los niveles de prioridad se asignan dependiendo del tipo de servicio, la política de control de trá£co, y los requerimientos de QoS relacionados con la temporización (ej. timeout). Los formatos de transmisión se ajustan dependiendo del número y tipo de demandas existentes para la trama siguiente (cuando se produzcan muchas demandas para la siguiente trama, los formatos de transmisión de los servicios se £jaran al formato que genere menos interferencias). Además de controlar los paquetes preparados para su transmisión, la estación de base (BTS) monitoriza las atenuaciones que sufren los usuarios con demandas activas. Esta información esta disponible gracias al la mantenimiento de un canal DPCCH por parte de los usuarios con servicios en tiempo real conectados a la célula, o bien a partir de la propia demanda cuando el DPCCH no esta disponible (nuevos usuarios o usuarios de datos con una nueva ráfaga de paquetes a transmitir). El sistema utiliza todos estos datos para calcular cuantos paquetes deben servirse en la trama siguiente, empezando por aquellos con mayor prioridad.

La BTS comunica su permiso a los usuarios a los que se permite transmitir. Este permiso de transmisión incluye la potencia a la que deben empezar a transmitir óptima para esa combinación concreta de usuarios.

Los sub-apartados siguientes incluyen un análisis más detallado del CDMAC: primero se describe el procedimiento para enviar las demandas de transmisión, después el plani£cador que ordena las demandas, y £nalmente los algoritmos utilizados para determinar la combinación de demandas que puede ser servida en la siguiente trama.

3.1 Envío de las demandas de transmisión

El procedimiento para realizar las demandas es ligeramente diferente dependiendo del tipo de servicio. Sin embargo como regla general lo que se intenta es minimizar la carga de señalización y el retardo que genera la etapa de reservas.

Respecto a los servicios en tiempo real (CBR y rt-VBR) la principal característica es el mantenimiento de un enlace de señalización (DPCCH) durante toda la conexión. Esto permite a estos terminales mantenerse sincronizados y con el control de potencia rápido activo aunque el sistema este utilizando un esquema de transmisión discontinuo.

La secuencia de trá£co producida por una fuente CBR es fácilmente prevista por el plani£cador de manera que esos servicios tan solo necesitan enviar una demanda inicial y la red preverá y plani£cará las transmisiones del resto de paquetes hasta el £nal de la conexión.

En el resto de servicios, el terminal enviará una nueva demanda únicamente cuando se produzca un incremento en su cola de paquetes esperando para ser transmitidos de forma que el plani£cador mantenga una modelo de conexión actualizado de todos los terminales.

En el caso de servicios rt-VBR (posiblemente el tipo de trá£co más abundante en los sistemas móviles futuros ya que corresponde a una fuente de voz comprimida con detección de actividad vocal) además de mantener un DPCCH durante toda la conexión también mantienen su código de "scrambling"particular en el enlace ascendente(ver [**?**])y un código de canalización de baja tasa (los más abundantes) que es utilizará para enviar las demandas de trasmisión. Es decir aunque estos servicios en tiempo real deban realizar demandas estas no se transmiten en modo de contención y tienen un control de potencia rápido. Obviamente como en el resto de tipos de trá£co el terminal incluirá sus nuevas demandas en paquetes que se estén transmitiendo ya siempre que esa posible.

Solamente los servicios de datos (nrt-VBR, ABR y UBR) así como los nuevos usuarios que acaban de ser aceptados por el controlador de admisión de llamadas (CAC) deben enviar sus demandas a través del canal de acceso común RACH en modo de contención siguiendo un protocolo ALOHA estabilizado.

3.2 Plani£cador

Si queremos una red donde se garanticen la QoS de las conexiones deben implantarse un conjunto de mecanismos de control de la QoS. El plani£cador propuesto a

continuación lleva a cabo la mayor parte del control de la QoS: establece una política de modelado y control de trá£co, proporciona una gestión e£ciente de los recursos y £nalmente ayuda a asegurar los requerimientos de BER de los diferentes servicios.

El modelado y control de trá£co se lleva a cabo a partir de la utilización de un regulador de trá£co "token bucket" [**?**]. Este modelo permite controlar el trá£co entrante dividiendo las demandas en conformadas y no conformadas. Las demandas conformadas se sirven primero y si queda algún espacio libre se transmiten las no conformadas.

Dentro de las categorías de demandas conformadas y no conformadas, estas se ordenan dependiendo del tipo de servicio en diferentes prioridades. La prioridad más alta corresponde a los servicios en tiempo real (CBR i rt-VBR), a continuación los servicios de tipo nrt-VBR y ABR y £nalmente la menor prioridad es para los servicios UBR.

Por lo tanto los paquetes conformados de tiempo real serán los primeros en servirse. Dentro de esta clase el orden de transmisión es establecerá según el tiempo de expiración de los paquetes. Con este mecanismo el protocolo trata de asegurar los requerimientos de retardo de este tipo de trá£co. A continuación se transmiten los paquetes conformados de los servicios nrt-VBR y ABR, dentro de esta clase los paquetes se ordenaran dependiendo del número de tokens disponibles para esa conexión, favoreciendo así las conexiones que menos han utilizado la velocidad de transmisión contratada. Después de los paquetes conformados se transmiten los no conformados organizados de forma equivalente y £nalmente se servirán los paquetes UBR si todavía queda espacio libre. Todas las conexiones UBR serán servidas siguiendo una política de colas equitativa (ej. Round Robin). Una vez ordenadas todas las demandas el siguiente paso consiste en determinar cuantas de esas demandas pueden ser transmitidas en la siguiente trama. El cálculo de cuantos paquetes pueden transmitirse se llevará a cabo a través de un algoritmo de estimación de la capacidad que se describirá en el siguiente apartado. Una vez que los paquetes hayan sido transmitidos los colas para cada una de las conexiones en el plani£cador se actualizarán en consecuencia.

3.3 Determinación del vector de potencia óptimo

Él número de paquetes transmitidos que puede soportarse en la siguiente trama se calcula a cada trama para asegurar los requerimientos de BER de todos los tipos de usuarios. Dado que el SIR es matemáticamente más tratable que el BER se utilizará este factor en su lugar. El protocolo MAC se aprovecha de los valores de SIR obtenidos por la función de control de potencia para las conexiones con demandas. Una vez que las limitaciones de SIR se han £jado para cada conexión el problema consiste en la determinación del vector de potencia óptimo que veri£ca dichas limitaciones y maximiza la capacidad del sistema.

Consideraremos el enlace ascendente del sistema W-CDMA descrito anteriormente. En este sistema la tasa de chip es £ja para todos los usuarios y por tanto el ancho de banda del sistema W, es utilizada por todos los terminales.

Si C es el número de células en el sistema y S el número de tipos de trá£co existentes. Entonces U_{cs} es el número de usuarios de clase s en la célula c y M_{csu} el número de códigos asignados al usuario u de la clase s en la celda c. Cada código asignado a un usuario tiene un requerimiento de QoS de la forma:

$$\gamma_{csum} = (\frac{E_b}{N_0})_{TARGETcsum}$$

obtenido directamente del SIR objetivo estimado por el bucle externo del mecanismo de control de potencia. Además cada terminal tiene especi£cado un límite de potencia máximo y una tasa de transmisión mínima para cada uno de sus códigos. Los límites de potencia están representados por el vector $\mathbf{p}$, con componentes p_{csum} y las tasas mínimas requeridas por $\mathbf{r}$, con componentes r_{csum}. Las ganancias de camino entre los diferentes usuarios y células se representan por el vector $\mathbf{G}$,$\{G_{c_1c_2,su}\}$ donde c_1 y c_2 son dos células del sistema. Además, se asume un ruido blanco gaussiano aditivo con una densidad espectral de potencia η_0. Para un sistema con múltiples células, la limitación de SIR para cada código viene dada por:

- Limitaciones de QoS:

$$\gamma_{cu} \leq \frac{W}{R_{cu}} \frac{P_{cu}G_{cc,u}}{\eta_0 W + \sum_{\substack{i=1}}^{C}\sum_{j=1}^{U_i}{}_{\forall(i,j)\neq(c,u)} P_{ij}G_{ic,j}} \quad (1)$$

- Limitaciones de potencia y velocidad de transmisión:

$$0 < P_{cu} \leq p_{cu}, \quad R_{cu} \geq r_{cu} \quad (2)$$

donde R_{cu} es la tasa de bit requerida para ese código y P_{cu} la potencia transmitida. Debe subrayarse el hecho de que en los términos correspondientes a la interferencia en la expresión anterior no esta el componente correspondiente a los posibles otros códigos de ese mismo usuario. Esto es debido a que todos los códigos del mismo usuario se ven afectados por las mismas perturbaciones en el canal y por tanto la ortogonalidad de esos códigos se mantiene en el receptor. El problema puede formularse entonces como sigue, para todos los códigos que transmiten en el sistema: Dadas las limitaciones anteriores, existe algún vector P y vector R que las cumplan?, sino, o bien algún usuario debe descartarse o bien debe relajar sus requerimientos. Este problema ha sido resuelto en [**?**] para el caso de una única célula, aquí extenderemos el resultado para múltiple células y además propondremos una aproximación que lo haga realizable.

3.3.1 Sistema unicelular sin restricciones de potencia

Para un vector dado de tasas de transmisión, si existe más de un vector de potencias adecuado seria deseable

escoger aquel que minimiza la potencia transmitida total. En una única célula con N usuarios el problema es entonces:

$$\text{Minimizar} \left\{ P_R = \sum_{i=1}^{N} P_i \right\} \tag{3}$$

sujeto a las limitaciones (1) y (2). Una observación fácilmente demostrable acerca de la solución es que *"Para la solución óptima todas las limitaciones de QoS se veri£can con la igualdad "* [**?**].

Así el vector de potencia óptimo puede obtenerse resolviendo un sistema de QoS ecuaciones en las potencias:

$$\gamma_i = \frac{W}{R_i} \frac{P_i G_i}{\eta_0 W + \sum_{\forall j \neq i} P_j G_j} \quad \forall i = 1...N \tag{4}$$

Si $R'_i = R_i \gamma_i$, obtenemos la ecuación matricial

$$\mathbf{A}\mathbf{P}^* = \eta_0 W \mathbf{1} \tag{5}$$

donde **A** corresponde a

$$\begin{pmatrix} WG_1/R'_1 & -G_2 & \dots & -G_N \\ -G_1 & WG_2/R'_2 & \dots & -G_N \\ -G_1 & -G_2 & \dots & \dots \\ \dots & \dots & \dots & \dots \\ -G_1 & \dots & \dots & WG_N/R'_N \end{pmatrix} \tag{6}$$

y $\mathbf{P}^* = [P_1^*, P_2^*, ..., P_N^*]^T$ es el vector de potencia óptimo y $\mathbf{1} = [1, 1, ..., 1]^T$. A través de operaciones elementales con las £las (resta de una £la con la siguiente) y utilizando la notación $K_i = (\frac{WG_i}{R'_i} + G_i)$ el sistema de ecuaciones correspondiente puede escribirse como:

$$\begin{aligned} P_1 K_1 - P_2 K_2 &= 0 \\ P_2 K_2 - P_3 K_3 &= 0 \\ &\vdots \\ P_N \frac{WG_N}{R'_N} - \sum_{i=1}^{N-1} P_i G_i &= \eta_0 W. \end{aligned} \tag{7}$$

De esta forma las potencias pueden expresarse como una función de P_N como:

$$P_i = \frac{P_N K_N}{K_i}$$

Sustituyendo esta solución en la última columna de (7) Obtenemos la siguiente expresión para P_i:

$$P_i = \frac{\frac{\eta_0 W}{G_i}}{(\frac{W}{R'_i} + 1)[1 - \sum_{j=1}^{N} \frac{1}{\frac{W}{R'_j + 1}}]}. \tag{8}$$

A partir de esta expresión obtenemos una condición de existencia derivada de la positividad de **P**:

$$\sum_{i=1}^{N} \frac{1}{\frac{W}{R_i} + 1} < 1. \tag{9}$$

Esta es una condición necesaria y su£ciente para que exista una solución. Si se satisface para un conjunto de tasas de transmisión y requerimientos de $\frac{E_b}{N_0}$, Las potencias pueden calcularse utilizando (4). Esto ilustra el hecho de que aunque no existan restricciones de potencia, no todos los requerimientos de $\frac{E_b}{N_0}$ y velocidad de transmisión pueden satisfacerse.

3.3.2 Sistema unicelular con restricciones de potencia

En este caso la existencia de potencia máximas implica limitaciones en el vector de potencias de la forma:

$$\frac{\frac{\eta_0 W}{G_i}}{(\frac{W}{R'_i} + 1)[1 - \sum_{j=1}^{N} \frac{1}{\frac{W}{R'_j + 1}}]} \leq p_i. \tag{10}$$

Con esta limitación en **P** La condición de existencia con limitaciones de potencia puede expresarse ahora como:

$$\sum_{i=1}^{N} \frac{1}{\frac{W}{R'_i} + 1} \leq 1 - \frac{\eta_0 W}{\min_{\forall i}[p_i G_i (\frac{W}{R'_i} + 1)]}. \tag{11}$$

3.3.3 Sistema multicelular sin restricciones de potencia

En el caso multicelular, con una red de C células la matriz **A** de (5) se convierte en:

$$\begin{pmatrix} \frac{WG_{11,1}}{R'_{1,1}} & -G_{11,2} & \dots & -G_{21,1} & \dots & -G_{C1,N_C} \\ -G_{11,1} & \frac{WG_{11,2}}{R'_{1,2}} & \dots & -G_{21,1} & \dots & -G_{C1,N_C} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -G_{1i,1} & -G_{1i,2} & \frac{WG_{ii,1}}{R'_{i,1}} & \dots & \dots & -G_{Ci,N_C} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -G_{1C,1} & -G_{1C,2} & \dots & \dots & \dots & \frac{WG_{CC,N_C}}{R'_{C,N_C}} \end{pmatrix} \tag{12}$$

donde $G_{ij,k}$ corresponde a la ganancia del camino entre el usuario k de la célula i y la estación de base j. La matriz **A** de compone ahora de C submatrices cada una correspondiente a una célula del sistema. Si utilizamos las mismas transformaciones que en el caso unicelular para cada submatriz de **A** y la de£nición equivalente: $K_{i,j} = (\frac{WG_{ii,j}}{R'_{i,j}} + G_{ii,j})$. La matriz anterior corresponde al sistema de ecuaciones:

$$\begin{aligned} P_{11} K_{11} - P_{12} K_{12} &= 0 \\ P_{12} K_{12} - P_{13} K_{13} &= 0 \\ &\vdots \\ P_{1N_1} \frac{WG_{11,N_1}}{R'_{1N_1}} - \sum_{i=1}^{N-1} P_{1i} G_{11,i} - \sum_{c=2}^{C} \sum_{i=1}^{N_c - 1} P_{ci} G_{c1,i} &= \eta_0 W \\ P_{21} K_{21} - P_{22} K_{22} &= 0 \\ &\vdots \end{aligned}$$

Con estas manipulaciones y siguiendo el mismo razonamiento desarrollado en el caso unicelular obtenemos

la solución siguiente para el vector óptimo de potencias $\mathbf{P}^*$,

$$\mathbf{P}^{*T} = \left[\frac{D_1}{K_{1,1}}, \frac{D_1}{K_{1,2}}, \ldots, \frac{D_1}{K_{1,N_1}}, \frac{D_2}{K_{2,1}}, \ldots, \frac{D_C}{K_{C,N_C}}\right]. \quad (13)$$

Donde como generalización del caso unicelular obtenemos un vector de constantes reales positivas D_1, D_2,..., D_C cada una correspondiendo a una célula. Por tanto, hemos transformado un sistema de ecuaciones lineales con una dimensión de $\{\sum_{i=1}^{C} N_i\}$ en un nuevo sistema de ecuaciones en las constantes (una ecuación por célula)de la forma:

$$\mathbf{B}D^* = \eta_0 W\mathbf{1} \quad (14)$$

donde $\mathbf{B}$ en un sistema con M células toma la forma:

$$\mathbf{B} = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \ldots & \alpha_{1C} \\ \alpha_{21} & & \ldots & \alpha_{2C} \\ \vdots & \vdots & & \vdots \\ \alpha_{C1} & & \ldots & \alpha_{CC}, \end{pmatrix} \quad (15)$$

donde las α_{ij} son los factores de interferencia y representan la cantidad de interferencia que llega a la BTS_i desde todos los móviles conectados a la célula j

$$\alpha_{ii} = 1 - \sum_{k=1}^{N_i} \frac{G_{ii,k}}{K_{i,k}} \quad \text{y} \quad \alpha_{ij} = -\sum_{k=1}^{N_j} \frac{G_{ji,k}}{K_{j,k}}.$$

En este caso no es posible derivar una condición de existencia sin resolver (14) y encontrar D_c. Hay que subrayar que la resolución de (14) implica la existencia de un nodo central al cual las BTS deben comunicar sus factores de interferencia computados previamente α_{ij} con el resto de las células del sistema. El caso multicelular tendrá solución si las condiciones siguientes derivadas de la positividad de $\mathbf{P}_*$ se cumplen:

$$\sum_{j=1}^{N_c} \frac{1}{\frac{W}{R'_{c,j}} + 1} < 1 \quad \text{y} \quad D_c > 0 \quad \forall c \in [0, C] \quad (16)$$

3.3.4 Caso multicelular con restricciones de potencia

La única diferencia es que la limitación de potencia hace que la condición de existencia sobre el vector $\mathbf{D}_*$ sea más restrictiva:

$$D_c \leq \min_{\forall i \in [0, U_c]} [p_{ci} G_{cc,i} (\frac{W}{R'_{ci}} + 1)] \quad \forall c \in [0, C]. \quad (17)$$

Añadiendo esta nueva expresión a (16) obtenemos el conjunto de condiciones necesario para tener una solución en el caso multicelular con restricciones de potencia.

3.3.5 Implementación distribuida del caso multicelular

Obviamente el retardo y el coste computacional asociado con esta implementación centralizada no puede ser asumido por un protocolo MAC en un sistema real, en cambio una implementación distribuida más simple como la presentada a continuación si podría ser factible. Esta propuesta se basa en la hipótesis de que la la suma total de interferencias de otras células afectando a una BTS concreta puede modelarse como una variable aleatoria gaussiana de variación lenta que no cambia signi£cativamente de una trama a la siguiente.

La validez de esta hipótesis ha sido veri£cada utilizando la herramienta de simulación desarrollada para evaluar las prestaciones de este protocolo. Los resultados de las simulaciones revelan que los cambios que sufre la interferencia externa entre una trama y la siguiente en escenarios con cargas de trá£co mixto(voz y datos) medias o altas no afectan signi£cativamente al throughput del sistema. Por tanto el término de interferencia externa medido en la trama anterior puede utilizarse para los cálculos de la trama actual. Esta hipótesis implica que la BTS es capaz de medir la interferencia externa y que puede utilizar esa medida como una estimación del nivel de interferencia que sufrirá la siguiente. Así la potencia de cada móvil j en la célula i puede ser calculado localmente como:

$$P_{c,i} = \frac{D_c}{K_{c,i}}, \quad (18)$$

donde

$$D_c = \frac{\eta_0 W + I_{EXT_c}}{(1 - \sum_{i=1}^{N_c} \frac{G_{cc,i}}{K_{c,i}})}. \quad (19)$$

y la condición de existencia para el caso limitado en potencia es:

$$\sum_{j=1}^{N_c} \frac{1}{\frac{W}{R'_{c,j}} + 1} < 1 - \frac{\eta_0 W + I_{EXT_c}}{\min_{\forall i \in [0, U_c]} [p_{ci} G_{cc,i} (\frac{W}{R'_{c,i}} + 1)]}. \quad (20)$$

3.4 Algoritmo iterativo de determinación de la capacidad

Empezando por la lista ordenada de demandas proporcionada por el plani£cador y utilizando las expresiones obtenidas en el apartado anterior, un algoritmo iterativo determinará cuantos paquetes pueden transmitirse en la trama siguiente.

En cada iteración el algoritmo primero calcula las constantes D_C de cada célula utilizando (14) o (19) dependiendo de si se utiliza la opción centralizada o distribuida respectivamente. A continuación se calculan las potencias de todos los móviles usando (13) o (18) respectivamente, después se veri£ca que las potencias de los terminales son inferiores a la potencia máxima para ese terminal y servicio. La veri£cación en el caso centralizado implica que (16) y (17) deben ser respetadas, mientras que (20) será la que se utilice en el caso distribuido.

En la primera iteración el sistema intenta servir todas las demandas de la lista, si no es posible, un algoritmo de eliminación debe ser utilizado para seleccionar la demanda que será descartada en la próxima iteración. La solución óptima requeriría la implementación de un

algoritmo de fuerza bruta (BFA) con un coste computacional muy alto. Para reducir este coste, un algoritmo sub-óptimo del tipo DS/CDMA-SMIRA [**?**] adaptado a la operación con múltiples tipos de trá£co será utilizado en su lugar. Partiendo de la lista total de demandas, seleccionamos el grupo con menor prioridad y dentro de ese grupo eliminamos:

- el móvil con el R' más elevado,es decir el móvil que requiere más recursos en el caso que (16) no se cumpla (y por tanto la transmisión de todas las demandas no es posible incluso sin calcular las potencias),
- o el móvil que genera un nivel más elevado de interferencia en el sistema, es decir, el móvil para el cual las sumas siguientes son mayores:

$$S_{c,u} = \sum_{\substack{i=1 \\ i\neq c}}^{C} \sum_{\substack{j=1 \\ j\neq u}}^{U_i} P_{ij} \frac{G_{ic,j}}{G_{cc,u}} \tag{21}$$

$$S^T_{c,u} = P_{cu} \sum_{\substack{i=1 \\ i\neq c}}^{C} \sum_{\substack{j=1 \\ j\neq u}}^{U_i} \frac{G_{ci,u}}{G_{ii,j}} \tag{22}$$

donde, $S_{c,u}$ representa la interferencia total en el receptor de la BTS $c-th$ correspondiente al móvil u asignado a esta célula y $S^T_{c,u}$ representa la interferencia total a otras BTS producida por el móvil $u-th$ de la célula c.

En este punto hemos obtenido el máximo numero de paquetes que pueden transmitirse y las potencias que los terminales deben utilizar para iniciar la transmisión en la trama siguiente. La hipótesis implícita aquí es que las posibles variaciones de potencia sufridas por todos los móviles durante los 10ms que dura una trama se verán estadísticamente compensados y por tanto (16) y (17)serán válidos durante toda la trama.

Esta hipótesis básica ha sido investigada a través de simulaciones con diferentes combinaciones de trá£co y niveles de carga. Se han realizado simulaciones "estáticas", múltiples iteraciones manteniendo el mismo trá£co de entrada y posición de los usuarios (como ocurre dentro de una trama) pero con ganancias de camino diferentes. Los resultados revelan que la hipótesis anterior es razonablemente válida incluso en el caso en que las ganancias de los caminos utilizadas en las iteraciones sean completamente incorreladas y por tanto se aproximará todavía mejor al comportamiento esperado en escenarios reales donde las ganancias de camino en un intervalo de $10ms$ están fuertemente correladas.

4 Descripción de la simulación

Las prestaciones del CDMAC se han evaluado a través de una herramienta que incorpora todos los procedimientos descritos en los apartados anteriores. Se ha utilizado un mallado hexagonal con 19 células. El canal sufre desvanecimientos Rayleigh no selectivos en frecuencia con unas pérdidas de camino proporcionales a $10^{\varepsilon/10}\gamma r^{-\alpha}$ donde r es la distancia del terminal hasta la base, γ la componente Rayleigh y ε es una v.a. gaussiana de media cero y desviación típica σ.

El principal objetivo es analizar el CDMAC en un escenario con múltiples tipos de trá£co. Los tipos de trá£co utilizados en las simulaciones han sido: voz (rt-VBR), trá£co audio de alta calidad (CBR), y trá£co de datos (ABR y UBR). La fuente de trá£co CBR utilizada representa la producción de un ¤ujo constante de paquetes de audio FM estéreo con velocidad constante de $64Kbps$. Un simple modelo de Bernouilli con diferentes probabilidades decide a cada trama si se transmite o no, un paquete ABR y UBR. El modelo de trá£co adoptado para las fuentes de voz es el mismo modelo on-off con supresión de silencios utilizado en [**?**] [**?**] con idénticos parámetros. La tabla 1 incluye los principales parámetros utilizados en las simulaciones.

5 Resultados de las simulaciones

Las simulaciones realizadas muestran las prestaciones de CDMA. Fig.**??** presenta el efecto de la carga de las células adyacentes sobre la célula central. Para el caso unicelular el máximo throughput coincide con el máximo teórico obtenido usando 9 (esta £gura corresponde al caso sin restricciones de potencia), hay que hacer notar que este es el throughput máximo si no se utiliza transmisión multicódigo. Es decir un único usuario transmitiendo con múltiples códigos no se ve afectado por este límite.

La coexistencia de múltiples tipos de trá£co se muestra en Fig.**??** demostrando claramente el funcionamiento del plani£cador. Esta £gura se ha obtenido aumentando simultáneamente los usuarios de todos los tipos de trá£co, aunque manteniendo siempre las mismas proporciones: 25% de usuarios de voz, 5% de usuarios CBR y 22.5% para los dos tipos de trá£co de datos que corresponde a un porcentaje de trá£co ofrecido al sistema del 25% para cada tipo. Para cargas bajas, todos los paquetes son servidos por el sistema, sin embargo a medida que el trá£co en tiempo real aumenta el trá£co UBR primero y luego el trá£co ABR van disminuyendo su throughput. Durante un cierto intervalo el trá£co CBR es superior al de voz, esto se debe simplemente a que en ese punto no hay más paquetes de trá£co ofrecidos al sistema, en saturación la vos alcanza un mayor throughput debido a que su menor tiempo de expiración le da una relativa mayor prioridad.

Fig.**??**,Fig.**??** y Fig.**??** han sido obtenidas utilizando también el incremento gradual simultaneo en la carga de todos los tipos de trá£co.Fig.**??** muestra la mejora en términos del número de paquetes de servicios en tiempo real perdidos con y sin el plani£cador propuesto para la zona de cargas operativas. De hecho para el servicio de voz la cota del 1% de paquetes descartados se alcanza con entre 92 y 96 usuarios totales para el ejemplo con plani£cador mientras que sin plani£cador esa misma cota se alcanza con la mitad de usuarios totales. De la misma forma, Fig.**??** presenta la mejora en el retardo sufrido por los paquetes abr mientras el trá£co en tiem-

Parámetros de la simulación		
Parámetros del Canal		
σ	8dB	
α	4	
Parámetros de Potencia		
Potencia máxima del terminal de voz	0,8W	
Potencia máxima del terminal de datos	2W	
Parámetros de tráfico		
Tráfico de voz	Timeout	30ms
	Tasa contratada	8Kbps
	Tasa máxima	32Kbps
	E_b/N_0	7dB
Tráfico CBR	Timeout	90ms
	Tasa contratada	64Kbps
	Tasa máxima	86Kbps
	E_b/N_0	8,5dB
Tráfico de Datos abr/ubr	Timeout	Na
	Tasa contratada	16Kbps (sólo abr)
	Tasa máxima	32Kbps
	E_b/N_0	10dB

Tabla 1: Tabla 1

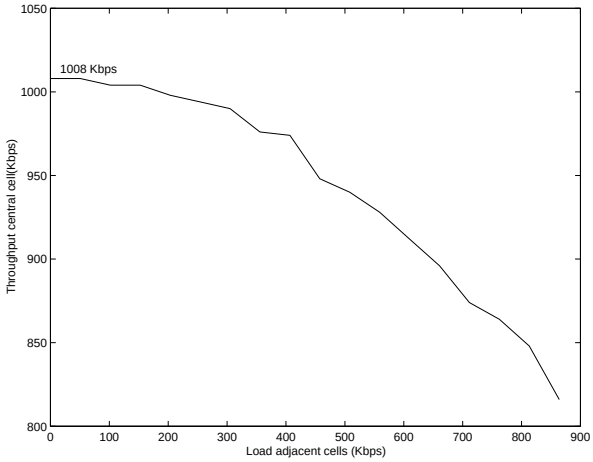


Figura 1: Efecto de la carga en las células adyacentes

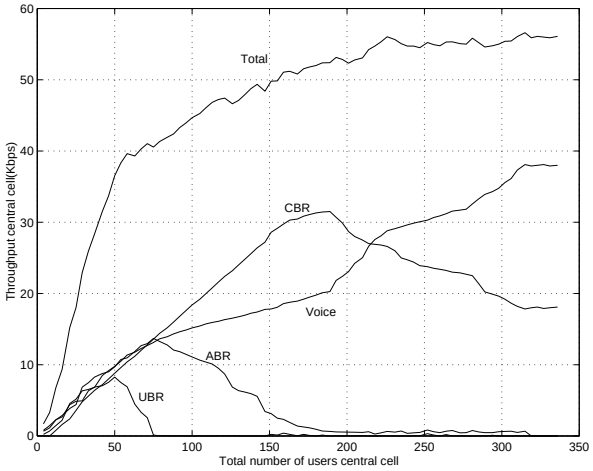


Figura 2: Throughput de los diferentes tipos de tráfico

po real no sea demasiado alto esto es debido a que en el caso sin planificador la probabilidad de elegir un paquete abr para transmitir depende del numero de paquetes ABR respecto al número de paquetes totales y es un valor pequeño. Cuando la carga en tiempo real aumenta la situación se invierte y el retardo de los dos servicios de datos se incrementa rápidamente a medida que el planificador los deja sin recursos y su throughput tiende a cero. En el caso del tráfico UBR Fig.**??**, el tipo de tráfico con menor prioridad, es lógico y deseable que se vea desfavorecido en el caso con planificador.

6 Conclusiones

En este documento, se describe un protocolo MAC que tiene en cuenta el hecho de que CDMA es una técnica limitada por el nivel de interferencia. Con este esquema puede explotarse la capacidad y flexibilidad de los sistemas CDMA. Además el protocolo es capaz de mantener requerimientos de QoS gracias a sus mecanismos de control y gestión de tráfico realizados por su planificador. Otras ventajas de este protocolo serían la facilidad con que pueden incorporarse a este protocolo conceptos de calidad de servicio "soft", es decir, si el número de paquetes descartados para transmitir aumenta por encima de un cierto límite, el protocolo CDMAC disminuirá los requerimientos de QoS de las conexiones activas. Esto puede llevarse a cabo, si el controlador de admisión de llamadas (CAC) es capaz de negociar un contrato de transmisión con parámetros de calidad variables dentro de un margen aceptable. Así permite a la red usar los parámetros menos restrictivos en situaciones de carga elevada. En este caso el CDMAC podría empezar por ejemplo disminuyendo la velocidad en los servicios ABR, si no fuera suficiente también se aplicarían velocidades de transmisión y valores de BER menores en los demás servicios de datos, otra medida podría ser la disminución en el BER de los servicios de voz y vídeo

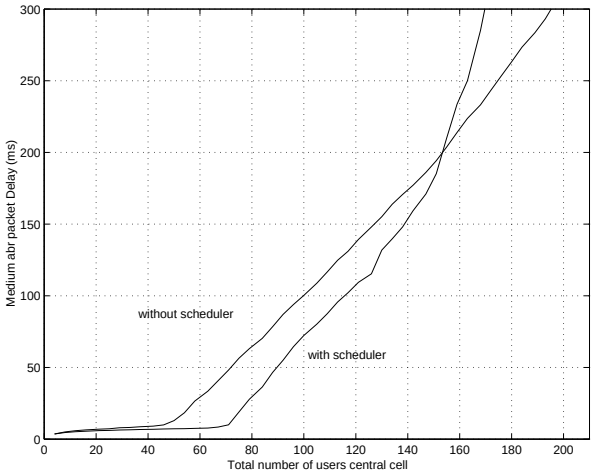


Figura 3: Retardo medio sufrido por el trá£co ABR

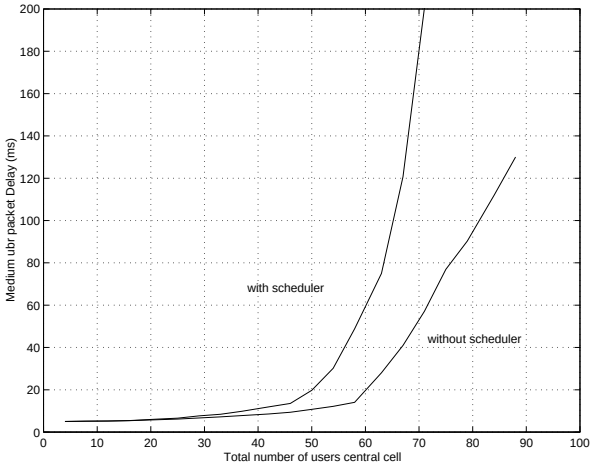


Figura 4: Retardo medio sufrido por el trá£co UBR

por ejemplo. Todo esto puede llevarse a cabo fácilmente en la capa MAC a partir de cambios en los modelos que tiene de cada conexión el plani£cador asegurando la estabilidad del sistema sin la intervención de capas superiores.

Futuras líneas de investigación incluyen la evaluación del efecto de la multipropagación en el sistema y la conexión entre el CDMAC y el CAC y así poder aplicar conceptos de "QoS soft"en el sistema. Además se pretende mejorar la herramienta de simulación incorporando simulaciones dinámicas para permitir una evaluación más detallada del protocolo propuesto.

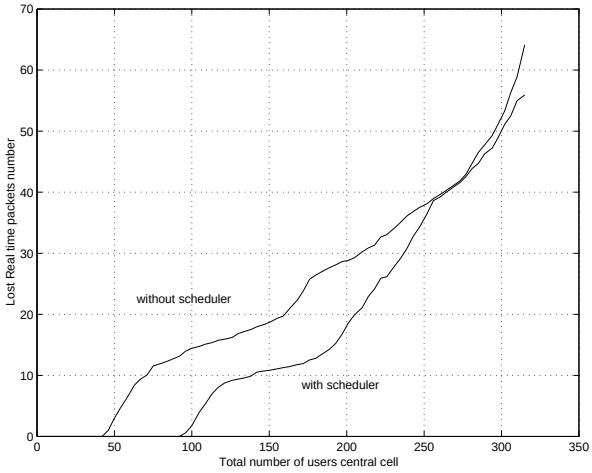


Figura 5: Paquetes perdidos (voz y cbr)

Referencias

[1] I. Akyildiz, J. McNair and L. Carrasco, “Medium Access Control Protocols for Multimedia Traf£c in Wireless Networks ,” *IEEE Network Magazine*, Vol.13, NO.4 July/August 1999.

[2] T. Ojanpera, R. Prasad “An Overview of Air Interface Multiple Access for IMT-2000/UMTS,” *IEEE Communications Magazine*, Vol. 36, No. 9, pp. 70-79, September 1998.

[3] F. Ovesjo, E. Dahlman, T. Ojanpera, A. Toskala, and A. Klein, “FRAMES Multiple Access Mode 2- Wideband CDMA,” *Proc. IEEE Int. Conf. on Personal Indoor and Mobile Radio Commununications (PIMRC’97)*, Helsinki, Finland, September 1997.

[4] N. Passas et al., “Quality-of-Service-oriented Medium Access Control for Wireless ATM Networks,” *IEEE Communications Magazine*, November 1997.

[5] A. Sampath, P.Sarath Kumar, J. Holtzman, “Power Control and Resource Management for a Multimedia CDMA Wireless System” *Proc. IEEE Int. Conf. on Personal Indoor and Mobile Radio Commununications (PIMRC’95)*, 1995.

[6] G. Femenias, et al., “Transmitter Power Control for DS/CDMA Cellular Mobile Radio Networks” *Proc. IEEE Int. Conf. on Personal Indoor and Mobile Radio Commununications (PIMRC’95)*, 1995, pp. 46-49.

[7] A.E. Eckberg, et al, “Meeting the challenge: Congestion and Flow Control Strategies for Broadband Information Transport ” *Proc. IEEE GLOBECOM’89*, pp. 49.3.1-49.3.5, 1989.

[8] S. Nanda,D.J. Goodman and U. Timor, “Performance of PRMA: a packet voice protocol for Cellular Systems,” *IEEE Transactions on Vehicular Technology*, Vol. 40, No. 3, pp. 584-598, 1991.

[9] G. Bianchi,F. Borgonovo et al, “C-PRMA: A Centralized Packet Reservation Multiple Acces System for Local Wireless Communications” *IEEE Transactions on Vehicular Technology*, Vol. 46, No. 2, pp. 442-435, 1997.

Gestión Integrada de la Calidad de Servicio para VoIP en Redes UMTS Interconectadas con Redes IP Externas

G. Gómez, R. Cuny, H. Montes y J. F. Paris*
Nokia Networks, IP Mobility Networks, WNP-Málaga SCT, P.T.A, Málaga (Spain)
*Dept. Ingeniería de Comunicaciones, Universidad de Málaga, Campus de Teatinos s/n, Málaga (Spain)
E-mail: {ext-gerardo.gomez, renaud.cuny, ext-hector.montes}@nokia.com, paris@ic.uma.es

Abstract. *In this work, an end-to-end Quality of Service (QoS) framework for Voice over IP (VoIP) services when UMTS interworks with an external IP packet data network (IP-PDN) is described. For this scenario, the interaction between UMTS and IETF's protocols and mechanisms for a VoIP call is analyzed. By signaling flowcharts, it is shown that both groups of protocols and mechanisms can co-operate to provide seamless end-to-end VoIP services. Finally, a more detailed description of the interaction between UMTS and RSVP protocols is tackled, proposing a possible solution about the mapping between the QoS attributes used by both protocols.*

1 Introducción

Los servicios de Voz sobre IP (*Voice over IP*, VoIP), como la telefonía sobre IP, se encuentran de plena actualidad en el negocio de las redes móviles [1]. La capacidad de proveer servicios VoIP es un aspecto clave para la evolución hacia las redes de telefonía basadas en conmutación de paquetes IP. Una tendencia de estandarización interesante es el desarrollo de una arquitectura basada en IP para UMTS que permita la convergencia entre las redes de telefonía y las tecnologías basadas en IP [2].

La gestión y el control de la calidad de servicio entre usuarios finales representa uno de los principales desafíos en UMTS. La capacidad de suministrar calidad de servicio extremo a extremo implica que los operadores de Redes Móviles (*Public Land Mobile Networks*, PLMN) deberán ofrecer servicios portadores, entre origen y destino, con unas características muy concretas. En UMTS versión 1999 se aborda el problema de la garantía de la calidad de servicio para un servicio portador dentro de UMTS [3]. Sin embargo, para extender dicho servicio portador de extremo a extremo y con garantías de calidad, deberá tenerse en cuenta la interacción con las Redes de Paquetes externas basadas en IP (*IP Packet Data Networks*, IP-PDN). Esta necesidad ya se ha identificado para la versión 4 y posteriores de UMTS.

Para proporcionar un servicio portador extremo a extremo de VoIP con una determinada calidad de servicio, deben tenerse en cuenta dos aspectos básicos: la calidad de servicio y el control de la llamada. En primer lugar debe abordarse la interacción entre los protocolos de calidad de servicio usados en UMTS y los protocolos y mecanismos usados en las IP-PDNs. En segundo lugar, una vez establecida la arquitectura de calidad de servicio extremo a extremo, los servicios VoIP requieren que los mecanismos y protocolos de calidad de servicio, tanto de UMTS como de las redes externas IP-PDN, cooperen con los protocolos de control de llamada como el *Session Initiation Protocol* (SIP) del IETF o el H.323 del ITU.

En este trabajo se analiza, mediante diagramas de flujo, la integración de la calidad de servicio entre UMTS y IP-PDNs para las distintas fases de una llamada VoIP de manera conjunta con el control de dicha llamada. Se presta especial atención a la señalización en el plano de control para el establecimiento y liberación de la llamada. Para la reserva de recursos en la IP-PDN externa se ha considerado el uso del *Resource Reservation Protocol* (*RSVP*) del IETF [4], así como la utilización de SIP para el control de la llamada al nivel de aplicación [5].

El resto de este artículo está organizado como se detalla a continuación. En la sección 2 se describe la arquitectura de calidad de servicio extremo a extremo para el escenario considerado. En la sección 3 se presenta el análisis de la señalización para una llamada VoIP. En la sección 4 se profundiza en la interacción entre los protocolos UMTS y RSVP. En la sección 5, se plantea la posibilidad de incorporar a la arquitectura de la sección 2, protocolos del IETF específicos para preservar la calidad de servicio durante la llamada en el plano de usuario. Por último, en la sección 6 se establecen las conclusiones finales.

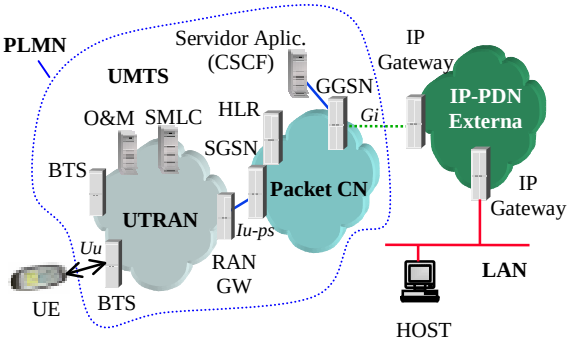


Figura 1: Arquitectura de red extremo a extremo.

2 Arquitectura para la Calidad de Servicio Extremo a Extremo

2.1 Descripción General

En la Fig.1 se muestra una visión general de la arquitectura de red considerada, cuando UMTS interacciona con redes externas de datos basadas en IP. En [6] y [7] pueden encontrarse descripciones detalladas de las entidades, interfaces y protocolos en UMTS, que suponemos son familiares para el lector. Además del Equipo de Usuario (*User Equipment*, UE), las principales entidades involucradas en la gestión de la calidad de servicio son:

- *UMTS Terrestrial Radio Access Network* (UTRAN)
- *Serving GPRS Support Node* (SGSN)
- *Gateway GPRS Support Node* (GGSN)
- Nodos RSVP: IP *Gateways*, IP *routers* intermedios y el *host*.

La calidad de servicio extremo a extremo en UMTS versión 4 y posteriores se basa en el concepto de Servicio Portador IP (*IP Bearer service*, IP BS). Como se muestra en la Fig. 2, un servicio portador IP consiste en la extensión del servicio portador UMTS definido en la versión 1999 [3] para tener en cuenta la calidad de servicio en la IP-PDN externa.

RSVP se usa para la coordinación del control de admisión en la red externa y para la reserva de recursos en cada nodo RSVP. En el modelo considerado, el GGSN soporta RSVP y es el nodo encargado de la negociación de calidad de servicio entre la red UMTS y la red externa IP-PDN. Se supone que el UE no soporta RSVP.

El modelo basado en políticas para IP (IP *Policy Model*) permite crear una arquitectura integrada para la gestión de los servicios portadores IP. Las políticas representan acuerdos en el nivel de servicio (*Service Level Agreements*, SLAs) entre proveedores de servicio y usuarios. Los SLAs especifican un conjunto de reglas acordadas para el control de admisión, de forma que ésta no se base únicamente en la disponibilidad de los recursos solicitados, sino en otros parámetros como: calidad de servicio, seguridad u otros aspectos de funcionamiento de red.

En la Fig. 2 se observa el uso del protocolo *Common Open Policy Service* (COPS) [8] del IETF entre el *Policy Control Function* (PCF), situado en principio en un servidor de políticas, y el *Policy Enforcement Function* (PEF) situado en el GGSN. En el modelo de políticas de tipo *outsourcing*, el PCF responde activamente a las peticiones provenientes del PEF.

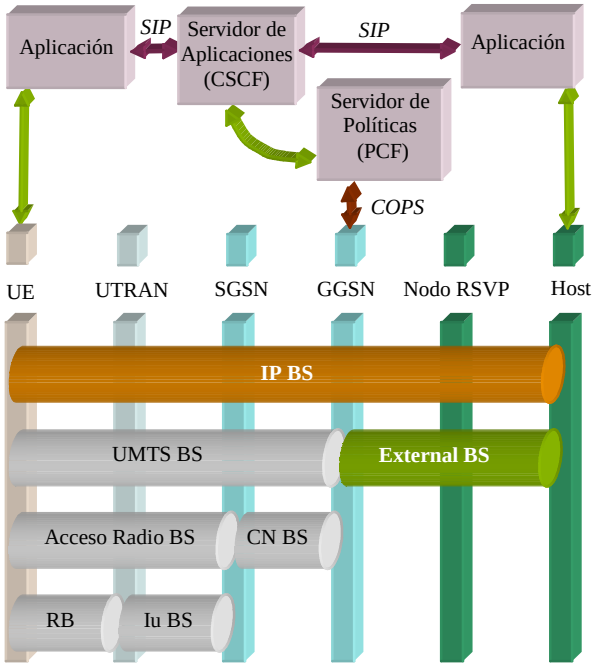


Figura 2: Arquitectura de calidad de servicio extremo a extremo

El PEF en el GGSN recibe decisiones del PCF, pudiendo actuar como una puerta que restringe el conjunto de destinos IP alcanzables por un determinado servicio.

2.2 Integración del Control de la Llamada y los Mecanismos de Calidad de Servicio

Al igual que en redes de conmutación de circuitos, las redes de conmutación de paquetes necesitan alguna entidad encargada de coordinar las llamadas VoIP mediante el protocolo SIP. En UMTS, esta entidad se conoce como *Call State Control Function* (CSCF) y suele estar localizada en el servidor de aplicaciones, donde también se podría ubicar el PCF (ver fig. 1 y 2). Bajo esta suposición, la información referente a las llamadas puede ser compartida por ambas entidades (CSCF y PCF), por lo que únicamente se necesita un interfaz abierto (COPS) para interactuar con el GGSN.

3 Análisis de una Llamada VoIP con Calidad de Servicio

3.1 Descripción del Escenario

En esta sección se presenta el análisis de un servicio VoIP a través de la arquitectura de red representada en la Fig.1. El estudio se centra en una llamada VoIP originada por el UE hacia el *host* situado en la red externa, de manera que se requieren características específicas de calidad de servicio. Se ha asumido la necesidad de usar dos *Packet Data Protocol* (PDP) *Contexts* en UMTS (ver [6]).

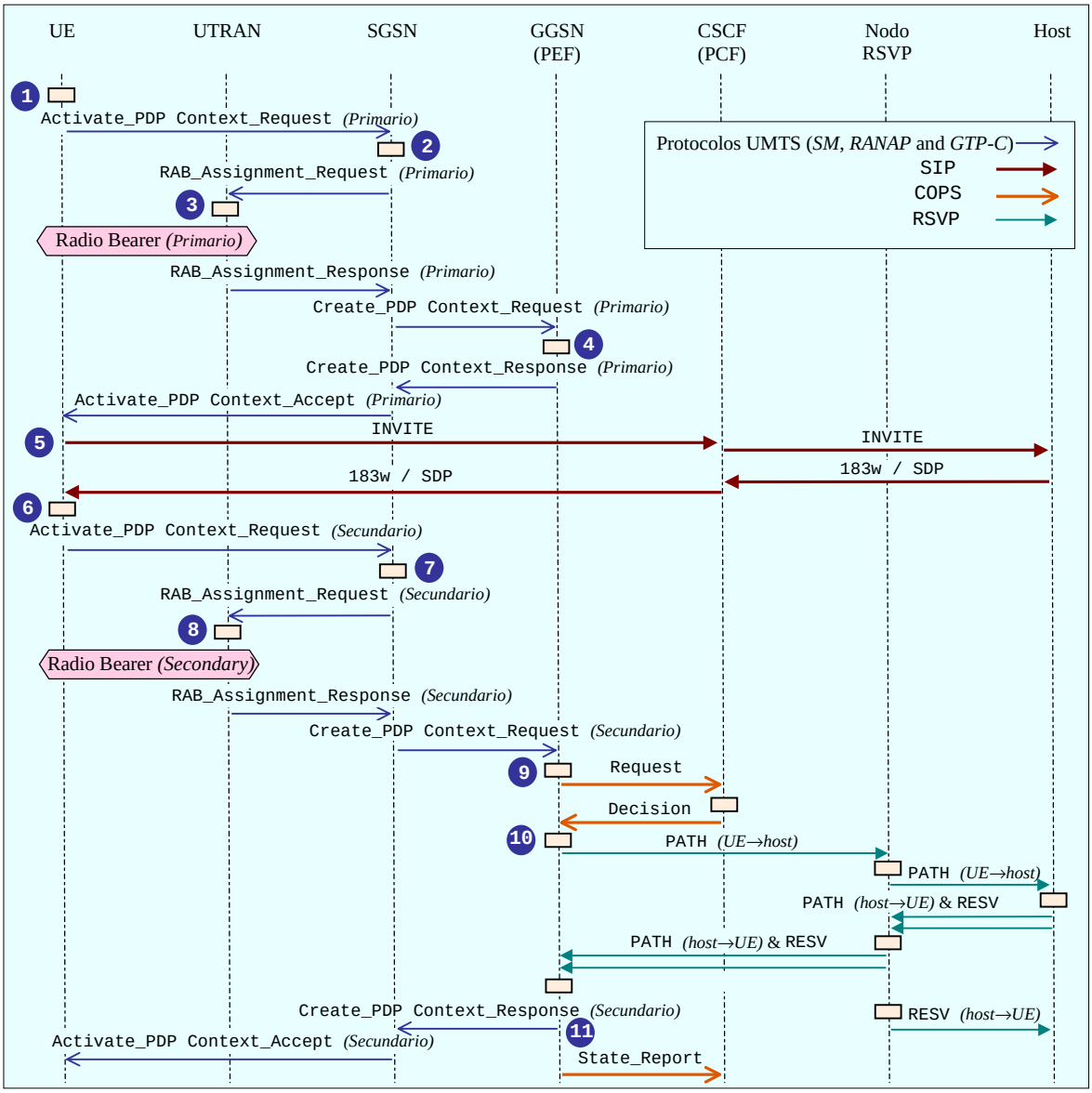


Figura 3: Diagrama de flujo del establecimiento de llamada.

El *PDP Context* primario se dedica a la señalización en el plano de control (SIP), mientras que el *PDP Context* secundario se usa para los datos de usuario y el control de dichos datos (p.e. *Real Time Protocol (RTP)* y *Real Time Control Protocol (RTCP)*). Como se explica posteriormente, la duración de la llamada se puede dividir cronológicamente en tres fases: establecimiento de la llamada, utilización del servicio y liberación de la llamada.

3.2 Establecimiento de la Llamada

Esta fase se extiende desde la petición de la activación del servicio por parte del UE hasta el momento en el cual la aplicación comienza a enviar datos de usuario. Toda la señalización asociada a esta fase es soportada por el plano de control a través de diferentes funciones de gestión de la calidad de servicio como son: gestión del servicio portador, subscripción, translación y admisión (ver [3] para más detalles). Estas funciones del plano de control están distribuidas en diferentes capas a través de varias entidades de la red. Asumiendo que el establecimiento de la llamada es satisfactorio, en la Fig. 3 se muestra un diagrama de flujo con una descripción detallada del desarrollo de esta fase.

Paso 1- Los requisitos de calidad de servicio de la aplicación son mapeados en atributos de calidad de servicio de UMTS. Puesto que el primer *PDP Context* es usado para señalización SIP, éste requiere baja velocidad binaria y alta fiabilidad. Por tanto, para este caso, resulta apropiado un perfil de calidad de servicio con clase de tráfico *Interactive*, alta prioridad y baja tasa de error. Un mensaje desde el UE al SGSN a nivel del protocolo *Session Management* (SM) inicia el procedimiento de activación del *PDP Context* primario.

Paso 2- Después de que el SGSN valide el servicio a dicho usuario, éste realiza un control de admisión basado en su estado interno, comprobando los *buffers*, carga de la CPU, etc… Posteriormente, el SGSN mapea los atributos de calidad de servicio en

atributos de servicios portadores al nivel de acceso radio (*Radio Access Bearer*, RAB, ver Fig.2), disparando un procedimiento de asignación de RAB en UTRAN mediante el *Radio Access Network Application Protocol* (RANAP).

Paso 3- El control de admisión en UTRAN está basado principalmente en la disponibilidad de recursos radio. Una vez que el nuevo *PDP Context* se ha aceptado, los atributos RAB son mapeados a parámetros *Radio Bearer* (RB) correspondientes a las capas física y de enlace (como los códigos de ensanchamiento, necesidad de retransmisiones,…). Tras establecer un RB de estas características, el SGSN utiliza el *GPRS Tunneling Protocol* en el plano de control (GTP-C) para indicar al GGSN el establecimiento del *PDP Context*.

Paso 4- Debido a que el primer *PDP Context* no se usa para tráfico de tipo tiempo real, no se requiere realizar reserva de recursos en la red externa. El GGSN realiza un control de admisión similar al realizado por el SGSN y, posteriormente, notifica al SGSN el correcto establecimiento del PDP Context. El SGSN lo notifica al UE mediante un mensaje con el protocolo SM.

Paso 5- Una vez que el servicio portador está disponible para la señalización SIP, se inicia un diálogo a nivel de aplicación entre el UE y el *host*, intermediado por el CSCF, con el objeto de establecer una sesión SIP. En este paso, se realiza la negociación de los parámetros de la aplicación (codificadores de voz, etc...) [9].

Pasos 6 a 8- El procedimiento de activación del *PDP Context* secundario se realiza de forma similar a la descrita en los pasos 1, 2 y 3. Sin embargo, existen algunas diferencias debido a que el *PDP Context* secundario se va a emplear para datos de usuario VoIP. Ello implica que el perfil de calidad de servicio de este *PDP Context* ha de utilizar clase de tráfico *Conversational*, unos requisitos de retardo de transferencia severos (media y varianza) y una tasa de error no demasiado estricta. Además, se necesita realizar reserva de recursos en toda la red para soportar el perfil de calidad de servicio descrito para la llamada de VoIP.

Paso 9 - Una vez que el GGSN lleva a cabo un control de admisión local basado en su propia capacidad, similar al realizado por el SGSN, éste consulta al PCF (localizado junto al CSCF en el servidor de aplicaciones) enviando un mensaje mediante el protocolo COPS.

Paso 10 - El GGSN mapea los atributos de calidad de servicio UMTS en atributos RSVP con objeto de comenzar la reserva de recursos en la red externa. Esta translación de parámetros no es directa ya que no hay una correspondencia entre ambos tipos de atributos (por ejemplo, no existe atributo de retardo en RSVP, véase la sección 4 para más detalles). En cada nodo RSVP se lleva a cabo un control de admisión y, finalmente, se establece un camino bidireccional con los recursos necesarios a lo largo de la IP-PDN externa.

Paso 11 - El GGSN confirma al SGSN y al PCF el establecimiento del PDP context secundario. Para finalizar, el SGSN envía el correspondiente mensaje SM al UE, de forma que éste conozca el fin de la fase de establecimiento de llamada.

3.3 Utilización del Servicio

Una vez que la aplicación del UE tiene el servicio portador IP apropiado para VoIP, el tráfico de datos de usuario y de control de datos pueden ser dirigido hacia el host externo. Durante la utilización del servicio, la calidad de servicio negociada se mantiene gracias a varios mecanismos pertenecientes a diferentes capas y entidades en el plano de usuario: mapeado, clasificación, condicionamiento del tráfico y gestión de los recursos [3]. Existen diversos protocolos del IETF relacionados con la calidad de servicio en el plano de usuario (*Diffserv* y MPLS), tal y como se describe en más detalle en la sección 5. No obstante, cuando el plano de usuario no es capaz de preservar la calidad de servicio negociada, puede dispararse un procedimiento de modificación del *PDP Context* secundario iniciado por diferentes entidades de UMTS: UE, UTRAN, SGSN y GGSN. Este procedimiento en el plano de control permite renegociar el antiguo perfil de calidad de servicio para el *PDP Context* secundario, siempre que el UE acepte la calidad ofrecida por la red en dicha renegociación [6].

3.4 Liberación de la Llamada

En la Fig. 4 se muestra un diagrama de flujo con una propuesta para el procedimiento de liberación de la llamada.

Paso 1 - La aplicación en el UE indica al CSCF su intención de finalizar la sesión SIP. Este mensaje es reenviado al *host* externo.

Paso 2 - El CSCF fuerza en el GGSN la desactivación del *PDP Context* secundario.

Paso 3 - El SGSN se encarga de indicar al UE que el PDP *Context* secundario ha sido desactivado y que los recursos reservados en UTRAN (p.e. el servicio portador radio) van a ser liberados.

Paso 4 - El *host* externo recibe la aceptación de la finalización de la sesión mediante un mensaje

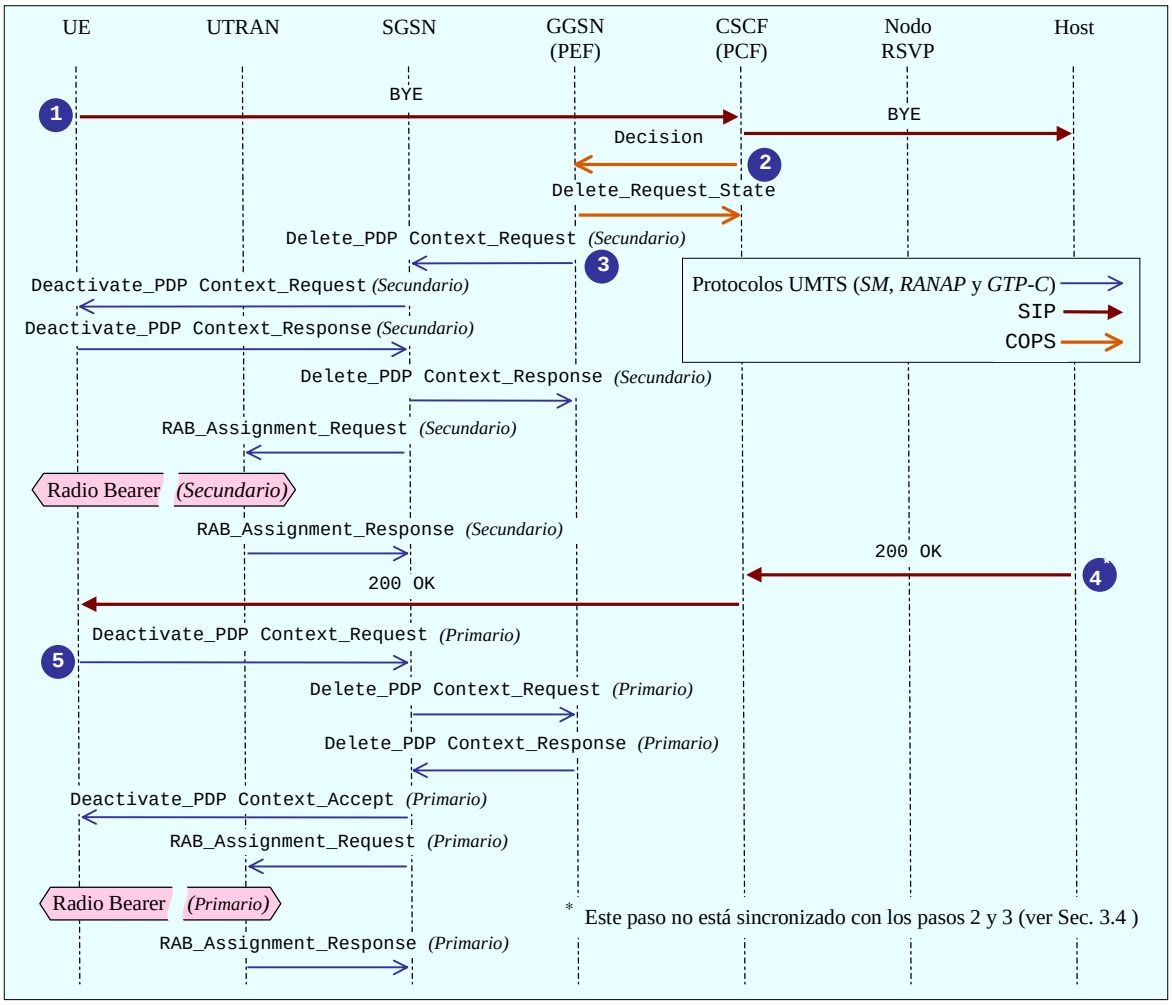


Figura 4: Diagrama de flujo para la liberación de llamada.

`200 OK`. Este paso no está necesariamente sincronizado con los pasos 2 y 3, ya que dicho mensaje puede ser recibido con anterioridad a los pasos 2 y 3 en el nivel de aplicación del UE.

`Paso 5 -` El UE dispara el procedimiento de desactivación del *PDP Context* primario. Aunque este paso es consecutivo al paso 4, también se podría superponer en el tiempo al paso 3, cuando el mensaje `200 OK`, así como el mensaje `Deactivate_PDP Context_Request` son recibidos por el UE durante el paso 3. Realmente, es en este punto del desarrollo del paso 3 cuando el UE sabe que el *PDP Context* secundario se está desactivando, por lo tanto ya no se necesita más el primer *PDP Context* para señalización SIP.

Otra posible solución alternativa para la liberación de la llamada, podría estar basada en cierta señalización COPS añadida entre el CSCF y el GGSN cuando el CSCF recibe el mensaje `200 OK` desde el *host* externo. En este punto, el GGSN podría disparar la desactivación del primer *PDP Context* en lugar del UE. Esta alternativa permitiría mayor control por parte de la red de todo el proceso de liberación. El principal inconveniente de esta solución es que introduce mayor retardo y redundancia de señalización en el plano de control.

4 Interacción de RSVP con UMTS

Como se comentó en la sección 3.2, la interacción entre los protocolos UMTS y RSVP llevada a cabo en el GGSN no es inmediata, ya que no existe una correspondencia directa entre los atributos de calidad de servicio soportados por ambos.

El primer paso para la utilización del protocolo RSVP es la creación del camino que seguirá el flujo de datos con los requisitos de calidad de servicio necesarios. El mensaje RSVP que realiza este cometido es el mensaje `PATH`. Cada emisor transmite mensajes `PATH` en el sentido del flujo de datos por las rutas proporcionadas por el protocolo de encaminamiento. En el caso de un servicio bidireccional, como el de VoIP para telefonía, se requiere el envío de estos mensajes en ambos sentidos. El mensaje `PATH` contendrá información para definir el tipo de tráfico que el emisor espera generar. Esto servirá a los receptores a la hora de calcular las necesidades de calidad de servicio, de forma que dicho tráfico se reciba correctamente.

En este primer paso surge el problema de la correspondencia entre el perfil de calidad de servicio que la aplicación en el UE ha solicitado, el cual viaja en el *PDP Context* secundario hacia el GGSN, y los atributos de calidad de servicio soportados por el protocolo RSVP.

Es importante destacar la carencia de ciertos atributos de calidad de servicio en RSVP, como son el retardo de transferencia u otros atributos referentes a errores y prioridades entre flujos con respecto a los atributos UMTS. En la Tabla 1 se muestra una posible solución de compromiso para establecer una correspondencia entre ambos tipos de atributos.

Tabla 1: Correspondencia entre atributos UMTS y RSVP.

Atributos UMTS	**Atributos RSVP**
Traffic Class	*Service Class*
Maximum Bitrate	*Peak Data Rate*
Guaranteed Bitrate	*Token Bucket Rate*
Maximum SDU Size	*Maximum Packet Size*
Delivery Order	-
SDU Format Information	-
SDU Error Ratio	-
Residual Bit Error Ratio	-
Delivery of Erroneous SDUs	-
Transfer Delay	-
Traffic Handling Priority	-
Allocation/Retention Priority	-
-	*Token Bucket Size*
-	*Rspec (Rate/Slack Term)*
-	*Minimum Policed Unit*

Una vez establecido el camino, cuando un nodo RSVP que recibe un datagrama IP perteneciente a una de las reservas que tiene instaladas, debe enviar el paquete de datos siguiendo el camino en el que se ha instalado la reserva. El estado de reserva en cada nodo del camino debe ser refrescado periódicamente, dado que sólo son válidos durante un cierto periodo de tiempo, por lo que se dice que los estados RSVP en los nodos son blandos (*soft states*) en el sentido de que deben refrescarse continuamente.

Esta característica de RSVP implica un exceso de señalización, que unido a la señalización previa necesaria para el establecimiento del camino, resulta un inconveniente para su uso dentro de la red UMTS. Este continuo envío de mensajes entre *routers* puede dar lugar a cuellos de botellas en la red de transporte. Además, un nuevo camino tendría que establecerse para cada *handover* que ocurriera en la red. Esto conllevaría un retardo inaceptable para servicios VoIP como el que se analiza en este trabajo.

Por este motivo, otros mecanismos de calidad de servicio en plano de usuario serían más adecuados dentro de UMTS, como son *Differentiated Services* (*DiffServ*) [10] o *MultiProtocol Label Switching* (MPLS) [11], ambos del IETF.

5 Utilización de *DiffServ* y MPLS

Una vez establecido el *PDP Context* secundario, la red dispone de los recursos necesarios para comenzar el servicio VoIP. Sin embargo, se hace necesaria la utilización de algún mecanismo añadido en el plano de usuario encargado de diferenciar flujos con diferentes requisitos de calidad de servicio y tratarlos de forma diferente, ya sea dándoles diferente prioridad y/o enviándolos por caminos distintos. Esta es la filosofía de funcionamiento de los protocolos *DiffServ* y MPLS, respectivamente.

DiffServ se basa en la utilización de un campo de 8 bits situado en la cabecera de los paquetes IP denominado *DiffServ CodePoint* (DSCP). Todos los paquetes marcados con el mismo DSCP deben sufrir el mismo tratamiento por los *routers* dentro de la red, es decir, tienen la misma prioridad para ser reenviados al siguiente *router*. Esta característica se denomina "comportamiento por salto" (*Per-Hop-Behavior*, PHB). Con *DiffServ*, los *routers* no necesitan almacenar ningún tipo de información acerca de los flujos, como ocurre en RSVP.

Mediante este mecanismo, una vez comenzado el envío de los paquetes de voz, éstos deberían ser marcados con alta prioridad de manera que el retardo experimentado por dichos paquetes en cada *router* sea inferior al de otros paquetes con requisitos de calidad de servicio más relajados.

Con MPLS, el reenvío de los paquetes IP se simplifica mediante el uso de etiquetas de pequeño tamaño que identifican a los diferentes flujos. De esta forma no es necesario procesar la extensa cabecera IP para el reenvío de los paquetes.

A diferencia de *DiffServ*, en el cual se marcan los paquetes para determinar la prioridad de ese paquete frente a otro dentro de un *router*, con MPLS se marcan los paquetes para determinar el salto al siguiente *router*, de forma que todos los paquetes con la misma etiqueta deben seguir el mismo camino (*Label Switching Path*, LSP). El reenvío de los paquetes se basa en una conmutación de la etiqueta en cada salto (al igual que ocurre en ATM).

Ambos mecanismos de calidad de servicio podrían ser aplicados simultáneamente en UTRAN, ya que son mecanismos independientes que funcionan en niveles diferentes de la capa de protocolos. La utilización combinada de ambos protocolos con RSVP dentro de UTRAN sería poco recomendable según lo expuesto en la sección 4.1.

En la red externa, todos ellos (RSVP, *Diffserv* y MPLS) se podrían utilizar conjuntamente, dependiendo de la complejidad de la IP-PDN en cuestión.

6 Conclusiones

En el presente trabajo, se describe en primer lugar la arquitectura de calidad de servicio extremo a extremo para servicios VoIP cuando UMTS se interconecta con una red externa IP-PDN.

En segundo lugar, se presenta una propuesta para la coordinación entre los mecanismos y protocolos IETF y UMTS en una llamada VoIP. En tal propuesta se han considerado protocolos IETF relacionados con la calidad de servicio en el plano de control y para el control de la llamada, a saber: RSVP, COPS y SIP. Mediante diagramas de flujo de señalización, se muestra que es posible una coordinación apropiada para cada fase de la llamada VoIP.

Finalmente, se ha analizado la posibilidad de emplear en el escenario descrito, otros mecanismos del IETF relacionados con la calidad de servicio en el plano de usuario durante la fase de utilización del servicio (*Diffserv* y MPLS).

Agradecimientos

Este trabajo está subvencionado por Nokia Networks. Los autores quieren agradecer a Z. C. Honkasalo y J. M. Melero su apoyo e interesantes comentarios.

Referencias

[1] H. C. H. Rao, Y. B. Lin, and S. L. Cho, “iGSM: VoIP service for mobile networks”, IEEE Comm. Mag., vol. 4, pp. 62-69, Apr. 2000.

[2] 3rd Generation Patnership Project; Technical Specification Group Services and Systems Aspects, “Architecture for an all IP network”, TR 23.922 v1.0.0, 1999.
http://www.3gpp.org/ftp/Specs/2000-06/R1999/23_series/23922-100.zip

[3] 3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects, “QoS concept and architecture”, TS 23.107 v4.0.0, 2000.
http://www.3gpp.org/ftp/Specs/2000-12/Rel-4/23_series/23107-400.zip

[4] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, “Resource ReSerVation Protocol (RSVP) version 1 functional specification”, IETF RFC 2205, Sept.1997.
http://www.ietf.org/rfc/rfc2205.txt

[5] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, “SIP: Session Initiation Protocol”, IETF RFC 2543, March 1999.
http://www.ietf.org/rfc/rfc2543.txt

[6] 3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects, “General Packet Radio Service (GPRS); service description; stage 2”, TS 23.060 v3.5, 2000.
http://www.3gpp.org/ftp/Specs/2000-09/R1999/23_series/23060-350.zip

[7] 3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects, “Architectural principles for release 2000”, TR 23.821 v1.0.1, 2000.
http://www.3gpp.org/ftp/Specs/2000-06/R2000/23_series/23821-101.zip

[8] R. Braden, et al., “The COPS (Common Open Policy Service) protocol”, IETF RFC 2748, Jan. 2000.
http://www.ietf.org/rfc/rfc2748.txt

[9] Alan B. Johnston, “SIP. Understanding the Session Initiation Protocol”, Artech House, 2001.

[10] S. Blake, D. Black, M. Carlson, Z. Wang and W. Weiss, “Architecture for Differentiated Services”, IETF RFC 2475, Dec. 1998.
http://www.ietf.org/rfc/rfc2475.txt

[11] A. Viswanathan, and R. Callon, "Multi-protocol Label Switching Architecture", IETF RFC 3031, Jan. 2001.
http://www.ietf.org/rfc/rfc3031.txt

Personalización de servicios WAP mediante gestión de contenidos XML y bases de datos relacionales[1]

Albert Vallespí i Capdevila, Miquel Oliver i Riera
Departament d'Enginyeria Telemàtica. Universitat Politècnica de Catalunya.
Jordi Girona 1 y 3. Campus Nord, Mód C3, UPC. 08034 Barcelona
Teléfono: 934 0168 30 Fax: 934 015 981

Abstract: A WAP application development, based on content and terminal personalization, using XML and relational data bases is presented in this article. The application has been focused as an extension of an universitary intranet portal, combining several related services such as comunitary and personal agendas, subjects info, academic news, library services and job posting. The aim of the application has been to explore the XML possibilities to structure a complex data base, personalizing the data shown depending on the type of access (fixed or mobile) and past choices from each user depending on their preferences and use.

1 Presentación del Trabajo

1.1 Introducción

En este documento se describe la especificación técnica relativa al desarrollo de una aplicación WAP basada en la extensión de una intranet para un entorno Universitario que se ha denominado **Campus Móvil Universitario de Cataluña**. Para ello se ha desarrollado un portal de servicios orientado a la comunidad universitaria con acceso móvil.

En este trabajo se va a estudiar la personalización de los servicios que puede ofrecer un portal de contenidos **WAP** (Wireless Application Protocol) a distintos usuarios. Se va a utilizar el lenguaje de marcas conocido como **XML** (eXtensible Markup Language) para estructurar el contenido de un modo eficaz, a la vez que posibilitar la explotación del portal desde varias plataformas distintas. El Portal permite el acceso desde cualquier dispositivo móvil que adapte el protocolo WAP y desde navegadores Web tradicionales.

Se ha escogido como portal para ofrecer los servicios personalizados mediante acceso WAP un **portal universitario**. Este portal estará centrado en las necesidades de un entorno universitario, y se ofrecerán servicios a los miembros de esta comunidad. Los servicios desarrollados intentan facilitar todo tipo de información de interés universitario a los miembros de esta comunidad. Por otro lado también se van a desarrollar una serie de utilidades que tienen la intención de facilitar ciertas actividades cotidianas en un entorno móvil.

1.2 Motivación

En los últimos años las nuevas tecnologías están emergiendo con gran fuerza, siendo el principal motor económico de los países occidentales. Año tras año, surgen numerosos avances tecnológicos que pretenden facilitar el estilo de vida, de trabajo y las relaciones entre las personas.

El sector empresarial privado es el máximo impulsor de los avances en las nuevas tecnologías. No obstante el mundo universitario, tiene un papel importante dentro de la investigación y del desarrollo de estas nuevas tecnologías. Ya que además de participar en muchos proyectos activamente o como patrocinador, tiene la misión fundamental de formar a los profesionales de la sociedad de la información del futuro.

El presente documento es la especificación técnica de un proyecto que tiene como objetivo diseñar e implementar un portal de servicios **WAP** para la comunidad universitaria. Concretamente se centrará en un conjunto de servicios y utilidades dirigido a los miembros de una escuela técnica superior perteneciente a una universidad politécnica.

Este proyecto está motivado por varios factores que a continuación se van a detallar:

- Increíble crecimiento en Europa de las comunicaciones mediante teléfonos móviles digitales. Actualmente la mayoría de la sociedad, y concretamente un enorme porcentaje de estudiantes universitarios disponen de teléfono móvil, y lo utilizan como instrumento principal para comunicarse con otras personas.
- Importante penetración del uso de Internet entre la sociedad, y en especial entre las personas con una edad comprendida entre los 15 y 30 años. Se puede afirmar que en nuestro país prácticamente la totalidad de miembros de la comunidad

[1] Este proyecto ha sido parcialmente financiado por la CICYT TIC2000-1120-C03-02

universitaria utilizan Internet como principal fuente de información.

- Saturación habitual de la red Internet debido a su gran éxito. Mediante WAP se accede a la misma información ocupando menos ancho de banda.
- Gran demanda de equipos destinados para dar acceso a Internet a los estudiantes dentro de los centros universitarios. Los equipos que existen suelen estar colapsados, obligando a los estudiantes a esperar bastante tiempo, cuando a veces precisan una información muy concreta.
- Dificultad de informar a los miembros de una comunidad tan variada como la universitaria de ciertos aspectos de gran interés. Esta dificultad se debe a que muchas veces los estudiantes no tienen fácil acceso a Internet, se alojan en residencias, pasan mucho tiempo en bibliotecas...
- Promover el uso de las nuevas tecnologías entre la comunidad universitaria, es una pieza fundamental para la formación de los universitarios que probablemente se incorporarán a una empresa e-business. En este tipo de empresas, los dispositivos móviles son una pieza clave en el entorno de trabajo, ya que actualmente existen aplicaciones que facilitan toda la información y recursos de que dispone una empresa a sus trabajadores usando dispositivos móviles como teléfonos WAP, microordenadores y ordenadores portátiles.

Debido a las motivaciones que se explican sobre estas líneas, se considera que el desarrollo de un **portal WAP** es de gran importancia para informar a la comunidad universitaria de diversos aspectos. También es una herramienta correcta para dar acceso a las personas a varias utilidades necesarias y hasta hace poco desconocidas en un entorno móvil.

Un aspecto fundamental es que este servicio estará dirigido a un conjunto de personas jóvenes y dinámicas. Un segmento de personas habituadas a trabajar con Internet y a comunicarse mediante teléfonos móviles. Es previsible que en poco tiempo, un gran número de estudiantes dispondrá de teléfonos WAP con los que tendrán acceso a toda la información que ofrece la red de redes, pero sin tener que esperar. Además existe la gran ventaja que utilizarán un dispositivo propio, por lo tanto podrán acceder a un conjunto de servicios totalmente personalizados.

Otro aspecto importante a favor de esta tecnología, es que mediante ella sólo se accede a la información que se necesita en cada momento. Se transmiten pequeñas cantidades de bytes para cada consulta. Por tanto promover el uso de los servicios WAP puede ser importante para liberar de tráfico innecesario las redes universitarias.

1.3 Objetivos

El objetivo principal de este proyecto es el desarrollo e implementación de un portal de servicios WAP dirigido a los miembros de la universidad. Se trata de un portal que posibilitará la comunicación automática entre los miembros de esta comunidad que podrán acceder a la información desde varias plataformas distintas.

Mediante este portal se podrá acceder a varios servicios desde un entorno móvil, así como recibir avisos y notificaciones de importancia.

Un objetivo principal de este proyecto es la personalización de contenidos para cada uno de los miembros de la universidad. Debido a que cada persona dispone de un terminal propio, se cree adecuado personalizar los contenidos en función de las preferencias del usuario.

El acceso a la información, puede resultar más o menos ameno, en función de su presentación y de la estructura de la pantalla. Un objetivo importante de este proyecto es desarrollar distintas presentaciones del portal adaptándolas a los distintos terminales WAP del mercado y presentar al usuario aquella que coincida con su terminal. Adicionalmente también se ha desarrollado la presentación para acceder al portal desde un entorno web.

Una herramienta importante para el desarrollo de este proyecto es el lenguaje de marcas extensible conocido como **XML**. Mediante este lenguaje se puede separar la presentación de los servicios concretos y los contenidos o información a presentar. De este modo cada usuario podrá acceder a una información o a otra, con más facilidad, dependiendo de sus preferencias, y de su recorrido habitual por el portal.

Otro aspecto importante para escoger XML como lenguaje de marcas para gestionar el contenido del portal, es que XML es un lenguaje estándar, abierto y con gran futuro que permite presentar cualquier contenido a cualquier tipo de dispositivo. Con XML podemos ofrecer un servicio parecido, a un Ordenador, a un teléfono WAP, o a cualquier dispositivo que pueda acceder a Internet, lo único necesario es aplicar la plantilla de presentación adecuada a cada tipo de dispositivo.

Debido a lo mencionado anteriormente, al diseñar el portal de servicios WAP utilizando XML para la gestión de contenidos se está dejando una puerta abierta para que fácilmente se pueda adaptar a cualquier dispositivo que surja en el futuro. Este es un aspecto muy importante, ya que la tecnología avanza a pasos agigantados, y no nos podemos imaginar con que tipo de terminales nos sorprenderá en los próximos años.

2 Tecnologías Utilizadas

En esta sección se describen brevemente los principales protocolos, lenguajes de programación y tecnologías que se han utilizado en el desarrollo del proyecto.

2.1. Java

La tecnología Java impulsada por Sun Microsystems se ha asentado en el núcleo mismo de la informática corporativa, gracias a su axioma: Write Once, Run Anywhere (escribir el código una sola vez y ejecutado en cualquier plataforma).

Como lenguaje de programación, Java es ideal para la creación de intranets y destaca sobre C, C++, Visual Basic, e incluso sobre la suma de todos juntos.

Desde sus inicios como nuevo lenguaje de programación, cogió por sorpresa al mundo Internet y convirtió las páginas Web en interactivas. La plataforma Java (en contraposición al lenguaje en sí) ha tomado forma entre un torbellino de atención por parte de los medios de comunicación y especialistas, y entre un período de avidez que desembocó en escepticismo cuando la plataforma superó sus primeros contratiempos.

Asimismo, la tecnología Java ha entrado también en el terreno del servidor, se puede encontrar en cualquier máquina que disponga de Java Virtual Machines: desde el servidor Web freeware de Apache a los mainframes IBM System 390. Java está haciendo la programación para los servidores, en muchos casos, más fácil que con el típico uso de los exploradores.

Se ha elegido Java como entorno de desarrollo de las aplicaciones por varios motivos. Actualmente Java es el principal entorno de desarrollo para aplicaciones en Internet mediante Servlets o Java Beans. Java tiene una particularidad que lo distingue de los otros lenguajes de 4GL, Java es independiente de la plataforma Hardware/Software donde se desarrolle. Java permite portar una aplicación a cualquier plataforma, independientemente de sistema operativo y marca comercial.

Actualmente se puede desarrollar aplicaciones Java 300%, esto significa que se puede desarrollar con el lenguaje Java en las tres capas de la arquitectura de una aplicación Internet. Esta tecnología permite desarrollar aplicaciones Java para el Servidor de Aplicaciones, aplicaciones Java y Applets para el cliente, y ya existe alguna BBDD relacional, como por ejemplo Oracle8i, que permite desarrollar toda la mecánica interna de procedimientos, triggers y funciones con Java. Por este motivo Java es el lenguaje idóneo para desarrollar en la conocida estructura a tres niveles de una aplicación en Internet.

En la siguiente figura se muestra la función de Java en cada nivel.

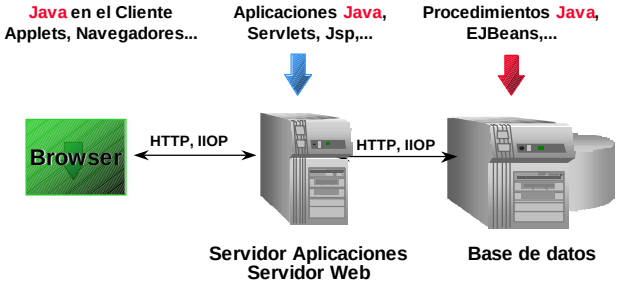


Figura 1: Posibilidades de Java en los tres niveles de Internet

Como es fácil creer, es de gran importancia unificar el entorno de programación en las tres capas, para facilitar la tarea a los programadores.

Adicionalmente se pueden destacar las siguientes características del entorno Java:

- Ofrece productividad real
- Rápido ensamblaje de aplicaciones
- Una Internet, intranet y extranet corporativas unificadas tecnológicamente
- Menor coste y rápida implementación

2.2. Extensible Markup Language

XML (eXtensible Markup Language) no es, como su nombre podría sugerir, un lenguaje de marcado o un lenguaje de marcas. XML es un meta-lenguaje que nos permite definir lenguajes de marcado adecuados a usos determinados. Esto es lo que se ha hecho en este proyecto, crear un lenguaje XML apropiado para los documentos que componen un portal universitario.

En la figura 2 se puede observar la situación actual de los principales lenguajes de marcas. XML forma parte de la familia de SGML (Standard Generalized Markup Language). Un ejemplo de lenguaje creado con XML es WML (Wireless Markup Language). Destaca la situación actual de HTML (Hiper Text Markup Language), en un principio miembro de la familia de SGML, pero actualmente no se ajusta totalmente a sus recomendaciones.

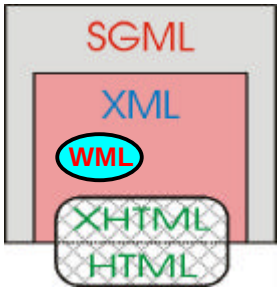


Figura 2: Ubicación de los principales lenguajes de marcas.

Recientemente en el W3C se ha desarrollado XHTML(eXtensible Hiper Text Markup Language), como una aplicación de XML que permite realizar las mismas funciones que HTML, pero siguiendo las restricciones de SGML.

Mediante el lenguaje XML diseñado, únicamente se han estructurado los contenidos del portal. Estos contenidos se han de ajustar a las reglas que se han definido previamente en el DTD (Document Type Definition). Se ha creado el DTD correspondiente.

El siguiente paso es aplicar al documento XML una plantilla de presentación XSL determinada, en función del dispositivo que el usuario está utilizando.

XML permite presentar un mismo documento para distintos dispositivos de un modo sencillo y limpio.

Por otro lado, XML se esta convirtiendo en el lenguaje standard para comunicar la mayoría de aplicaciones de comercio electrónico y B2B. Se está definiendo una estructura de datos estándar basada en XML para que todas las aplicaciones sean capaces de recibir y enviar información entre ellas.

2.3. Arquitectura de Protocolos

En este proyecto se han utilizado dos arquitecturas de protocolos consolidadas y muy bien definidas. En primer lugar la arquitectura de protocolos para aplicaciones inalámbricas, conocido como WAP. Mediante este protocolo se ha posibilitado al acceso de los contenidos desarrollados a dispositivos móviles.

Las aplicaciones desarrolladas están ubicadas en un servidor en Internet. La comunicación entre este servidor y la Pasarela Wap se realiza mediante TCP/IP. En estos casos y cuando el cliente sea un navegador Web conectado a Internet se utiliza la estructura de protocolos HTTP para la comunicación.

2.4. Oracle8i: SQL

Se ha escogido la base de datos Oracle8i para gestionar la información del portal. Se trata de una base de datos relacional, más concretamente objeto-relacional. Está diseñada para ser la base de datos del Internet Computing, y de hecho es la más utilizada entre las nuevas empresas de contenidos en Internet.

Adicionalmente Oracle8i incorpora una máquina virtual Java nativa que permite desarrollar aplicaciones que se ejecuten en la propia base de datos. No obstante esta funcionalidad no se ha utilizado en este proyecto. Se ha creído oportuno abrir la aplicación a cualquier otra base de datos relacional, y por tanto no se han desarrollado todas las posibilidades de Oracle8i. Para comunicarse con la base de datos se ha utilizado drivers estándares JDBC.

Se ha utilizado exclusivamente el lenguaje estándar SQL (Structured Query Language) para entrar y recuperar datos de la base de datos. Debido a la compleja estructura relacional de la base de datos diseñada, en algunos casos ha sido necesario elaborar sentencias SQL compuestas y complejas. No obstante las mismas sentencias se pueden utilizar para trabajar contra cualquier base de datos relacional.

2.5. HTML, WML

Los navegadores o browsers de Internet son unos agentes de usuario que interpretan los lenguajes de marcas para presentar la información de los proveedores de contenidos. El lenguaje de marcas más utilizado en la WWW es HTML, para presentar los contenidos a navegadores Web. En el caso de los dispositivos inalámbricos, el lenguaje utilizado para esta función es el WML.

En este proyecto se han generado documentos con formato HTML o WML a partir de contenidos dinámicos XML. Adicionalmente se han generado mediante distintas plantillas XSL varias versiones de los documentos WML adaptadas a los móviles más populares.

2.6. JavaScript, WMLScript

Los lenguajes de marcas están diseñados únicamente para presentar documentos en Internet. No permiten realizar ningún tipo de operaciones como los lenguajes de programación que conocemos. Las necesidades actuales de Internet han llevado a la creación de sencillos lenguajes de programación o scripts que aporten más funcionalidad a estos lenguajes de marcas.

Entre estos lenguajes destacan JavaScript, con sintaxis parecida a C o Java, VisualBasicScript, o WMLScript. Los dos primeros son complementos de HTML, y el tercero trabaja conjuntamente con WML.

Para incrementar el dinamismo en las presentaciones, hacer validaciones en la entrada de datos, así como para dar valores a variables, se ha utilizado JavaScript y WMLScript en el portal.

3 Estructura Tecnológica

En esta sección se describen los fundamentos básicos de la estructura tecnológica que se ha seguido para el desarrollo de este proyecto. Principalmente se describirán los tres bloques principales, la estructura lógica del desarrollo, la plataforma de comunicaciones utilizada, y el modelo relacional de datos.

3.1. Estructura Lógica del Proyecto

En la siguiente figura se puede observar esquemáticamente esta estructura.

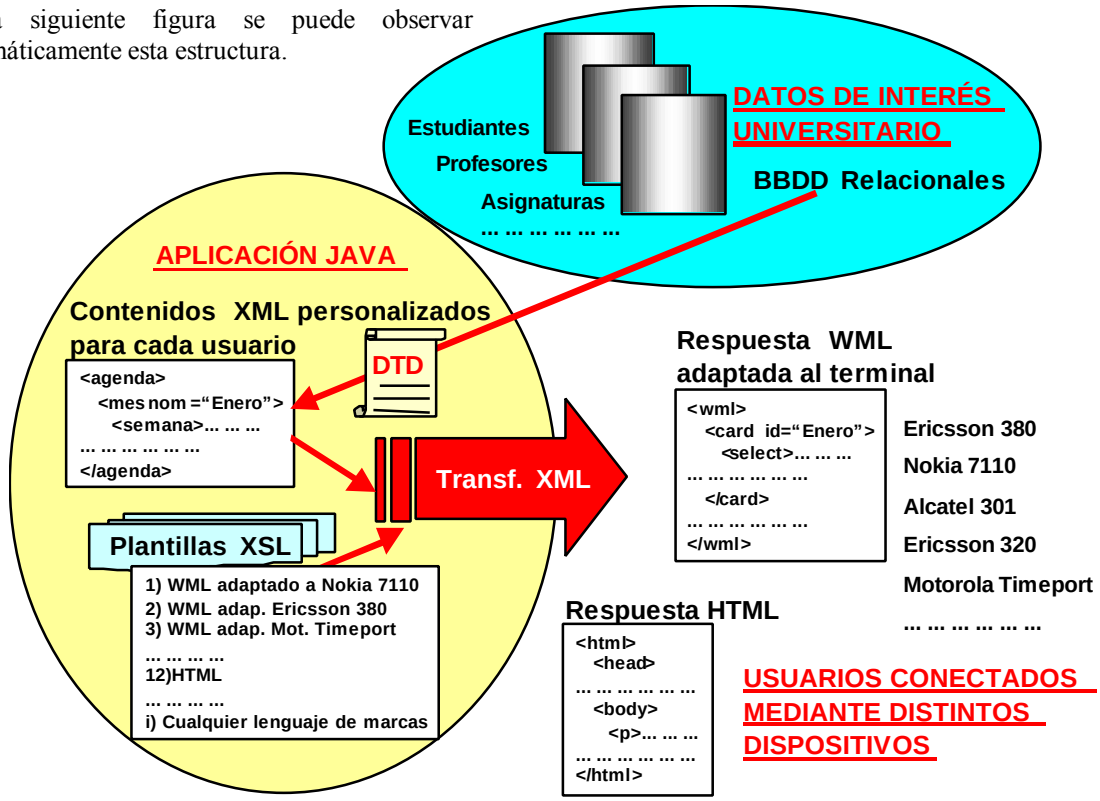


Figura 3: Representación de la lógica de programación del proyecto

Los usuarios realizan peticiones a los Java Servlets mediante URLs (Uniform Resource Locator). Ya sea mediante WAP o HTTP.

Los contenidos del portal se generan en XML mediante una aplicación Java. Esta aplicación es la encargada de identificar al usuario, y generar unos contenidos personalizados accediendo a la base de datos. La aplicación se conecta a la base de datos mediante tecnología de acceso JDBC, y realiza todo tipo de peticiones y modificaciones sobre la información que esta almacena, a medida que los usuarios lo soliciten.

Todas las tareas de interacción con la base de datos se llevan a cabo mediante un objeto único de la clase PM (Persistant Manager). Este objeto garantiza que los datos estarán protegidos por exclusión mutua, dotando de robustez al sistema.

Para cada aplicación dentro del portal se generarán unos contenidos XML totalmente personalizados, en función de la petición y del usuario que la realiza. En función del rol de cada usuario (Profesor, Estudiante, Personal de Administración) se puede acceder o no, y con ciertos privilegios a la información.

Los contenidos XML generados dinámicamente por la aplicación Java son traspasados a un objeto de la clase transWML, que se encarga de transformarlos a documentos WML, HTML u otro formato. Esta función se realiza aplicando una de las distintas plantillas XSL. Mediante estas plantillas adicionalmente se añaden referencias a scripts creados para complementar la robustez de los documentos.

Finalmente la aplicación retorna mediante el protocolo HTTP la respuesta (WML o HTML) a la petición URL que el cliente ha realizado.

3.2. Plataforma de Comunicaciones

El modelo de comunicaciones WAP es bastante similar al de la WWW. La información es transportada mediante protocolos estándares abiertos de comunicación basados en los empleados para la WWW.

El usuario accede a Internet mediante un dispositivo inalámbrico. Este terminal dispone de un micro browser encargado de realizar las funciones de interfaz con el usuario de forma análoga al web browser.

Dada la singularidad de los dispositivos móviles, estos no pueden conectarse directamente a la Internet cableada. Utilizan protocolos de acceso vía radio para acceder a la red de la operadora móvil. Los formatos

en los que serán presentados los contenidos, y los protocolos WAP, han sido optimizados para su empleo en pequeños dispositivos inalámbricos (terminales hand-held).

WAP redefine todos los niveles de la arquitectura TCP/IP adecuándolos al inalámbrico.

Para poder acceder y aprovechar los recursos de Internet (almacenados en los servidores Web o generados dinámicamente por servidores de aplicaciones), es necesario contar con un sistema que actúe como intermediario entre la red inalámbrica y la WWW. Esta función la realiza un servidor proxy que puede estar ubicado en la red GSM o equivalente o en Internet, pero necesariamente debe tener salida a ambas redes. Un servidor *proxy* es un programa que actúa tanto como cliente como servidor, captura las peticiones de sus clientes, realizándolas en su lugar, para devolver los datos al equipo que los solicitó originalmente. El servidor proxy realiza las siguientes tareas:

- **Conversión de protocolo** - La parte encargada de esta misión se denomina WAP Gateway o Pasarela WAP, lo que hace es traducir las peticiones de un dispositivo, hechas implementando la estructura de protocolos WAP, a peticiones HTTP convencionales.
- **Codificación y decodificación de contenidos** - Los codificadores de contenidos comprimen la información WAP en formatos más compactos, para reducir el tamaño de los datos que circulan sobre la red inalámbrica.

Esta infraestructura está desarrollada para permitir a los usuarios de terminales móviles la navegación a través de una abundante variedad de aplicaciones y documentos WAP. La utilización del WAP proxy como intermediario entre las dos redes heterogéneas (inalámbrica e Internet), permite que los recursos y aplicaciones WAP puedan estar alojados en servidores WWW convencionales y que sean desarrolladas empleando las tecnologías habituales de la WWW, a la vez que los usuarios de dispositivos móviles pueden acceder a ellos.

La figura del WAP Gateway, se ha incluido en el modelo WAP para formar una arquitectura de cuatro niveles: Cliente, WAP Gateway, Servidor WAP/Aplicaciones, Base de Datos.

De un modo análogo se define la arquitectura de tres niveles de la WWW: Cliente, Servidor Web/Aplicaciones, Base de Datos.

En el proyecto desarrollado existe la posibilidad de conectarse directamente mediante un browser Web (HTTP), y la posibilidad de conectarse mediante un dispositivo inalámbrico (WAP+Gateway+HTTP)

El modelo de comunicaciones global del proyecto se puede observar en la siguiente figura:

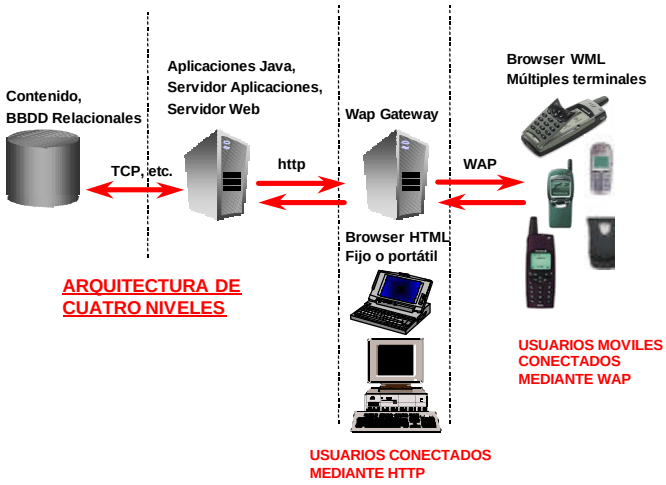


Figura 4: Modelo de comunicaciones del proyecto

3.3. Modelo de datos relacional

Las aplicaciones que se han desarrollado, necesitan estar permanentemente comunicadas a una gran estructura de datos donde almacenar la información del portal. Esta es la función que realiza la base de datos. La base de datos del proyecto ha de estar compuesta por un gran número de tablas.

La alternativa más adecuada es realizar la aplicación utilizando una base de datos real de una escuela universitaria, en la que sólo se tendrían que añadir aquellas tablas específicas del portal que no tuvieran referencia en ella. En las aplicaciones WAP se necesitarán únicamente algunos campos de estas tablas, ya que se presentará sólo la información importante.

Al no disponer de ningún modelo de base de datos real, se ha creado uno específico para esta proyecto. Se ha diseñado una estructura de la base de Datos compleja formada únicamente por las tablas necesarias para este portal. Una representación de este modelo relacional se puede observar en el gráfico adjunto. Es importante señalar las relaciones entre las tablas, y a su vez indicar que las claves primarias de cada tabla aparecen marcadas con negrita y subrayado.

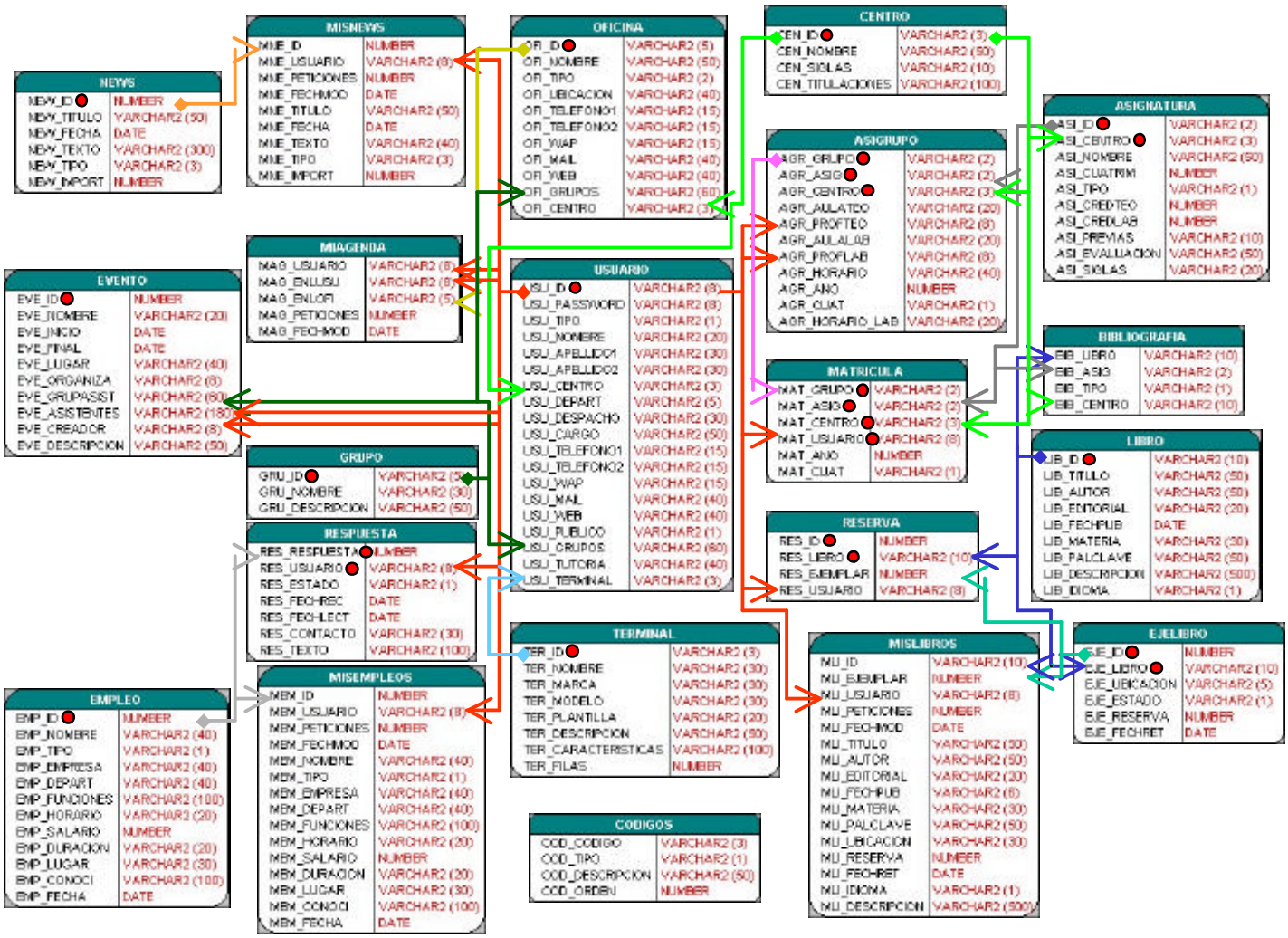


Figura 5.- Modelo de Datos Relacional.

Será necesario utilizar una tabla de usuarios, una tabla de asignaturas, una tabla de libros, una tabla de eventos... Se tratará de una estructura complicada que facilite al máximo la obtención de la información por parte de las aplicaciones.

4 Descripción Funcional del Portal

En esta sección se va a explicar brevemente el funcionamiento del Portal UMC que se ha desarrollado.

Para ello se van a utilizar impresiones de los displays de los terminales móviles durante una sesión de usuario por el portal. No obstante, se cree importante destacar que el mejor modo de saber el funcionamiento del portal es navegando por él, por tanto animamos a todo aquel que lo quiera conocer a que se introduzca en él.

El primer paso para entrar en el portal es identificarse, la aplicación solicita un username y password, y luego comprueba en la base de datos que el usuario sea correcto, en caso de que no sea correcto, no permite introducirse en la aplicación.

Figura 6: Página acceso (Terminal Nokia BluePrint)

Una vez identificado y verificado por el sistema, el usuario accede a la página de bienvenida, donde se le recibe, y en 5 segundos se le da acceso al menú principal del portal.

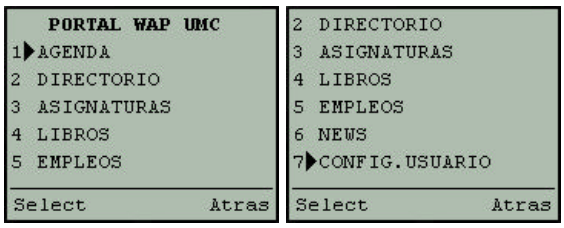


Figura 7: Menú principal del Portal

En esta pantalla se tiene acceso a todos los servicios del portal que están finalizados. Se ha creído oportuno incluir 6 servicios representativos de las necesidades que pueden tener los miembros de una universidad en un entorno móvil. Adicionalmente se ha incluido una entrada para la personalización de la configuración de usuario.

El servicio de la AGENDA, está diseñado para comunicar de un modo fácil y fluido las agendas de actividades de todos los usuarios del Portal. De este modo un usuario puede invitar a otro a una cita que el haya creado, incluyéndolo en la lista de asistentes. El invitado puede confirmar o desestimar su asistencia.

Al entrar en el servicio, se ofrecen las posibilidades de consultar la agenda por fechas, de ver las próximas citas, de buscar agendas por campos e introducir una nueva cita.

Para consultar la agenda por fecha, se ofrecen las posibilidades de consultar los próximos 10 días, consultar el presente mes, el próximo, o bien consultar un día o mes concreto:

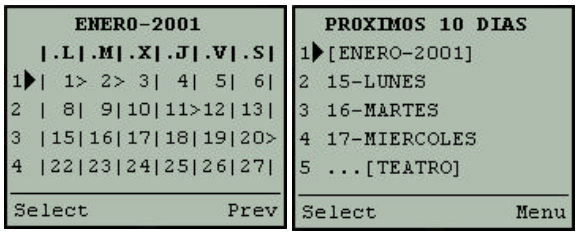


Figura 8: Ejemplo de consulta en la AGENDA

En la consulta de un mes concreto, se puede ver aquellos días en los que existen eventos en los que el usuario está involucrado, ya sea porque es el creador o porque está invitado. En este caso se observa con el símbolo > antes del día.

El servicio de DIRECTORIO permite a los usuarios obtener información y datos de contacto sobre los otros usuarios y/o unidades estructurales de la universidad. No obstante es importante mantener la privacidad de los datos de los usuarios que así lo deseen, y por tanto todo usuario tiene la posibilidad de publicar o no sus datos en el directorio del portal UMC. Este servicio incluye las carpetas de Favoritos e Historial, que más adelante se explican.

Los otros módulos que forman la aplicación son los de ASIGNATURAS, LIBROS, EMPLEOS, y NEWS.

Estos servicios son plenamente personalizados de modo que cuando un usuario entra puede acceder a su horario, a sus asignaturas, a sus libros más consultados, etc. Un aspecto muy importante es la posibilidad de acceder a la sección Favoritos y Historial de cada uno de los módulos.

En la sección Favoritos se almacenan aquellos enlaces que se han creído oportunos guardar en los favoritos privados. Por otro lado en la sección de Historial se pueden recuperar aquellas visitas anteriores al módulo, ya sea a detalles de Asignaturas, libros, empleos o news. Por otro lado también se pueden recuperar los criterios de búsqueda de consultas anteriores. Esto se puede ver en las siguientes capturas de pantallas.

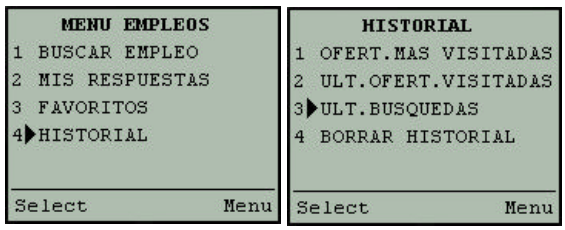


Figura 9: Ejemplo Favotitos y recuperación de búsquedas

Para finalizar la descripción de la aplicación, señalar que existe el menú de CONFIGURACIÓN de usuario. Donde el usuario puede actualizar sus datos personales, hacer público o privado sus datos personales en el portal y cambiar de plantillas de presentación en función del terminal del que disponga en ese momento.

Conclusiones.

Mediante este portal se intentan demostrar las enormes posibilidades de XML para integrar aplicaciones únicas y presentarlas en distintos dispositivos. Un resumen sería, aprovechando el axioma de Java explicado anteriormente: "*Write Once, **Show** Everywhere*".

Referencias.

[1] WAP Forum. http://www.wapforum.com
[2] The XML Industry Portal. http://www.xml.org
[3] Nokia WAP Developer Forum. http://www.forum.nokia.com
[4] Ericsson Developer Zone. http://www.ericsson.com/developerszone/
[5] Oracle Technical Network. http://otn.oracle.com
[6] Phone.com. http://www.phone.com
[7] Java-Sun MicroSystems.http://www.java.sun.com
[8] Jakarta Tomcat Project. http://jakarta.apache.org

Provisión de Servicios IP sobre infraestructuras inalámbricas heterogéneas

Johnny Choque, Luis Muñoz, Marta García, Ramón Agüero
Departamento de Ingeniería de Comunicaciones
ETSII y Telecomunicaciones – Universidad de Cantabria
Avda. Los Castros s/n, 39005 SANTANDER
Teléfono: 942-201497 Fax: 942-201488
E-mail: {jchoque, marta, luis, ramon}@tlmat.unican.es

***Abstract**. This paper has been carried out in the framework of the Wireless Internet NEtworks (WINE) project, which introduces the concept of the WAL. This one is targeted to boost the performance of IP traffic in cases where access to the Internet is provided by a wireless link at the last hop. In particular, in this work the WAL signaling architecture is outlined. Precisely, communication between the WAL modules (i.e., link layer protocol entities) is discussed, as well as communication between peer WAL entities, which in turn, reside on the MT and AP, respectively. Furthermore, an overview of the WAL signaling services is given.*

1 Introducción

La evolución que han sufrido en los últimos años los equipos portátiles, tanto en la reducción de su tamaño y coste como en el aumento de su capacidad, ha hecho que se incremente el número de usuarios de estos equipos y que, por tanto, aumente la necesidad de acceder a las aplicaciones que se ofrecen normalmente en la red fija. Para dar solución inmediata a esta necesidad surgieron las denominadas redes de área local inalámbricas (Wireless Local Area Networks, WLAN), plataformas que implementan soluciones a nivel de enlace y físico pero que ofrecen una calidad de servicio (Quality of Service, QoS) deficiente al tráfico cursado por el medio inalámbrico.

El desarrollo generalizado de Internet sobre las redes fijas se beneficia de su capacidad para brindar a los usuarios aplicaciones multimedia con una calidad cada vez mayor, por lo que llevar dichas aplicaciones al entorno inalámbrico no resulta una tarea sencilla debido a las diferencias que existen entre las redes inalámbricas y fijas. Uno de los principales problemas en este sentido es que los canales inalámbricos tienen un comportamiento impredecible y son propensos a altas tasas de error, debido a las interferencias, los desvanecimientos y la movilidad del terminal, condiciones que no se dan en las redes fijas. En definitiva existen una serie de limitaciones en la naturaleza inherente a la pila de protocolos de Internet que impiden brindar la misma QoS a los usuarios de plataformas inalámbricas, ya que protocolos como TCP (Transport Control Protocol) han sido diseñados teniendo en cuenta las características de las redes fijas. La pérdida de paquetes en el canal radio es interpretada por TCP como pérdidas causadas por congestión y, por lo tanto, activa diversos mecanismos para evitarla, reduciendo la cantidad de datos que puede enviar el transmisor. La consecuencia inmediata es una dramática reducción del caudal efectivo de TCP.

Aunque existen numerosos estudios y propuestas relativos a las comunicaciones TCP en entornos inalámbricos [1], éstas en su mayoría aportan soluciones parciales. El presente artículo describe el trabajo realizado en el marco del proyecto Wireless Internet NEtwork (WINE, proyecto europeo IST-1999-10028). El objetivo de WINE es especificar, diseñar, simular y desarrollar una plataforma que soporte la pila de protocolos TCP/IP sobre infraestructuras inalámbricas heterogéneas. El núcleo del proyecto se basa en el desarrollo de una capa de adaptación al medio radio, denominada Wireless Adaptation Layer (WAL), la cual está ubicada justamente por encima de las capas propias de la infraestructura de comunicaciones y debajo de la capa IP. Como su nombre indica, la WAL adapta su comportamiento en función de los requerimientos de las capas superiores y de las condiciones actuales del canal inalámbrico, proporcionando una interfaz única a IP, independiente de la tecnología de red inalámbrica subyacente.

2 La arquitectura WINE

Como se muestra en la figura 1, la arquitectura WINE gira entorno al núcleo constituido por la capa WAL. Esta capa está formada por una serie de módulos que implementan diversas técnicas que intentan compensar la deficiente calidad de los enlaces inalámbricos, con el fin de hacer visible a las capas superiores un medio de transporte eficiente y fiable. Igualmente, la WAL incorpora técnicas de gestión de la calidad de servicio para optimizar las comunicaciones soportadas por la pila TCP/IP sobre enlaces inalámbricos. Finalmente, cabe remarcar que mediante la inclusión, en la capa

WAL, de un módulo dependiente de la infraestructura ésta es flexible hasta tal punto que permite ser soportada sobre infraestructuras tan variadas como Bluetooth [2], IEEE802.11 [3] e HIPERLAN/2 [4].

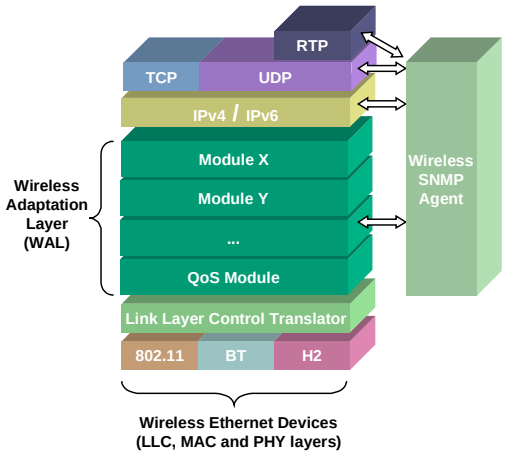


figura 1: Arquitectura WINE

En definitiva la capa WAL se caracteriza por:

Adaptación a las condiciones del enlace radio. Dado el comportamiento ofrecido por el medio radio a las capas superiores, la WAL negocia en cada instante los parámetros óptimos del conjunto de módulos definido para cada conexión durante el proceso de establecimiento de la comunicación entre el punto de acceso (Access Point, AP) y el terminal móvil (Mobile Terminal, MT).

Consideraciones de QoS para tráfico IP. En la parte inferior de la capa WAL se incorpora un módulo de QoS que trata de mapear el esquema DiffServ [5] de la parte fija a la parte inalámbrica.

2.1 Operación de la WAL

El protocolo de señalización diseñado basa su funcionamiento en dos conceptos novedosos, a saber, *clase* y *asociación*.

Una *clase* se define como el servicio a ofrecer a un determinado tipo de tráfico (transferencia de archivos, web, etc.), estando constituida por la concatenación de módulos de la WAL que proporcionan tal servicio. La clase se determina por el campo Time of Service (ToS) en IPv4 o el campo Traffic Class en IPv6, y el tipo de protocolo de la cabecera del paquete IP.

Una *asociación* identifica un flujo de datagramas IP que, perteneciendo a la misma clase, están destinados a un terminal móvil específico. Por lo tanto, una asociación queda unívocamente identificada mediante el par <*Clase_WAL*, *IP_Addr_MT*>.

La razón fundamental de introducir el concepto de clase y asociación se debe a que la adaptación de la WAL a los cambios del medio radio no es necesario hacerla por cada una de las conexiones que puedan existir entre el AP y el MT sino para el conjunto de clases que se estén soportando entre AP y MT.

La figura 2 muestra el orden que siguen los datagramas en la WAL así como los módulos constitutivos de ésta. La inteligencia de la WAL reside en el WAL Coordinator, siendo éste el módulo por el que pasan todos los datagramas, ya sea en "bajada" o "subida", antes de visitar los módulos que le correspondan en base a la clase a la que pertenecen. La información relativa a los módulos que cada datagrama debe visitar, el WAL Coordinator los extrae de la información de la cabecera de la trama WAL.

Cabe así mismo resaltar que la WAL incluye un módulo de "gestión de red inalámbrica", W-SNMP, mediante el cual se pueden fijar los distintos umbrales en base a los cuales dicha capa decide en cada momento que parámetros, correspondientes a cada módulo, es preciso emplear para cada clase.

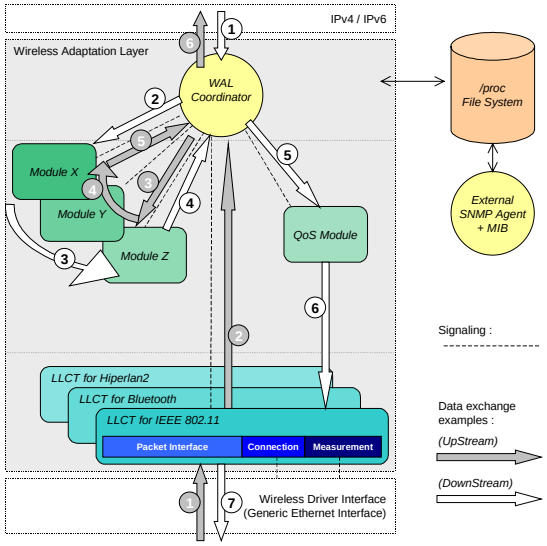


figura 2: Flujo de la información dentro de la WAL

El WAL Coordinator mapea las clases de servicio DiffServ sobre clases WAL, para proporcionar un flujo de paquetes separado de acuerdo a la calidad de servicio que se desea brindar a los diferentes tipos de tráfico. La entidad que regula el flujo de salida de los paquetes es el módulo QoS.

Dicho módulo básicamente lleva a cabo dos funciones. Por un lado intenta maximizar el throughput de cada MT y, por otra parte, trata de asignar de forma equitativa el ancho de banda en base a las condiciones del canal de cada MT.

Los módulos X/Y/Z son entidades del tipo corrección de errores, Automatic Repeat reQuest (ARQ), compresión de cabeceras, SNOOP [6], y otros cuya misión es combatir las restricciones que impone el medio radio como soporte a las tradicionales comunicaciones basadas en la pila TCP/IP.

Finalmente, y a modo de interfaz con los drivers inalámbricos de las posibles plataformas, se ha introducido el módulo LLCT (Logical Link Control Translator). Las principales funciones de este módulo son: gestionar el estado de la conexión con el driver inalámbrico, asegurar la conversión del flujo de datos hacia el driver inalámbrico, realizar medidas del canal a través del driver, y controlar los procesos de registro y deregistro de los MT.

2.2 Señalización de la WAL

Comunicación entre módulos WAL

Como se ha comentado, cada vez que un datagrama IP es interceptado por la WAL, es clasificado por el WAL Coordinator. La clase a la que pertenece el datagrama IP identifica la secuencia de módulos que dicho datagrama ha de atravesar. Para ello, en primer lugar, el WAL Coordinator busca en una tabla la clase a la que pertenece el datagrama IP y así determinar la secuencia de módulos que tiene que visitar, enviando seguidamente el datagrama IP al primero de los mismos. Dicho módulo procesa el datagrama IP, le añade su respectiva cabecera y lo envía al siguiente módulo a través del WAL Coordinator, y así sucesivamente hasta que se alcance el último módulo. Finalmente, el WAL Coordinator añade la cabecera WAL de dos bytes formando, de este modo, la unidad de datos de protocolo WAL (WAL PDU), tal como se muestra en la figura 3. Finalmente dicha PDU se envía al módulo QoS, el cual es responsable de pasarla, a través del LLCT, al driver respectivo de la plataforma inalámbrica subyacente.

Comunicación entre entidades WAL

Las entidades WAL son aquellas que se encuentran tanto en el AP como en el MT. La comunicación entre dichas entidades se realiza a través de la plataforma inalámbrica subyacente. Antes de que un AP con capacidad WAL se comunique con un MT para enviar cualquier información de control o datos, necesita saber si dicho MT tiene "capacidad WAL" y, en su caso, el conjunto de módulos que soporta. Para ello se ha implementado un proceso de registro en el que el MT informa acerca de su capacidad WAL así como el conjunto de módulos que soporta.

Señalización de los Servicios de la WAL

El servicio de transporte implementado por la WAL se define como "orientado a la asociación" en el sentido que se realiza un proceso de negociación para permitir una transferencia fiable y con una calidad adecuada de los paquetes que pertenezcan a una misma clase y que estén destinados al mismo MT. Cada vez que un datagrama IP, que no pertenezca a ninguna asociación, alcanza la WAL, se inicia un proceso de negociación y se establece una nueva asociación para enviarlo. Los subsiguientes datagramas IP que pertenezcan a dicha asociación son procesados por los mismos módulos con los parámetros negociados. Este esquema incurre en mucho menos overhead que el clásico esquema "orientado a la conexión", particularmente si están presentes múltiples flujos IP, como los establecidos por una aplicación web que inicia varias conexiones TCP.

Los dos bytes de la cabecera WAL están constituidos por cuatro campos: (1) la versión de la WAL, (2) el tipo de PDU (6 bits, permitiendo 64 tipos de WAL PDUs), (3) un bit que permite distinguir entre PDU de datos y PDU de señalización, (4) el identificador de la asociación para las PDUs de datos. En la figura 4 se muestra el formato de la cabecera de la WAL PDU.

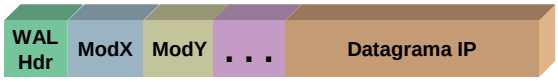


figura 3: Formato del WAL PDU de datos

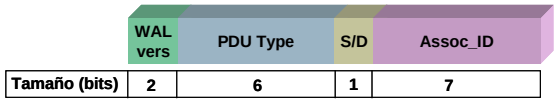


figura 4: Formato de la cabecera WAL

Proceso de registro

Como ya se ha comentado, previamente a la comunicación entre entidades WAL y para determinar si un MT tiene capacidad WAL, se realiza un proceso de registro a nivel MAC (Medium Access Control), gobernado por el módulo LLCT, ya que este es un proceso dependiente de la plataforma inalámbrica subyacente. Una vez realizado el registro a nivel MAC, el módulo LLCT avisa al WAL Coordinator que se ha registrado un nuevo MT, y así se inicia el proceso de registro a nivel WAL, enviando al MT la primitiva WAL_CAPABILITY_REQUEST. Si el MT responde a dicha primitiva, lo hace mediante la primitiva WAL_CAPABILITY_CONFIRM, que contiene la lista de módulos que soporta. En caso contrario, si el MT no responde y tras la expiración de un temporizador, el WAL Coordinator vuelve a enviar la misma primitiva, considerando que el MT no tiene capacidad WAL si vuelve a fallar en el reconocimiento.

En el AP, cuando el WAL Coordinator recibe un WAL_CAPABILITY_CONFIRM de un MT, aquél transmite un SIGNALLING_ASSOC_REQUEST a dicho MT, en el que informa acerca de los parámetros de los módulos que deberán ser utilizados para proteger los mensajes de señalización. El MT acepta la configuración propuesta ya que el AP conoce las clases que puede implementar el MT, por lo que el MT siempre responderá con un SIGNALLING_ASSOC_CONFIRM.

Establecimiento de la asociación

Este proceso es necesario cada vez que llega un datagrama IP que no pertenezca a ninguna asociación existente. Suponiendo que el inicio de este proceso lo realiza el MT, éste enviará la primitiva ASSOC_REQUEST indicando la clase a la que pertenece el datagrama IP y los parámetros de cada módulo, que se determinan en función de las condiciones del canal. Cuando el AP reciba dicha primitiva generará la primitiva ASSOC_RESPONSE, indicando los parámetros definitivos de los módulos, los cuales pueden ser diferentes a los especificados por el MT. El MT tiene dos opciones: aceptar la asociación o rechazarla. Si la asociación es rechazada el MT envía al AP la primitiva ASSOC_REJECT, para que, posteriormente, la entidad WAL que inició el proceso trate de establecer otra asociación transcurrido un determinado tiempo. Si por el contrario el MT acepta la asociación entonces envía la primitiva ASSOC_CONFIRM.

Tanto en el AP como en el MT se registran en tablas las asociaciones que están actualmente en servicio. Dichas tablas están formadas por el identificador de la asociación, el identificador de la clase (el cual define el orden de los módulos) y los parámetros a ser usados en cada módulo. Dentro del AP, también es necesario guardar en la tabla el identificador del MT.

Proceso de Reasociación

Como se ha comentado, una de las características principales de la arquitectura WINE es su capacidad de adaptar su comportamiento en base a las condiciones del canal. Para ello es necesario implementar un módulo de monitorización del enlace, cuya tarea básica es realizar medidas del canal y generar eventos cuando el estado del enlace traspase alguno de los umbrales predefinidos. Esos eventos serán enviados al WAL Coordinator el cual iniciará un proceso de reasociación, similar al de establecimiento de la asociación descrito anteriormente, y en el que se cambiarán los parámetros de funcionamiento de los módulos de la clase correspondiente.

Para la reasociación se asigna un nuevo identificador de asociación, de tal manera que los paquetes que actualmente se encuentran en las entidades WAL terminen de ser procesados usando los parámetros antiguos. Las primitivas usadas para este proceso son bastante similares a aquellas que fueron descritas para el proceso de establecimiento de la asociación.

Eliminación de las asociaciones obsoletas

Debido a que las asociaciones se establecen para un determinado tráfico IP éstas son eliminadas después que finaliza la comunicación entre el AP y el MT. El AP es responsable de eliminar las asociaciones obsoletas (excepto las de señalización) cuando la asociación no procesa tráfico durante un período de tiempo definido. En este caso el AP envía la primitiva END_ASSOC_REQUEST al MT. En el lado del MT, al recibir dicha primitiva se elimina la asociación, respondiendo con la primitiva END_ASSOC_CONFIRM, la cual al ser recibida por el AP también causa la eliminación de la asociación de ese lado de la comunicación.

Proceso de Finalización

Para cubrir algunas situaciones especiales es necesario definir un proceso de finalización de la comunicación entre entidades WAL (caso de un apagado súbito del MT o de salida del rango de cobertura del AP). El AP tiene que tener conocimiento de esta nueva situación para no mantener los mensajes que van dirigidos hacia dicho MT. Para ello, el mismo módulo LLCT que realizó el registro a nivel MAC debe de encargarse de detectar que un MT no está al “alcance” del AP. La implementación de ambas tareas es dependiente de la plataforma inalámbrica subyacente.

3 Movilidad en WINE

En la actualidad la elección de un protocolo único que proporcione soporte a la movilidad en el marco de la Internet difícilmente representaría una solución optima. De hecho, los mensajes de señalización, asociados a los MT, propagados sobre grandes distancias podrían redundar en bajos niveles de rendimiento debido a que pueden verse afectados por grandes latencias o por posibles congestiones. A ello hay que añadir la carga adicional que representan los mensajes de señalización por sí mismos, y que podría contribuir a congestionar el backbone de la Internet en un posible escenario futuro, constituido por un gran número de nodos móviles moviéndose rápidamente. Estas y otras consideraciones sugieren separar la movilidad al menos en dos niveles: (1) Movilidad entre dominios (inter-domain mobility), que puede ser gestionada con Mobile IP versión 4 [7] o versión 6 [8]. (2) Movilidad intradominios (intra-domain mobility), en la que diferentes protocolos alternativos podrían coexistir, manteniendo la transparencia respecto a Mobile IP.

Actualmente el proyecto WINE está abordando el estudio de dos protocolos de movilidad intradominio. El primero de ellos es el denominado Cellular IP [9] el cual inicialmente estuvo diseñado para redes IPv4, aunque también se trabaja en busca de una solución optimizada para redes IPv6. Como segunda alternativa se está estudiando un protocolo de movilidad de nivel de enlace denominado SIMPLE (Scalable Intra-domain Mobility Protocol using Local Encapsulation) [10].

4 Conclusiones

En este artículo se ha presentado el progreso del trabajo realizado por el Grupo de Ingeniería Telemática de la Universidad de Cantabria en el proyecto WINE. WINE tiene como objetivo optimizar la transmisión del tráfico IP sobre enlaces inalámbricos, para lo que se ha adoptado un esquema de Performance Enhancing Proxy (PEP) de nivel de enlace. El núcleo de dicho PEP lo constituye la WAL, con el conjunto de servicios y parámetros que en ella se implementan. Hasta el momento los módulos que se han incorporado son SNOOP, FEC y LLCT, habiéndose derivado resultados muy esperanzadores en cuanto a la mejora de la pila TCP/IP sobre infraestructuras inalámbricas como las que se han mencionado en el artículo.

Agradecimientos

Este trabajo ha sido realizado en el marco del proyecto IST-1999-10028 “Wireless Internet Networks” (WINE), financiado por la Unión Europea dentro del contexto del programa IST (Information Society Technologies). Los autores agradecen las contribuciones de los integrantes del consorcio: VTT Electronics (Finlandia), Philips Research Monza (Italia), Universidad de Roma “La Sapienza” (Italia), AQL (Francia), Cefriel (Italia), Intracom S.A. (Grecia), Universidad de Atenas (Grecia), Acorde (España), Universidad de Cantabria (España) y la Universidad de Queen de Belfast (Irlanda).

Referencias

[1] H. Balakrishnan et. al., “A Comparison of Mechanisms for Improving TCP Performance over Wireless Links,” Proceedings of SIGCOMM 1999.

[2] Bluetooth consortium, “Specification of the Bluetooth system”, v1.0B, Dec. 1999.

[3] “IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”, P802.11, Nov. 1997.

[4] ETSI TR 101 683, “Broadband Radio Access Networks (BRAN); HIPERLAN Type 2, System Overview”, February 2000.

[5] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, “An Architecture for Differentiated Services”, RFC 2475, December 1998.

[6] H. Balakrishnan, S. Seshan, R. Kati, “Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks”, ACM Wireless Networks, December 1995.

[7] C. Perkins, “IP Mobility Support”, RFC 2001, October 1996.

[8] D. B. Johnson, C. Perkins, “Mobility Support in IPv6”, Internet Draft (work in progress), <draft-ietf-mobileip-ipv6-13.txt>.

[9] A. T. Campbell, J. Gomez, C-Y. Wan, S. Kim, Z. Turanyi, A. Valko, “Cellular IP”, Internet Draft (work in progress), <draft-ietf-mobileip-cellularip-00>, January 2000.

[10] T. Inzerilli, “SIMPLE, a Scalable Intra-domain Mobility Protocol using Local Encapsulation for Mobile IPv6 and Mobile IP”, IST Mobile Communications Summit 2000, Galway, Ireland, October 2000. pp. 587-594.

Introducción de IP en las redes móviles celulares: Evolución del IETF, 3GPP y 3GPP2. Cellular IP como ejemplo de solución

Rafael Vidal Ferré, Eduard García Villegas
Departament d'Enginyeria Telemàtica. Universitat Politècnica de Catalunya.
Jordi Girona 1 y 3. Campus Nord, Mód C3, UPC. 08034 Barcelona
Teléfono: 934 016 013 Fax: 934 015 981
E-mail: rvidal@mat.upc.es, eduardo.garcia2@upcnet.es

Abstract. *This paper wants to give a general overview of the problems for mobility support using IP protocol, focusing in cellular communications networks. Firstly, a full description of the IETF work and its latest evolution is done. Then, the strategies of third generation groups for integration of IETF work are commented. In a second part, the IETF proposed Mobile IP protocol is analysed showing its problems for using this protocol in cellular world. Later, the Cellular IP protocol is described. Cellular IP is one of possible counterparts for Mobile IP in order to fulfil micro-mobility requirements. Finally, a Cellular IP testbed is presented and comments on the implication of mobility on the performance of TCP and UDP traffic are given.*

1 Introducción

De la mano de Internet, el protocolo IP está convirtiéndose en la *lingua franca* de las redes de comunicaciones actuales. Sin embargo, su adopción para ciertas aplicaciones no esta exenta de problemas, muchos de los cuales permanecen todavía por resolver. En el presente artículo nos centraremos en uno de estos problemas, el del soporte de la movilidad en IP.

1.1 IP en las redes móviles

El crecimiento de las comunicaciones móviles y en especial el de las celulares que se está produciendo en los últimos años no tiene precedentes. Para el caso de España la cuota de penetración ya supera el 50% y el número de líneas móviles ya supera al de fijas.

Por otra parte, la evolución de las redes celulares actuales de segunda generación (2G) hasta la tercera generación (3G) pasa por ofrecer velocidades más elevadas y acceso por conmutación de paquetes, como es el caso de GPRS (*General Packet Radio Service*). Todo ello a fin de prepararse para un horizonte en que el tráfico de datos va a ser superior al de voz; apareciendo el acceso a redes IP en general y a Internet en concreto como principal artífice de esta situación.

Si a todo ello añadimos el trabajo que se está realizando para soportar el transporte de voz sobre IP está bastante maduro, con soluciones comerciales disponibles, puede entenderse porque la opción de una red de acceso de comunicaciones celulares basada totalmente en IP va ganando peso.

El siguiente paso lógico sería que esta red de manera natural asumiera todas las funciones necesarias para el soporte de la movilidad. En está dirección se están moviendo los dos grupos que desarrollan la 3G: 3GPP (*Third Generation Partnership Project*) y 3GPP2 (*Third Generation Partnership Project* 2).

Cada uno siguiendo su camino, los grupos 3G estudian como convertir sus redes de acceso en redes totalmente IP basadas en *routers* y en como adoptar el trabajo que está realizando el IETF (*Internet Engineering Task Force*) para ofrecer movilidad mediante IP.

El IETF ha estado trabajando en una solución universal para conseguir la movilidad en IP conocida como Mobile IP. Se trata pues, de una solución general no optimizada para ningún tipo de red de acceso y es aquí donde surgen los problemas. Uno de los requisitos clave que demandan las redes de 3G y de manera general las redes celulares es el soporte de la micro movilidad; entendiendo por micro movilidad, la posibilidad de cambiar de una manera frecuente y rápida de punto de acceso dentro de red. Como se verá más adelante, Mobile IP tiene serias limitaciones para cumplir con este requisito.

Actualmente existe una sinergia importante entre los grupos de 3G y el IETF: por una lado se intenta ver la como mejorar el protocolo Mobile IP para cumplir con los requisitos de 3G sin perder de vista su carácter de solución universal y por otro se ha creado un nuevo grupo especializado en micro movilidad.

Dejando a un lado el IETF y los grupos de 3G también cabe mencionar al MWIF (*Mobile Wireless Internet Forum*). Creado en febrero del 2000, este foro está formado por empresas muy significativas en ámbitos como las comunicaciones móviles, las redes de datos, el software o la electrónica. Su propósito es el desarrollo de unas especificaciones clave que

permitan la utilización de IP en cualquier tipo de redes sin hilos buscando aunar en una única aproximación los esfuerzos de otros grupos como el IETF, 3GPP o 3GPP2.

El artículo se compone de tres partes. En la primera se presenta la actividad de los grupos de trabajo del IETF relacionados directa o indirectamente con la movilidad, viendo el impacto que ha tenido en ellos el mundo de las comunicaciones móviles celulares, provocando la restructuración de algunos grupos y la creación de otros. Seguidamente se comentará el trabajo realizado en los grupos de 3G para la adopción de los protocolos desarrollados por el IETF para soportar la movilidad.

En la segunda parte se describirá la solución general para el soporte de la movilidad del IETF basada en el protocolo Mobile IP (MIP). A continuación, después de ver sus limitaciones para su utilización en redes celulares, se presentará el protocolo Cellular IP (CIP), diseñado para superar estas limitaciones y para cooperar con Mobile IP ofreciendo una solución global para el soporte de la movilidad.

Finalmente, en la tercera parte, se describirá la maqueta que se ha construido para analizar el protocolo Cellular IP y se comentarán los resultados obtenidos.

2 Movilidad en IP. IETF

En los siguientes apartados se describirá el trabajo que llevan a cabo diferentes grupos del IETF relacionados de una manera directa o indirecta con la movilidad.

2.1 Grupos de trabajo relacionados

Dejando a un lado las redes *ad-hoc* y la mobilidad de usuarios entre ISPs (*Internet Service Providers*), temas que se tratan respectivamente en los grupos de trabajo MANET (*Mobile Ad-hoc Netwroks*) y ROAMOPS (*Roaming Operations*) el foco de trabajo en movilidad en IP ha sido y es el grupo MOBILEIP (*IP Routing for Wireless/Mobile Hosts*)

El trabajo de MOBILEIP gira en torno al protocolo Mobile IP (MIP). Este protocolo ofrece el mantenimiento de la dirección IP independientemente de la localización de la máquina que la posea, con un encaminamiento trasparente de los paquetes IP e intentando aumentar en lo mínimo los flujos de señalización. Todo esto manteniendo activas las conexiones TCP y las vinculaciones con los puertos UDP. Existen dos versiones de este protocolo, una estandarizada para IPv4 [1] y otra que todavía tiene el carácter de *draft* para IPv6 [2] pero que se espera su propuesta como va estándar a corto plazo.

Dejando a un lado la estandarización del de la versión para IPv6 de Mobile IP y la revisión de la versión para IPv4 [3], la actividad actual del grupo gira en torno a la solución de dos grandes problemas: la seguridad y la mejora de prestaciones para conseguir traspasos rápidos.

En verano del 2000 se produjeron una serie de discusiones en torno a las diferentes, y muy numerosas, propuestas presentadas en forma de *draft* para conseguir un traspaso rápido. Dentro de las propuestas se podían diferenciar dos grupos. El primero las dirigidas a solucionar el problema de la micro movilidad, con un grado de compatibilidad y cooperación respecto a Mobile IP más o menos grande. En la mayoría de los casos la red de referencia era de tipo celular. Dentro de este grupo destacaron los protocolos HAWAII *(Handoff-Aware Wireless Access Internet Infrastructure*) [4][5] y Cellular IP [6] [7], del que se hablará más adelante.

En el segundo grupo figuraban las soluciones basadas completamente en Mobile IP. Se trataba de soluciones que, respetando el carácter de solución universal no ligada a ninguna tecnología de Mobile IP, pretendían mejorar su rendimiento para conseguir traspasos más rápidos.

El resultado de estas discusiones fue una refundación del grupo de trabajo MOBILEIP que dejaba fuera las propuestas del primer grupo con objeto de mantener el carácter universal de Mobile IP como solución para el soporte de la movilidad. Además se crearon dos equipos de trabajo para buscar una solución en común para MIPv4 y otra para MIPv6. El trabajo de estos grupos ha cuajado de momento en sendos *drafts* [8] [9].

Todo esto no significa que MOBILEIP deje de lado toda la problemática de las redes celulares. Como ya se ha comentado en la introducción existe una estrecha relación con los grupos de 3G y como ejemplo de ella puede verse el *draft* [10] en el que se plantean las extensiones necesarias para que Mobile IP pueda administrar la movilidad en redes cdma2000. Otra de las implicaciones de la refundación fue la creación de dos equipos que debían aunar las propuestas

Otra consecuencia de la refundación fue la reciente aparición de un nuevo grupo, SEAMOBY (*Context and Micro-mobility routing*). Los objetivos de SEAMOBY son el desarrollo de un protocolo que soporte la micro movilidad, con traspasos rápidos en la red de acceso y *paging*, y la provisión de mecanismos que permitan el intercambio de información de estado, como pueden ser el nivel de calidad de servicio asociado al usuario o un contexto de seguridad.

2.2 Otros grupos

A parte de los grupos comentados en el apartado anterior, cuya dedicación es exclusiva a los temas de movilidad, existen toda una serie de grupos cuyo trabajo tiene una relación significativa con esta

problemática. Destacaremos dos: ROHC (*Robust Header Compression*) y AAA (*Authentication, Authorization and Accounting*).

El objetivo de ROHC es conseguir un sistema de compresión que funcione correctamente sobre enlaces con tasas de error elevadas y retardos importantes. La motivación principal es el envío de información en tiempo real (voz o vídeo de baja calidad) sobre enlaces celulares. La combinación de protocolos IP/UDP/RTP/TCP utilizada para el transporte de tráfico *real-time* conlleva un alto *overhead*. Para trabajar eficientemente sobre enlaces de baja velocidad, como son los de las redes celulares, es necesario utilizar métodos de compresión. Una posible solución pasaría por la utilización de los algoritmos tradicionales de compresión de cabeceras [11][12] pero la elevada tasa de error así como los elevados retardos que se puedan dar en una red celular hacen que su comportamiento no sea el idóneo. De ahí la necesidad un nuevo tipo de compresión. En los *draft* [13] y [14] se especifican los requerimientos que debería cumplir esta nueva codificación y se da una posible especificación de ella respectivamente.

El AAA es el grupo encargado de desarrollar los requerimientos para la autenticación, autorización y contabilidad. Estas funciones son de vital importancia para control del acceso a cualquier sistema. El AAA trata el caso de un sistema con terminales como un caso particular con unas necesidades propias que requieren de unas extensiones determinadas. Esto se ha traducido en un listado de requerimientos formulado por el grupo MOBILEIP [15] y en *draft* del AAA sobre las extensiones a realizar para cumplir con estos requerimientos [16].

3 Integración de los protocolos del IETF en 3G

Las aproximaciones realizadas por los dos grupos de 3G para integrar los protocolos desarrollados por el IETF están siendo diametralmente opuestas. Por un lado el 3GPP2 cuenta ya desde hace más de un año con un estándar [17] de lo que ellos denominan *Wireless* IP. En este documento se describen los requerimientos para soportar redes de paquetes inalámbricas en las redes de 3G basadas en cdma2000; diferenciando dos alternativas: *Simple IP*, basado en el protocolo PPP (*Point to Point Protocol*); y *Mobile IP* basado en el protocolo del mismo nombre. El documento también propone la utilización de servidores RADIUS (*Remote Authentication Dial In User Service*) para labores de AAA y la utilización de *Diffserv* para ofrecer calidad de servicio. Se trata pues de utilizar las soluciones ofrecidas por el IETF, aunque, como ya se ha dicho, no estén optimizadas para sistemas celulares.

Paralelamente el 3GPP2 tiene abierto otro proyecto en fase de definición denominado *All* IP, que consiste en el desarrollo de una red que se basa en IP como principal mecanismo para el transporte y la conmutación.

El camino por el que ha optado el grupo 3GPP es mucho más ambicioso. Todavía no se ha publicado ningún estándar referente al tema pero existe un trabajo minucioso y continuado que se refleja en [18]. En este *report* técnico el 3GPP propone una arquitectura basada totalmente en IP, *All* IP, para el transporte de todos los datos de usuario y señalización. El documento tiene una doble vertiente: la identificación de los problemas clave a resolver y la proposición de un plan de trabajo para ofrecer una *All* IP *release* 2000 del estándar UMTS (*Universal Mobile Telecommunications System*).

4 El protocolo Mobile IP

En los siguientes apartados se realizará una breve descripción del funcionamiento del protocolo Mobile IP acompañada de una reflexión sobre sus limitaciones para el soporte de micro movilidad.

4.1 Descripción general

Mobile IP define dos nuevas entidades conocidas genéricamente como agentes: el *Home Agent* (HA) y el *Foreign Agent* (FA), que no son más que dos encaminadores, uno en la red de origen del nodo móvil y otro en la que visita y que realizan funciones de gestión de datos similares a las del HLR (*Home Location Register*) y del VLR (*Visitor Location Register*) de las red celular GSM *(Global System for Mobile Communications*). El funcionamiento es simple. Cuando un nodo móvil se mueve hasta otra red recibe un aviso del FA de la red visitada para que se registre. De esta manera detecta su cambio de localización, ya sea respecto al HA o a un FA anterior. Entonces adquiere la dirección del FA (*care-of addres*) que queda registrada en su HA. A partir de aquí cualquier datagrama enviada al nodo móvil pasa por su HA que lo envía al FA mediante un túnel, que se encarga de hacerlo llegar al nodo móvil. En sentido contrario, el nodo móvil envía directamente los datagramas a su nodo destino.

4.2 Limitaciones

La simplicidad del protocolo Mobile IP tiene su precio: el soporte de la micro movilidad. En entornos de alta movilidad como los celulares en los que el nodo móvil cambia de punto de acceso con gran frecuencia el rendimiento del protocolo puede no ser el adecuado según el tipo de servicio que se quiera soportar. Cada cambio, aunque sea dentro de una misma red, requiere de un intercambio de señalización con el HA lo que ralentiza el proceso de actualización con la posterior pérdida de paquetes que esto supone. Como ya se ha comentado en secciones anteriores la solución de este problema a dado lugar a un sinfín de propuestas, centrándose el presente artículo en una de ellas, Cellular IP.

5 El protocolo Cellular IP

En la sección anterior hemos visto el protocolo Mobile IP así como sus limitaciones para soportar micro movilidad. En esta se describirá uno de los protocolos diseñados exclusivamente con el fin de soportar este tipo de movilidad y que combinado con Mobile IP permiten ofrecer una movilidad total dentro de una red IP (Fig. 1), el protocolo Cellular IP.

5.1 Características generales

Al tratarse de un sistema celular, Cellular IP ofrece una serie de ventajas, que aplicadas correctamente, pueden mejorar las prestaciones de las futuras redes IP inalámbricas, sin perder ninguna de las propiedades que caracterizan a las redes IP, como la flexibilidad, la escalabilidad y la robustez. De los sistemas celulares hereda los principios de administración de la movilidad, control de traspasos y localización de nodos inactivos. Cellular IP se basa en nodos sencillos y baratos que pueden ser interconectados para formar topologías arbitrarias y operar sin una configuración previa complicada.

El componente universal de una red Cellular IP es la estación base, que sirve de punto de acceso radio a la red y al mismo tiempo encamina paquetes IP y integra funciones de control de sistemas celulares, tradicionalmente implementadas en los MSCs (*Mobile Switching Center*) y en las BSCs (*Base Station Controller*). El encaminamiento IP se sustituye por el encaminamiento propio de Cellular IP, donde se integra localización y soporte a los traspasos sin alterar la pila de protocolos IP. Además, una estación base puede configurarse para desarrollar las funciones de *gateway*. El *gateway* es el nodo que se encargara de conectar la red de acceso Cellular IP con Internet. Cuando se da servicio a un nodo móvil originario de una red externa, el *gateway* ejerce de agente de movilidad (*Foreign Agent*) del protocolo Mobile IP.

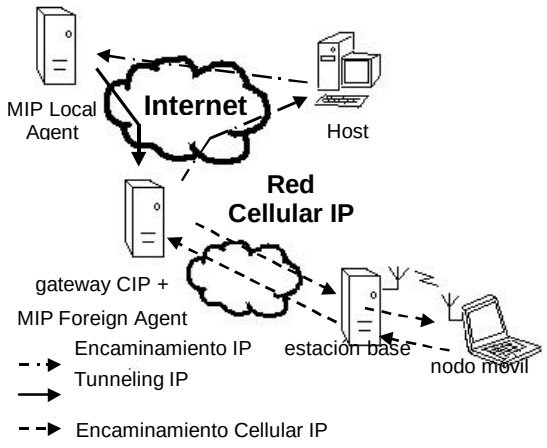


Figura 1 Solución global MIP + CIP

5.2 Funcionamiento

Antes de entrar a describir la maqueta Cellular IP y valorar su rendimiento, se hace necesaria una breve introducción al funcionamiento del protocolo. Este se resume de la siguiente forma:

Las estaciones base emiten periódicamente unas señales llamadas señales faro (*beacon signals*). Estas señales incorporan información sobre la estación base y sobre la red CIP. Los nodos móviles las utilizan para detectar cual es la estación base más cercana. Un nodo móvil transmite sus paquetes a través de la estación base de la que recibe una señal de mayor calidad. Para el nodo móvil, esa estación base funciona como su *router* por defecto.

Todos los paquetes IP que envía el nodo móvil se encaminan directamente desde la estación base hacia el *gateway*, independientemente de la dirección de destino. Para hacerlo se sigue un algoritmo de encaminamiento del tipo *hop-by-hop shortest path*, es decir, el paquete recorre el camino más corto hacia el *gateway*.

Todos los nodos CIP disponen de una tabla denominada *Route Cache*. Los paquetes enviados por un nodo móvil actualizan los datos de estas tablas en todos los nodos por los que pasan. Una entrada en esta *cache* guarda la dirección IP del nodo móvil que generó el paquete, la interfaz de red por donde llegó y la dirección física del último vecino de bajada por donde pasó.

La concatenación de la información de estas tablas, referida a un nodo móvil, sirve para reconstruir el camino necesario para que los paquetes destinados a ese nodo móvil lleguen a su destino. Aunque un nodo móvil cambie el punto de acceso a la red, las *route caches* apuntan a la nueva localización del nodo, ya que éste se encarga de generar nuevas entradas mediante el envío de paquetes de control (paquete *route-update*). Estos paquetes también sirven para evitar que las entradas referentes a un nodo móvil caduquen, ya que tienen un tiempo de vida limitado dentro de las *route caches*. Así, si un nodo quiere mantenerse activo y localizable en la red, pero no tiene paquetes para enviar que mantengan sus entradas en las *caches*, deberá enviar periódicamente paquetes *route-update*. Un ejemplo de esta sería una transmisión UDP en la que el nodo móvil es el receptor y no tiene la necesidad de enviar nada.

Celluar IP también incorpora un mecanismo de *paging* que permite a un nodo móvil inactivo mantenerse localizable dentro de la red. Para ello se utilizan otras tablas, las denominadas *paging caches*. Consultando estas tablas se consigue encaminar los paquetes hacia nodos inactivos de los que no se dispone de información en las *routing caches*.

6 Maqueta Cellular IP

Una vez visto el funcionamiento del protocolo Cellular IP, en los sucesivos apartados se verá la descripción de la maqueta realizada para su evaluación, así como los resultados de esta.

6.1 Características software y hardware

La implementación del protocolo Cellular IP utilizada ha sido desarrollada por el grupo Comet de la Universidad de Columbia, el mismo que ha creado el protocolo Cellular IP. Actualmente se pueden obtener libremente versiones para FreeBSD y para Linux, esperándose una futura versión para Windows NT. En la maqueta se utilizó la versión para Linux, utilizando como sistema operativo Red Hat 6.2 (kernel 2.2.14-5.0).

Cellular IP se basa en dos módulos *software*: el nodo y el nodo móvil. Ambos, mediante el uso de la *librería Berkeley Packet Filter's Packet Capture* (PCAP), filtran paquetes IP del medio físico para moverlos a espacio de usuario, donde serán procesados. El módulo del nodo incorpora funcionalidades de *router*, implementa servicios de localización y si dispone de interfaz radio, también hará de punto de acceso y de *router* por defecto de los nodos móviles que se conecten a él. Además puede realizar la función de *gateway* confiando en las operaciones de encaminamiento IP implementadas en el núcleo del sistema operativo. El módulo del nodo móvil funciona como un demonio que se ejecuta en espacio de usuario. La pila estándar del protocolo IP no se ve afectada por este demonio y las aplicaciones serán trasparentes a la movilidad. Este módulo es quien controla los traspasos gracias a que mantiene estadísticas de la calidad de señal de las diferentes estaciones base.

Los requisitos de *hardware* para utilizar CIP son sólo dos: disponer de procesadores Pentium 200 MHz o superior, y para los nodos que requieren interfaz radio es aconsejable un *slot* PCMCIA ya que la implementación de CIP sólo soporta tarjetas WaveLAN (actualmente Orinoco) y Aironet, que mayoritariamente son de este tipo.

6.2 Descripción de la maqueta

En la Fig. 2 se representa la maqueta montada con el fin de evaluar el rendimiento del protocolo Cellular IP. La red de acceso CIP construida consta de tres PCs, dos de ellos disponen de interfaces radio y servirán de punto de acceso a la red (estaciones base) a los nodos móviles. Las dos estaciones base se conectan junto al tercer PC a un *hub* mediante interfaces Ethernet 10 Mbps. Este tercer nodo hará el papel de *gateway*, es decir, será el punto de conexión de la red CIP con otras redes. En este caso, el nodo *gateway* está conectado a un *router* que sirve a dos redes: la red de acceso CIP y otra red formada por un

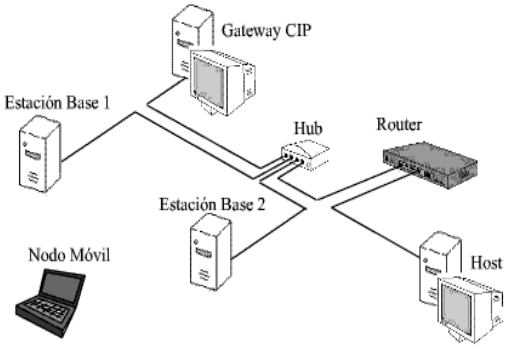


Figura 2 Esquema de la maqueta Cellular IP

único nodo. Como nodo móvil se usa un ordenador portátil con una interfaz radio.

El objetivo del montaje consiste en mantener diferentes tipos de conexiones entre el nodo fijo y el nodo móvil mientras éste cambia su punto de acceso a la red de una estación base a otra.

Como estaciones base se han utilizado dos PCs Pentium II 350 MHz con 64 MB de RAM, como nodo *gateway* un Pentium II 266 MHz con 64 MB de RAM y como nodo móvil, un PC portátil Celeron 300 MHz con 64 MB de RAM. El router usado es un Cisco 2621 y las interfaces radio son tarjetas PCMCIA WLAN IEEE 802.11b de Orinoco.

6.3 Evaluación de la maqueta

Para el estudio detallado del rendimiento de la maqueta CIP era fundamental observar el comportamiento ante el cambio de estación base durante transmisiones TCP y UDP, protocolos más comunes sobre IP. Básicamente un traspaso consiste en cambiar la frecuencia del nodo móvil a la de la nueva estación base y enviar un paquete de control para actualizar su posición en la red, esto provoca que, mientras este paquete no actualiza el nuevo camino de bajada, los paquetes que ya habían sido enviados por el antiguo camino se pierdan. Desgraciadamente las soluciones que permiten soportar traspasos con tarjetas IEEE 802.11 son propietarias; disponiendo el grupo Comet de *drivers* que soportan esta facilidad pero que sin embargo no son públicos. Por esta razón, las pruebas realizadas difieren respecto a las suyas [19] en que la maqueta que aquí se presenta trabaja en un entorno sin diversidad de frecuencias, es decir, las dos estaciones base y el nodo móvil transmiten en el mismo canal. Como se verá más adelante, esto unido al hecho de tener las dos estaciones base conectadas a la misma interfaz del *gateway*, va a reducir el rendimiento de la maqueta.

En una primera prueba se realizaron transmisiones UDP ascendentes, el origen de la transmisión es el nodo móvil, y descendentes, el destino de la transmisión es el nodo móvil. Se observa que los

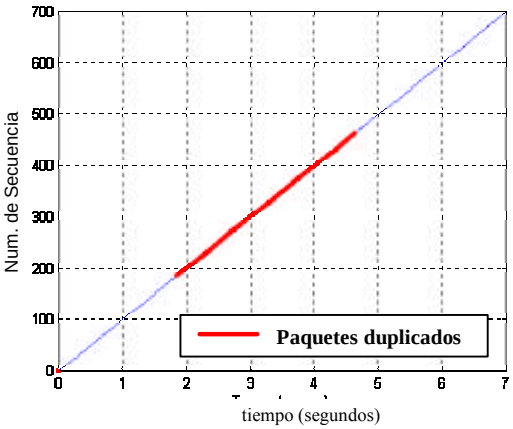


Figura 3 Transmisión UDP descendente

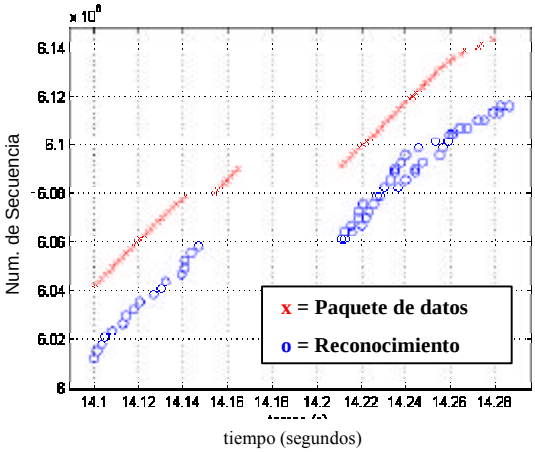


Figura 4 Transmisión TCP ascendente

traspasos no tienen ningún efecto sobre las transmisiones ascendentes, en las descendentes, en cambio, un traspaso significaba la recepción de paquetes duplicados durante un tiempo constante (ver Fig. 3). Este tiempo era igual a *route-timeout*, tiempo de vida de una entrada en la tabla *Route Cache*. Este efecto se produce debido a que, al estar la antigua estación base conectada a la misma interfaz del *gateway* que la nueva, mediante PCAP continúa capturando paquetes IP aunque no vayan dirigidos a su dirección física, y mientras tenga información sobre el destino del paquete, seguirá transmitiéndolos al canal radio común.

La recepción de paquetes duplicados, aparte de la utilización ineficaz que representa, puede ser solucionada por los niveles superiores. Sin embargo las aplicaciones en tiempo real probadas no responden igual. Gphone (aplicación de voz sobre IP) no descarta los paquetes duplicados, lo que introduce distorsiones sobre la voz que oye el usuario del nodo móvil. En cambio, la recepción de vídeo y audio con RealPlayer no se ve afectada por recibir paquetes duplicados, además, con una interfaz que proporciona una tasa de datos de hasta 5Mbps, se consiguen transmisiones de gran calidad.

En transmisiones TCP, la recepción de ráfagas de paquetes duplicados va a ser más crítico. Como antes, los paquetes duplicados sólo se reciben en sentido descendente, cuando el nodo móvil es la fuente de paquetes TCP, recibirá reconocimientos duplicados, cuando es el receptor de la transmisión y es quien envía reconocimientos, recibirá paquetes de información duplicados. El primer caso es el que se observa en la Fig 4, a partir del traspaso (segundo 14,20) el nodo móvil recibe reconocimientos duplicados, pero ese hecho no altera el rendimiento del protocolo, ya que la mayoría de implementaciones de TCP no consideran la retransmisión de un paquete hasta el tercer reconocimiento duplicado [20] y en nuestro caso sólo se reciben dos reconocimientos iguales, uno de cada estación base.

La situación resulta muy diferente cuando lo que se recibe duplicado son paquetes de datos en lugar de los reconocimientos. En ese caso, un traspaso se traduce en un empeoramiento del rendimiento que aumenta con la frecuencia de los traspasos. En la Fig. 5 se observan dos traspasos (segundos 5 y 15) en una transmisión TCP descendente y ascendente. Como se ha explicado, no se observa ningún efecto cuando la fuente de datos es el nodo móvil, pero sí cuando es la fuente de los reconocimientos.

Esa pérdida de pendiente en el caudal se debe a la recepción de ráfagas de paquetes duplicados. Cuando llega al nodo móvil una ráfaga de paquetes que ya se ha recibido anteriormente, éste responde con tantos reconocimientos duplicados como paquetes han llegado en ráfaga. Al llegar estos reconocimientos al emisor, provocarán la retransmisión del paquete correspondiente pensando que éste no ha alcanzado el destino. Así que este paquete que el nodo móvil ya había recibido por duplicado, le volverá a llegar a través de las dos estaciones base, lo que volverá a provocar nuevos reconocimientos. Este comportamiento es el que provoca un descenso del rendimiento durante un tiempo ligeramente superior a *route-timeout*. En la Fig. 6 se muestra un ejemplo donde se puede observar el deterioro de la transmisión debido a la recepción de ráfagas de paquetes duplicados.

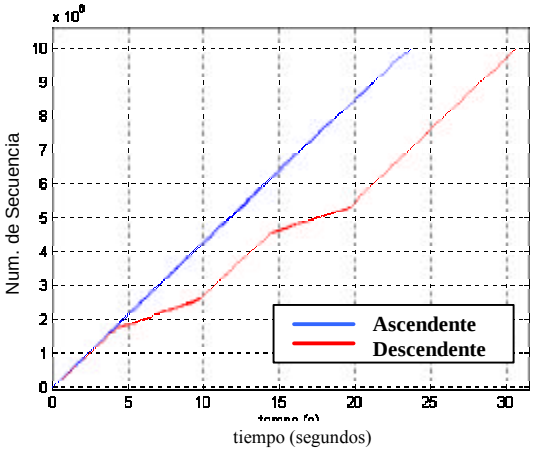


Figura 5 Transmisión TCP descendente y ascendente

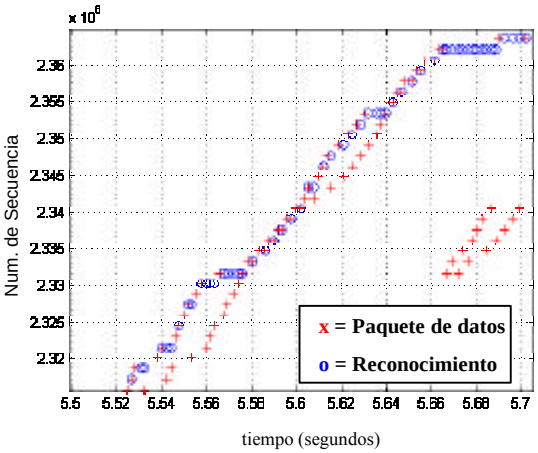


Figura 6 Deterioro transmisión TCP debido recepción duplicados

Como se ha visto, la falta de sincronización entre las estaciones base es una grave fuente de problemas. Se puede evitar en parte, usando el método *semisoft-handoff* que ofrece la implementación de CIP. Este método, pensado para reducir el número de paquetes perdidos durante los traspasos que incluyen cambio de frecuencia, añade un retardo a la nueva estación base para tratar de sincronizar al máximo el flujo de paquetes a través de ambas estaciones base. Otras soluciones consisten en el cambio en la configuración de la red, si las estaciones base estuvieran conectadas al *gateway* en interfaces diferentes, un traspaso se traduciría en unos pocos paquetes duplicados, dependiendo de la topología de la red, efecto preferible a la pérdida de paquetes.

7 Conclusiones

En el presente artículo se ha pretendido dar una visión general del trabajo del IETF para conseguir el soporte de la movilidad en IP así como de las estrategias de los grupos de 3G para incorporar este trabajo en sus redes. De todo ello se desprende que el camino para conseguir redes celulares totalmente IP está abierto aunque quizás su final pueda verse reflejado a más largo plazo, en la llamada cuarta generación.

Por otra parte se ha descrito el funcionamiento del protocolo Mobile IP desarrollado por el grupo de trabajo MOBILEIP del IETF; indicando sus carencias para el soporte micro movilidad. A continuación se ha expuesto el funcionamiento del protocolo Cellular IP uno de los posibles candidatos que han sido presentados para cooperar con Mobile IP para resolver las citadas carencias.

Finalmente se ha descrito la maqueta realizada para la evaluación del protocolo Cellular IP sobre la que se ha realizado un análisis detallado del efecto de los traspasos sobre tráfico UDP y TCP. Los resultados obtenidos verifican los presentados por los autores del protocolo [19] con la salvedad de los ya comentados efectos derivados de trabajar sin diversidad en frecuencia en las interfaces radio IEEE 802.11b.

Agradecimientos

El trabajo presentado en este artículo se enmarca dentro del proyecto TIC2000-1041-C03-01 financiado por Comisión Interministerial de Ciencia y Tecnología (CICYT)

Referencias

[1] C.Perkins (editor). "IP Mobility Support". Octubre 1996. IETF RFC 2002

[2] D.B.Johnson, C.Perkins, "Mobility Support in IPv6". Noviembre 2000. IETF draft mobileip-ipv6-13

[3] C. Perkins (editor). "IP Mobility Support for IPv4, revised". Febrero 20001. IETF draft mobileip-rfc2002-bis-04

[4] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, L. Salgarelli. "IP micro-mobility support using HAWAII". Julio 2000. IETF draft mobileip-hawaii-01

[5] R. Ramjee, T. La Porta, L. Li. "Paging support for IP mobility". Julio 2000. IETF draft mobileip-paging-hawaii-01

[6] A.T.Campbell, J.Gomez, C-Y.Wan, S.Kim, Z. Turanyi, A. Valko. "Cellular IP". Enero 2000 IETF draft mobileip-cellularip-00

[7] Z.D.Shelby, D.Gatzounas, A.T.Campbell, CY.Wan. "Cellular IPv6". Noviembre 2000. IETF draft shelby-seamoby-cellularipv6-00.txt

[8] MIPv4 Handoffs Design Team. K. El Malki (editor). "Low latency Handoffs in Mobile IPv4". Febrero 2001. IETF draft mobileip-lowlatency-handoffs-v4-00

[9] Design Team on Fast Handovers with Mobile IPv6. G.Tsirtsis (editor) MIPv6. "Fast Handovers for Mobile IPv6" Julio 2001. IETF draft mobileip-fast-mipv6-00

[10] Y. Xu (editor). "Mobile IP Based Micro Mobility Management Protocol in The Third Generation Wireless Network ".Noviembre 2000. IETF draft mobileip-3gwireless-ext-05.txt

[11] V. Jacobson. "Compressing TCP/IP Headers for Low-Speed Serial Links". Febrero 1990. IETF RFC 1144

[12] S. Casner, V. Jacobson. "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links". Febrero 1999. IETF RFC 2508

[13] M. Degermark (editor). "Requirements for robust IP/UDP/RTP header compression". Febrero 2001. IETF draft rohc-rtp-requirements-05

[14] C. Bormann (editor). "RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed". Febrero 2001. IETF draft rohc-rtp-09

[15] S.Glass, T.Hiller, S.Jacobs, C.Perkins. "Mobile IP Authentication, Authorization, and Accounting Requirements". Octubre 2000. IETF RFC 2977

[16] P.R.Calhoun, C.Perkins. "Diameter Mobile IP Extensions". Marzo 2001. IETF draft aaa-diameter-mobileip-01

[17] 3rd Generation Partnership Project 2; Technical Specification Group Wireless Packet Data Networking. "Wireless IP Network Standard". Versión 1.0 Diciembre 1999. 3GPP2 P.S0001

[18] 3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects. "Architecture for an All IP network". Octubre 1999. 3G TR 23.922 version 1.0.0.

[19] A.T. Campbell, J. Gomez, S. Kim, A.G. Valkó, C.Wan."Design, Implementation and Evaluation of Cellular IP". IEEE Personal Communications Magazine. Agosto 2000, páginas 42-49

[20] W. Stevens. "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms". Enero 1997. IETF RFC 2001

Provisión de Calidad de Servicio Basada en Reservas para Entornos Móviles

Alberto López[1], Héctor Velayos[3], Tomás Robles[2], Nuria Villaseñor[3]

[1]Departamento de Ingeniería de la Información y las Comunicaciones, Universidad de Murcia,
Facultad de Informática, Campus Universitario de Espinardo - 30071 Murcia
Teléfono 968 364607, Fax: 968 364151
E-mail: alberto@dif.um.es

[2]Departamento de Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid
ETSI Telecomunicación, Ciudad Universitaria – 28040 Madrid
Teléfono: 91 336 7332, Fax: 91 336 7333
E-mail: robles@dit.upm.es

[3]Agora Systems S.A.
c/ Aravaca 12 3ºB - 28040 Madrid
Teléfono: 91 533 5857, Fax: fax: 91 534 8477
E-mail: {hvelayos, nuria_villasenor}@agoratechnologies.com

Abstract. *With the fast adoption of IP-based communications for mobile computing, users are expecting a similar service in wireless and wired networks. This raises the need for setting guarantees to the quality of the offered service (QoS), despite the technology of the access network or the mobility of the terminal. This generates a new challenge for QoS provision, as it will have to deal with terminals changing their point of attachment to the network. In this paper an optimisation for the operation of reservation based QoS is given for mobile environments: the coupling of the reservation protocol RSVP with different per-host micro-mobility protocols. The micro-mobility and QoS signalling mechanism are coupled either loosely via a triggering mechanism, or more tightly so the QoS and mobility information is carried by the same protocol. Qualitative and quantitative results of this coupling is presented. The procedure includes the comparison of performance parameters such as delay, loss and throughput when protocols are coupled and de-coupled.*

1 Introducción.

El crecimiento de la industria de telefonía móvil en la última década ha sido exponencial, basado principalmente en sistemas existentes de 2ª generación como GSM y tráfico de voz. Con la llegada de sistemas de 3ª generación como UMTS se espera que aproximadamente en el año 2005 el número de subscriptores de servicios móviles en el mundo supere al número de usuarios de telefonía fija, si no antes. Por otro lado, la cantidad de tráfico de datos de las redes fijas ha sufrido un incremento explosivo, debido principalmente al crecimiento de Internet y a la proliferación de redes intranet corporativas. Las aplicaciones usadas en esos entornos están basadas principalmente en IP, como el Web y las aplicaciones multimedia de banda ancha. Las redes IP pronto soportarán un rango de servicios que van desde los tradicionales IP hasta las aplicaciones interactivas multimedia y servidores de voz. Todos esos servicios requerirán garantías de Calidad de Servicio (QoS) diferentes por parte de la red, y los mecanismos de calidad de servicio presentes deberán asegurar a los usuarios el servicio adecuado para sus datos de aplicación.

Si tenemos en cuenta que los nuevos sistemas de 3ª generación están moviéndose hacia sistemas de transporte basados en IP, el siguiente paso lógico es el de extender ése transporte IP hacia el usuario final de esos servicios, ofreciendo por tanto la verdadera Internet móvil. Uno de los principales problemas a resolver en ese entorno es la provisión de QoS.

Entre las propuestas para proporcionar un tratamiento privilegiado a ciertos flujos, el mecanismo de señalización estándar de-facto para reserva de recursos es el denominado Servicios Integrados [1] y el Protocolo de Reserva de Recursos (RSVP) [2]. Éstos fueron diseñados para proporcionar reservas de recursos explícitas basadas en flujos principalmente en redes fijas. Sin embargo, la provisión y mantenimiento de QoS en un entorno móvil dinámico no es una tarea sencilla. Además de las propias dificultades que podemos encontrar a la hora de proporcionar QoS en redes fijas nos encontramos con que el nodo móvil puede

cambiar potencialmente su punto de acceso a la red[1] numerosas veces durante una sesión, por lo que el verdadero desafío es poder mantener el nivel de servicio original (el solicitado) a medida que el terminal se mueve. Además existen otros problemas como los cambios de dirección IP (Mobile IP [3]) y la variabilidad y escasez de recursos en el enlace inalámbrico, que pueden crear situaciones en las que no se pueda garantizar ciertos niveles de servicio a los terminales, y por tanto se produzcan violaciones de la QoS. Una violación de la QoS puede resultar en retardos excesivos, pérdidas de paquetes o incluso en una total denegación del servicio.

Por norma general los mecanismo de QoS y movilidad han evolucionado de manera independiente. El protocolo RSVP estándar puede reparar cambios producidos en el camino pero no es consciente del origen o la causa del cambio. La propuesta aquí presentada consiste en acoplar el protocolo de movilidad con RSVP. De esta manera se podría realizar un restablecimiento más rápido de la reserva tras el handover y se minimizaría el impacto causado a los flujos con recursos reservados. Nuestras simulaciones mostrarán cómo el acoplado de los protocolos junto con la priorización de la señalización de establecimiento de reservas reducen el impacto en el rendimiento de manera considerable.

A lo largo del texto nos referiremos de manera implícita tan solo a mecanismos del tipo soft-state tales como RSVP, aunque estos métodos pueden ser aplicadados igualmente a mecanismos del tipo hard-state.

2. Acoplado de protocolos.

La calidad de servicio basada en reservas asume, de manera implícita, un camino relativamente estable a lo largo de la red. Los cambios en las rutas sólo se reflejan en las reservas una vez que el mensaje de refresco ha recorrido todos los nodos del nuevo camino, lo cual puede introducir un retardo extremo a extremo muy elevado desde el nodo emisor hasta el nodo móvil. Mecanismos de refresco y soft-state en protocolos basados en reservas como RSVP se diseñaron originalmente para tratar casos de enlaces caídos, que por otro lado ocurren con poca frecuencia. Mecanismos más avanzados como el de reparación del camino local (Local Path Repair) se diseñaron para reparar de manera eficiente las reservas de RSVP tras un cambio en las rutas, pero su funcionamiento no es óptimo si el cambio en la ruta no es visible de manera explícita para los routers. La mayoría de los protocolos de movilidad más comunes tales como MobileIP o MobileIP jerárquico [4] funcionan de esa manera. Además, dado que un cambio en la ruta suele implicar que el terminal móvil sea responsable de activar o parar el mecanismo de reparación de camino, se introduce una sobrecarga de señalización en el terminal móvil.

2.1 Cooperación entre protocolos.

La solución aquí propuesta consiste en la colaboración entre los mecanismos de señalización de QoS los de movilidad local. Esta colaboración o acoplado se puede diseñar de formas muy diversas, aunque podemos identificar tres niveles fundamentales:

- **No acoplados**: Este es el estado actual, donde un protocolo no es consciente de la existencia del otro, aparte de por los efectos externos como por ejemplo el propio cambio en la ruta.

- **Acoplado 'débil'**: Se utilizan mecanismos de disparo para informar a un protocolo sobre cambios o acciones del otro.

- **Acoplado 'fuerte'**: La información de calidad de servicio y movilidad es transportada conjuntamente de alguna manera, por ejemplo añadiendo información de QoS en los mensajes del protocolo de movilidad. Un ejemplo claro de este acoplado aplicado a QoS es el protocolo INSIGNIA [5].

La elección de una de estas opciones es un compromiso entre aplicabilidad, complejidad y rendimiento. Si ambos protocolos conviven sin ningún tipo de información sobre el otro no es posible aprovecharse de algunas de sus características avanzadas, y por lo tanto no es posible obtener un aumento del rendimiento, aunque la transparencia se conserva. Esta transparencia hace posible el desarrollo libre e independiente de los protocolos. Por otro lado el acoplamiento fuerte tiene la ventaja de poder obtener un rendimiento óptimo a un mayor coste en aplicabilidad y desarrollo, ya que las soluciones existentes deben ser modificadas de manera más profunda. En general un mayor nivel de acoplamiento entre elementos de la red no es una buena práctica de diseño ya que puede violar algunos de los principios arquitectónicos de Internet, tales como la división en capas o el principio extremo a extremo [6].

2.2 Acoplado 'débil' de protocolos de QoS y movilidad.

Entre las alternativas anteriores proponemos el acoplado débil de los mecanismos de QoS y los protocolos de movilidad local. Al mejorar le mecanismo de QoS en el entorno móvil la reparación del camino local es posible y los cambios en la reserva son tan solo locales al área

[1] Este proceso se conoce con el nombre de *handover*.

afectada por el cambio en la ruta, sin ninguna sobrecarga de procesamiento o señalización en los terminales móviles.

En la aproximación 'débil', el cambio de posición del nodo móvil, y por lo tanto las actualizaciones de la información de encaminamiento en la propia red, disparan la generación de mensajes de reparación RSVP PATH. Éste mecanismo tan sólo repara la parte de la reserva que se ha perdido, provocando que la reserva extremo a extremo pueda instalarse de manera más rápida ya que no se necesita señalizar nuevamente desde el emisor hasta el receptor (con el retardo que ello supone). Hay que tener en cuenta que la señalización de reparación de la reserva no debe realizarse hasta que exista la seguridad de que la nueva ruta en la red es estable, por lo que existe un retardo fijo (el tiempo en que la nueva ruta se establece) que no puede ser reducido y depende directamente del mecanismo de movilidad subyacente. La implementación de este mecanismo implica cambios en todos los nodos envueltos en la provisión de QoS excepto en los nodos móviles.

2.3 Mecanismos complementarios.

En un entorno móvil el acoplamiento débil proporciona una mejora en el rendimiento pero puede no ser suficiente. Existen una serie de mecanismos que complementan éste acoplado:

Priorización de la señalización de QoS: El acoplado débil proporciona un mecanismo por el cual una reserva puede reinstalarse tan pronto como el nuevo camino es estable, permitiendo un uso más eficiente de los recursos y minimizando el impacto del handover. Sin embargo, si el nuevo camino contiene enlaces sobrecargados y los mensajes de QoS se pierden, el tiempo asignado al soft-state vencerá y los paquetes de datos pertenecientes al flujo de la reserva caerán a un prioridad best—effort, pudiendo producirse una violación de la QoS. Si proporcionamos prioridad a los paquetes de señalización de QoS este efecto puede minimizarse y por lo tanto la nueva reserva puede instalarse. La priorización puede realizarse de diversas maneras, con mecanismos como DiffServ [7] o simplemente reservando una cierta cantidad de ancho de banda en los routers con colas del tipo CBQ [8]. Si no hay suficientes recursos en los nuevos enlaces (p. ej. ya existen otras reservas y no queda ancho de banda), entonces la reserva no podrá ser reinstalada. Esto puede resolverse con mecanismos de reserva anticipada como MRSVP [9] y otros, que quedan fuera del alcance de este documento.

Priorizaciuón de paquetes 'en handover': Denominaremos paquetes 'en handover' a aquellos paquetes pertenecientes a un flujo de datos que, a pesar de tener una reserva instalada en su camino de datos anterior, están pasando durante un periodo reducido de tiempo por unos nodos que no poseen (aún) información de reserva debido al cambio de ruta producida por un handover, y por lo tanto están siendo tratados como tráfico best-effort.

Muchas modificaciones del protocolo RSVP han intentado establecer una reserva antes de que el handover ocurra. En esta propuesta nosotros evitamos esta opción – primero porque no todos los handover son planeados y por lo tanto no hay tiempo de hacerlo, segundo por la carga de proceso y de señalización que imponen, así como la posible ineficiencia en el uso de los recursos de la red que son inevitables dado que la nueva ruta no puede ser determinada hasta que se ha producido el cambio en ella. Por lo tanto se hace necesario un mecanismo para tratar el tráfico que temporalmente carece de reserva. La priorización de estos paquetes proporcionan un mecanismo para el tráfico en handover basado en reservas acceda a unas bandas de guarda de ancho de banda, reservado únicamente para tráfico de alta prioridad proveniente de un handover. La priorización de los datos 'en handover' que tiene que ser encaminada por un túnel hacia el nuevo destino proporciona una QoS mejorada sin necesidad de usar reservas temporales (que producen sobrecarga tanto en señalización como en tiempo de proceso) [10].

Esta priorización también puede usarse en procedimientos de reserva salto a salto (como RSVP) que son afectados por la movilidad. Permite a este tráfico tener una prioridad alta mientras la red espera a que el nuevo camino se estabilice antes de intentar reparar la reserva.

Protocolos de transferencia de contexto: Un protocolo de transferencia de contexto transfiere la información de estado sobre los requisitos de QoS del nodo móvil desde el antiguo router de acceso hasta el nuevo. Este intercambio puede ser iniciado de varias maneras: por indicaciones de handover de la capa de enlace, o por ejemplo, en el caso de protocolos de movilidad local basados en túneles, podría ser iniciados por los nodos extremos de los túneles.

El protocolo de transferencia de contexto requiere el soporte de todos los nodos que soportan la movilidad en la red de acceso. El protocolo necesario para activar este procedimiento, así como los parámetros a ser intercambiados, son un objeto de estudio hoy en día. Por ejemplo, el concepto de protocolo de transferencia de contexto está empezando a ser considerado por el grupo de trabajo Seamoby [11] (Seamless Mobility WG) del IETF. En particular, Seamoby propone un concepto de transferencia de contexto en unos términos más amplios que los aquí tratados (QoS), al transferir información sobre seguridad, compresión de cabeceras y otros conceptos además del ya comentado de QoS.

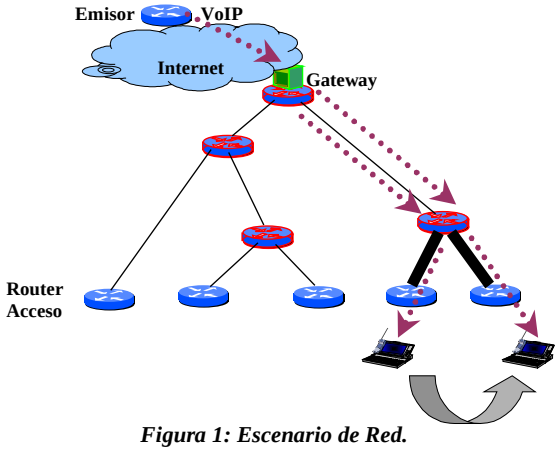


Figura 1: Escenario de Red.

Un protocolo de este tipo proporciona un soporte adecuado para los handovers eficientes y sin pérdidas[2], y en particular, también da soporte a otras posibles mejoras como la reparación local RSVP. Se asume que las reservas de QoS de la capa de enlace son restauradas también como resultado de este proceso de handover. El protocolo de transferencia de contexto proporciona al nuevo router de acceso suficiente información como para que todos los paquetes IP recibidos sean vinculados a las reservas inalámbricas adecuadas. De esta manera se puede restaurar de manera rápida en el enlace inalámbrico, que generalmente es el enlace más débil de todo el camino de datos.

Por otro lado, si se utiliza un protocolo de transferencia de contexto junto con la reparación local de RSVP, se puede reducir la carga de señalización en el enlace inalámbrico como veremos a continuación

---

De todos estos mecanismos, tan solo la priorización de la señalización de QoS es un requisito indispensable para la propuesta de acoplamiento 'débil'. De todas formas, todos estos mecanismos proporcionan un marco para el seamless handover en entornos con QoS basada en reservas.

3. Simulaciones

En este apartado se presentan diversas simulaciones que dan soporte a las propuestas teóricas planteadas anteriormente. Se pretende así comprobar la validez tanto cualitativa como cuantitativa de las propuestas de optimización expuestas, aplicándose a entornos reales significativos con protocolos actuales.

Las simulaciones se han realizado con el simulador de nivel de red NS-2 [12]. En particular se presentan las simulaciones realizadas con el protocolo de micro-movilidad HAWAII [13] y el protocolo de señalización de calidad de servicio RSVP en el escenario mostrado en la figura 1.

Entre los posibles escenarios de red de acceso hemos seleccionado uno que corresponde a una empresa pequeña típica. Es una topología de árbol básica. Proporciona un modelo inicial para la prueba de las mejoras así como otros protocolos de nivel de red, y se utilizó, entre otros, para la definición de la arquitectura mostrada en el proyecto BRAIN [14]. La topología se ha diseñado de tal forma que permita la existencia de diferentes distancias desde los routers de acceso al router de divergencia[3] (uno, dos y tres saltos).

La red de acceso se compone de routers de acceso con interfaces inalámbricos y routers intermedios que conectan los distintos routers de acceso. Uno de estos routers intermedios actúa como puerta de enlace a otras redes. Los enlaces de la red de acceso están caracterizados como enlaces de 512 Kbytes de ancho de banda y 10 ms de retardo (en cada sentido). Hay que observar que el retardo depende en gran manera de la tecnología de red utilizada, de manera que este valor puede cambiar.

Para los nodos inalámbricos se ha utilizado la tecnología HIPERLAN/2 [15]. Dado que el simulador ns-2 carece de soporte para esta capa de enlace, se ha modelado basándose en las simulaciones llevadas a cabo por por Nokia usando simuladores de nivel de enlace en el marco del proyecto BRAIN. Utilizamos una aproximación para cada celda usando una sala industrial sin paredes internas. Nokia ha evaluado el comportamiento del interfaz de aire HIPERLAN/2 para diferentes niveles de carga. El tamaño de la sala es de 250 metros cuadrados y tiene un nivel de carga similar a 5 nuevas transferencias FTP por segundo.

En el simulador hemos caracterizado los enlaces HIPERLAN/2 mediante dos enlaces simplex (de subida y bajada) con dos parámetros: retardo, ancho de banda. Para el nivel de carga seleccionado en nuestras simulaciones y según las simulaciones de

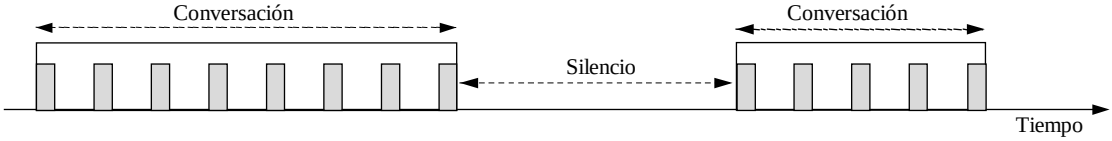


Figura 2: Tráfico de Voz sobre IP (VoIP)

---

[2] Comocidos como *seamless handover*.

Nokia, éstos parámetros corresponden a 3.2 Mbps de ancho de banda y a 15 ms de retardo.

Situamos el nodo emisor fuera de la red de acceso. Tan sólo se encuentra a un salto del gateway aunque podría haberse situado en cualquier lugar de Internet. El nodo emisor envía tráfico de voz sobre IP (VoIP) hacia el nodo móvil en el interior del dominio. Consideraremos el caso en el que el nodo móvil cambia suposición entre dos celdas consecutivas mientras se está realizando la comunicación tal y como lo muestra la figura 1.

La simulación se ha llevado a cabo de la siguiente manera: durante los 100 primeros segundos el emisor realiza la reserva con RSVP y comienza a transmitir paquetes de voz hacia el nodo móvil. Posteriormente se produce el handover entre dos celdas consecutivas. Esto implica la modificación de las tablas de rutas del protocolo HAWAII y la necesidad de establecer la reserva a través de la nueva ruta. En nuestro estudio sólo hemos contemplado el caso de handover planeado (aquel en el que el nodo móvil es consciente de que se va a producir el handover, y por tanto, puede reaccionar). El nodo móvil cambia su punto de acceso a la red y durante 20ms ambos enlaces inalámbricos están activos. Bien es cierto que, a pesar de ser un handover planeado, la cantidad de tiempo en el que ambos enlaces están activos es bastante corto, lo cual tiene su impacto sobre el rendimiento.

Hemos introducido tráfico interferente en los enlaces que comunican el nodo de divergencia y los routers de acceso involucrados en el handover (en la figura 1 aparecen más gruesos). Este tráfico interferente utiliza el 100% de los citados enlaces y nos permite observar el efecto de las reservas de RSVP cuando deben reinstalarse a lo largo del nuevo camino. Observar que el tráfico interferente sólo ocupa un enlace del camino de la nueva ruta y que éste enlace es fijo. Por otro lado nos permite comparar la mejora de rendimiento obtenida cuando el protocolo de micro-movilidad se acopla con RSVP, y podemos observar la ventaja de reservar cierto ancho de banda para los paquetes de señalización de QoS. El tráfico interferente se caracteriza como tráfico con ancho de banda constante.

El modelo de tráfico de VoIP extraído de [14] se puede describir como un proceso de nacimiento-muerte con un modelo de recepción basado en una distribución de Poisson y una duración de llamada con distribución exponencial. Durante la conversación cada participante está o bien hablando o bien en silencio. Hemos simulado este tráfico de manera que los periodos de actividad y silencio son generados con una variable aleatoria con distribución exponencial. El valor medio de esta variable es igual a T_on durante los periodos de actividad y a T_off en los periodos de silencio. La figura 2 muestra un ejemplo gráfico de este modelo.

Los parámetros principales principales del modelo de tráfico de VoIP son los siguientes:

- Intervalo de actividad: 50 %
- Duración media de llamada: 120 seg.
- Duración media de la fase de actividad T_on: 3 sec
- Duración media de la fase de silencio T_off: 3sec
- Tamaño de los datos del paquete IP: 32 Bytes. Los paquetes son de tamaño fijo.
- Ratio de transmisión: 12.2 Kbps

Realizamos las medidas de retardo de los paquetes de VoIP tan sólo en un sentido. Esta aproximación es correcta dado que los enlaces tienen colas diferentes para ambos sentidos. Los paquetes que llegan del nodo emisor no interfieren con los paquetes que salen del nodo móvil, así que el retardo obtenido al medir tan sólo un sentido es significativo.

Las simulaciones se han realizado usando la versión del simulador 2.1b5. Más detalles obre la implementación de los protocolos y de los generadores de tráfico pueden encontrarse en [14].

3.1. Resultados de la simulación.

En esta sección mostraremos el rendimiento de HAWAII y RSVP en el escenario comentado anteriormente, cuando éstos protocolos están acoplados de manera 'débil' y cuando están desacoplados. En ambos casos hemos reservado

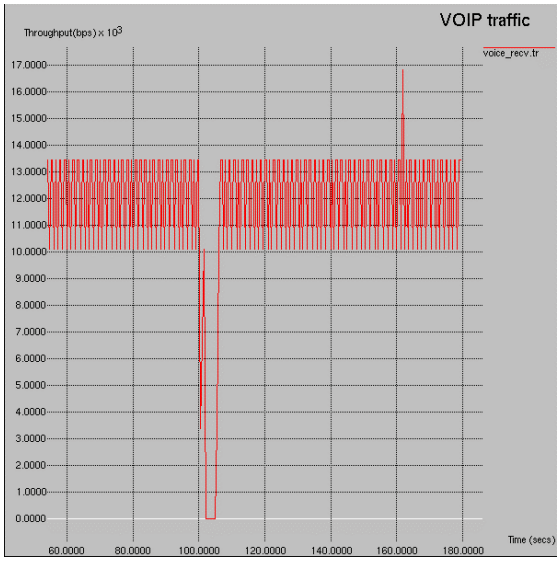


Figura 3: Rendimiento del tráfico VoIP en el caso desacoplado.

[3] Conocido como *crossover router* en inglés. Es el router común más cercano a las rutas antigua y nueva tras un handover.

una cantidad de ancho de banda fija para los paquetes de señalización de RSVP, tal y como se indica en la sección 2.4. Se añadió una cola WFQ (Weighted-Fair Queuing) para evitar la pérdida de paquetes de RSVP y por tanto favorecer la reparación rápida de las reservas tras el handover. La cantidad de ancho de banda reservada responde a la formula *n**s*8/30 (donde *n* es el número de sesiones que van a instalarse en el enlace y *s* es el tamaño medio esperado del paquete. La formula representa 1/30 de todo el ancho de banda necesario para los paquetes RSVP en caso de que fueran refrescados cada Segundo: para un tiempo de refresco de 3 segundos supone el ancho de banda para acomodar el 10% de todos los paquetes RSVP). Este valor debería incrementarse si existiesen numerosos cambios en las reservas. Considerando un valor promedio del paquete de 100 bytes y 30 sesiones por celda obtenemos un valor de 750 bps en el enlace inalámbrico. Para la parte de la troncal que tienen en común ambos enlaces reservaremos 1500 bps (por agregación).

En las siguientes figuras es necesario tener en cuenta que el handover ocurre en el segundo número 100.

Caso desacoplado.

Este caso muestra el rendimiento de los protocolos HAWAII y RSVP cuando no tienen constancia de la existencia del otro.

La figura 3 muestra el rendimiento del tráfico VoIP en el caso desacoplado. Cuando se produce el handover (segundo 100), la nueva ruta tan sólo tiene reserva hasta el router de divergencia y el tráfico interferente, que es mucho mayor en proporción que el tráfico de VoIP, no permite que los paquetes de voz lleguen al terminal, de manera que es necesario esperar hasta que el refresco de RSVP se produce para que la nueva reserva se instale a lo largo de todo el camino. A los 105

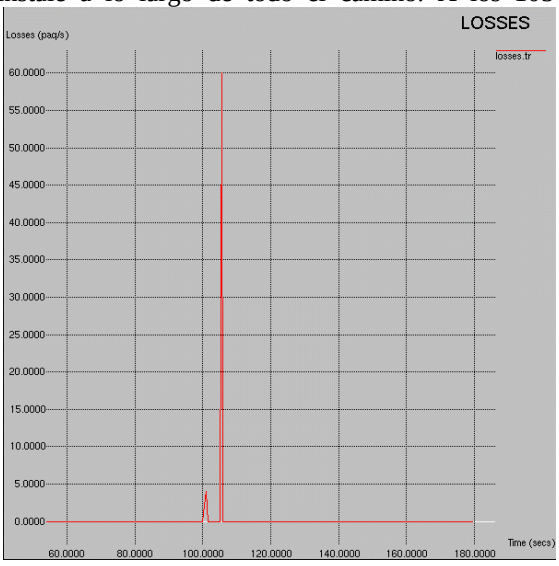


Figura 4: Paquetes de VoIP perdidos por segundo en el caso desacoplado.

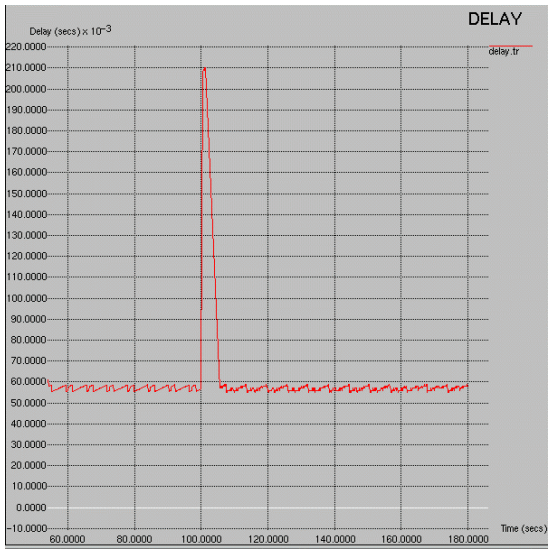


Figura 5: Retardo de los paquetesde VoIP en el caso desacoplado

segundos aproximadamente la nueva reserva es instalada y de nuevo los paquetes de VoIP llegan al terminal móvil, de manera que el rendimiento vuelve a la tasa sostenida previa al handover.

Como podemos ver en la figura 4, los paquetes por segundo perdidos se disparan hasta que la nueva reserva es instalada. Hay que tener en cuenta que en ésta figura se miden los paquetes *por segundo*, y no los acumulados. Tras el handover llegan a perderse hasta 60 paquetes por segundo lo que implica que la llamada se ve seriamente afectada debido al movimiento del terminal. La ausencia de paquetes perdidos durante los dos picos es el resultado del propio patrón de tráfico de VoIP: simplemente no hay tráfico emitido en ese momento por lo que no hay pérdida de paquetes.

Como consecuencia del handover, los paquetes de VoIP que no se pierden sufren un gran retardo durante un tiempo como muestra la figura 5. La

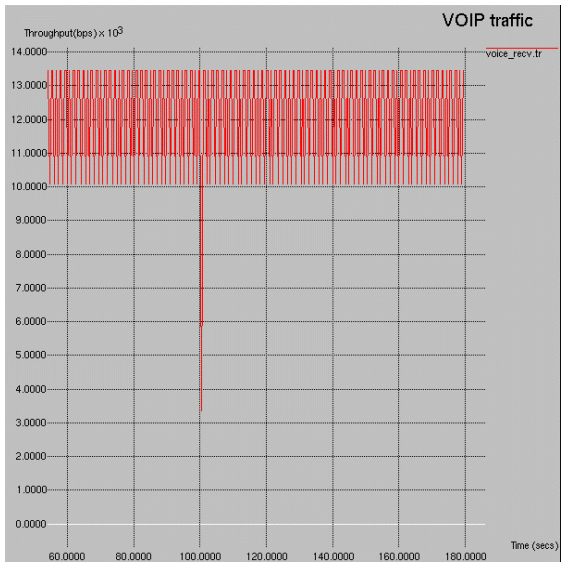


Figura 6: Rendimiento del tráfico VoIP en el caso acoplado.

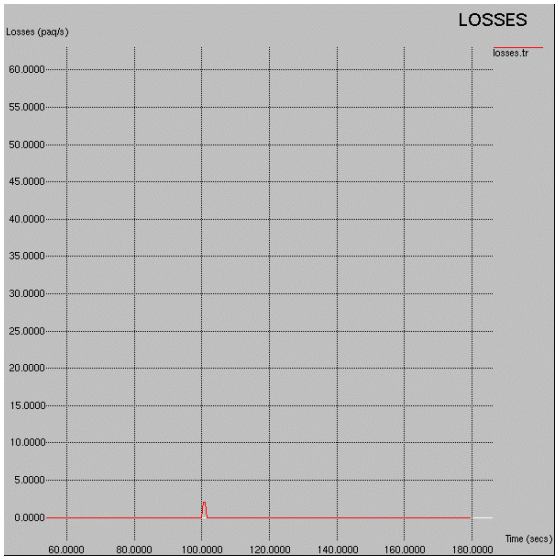


Figura 7:Paquetes perdidos por segundo de VoIP en el caso acoplado.

topología es simple por lo que podemos inferir que la causa del retardo es la ya comentada: la ausencia de reserva una vez que la nueva ruta se ha establecido. El enlace está saturado la cola de best-effort está llena, y los paquetes sufren un retardo proporcional a la longitud de la cola y, como hemos visto, algunos se pierden.

Caso acoplado ('débil')

Este caso es idéntico al anterior salvo por el hecho de que los protocolos HAWAII y RSVP están acoplados de manera débil como hemos visto en la sección 2. Hemos diseñado un mecanismo de acoplamiento de ambos protocolos de tal forma que intercambian información en el handover. En el mismo instante que la nueva ruta es estable el agente RSVP procede a la reparación de RSVP. Así aseguramos que la reserva se instala lo antes posible.

La figura 6 confirma nuestra hipótesis. El rendimiento del tráfico de VoIP tras el handover se ve afectado pero el impacto es mucho menor que en el caso desacoplado. La figura 7 muestra que las pérdidas de paquetes de VoIP durante el handover se han minimizado. Tan sólo 3-4 paquetes se han perdido, debido al handover propiamente dicho. El resto de pérdidas producto de la ausencia de reserva en la nueva ruta se ha eliminado dado que los mensajes de refresco de la reserva se envían tan pronto como la nueva ruta es estable, así que el impacto del tráfico interferente es mínimo.

Por último, la figura 8 muestra el impacto del handover en el retardo. A pesar de que el retardo máximo no se reduce, el intervalo de tiempo en el que los paquetes se ven afectados sí disminuye considerablemente (comparar con la figura 5). Los únicos paquetes que se ven afectados y que tienen un retardo elevado son aquellos que estaban en vuelo en el momento en el que ocurre el handover. Estos paquetes deben esperar a que se actualice la nueva ruta y por lo tanto sufrirán un retardo dependiendo del rendimiento del protocolo de movilidad local subyacente. El acoplamiento solamente optimiza la reinstalación de las reservas para minimizar el impacto del handover. Esto es independiente del mecanismo de movilidad usado, por lo que no afecta al tiempo que tarda la nueva ruta en ser establecida.

4. Conclusiones

Presentamos una mejora para el funcionamiento de las reservas en entornos móviles basada en la colaboración de los protocolos de QoS y movilidad. Aunque se pueden concebir distintos tipos de colaboración hemos optado por el acoplamiento 'débil' como el más prometedor, donde los protocolos de QoS y movilidad intercambian información mediante algún mecanismo de disparo cuando ocurre un handover. Hemos observado mediante las simulaciones que este acoplamiento proporciona una ventaja clara en algunos escenarios. A pesar de que el propio handover en sí no puede ser acelerado sí que se consigue que las reservas se reinstalen tan pronto como el nuevo camino es estable. Esto es especialmente interesante en escenarios como el aquí mostrado, donde tráfico interferente puede hacer que el tráfico con reserva pueda ser descartado. Hemos simulado también otro mecanismo complementario como la priorización de los paquetes de señalización que ofrecer un marco para el seamless handover cuando se utilizan mecanismos basados en reservas.

En conclusión, el acoplado de los protocolos de micro-movilidad y protocolos de reserva tiene un coste reducido y las ventajas, al menos en el caso de protocolos de movilidad salto a salto (como es el caso de HAWAII, o Cellular IP [16]), combinado con la pre-reserva para la señalización de QoS son suficientes como para justificarlo.

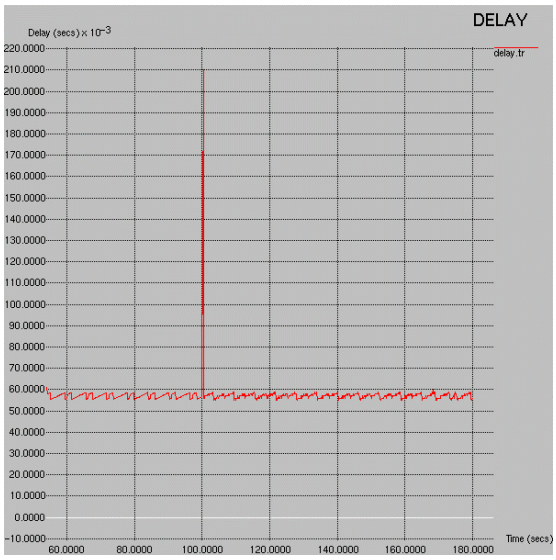


Figura 8: Retardo de los paquetes de VoIP en el caso acoplado

Agradecimientos

This work has been performed in the framework of the IST project IST-1999-10050 BRAIN, which is partly funded by the European Union. The authors would like to acknowledge the contributions of their colleagues from Siemens AG, British Telecommunications PLC, Agora Systems S.A., Ericsson Radio Systems AB, France Télécom – R&D, INRIA, King's College London, Nokia Corporation, NTT DoCoMo, Sony International (Europe) GmbH, and T-Nova Deutsche Telekom Innovationsgesellschaft mbH.

Alberto López agradece en particular a la Fundación Séneca (Comunidad Autónoma de la Región de Murcia) su apoyo y la financiación para llevar acabo este trabajo.

Referencias

[1] Braden, R., Clark, D. Shenker, S., "Integrated Services in the Internet Architecture: an Overview ". Internet Engineering Task Force, Request for Comments (RFC) 1633, Junio 1994.

[2] Braden, R., et. Al. "Resource ReSerVation Protocol (RSVP) -- Version 1, Functional Specification". Internet Engineering Task Force, Request for Comments (RFC) 2205, Septiembre 1997.

[3] Perkins, C., "IP Mobility Support". Internet Engineering Task Force, Request for Comments (RFC) 2002, Octubre 1996.

[4] Gustafsson, E., Jonsson, A., Perkins, C,. "Mobile IP Regional Registration", Internet Draft (work in progress), Marzo 2001.

[5] Lee, S.B., Gahng-Seop, A., Zhang, X., and A.T. Campbell, "INSIGNIA: An IP-Based Quality of service Framework for Mobile Ad Hoc Networks", Journal of Parallel and Distributed Computing (Academic Press), Special issue on Wireless and Mobile Computing and Communications, Vol. 60 No.4. p.374-406, Abril 2000.

[6] B. Carpenter, "Architectural Principles of the Internet," Internet Engineering Task Force, Request for Comments (RFC) 1958, Junio 1996.

[7] M. Carlson, W. Weiss, S. Blake, Z. Wang, D. Black, and E. Davies, "An Architecture for Differentiated Services", Request for Comments (RFC) 2475, Deciembre 1998.

[8] Floyd, S., Jacobson, V., "Link Sharing and Resource Management Models for Packet Networks", IEEE/ACM Transactions on Networking, Vol. 3, No. 4, 365-386, Agosto 1995.

[9] A. Talukdar, B. Badrinath, A. Acharya, MRSVP: A Resource Reservation Protocol for an Integrated Services Packet Network with Mobile Hosts, In *Proceedings of ACTS Mobile Summit'98*, Junio 1998.

[10] MSc Thesis "Towards Full Quality of Service Support in a Mobile, Wireless Internet", Anne-Louise Burness, University of Essex, Enero 2001.

[11] IETF Seamoby WG. http://www.ietf.org/ html.charters/seamoby-charter.html

[12] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu. "Advances in network simulation.", IEEE Computer, 33(5):59(67), Mayo 2000.

[13] Ramjee, R., La Porta, T., Thuel, S., Varadhan, K., Wang, S.Y., "HAWAII: a domain-based approach for supporting mobility in wide-area wireless networks". Proceedings of the Seventh International Conference on Network Protocols (ICNP) 1999, pp. 283 – 292.

[14]IST-1999-100050 BRAIN D2.2 "BRAIN architecture specifications and models, BRAIN functionality and protocol specification", Marzo 2001.

[15] ETSI TR 101 683 V1.1.1 (2000-02), "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; System Overview".

[16] Campbell, A., Gomez, J., Kim, S., Valko, A., Chieh-Yih Wan, Turanyi, Z., "Design, implementation, and evaluation of cellular IP". IEEE Personal Communications, Vol. 7, Agosto 2000, pp. 42-49..

Análisis de las búsquedas realizadas, categorías accedidas y documentos vistos en un directorio Web

Montse Ponte, Fidel Cacheda, Ángel Viña
Departamento de Tecnoloxías da Información e das Comunicacións
Facultad de Informática, Universidad de A Coruña
Campus de Elviña s/n, 15071 A CORUÑA, SPAIN
Teléfono: 981 167000 Fax: 981 167160
Email: {fidel,avc}@udc.es, montse@mail2.udc.es

Abstract.

In this paper we present an analysis of the behaviour of the users in a Web directory, using the transaction log of a Spanish Web directory, with more than 320,000 requests. This study focuses on showing an analysis of searches, categories browsed by the users and documents viewed. Furthermore we have identifyed the users' sessions, lending us obtain data so as: number of queries per session, number of categories visited per session, number of documents viewed per session, and session life. The results indicate the access to the Web directory is done, firstly, to search, the queries are still made short, and the users only check the results of the first screens. Finally, we have developed a statistical analysis to achieve the statistical distribution of the most important parameters in the sessions: number of queries, number of categories, and number of documents through time. We can observe that in most of the cases weren't fitted any statistical distribution.

1. Introducción

Como consecuencia del crecimiento del World Wide Web, fue necesario desarrollar buscadores para facilitar la búsqueda de información a los usuarios de la Web. Esto ha ido acompañado de la aparición de diversos estudios que se centran sobre todo en el análisis del comportamiento de los usuarios ante el buscador, pero ninguno entra a analizar las categorías visitadas o los documentos vistos. Así tenemos el artículo realizado por Kirsch [1] que presenta estadísticas del uso de Infoseek, donde algunas de sus conclusiones, "la mayoría de las búsquedas de Internet son pequeñas y simples", serán ratificadas en todos los estudios posteriores [2] [3] [4] [5]. De nuevo coincidirán Jansen [2] y Silverstein [4] en sus resultados al afirmar que los usuarios de búsquedas en la Web difieren significativamente de los usuarios de los tradicionales sistemas IR (Information Retrieval), y por tanto que el diseño de los buscadores de la Web tendrá que tener en cuenta estas diferencias. Jansen realizó su estudio con los ficheros log del buscador Excite, y Silverstein con los obtenidos de AltaVista.

Nosotros nos centraremos en las búsquedas, pero no olvidaremos que los internautas que usan un directorio en Internet pueden acceder también a las categorías o a los documentos.

Además de realizar estudios de las búsquedas, categorías y documentos, se identificó las sesiones analizando estadísticas como: media de duración de sesión, y media del número de búsquedas, categorías o documentos por sesión, así como de la distribución de los parámetros relevantes de las sesiones. Son muy pocos los estudios encontrados con los que comparar nuestros resultados de las sesiones. En el análisis de Jansen [2] y Silverstein [4] se presentan datos del número de búsquedas por sesión, pero no ofrecen ninguna información sobre la duración de las sesiones.

El artículo se estructura de la siguiente forma: Sección 2 describe el entorno de BIWE y las peticiones del fichero log. Sección 3 contiene el análisis de las búsquedas, categorías, documentos y sesiones. Sección 4 el estudio de la distribución de los parámetros de las sesiones. Concluyendo en la sección 5.

2. El entorno

El estudio se realizó sobre el directorio Web español BIWE. En la siguiente sección describiremos BIWE (http://www.biwe.es), y los ficheros log de transicción donde se almacenan todas las peticiones realizadas sobre el directorio Web.

2.1 El portal BIWE

Un directorio Web es una taxonomía jerárquica que clasifica la información que puede encontrar en el World Wide Web [6] y, en nuestro caso, restringe la información escrita en español.

BIWE permite realizar búsquedas sobre la información que se desea, o visitar categorías o documentos. El acceso a las categorías y documentos se puede realizar directamente o como resultado de una búsqueda.

Hay dos modos de realizar una búsqueda: búsqueda simple y búsqueda avanzada. La simple es el sistema más rápido de hacerlo, consiste en una colección de palabras donde aplicamos el operador OR sobre ellas para realizar la búsqueda, pero los resultados que aparecen en los primeros lugares resultan de aplicar el operador AND a las palabras. Además de los operadores lógicos se dispone de otros operadores más simples: -término indica al motor de búsqueda que ignore todos los documentos que contengan la palabra término; +término indica al motor de búsqueda que ignore todos los documentos que no contengan la palabra término. El operador " es usado para encerrar frases, y el motor de búsqueda ignorara a los documentos que no contengan esa frase.

Los operadores lógicos que se pueden utilizar para realizar la petición son AND, OR, NOT, y sus versiones españolas, y, o, no. Así mismo se pueden utilizar los paréntesis para combinar los distintos operadores.

En la búsqueda avanzada es posible personalizar las características de las búsquedas permitiendo definir:

1- En que campo de la entrada deben aparecer las condiciones de búsqueda. Activando una casilla se puede elegir una o varias de las siguientes posibilidades: palabras clave, título, descripción o URL.

2- El criterio por el que se agrupan los resultados (por categoría, sin agrupar pero con información de categoría, sin agrupar y sin información de categorías).

3- El criterio para ordenar los resultados (alfabéticamente, número de accesos que haya tenido cada página, según la fecha de alta).

4- Número de resultados que se mostrarán en cada intervalo. En BIWE por defecto el número de resultados por página es 10.

Además se puede restringir la búsqueda a algunas categorías.

Después de que una cadena de búsqueda es introducida y los términos de la búsqueda son procesados, BIWE retorna una pantalla completa con 10 URLs (o el número previamente seleccionado por el usuario) mostrando el título y la descripción de cada URL. El orden de aparición de todos los URLs depende de su importancia en la búsqueda. El usuario puede hacer clic sobre alguno de los URL para explorar la página Web asociada. Además el internauta puede hacer clic sobre los botones de navegación para explorar otras pantallas con más resultados, esta acción producirá otra búsqueda en el motor de búsqueda.

2.2 Los ficheros log de BIWE

El registro de log transaccional de BIWE se realiza sobre dos tipos de fichero. El fichero log transaccional que almacena las peticiones realizadas, y el fichero log del servidor Web que almacena la dirección IP del cliente y la acción ejecutada.

En el fichero log transaccional, según el tipo de petición (realización de una búsqueda, acceso a una categoría o a un documento) la información almacenada es distinta. En el caso de las búsquedas los campos relevantes son:

- La **cadena de búsqueda** introducida por el usuario.
- Los **términos de la búsqueda**, eliminando las palabras reservadas, operadores, y nexos.
- **Timestamp** que indica el día y la hora en la que se realizo la búsqueda.
- El número del primer documento que aparece en la página de resultados.
- El número de documentos que se muestran en la página de resultados.
- Campo booleano que indica si la búsqueda se restringe a una categoría, y en ese caso se guarda el identificador de la categoría.
- Los campos de los documentos donde se realiza la búsqueda.
- El tipo del grupo del documento, si lo hay.

En las categorías tenemos:

- El **identificador de categoría**, valor único para cada uno de las categorías del directorio.
- **Timestamp** que indica el día y la hora en la que se visitó la categoría.

Y en los documentos:

- El **identificador de documento**, valor único para cada uno de los documentos del directorio.
- **Timestamp** que indica el día y la hora en la que se accedió al documento.

En el fichero log del servidor web se almacenan las direcciones IP de cada uno de los usuarios del portal, junto con las acciones que llevan a cabo. Esto es lo que permite identificar las sesiones de los usuarios.

3. Análisis de las peticiones

Para la realización de nuestro estudio se utilizaron 324,503 peticiones recogidas de los ficheros log del portal BIWE durante 16 días, desde el 3 Mayo 2000 a las 3:00 hasta el 18 Mayo 2000 a las 7:00. Como muestra la tabla 1, del conjunto de peticiones recogidas un 32.60 % corresponden a búsquedas, un 18.81 % a categorías y un 48.59 % a documentos.

	Número total	Porcentaje
Búsquedas	105,786	32.60 %
Categorías	61,050	18.81 %
Documentos	157,667	48.59 %
Total de Peticiones	324,503	

Tabla 1: Peticiones realizadas

3.1 Análisis de las búsquedas

Utilizando las 105,786 peticiones que se corresponden con búsquedas realizamos la investigación. Un tema que se ha estudiado desde los primeros análisis de las búsquedas, es la longitud de estas. En la tabla 2 vemos el alto porcentaje de búsquedas con 1 y 2 términos, y tan solo el 1,527 % de las búsquedas tienen más de 6 términos.

Los usuarios siguen utilizando búsquedas pequeñas como ya mencionó [2] [4], con una media de palabras por búsqueda de 1.63, valor un poco más pequeño que los proporcionados por Jansen [2] y Silverstein [4].

La tabla 3 refuerza la idea de que los usuarios de los buscadores emplean peticiones pequeñas y simples. Los distintos operadores que le proporciona el buscador son muy poco utilizados.

Nº palabras	Frecuencia	Porcentaje
0	2,151	2.051 %
1	51, 045	48.662 %
2	39,071	37.247 %
3	9,717	9.263 %
4	2,022	1.928 %
5	506	0.482
6	227	0.216 %
7	59	0.056 %
8	59	0.056 %
9	18	0.017 %
10	7	0.007 %
11	3	0.003 %
12	6	0.006 %
13	3	0.003 %
19	1	0.001 %
25	1	0.001 %

Tabla 2: Número de términos por búsqueda

Un pequeño porcentaje de búsquedas usan OR, O, AND, NOT, NO tan solo el 0.637 % de las 105,786 búsquedas que se realizaron durante los 16 días estudiados. El que presenta un mayor porcentaje es el operador AND español, Y, haciéndonos pensar que los usuarios tienden a usar los operadores lógicos de su lengua materna [5]. En los otros operadores lógicos (OR, NOT) también se observa un mayor uso

Operador español AND: Y	3.539 %
Operador AND	0.266 %
Operador español NOT: NO	0.016 %
Operador NOT	0.007 %
Operador español OR: O	0.033 %
Operador OR	0.018 %
Paréntesis	2.94 %
Operador +	1.962 %
Operador -	0.862 %
Operador “	2.471 %

Tabla 3: Utilización de operadores lógicos

de sus versiones españoles (O, NO). Esto seguramente, debido a su pobre conocimiento de la lógica booleana.

Un aspecto importante es la frecuencia de las búsquedas y cuánto se repiten. En la tabla 4 se muestra parte del histograma del número de veces que se realiza una búsqueda. Si estudiamos más detalladamente esta tabla, con la parte izquierda de la tabla 5, se deduce que hay una pequeña cantidad de búsquedas que se repiten muchas veces, al mismo tiempo que hay una cola de búsquedas distintas que se han realizado una única vez. Es importante destacar que la mitad de todas las búsquedas es solamente el 8 % de las cadenas más buscadas. Es interesante destacar que la cadena más buscada fue la cadena vacía (2173 veces), lo que incita a pensar que las interfaces gráficas de los buscadores en Internet no son intuitivas ni fáciles de utilizar para sus usuarios.

También nos interesó conocer los términos que se buscan en el directorio. La parte derecha de la tabla 5 presenta las 23 palabras más populares, observando con pena el pobre uso que hacen los internautas de los buscadores, los temas más buscados no son precisamente sobre información cultural o tecnológica.

BIWE, como explicamos en la Sección 2, muestra 10 resultados de la búsqueda por página. Examinando las páginas a las que acceden los usuarios vemos en la tabla 6, que el 67.881 % solo miran en la primera página. ¿Estarán ellos satisfechos con los resultados ofrecidos en la primera página?. Sea cual sea la respuesta, en cualquier caso lo que implica esto es, la necesidad de disponer de algoritmos IR de alta precisión para la Web [2], pues los primeros resultados que se muestran son los que son examinados.

Número de veces solicitada una búsqueda	Porcentaje
1	20,27 %
2	11,25 %
3	7,58 %
>3	60,9 %

Tabla 4: Porcentajes del número de veces que se repite una búsqueda

Páginas vistas	Porcentaje
1	67.881 %
2	13.234 %
3	5.966 %
4	3.468 %
5	2.272 %
6	1.538 %
7	1.119 %
8	0.819 %
9	0.598 %
10	0.49 %
Más de 10	2.615 %

Tabla 6: Páginas de resultados vistas por los usuarios

Cadenas buscadas	Frecuencia	Porcentaje	Términos	Frecuencia	Porcentaje
	2173	2.054 %	sexo	2016	1.164 %
sexo	814	0.769 %	gratis	1857	1.073 %
gran hermano	663	0.627 %	fotos	1244	0.719 %
mp3	651	0.615 %	mp3	1190	0.687 %
gay	582	0.55 %	gay	1150	0.664 %
porno	311	0.294 %	denudas	939	0.542 %
relatos eroticos	263	0.249 %	porno	860	0.497 %
chat	262	0.248 %	gran	811	0.468 %
hentai	249	0.235 %	hermano	771	0.445 %
famosas desnudas	237	0.224 %	madrid	745	0.43 %
famosas	221	0.209 %	famosas	674	0.389 %
moviles	215	0.203 %	com	629	0.363 %
caratulas	212	0.2 %	videos	617	0.356 %
sexo gratis	203	0.192 %	moviles	584	0.337 %
tetas	192	0.181 %	relatos	555	0.321 %
hacker	188	0.178 %	hentai	494	0.285 %
voyeur	180	0.17 %	juegos	458	0.265 %
lolitas	175	0.165 %	chat	456	0.263 %
fotos	172	0.163 %	barcelona	434	0.251 %
amateur	171	0.162 %	caratulas	424	0.245 %
chicas	168	0.159 %	musica	422	0.244 %
zoofilia	167	0.158 %	desnudos	383	0.221 %
contactos	164	0.155 %	eroticos	370	0.214 %

Tabla 5: Las 23 cadenas de búsquedas y términos más populares

3.2 Análisis de las categorías

Los internautas que entran en el directorio en Internet pueden acceder a las categorías directamente, o como resultado de una búsqueda. Todos visitan la categoría 0 al entrar en BIWE, pero ésta se ha incluido en el análisis viendo que su presencia o ausencia no influía apenas en los resultados.

Nº de veces visitada una categoría	Porcentaje
0	1.563 %
1	2.567 %
2	1.897 %
3	4.018 %
4	4.129 %
5	2.790 %
6	4.464 %
7	1.786 %
> 7	76.786 %

Tabla 7: Porcentajes del número de visitas de las categorías

En la tabla 7 se muestra información sobre el número de veces que fueron accedidas las distintas categorías, y como esperamos son muy pocas las categorías sin ninguna visita. En el caso de las categorías que no presentan ningún acceso se podría plantear la posibilidad de reubicarlas o agruparlas en otras categorías, si el estudio se hubiese realizado en un período de tiempo mayor.

3.3 Análisis de los documentos

Lo mismo que en las categorías, el acceso a los documentos se debe a que los internautas van a ellos navegando a través de las categorías o como resultado de una búsqueda. En la tabla 8 se muestra información sobre las veces que los documentos fueron visitados. No incluimos los documentos que no fueron visitados ninguna vez porque no aporta ninguna información, el número de documentos que proporciona BIWE es elevadísimo, por tanto lógicamente habrá muchos documentos que nunca fueron visitados durante esos 16 días.

Nº de veces visitado un documento	Porcentaje
1	44.512 %
2	18.772 %
3	9.650 %
4	5.736 %
5	3.824 %
6	2.722 %
7	2.110 %
8	1.661 %
9	1.246%
10	0.961 %
11	0.857 %
> 11	7.949 %

Tabla 8: Porcentajes del número de veces que se visitan los documentos

En los datos obtenidos observamos porcentajes no muy dispares con los encontrados en el número de ocurrencias de las búsquedas, ya que los usuarios accederán a los documentos que buscan, y como se ha visto anteriormente hay un gran porcentaje de búsquedas que se repiten, lo que implica también que se repetirán los accesos a los mismos documentos.

4. Análisis de las sesiones

Una sesión es el conjunto de peticiones que se asocia a un usuario en un intervalo de tiempo. En nuestro caso, para identificar las sesiones de usuario utilizamos las direcciones IP almacenadas en los ficheros log del servidor Web, y se ha considerado que las peticiones de un mismo usuario espaciadas en un tiempo superior a los 30 minutos formaran parte de sesiones distintas. Silverstein en su análisis sobre Altavista [4], utilizó un gap de 5 minutos para identificar las peticiones de una sesión, pero nosotros hemos considerado que ese tiempo es muy poco representativo de la realidad, para la iteración de un usuario con un directorio Web. Para los temas tratados en su estudio, no es una mala elección.

El número de sesiones identificado a partir de los ficheros log de los 16 días fueron 57,529. En cuanto a la duración de las sesiones como vemos en la tabla 10 la mayoría de las sesiones tiene una duración entre los 0 y 500 segundos (aproximadamente 8 minutos), solo el 2.624 % tiene una duración mayor de 1 hora.

En cuánto, lo que hacen los usuarios en el buscador, los resultados nos dicen que la media de búsquedas por sesión es 1.7037, ligeramente inferior que en el caso de [2] y [4], por tanto los internautas que acceden al directorio en Internet tuvieron que hacer menos modificaciones en sus búsquedas que los de Excite y Altavista, debido seguramente a una mayor eficiencia de los algoritmos de búsqueda IR de la Web (los estudios de Jansen y Silverstein fueron realizados en el año 98 y 99 respectivamente). La media de categorías por sesión fue 1.7622 y en los documentos 2.4606.

Recordemos que las categorías y documentos pueden ser visitados como resultado de una búsqueda, por tanto estos datos nos dan una idea del número de resultados que miran los internautas después de una búsqueda (aproximadamente 5 resultados son los visitados), y que el acceso de los internautas al directorio es principalmente para realizar alguna búsqueda.

La media de búsqueda por sesión nos lleva a la conclusión, que en la mayoría de los casos las búsquedas son modificadas, seguramente porque los resultados no son los esperados.

Número de sesiones	57,529
Media de duración de sesión	570.529 sg
Búsquedas por sesión	1.7037
Categorías por sesión	1.7622
Documentos por sesión	2.4606

Tabla 9: Resumen de los datos de las sesiones

Duración de Sesiones (en sg.)	Porcentaje
0 – 500	72.131 %
500 – 1000	10.878 %
1000 – 1500	6.315 %
1500 – 2000	3.940 %
2000 – 2500	2.182 %
2500 - 3000	0.943 %
3000 – 3500	0.576 %
3500 – 4000	0.411 %
Más	2.624 %

Tabla 10: Resumen de la duración de las sesiones en intervalos de 500 segundos

Tiempo entre búsquedas	204.48 sg.
Tiempo entre categorías	107.05 sg.
Tiempo entre documentos	193.13 sg.

Tabla 11: Medias de los tiempos entre las distintas peticiones

En cuánto a los datos de los tiempos entre las distintas peticiones vemos esta información en la tabla 11. La media del tiempo entre búsquedas, 3.4 minutos, nos indica el tiempo que un usuario tarda en mirar los resultados, comprobar que no son los deseados y por tanto realizar otra petición.

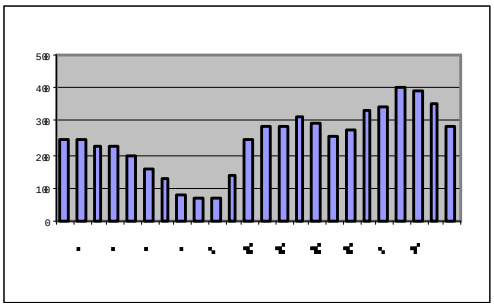


Figura 1: Número de sesiones por hora del día 4/5/2000

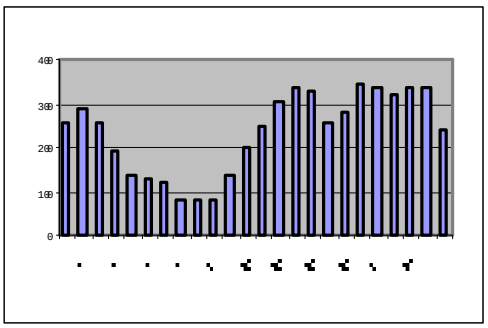


Figura 2: Número de sesiones por hora del día 17/5/2000

Las figuras 1 y 2 muestran las distribuciones de sesiones por hora a lo largo de dos días distintos, pero como podemos observar la situación es muy parecida en las dos. En ambas, el mayor número de sesiones se da en los rangos de 13 h a 15 h, y de 18 h a 22 h, y lo mismo sucede en los otros 14 días que se analizaron.

5. Distribución de los parámetros de las sesiones

A partir de los datos recogidos de los ficheros log durante los 16 días se calcularán los parámetros relevantes para las sesiones.

En muchos de ellos, duración de sesión, número de búsquedas por sesión, número de categorías por sesión, número de documentos por sesión, tiempo entre categorías, tiempo entre búsquedas, tiempo entre documentos, no se observo una tendencia muy definida hacia ningún tipo de distribución.

En los casos anteriores el estudio se realizo aplicando el test de Kolmogorov-Smirnov sobre cada variable, obteniendo distintos parámetros estadísticos, pero sin encontrar una distribución que se les ajustase.

Posteriormente analizamos el número de sesiones, el número de búsquedas, el número de categorías, y el número de documentos por unidad de tiempo. Se escogieron intervalos de 1 hora y de 10 minutos.

En el primer caso el test de Kolomogorov-Smirnov se aplicó a cada uno de los días, analizando el número de sesiones, búsquedas, categorías y documentos realizadas en 1 hora, obteniendo una Normal. En el segundo caso el test se aplico sobre grupo de 2 horas y analizando las mismas variables pero en intervalos de 10 minutos. Con la variable número de sesiones, los test realizados siguieron una Normal un 91.6 % de los grupos de 2 horas, frente al 25% que siguieron una Poisson, con las otras variables solo se acepto una Normal. Cada grupo de horas tiene parámetros distintos, es decir distinta media y desviación típica. En la siguiente figura se muestra el histograma de la variable número de categorías de uno de los grupos de horas.

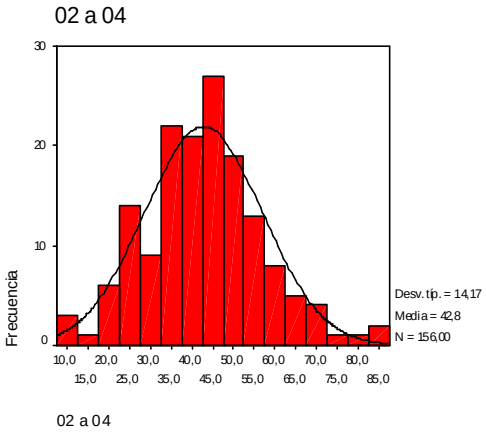
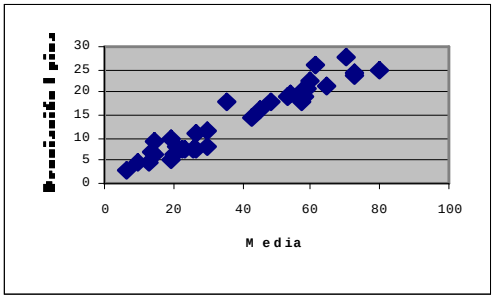


Figura 3: Histograma del nº de categorías en el rango de 2 h a 4 h



Una vez establecido la distribución de estas variables, se analizo si existía alguna relación entre los parámetros de la Normal (media y desviación típica), comprobando que existía una relación lineal entre ambos parámetros, como vemos en la figura 4.

Figura 4: Relación entre los parámetros de la Normal

6. Conclusiones

Se ha estudiado el comportamiento de los usuarios de un directorio Web, analizando las entradas de sus ficheros log.

Nuestros resultados corroboran muchas de las conclusiones alcanzadas en estudios anteriores:

- Los usuarios realizan búsquedas formadas por muy pocas palabras (de media 2 palabras), poco complejas y muy genéricas.
- Los usuarios tienden a consultar los resultados mostrados en las dos primeras pantallas.
- Los operadores booleanos son muy poco utilizados en las peticiones.
- Todos los usuarios tienden a buscar la misma información.

En el análisis de las sesiones se observó que los usuarios suelen tener sesiones cortas, 9 minutos, que el acceso al directorio de Internet se hace sobre todo para ejecutar búsquedas, y que de los resultados mostrados como consecuencia de la búsqueda suelen mirar 5.

Esto confirma la idea de que un usuario de un buscador en Internet es un usuario con poca experiencia y conocimientos en informática e Internet en general, y que se conecta para resolver una falta de información concreta y puntal.

En el análisis de la distribución matemática de los parámetros más relevantes de las sesiones, se observó que muy pocos se han ajustado a alguna de las distribuciones estudiadas, y los que lo han hecho han seguido una Normal.

Este trabajo pretende ser el primer paso para, en futuros trabajos, desarrollar herramientas que simulen el comportamiento de los internautas conectados a un directorio Web. De esta forma, se permitirá evaluar el rendimiento ofrecido a sus usuarios y conseguir sistemas de información más eficientes, ya que los actuales sistemas de evaluación basados en TREC o Web-TREC se centran especialmente en parámetros como la precisión y la exhaustividad de la búsqueda [8].

Referencias

[1] S. Kirsch, “Infoseek’s experiences searching the Internet”. SIGIR FORUM Fall 98.

[2] B. Jansen, A. Spink, J. Bateman, T. Saracevic, “Real Life Information Retrieval: A Study Of User Queries On The Web”. SIGIR FORUM Spring 98.

[3] B. Jansen, A. Spink, J. Bateman, T. Sarevic, “Searchers, the subjects they search, and sufficiency: A Study of a large example of Excite searchers”. World Conference on the WWW and Internet, Florida 1998.

[4] C. Sliverstein, M. Henzinger, H. Marais, M. Moricz. “Analysis of a Very Large Web Search Engine Query Log”. SIGIR FORUM Fall 99.

[5] F. Cacheda, A. Viña. “Experiencies retrieving information in the World Wide Web”, aceptado para ISCC 2001, 6th IEEE Symposium on Computers and Communications, 3-5 de Julio de 2001.

[6] R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, Ed. Addison Wesley, 1999.

[7] H. Leighton and J. Srivastava, *Precision among World Wide Web Search Services (Search Engines)*. Journal of the American Society for Information Science, 50(10): 870-881, 1999.

[8] D. Hawking, N. Craswell, P. Thistlewaite and D. Harman, *Results and challenges in web search evaluation*. WWW8, Toronto, pages 243-252. Elsevier, 1999

Caracterización de los enlaces de Internet utilizando tecnología de Redes Activas

Marifeli Sedano[1], Bernardo Alarcos[1], María Calderón[2] y David Larrabeiti[2]

[1] Área de Ingeniería Telemática.
Dpto. Automática
Escuela Politécnica. Universidad de Alcalá.
Carretera Madrid-Barcelona, Km 33,600 - 28871
Alcalá de Henares (Madrid)
E-mail: {marifeli, bernardo}@aut.alcala.es

[2] Dpto. de Ingeniería Telemática.
Universidad Carlos III.
Avd. de la Universidad, 30 – 28911
Leganés (Madrid)
E-mail: {maria, dlarra}@it.uc3m.es

Abstract. *This paper presents the design, implementation and trials of **a-clink**, which is a hop-by-hop performance estimation tool based on active networks. The paper begins by analyzing different alternatives of hop-by-hop performance estimation tools: pathchar, clink, pchar and nettimer. Based on this analysis, several deficiencies are identified on the different tools. In order to improve the efficiency and accuracy of the estimations, one of the tools is selected, clink, to design an extension based active network technology. This extension, a-clink, has been implemented over the public domain active network platform SARA. The implementation of a-clink has been trialed on a simple active network prototype spanning two universities connected through public Internet, and its results compared with those obtained by the original clink. The paper concludes describing the advantages of the active version of clink over the conventional passive performance estimation tool.*

1 Introducción

La topología densamente interconectada de Internet y la tecnología IP hacen de Internet una red bastante robusta, por lo tanto, es muy poco frecuente que se pierda la conectividad a través de la red. Sin embargo, el dinamismo y la complejidad de la red, tanto en la configuración de las rutas, como en la distribución del tráfico en múltiples enlaces, hace que las prestaciones sean difíciles de predecir y fuertemente variables.

En este entorno comienza a ser imprescindible disponer de herramientas de diagnóstico que sean capaces de determinar las características, en cuanto a prestaciones y posibles cuellos de botella, del camino entre un origen y un destino. Dicho análisis ayudará a los administradores y usuarios de la red a detectar dónde se encuentran los cuellos de botella que están causando situaciones indeseables, y en general a conocer las prestaciones que se pueden obtener para una determinada comunicación.

Entre las herramientas de diagnóstico más utilizadas en la actualidad podemos encontrar el *traceroute*. Dicha herramienta permite averiguar el número y dirección de los nodos intermedios por los que pasarán los paquetes en la comunicación entre un origen y un destino, proporcionando adicionalmente datos de retardo de tránsito desde el origen a cada uno de los nodos intermedios. Dicha herramienta fue diseñada con el objetivo principal de determinar el camino que siguen los paquetes en Internet.

En 1997 Van Jacobson ante los problemas de congestión que ya comenzaban a materializarse en la red, desarrollo una nueva herramienta de diagnóstico de redes: *pathchar* [1], que permite a un usuario determinar entre otras características el ancho de banda y retardo en cada salto entre un origen y un destino.

Esta herramienta, que ha sido la base de la mayoría que se han planteado posteriormente[2, 3], adolece de una serie de problemas entre los que cabe destacar: relación señal/ruido baja, los errores de las estimaciones se propagan, y sobrecarga de la red. Las posteriores herramientas que se han propuesto han ido modificando los mecanismos de cálculo de las estimaciones y resolviendo parte de los problemas que presentaba *pathchar*. Pero hay dos problemas principales que siguen sin ser resueltos, como son la propagación de los errores en las estimaciones de un salto a los siguientes y la sobrecarga que se origina en la red.

Por otro lado, la tecnología de redes activas [4, 5, 6] se ha propuesto como evolución de los modelos de red tradicionales. La idea fundamental es añadir programabilidad a las redes. Las redes activas constituyen una arquitectura de red en la que los nodos de la misma pueden realizar procesamiento a medida sobre los paquetes que los atraviesan. Las redes activas proporcionan un cambio en el paradigma de red: de nodos capaces exclusivamente de transportar octetos de forma pasiva, a nodos capaces de procesar los paquetes a cualquier nivel de la pila de protocolos.

Las redes activas introducen el concepto de procesamiento específico de los paquetes en base a código móvil que se ejecuta en los nodos de la red. Esto quiere decir que los nodos de la red no son sistemas de procesamiento especializados en un

* Este trabajo ha sido financiado por la CICYT dentro del proyecto TEL99-0988-C02-02

protocolo de red dado (para el caso de redes multiprotocolo en un limitado número de ellos), como sucede en la actualidad, sino que son plataformas de ejecución genéricas en las que se puede descargar dinámicamente código específico para el procesamiento de los distintos tipos de paquetes que se desee definir.

En este artículo se plantea la aplicación de la tecnología de redes activas al desarrollo de herramientas de diagnóstico de redes, con el objeto de solucionar parte de la problemática actual que presentan este tipo de herramientas. En concreto, se propone una herramienta activa, *a-clink*, que proporcionará una mayor precisión en las estimaciones evitando la propagación de errores, una mayor rapidez en los cálculos y una limitada sobrecarga de la red. Los motivos que justifican la aplicación de la tecnología de redes activas se basan en la conveniencia de disponer en la red de elementos programables que realicen un análisis de las características de la red distribuido y de alcance limitado.

El resto del artículo se ha organizado de la siguiente forma. Primero analizaremos cómo han ido evolucionando las herramientas de diagnóstico de redes y su problemática actual, para a continuación describir la herramienta de diagnóstico activa, *a-clink*, que presentamos en este artículo. Mediante dicha herramienta se demostrará cómo la tecnología de redes activas puede ayudar a mejorar el comportamiento de dichas herramientas. Seguidamente describiremos la implementación que hemos realizado sobre la plataforma de red activa SARA (Simple Active Router-Assistant architecture) [7] y los resultados que hemos obtenido en las pruebas realizadas. Terminaremos con las conclusiones obtenidas y las líneas de futuros trabajos.

2 Evolución de las herramientas de evaluación de prestaciones de red

En este apartado vamos a describir distintas herramientas que se han desarrollado en los últimos años para estimar las características de los enlaces de la red desde sistemas fianles, analizando las ventajas e inconvenientes de cada una de ellas. Aunque no pretende ser un listado completo de todas las aplicaciones, sí se intenta mostrar una selección de las más significativas. Finalmente, seleccionaremos una de ellas como base de nuestra aplicación.

2.1 Traceroute

La primera herramienta a la que haremos referencia es *traceroute*, desarrollada por Van Jacobson en 1988. Su funcionamiento se basa en el envío de una secuencia de paquetes IP desde una fuente a un destino incrementando sucesivamente el valor del campo TTL de la cabecera IP. Los routers por los que pasan los paquetes decrementan el valor TTL y si alcanza el valor 0, descartan el paquete y envían al emisor un mensaje de error ICMP. El emisor aprovecha estos mensajes de error para averiguar la dirección IP de los routers que hay en el camino entre el origen y el destino y su localización en número de saltos desde el origen (igual al valor TTL de salida). El emisor también calcula el RTT (round trip time), tiempo desde que sale el paquete IP hasta que llega el mensaje ICMP del router. Esta herramienta únicamente nos permite averiguar de forma fiable el camino seguido desde una fuente hasta llegar a un destino.

2.2 Pathchar

Desde 1991 Van Jacobson trabaja en el desarrollo de herramientas que además de descubrir rutas obtengan información sobre las características de los enlaces que componen el camino entre una fuente y un destino. En 1997 lanza una herramienta denominada *pathchar* [1] que muestra el camino entre una fuente y un destino y parámetros del estado de cada enlace en el camino: RTT y ancho de banda.

Basada en el funcionamiento de *traceroute*, *pathchar* envía una secuencia de paquetes a un destino. Para cada router que se encuentra en el camino (manteniendo un determinado valor de TTL) realiza una serie de pruebas que consisten en enviar una secuencia de paquetes de distintos tamaños. De esta forma mide el RTT total desde el origen a los distintos routers del trayecto en función del tamaño de los paquetes (Fig. 1).

Para expresar analíticamente esta curva, que se puede observar que se aproxima a una recta cuando se toman los valores de RTT(S) menores, podemos utilizar el modelo de la Fig. 2 en la que se observa el flujo de los datos en un enlace entre dos routers.

Los retardos que sufre un paquete en este escenario son:

1) Tiempos de propagación en cada uno de los enlaces (lo denominamos latencia).
2) Tiempos de transmisión de los paquetes a la velocidad nominal del enlace.
3) Tiempos de procesamiento de los paquetes en los sistemas finales y en los routers.

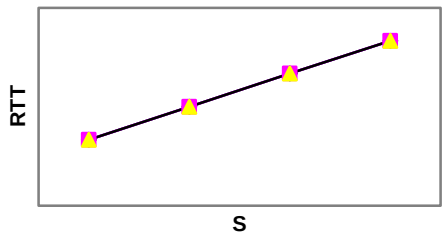


Fig. 1: Valor de RTT en función del tamaño de los paquetes S

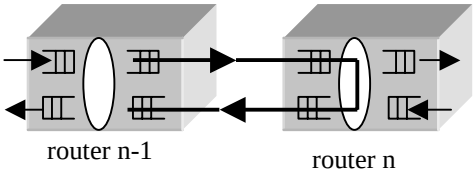


Fig. 2: Modelo del camino recorrido por la información.

4) Tiempo de espera en las colas de los routers y de los sistemas finales.

Todos estos parámetros se pueden considerar deterministas excepto el tiempo de espera en las colas, que tiene un carácter aleatorio. Para eliminar este tiempo se recurre a un método de repetición. Consiste en repetir las pruebas con cada tamaño de paquete un número suficiente de veces para poder tener una alta probabilidad de que una de las muestras no haya sufrido retardo de colas. Para dibujar la recta se utiliza un filtro que selecciona el valor mínimo de cada batería de repeticiones.

Una simplificación de este modelo en el que no se consideran: los tiempos de procesamiento en los sistemas finales y routers, el tiempo invertido en transmitir el mensaje ICMP de vuelta, y los tiempos de espera en cola, es expresada en la ecuación 1:

$$RTT_n(s) = \sum_{i=1}^{n}\left(2lat_i + \frac{S}{B_i}\right)$$

Ecuación 1. Modelo matemático simplificado.

Se representa el RTT desde el origen al router n teniendo en cuenta el retardo introducido en cada uno de los enlaces. Donde S es el tamaño del paquete, B_i es el ancho de banda del enlace i y lat_i representa el retardo de propagación del enlace i. Con esta simplificación lo que obtenemos es la ecuación de una recta de pendiente 1/B. Donde podemos estimar el valor de ancho de banda B.

Para realizar los cálculos de cada enlace se utiliza un método de cálculo regresivo (Fig. 3). Se comienza realizando los cálculos del primer enlace (n = 1), aplicando directamente la ecuación 1 y obteniendo el RTT en función de S para el dicho primer enlace. De la recta obtenida se mide:

1. **Latencia** igual a la mitad del valor de la recta cuando S = 0.
2. **Ancho de banda** igual al inverso de la pendiente de la recta.

Para el siguiente enlace realizamos los mismos cálculos pero basándonos en los resultados del enlace anterior. Es decir, la latencia será la diferencia entre la mitad de la latencia medida (total cuando S = 0) y la latencia del enlace anterior.

De la misma forma el ancho de banda del siguiente enlace vendrá dado por la inversa de la diferencia

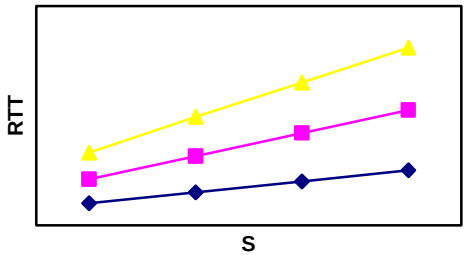


Fig. 3: Curvas de RTT(S) para distintos enlaces.

de pendientes de las curvas actual y la del enlace anterior. De esta forma se van calculando sucesivamente los parámetros para los distintos enlaces hasta llegar al destino.

Los principales problemas de *pathchar* son:

1. Relación señal/ruido baja: cuando se utilizan enlaces de alta velocidad (Ethernet, T3, OC-3 u OC-12), el tiempo de transmisión (señal) está comprendido entre 19 y 1200 µs para paquetes con longitud máxima de 1500 octetos. Este tiempo puede llegar a ser del orden de 1000 veces menor que el de propagación y que las fluctuaciones de tiempo en cola (ruido). Esta baja relación señal/ruido impide obtener medidas precisas.
2. Agregación de errores: los errores en las estimaciones se propagan al salto siguiente en el cálculo regresivo.
3. No funciona bien ante enlaces con múltiples canales que reparten el ancho de banda total del enlace en canales paralelos de menor ancho de banda.
4. Limitaciones de mensajes ICMP por temas de seguridad: puesto que la aplicación utiliza mensajes ICMP encontrará problemas en sistemas como Linux y Solaris que limitan la tasa de envío de estos mensajes. También se pueden encontrar rutas que pasan por routers o cortafuegos que filtran estos mensajes.
5. Sobrecarga en la red: debido a la gran cantidad de pruebas que se deben realizar y a que todos los paquetes se mandan desde la fuente a cada router.

2.3 Clink

En 1999 Allen B. Downey desarrolla una nueva herramienta denominada *clink* [2]. Basada en *pathchar*, utiliza las mismas técnicas de recolección de datos, presentando algunas mejoras en las técnicas de cálculo del ancho de banda con las que consigue estimaciones más precisas.

Al mismo tiempo Downey realiza un estudio para averiguar si es más eficiente aumentar el número de tamaños de paquete (más precisión en la curva) o el número de paquetes por tamaño (más posibilidades de encontrar el mínimo RTT). En

dicho estudio llega a la conclusión que es mejor aumentar la variación en el número de tamaños.

En la aplicación que implementa realiza por cada salto 8 pruebas por cada uno de los 93 tamaños distintos de paquete (28 y 1500 octetos con saltos de 16). Esto supone reducir la carga que se introduce en la red a prácticamente la mitad respecto a *pathchar*.

Con el objetivo de limitar el numero de muestras al estrictamente necesario para obtener una determinada precisión en las medidas, Downey propone realizar una recogida de datos adaptativa que consiste en ir aumentando progresivamente el número de muestras por cada tamaño y recalculando el RTT y ancho de banda hasta que se vea que los valores estimados convergen.

2.4 Pchar

También en 1999 Bruce A. Mah desarrolla *pchar* [3] una nueva herramienta basada en *pathchar*. Como principal mejora soporta algoritmos alternativos con el objetivo de tener una mayor precisión en el cálculo del RTT y ancho de banda. En concreto permite elegir entre dos algoritmos: 1) mínimos cuadrados o 2) método basado en tests estadísticos de Kendall.

Con esta herramienta se consiguen resultados similares a los obtenidos con *clink*, con el agravante de que es más agresiva en cuanto a tráfico generado.

2.5 Nettimer

En el año 2000 Kevin Lai y Mary Baker proponen una nueva técnica denominada P*acket Tailgating* y la desarrollan en una herramienta denomina *nettimer*[8]. En esta herramienta se utiliza un nuevo modelo denominado Pair-Packet, que consiste en enviar dos paquetes seguidos, uno muy pequeño y otro muy grande, que permitan medir diferencias de tiempos de llegadas en el receptor. A partir de estos datos se pude estimar el ancho de banda del cuello de botella.

La principal ventaja de este método es que las pruebas son menos agresivas que las propuestas por Jacobson, al tener que enviar una cantidad considerablemente menor de paquetes. Obteniéndose resultados con precisiones similares a las de las aplicaciones anteriores.

Como limitación principal podemos destacar que no obtiene buenos resultados si en el camino existe un enlace muy rápido después de uno lento. En concreto la relación entre el enlace rápido y el lento no debe superar 37.5.

2.6 Selección de herramienta base

En este artículo planteamos cómo la tecnología de redes activas puede ayudar al desarrollo de este tipo de herramientas de evaluación de prestaciones de red. Para lo cual, partiendo de una de estas herramientas desarrollaremos una herramienta activa, que aporta las ventajas que proporciona la tecnología de redes activas. A la hora de decidir la herramienta base de entre las analizadas, nos hemos basado en el estudio comparativo que se realiza en [8]. Dicho estudio se realiza en base a dos aspectos:

1. **Estimación de ancho de banda de los enlaces**: no destaca considerablemente ninguna herramienta con respecto a las demás.
2. **Número de paquetes enviados en las pruebas**: *nettimer* es la menos agresiva seguida por *clink*, que en pruebas referenciadas es entre 2 y 6 veces más agresiva, y finalmente *pathchar* y *pchar* que son el doble de agresivas que *clink*.

Por lo tanto se puede decir que *clink* es más agresiva que *nettimer*, pero dado que *nettimer* tiene la limitación de no obtener buenos resultados sobre enlaces rápidos seguidos de lentos, situación que se da en Internet, y que en nuestra propuesta pretendemos reducir la sobrecarga que introducen estas herramientas en la red, hemos decidido seleccionar *clink.*

3 Aplicación activa a-clink

La aplicación activa *a-clink* se basa en utilizar la tecnología de redes activas para resolver parte de los problemas que presentan las actuales herramientas de evaluación de prestaciones. En concreto, se ha visto que uno de los principales problemas de los que adolecen este tipo de herramientas es la propagación de los errores en las estimaciones en un enlace a los siguientes enlaces. Mediante la aplicación *a-clink* se lanzan varias aplicaciones de estimación de características de los enlaces en paralelo, una en cada router activo que exista en el camino entre el origen y el destino. Cada una de estas aplicaciones realizará los cálculos correspondientes a los enlaces entre dicho router activo y el siguiente router activo o el destino si es el último. De esta forma, el número de enlaces que estimará cada aplicación será mucho menor, limitándose por tanto la propagación de errores a los enlaces del tramo entre routers activos.

Otro de los problemas de dichas herramientas es la sobrecarga que producen en la red, sobre todo en los enlaces más cercanos a la fuente que deben soportar no solo el tráfico que se envía para calcular sus características, sino también el tráfico que se envía para calcular las características del resto de enlaces que se encuentran entre ellos y el destino. En concreto, cada enlace deberá soportar de forma adicional el tráfico necesario para realizar los cálculos de un enlace multiplicado por el número de

enlaces que existen hasta el destino. En *a-clink* la sobrecarga se reduce dado que el número de enlaces que hay entre un enlace determinado y el router activo o destino contra el que se lanza el cálculo de estimaciones, se ve reducido considerablemente y por lo tanto también el tráfico que se inyecta.

Al mismo tiempo, dado que se distribuye el proceso de realización de pruebas entre los routers activos que hay en el camino entre la fuente y el destino, se reducirá el tiempo necesario para realizar las pruebas y obtener las estimaciones pertinentes, frente a las soluciones tradicionales que únicamente lanzan un proceso en la fuente.

3.1 Descripción del escenario de la aplicación

En la aplicación *a-clink* se divide el camino entre los sistemas finales en *segmentos* delimitados por los routers activos. De esta forma cada router activo realizará los cálculos de estimación de los enlaces de su segmento y enviará los resultados al sistema final que actúe como origen. Por lo tanto, en este nuevo escenario podemos distinguir cuatro tipos de sistemas que intervienen en la aplicación:

- **Sistema final origen**: inicia la aplicación y recopila los resultados de los routers activos.
- **Router activo**: realiza cálculos en su segmento devolviendo los resultados al sistema final de origen.
- **Router no activo**: situado dentro de los segmentos no intervienen directamente en el proceso activo, sólo en el proceso tradicional descrito en el funcionamiento de *clink*.
- **Sistema final de destino**: delimita el final de la ruta.

Los routers activos descargarán la aplicación *a-clink* que básicamente tendrá dos funciones principales:

- Descubrir cual es su segmento de ruta.
- Realizar los cálculos de estimación de parámetros de los enlaces en su segmento utilizando la aplicación *clink* tradicional.

Los routers activos mantiene la siguiente información de estado sobre la aplicación *a-clink*:

- **Id. Sesión**: identifica la sesión de forma única.
- **IP Origen**: sistema final origen.
- **IP destino**: sistema final destino.
- **Next**: siguiente router activo en la ruta, contra el que realizan los cálculos de medidas de los enlaces.
- **Saltos**: número de saltos de routers activos desde el origen hasta este router.

3.2 Proceso de segmentación de la ruta

El sistema final origen comienza el proceso enviando un paquete **explorador** dirigido al sistema final destino. El paquete va acumulando la siguiente información:

- **Sentido**: sentido del paquete explorador (origen a destino o destino a origen).
- **Saltos**: número de routers activos en el camino hasta el destino.
- **ACK[n]**: confirmación de cada router activo del mensaje explorador de vuelta. Hay un campo por cada router activo.
- **IP[n]**: dirección de cada router activo en la ruta. Hay un campo por cada router activo.

Cuando el paquete explorador, en dirección origen a destino, pasa por un router activo, éste realiza los siguientes procesos:

- Registra el identificador de sesión.
- En el paquete explorador incrementa el valor del campo s*altos* y le añade su *dirección IP*.
- Actualiza sus variables de estado (*origen, destino* y *saltos*).
- Reenvía el paquete explorador hacia el destino.

Cuando el paquete explorador llega al destino, lleva una lista ordenada de las direcciones IP de cada uno de los routers activos en la ruta.

El destino cambia el bit de sentido del paquete explorador y lo reenvía al origen siguiendo el camino inverso. Cada router activo al recibir el paquete explorador de vuelta actualiza su variable de estado *next* con la dirección IP del sistema siguiente a él en el camino de la fuente al destino, este sistema puede ser el siguiente router activo o el propio destino. Pone el bit ACK correspondiente a uno y reenvía el paquete hacia el origen.

Cuando el origen recibe el paquete explorador de vuelta, comprueba que todos los routers activos han visto el mensaje de vuelta (ACK=1), y han registrado por tanto la dirección de su siguiente salto en la ruta, y actualiza su variable de estado *next* apuntando al primer router activo en la ruta.

Cuando termina este proceso de exploración de routers activos en la ruta entre el origen y el destino, el origen conoce exactamente los routers activos que se encuentran en su camino al destino y cada uno de los routers activos conoce quién es su propio destino para el cálculo de estimaciones.

3.3 Cálculos de estimación de parámetros de los enlaces

Cada uno de los routers activos y el origen lanzarán la aplicación de cálculo de estimación de las características de los enlaces cuando reciban el paquete explorador de vuelta, dado que en ese momento ya conocen cuál es el destino de su segmento. Entonces cada uno lanzará la aplicación *clink* contra dicho destino. Terminado dicho proceso cada router activo enviará a la fuente los resultados de los cálculos en su segmento.

El origen, una vez que haya terminado el cálculo de su propio segmento de ruta y recibido los resultados del resto de routers activos, ordenará y mostrará las

estimaciones calculadas para cada uno de los enlaces de la ruta completa.

4 Implementación y pruebas realizadas

Para la implementación de la aplicación *a-clink* se han analizado las principales plataformas de redes activas. ANTS [9] desarrollada por el MIT y basada en una máquina virtual java (JVM) es una de las plataformas más utilizadas debido a su relativa facilidad de utilización y a que fue una las primeras plataformas de red activa desarrolladas. Pero como se pudo comprobar en [10] las prestaciones que proporciona hace que no sea la más adecuada para implementaciones en las que uno de los objetivos principales sea evaluar prestaciones. Otra de las plataformas más conocidas se corresponde con la propuesta desarrollada en el proyecto Switchware por la Universidad de Pensilvania. Dicha plataforma se basa en PLAN[11], un lenguaje funcional de propósito específico basado en OCAML, lo que complica bastante las tareas propias de la implementación. BBN en el proyecto Smart Packets [12] propone una plataforma específica para usarla en tareas de gestión de red. La Universidad Carlos III de Madrid ha desarrollado en el proyecto europeo IST-GCAP[7], una plataforma de redes activas, denominada SARA, caracterizada por estar orientada a proporcionar altas prestaciones y por su facilidad de utilización. Es debido a estas características el que se haya seleccionado dicha plataforma para la implementación de *a-clink*. Esta plataforma utiliza Java para el desarrollo de aplicaciones y se instala sobre el sistema operativo Linux.

La aplicación *a-clink* se descarga bajo demanda desde un servidor de código ante la llegada del primer paquete activo que lleve el identificador de la aplicación.

4.1 Descripción del escenario de pruebas

El escenario real sobre el que se han realizado las pruebas está formado por dos redes, una en la Universidad de Alcalá y otra en la Universidad Carlos III de Madrid, unidas por un túnel. En la Fig. 4 se puede ver el esquema. Las características de los equipos y los enlaces se describen a continuación.

Universidad de Alcalá:

- SF1: K6/266, 98 MB de RAM.
- R1: Smart Switch Router 2000 de Cabletron.
- RA1: Pentium III/600, 64 MB de RAM.
- R2: Pentium II/350 , 64 MB de RAM.

Universidad Carlos III de Madrid:

- RA2 y R3: Pentium III 733MHz, 128 MB de RAM.
- SF2: Pentium III 700MHz, 128 MB de RAM.

Todos los ordenadores tienen instalado el sistema operativo Linux. Los enlaces tienen las siguientes características:

- Los enlaces E1, E2 y E3 son Ethernet a 10Mbps.
- El enlace E4 es un túnel que une las redes de la Universidad de Alcalá y la Universidad Carlos III.
- Los enlaces E5 y E6 son Fast Ethernet a 100Mbps.

En los enlaces E1, E2, E3, E5 y E6 no se ha introducido más tráfico que el generado por la propia aplicación de prueba.

Nos hemos encontrado dos problemas a la hora de implementar este escenario:

1. En la entrada de la red de la Universidad Carlos III se filtra el tráfico ICMP. Esto no permite el funcionamiento de *clink* que se basa en la recogida de paquetes ICMP. Para resolverlo se ha implementado un túnel IP entre R2 y RA2 (Fig. 4).
2. Linux limita la velocidad de envío de paquetes ICMP. Hemos resuelto este problema desprotegiendo dicha limitación en el kernel del sistema operativo para los paquetes ICMP que utiliza la aplicación *clink*.

4.2 Pruebas realizadas y análisis de los resultados

Para probar la aplicación activa *a-clink* y comparar sus resultados con los de *clink*, hemos realizado diversas pruebas de ambas aplicaciones en el escenario de la Fig. 4. Las pruebas se han realizado enviando paquetes a cada router de 91 tamaños distintos comprendidos entre 28 y 1468 bytes con saltos de 16. Cada tipo de paquete se ha repetido 8 veces. Por lo tanto, el total de paquetes enviados a cada router del camino es 728 (91*8).

La tabla 1 muestra los resultados de las pruebas para cada una de las aplicaciones. La columna LAT corresponde a la estimación de la latencia y la columna BW a la estimación del ancho de banda. Cada fila muestra los resultados en cada uno de los enlaces del camino en el escenario de pruebas. Este camino, desde el punto de vista de la aplicación activa, se divide en tres segmentos; el primero formado por E1 y E2, el segundo por E3 y E4, y el tercero por E5 y E6.

De los resultados de estas pruebas podemos deducir que los valores de estimación de latencia no corresponden realmente con el retardo de propagación, según suponía el modelo teórico. En dicho modelo los retardos de procesamiento interno se despreciaban frente al retardo de propagación.

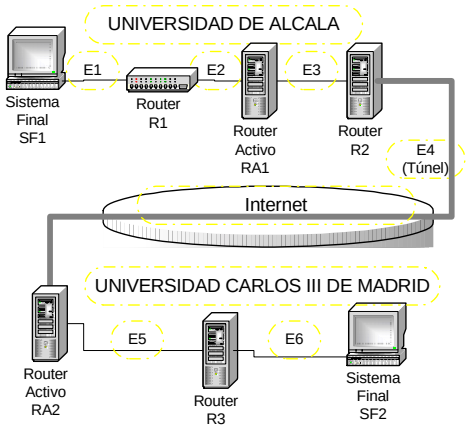


Fig. 4. Escenario de pruebas.

Esto puede ser cierto en el entorno de aplicación de Internet, en el que las distancias son mayores con dispositivos de conmutación rápidos. En nuestro escenario las distancias son cortas, abarcando valores que van desde unos metros hasta unas pocas decenas de kilómetros (en el caso del túnel). Esto se traduce en unos tiempos de propagación que pueden estar en un rango aproximado entre 50 ns y 150 µs. Por otra parte los tiempos de procesamiento interno dependen de la CPU del equipo, pudiendo llegar a alcanzar valores de varios milisegundos y por lo tanto nada despreciables. Resumiendo podemos decir que la estimación de latencia (columna LAT de la tabla 1), realmente nos da idea de los tiempos consumidos en procesamiento de paquetes en cada enlace.

También podemos observar que las estimaciones de ancho de banda no se aproximan con suficiente precisión a los valores nominales de los enlaces. La explicación está en la suposición del modelo teórico de que el único tiempo que depende del tamaño de los paquetes es el tiempo de transmisión. En las máquinas hay procesos internos cuyo tiempo de proceso depende del tamaño de los paquetes. Este tiempo se suma al de transmisión, ocasionando desviaciones en la estimación del ancho de banda.

Tabla 1. Resultados de las pruebas.

Enlace	Seg-mento	CLINK LAT (ms)	ACLINK LAT (ms)	CLINK BW (Mbs)	ACLINK BW (Mbs)
E1	Seg. 1	0,277	0,263	6,98	6,994
E2		0,384	0,368	4,131	4,009
E3 (RA)	Seg. 2	0,211	0,238	6,667	6,668
E4		50,683	49,953	0,276	0,302
E5 (RA)	Seg. 3	-8,424	0,051	-0,813	65,852
E6		2,655	0,035	2,269	64,496

Analizando los resultados que se obtienen con *clink* vemos que las estimaciones de los enlaces E5 y E6 no son buenas, llegando incluso a dar valores negativos. Analizando los resultados que se obtienen con *a-clink* vemos que las estimaciones correspondientes a los enlaces E1, E2, E3, E5 y E6 son correctas, con las matizaciones comentadas previamente. El enlace E4 queda fuera de nuestro control con lo cual es difícil determinar la precisión de sus estimaciones. Lo que si queda claro es que *a-clink* aísla los posibles errores de las estimaciones que produce este enlace, al iniciarse de nuevo los cálculos en el segmento 3, evitando como ya hemos comentado la propagación de los errores. Estos posibles errores pueden ser debidos a varios motivos entre los que podemos destacar congestión en los routers y ocultación de nodos.

Otra mejora que se obtienen con la aplicación *a-clink* se refiere a los tiempos invertidos en la ejecución de la aplicación frente a *clink*, reduciéndose considerablemente el tiempo necesario para la realización de las pruebas. Esto es debido a que la aplicación *a-clink* se ejecuta en paralelo en cada uno de los tres segmentos del camino. En nuestro caso los tiempos invertidos en la ejecución de *clink* y *a-clink* han sido 11:47 y 6:09 minutos respectivamente.

Por último en la tabla 2 se representa el número de paquetes que pasan por cada uno de los enlaces, para cada una de las aplicaciones. Para calcular el número de paquetes que soporta cada enlace podemos utilizar la expresión $\mathbf{N_i=2n(k-i+1)}$. Donde N_i es el número de paquetes que soporta el enlace i, n es el número de paquetes generados para realizar las pruebas, k es el número de enlaces del camino (*clink*) o del segmento (*a-clink*) e i es el número de orden del enlace en el camino o segmento. En nuestro caso n = 728 paquetes. Remarcar que en estos cálculos se incluyen los mensajes ICMP de respuesta.

Analizando dicha formula se deduce claramente que el número de paquetes que debe soportar un enlace depende directamente del número de enlaces en el camino entre el origen y el destino. Para c*link* ese número se corresponderá con el número total de enlaces entre el origen y el destino. Para *a-clink* dicho número queda reducido en función del número de segmentos que se establezcan, que dependerá del número de routers activos en el camino entre la fuente y el destino.

En la tabla 2 vemos como los primeros enlaces con *clink* soportan una carga elevada al acumular los mensajes del resto de los enlaces. Con *a-clink* únicamente soportan los mensajes de su segmento.

En base a los resultados obtenidos podemos concluir que la aplicación *a-clink* al dividir el camino en segmentos, reduce el tiempo de cálculo y el número de paquetes soportados por los enlaces, y

aísla las perturbaciones introducidas en determinados enlaces, de forma que no afecten a las medidas en enlaces posteriores.

5 Conclusiones y trabajos futuros

En este artículo hemos descrito una nueva herramienta de caracterización del estado de los enlaces basada en la tecnología de redes activas y que utiliza como base una herramienta ya existente. Se han realizado pruebas comparativas de ambas herramientas y hemos observado que nuestra propuesta reduce el número de paquetes necesarios para las estimar el estado de los enlaces, reduce el tiempo de ejecución de la aplicación y se recupera de errores en enlaces en los que las perturbaciones no permiten obtener buenos resultados. Al mismo tiempo se ha visto como las estimaciones obtenidas con la herramienta *clink*, que como se demostró en [8] proporciona estimaciones equivalentes a las obtenidas con otras herramientas de estimación de prestaciones en red, hay que manejarlas con cierta cautela.

Queremos remarcar que dichos resultados se han obtenido en un entorno de pruebas donde el número de enlaces totales es reducido. Si extrapolamos los resultados obtenidos a un entorno más real, Internet, en donde el número de enlaces entre un origen y destino es bastante mayor, podemos concluir que el impacto de las ventajas obtenidas se verá en gran medida amplificado.

En cuanto a la plataforma de red activa SARA, podemos concluir que la familiarización y utilización de dicha plataforma ha sido fácil y rápida. Y que la plataforma se encuentra en un estado de desarrollo lo suficientemente maduro como para poder realizar implementaciones de aplicaciones activas en un entorno real. En relación a la tecnología de redes activas, la herramienta desarrollada es una más de los ejemplos de aplicaciones que aprovechan las ventajas de dicha tecnología.

Tabla 2. Cantidad de paquetes en cada enlace.

Enlace	Segmento	CLINK (paquetes)	ACLINK (paquetes)
E1	Seg. 1	8736	2912
E2		7280	1456
E3 (RA)	Seg. 2	5824	2912
E4		4368	1456
E5 (RA)	Seg. 3	2912	2912
E6		1456	1456

Para terminar, podemos destacar tres líneas de actuación futuras. 1) Completar el diseño del protocolo de segmentación de la ruta para que soporte pérdida de paquetes de control. 2) Modificaciones del protocolo para que no sea necesario tener un sistema final de destino con procesamiento de routers activos. 3) Simulación de la aplicación para poder comparar eficiencias en escenarios más complejos.

Agradecimientos

Queremos mostrar nuestro agradecimiento a los becarios José Alberto Fernández, Clara Santos y Manuel Urueña y a los proyectandos Andrés Navarro y Graciela Garrido por haber participado en la implementación de la aplicación *a-clink* y en la realización de las pruebas.

Referencias

[1] V.Jacobson. *Pathchar-A tool to infer characteristics of Internet paths*. Presented at the Mathematical Sciences Research Institute, April 1997.

[2] A. B. Downey. *Using pathchar to Estimate Internet Link Characteristics*. In proceedings of ACM SIGCOMM 1999, pp. 241-250, July 1999.

[3] Bruce A. Mah. *Pchar: Child of Pathchar*. Presented at the DOE NGI Testbed Workshop, Berkeley, CA, 21 July 1999.

[4] D.L. Tennenhouse, J.M. Smith, W.D. Sincoskie, D.J. Wetherall and G.J. Minden. *A survey of Active Network Research*. IEEE Communications Magazine, pp. 80-86, January 1997.

[5] M. Calderón, M. Sedano, S. Eibe García. *Principios y Aplicaciones de las redes Activas*. Proceedings de Jitel´99. pp 311-319. Sep 1999.

[6] Jonathan T. Moore Scott M. Nettles. *Towards Practical Programmable Packets*. Technical Report MS-CIS-00-12 . May 2000.

[7] Servidor www proyecto IST GCAP (Global Communication Architecture and Protocols for new QoS services over IPv6 networks). http://www.laas.fr/GCAP/

[8] K Lai, M Baker. *Measuring Link Bandwidths Using a Deterministic Model of Packet Delay*. In Proceedings of ACM SIGCOMM 2000. August 2000.

[9] D. Wetherall, J. Guttag and D.L. Tennenhouse. *ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols*. *IEEE OPENARCH'98*, San Francisco, CA, April 1998.

[10] M. Calderón, M. Sedano, A. Azcorra and C. Alonso. *Active Networks Support for Multicast Applications*. IEEE Network Magazine, Special Issue: Active and Programmable Networks, Vol. 12 No. 3 pp 46-52, Mayo/Junio 1998.

[11] S. Alexander, W. A. Arbaugh, M. W. Hicks, P. Kakkar, A. D. Keromytis, J. T. Moore, C. A. Gunter, S. M. Nettles and J. M. Smith. *The SwitchWare Active Network Architecture*. IEEE Network, Special Issue: Active and Programmable Networks, Vol 12(3) pp. 29-36, May/June 1998.

[12] B. Schwartz, A. Jackson, T. Strayer, W. Zhou, R. Rockwell, and C. Partridge. *Smart packets for active networks*. In proceedings of the 1999 IEEE 2nd Conference on Open Architectures and Networks Programming (OPENARCH'99), March 1999.

Modelado y Simulación de Protocolos para Redes Activas

Guillermo Rodríguez, Pedro Merino, María del Mar Gallardo
Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga.
Campus Teatinos, 29071 Málaga (SPAIN)
E-mail: guille@iies.es, pedro@lcc.uma.es, gallardo@lcc.uma.es

Abstract. *Active Networks (ANs) represent a new paradigm of computer network which will enable new Internet applications and services and improve end-to-end performance for existing ones. However, ANs being an emerging technology, there is still a significant lack of tools for the design and evaluation of active network protocols. In particular, network simulators have proven to be a very valuable tool and they have been widely used in the Internet research community. In this paper, we present a novel extension for the well-known network simulator* ns *to incorporate AN support. Our solution is versatile yet powerful, providing a consistent framework for researchers to design and evaluate active protocols.*

1 Introducción

Las *Redes* Activas [1] representan un nuevo paradigma de redes de ordenadores, basado en el concepto de *infraestructura programable*. En este modelo, el comportamiento de la red se puede modificar dinámicamente a través de *paquetes activos* especiales que inyectan código en los nodos. Esto permite introducir nuevos servicios en la red de forma rápida y eficaz. Además, la existencia de una infraestructura programable permite particularizar el comportamiento de la red de forma independiente para cada usuario o aplicación, facilitanto así la especialización de los servicios. Las redes activas ya han demostrado ser una herramienta muy valiosa para el estudio de diversos problemas comunes en Internet hoy en día, tales como el transporte multicast fiable [2, 3], los sistemas de caché a nivel de red [4], el control de congestión [5] y muchos otros.

Puesto que la tecnología de redes activas es aún muy novedosa, existe una carencia importante de herramientas de desarrollo en este campo. En particular, resulta especialmente significativa la falta de herramientas de simulación para el diseño y evaluación de *protocolos activos*. En los últimos años, la simulación por ordenador se ha consolidado como la metodología predominante para el desarrollo de protocolos de comunicaciones. Las herramientas de simulación constituyen un entorno de desarrollo flexible y versátil, permiten el estudio de sistemas de gran complejidad a diferentes escalas o niveles de detalle, y proporcionan otras ventajas adicionales, como la generación sistemática de casos de prueba cubriendo un amplio espectro de topologías y distribuciones de tráfico, o la disponibilidad de técnicas avanzadas para la visualización de datos.

En el marco de la simulación de redes, el simulador *ns* [6, 7] ha venido disfrutando de una creciente popularidad y aceptación en el ámbito académico, entre otros. *Ns* es un simulador multiprotocolo de propósito general, desarrollado en el marco del proyecto VINT (*Virtual Inter-Network Testbed*) con el objetivo de proporcionar una plataforma universal de simulación para el estudio del comportamiento e interacción de protocolos de comunicaciones a diferentes niveles de granularidad. Como muestra de su amplia difusión, valga decir que fue el simulador más usado en SIGCOMM'98, según se afirma en [6].

En este artículo abordamos el diseño e implementación de un conjunto de extensiones para el simulador *ns* que incorporan soporte para redes activas. El objetivo que perseguimos es proporcionar una marco de desarrollo común que facilite y promueva la investigación y el desarrollo de nuevos protocolos y servicios.

El resto de este documento se ha estructurado como se describe a continuación. La sección 2 describe brevemente los principales componentes arquitecturales de una red activa, y proporciona una somera introducción a la singular arquitectura software del simulador *ns*. La sección 3 analiza los problemas asociados a la extensión de *ns* para la simulación de redes activas, y describe la solución propuesta. La sección 4 incluye una serie de ejemplos que ilustran el modelado de protocolos activos en el marco de las extensiones desarrolladas. La sección 5 describe otros trabajos relacionados, y la sección 6 presenta las conclusiones obtenidas.

2 Preliminares

Esta sección describe brevemente los componentes arquitecturales de una red activa, e introduce la arquitectura software del simulador *ns*.

2.1 Redes activas

Tal y como se describe en [8], la funcionalidad de cada nodo activo se divide en un *sistema operativo de nodo* (*NodeOS*), uno o más *entornos de ejecución* (*EEs*) y una serie de *aplicaciones activas* (*AAs*) o "protocolos activos".

El *NodeOS* es el responsable de la asignación y gestión de los recursos del nodo: ancho de banda de transmisión, tiempo de proceso (ciclos de CPU) y capacidad de almacenamiento (memoria). El NodeOS desempeña, en el nodo activo, un papel equivalente al de un sistema operativo en un computador de propósito general.

Cada *EE* define una máquina virtual capaz de interpretar las instrucciones codificadas en los paquetes activos que llegan al nodo. Esta máquina virtual se puede programar o controlar a través de un determinado interfaz de programación (API) exportado por el EE. Un EE realiza, en cierto modo, una función análoga a la de un *shell* en un sistema operativo de propósito general. En cada nodo activo pueden coexistir diversos EEs.

Las *AAs* son programas que se ejecutan en el contexto de un determinado EE, como consecuencia de la llegada de paquetes activos al nodo, y que utilizan el API proporcionado por dicho EE para implementar servicios extremo a extremo, es decir, servicios de red visibles para las aplicaciones y usuarios finales.

Desde el punto de vista del desarrollador de AAs, los EEs se pueden clasificar según tres atributos clave:

1. *Grado de programabilidad*. Las máquinas virtuales definidas por distintos EEs pueden ser de naturaleza muy diferente: desde máquinas completamente genéricas y universales (interfaz Turing-completo), hasta otras que sólo permitan elegir una opción de entre varias predefinidas.

2. *Recursos y servicios accesibles a las AAs*. Por ejemplo, un aspecto fundamental es si el EE permite a los paquetes activos almacenar información de estado en el nodo, y recuperar información almacenada por otros paquetes, suponiendo que estén autorizados a acceder a la misma.

3. *Granularidad de control*. Es decir, la granularidad con la que es posible programar la máquina virtual; por ejemplo, una sola vez por sesión o usuario, una vez para cada flujo de paquetes, o de forma independiente con cada paquete individual. Este aspecto está estrechamente relacionado con los mecanismos utilizados para distribución del código asociado a las AAs por la red.

2.2 El simulador *ns*: arquitectura software

El simulador *ns* se basa en una arquitectura modular y extremadamente flexible, diseñada específicamente para fomentar la extensión y particularización de la herramienta por parte de los usuarios. En *ns*, los experimentos se expresan en forma de *programas* o *scripts* de simulación que crean y configuran dinámicamente los distintos objetos que componen el escenario de simulación.

Ns utiliza un *modelo de programación híbrido* en el cual el *kernel* o núcleo del simulador y el conjunto básico de primitivas para el procesado de datos a bajo nivel se implementan en un lenguaje compilado (C++), mientras que los modelos se definen, configuran y controlan mediante un *script* de simulación usando un lenguaje interpretado (OTcl, una extensión de Tcl orientada a objetos desarrollada en el MIT). La separación entre procesado de datos y operaciones de control permite maximizar la eficiencia, manteniendo al mismo tiempo un interfaz de configuración cómodo y fácil de usar.

Los componentes más básicos, implementados en C++ modelan elementos tales como demultiplexores, colas o módulos de retardo. Estos componentes se pueden combinar mediante OTcl para formar "macro-objetos" (clases OTcl) de más alto nivel, tales como *routers* o enlaces. Los macro-objetos, a su vez, pueden formar parte de otros macro-objetos, y así sucesivamente; por ejemplo, en los niveles superiores, un único objeto podría modelar una topología completa, incluyendo fuentes y patrones de generación de tráfico. En última instancia, se llega al *script* de simulación, que combina diferentes objetos para definir el escenario de un experimento. El resultado es una arquitectura jerárquica organizada en múltiples niveles de abstracción.

Esta arquitectura permite al usuario implementar su simulación eligiendo el máximo nivel de abstracción que le proporcione el grado de control necesario, e ignorar los detalles del funcionamiento de las capas inferiores. Por ejemplo, una gran parte de los experimentos se puede realizar trabajando exclusivamente desde OTcl, creando, configurando y combinando objetos existentes. Para aquellos usuarios más exigentes que deseen particularizar los detalles internos de la herramienta, o aquellos que necesiten implementar nuevas primitivas y componentes básicos, es posible trabajar en C++ al nivel más bajo, extendiendo y modificando directamente el núcleo del simulador.

Los tipos de objetos más comunes en *ns* son los siguientes:

- *Nodos*: macro-objetos OTcl que representan tanto *routers* como sistemas finales. Están compuestos principalmente de un conjunto de *clasificadores* (demultiplexores) que clasifican los paquetes entrantes y los reenvían a través de enlaces salientes o los entregan a agentes locales.

- *Enlaces*: macro-objetos OTcl compuestos de una cadena de *conectores*, que son componentes capaces de recibir paquetes, procesarlos, y entregarlos al siguiente componente de la cadena. Un enlace típico suele contener una cola de entrada, un módulo de retardo, y un módulo decrementador de TTL.

- *Agentes*: componentes básicos que se instalan en los nodos para modelar entidades de un determinado protocolo. A su vez, es posible conectar un agente con una *aplicación simulada* (telnet, ftp) o con un *generador de tráfico* (exponencial, pareto, CBR), que son componentes que modelan fuentes de tráfico en la red.

Existen también componentes especializados para la simulación de redes locales, o de sistemas de comunicaciones móviles o vía satélite, objetos que implementan técnicas de abstracción para la simulación a gran escala, otros que permiten conectar el simulador con una red real, y un largo etcétera.

3 Extensión de *ns* para la simulación de redes activas

En esta sección abordamos el problema de incorporar soporte para redes activas en el simulador *ns*, y tras analizar los problemas asociados, describimos la solución propuesta. El diseño de esta solución se ha hecho atendiendo a los siguientes objetivos:

1. *Genericidad*. Las extensiones deben ser flexibles y versátiles, soportando el modelado de redes "parcialmente activas", compuestas por nodos activos y *routers* convencionales, así como la simulación de un número arbitrario de protocolos de forma simultánea. Además, el sistema debe ser lo suficientemente genérico como para permitir el diseño y evaluación de AAs que más tarde se puedan implementar sobre un EE arbitrario.

2. *Facilidad de uso*. El objetivo principal de este proyecto es evitar la duplicación de trabajo, proporcionando la infraestructura básica y todos los componentes básicos necesarios para la simulación de protocolos sobre redes activas, como por ejemplo la infraestructura necesaria para el procesado y encaminamiento de paquetes activos, o facilidades para el almacenamiento de información en los nodos activos. Para que este enfoque resulte efectivo, es indispensable que la herramienta sea fácil de usar, con una curva de aprendizaje suave y gradual.

3. *Escalabilidad*. La escalabilidad es uno de los aspectos más importantes en la simulación por ordenador, y uno de los puntos fuertes en el simulador *ns*. Por tanto, es especialmente importante que las extensiones desarrolladas preserven esta característica.

4. *Compatibilidad.* El propio *ns* es en sí mismo un proyecto en continua evolución, sujeto a constantes cambios y modificaciones. Además, su arquitectura software se ha diseñado con el objetivo explícito de promover la extensión de la herramienta por parte de los propios usuarios. Para maximizar la compatibilidad, tanto con futuras versiones "oficiales" del simulador como con las distintas versiones modificadas en circulación, se deben minimizar las dependencias a nivel de implementación, y en particular, evitar la modificación de los componentes básicos o de la estructura interna del simulador.

El desarrollo de extensiones para la simulación de redes activas en *ns* es un problema complejo en el que hemos identificado tres aspectos clave: (i) el diseño de la infraestructura básica necesaria para el encaminamiento y procesado de paquetes activos, (ii) la disposición de mecanismos para el modelado de AAs, y (iii) la definición del EE, es decir, de la máquina virtual en la que se ejecutarán estas AAs.

Un aspecto importante que no hemos modelado de forma explícita es el sistema de distribución del código asociado a las AAs. Nuestro modelo asume que este código está ya disponible en los nodos correspondientes. Esto se debe a que, pese a que el sistema de distribución de código es un aspecto fundamental del diseño de los propios EEs, en la mayoría de los casos no tiene un impacto significativo en el comportamiento o prestaciones de las AAs que corren sobre los mismos. Es más, este efecto, de existir, estará acotado temporalmente y sólo será apreciable durante un breve transitorio inicial, tras el cual todo el código habrá sido cargado en los nodos correspondientes, y por tanto la operación de las AAs no se verá afectada.

3.1 Infraestructura básica

En primer lugar, debemos diseñar e implementar la infraestructura básica necesaria para el encaminamiento y procesado de paquetes activos.

Puesto que se espera que la introducción de nodos activos en Internet se haga de forma gradual, una red activa típica contendrá tanto nodos activos como *routers* convencionales. Por este motivo, se puede considerar que las redes activas son redes normales de conmutación de paquetes en las que ciertos nodos "especiales" realizan un procesado adicional sobre los paquetes activos que pasan a través de ellos.

En nuestro modelo, el encaminamiento de paquetes activos se hace de la misma forma que si se tratase de paquetes convencionales, y son los nodos activos intermedios los que deben encargarse de detectar e interceptar estos paquetes para su evaluación. Se trata de un mecanismo conceptualmente similar a la opción *router alert* del protocolo IP: los paquetes activos son tratados como cualquier otro paquete por un *router* convencional, pero detectados e interceptados automáticamente por cada nodo activo que encuentran en su camino.

La fig. 1 muestra la estructura interna de un nodo activo. Se trata de un nodo convencional, a cuya entrada se ha insertado un nuevo componente, el *procesador activo* (`ANProcessor`), que implementa el mecanismo tipo *router alert* antes descrito. El

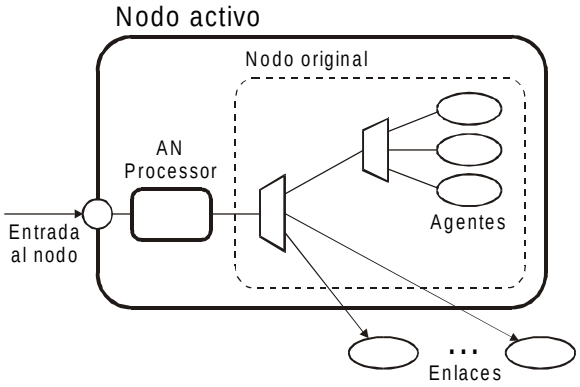


Fig. 1: Estructura interna de un nodo activo

procesador activo monitoriza todo el tráfico entrante. Los paquetes no activos lo atraviesan de forma transparente, y son procesados por el nodo de la forma habitual. Los paquetes activos son interceptados y evaluados, lo cual implica la ejecución del código asociado a la AA correspondiente. Tras este procesado, normalmente, los paquetes activos proseguirán su camino. Así, el procesador activo desempeña también el papel del NodeOS y del EE, al proporcionar el contexto en el que se ejecutan las AAs.

3.2 Modelado de Aplicaciones Activas

El proceso habitual a seguir para incorporar un nuevo protocolo al simulador *ns* es el siguiente. En primer lugar, se debe definir un nuevo tipo de paquete, especificando los campos presentes en la cabecera, junto con sus tamaños y posiciones. A continuación, se debe desarrollar un agente capaz de generar estos paquetes. Este agente se implementa en una clase derivada de la superclase C++ `Agent`. La clase derivada deberá redefinir diversos métodos, definidos como virtuales en la superclase, para modelar aspectos específicos del funcionamiento del protocolo en cuestión. Por último, es necesario implementar una clase OTcl asociada al agente, que proporcionará el interfaz de control y configuración del objeto desde el *script* de simulación.

El problema inherente a este enfoque es que, al ser C++ un lenguaje compilado, el proceso anterior implica recompilar la herramienta cada vez que se introduce un nuevo protocolo, o cada vez que alguno de los protocolos existentes sufre algún tipo de cambio o modificación. Esto no representa un obstáculo importante en lo que se refiere al desarrollo de protocolos convencionales, puesto que el ritmo de aparición de nuevos protocolos es muy bajo. Sin embargo, estos argumentos no son aplicables cuando se trata de protocolos activos. La propia naturaleza de las redes activas sugiere que el ritmo de aparición de AAs será muy superior al correspondiente a protocolos basados en redes convencionales. En un modelo como éste, caracterizado por su dinamismo, el hecho de tener que atravesar un ciclo completo "*edit-build-run*" (edición-compilación-ejecución) para cada pequeño cambio o modificación supondría un serio perjuicio para la productividad.

Para evitar este problema, hemos elaborado un enfoque alternativo. En lugar de introducir un nuevo tipo de paquete y un nuevo agente para cada AA, hemos desarrollado un único "paquete activo" genérico y un único agente activo, capaz de enviar y recibir estos paquetes. Cada paquete activo transporta el nombre de un procedimiento Tcl que se debe ejecutar en cada nodo activo, así como un campo adicional que las AAs pueden utilizar para almacenar información de naturaleza arbitraria. Cuando el objeto `ANProcessor` detecta la llegada de un paquete activo al nodo, extrae el nombre del procedimiento Tcl asociado, y lo ejecuta. El usuario deberá haber definido este procedimiento en el *script* de simulación. Desde dicho procedimiento, es posible acceder a los servicios exportados por el objeto `ANProcessor`, que desempeña así el papel del EE.

Esta solución presenta diversas ventajas. Al modelar las AAs como procedimientos Tcl, no es necesario recompilar el simulador tras cada cambio o modificación, con lo que el ciclo "*edit-build-run*" anterior se transforma en un ciclo "*edit-run*", con el consiguiente beneficio en cuanto a productividad y facilidad de uso. Además, Tcl es un lenguaje de mucho más alto nivel que C++; como consecuencia, los costes iniciales del proceso de desarrollo son menores, y esto permite un prototipado rápido de las AAs durante las primeras fases de dicho proceso.

3.3 El Entorno de Ejecución

Para permitir el modelado y evaluación de AAs que posteriormente puedan ser implementadas sobre cualquier EE, es necesario proporcionar un modelo de EE *universal*, que permita expresar algoritmos y protocolos funcionalmente equivalentes a los que se podrían desarrollar sobre cualquier otro EE. Según la clasificación expuesta en la sección 2.1, el modelo más genérico y flexible es el correspondiente a un EE que presenta un interfaz Turing-completo, no restringe el acceso de las AAs a los recursos y servicios del nodo y permite el almacenamiento de información de estado, y la granularidad de control es la más fina posible, es decir, cada paquete activo puede estar asociado a un conjunto de instrucciones potencialmente diferentes.

La herramienta ANTS [9] implementa un EE de estas características. En ANTS, cada paquete activo es una *cápsula* que transporta o referencia un miniprograma Java, que se ejecuta en cada nodo activo. Se trata por tanto de un interfaz Turing-completo, puesto que el EE en el que se evalúan las cápsulas es una máquina virtual Java aumentada con una serie de clases que proporcionan servicios adicionales. ANTS también exporta un API completo y flexible para acceder a los recursos del nodo, y permite a las cápsulas almacenar información de naturaleza arbitraria en los mismos. La granularidad de control en ANTS es la más fina posible: cada paquete o cápsula puede estar asociado a un programa potencialmente distinto.

Debido a estas características, y a la gran popularidad de ANTS, hemos diseñado nuestro EE tomando esta arquitectura como referencia. Cabe destacar que esto no implica ninguna dependencia intrínseca entre nuestro modelo y el de ANTS. Sin embargo, este enfoque proporciona dos ventajas importantes. En primer lugar, muchos autores y grupos de investigación ya han trabajado con esta herramienta, demostrando indirectamente la flexibilidad del modelo y su capacidad para expresar una amplia variedad de servicios y protocolos de red. En segundo lugar, existe ya un buen número de AAs escritos para la plataforma ANTS, y en muchos casos el código está disponible públicamente. Al proporcionar un EE equivalente, facilitamos el modelado de estas AAs en nuestro sistema; esto constituye una validación adicional de nuestro diseño y permite demostrar de forma sencilla el uso del mismo para la simulación de protocolos activos.

En la tabla 1 se muestra un subconjunto del API exportado por nuestro EE a las AAs que se ejecutan en el mismo. Las similitudes con el API de ANTS son evidentes, y en la mayor parte de los casos existe una correspondencia total entre ambos. Hemos identificado tres grandes categorías o grupos de operaciones, que se describen a continuación.

1. *Manipulación de la cápsula.* Estas operaciones permiten a las AAs consultar y/o modificar campos de la cabecera del paquete, tales como las direcciones de origen y destino (y a través de esta última, el encaminamiento de la cápsula), atributos del paquete, tales como su tamaño, y también, mediante la operación `setData`, cualquier información de estado o datos de naturaleza arbitraria transportados en el paquete.

2. *Operaciones de control.* A través de estas operaciones, una cápsula puede crear nuevas cápsulas o solicitar su propia eliminación.

3. *Acceso al entorno y almacenamiento*. Estas operaciones permiten a las AAs obtener información acerca del entorno en el que se está evaluando una cápsula, o bien almacenar y recuperar información de estado en la memoria del nodo activo.

3.4 Evaluación de prestaciones

Los experimentos de simulación están limitados en complejidad y escala por la disponibilidad de recursos computacionales, en particular la memoria física y la capacidad de proceso. Para evaluar la degradación de prestaciones introducida por nuestras extensiones, hemos realizado diversos experimentos destinados a medir los incrementos en el consumo de memoria y en el tiempo total de simulación. Todos los resultados se han obtenido sobre un Pentium-II 266 MHz con 256 MB de RAM, usando ns-2.1b6a bajo Linux (*kernel* 2.2.5-15).

Tabla 1: subconjunto del API exportado por el EE

Manipulación de la cápsula	Operaciones de control	Acceso al entorno / almacenamiento
`getSrc`	`sendto`	`getAddr`
`getDst`	`discard`	`getTime`
`setDst`		`getNeighbors`
`getSize`		`get`
`setSize`		`put`
`setData`		`remove`

En principio, cabe esperar un incremento en el consumo de memoria debido a la mayor complejidad estructural de los nodos activos. El incremento total dependerá por tanto del número de nodos activos en la topología de la red. Para evaluar este coste, hemos medido el consumo de memoria asociado a la construcción de topologías activas (todos los nodos son activos) y no activas (ningún nodo activo) de entre 25 y 500 nodos. Como se muestra en la fig. 2, las diferencias son prácticamente despreciables.

La evaluación de las AAs en cada nodo activo tiende a aumentar el tiempo total de simulación respecto al caso convencional. La magnitud de este incremento será proporcional al número de nodos activos en la *ruta promedio* recorrida por los paquetes activos. El incremento por cada nodo se debe, a su vez, a dos contribuciones diferentes: (i) un *overhead* fijo, inherente al modelo, que resulta de los propios mecanismos de detección, intercepción y evaluación de paquetes activos, y (ii) un coste adicional que depende de la complejidad de la AA en cuestión.

Para evaluar el coste fijo inherente al modelo, hemos simulado la transmisión de una serie de paquetes activos simples que se limitan a transportan datos extremo a extremo en modo datagrama, sin realizar ningún procesamiento adicional en los nodos intermedios. Este servicio representa el equivalente al "RPC nulo" que se usa a menudo para estimar el coste base de implementación de un sistema distribuido. Los tiempos de simulación obtenidos se han comparado con los de experimentos equivalentes en escenarios no activos, en los que se transmiten paquetes UDP convencionales. En la fig. 3 se aprecia que el *overhead* se mantiene de unos límites razonables, con un incremento promedio del tiempo de simulación del 30%. En todos los experimentos existe una única fuente de tráfico que genera 10 paquetes por segundo, con un tiempo total de simulación de 60 segundos. Las pruebas se han realizado para rutas de entre 1 y 50 nodos.

El coste adicional asociado a la complejidad de las AAs depende de diversos factores y resulta bastante más difícil de cuantificar, principalmente debido a la falta de un modelo no activo equivalente que usar como línea de base o referencia. Sin embargo, nuestra experiencia sugiere que los tiempos totales de proceso se mantienen dentro del mismo orden de magnitud que los correspondientes a simulaciones no activas de similar complejidad.

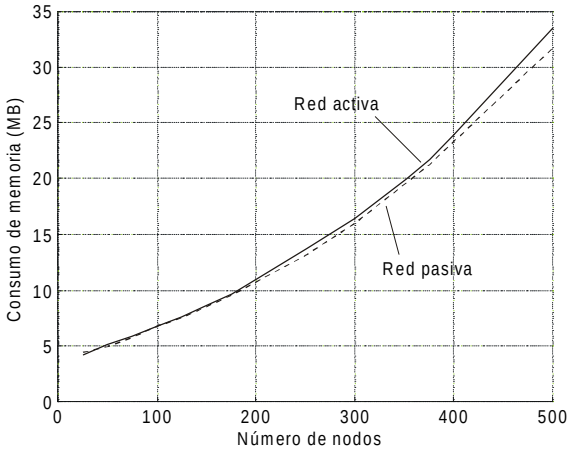


Fig. 2: Consumo de memoria

Resumiendo, esta leve degradación de prestaciones es un precio que en nuestra opinión merece la pena pagar, teniendo en cuenta los beneficios potenciales y facilidad de uso del modelo propuesto. Por supuesto, en aquellos casos en los que la eficiencia es un factor crítico, siempre se puede volver al modelo "tradicional" de desarrollo antes descrito.

4 Aplicación práctica del modelo

En esta sección presentamos algunas AAs que hemos modelado y simulado usando el simulador *ns* y las extensiones aquí descritas.

4.1 Protocolos de ejemplo

Durante el desarrollo de este proyecto, hemos ido implementando una serie de AAs para verificar el correcto funcionamiento del sistema, y al mismo tiempo demostrar el modelado de protocolos activos en el marco de la solución propuesta.

Algunas de estas AAs implementan servicios casi triviales, como por ejemplo el protocolo activo `ping`, que corresponde a las funciones *echo request* y *echo reply* del protocolo ICMP, o los protocolos activos `record` y `trace`, el primero de los cuales equivale a la opción *record route* del datagrama IP, y el segundo funciona de forma similar al programa *traceroute* disponible en la mayoría de los sistemas operativos.

Otras AAs corresponden a servicios más complejos, como es el caso de los protocolos `auction`, `mobile` y `multicast`, que implementan, respectivamente, un servicio de subastas *online*, un servicio para *hosts* móviles similar a MobileIP y un servicio de transporte multicast no fiable. En los tres casos se trata de adaptaciones directas de protocolos activos públicamente disponibles que fueron originalmente implementados sobre ANTS [9].

En [10] se proporciona una descripción detallada del funcionamiento de algunas de estas AAs, incluyendo el código correspondiente al modelado de las mismas sobre nuestro sistema.

A modo de ejemplo, en la fig. 4 se reproduce el código correspondiente al modelado de los dos tipos de cápsulas que constituyen el protocolo activo `mobile`. Cuando un MH (*mobile host*) abandona su estación base (HA, *home agent*), y a medida que se va desplazando por la red, envía periódicamente cápsulas `register` en dirección al HA. La función de estas cápsulas es instalar en determinados nodos intermedios (FAs, *foreign agents*) una serie de punteros que indican la posición actual del *host*. Para comunicarse con el MH, el resto de los *hosts* envían cápsulas `data` en dirección al HA. Si el MH no se encuentra en su base, estas cápsulas irán siguiendo de forma automática los punteros instalados por las cápsulas de registro, hasta llegar a la posición actual del MH. El funcionamiento de este protocolo se describe con mayor detalle en [9].

4.2 Un caso de estudio: el protocolo ARS

Las extensiones descritas en este artículo vienen motivadas, en parte, por el desarrollo del protocolo ARS (*Active Reservation System*): un protocolo orientado a agencias de viaje, que proporciona mecanismos para la interacción cliente / servidor en forma de consultas y respuestas, y se basa en el uso de cachés activas distribuidas a lo largo la red para optimizar estas operaciones.

La filosofía básica de ARS es similar a la del protocolo de *stock quotes* desarrollado en el MIT [11]. Un ejemplo de aplicación de ARS podría ser un sistema de información de líneas aéreas, en el que los usuarios (las agencias de viaje) solicitan información acerca de la disponibilidad de un cierto conjunto de vuelos. En los sistemas actuales, la información solicitada se ensambla en el servidor en forma de página web, y se envía esta página al cliente. Sin embargo, las estrategias convencionales de caché web no son aplicables a este problema, ya que: (i) se trata de información dinámica, en continuo proceso de cambio, y (ii) incluso aunque haya ciertos elementos muy populares, cada cliente puede solicitar una combinación diferente, con lo que las tasas de acierto en caché serían muy bajas.

La solución propuesta en [11] consiste en almacenar la información en las cachés con la mínima granularidad posible, es decir, en lugar de almacenar respuestas completas, almacenar por separado cada uno de los elementos que las componen. Así, los elementos más populares siempre registrarán mayores tasas de acierto, independientemente de la combinación solicitada por cada cliente.

Debido al carácter dinámico de la información, los clientes deben especificar, con cada consulta, el periodo máximo de antigüedad admisible en la respuesta. Lógicamente, las cachés más cercanas al servidor dispondrán de información más actualizada, pero también estarán a mayor distancia. Esto permite a los clientes controlar, para cada operación, hasta qué punto están dispuestos a renunciar a cierta

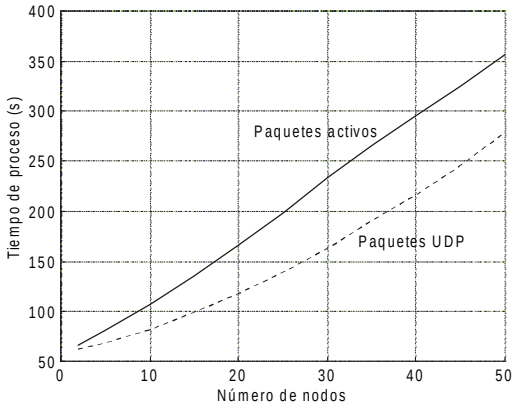


Fig. 3. Tiempo de proceso

precisión en los datos a cambio de mejores tiempos de respuesta. Por ejemplo, una consulta informativa sobre vuelos disponibles hacia un determinado destino puede ser válida incluso aunque los datos tengan varios minutos de antigüedad, mientras que una comprobación previa a una operación de reserva requerirá información totalmente actualizada.

Sobre estas ideas básicas, ARS añade una serie de características adicionales, como la implementación de un sistema de cachés tipo "*modulo caching with workaround*" [4]. En un escenario realista, se espera que el universo de elementos diferentes sea mucho mayor que el espacio disponible en los nodos activos para el almacenamiento de los mismos. La estrategia *modulo caching* consiste en almacenar cada elemento sólo en uno de cada *n* nodos, y hacerlo de tal forma que los distintos elementos queden uniformemente distribuidos a lo largo de toda la topología de la red. Además, cada vez que un nodo almacena un objeto en la caché, envía un mensaje *broadcast* de alcance limitado a otros nodos cercanos, que toman nota de la información. De esta forma, aquellos nodos que no tienen almacenado un determinado elemento, sabrán al menos dónde encontrarlo (*lookaround*).

También es posible generalizar las operaciones de consulta antes descritas para que realicen *búsquedas* de múltiples elementos. En ese caso, los clientes inyectan en la red paquetes activos especiales que contienen el conjunto de parámetros de la búsqueda, y estos paquetes viajan por la red recolectando información sobre los elementos solicitados.

Pese a que no se incluye aquí un modelo completo ni una descripción detallada del protocolo, lo cierto es que los buenos resultados obtenidos se deben a la disponibilidad de una herramienta apropiada para el modelado y simulación de protocolos activos.

5 Trabajos relacionados

La falta de herramientas de simulación para redes activas ha obligado a diversos autores a introducir modificaciones sustanciales sobre herramientas existentes. Por ejemplo, en [2, 5, 11] se menciona el uso de versiones modificadas del simulador *ns*. Sin embargo, en todos los casos se trata de cambios y modificaciones introducidos *ad hoc*, para cubrir las necesidades de cada problema particular, y no para proporcionar una herramienta genérica.

El grupo de investigación del proyecto PANAMA [3] también está utilizando *ns* para analizar el problema del transporte multicast fiable a gran escala en el ámbito de las redes activas. Este grupo ha publicado un paquete con todos los cambios y modificaciones introducidos en el simulador durante el desarrollo del proyecto. Si bien se hacen ciertas referencias a la posibilidad de usar este paquete para la simulación de cualquier tipo de protocolo activo, en realidad las modificaciones están claramente orientadas a la simulación de protocolos multicast en general, y a su propio protocolo (AER) en particular. Por tanto, pese a que se trata sin duda alguna de una loable iniciativa, no cubre las necesidades antes descritas.

El único simulador para redes activas del que se tiene constancia hasta la fecha es AN-Sim, desarrollado en el Instituto Tecnológico de Georgia como una herramienta de apoyo para sus propios proyectos internos [4]. Pese a que los pocos datos disponibles son bastante interesantes, el simulador no está públicamente disponible y todo parece indicar que se trata de una versión preliminar y aún en pleno proceso de desarrollo.

6 Conclusiones

En este artículo hemos presentado una arquitectura diseñada para extender el conocido simulador *ns*, añadiendo soporte para redes activas. Nuestro trabajo viene motivado por la falta de herramientas de simulación en este campo.

Las extensiones desarrolladas proporcionan todos los elementos necesarios para el modelado y la simulación de protocolos sobre redes activas. Entre estos elementos se incluye la infraestructura necesaria para el encaminamiento y procesado de paquetes activos, una serie de mecanismos para el modelado de AAs, y un EE universal que proporciona a las AAs acceso a servicios y recursos del nodo activo.

Nuestro sistema es versátil y flexible, permitiendo la simulación de redes híbridas con nodos activos y *routers* convencionales, así como la simulación de múltiples protocolos simultáneamente. La facilidad de uso ha sido uno de los objetivos principales. Al modelar las AAs como procedimientos Tcl, hemos convertido el ciclo "*edit-build-run*" característico del modelo tradicional de desarrollo en un ciclo simplificado "*edit-run*", lo que sin duda contribuirá a aumentar la productividad. A cambio, se produce una ligera degradación de las prestaciones. El consumo de memoria prácticamente no varía, pero los tiempos de simulación son mayores. Sin embargo, estamos convencidos de que este es un precio aceptable. Por último, se han evitado las dependencias a nivel de implementación respecto a los detalles internos de

funcionamiento del simulador, y no se requieren modificaciones en el núcleo de la herramienta, ni en ninguno de los componentes básicos de la misma.

Al proporcionar un EE muy similar al de la arquitectura ANTS, hemos podido convertir un buen número de AAs con gran facilidad, lo cual constituye una validación adicional de nuestro modelo. Sin embargo, no existe ninguna dependencia intrínseca entre nuestro sistema y la arquitectura ANTS. Nuestro modelo permite el diseño y evaluación de AAs independientemente del EE sobre el que se vayan a implementar posteriormente.

Nuestra experiencia sugiere que estas extensiones estimularán la experimentación y contribuirán a acelerar el desarrollo de nuevos protocolos en el apasionante campo de las redes activas. Cualquier información acerca de los últimos avances y resultados obtenidos en este proyecto se publicará en: http://www.lcc.uma.es/~gisum/active.html

Agradecimientos

Este trabajo se ha beneficiado en gran medida de las discusiones mantenidas en la lista de correo del simulador *ns*. En particular, queremos agradecer a Lidia Yamamoto, de la Université de Liège (ULg), Bélgica, su ayuda y comentarios.

Referencias

[1] D. L. Tennenhouse and D. Wetherall, “Towards an Active Network Architecture”, Proc. Multimedia Comp. and Networking ’96, MMCN ’96, San Jose, CA, enero 1996.

[2] L. H. Lehman, S. J. Garland, D. L. Tennenhouse, “Active Reliable Multicast”, Proc. IEEE INFOCOM ’98, San Franscisco, CA, marzo 1998

[3] Sneha K. Kasera et al., “Scalable Fair Reliable Multicast Using Active Services”, University of Massachussets CMPSCI Technical Report TR 99-44, agosto 1999. Software y documentación en http://www.tascnets.com/panama/

[4] S. Bhattacharjee, K. L. Calvert, E. W. Zegura, “Self-Organizing Wide-Area Network Caches”, IEEE INFOCOM '98, San Francisco, CA, marzo 1998

[5] T. Faber, “ACC: Using Active Networking to Enhance Feedback Congestion Control Mechanisms”, IEEE Network, 12/3, mayo/junio 1998.

[6] Sandeep Bajaj et al., “Improving Simulation for Network Research”, USC Computer Science Department Technical Report 99-702b, septiembre 1999.

[7] K. Fall, K. Varadhan, Editors, “The *ns* Manual”, mayo 2001. Software y documentación disponible en http://www.isi.edu/nsman/ns/

[8] K. Calvert, Editor, “Architectural Framework for Active Networks”, Draft, AN Architecture Working Group, julio 1999.

[9] D. Whetherall, J. V. Guttag, D. L. Tennenhouse, “ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols”, OPENARCH ’98, 1998. Software y documentación disponible en http://www.tns.lcs.mit.edu/activeware

[10] G. Rodríguez, P. Merino, “Modeling and Simulation of Active Network Protocols”. Proceedings of the IEEE ICDCS International Workshop on Internet (IWI’2000). Taipei, Taiwan, abril 2000.

[11] U. Legedza, D. Wetherall, J. Guttag, “Improving the Performance of Distributed Applications Using Active Networks”, Proc. IEEE INFOCOM ’98, San Franscisco, CA, marzo 1998.

```
# Mobile register capsule
#
# On entry, 'data' contains three arguments:
#   home = "home" node (HA)
#   forward = next address to be registered
#   mobileid = id of the mobile host
#
# Initially sent towards "foreign" node (FA)
#
proc register { data ee } {
  set home [lindex $data 0]
  set forward [lindex $data 1]
  set mobileid [lindex $data 2]

  # go to foreign, then home
  if { [$ee getAddr] == [$ee getDst] } {
    $ee put $mobileid $forward $MOBILE_TTL
    # if we're at home, discard, else head
    # for home
    if { [$ee getAddr] == $home } {
      $ee discard
    } else {
      # update the 'forward' parameter
      $ee setData "$home [$ee getAddr]
                   $mobileid"
      $ee setDst $home
    }
  }
}


# Mobile data capsule
#
# On entry:
#   mobileid = id of the mobile host
#
proc data { mobileid ee } {
  # look up forwarding record
  set f [$ee get $mobileid]

  # if found, update our route
  if { $f != "" } {
    $ee setDst $f
  }
}
```

Fig. 4: Implementación de `mobile`

Sistema de evaluación docente a través de la red mediante Active Server Pages

JORGE LÁZARO LAPORTA, OSCAR LOZANO GARCÍA
Área de Ingeniería Telemática, Departamento de Comunicaciones
Escuela Politécnica Superior de Alcoy – Universidad Politécnica de Valencia
Plaza Ferrándiz y Carbonell nº2, 03801 Alcoy, Alicante. España
E-Mail: jlazaro@dcom.upv.es, oslogar@latinmail.com

Abstract:

This paper describes a system providing tools for the examination of students through the Net (either the Internet or an Intranet). Such tools are based on the powerful characteristics of ASP (Active Server Pages) technology. The included applications manage the activity of both teachers and students in a way that there is no need for external utilities apart from the browser itself.

1 Introducción

La informática está cada día más presente en todas nuestras actividades cotidianas, desde el trabajo al ocio pasando por todos los sectores, tanto económicos como sociales; y esto también incluye el área de la enseñanza donde resulta inevitable la inclusión de las herramientas informáticas en la mejora de la actividad educativa no sólo desde el punto de vista de las facilidades que ofrece sino también por lo atractivo e interesante que resulta a los alumnos todo lo que concieme a esta disciplina.

Pero entrando ya en un nuevo milenio, no podemos atender sólo al fenómeno de la informática como el disponer de un ordenador aislado para uso personal. Ha surgido algo que, aun habiéndose de basar en la propia informática, está llamado a ser algo mucho más grandioso para la humanidad: Internet, por supuesto. Un concepto tan interesante y prometedor que gana adeptos a un ritmo exponencial, de tal forma que no podemos obviarlo y quedamos obligados a considerarlo como la parte importante que va a ser en nuestro entorno.

En este marco social y con una vorágine tecnológica que destruye toda actividad o procedimiento que parezca tedioso o poco productivo por antiguo, este método afronta la renovación tecnológica de una actividad que ha sido, es y será cotidiana a la par que trascendental para un relevante grupo social, el colectivo de estudiantes y profesores. Esta actividad no es otra que la evaluación de conocimientos por medio de exámenes escritos.

La intención de este sistema es la elaboración de una herramienta que permita la realización de los exámenes involucrando dos conceptos que ya se han señalado como imprescindibles: la informática e Internet. Así, el resultado final permitirá a los alumnos hacer exámenes desde un ordenador que accederá vía web a los contenidos necesarios, por un lado, y a los profesores preparar dichos exámenes y consultar los resultados, por otro.

En una sociedad que cada día tiene más prisa y demanda mayor comodidad en todas las tareas que el individuo debe afrontar, resulta inexcusable la adaptación de la actividad educativa al mundo de la informática. Por ello se exige que actividades como la realización de exámenes presenten la pantalla del ordenador como interlocutor.

De igual manera se impone una comodidad que implica la desaparición de las distancias a través del entramado de las redes, tanto en Internet como en una Intranet.

Otra consideración de trascendencia sobre el entorno actual, en el que se enmarca este sistema, es la disponibilidad de nuevas herramientas de desarrollo muy enfocadas al campo de las aplicaciones de Internet, como es el caso de ASP, la cual abre nuevas vías de interés en el ámbito que abordamos.

Aprovechando las ventajas que ofrece una tecnología como la anterior se facilita la continuidad de las citadas tendencias: la informatización de la mayoría de las actividades y, al mismo tiempo, posibilitar el acceso a las mismas a través de una red de área local (Intranet) o global (Internet).

El sistema que nos ocupa trata de cumplir con los mencionados objetivos mediante el uso de la tecnología ASP en el ámbito de la docencia tradicional o de la educación online, y más concretamente en los procesos de evaluación de alumnos a través de exámenes escritos (Fig. 1). Se permite a los alumnos, por un lado, demostrar sus conocimientos desde un ordenador que accede vía web a los contenidos correspondientes, y por otro, a los profesores preparar dichos exámenes y consultar los resultados correspondientes.

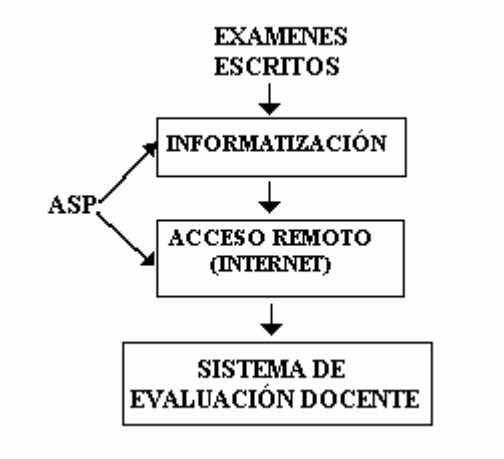


Figura 1. Fusión de elementos que originan el sistema

Así, los objetivos del sistema enumerados a grandes rasgos son:

- Acceso vía web de alumnos y profesores.
- Visualización del examen y recogida de resultados a cargo del sistema.
- Flexibilidad máxima de conformación del examen para el profesor.
- Seguridad en los intercambios de datos y en la identificación de alumnos.

2 Active Server Pages

Microsoft introdujo esta tecnología llamada ASP en diciembre de 1996, por lo que hoy en día no es nada nueva. Es parte del Internet Information Server (IIS) desde la versión 3.0 y es una tecnología de páginas activas que permite el uso de diferentes scripts y componentes en conjunto con el tradicional HTML para mostrar páginas generadas dinámicamente.

Este tipo de páginas se identifican rápidamente al observar que tienen por extensión ".asp".

Traduciendo la definición de Microsoft: "Las Active Server Pages son un ambiente de aplicación abierto y gratuito en el que se puede combinar código HTML, scripts y componentes ActiveX del servidor para crear soluciones dinámicas y poderosas para la web".

El ASP es una tecnología dinámica funcionando en el lado del servidor, esto significa que cuando el usuario solicita un documento ASP, las instrucciones de programación dentro del script son ejecutadas para enviar al navegador únicamente el código HTML resultante.

La ventaja principal de las tecnologías dependientes del servidor radica en la seguridad que tiene el programador sobre su código, ya que éste se encuentra únicamente en los archivos del servidor que al ser solicitado a través del web, es ejecutado, por lo que los usuarios no tienen acceso más que a la página resultante en su navegador.

El desarrollo que se ha venido dando en lo que se refiere a ASP ha sido bastante amplio. Entre sus funciones principales están el acceso a base de datos, envió de correo electrónico, creación dinámica de gráficos y otros.

Básicamente, muchas cosas que podemos realizar por medio de CGI pueden ser realizadas con esta tecnología. Esto es debido a que el ASP es tan eficiente que le basta con escribir código directamente a la interfaz de aplicación del servidor, con la ventaja de que es más eficiente que el CGI que depende de un compilador ya que el ASP corre como un servicio en el servidor, tomando ventaja de la arquitectura de multitareas.

La creación de páginas ASP es tan sencilla como la propia edición de código HTML en el que se insertan instrucciones ASP encerrándolas entre "<% %>". Estos comandos son los que procesa el servidor antes de enviar la página al navegador.

El principio de la tecnología ASP es el VBScript, pero existe otra diversidad de lenguajes de programación que pueden ser utilizados como lo es Perl, JScript, etc. Así pues se pueden emplear todas las instrucciones de control que estos lenguajes ofrecen.

La escritura sobre la página HTML enviada al usuario se realiza, bien escribiendo directamente sin limitar entre <% %>, o bien utilizando el objeto "Response" que incorpora ASP sin necesidad de crearlo.

Inevitablemente contempla también el envío de información y éste no puede hacerse de mejor manera que a través de formularios. Los datos recogidos son procesados en el servidor mediante los scripts incluidos en las páginas ASP de manera que se pueda generar una página de respuesta de acuerdo con la información obtenida.

Para el manejo de la información emitida por el usuario, las páginas ASP manipulan otro objeto incorporado como es el "Request".

Un tercer objeto incorporado es el "Server" el cual contiene varios métodos y propiedades del servidor.

Finalmente, los dos últimos objetos incorporados de interés son "Application" y "Session". Ambos mantienen información desde que se crean al ejecutar la primera página ASP en un sitio hasta que el

usuario abandona el lugar. La diferencia estriba en que el objeto Application contiene valores útiles para todos los usuarios, por lo que existe desde que entra el primero hasta que sale el último. Por el contrario, el objeto Session sólo existe en el ámbito de un usuario concreto pues su existencia depende exclusivamente de la permanencia de éste y su contenido se centra en datos personales del mismo, siendo inaccesible a otros usuarios.

Un tipo peculiar de archivo ASP es el "Global.asa". Presenta una extensión distinta porque también lo es su filosofía. Este archivo debe ser único en un directorio virtual. Se ejecuta automáticamente cuando el usuario solicita por vez primera una página ASP de cierto directorio. Su finalidad suele ser la creación e inicialización de diversas variables que existirán durante toda la actividad del usuario en el sitio. Estas variables quedaran circunscritas, bien al ámbito del objeto Application, bien al del objeto Session, por lo que su existencia está ligada a la de dichos objetos.

Un aspecto imprescindible a analizar respecto a ASP es la posibilidad de acceso a bases de datos. La tecnología ASP no defrauda en este sentido pues aprovecha las grandes ventajas que ofrece la tecnología ADO, con la que es compatible.

ActiveX Data Objects (ADO) es una tecnología ampliable y de fácil uso para agregar acceso a bases de datos a las páginas Web. Se puede utilizar ADO para escribir secuencias de comandos compactas y escalables que conecten con bases de datos compatibles con Open Database Connectivity (ODBC, Conectividad abierta de bases de datos) y orígenes de datos compatibles con OLE DB. Si no se tiene mucha experiencia en conectividad con bases de datos, se encontrará que las instrucciones de ADO son asequibles y no complicadas. Del mismo modo, si ya se tiene experiencia en la programación con bases de datos, se apreciarán las características avanzadas de conexión y de manipulación de consultas independientes del lenguaje de ADO.

3 Estado del arte

La informatización en el proceso de evaluación docente no es, obviamente, una novedad que presente el sistema que se plantea.

Se pueden encontrar algún ejemplo de sistema con una función similar navegando por la red, si bien no están tan orientados específicamente al uso en instituciones como la Universidad, en el que debe haber un control exhaustivo tanto de confidencialidad como de identidad.

Una breve introducción a este tipo de sistemas se puede encontrar en webs dirigidas a la enseñanza de la tecnología ASP y sus aplicaciones.

Por ejemplo, en "asptoday.com" ofrecen un ejemplo de sistema para identificar y evaluar usuarios mediante sencillas páginas ASP. No se trata de un sistema completo y mulifuncional adecuado para uso en evaluaciones rigurosas pero sirve de ejemplo introductorio a la evaluación a través de la red.

Dicho ejemplo ofrece la opción de probarlo viendo el aspecto y posibilidades que permite a su vez disponer del código fuente de manera que pueda servir de punto de partida a interesados en el tema.

Un nivel mayor presenta el sistema desarrollado por **ExamServe [3]**. Esta web ofrece la posibilidad de que una empresa evalúe a su personal – por ejemplo- del modo que se viene explicando.

En primer lugar hay que enviarles las cuestiones que se van a someter a examen para que se almacenen en su sistema. Posteriormente, los usuarios a evaluar se conectan con la página y acceden al examen en cuestión.

El sistema nos permite especificar un gran número de variables referentes al examen que se va a presentar a los usuarios. Éstas abarcan aspectos como el tipo de respuesta que se espera del examinado, la inclusión de archivos de sonido, imágenes y vídeos junto a la pregunta realizada, etc.

Con todas estas características y muchas otras, el citado sistema es un buen punto de referencia para el desarrollo de otros en ámbitos más específicos como pueda ser la Universidad.

Aun siendo un sistema bastante completo, no inhibe el desarrollo del proyecto que nos ocupa pues en ningún caso contempla las particularidades de la evaluación en organismos oficiales – aquella está orientada a la evaluación de empleados en el ámbito de la empresa – y en todo caso conviene que se disponga de un sistema propio, y al mismo tiempo flexible, sobre el que se tenga total control sin necesidad de delegar en entidades ajenas.

4 Desarrollo

El sistema expuesto está constituido principalmente por dos tipos de elementos:

- Las páginas ASP que interactúan con los usuarios y realizan las actividades pertinentes en cada caso.

- La base de datos que mantiene toda la información del sistema, referente tanto a alumnos como a los propios exámenes.

De la interrelación de dichos elementos entre sí y con los dos tipos de usuarios del sistema, entiéndanse alumnos y profesores, surge el modelo que caracteriza el sistema y que se muestra en la Figura 2.

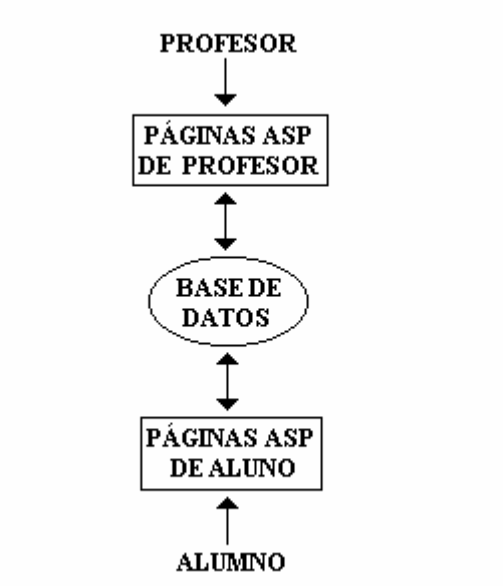


Figura 2. Modelo del sistema

4.1 Páginas ASP

Se pueden dividir en dos grupos atendiendo al tipo de usuario que las va a utilizar:

- Páginas de profesor. Sólo debe poder acceder a ellas este tipo de usuario en tanto que proporcionan las funcionalidades para controlar la información de la base de datos, esto es, desde ellas se determinan todas las características de los exámenes y también se accede a los datos de los alumnos que se examinan (notas, contraseñas,...). El acceso se debe realizar a través de la página principal denominada "profesor.asp" (Fig. 3), dicha página centraliza en un mismo lugar todas las funciones que el profesor puede realizar. Una vez en ella se determina la acción que se pretende llevar a cabo seleccionándola de entre las siguientes:

  - Crear un nuevo examen.
  - Modificar un examen existente.
  - Borrar un examen existente.
  - Renombrar un examen existente.
  - Introducir lista de alumnos susceptibles de ser examinados.
  - Listar las notas de los alumnos.
  - Ver el examen de cierto alumno.
  - Obtener un fichero de alumnos y sus notas.
  - Modificar la apariencia de los exámenes.
  - Obtener un fichero con las contraseñas que el sistema ha asignado a cada alumno.

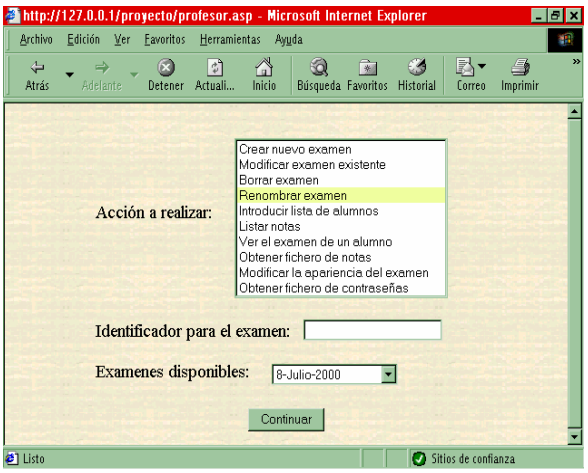


Figura 3: Página principal para la actividad del profesor

Una vez seleccionada la actividad y proporcionada la información adicional que el sistema pueda requerir en función de la acción solicitada, si la propia página tiene capacidad para llevar a cabo la acción, ejecuta la sección de código correspondiente; de lo contrario, carga la página encargada de cumplir la acción, la cual será, a partir de ese momento, la que interactúe con el usuario.

De entre todas las funcionalidades, la de crear/ modificar un examen (Fig. 4) resulta especialmente compleja pues se solicita del profesor no sólo los enunciados de preguntas y respuestas, sino también una serie de características del examen (al margen de la apariencia que se manipula por otra vía) como son su duración máxima, puntuaciones parciales de cada pregunta, puntos que se descuentan al errar contestando, inserción de imágenes auxiliares junto al enunciado de las preguntas, distintas aleatorizaciones o permutaciones en las preguntas y respuestas para dificultar que los alumnos se copien unos de otros,...

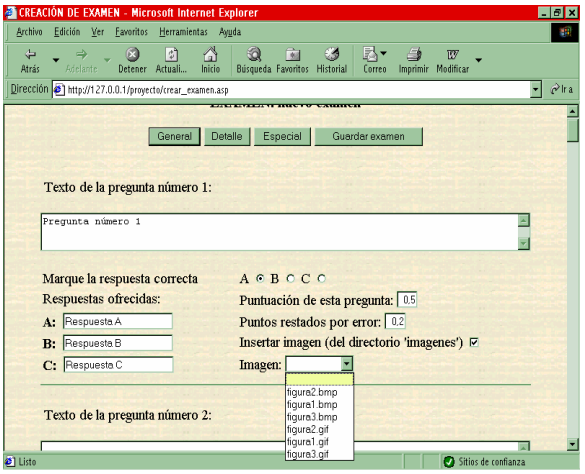


Figura 4: Creación de exámenes

Además, la modificación de ciertas características del examen provoca una regeneración del contenido de la página que se adecua dinámicamente a la nueva forma del examen (por ejemplo, si el tipo de respuesta pasa de V/F a ABCD aparecerán los cuatro campos correspondientes a las cuatro respuestas posibles que se van a ofrecer en detrimento de la opción V o F).

- Páginas de alumno. Están orientadas a dirigir el proceso de la evaluación desde el lado de la actividad que realizan los alumnos. Éstos tan sólo tendrán que cargar una página por si mismos, la cual los irá guiando y redirigiendo a otras páginas cuando sea oportuno durante todo el proceso. Dicha página ("identificacion.asp") se encarga en primera instancia de autentificarlos como alumnos validos mediante una contraseña. Adicionalmente, se controla que el usuario concreto no haya accedido ya al examen, pues se entiende que ya lo ha realizado por lo que se le impide el nuevo acceso. Acto seguido les ofrece un listado de enlaces a los exámenes disponibles de manera que seleccionen el adecuado. Hecha la selección, se carga otra página (Fig. 5) que contiene el examen solicitado con las preguntas, respuestas y demás características que el profesor determinó en el proceso de creación o modificación del mismo. El alumno entonces ha de contestar a las preguntas que se le plantean marcando alguna de las opciones disponibles (incluida la de dejar la pregunta sin contestar) en el plazo máximo (finalizado el tiempo fijado se le recoge el examen automáticamente) que se le indica mediante un cronómetro que permanece visible siempre y mediante avisos periódicos. Entregado o recogido el examen, el sistema calcula la nota en función de las puntuaciones parciales determinadas por el profesor en la creación, e informa de la misma al alumno. Igualmente le ofrece la opción de solicitar copia del examen que recibirá vía e-mail.

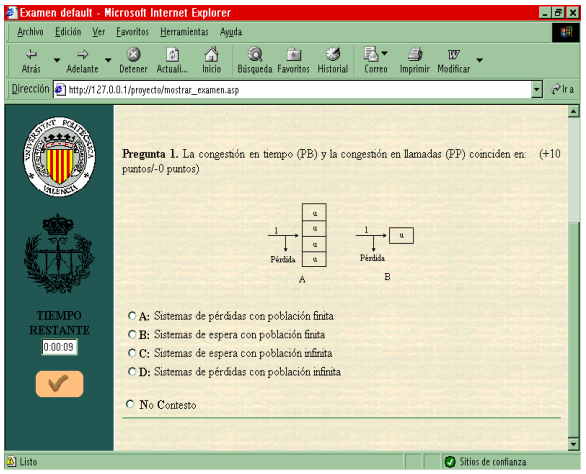


Figura 5: Presentación del examen

4.2 Base de datos

La base de datos (BD) constituye el núcleo del sistema en tanto que el principio básico de la actividad que se está cubriendo es el intercambio de información entre alumnos y profesores; el cual se realiza escribiendo y leyendo de ella.

Sin embargo, ninguna operación la realiza el usuario directamente sobre la base de datos, sino a través de las páginas ASP capacitadas para comunicarse con ella. Esto preserva la integridad de los datos y hace innecesario un conocimiento de la estructura interna de la BD.

Profesores y alumnos escriben y leen de la misma base de datos pero la mediación de las páginas ASP es la que limita el acceso a ciertos contenidos tanto por usuarios no autorizados en ningún sentido (intrusos) como por alumnos que sólo deben conocer ciertos contenidos (obviamente no deben acceder a las respuestas pero sí a las preguntas).

En todo caso, puede ser útil conocer su organización en aras de mejorar la comprensión del funcionamiento del sistema.

La base de datos contiene tres tablas:

- **Tabla de alumnos**: formada por una entrada de registro por cada alumno introducido en el sistema. Para cada uno de ellos se conserva su identificador y contraseña (para su autenticación), nota obtenida, dirección de correo electrónico donde recibir copia del examen (si lo solicita), fecha y hora de entrega de examen, preguntas que se le hicieron y respuestas que eligió (para posteriores revisiones de examen). Para cada proceso de evaluación el profesor ha de regenerar el contenido de esta tabla mediante la pertinente inclusión del listado de alumnos y la posterior generación de sus contraseñas, de manera que alumnos de evaluaciones posteriores vean imposible su acceso y se evite que el tamaño de la tabla crezca indefinidamente en sucesivas evaluaciones.

- **Tabla de exámenes**: no se organiza por exámenes, sino que cada entrada representa una pregunta de un examen, si bien se incluye un campo identificador que indica a que examen pertenece. Adicionalmente, de cada pregunta se conserva su enunciado, las respuestas posibles, la respuesta correcta, puntuaciones positivas y negativas, imagen auxiliar incluida (si la hubiera), además de otros campos cuyos valores coinciden en todas las de un mismo examen por tratarse de características genéricas del mismo, como la duración máxima permitida, las aleatorizaciones, etc. Las entradas permanecen en la tabla hasta que se solicite la eliminación del examen al que pertenecen o se modifique dicho examen eliminando la pregunta concreta. Si nada de esto ocurre, las entradas se

mantienen indefinidamente con el consiguiente crecimiento de la tabla.

- **Tabla de apariencia**: en una única entrada se guardan los valores correspondientes a las determinaciones del profesor en lo referente al examen que se presenta al alumno, no desde el punto de vista de los contenidos, sino del aspecto del mismo: imágenes de fondo, iconos, botones,... El número de campos podría extenderse enormemente añadiendo innumerables aspectos a la lista de configurables y extendiendo así la personalización de la apariencia hasta donde se quisiera, si bien el interés productivo de ésta actividad resultaría bastante escasa.

4.3 Seguridad

El sistema responde a las necesidades de seguridad en dos campos distintos: seguridad ante intrusos sin derecho a acceso y seguridad ante irregularidades de los propios alumnos que se han de examinar.

En lo referente a los alumnos que se han identificado correctamente, el primer control que sufren es determinar si ya han accedido al examen anteriormente y quieren volver a hacerlo para modificarlo, en cuyo caso se les rechaza informándoles sobre este punto. Durante el examen propiamente dicho, el profesor puede haber establecido sistemas tradicionales que complican la cooperación entre examinados como son la alteración del orden de las respuestas para cada examen, la reordenación de las propias preguntas o incluso la diversidad de las mismas de uno a otro

El control de intrusos se regula desde el uso de pares usuario/contraseña a los cuales sólo tiene acceso el profesor hasta el mismo instante del comienzo del examen en que informa de ellos a los respectivos alumnos.

Contra la acción de intrusos con mayor conocimiento del funcionamiento de la red, se prevé el blindaje del canal de comunicación mediante sistemas cifrados mediante certificados, protocolos seguros, perfectamente integrables con sistemas tales como Windows 2000 Server.

Otro aspecto no menos relevante es el de la posibilidad de la suplantación de la identidad (in situ) sobre la máquina sobre la que se hace el examen. Estos sistemas posibilitan la autentificación mediante tarjetas inteligentes (smart cards). De todas formas existen sistemas con una mayor seguridad de autentificación, piénsese en sistemas biométricos (control de la huella de la mano, del iris, de la voz,…).

La combinación de todos estos elementos confiere una alta fiabilidad y seguridad al sistema que permite utilizarlo no sólo en entornos de evaluación sino, en definitiva, en cualquier empresa o institución.

5 Conclusiones

El sistema diseñado ofrece los medios para establecer un método de evaluación de alumnos que aproveche las ventajas de la mecanización informática y de la eliminación de distancias gracias a las redes telemáticas.

Se abarcan tanto las necesidades del profesor, diseño de exámenes y acceso a la información derivada de la evaluación automática de los alumnos, como las propias de los alumnos, acceso al examen, realización del mismo y conocimiento de los resultados.

Referencias

[1] Jesús Bobadilla Sancho. “Creación de aplicaciones Web en Windows NT: Active Aerver Pages”. Ra-Ma

[2] J. Manuel Alarcón Aguín. “Programación con VBScript”. Anaya Multimedia

[3] www.examserve.co.nz ,

[4] Óscar Lozano García, J. Lázaro Laporta. “Sistema de evaluación docente a través de la red mediante ASP’s”. PFC, EPSA, UPV. 2001.

Conexiones Robustas para Flujos TCP sobre ATM Mediante un Protocolo Activo en una Arquitectura Multiagente[1]

José Luis González-Sánchez (*) y Jordi Domingo-Pascual (+)
(*) Universidad de Extremadura. Dpto. Informática, Área de Ingeniería Telemática
Escuela Politécnica de Cáceres. Avda. Universidad S/N. (10.071) Cáceres
E-Mail: jlgs@unex.es
(+) Universitat Politécnica de Catalunya.
Campus Nord, Modul D6. Jordi Girona 1-3 (08.034) Barcelona
E-Mail: jordi.domingo@ac.upc.es

Abstract. TAP (Trusted and Active PDU transfers) is a new distributed architecture and protocol for ATM networks that provides assured transfers to a set of privileged VPI/VCI. The distributed architecture manages the privileged connections and offers an improvement in the performance when network connections cause some cell loss by taking advantage of the idle time in the traffic sources to carry out the retransmissions. The trusted protocol is supported by our AcTMs (Active ATM switch) model. The protocol and architecture also offers an attractive solution to the chaotic nature of TCP Congestion Control. Several simulations demonstrate the effectiveness of the mechanism that recovers the congested PDU locally at the congested switches with better end-to-end goodput in the network. Also, the senders are alleviated of NACK and end-to-end retransmissions. TAP is an active and distributed architecture in the sense that our protocol implements several active coordinated and self-collaborative software and programmable agents.

1 Introducción

La tecnología ATM se caracteriza por su buen comportamiento ante diversos tipos de tráfico y por su capacidad de negociación de los parámetros de QoS (*Quality of Service*) [1]. Las congestiones son el tipo de errores más habitual y es aquí donde situamos este trabajo que ofrece transferencias garantizadas mediante nuestra arquitectura TAP (*Trusted and Active Protocol*). En TAP adoptamos ARQ (*Automatic Repeat Request*) con NACK (*Negative Acknowledgement*) usando células RM (*Resource Management*) para aliviar el efecto de la implosión. Los nodos activos intermedios se encargan de las retransmisiones para evitar las retransmisiones e-e (extremo-extremo). Hemos implementado como esquema de control de congestión una modificación de EPD (*Early Packet Discard*) que denominamos EPDR (*Early Packet Discard and Relay*) para aliviar el efecto de las congestiones y de la fragmentación de las PDU. Actualmente, el control de congestión se delega en protocolos que las resuelven mediante retransmisiones e-e como TCP. Esta es una técnica sencilla de implementar a altas velocidades que simplifica los conmutadores, pero toda la red se ve sobrecargada con las retransmisiones y no aporta protección contra fuentes egoístas. También realizamos asignaciones de ancho de banda justas basándonos en un esquema delegado extendiendo WFQ (*Weighted Fair Queueing*) [2] para reducir su complejidad de implementación y lograr un coste constante *O(1)*.

Actualmente las redes ATM se usan como la tecnología para soportar todo tipo de tráfico, con un destacable predominio de los protocolos TCP/IP. Por esto presentamos los beneficios que este mecanismo de recuperación de congestiones puede aportar, no sólo al trafico ATM nativo, sino también al tráfico generado por las fuentes TCP/IP. El protocolo TCP se ha convertido en los últimos años en el estándar de comunicaciones de datos. Éste es un protocolo fiable de la capa de transporte de la arquitectura TCP/IP, que usa control de error y control de flujo basados en mecanismos de ventana y se encarga del enrutamiento de paquetes en internet con control e-e [3]. Existen numerosas investigaciones para conseguir integrar dos tecnologías tan diferentes como ATM y TCP/IP; sin embargo, la integración de ambas se ha demostrado [9] con un pobre resultado en cuanto al comportamiento del *throughput* de TCP sobre ATM. Mientras ATM es una tecnología orientada a conexión, de conmutación de células de 53 octetos y de tamaño uniforme, TCP e IP se basan en mecanismos de enrutamiento de segmentos y datagramas de tamaño variable.

Las características que acabamos de comentar provocan un efecto bastante devastador en el *throughput* cuando los segmentos TCP atraviesan conmutadores ATM con tamaño de buffer mucho menor que el tamaño de ventana de TCP. Por esta causa las células se pierden y acaban generándose retransmisiones por *timeout*. Además, la pérdida de una sola célula dará como resultado la pérdida de

[1] Este trabajo ha sido patrocinado en parte por el proyecto CICYT Nº TEL99-1117-C03-03.

un segmento TCP en el receptor de la comunicación, por lo que solicitará una retransmisión al emisor que debe encargarse de reenviar nuevamente el segmento completo, en lugar de la célula perdida.

Presentamos una serie de simulaciones que demuestran el caótico comportamiento de TCP cuando se enfrenta a las congestiones. Para ello se usa el simulador NS (*Network Simulator*) [4] con el que se estudia el mecanismo de ventana de TCP, fijándonos en los efectos del umbral, de la ventana de congestión, de la probabilidad de pérdidas de segmentos, y en las consecuencias que todo esto acaba teniendo sobre el *throughput*. Veremos cómo el *goodput* acaba degenerándose cuando aparecen congestiones en los escenarios simulados. Si se introduce TAP en la red conseguiremos mejorar sustancialmente el *goodput* de las transmisiones TCP, ya que se logran reducir (o eliminar) las retransmisiones e-e y, además, se reducen las labores de autoajuste de las ventanas de congestión en las fuentes TCP.

En primer lugar se comentan las características generales de TCP, para pasar después a simular varios escenarios con NS (*Network Simulator*). La sección 3 expone las características de TCP sobre ATM y la siguiente sección propone la inclusión de TAP en un escenario *IPoverATM*. Terminamos el artículo con un apartado de conclusiones.

2 Funcionamiento de TCP

El protocolo TCP es un conjunto de algoritmos que envían paquetes a la red sin ningún tipo de reserva previa, pero que son capaces de reaccionar ante determinados eventos en la misma. Entre esos algoritmos destaca el *Control de Congestión* y el de *Recuperación de Segmentos Perdidos*.

El mecanismo de control de congestión en TCP tiene dos fases diferentes: *Slow Start* y *Congestion Avoidance*. Al iniciar una conexión, o al reiniciarse por el envío de un segmento perdido, el tamaño de la ventana de congestión es puesta a 1 paquete, y después es aumentada al doble en cada ACK recibido desde el receptor en el tiempo RTT. Durante esta fase, CWND es incrementada linealmente, al contrario de su crecimiento exponencial durante la fase *Slow Start*. Los siguientes son los aspectos básicos de funcionamiento de TCP:

- CWND: representa la ventana de congestión, y es una variable que limita la cantidad de datos que TCP puede enviar. Su tamaño varía dependiendo de las condiciones de la red, de forma que si ésta no descarta paquetes por congestión, el tamaño de esta ventana aumenta permitiendo incrementar también la velocidad de transmisión de las fuentes de tráfico.
- INITIAL_WINDOW: es el valor con que se inicia la ventana de congestión CWND.
- SMSS: expresa la cantidad máxima de datos que puede enviar una fuente de tráfico TCP.
- RWND: es la cantidad máxima de datos que puede recibir un receptor de tráfico TCP.
- RTT: es el *Round Trip Time*; es decir, el tiempo que transcurre desde que un segmento sale del emisor hasta que éste recibe la confirmación desde el receptor. En realidad, el RTT determina la velocidad de transmisión de TCP, ya que el emisor envía cada RTT el tamaño de datos fijado por la ventana CWND.
- CURRENT_WINDOW: representa la cantidad de información que envía el emisor cada RTT. Esta ventana toma como valor el más pequeño de CWND o RWND.
- SSTHRESH: determina qué algoritmo de control de congestión de los ya comentados se debe usar. Cuando CWND < SSTHRESH se emplea el algoritmo *Slow Start*, mientras que cuando CWND ≥ SSTHRESH se aplica el control de congestión determinado por *Congestion Avoidance*.

Por tanto, una fuente TCP establece la cantidad de datos que envía usando la ventana CWND y transmite una ventana de segmentos por cada RTT. TCP ajusta el tamaño de dicha ventana dependiendo de las condiciones de la red. Así, el tamaño de CWND se incrementa en el doble de segmentos por cada ACK recibido si estamos en *Slow Start*, y se incrementa en 1/CWND por cada ACK recibido en *Congestion Avoidance*. Todo esto ocurre en una conexión en la que no hay descartes de segmentos, y lo ilustramos con la *Fig. 1*, donde puede observarse la ventana de congestión sin pérdidas ya comentada. CWND aumenta exponencialmente mientras su tamaño es menor que SSTHRESH, esto es debido a que se está usando el algoritmo *Slow Start* que aumenta progresivamente el número de segmentos (1, 2, 4...) a medida que se van recibiendo los ACK. Cuando el tamaño de CWND se iguala al valor de SSTHRESH entra en acción el control de congestión de *Congestion Avoidance*. A partir de este momento la ventana incrementa en 1/CWND por cada ACK, dando lugar a un crecimiento lineal de la ventana CWND.

El algoritmo *Slow Start* es usado por TCP para probar la capacidad de la red (de la que desconoce su capacidad) y la cantidad de segmentos que puede soportar sin congestionarse. Cuando se acerca una posible congestión, TCP cede control a *Congestion Avoidance* que pasa al incremento lineal de CWND hasta que la congestión es detectada.

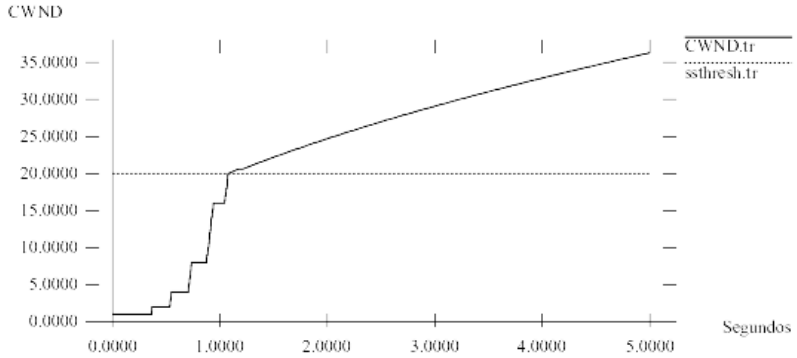


Figura 1. Evolución de CWND sin pérdidas

2.1 Reacción ante congestiones

Cuando la capacidad de transmisión de la red es inferior a la cantidad de información que desea transmitirse, la red comenzará a descartar segmentos para que los emisores disminuyan el volumen de información generada. TCP detecta que un segmento ha sido descartado por congestión si el número de ACK repetidos recibidos es 3, o cuando expira el *timeout* de las retransmisiones. TCP reacciona reiniciando SSTHRESH a la mitad de la ventana y reduce CWND al valor determinado por INITIAL_WINDOW, disminuyendo así la cantidad de segmentos que envía.

La *Fig. 2* representa esta situación donde puede observarse la evolución de CWND con pérdidas en la red debidas a congestiones. Tanto la *Fig. 1* como la *Fig. 2* se han obtenido con la simulación de la misma topología sobre NS. Esta topología consta de 7 nodos. Cada uno de los enlaces tiene un ancho de banda de 1 Mbps, un *delay* de 10 ms y usan *DropTail* como tipo de cola.

La diferencia de los dos escenarios simulados está en que en el segundo se ha introducido la probabilidad de pérdida mediante modelos de error que nos han permitido definir una probabilidad de pérdida de 0,02 en los enlaces 2-3 y 3-4 de la topología anterior. En el caso de la *Fig. 2* se ha empleado un tiempo de simulación superior (20 s.) para comprobar con claridad el efecto de las pérdidas sobre la ventana de congestión que es reducida a 1 en múltiples ocasiones para solventar el problema de las congestiones. Destacamos que el protocolo TAP propuesto intenta solventar estos problemas de pérdidas que afectan, tanto a la reducción de la ventana, como a la posterior retransmisión de las pérdidas e-e. Así, la fuente no se verá obligada a reducir su velocidad de envío tan a menudo como muestra la *Fig. 2* y, sobre todo, cuando aparezcan las congestiones éstas son resueltas localmente entre los nodos afectados.

2.2 *Throughput* y pérdidas

Suponiendo que el tamaño máximo de la ventana CWND es de *W* segmentos y, según la definición que hemos hecho del algoritmo *Congestion Avoidance*, en [5] se deduce que el total de datos entregados a la red en cada ciclo puede ser calculado por la expresión

$$\left(\frac{W}{2}\right)^2 + \frac{1}{2}\left(\frac{W}{2}\right)^2 = \frac{3}{8}W^2 \qquad (1)$$

Supóngase ahora una red sin pérdidas con un RTT constante porque tiene suficiente ancho de banda y con una baja carga total que nunca llene las colas. Pues bien, según [5] puede aproximarse la pérdida de paquetes aleatoria mediante una probabilidad constante *P* que asuma que el enlace entrega aproximadamente *1/P* paquetes consecutivos seguidos de un descarte, todo esto sin considerar los datos que se transmiten durante las recuperaciones de paquetes.

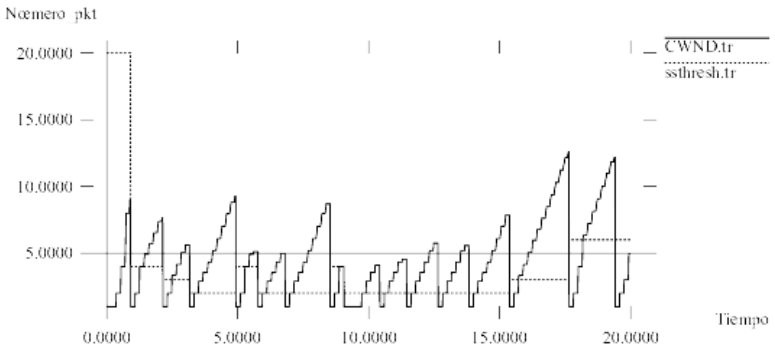


Figura 2. CWND y SSTHRESH en una red congestionada

Tenemos por tanto dos aproximaciones a la entrega de paquetes, la expresión (1) y *1/P*, por lo que uniendo ambas y despejando *W* obtenemos,

$$W = \sqrt{\frac{8}{3p}} \quad (2)$$

Por otro lado, podemos aplicar estos datos conocidos sobre la siguiente expresión (3) que calcula el ancho de banda (donde *MSS* es el máximo tamaño de segmento TCP) transmitido,

$$AB = \frac{datos/ciclo}{tiempo/ciclo} = \frac{MSS * \frac{3}{8} W^2}{RTT * \frac{W}{2}} = \frac{MSS/P}{RTT * \sqrt{\frac{2}{3P}}} \quad (3)$$

Se puede reestructurar la expresión (3) agrupando el término constante $K = \sqrt{\frac{3}{2}}$; llegando a,

$$AB = \frac{MSS}{RTT} \frac{K}{\sqrt{P}} \quad (4)$$

La fórmula (4) expresa el ancho de banda de la red y nos sirve así para aproximar de forma sencilla el funcionamiento de TCP tras haber realizado algunas simplificaciones como las explicadas. El artículo [5] presenta otras referencias con diversas aproximaciones para el valor de la constante *K* que, indiferentemente de su valor, puede servir para considerar que ésta es siempre menor que 1, con lo cual podemos quedarnos finalmente con la siguiente fórmula (5) como una buena expresión del *throughput* de TCP,

$$AB < \left(\frac{MSS}{RTT}\right) \frac{1}{\sqrt{P}} \quad (5)$$

Por nuestra parte hemos analizado también el comportamiento de TCP en diversas situaciones y planteamos el comportamiento y efecto del *throughput* estudiado con respecto a la probabilidad de pérdida de paquetes. Para ello reorganizamos la formulación (5) del algoritmo *Congestion Avoidance* de TCP que expresa el comportamiento "*steady state*" de TCP bajo condiciones ligeras de carga y para una pérdida de paquetes moderada en su expresión general,

$$TH = \frac{MSS}{RTT\sqrt{P}} \quad (6)$$

Si en (6) *TH* representa el *throughput* y consideramos constantes los valores de *MSS* y *RTT*, podemos obtener la fórmula (7) donde comprobamos que el *throughput* es inversamente proporcional a la probabilidad de pérdida,

$$TH = \frac{K}{\sqrt{P}} \quad (7)$$

La expresión (7) nos permite estudiar, por tanto, el comportamiento que va a experimentar el *throughput* a medida que se produzcan pérdidas de paquetes en *routers* congestionados.

De la formula (7) puede obtenerse una nueva función que aproxima el tamaño de la ventana *W* que emplea TCP cuando se tiene una velocidad media de pérdidas *P*. Así, ajustando el valor de la constante *K* de la fórmula (4), obtenemos la siguiente expresión que es resultado de la adaptación de (7) en [5] y de las propuestas realizadas en [6],

$$W = \frac{0{,}866}{\sqrt{P}} \quad (8)$$

En nuestro caso particular, para el valor de la constante *K* calculamos $\sqrt{3/4}$ suponiendo que se aplica la estrategia retardada en los ACK en lugar de aplicarla sobre cada uno de los paquetes y, según el planteamiento de pérdidas periódicas, en lugar de partir de un plantemiento de pérdidas aleatorias.

Así, y en línea con [7], podemos comprobar que la ecuación (8) puede verse desde dos puntos de vista diferentes. En primer lugar, la red descarta los paquetes a una velocidad independiente de la actividad del emisor, por lo que la fórmula expresa entonces la forma en que el emisor es capaz de reaccionar. En segundo lugar, si consideramos que la red es capaz de almacenar sólo un número determinado de paquetes, la ecuación puede entenderse como la velocidad que la red debe imponer para que la ventana de TCP quepa en esa capacidad de almacenamiento de la red. Según todo esto, podemos reorganizar la fórmula (8) y obtendremos la velocidad media de pérdidas *P* según la siguiente expresión,

$$P = \frac{0{,}75}{W^2} \quad (9)$$

La ecuación (8) puede entenderse como si la red descartase un porcentaje de segmentos independientemente de las acciones que realice la fuente. Es decir, describe la forma en que va a reaccionar el emisor.

Para estudiar este comportamiento hemos simulado con NS un nuevo escenario en el que se han empleado enlaces de 2 Mbps de ancho de banda, con retardos de 10 ms. y cola *DropTail*. Además, se ha asociado a cada enlace una probabilidad de pérdida inicial de 0,001 que es incrementada el 5% cada 5 segundos en todos los enlaces. El objetivo de este escenario es estudiar el comportamiento del ancho de banda con respecto a la probabilidad *P* de errores. Como resultado hemos obtenido el gráfico de la *Fig. 3* en el que nuevamente puede comprobarse cómo se cumple el comportamiento macroscópico que indicaba la intuitiva *Fig. 1*.

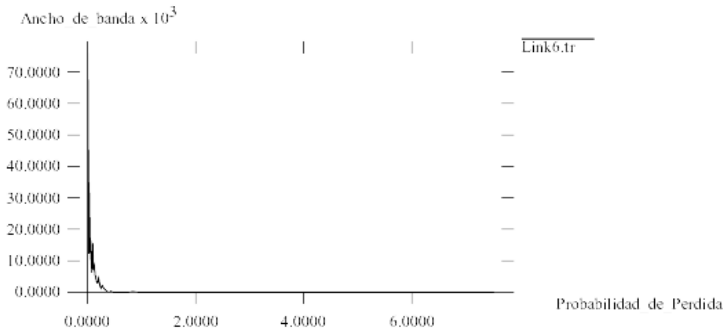


Figura 3. Simulación de la ecuación (4) con NS

Podemos ver cómo en la *Fig. 3* la probabilidad de pérdida *P* acaba determinando el ancho de banda de la fuente TCP, como también lo indica intuitivamente la ecuación (8) anterior. A medida que aumenta la probabilidad de pérdida, disminuye logarítmicamente el ancho de banda hasta acercarse a una evolución lineal. La pendiente logarítmica negativa refleja la caida de la efectividad de la red a medida que se pierden segmentos TCP. El problema lo encontramos en el hecho que TCP dobla los intervalos de los tiempos de retransmisión entre pérdidas sucesivas de paquetes. Es decir, si en la ecuación (6) no se considera constante el valor de RTT, el efecto de éste sobre la *Fig. 3* será aun más negativo para la eficiencia de la red.

3 TCP sobre ATM

En lo relativo a las investigaciones sobre la evaluación del rendimiento de TCP sobre ATM éstas se pueden dividir en tres grandes grupos [8]: 1) las que se fijan en el dinamismo de TCP; 2) las que atienden al comportamiento de ATM; y 3) las que prestan atención a la interacción entre las ventanas de TCP y los mecanismos de control de congestión de la capa ATM. Aunque la evaluación del rendimiento de TCP sobre ATM ha sido fuente de diversas investigaciones, las propuestas resuelven sólo problemas particulares como es la fragmentación de TCP, los requerimientos de *buffers*, la interacción entre los esquemas de congestión de TCP y ATM, y la degradación de TCP. En cierto modo se echan en falta propuestas generales que se enfrenten a todas o varias de estas problemáticas. En esta línea se dirigen nuestras investigaciones, aportando un SMA (Sistema MultiAgente) que busca la optimización del *goodput* con un adecuado tratamiento de las colas de entrada, y con una cuidada política de gestión de buffer mediante la delegación de actividades concretas en los agentes que forman parte del SMA.

La mayor parte de aplicaciones de datos no son capaces de predecir sus propias necesidades de ancho de banda, por lo que se necesita de algún servicio que permita a todos los usuarios activos de la red compartir dinámicamente el ancho de banda disponible. Sabemos que en el caso de ATM las Clases de Servicio ABR y UBR son la propuesta estándar para soportar el tráfico de datos. La referencia [9] presenta el estudio de congestiones de redes TCP sobre ATM comprobando cómo el *throughput* de TCP cae también cuando se comienzan a descartar células en los conmutadores ATM. El bajo *throughput* conseguido se debe al desaprovechamiento del ancho de banda en los enlaces congestionados que transmiten células de paquetes corrompidos; es decir, paquetes en los cuales se ha tirado alguna de sus células. Otras investigaciones [10,11] han demostrado que TCP sobre UBR con EPD experimenta una apreciable degradación en el funcionamiento en cuanto a la justicia, requiriendo además de un tamaño de buffer relativamente grande, incluso con pocas conexiones. Sin embargo, la CoS ABR con esquemas de realimentación de velocidad explícita ofrece a TCP mejor comportamiento en cuanto a justicia y aprovechamiento de los enlaces y, todo ello, con tamaños de buffer menores que con UBR.

Existen dos diferencias básicas entre los esquemas de control de congestión de TCP y la CoS ABR de ATM: 1) El mecanismo de realimentación de ABR con células RM controla la velocidad de transmisión de las células desde el emisor (control de velocidad), mientras el mecanismo de realimentación de TCP controla el tamaño de una ventana como hemos visto (control de créditos). 2) el mecanismo de realimentación de ABR puede ser realizado por conmutadores intermedios de la red, o por el extremo receptor del tráfico, mientras en TCP el mecanismo de realimentación es realizado sólo por el nodo destino mediante ACK e-e.

En las fuentes TCP el tráfico máximo es controlado por la ventana CWND tal como hemos visto. Sin embargo, en el caso de la CoS ABR de ATM el tráfico es controlado por parámetros como MCR (*Minimum Cell Rate*), PCR (*Peak Cell Rate*) y ACR (*Allowed Cell Rate*). Son también aspectos clave para TCP sobre ATM los mecanismos de gestión de tráfico usados en los nodos extremos de TCP, en los nodos extremos de ATM y en los conmutadores de la red para, entre todos ellos, aportar el adecuado *goodput* para reducir el retardo causado por las retransmisiones.

A la vista de todas estas características diferenciadoras, y dado que ATM es siempre un protocolo situado por debajo del protocolo de la capa de transporte TCP, se requieren soluciones para resolver los problemas de rendimiento provocados por la integración de ambas tecnologías. Estas soluciones parece lógico que estén en la línea de realizar cambios en los conmutadores ATM dentro de la red; o bien en la nueva implementación de extensiones para TCP; o también en la propuesta de protocolos especializados para los nodos que están en los límites de la red ATM con la red TCP. Destacamos que en nuestro caso TAP se enfrenta a estos problemas actuando dentro de la misma red con mecanismos hardware (conmutadores activos AcTMs) y también software (SMA con protocolo TAP), lo que configura toda la arquitectura TAP.

4 Beneficios aportados por TAP

En nuestro caso hemos propuesto EAAL-5 como extensión de AAL-5 que se diseñó específicamente para la comunicación de datos a través de ATM. En el caso de TCP sobre ATM, los datagramas IP son transmitidos en la zona del campo de datos (*payload*) de EAAL-5, tal como se intuye en la *Fig. 4* que presenta las pilas de protocolos de una fuente emisora y otra receptora TCP sobre ATM.

Antes de concluir estos breves comentarios sobre el control de congestión de ATM, se destaca que éste es un importante aspecto, por lo que se ha puesto especial interés en aspectos como: la escalabilidad, la justicia, la robustez y la facilidad de implementación del control de congestión. Aspectos que hemos procurado satisfacer en TAP.

Las redes activas, abiertas y programables son una nueva área técnica [12-14] para explorar vías por las que los elementos de la red puedan ser dinámicamente reprogramados por administradores, operadores o usuarios para obtener la QoS requerida. Esto ofrece atractivas ventajas y también cambios importantes en aspectos como el rendimiento, la seguridad y la fiabilidad. Por tanto, ésta es una línea abierta a la investigación y al desarrollo de la ingeniería de protocolos para lograr mover el código de servicio (colocado dentro de la capa de transporte de la red) a los nodos de conmutación.

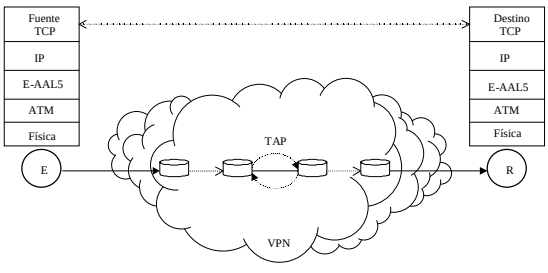


Figura 4. TCP sobre ATM con TAP

Conceptos como redes activas, protocolos *boosters* o agentes software fueron propuesto paras redes IP; sin embargo, las propuestas son insuficientes en el ámbito de las redes ATM donde situamos este trabajo. No hay consenso para decidir cuándo una red es activa pero existen dos grandes tendencias*: una red es activa si incorpora nodos activos con la capacidad de ejecutar programas de usuario*, o bien si ésta *implementa mecanismos de propagación de código entre los conmutadores ATM*. La arquitectura TAP es activa en ambos sentidos porque sitúa nodos activos en puntos estratégicos que implementan un protocolo que permite que el código de usuario sea cargado dinámicamente en los nodos de la red en tiempo de ejecución. También soporta la propagación de código en la red gracias a las células RM.

En trabajos previos [15,16] hemos presentado la arquitectura basada en el SMA-TAP, constituida por agentes software y dotada de una memoria dinámica DMTE que atiende las retransmisiones locales. La arquitectura de los conmutadores activos AcTMs puede observarse en la *Fig. 5*. Hemos implementado también el algoritmo QPWFQ (*Queues PDU Weighted Fair Queueing*) para aplicar justicia en las fuentes. Además, el algoritmo EPDR permite tratar las congestiones del buffer y evita la fragmentación de las PDU.

La motivación general de este trabajo se encuentra por tanto en encontrar soluciones para aliviar este negativo problema de retransmisiones e-e. Así, TAP se enfrenta a resolver las retransmisiones de forma local a donde se producen para evitar el coste del tiempo RTT total de la red y emplear sólo el *rtt* del enlace local congestionado. A la vez, se aprovechan los tiempos de inactividad en que se encuentran los enlaces. Es decir, empleamos los tiempos de inactividad de los enlaces para realizar las retransmisiones punto-a-punto en lugar de e-e.

De forma más intuitiva puede verse también la situación descrita en la *Fig. 6*, donde se tiene una red con 6 enlaces en la que cada uno de ellos experimenta igual retardo *d=10 ms*.

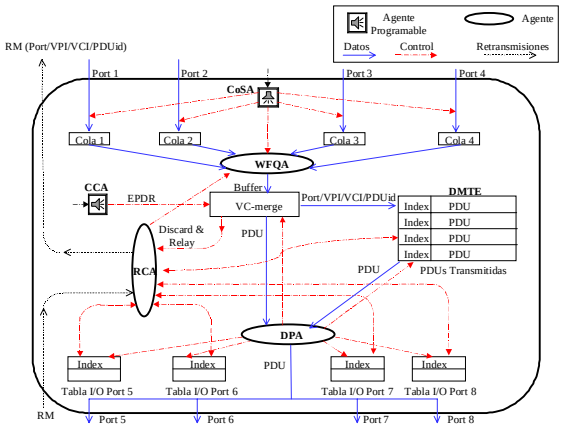


Figura 5. Arquitectura TAP con el subsistema SMA-TAP

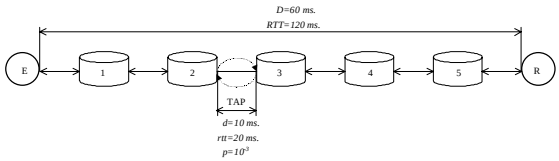


Figura 6. Retransmisión local al conmutador congestionado

En una situación de red ideal en el escenario de la *Fig. 6*, el retardo total de la red es de *D=60 ms*, por lo que tenemos un *RTT=120 ms* entre los dos extremos de la comunicación. Nuestro objetivo es situar el protocolo TAP entre dos enlaces de la red para conseguir amortiguar el efecto negativo de las congestiones producidas en el conmutador 3. Si suponemos que, tanto el conmutador 2 como el 3 soportan la arquitectura TAP, cuando se produzca congestión de un paquete en el conmutador 3, su retransmisión no tendrá un coste de 150 ms, sino que en realidad el coste será de 20 ms que corresponde el retardo debido al *rtt* local del enlace que ha experimentado congestión.

Ante la *Fig. 6* pueden estudiarse las relaciones entre RTT, *B* (tamaño del buffer en los routers), *P* (probabilidad de pérdida de la red) y *V* (velocidad de envío de las fuentes de tráfico). En el caso ideal en que no se produzcan pérdidas de segmentos TCP en la red, la memoria DMTE de TAP aporta mayor tamaño de buffer, lo que colabora a evitar las pérdidas. Por otro lado, si aparecen pérdidas, TAP las recupera en el punto donde se producen por lo que se reduce el RTT e-e al *rtt* p-p.

5 Evaluación del rendimiento

La *Fig. 7* muestra el efecto experimentado al variar el CAR (*Cell Arrival Rate*) entre 86 y 2.667 células por segundo (33.000 bps hasta 1 Mbps respectivamente). En esta simulación se ha fijado la probabilidad de congestión a 10^{-3}. Se usa un buffer de 3.000 octetos y la memoria DMTE almacena 2 PDU de 1.500 bytes para cada conexión.

Si el valor de CAR es de 64 Kbps (167 cells/s.); Ton=0,96 s.; y Toff=1,69 s, sobre el total de las 50 PDU descartadas por congestión, 50 PDU son recuperadas por TAP. También, cuando el CAR=56 Kbps y 33 Kbps, TAP recupera todas las PDU congestionadas. Así, el rendimiento es optimizado (50 PDU recuperadas de 50 PDU congestionadas) porque todas las PDU perdidas son recuperadas y no se producen fallos de DMTE, pues todas las PDU solicitadas se encuentran en la DMTE.

Como puede verse, cuando la velocidad de llegada de células es baja, el número de PDU recuperadas incrementa. Cuando el CAR incrementa a 256 Kbps, TAP recupera 48 de las 50 PDU, pero 2 de las PDU perdidas no son solicitadas porque el protocolo detecta insuficiente tiempo de inactividad (Toff) para poder realizar la retransmisión.

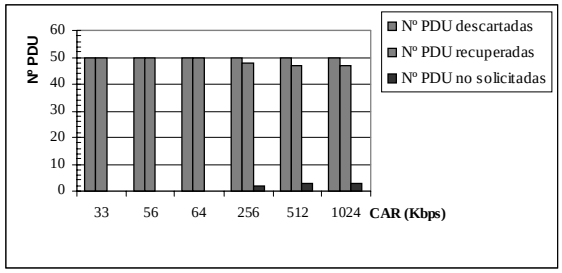


Figura 7. Número de PDU recuperadas en función del PCR

Podemos ver cómo el número de NACK no enviados (PDU no solicitadas) es mayor cuando el valor de CAR incrementa. De este modo, la red no será sobrecargada con retransmisiones inútiles cuando no existe garantía de éxito en la recuperación al no disponer de suficiente tiempo de inactividad agregado en los enlaces.

A mayores valores de CAR (1 Mbps), el número de PDU recuperadas es de 47 y, también en este caso, las 3 PDU no recuperadas no han sido solicitadas. De este modo, podemos ver que el *goodput* es también optimizado a elevada velocidad, siempre y cuando el número de fuentes seguras no exceda la capacidad de servicio del enlace.

La *Fig. 8* muestra los resultados obtenidos al variar el tiempo de inactividad (Toff) entre 0,1 y 2 segundos. En este caso usamos 10 fuentes ON/OFF sobre un enlace con ancho de banda de 25 Mbps. Cada fuente genera 500 PDU con un PCR=1 Mbps. Se ha fijado el *Cell Loss Rate* al 5% sobre el total de PDU emitidas, y hemos mantenido constante el valor de 25 PDU congestionadas. Como puede observarse, cuando el Toff agregado es suficiente (0,5 s.), todas las PDU congestionadas son recuperadas (25 recuperaciones de 25 congestiones). Cuando el tiempo Toff es menor de 0,5 s. las PDU recuperadas caen a 12 PDU a 0,3 s., y a 3 PDU recuperadas a 0,1 s. De este modo, cuando el tiempo Toff es insuficiente, el número de PDU irrecuperables crece, pero TAP garantiza el *goodput* ya que las PDU irrecuperables no son solicitadas para evitar sobrecargar la red.

Con estas y otras simulaciones hemos comprobado que la arquitectura TAP distribuida y activa aprovecha las ventajas de los conmutadores AcTMs. Hemos verificado que es posible recuperar un importante número de PDU con un aceptable tamaño de memoria DMTE y una razonable complejidad añadida en los conmutadores activos soportada por agentes software. Nuestras simulaciones demuestran también que la idea intuitiva de aprovechar los tiempos de silencio en las fuentes ON/OFF es cierta. Así conseguimos también mejor comportamiento y QoS en las redes ATM que soportan tráfico TCP. TAP evita además la implosión, los problemas de injusticia de las fuentes, la fragmentación de PDU y su *interleaving*.

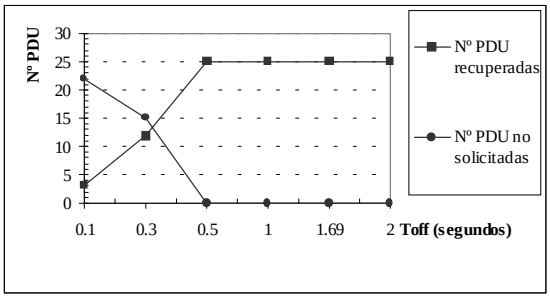


Figura 8. Efecto de la variación de Toff

6 Conclusiones

En los protocolos de la capa de transporte como TCP sobre ATM un paquete es descartado por la red cuando se pierde una o varias células del paquete, y el nodo destino solicita la retransmisión completa del paquete corrompido o perdido. Mediante simulaciones se ha demostrado la degradación que experimenta el *throughput* de TCP viendo cómo éste cae logarítmicamente a medida que aumenta la probabilidad de pérdida de células ATM. Con TAP se realizan las retransmisiones de forma local por lo que se consigue reducir la probabilidad de pérdida con el consiguiente efecto sobre el *throughput* de TCP que evita los retardos debidos al RTT e-e. La arquitectura distribuida TAP está constituida por un sistema multiagente con un protocolo que funciona sobre nodos activos AcTMs con agentes software.

Referencias

[1] R. Steinmetz, and L. C. Wolf, "Quality of Service: Where are We ?," *Fifth International Workshop on Quality of Service IWQOS'97*, pp. 211-221, (1997).

[2] Yoshihiro Ohba, "QLWFQ: A Queue Length Based Weighted Fair Queueing Algorithm in ATM Networks", *INFOCOM'97, Proceedings IEEE*, pp. 566-575 vol.2 (1997).

[3] W. Richard Stevens, "TCP/IP Illustrated, Volume 1," *Addison-Wesley Professional Computing Series*, (1994).

[4] UCB/LBL/VINT Network Simulator - ns, *http://www-mash.cs.berkeley.edu/ns/*

[5] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The Macroscopic behavior of the TCP Congestion Avoidance Algorithm," *Computer Communications Review of ACM SIGCOMM*, vol. 27, n. 3, (1997)

[6] Sally Floyd, "Connections with Multiple Congested Gateways in Packet-Switched Networks Part 1: One-way Traffic," *Computer Communications Review*, vol 21, n. 5 (1995).

[7] Robert Morris, "Scalable TCP Congestion Control," *Proceedings IEEE INFOCOM'2000*, pp. 1176-1183, (2000).

[8] K. Djemame, and M. Kara, "Proposals for a Coherent Approach to Cooperation between TCP and ATM Congestion Control Algorithms," *Procs. UKPEW'99*, pp. 273-284.

[9] Romanow, A. and Floyd, S., "Dynamics of TCP traffic over ATM networks," *IEEE Journal on Selected Areas in Communications*, pp. 633-641, (1995).

[10] Hongqing Li, Kai-Yeung Siu, Hong-Yi Tzeng, Ikeda, C., and Suzuki, H., "A simulation study of TCP performance in ATM networks with ABR and UBR services," *Proceedings IEEE INFOCOM'96*, pp. 1269-1276, (1996).

[11] Shunsaku Nagata, Naotaka Morita, Hiromi Noguchi, and Kou Miyake, "An analysis of the impact of suspending cell discarding in TCP-over-ATM," *Proccedings IEEE INFOCOM'2000*, pp. 1147-1156, (2000).

[12] D. L. Tennenhouse, J. M. Smith, W. D. Sincoskie, D. J. Wetherall, and G. J. Minden, "A Survey of Active Network Research," *IEEE Commun. Magazine*, pp. 80-86, (1997).

[13] D.A. Halls and S. G. Rooney, "Controlling the Tempest: Adaptive Management in Advanced ATM Control Architecture," *IEEE JSAC*, Vol. 16, Nº 3, pp. 414-423, (1998).

[14] German Goldszmidt, and Yechiam Yemini, "Delegated Agents for Network Management," *IEEE Communications Magazine*, Volume 36, 3, pp. 66-70, (1998).

[15] José Luis González-Sánchez and Jordi Domingo-Pascual, "TAP: Architecture for Trusted Transfers in ATM Networks with Active Switches," *ATM'2000 IEEE Conference on High Performance Switching and Routing Joint IEEE ATM Workshop 2000 and 3rd International Confer. on ATM (ICATM'2000 Heidelberg)*, pp. 105-112 (2000).

[16] José Luis González-Sánchez and Jordi Domingo-Pascual, "Trusted and Active Protocol over a Distributed Architecture in ATM Networks with agents," *IEEE International Confer. on Computer Communic. and Networks (IEEE IC3N'2000, Las Vegas)*, pp. 484-490 (2000).

Tecnología de Agentes en los Sistemas de Telefonía Móvil

Mª Celeste Campo Vázquez, Carlos García Rubio, Andrés Marín López, Carlos Delgado Kloos
Departamento de Ingeniería Telemática. Universidad Carlos III de Madrid.
Avda. de la Universidad 30, 28911 Leganés (Madrid)
E-mail: celeste@it.uc3m.es, cgr@it.uc3m.es, amarin@it.uc3m.es, cdk@it.uc3m.es

Abstract*. This paper studies the application of agents technology to mobile telephone networks, specially to future 3rd generation systems. We propose mobile agents to be used to compile information in the mobile terminal and build a profile associated to the user. This profile will be used both for more efficient network management (reducing signaling traffic due to mobility and detecting areas without coverage) and for providing services personalized to the characteristics of the terminal, the user and his exact localization. In order to overcome the limitations that terminal heterogeneity imposes, we propose the agent infrastructure to run in the SIM module of mobile terminals. ETSI has specified the use of Java Cards as SIM modules for UMTS systems, the so called USIM (UMTS-SIM). Java Cards execute Java code (a reduced version), so we can use them to build SIM-based terminal-independent applications.*

1 Introducción [1]

Debido al gran impacto que han tenido las redes de telefonía móvil de segunda generación, como GSM, varios organismos en los últimos años están estandarizando redes móviles de tercera generación. A nivel mundial el ITU está especificando las características comunes que tendrán estas redes para garantizar su interoperabilidad, es lo que se denomina sistema IMT-2000. El UMTS es uno de estos sistemas, y será el sucesor del sistema GSM.

Una de las principales ventajas que presentó GSM respecto a otros sistemas de telefonía móvil de segunda generación, fue la introducción del SIM (Subscriber Identification Module), que permite la independencia del terminal que emplea el usuario para acceder a los servicios que tiene contratados. Los SIM son tarjetas inteligentes que almacenan información de suscripción de un usuario y son las que permiten implantar mecanismos de seguridad en la parte radio de la red, almacenando el PIN, las claves de autenticación del usuario y realizando cálculos criptográficos.

Las tarjetas inteligentes se denominan así porque incorporan un circuito integrado con elementos usados para la transmisión, almacenamiento y manipulación de datos. En principio, las tarjetas inteligentes tenían sistemas operativos propietarios y su utilización se basaba fundamentalmente en el almacenamiento seguro de información y cálculo de algoritmos criptográficos. La evolución y aumento de capacidad de almacenamiento y proceso en los chips insertados en estas tarjetas, propició el desarrollo de tarjetas Java Card [1] que permiten la ejecución de código Java (en una versión reducida) en la propia tarjeta, pudiendo así construir aplicaciones que se ejecutan en cualquier tarjeta Java Card independientemente del tipo de sistema operativo. El ETSI ha especificado un API para la utilización de tarjetas de este tipo como módulo SIM para los sistemas UMTS, lo que ya se denomina USIM (UMTS-SIM).

Por otra parte, durante los últimos años la tecnología de agentes ha tenido un gran desarrollo. El concepto de agente ha sido muy discutido y aunque existen varias definiciones, la más aceptada es la que define a los agentes por sus características: movilidad, autonomía, inteligencia, comunicación, cooperación y coordinación. Un agente es aquel que posee una o varias de estas características. En general, ha habido dos tendencias en el desarrollo de tecnología de agentes: los agentes móviles y los agentes inteligentes. Los agentes móviles son aquellos que pueden moverse de un equipo a otro, dentro de entornos heterogéneos, para realizar las tareas que tienen asignadas. Los agentes inteligentes son entidades que son capaces de realizar tareas basándose en su conocimiento adquirido, y en su capacidad de comunicación y negociación con otros agentes. El lenguaje Java ha sido el que más éxito ha tenido para la implementación de infraestructuras de agentes.

Para entender los beneficios de usar agentes debemos considerar el especial impacto que ha tenido en los últimos años su aplicación en la computación distribuida. El modelo más ampliamente extendido es el modelo cliente / servidor, en el que un cliente que se ejecuta en un entorno envía un conjunto de datos a un servidor, y espera que éste le envíe los datos de respuesta de la operación realizada, antes de enviar nuevos datos.

[1] Este trabajo ha sido desarrollado dentro del proyecto E-TICKET CYCYT Nº2FD1997-1269-C02-01(TEL)

Cada mensaje intercambiado en la red implica una petición de un servicio y una respuesta a esa petición. La comunicación establecida precisa de un conexión permanente. Esto provoca el consumo de un gran número de recursos.

La introducción de tecnología de agentes permite realizar las mismas operaciones pero con la ventaja de que sean asíncronas y que además no precisemos conexiones permanentes para la ejecución de tareas, puesto que el agente que migra hacia otro sistema además de llevar los datos necesarios para realizar la operación, conserva la información de estado del proceso.

2 Aplicación de los agentes en telefonía móvil

Los sistemas de telefonía móvil se caracterizan por una serie de restricciones: ancho de banda limitado, alta probabilidad de error en el interfaz radio, cobertura discontinua y limitada, baja capacidad de procesamiento en los sistemas finales, interfaz de usuario limitada, etc.

La utilización de la tecnología de agentes en estos sistemas permite adaptarse a estas limitaciones para proporcionar mejores servicios a los usuarios finales y mejorar las prestaciones de la red, porque:

- Los agentes que proporcionan un servicio pueden enviarse dinámicamente y bajo demanda a los propios usuarios.
- Los agentes permiten realizar distribución de tareas para realizar actividades de gestión, siendo los propios agentes quienes recopilen datos y los procesen localmente en la parte del terminal móvil.
- La autonomía de los agentes permite que se realicen tareas de forma asíncrona.
- Los agentes pueden realizar gran parte del procesamiento de forma local, por lo que se conseguirá una reducción importante del tráfico en la red.
- Los agentes permiten una mayor independencia de la disponibilidad de la red, ya que su capacidad de movilidad les permite migrar a otros nodos de la red.

La implementación de una infraestructura de agentes en una red de telefonía móvil presenta una complejidad importante debido a la limitaciones que imponen los terminales, que poseen una capacidad de procesamiento y almacenamiento reducida. Las aplicaciones de agentes que vamos a proponer implican que en nuestro terminal móvil tengamos una plataforma capaz de ejecutar y lanzar agentes hacia otros elementos de red, que pueden ser otros terminales móviles o elementos de la red fija. Además debido a las limitaciones de almacenamiento será necesario que algunos agentes residan un tiempo limitado en el terminal, por lo tanto será clave el control y gestión del número y tipo de agentes que residen en los móviles.

Existen estudios realizados sobre la posibilidad de utilizar los estándares existentes de plataformas de agentes móviles en el contexto de sistemas de comunicaciones móviles de tercera generación. En concreto, en [2] se analiza la utilización del estándar MASIF (Mobile Agent System Interoperability Facility) del OMG (Object Management Group) para construir una plataforma de agentes orientados a la provisión de servicios personalizados. Ver también [3] y [4].

En este artículo proponemos realizar una plataforma básica de agentes en el lado del terminal móvil que va a residir en el terminal (J2ME, Java 2 Micro Edition) y la tarjeta inteligente USIM (Java Card). Si consideramos que el lenguaje Java ha sido el que más éxito ha tenido para la implementación de infraestructuras de agentes, vemos que la posibilidad de implantar una infraestructura de agentes incluyendo los terminales móviles es cada vez más cercana y abordable. El paso a dar es adaptar a las limitaciones de procesado de las tarjetas y a las del lenguaje Java Card, las plataformas de agentes ya existentes.

Proponemos dos líneas de aplicación de la tecnología de agentes en sistemas de telefonía móvil de tercera generación.

- Por una parte, su aplicación en tareas de gestión de red, siguiendo una tendencia ya explorada en redes fijas [5], pero que tiene mayor interés en redes móviles debido a que las propias características de los agentes móviles se adaptan a las limitaciones de los sistemas inalámbricos.
- Por otra, su aplicación en la realización del VHE (Virtual Home Environment), que permitirá la personalización y portabilidad de los servicios de los usuarios independientemente de la red que le da servicio y del terminal que empleen en el acceso. Asociamos la implementación del VHE con un agente móvil, que permitirá configurar el servicio para adaptarse a las preferencias del usuario y a las características del terminal, y además será el encargado de crear el propio perfil de usuario analizando su comportamiento y su posición.

3 Agentes para la gestión de la red

Analizaremos en esta sección dos de los problemas más importantes en la gestión de una red inalámbrica, y cómo la utilización de tecnología de agentes permite implantar de manera eficiente las soluciones propuestas.

3.1 Gestión de la movilidad

La aplicación de los agentes móviles en la gestión de la movilidad de los terminales nos permitirá reducir el tráfico de señalización que se genera con los esquemas actuales. En la primera parte de este apartado se explicarán cuales son los mecanismos que están siendo empleados y que, en principio, se mantendrán en las redes de tercera generación y en la segunda se explicarán algunos de los nuevos esquemas propuestos.

3.1.1 Esquemas actuales de gestión de la movilidad

Una de las principales dificultades introducidas por las redes móviles, comparado con las redes de telefonía tradicionales, es el hecho de que las estaciones móviles no tienen una conexión permanente con la red. Por esta razón la red en todo momento debe saber la posición del usuario móvil. Para ello, el esquema que se ha seguido en los sistemas de telefonía móvil celular es definir áreas de localización LA (Location Area).

Un área de localización es una zona geográfica cubierta por un conjunto de estaciones base pertenecientes a un mismo grupo, típicamente dependientes del mismo MSC (Mobile Switching Center), conforme se muestra en la figura 1.

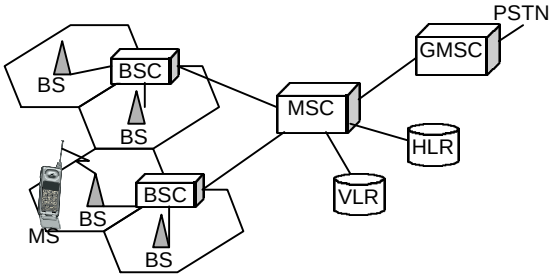


Figura 1: Arquitectura de redes celulares

Existen en la actualidad dos normas para la gestión de la movilidad, la GSM-MAP (Global System for Mobile Communications-Mobile Application Part), para redes GSM y la IS-41 en redes AMPS (Advanced Mobile Phone System). Ambas normas son muy similares, aunque en este documento consideraremos el esquema seguido en GSM. Ver para una descripción mas detallada [6]. La gestión de la movilidad se divide en dos procesos, por una parte gestionar la información de localización y por otra, localizar al móvil en el momento en que se produce una llamada entrante.

Gestión de la información de localización

En todo momento la red tiene que saber en que LA se encuentra un terminal móvil. Esta información de localización se almacena en dos tipos de bases de datos, VLR (Visitor Location Register) y HLR (Home Location Register).

- VLR es donde se almacenan todos los datos significativos del usuario móvil para proporcionar el servicio mientras se encuentre en su área de control. Entre la información almacenada se mantiene una entrada para cada terminal que gestiona, con el identificador de LA en el que se encuentra. Típicamente cada MSC tiene asociado un VLR.
- HLR es donde se almacenan permanentemente todos los parámetros de suscripción del usuario móvil. Como parámetro relativo a la localización, se almacena el VLR del que depende el MS (Mobile Subscriber) en un determinado momento.

Como el usuario se está moviendo en la zona de cobertura de la red, es necesario que los registros almacenados en estas bases de datos se actualicen periódicamente, esto se realiza a través de un proceso denominado registro de localización. Como el identificador de un LA es enviado periódicamente por un canal común de *broadcast*, a todos los MS que se encuentran en cada una de las estaciones base pertenecientes al LA, el móvil conoce en todo momento el LA en el que se encuentra, realizando el registro de localización:

- Cada vez que se conecta a la red, es decir, en el momento en el que el terminal móvil se enciende.
- Cada vez que finaliza un temporizador asociado al proceso de localización, que obliga al móvil a realizar un nuevo registro aunque se encuentre en el mismo LA.
- Cada vez que cambia de LA, que lo detecta comparando el identificador de LA que tiene almacenado en el SIM con el que está recibiendo. Se distinguen tres tipos diferentes de registro de localización dependiendo de la relación existente entre el anterior y el nuevo LA: si el nuevo LA está asociada al mismo VLR, si el nuevo LA está asociada a un nuevo VLR en el mismo MSC, o si el nuevo LA está asociada a un nuevo VLR en distinto MSC.

El proceso que es necesario llevar a cabo para el registro se detalla en los pasos siguientes:

- El MS inicializa el proceso de registro transmitiendo a su estación base el identificador del área de localización que está recibiendo.
- La BS envía este mensaje al MSC del que depende, que realiza la consulta necesaria en el VLR asociada.
- El VLR actualiza el registro de localización de MS. Si el nuevo LA pertenece a un VLR diferente, el nuevo VLR determina la dirección del HLR del terminal móvil, a través del número de identificación del MS, y envía un mensaje de actualización de la localización al

HLR. En otro caso, el registro de localización finaliza.

- El HLR realiza un proceso de autenticación con el terminal móvil, si es correcta, almacena el identificador del nuevo VLR en el registro correspondiente y envía un mensaje de confirmación al nuevo VLR.
- EL HLR envía un mensaje al VLR anterior del que dependía el MS, para que anule el registro asociado al MS.
- El anterior VLR elimina este registro de su base de datos y envía la confirmación correspondiente al HLR.

Todos estos mensajes son de señalización, con el correspondiente aumento del tráfico en la red. En la tabla siguiente se indica el número de mensajes transmitidos en los diferentes casos:

Tipo de registro	Nº mensajes	Bytes
En el mismo VLR	5	122
En el mismo MSC	21	1131
Entre diferentes MSC	36	1309

Si las LA son pequeñas, como ocurre en zonas urbanas, y como ocurrirá en los sistemas de telefonía de tercera generación de forma más generalizada, el promedio de registros de localización aumenta considerablemente, provocando que la cantidad de tráfico de señalización generado aumente tanto que supere al propio tráfico de llamadas, por lo tanto se precisan nuevos esquemas para mejorar este modelo.

Localización del terminal móvil

Cuando se produce una llamada entrante para un determinado MS, la red tiene que localizar ese terminal dentro de su zona de cobertura para poder establecer la llamada. El proceso puede dividirse en dos partes:

- Determinar el VLR al que está asociado el MS en ese momento, para ello se hace una consulta en el registro correspondiente en el HLR del que depende el subscriptor del MS.
- Localizar la celda en la que se encuentra el MS. El VLR envía un mensaje al MSC que gestiona el LA en el que está registrado el MS. A continuación, el MSC realiza una búsqueda (*paging*) en todas las celdas que pertenecen al LA, enviando un mensaje de *broadcast* con un identificador del móvil al que se debe entregar la llamada. Finalmente los MS analizan este mensaje y responde el MS buscado, en ese momento ya se puede establecer el canal necesario para cursar la llamada.

Si las LA son grandes, se produce un aumento considerable del ancho de banda radio consumido en los procesos de búsqueda para cada llamada entrante de un MS. En las nuevas redes de telefonía móvil, con el aumento del número de usuarios y sus requisitos de ancho de banda, será más complejo soportar estos esquemas de búsqueda de terminales móviles. Además, en este caso, el tiempo de búsqueda podría superar el retardo máximo de localización de un terminal, para cursar la llamada.

3.1.2 Nuevos esquemas con tecnología de agentes

Como acabamos de ver, por una parte tener LA pequeñas aumenta el tráfico de señalización, debido a que los registros de localización de los móviles son mucho más frecuentes, pero aumentar el tamaño de las LA provoca un coste en el ancho de banda radio que se consume cuando es necesario realizar proceso de búsqueda de móviles y un aumento en el retardo de entrega de llamada. Entonces, el principal problema de la gestión de la movilidad es minimizar los costes de registro de localización y de *paging* simultáneamente.

En los sistemas móviles de tercera generación deberían aplicarse nuevos esquemas para gestionar la movilidad siguiendo tres objetivos:

- Reducir el número de actualizaciones de localización realizados por los MS.
- Reducir la zona de búsqueda de un terminal cuando se produce una llamada entrante de forma que se cumplan las restricciones temporales impuestas.
- Distribuir de manera más eficiente el almacenamiento de la información de la localización en la red.

En este artículo nos centraremos en describir algunos de los esquemas que se pueden aplicar para conseguir los dos primeros objetivos y comentaremos cómo en algunos casos se pueden implementar de manera eficiente empleando la tecnología de agentes. El tercero de ellos también ha sido analizado ampliamente [6] y las soluciones aportadas se basan en tener jerarquías de bases de datos de mayor nivel y emplear técnicas de replicación de las bases de datos del sistema en diversos puntos de la red.

Gestión de la información de localización

Con el objetivo de reducir el número de actualizaciones de localización que debe realizar un MS en una red móvil, se han propuesto diversos algoritmos, que pueden agruparse en dos tipos: estáticos y dinámicos. Los algoritmos estáticos se basan en la topología de la red y los algoritmos dinámicos se basan en la construcción de perfiles de movilidad y patrones de llamadas de los usuarios móviles.

En los algoritmos dinámicos es importante la captura de información en el lado del terminal móvil para poder construir el perfil del usuario al que se adaptarán los diferentes esquemas. Es en este tipo de algoritmos, en los que la introducción de agentes móviles cobra especial importancia, siendo los encargados de capturar la información del lado del usuario y gestionar su perfil, y de poder dotar de cierta inteligencia al terminal, para que el mismo decida el algoritmo de gestión de movilidad que mejor se adapte a su situación actual.

A continuación se describen brevemente algunos de los algoritmos propuestos, para una explicación más detallada se referencia a [6] y [7].

- Registro de localización selectivo. Se basan en el hecho de que un usuario móvil aunque puede atravesar un elevado número de LA, un ejemplo típico es cuando los usuarios van de su casa al trabajo y viceversa, en la mayoría de LA permanece un breve periodo de tiempo. Por lo tanto, no será necesario que actualice su localización cada vez que atraviesa un nuevo LA, sino sólo cuando existe una alta probabilidad de encontrarse en un LA.
- Basados en perfiles de usuarios. Se basan en la posibilidad de predecir el comportamiento de los usuarios, obteniendo patrones de movilidad. La red mantiene un perfil de movilidad para cada usuario, que incluye una lista secuencial de LA en las que con mayor probabilidad se encuentra el usuario en diferentes periodos de tiempo. Cuando es necesario localizar a un usuario, primero se busca en las LA que están almacenadas en el perfil, realizando *paging* sucesivamente en cada una de ellas. Además cuando el MS se mueve entre las LA almacenadas en su perfil, no realiza actualizaciones de su localización, sólo se realizarán cuando el móvil pase a un nuevo LA no contenido en su perfil.
- Basados en movimiento. Cada MS cuenta el número de saltos realizados entre celdas en su movimiento. El registro de localización se realiza cuando el número de saltos pasa un cierto umbral. Este umbral puede ser adaptado a las características del usuario. Para su implementación, el MS sólo necesita un contador para almacenar el número de saltos entre celdas y el valor del umbral.
- Basados en tiempo. El terminal móvil actualiza su localización cada T unidades de tiempo. En este algoritmo no es necesario que el MS almacene o procese información durante el tiempo entre actualizaciones, porque simplemente para su implementación sólo es necesario tener en el terminal móvil un temporizador. Una variación de este esquema consiste en tener un umbral de tiempo no constante, que se modifica según la carga de tráfico de señalización en la red.
- Basados en distancia. Cada MS calcula la distancia recorrida en su movimiento, en número de celdas, desde la última vez que actualizó su localización y realiza una nueva actualización cuando la distancia excede un cierto umbral. Para su implementación es necesario que el MS tenga conocimiento de la topología de la red en la que se encuentra, para poder identificar si ha superado o no el umbral de distancia impuesto en el algoritmo, así cada vez que realiza una actualización de la localización, se descarga un conjunto de identificadores de celdas, próximas a su ubicación actual.
- Basados en predicciones de distancia. El MS durante el proceso de registro de localización envía a la red tanto su ubicación como su velocidad. Basándose en esta información, la red determina la función de densidad de probabilidad de localización del móvil, que empleará para determinar la localización del móvil en el futuro. Esta información estará disponible tanto en el terminal móvil como en la red. El MS chequea periódicamente su posición y realiza una actualización de su localización si su distancia excede la distancia predicha por el modelo. Si una llamada entrante llega, se realiza *paging* en la distancia predicha por el algoritmo, y si no se localiza se sigue un criterio de mínima distancia, hasta que se encuentre al MS.
- Basados en estados. El MS decide si realiza un registro de localización dependiendo del estado en el que se encuentra. La información de estado puede incluir el tiempo transcurrido o el número de celdas atravesadas desde la última actualización, la distancia entre la celda actual y la celda en la que se encontraba en la última actualización, o algún otro criterio. Entonces, mantiene diferente información de estado de distintos algoritmos de registro de localización.
- Actualización de LeZi. Este esquema está basado en un algoritmo de compresión propuesto por Lempel y Ziv. Este algoritmo puede ser considerado como un algoritmo basado en caminos en los que un histórico del movimiento realizado se envía a la red cada vez que se realiza un registro de localización. El histórico del movimiento consiste en un lista de identificadores de zonas (LA o celdas) que el MS atraviesa desde la última actualización de su localización. La red mantiene este histórico de forma comprimida como un árbol de búsqueda. Este histórico de movimiento se puede ver como una parte del perfil del usuario. Cuando existe una llamada entrante la red realiza *paging* del móvil basándose en las localizaciones proporcionadas por el árbol.

Localización del terminal móvil

El proceso de *paging* consiste en que la red debe determinar la localización exacta, celda en la que se encuentra, de un MS concreto. La búsqueda se puede dividir en varias iteraciones, en cada una de ellas se envían señales de búsqueda sobre un canal de control en el enlace descendente, a un conjunto de celdas donde se cree que se encuentra el terminal móvil. Todos los terminales móviles escuchan este mensaje, y sólo el que es buscado responde enviando un mensaje por el canal de control ascendente. En cada ciclo existe un periodo de *timeout*, si pasado este tiempo no existe una respuesta de ningún terminal móvil, entonces se realiza su búsqueda en otro conjunto de celdas.

El terminal móvil debe ser encontrado en un periodo de tiempo razonable, porque si no el que realiza la llamada puede desistir del intento. El máximo retardo de *paging* corresponde al máximo número de iteraciones de búsqueda que se realizarán para localizar a un MS.

El envío de estos mensajes consume ancho de banda en el canal radio, este coste es proporcional al número de iteraciones que se realizan, así como al número de celdas implicadas en cada iteración. El área en la que se realiza la búsqueda depende de la información proporcionada por el registro de localización. El coste de *paging* puede ser reducido prediciendo la localización actual del MS. En esta sección indicamos las diferentes estrategias propuestas.

- Criterio de la mínima distancia. La red comienza la búsqueda del terminal móvil empezando por la última celda en la que realizó la actualización de su localización, y sigue buscando en las celdas contiguas siguiendo un orden de mínima distancia. La distancia es medida en términos del número de celdas atravesadas. Si el esquema de actualización de registro utilizado, se basa en un umbral (de movimiento o de distancia), el área de búsqueda del móvil está limitada y podrá ser localizado en un número fijo de iteraciones. En cada iteración se realiza una búsqueda en las celdas que se encuentran a la misma distancia del punto de partida.
- Búsqueda secuencial basada en la probabilidad de localización de los usuarios. Este esquema se basa en obtener la localización actual de un MS basándose en la distribución de probabilidad de localización. Las señales de búsqueda sólo se envían en las celdas en la que se cree que el usuario puede estar presente, entonces la búsqueda se realiza en orden inverso a la probabilidad de localización del móvil. Esta estrategia presenta problemas sobre todo cuando existe una limitación en el retardo máximo de localización del móvil.
- Aumento de la velocidad de búsqueda. Se basa en reducir el coste de *paging* decrementando el tamaño de la zona en la que se realiza la búsqueda. El objetivo se consigue agrupando a los usuarios en diferentes clases de velocidades, basándose en la frecuencia de actualización de su localización. Cuando se produce una llamada entrante para un móvil el área de búsqueda es generada dinámicamente basándose en el tiempo en el que el móvil ha realizado su último registro en la red y la clase de velocidad a la que pertenece. Este esquema se puede implementar conjuntamente con otros esquemas de registro de localización.
- Búsquedas conjuntas. El modelo se basa en realizar búsquedas conjuntas de varios terminales móviles, basándose en el hecho que la búsqueda simultanea e independiente de varios MS provocaría un sobrecarga en el canal común de señalización dedicado a *paging*. Esta técnica se puede emplear conjuntamente con algunos de los algoritmos considerados anteriormente.

3.2 Identificación de zonas sin cobertura

Uno de los principales problemas que plantean las redes de telefonía móvil es realizar una planificación eficiente de la parte radio, para ofrecer a los usuarios una amplia cobertura y que la itinerancia entre celdas no suponga una pérdida del servicio en curso, pero sin emplear un gran número de recursos, es decir, no sobredimensionar excesivamente la red. Este problema se verá incrementado en los nuevos sistemas de telefonía móvil, en los que el radio de cobertura que proporcionará una estación base, no dependerá solamente de la potencia de señal radiada, sino que también dependerá del número de usuarios que se encuentran en ella.

En la actualidad, la planificación de la red se realiza empleando complejas simulaciones y medidas de cobertura en diferentes puntos a través de equipos móviles (típicamente en coches con unos equipos de medidas especializados), para ir descubriendo deficiencias en la red implantada y así mejorar la cobertura proporcionada a los usuarios. El problema que presenta este método es que con estos equipos no se llega a todos los puntos desde los que potencialmente un usuario móvil puede acceder a la red, lo cual provoca que el operador no tiene un conocimiento directo de posibles zonas sin cobertura.

Una de las soluciones propuestas, es implicar a los propios terminales móviles en la realización de estas medidas, aprovechando la capacidad de procesamiento que poseen. Así pueden utilizarse agentes móviles en la parte del terminal que realicen recolección y procesado de medidas y después envíen a la parte fija de la red el informe

correspondiente para que sea analizado en conjunto con el proporcionado por otros móviles. Analizamos esta propuesta a continuación.

Como parte del funcionamiento actual de una red móvil celular, cada cierto tiempo la red realiza una petición a los terminales móviles para saber el nivel de señal que reciben de las diferentes estaciones base que lo rodean y que, potencialmente, pueden darle servicio. Las medidas realizadas son enviadas a la red y en base a ellas se selecciona la estación base que dará un canal de comunicación al terminal móvil cuando se curse una llamada. Si el MS se encuentra en una zona sin cobertura no existirá una respuesta por parte del MS hacia la red. Si la red lleva un histórico de las peticiones de medidas realizadas y el MS junto con el nivel de señal, proporciona el identificador de la celda a la que se refiere la medida. Un análisis en la red, contrastando las medidas proporcionadas por varios MS, nos permitiría detectar zonas en las que no existe cobertura y poder mejorar la planificación de la red.

En nuestra propuesta, los agentes móviles serían los encargados de recopilar la información en la parte del MS, lo que permitiría que almacenasen esta información y la analizasen antes de mandarla a la red y de esa manera aprovechar de manera más eficiente el ancho de banda en la parte radio. También se podría utilizar agentes móviles que residieran en la parte fija, en concreto en el controlador de la estación base y que migraran de controlador cuando el móvil pasase a estar controlado por otro nuevo.

Si en las nuevas redes se soporta el nuevo protocolo de acceso radio incluido en el estándar, denominado ODMA (Oportunity Driven Multiple Access) que permite utilizar a los terminales móviles como repetidores de señal, para otros MS a los que no les da cobertura directa ninguna estación base, los agentes, residentes en los MS, podrán emplear su característica de movilidad para realizar saltos a otros MS, de manera que puedan determinar su localización actual y así precisar de manera más adecuada las zonas sin cobertura.

4 Agentes para personalización de servicios

Otro de los campos de aplicación de la tecnología de agentes en redes de telefonía móvil, es facilitar la provisión de servicios a los usuarios de manera que puedan subscribirse de forma sencilla a ellos, adaptarlos a sus propias características y acceder a ellos desde cualquier localización, red que le da acceso al servicio y características del terminal que utilizan. En redes UMTS esta característica de portabilidad e itinerancia de servicios es lo que se denomina VHE, idea que se ha integrado de forma global en todos los sistemas IMT-2000.

Asociar la implementación de VHE como agentes móviles es una idea aceptada en la literatura [2], ya que así por una parte, se dota de cierta inteligencia al terminal móvil para construir el perfil de usuario adaptándose no sólo a las preferencias establecidas directamente por el propio usuario, sino también a las definidas en su interacción con la red y a su localización. La capacidad de movilidad permite que los proveedores de servicio configuren agentes especializados en un servicio determinado y adaptados al perfil del usuario, que migran al MS adaptándose al tipo de terminal y a las características de la red a través de las que se accede al servicio de manera transparente al propio usuario. Como se deduce, la capacidad de comunicación entre estos agentes será fundamental para transmitir la información del perfil de los usuarios y las características de la red y el terminal.

4.1 Servicios basados en posicionamiento

Uno de los servicios que se espera que tengan mayor impacto en los sistemas de tercera generación son los que estén basados en el posicionamiento de los usuarios. En la actualidad, el grupo 3GPP del ETSI, que está llevando a cabo la estandarización de UMTS, está estandarizando un conjunto de técnicas que permitan obtener el posicionamiento de los usuarios móviles, basadas fundamentalmente en las características de las redes celulares, aunque ya existen soluciones propietarias, además de las soluciones basadas en el sistema GPS (Global Positioning System). Algunos de estos servicios serán:

- Información de lugares próximos a la localización actual: farmacias, cines, tiendas,...
- Actualización de servicios dependiendo de la localización de manera transparente al usuario, importante cuando se produce itinerancia entre redes de diferentes características.
- Localización de personas en situaciones críticas de emergencia.
- Oferta de servicios automática dependiendo del lugar donde se encuentra el usuario.
- Facturación adaptada a la localización del usuario fuente y destino de la comunicación.

En la especificación ETSI 3G TS 25.305 V3.2.0 se proponen siete familias de técnicas de localización del móvil. Estas técnicas necesitan en muchos casos realizar ciertas medidas en el propio terminal móvil, incluso en algunas de ellas la medición se hace completamente en el terminal. Se supone que dependiendo de la operadora, entorno geográfico, precisión deseada y tipo de servicio, se usará un sistema de localización u otro.

Por lo tanto es lógico pensar que la utilización de la tecnología de agentes que se ejecutan en el terminal y que se descargan dinámicamente según las

necesidades permitirá, por las características del entorno de ejecución (parte radio de la red), implementar de manera eficiente estas técnicas de localización.

4.2 Servicios personalizados

En general el uso de agentes puede permitir personalizar servicios al perfil y comportamiento del usuario, así como a las características gráficas y de velocidad del terminal. Son los servicios que en el informe técnico número 11 del UMTS Forum se denominan *"Infotainment and Edutainment"* [8].

5 Conclusiones

En este artículo se ha visto cómo la tecnología de agentes móviles se adapta a la características de los sistemas inalámbricos en los que el ancho de banda en la parte radio es limitado y por lo tanto, se precisan comunicaciones asíncronas y cierta autonomía en las aplicaciones para reducir el número de conexiones con la red, disminuyendo de esta forma el tráfico generado.

Se ha propuesto la tecnología de agentes para su utilización en tareas de gestión de red y se han identificado dos aplicaciones que se consideran que también será interesante introducir en las redes de telefonía de tercera generación, que son la gestión de la movilidad y la detección de zonas sin cobertura. La tarea de los agentes en estos casos es recolectar y procesar parcialmente datos en la parte del terminal móvil para que después se reporten a la red, que los analizará de forma global para adaptar los diferentes algoritmos de gestión de movilidad y mejorar la planificación de la red, respectivamente.

También se ha propuesto la aplicación de la tecnología de agentes para facilitar la provisión de servicios a los usuarios de manera que puedan subscribirse de forma sencilla a ellos, adaptarlos a sus propias características y acceder a ellos independientemente de la posición, red de acceso o características del terminal. Este concepto se ha denominado VHE, y su implementación la asociamos a un agente que tendrá que construir el perfil de usuario adaptándose no sólo a las preferencias establecidas directamente por el propio usuario, sino también a las definidas en su interacción con la red, comportamiento y posición.

Con el desarrollo de plataformas Java en tarjetas inteligentes, las denominadas Java Card, se abre la posibilidad de desarrollar plataformas de agentes en la parte de los terminales móviles. Vemos posible incluso hacerlo a partir de plataformas Java ya existentes, adaptándolas a los requisitos que imponen las limitaciones de procesado y almacenamiento de las tarjetas, que por otra parte, cada vez son menos restrictivas, gracias a los avances en la microelectrónica.

Uno de los problemas abiertos en las tarjetas inteligentes multiaplicación son los mecanismos de descarga de aplicaciones, en la actualidad el estándar proporcionado por VISA es el que más apoyo está teniendo, y el ETSI hace referencia a él en sus especificaciones. Los agentes son aplicaciones que se descargarán en la tarjeta y por lo tanto, la movilidad de los agentes hacia los terminales móviles estará condicionada a las normas que se estandaricen en este campo.

Existen además otro conjunto de problemas abiertos para la implantación de esta tecnología. Por una parte el tema de la seguridad, que es un tema crítico en todas las implementaciones de sistemas de agentes móviles y más aún en el caso de sistemas de telefonía móvil, en los que se debe garantizar itinerancias de servicios y por lo tanto de agentes, a través de diferentes redes de diferentes operadores. El tema es crucial y para su implementación se deberían establecer políticas de seguridad entre organismos de diferentes países.

Referencias

[1] Java Card Technology, http://java.sun.com/products/javacard/

[2] L.Hagen, M.Breugs, T.Magedanz. "Impacts of Mobile Agent Technology on Mobile Communication System Evolution". IEEE Personal Communications, Aug 1998.

[3] I.Brusic, V.Hassler, W.Lugmayr "Deployment of Mobile Agents in the Mobile Telephone Network Management". Institute of Communication Networks of Technical University of Vienna.

[4] J.Hartmann, W.Song. "Agent Technology for future mobile networks". ACTS CAMALEON project (ACTS 341).

[5] Y.I. Wijata, D.Niehaus, V.S.Frost, "A Scalable Agent-Based Network Measurement Infrastructure". IEEE Communications Magazine, Sep 2000.

[6] I.F.Akyldiz, J.Mcnair, J.Ho, H.Uzunalioglu, W.Wang. "Mobility Management in Next-Generation Wireless System". Proceedings of the IEEE, Aug 1999.

[7] V.Wong, V.C.Leung. "Location Management for Next-Generation Personal Communication Networks". IEEE Network, Sep/Oct 2000.

[8] Enabling UMTS Third Generation Services and Applications, UMTS Forum, 2000 Report 11, October 2000.

[9] J.F.Huber, D.Weiler, H.Brand. "UMTS, the Mobile Multimedia Vision for IMT-2000: A Focus on Standardization". IEEE Communications Magazine, Sep 2000.

NUEVOS SERVICIOS DE INTERMEDIACIÓN PARA ISPs: CERTIFICACIÓN Y NOTARÍA DIGITAL

Alejandro Muñoz, Mariví Higuero, Alfonso García, Igor Pérez
Departamento de Electrónica y Telecomunicaciones
Universidad del País Vasco / Euskal Herriko Unibertsitatea
Alda. Urquijo s/n. 48013 – Bilbao
Teléfono: 94 601 42 07. Fax: 94 601 42 59
Email: [jtpmumaa, jtphiapm]@bi.ehu.es, [jtbgamua, jtbperui]@bipt106.bi.ehu.es

***Abstract**: This paper describes a simple architecture for ISPs to provide their customers with advanced information services: electronic mail certification and digital notary. Our proposed architecture is aimed to set a suitable base for increasing Electronic Commerce services, such as digital invoice generation and delivery, based on new standards like EDI/XML, or even Application Service Providers (ASP), which share a common messaging service: the electronic mail. Bearing in mind the present growing legal and standard framework for electronic business, our solution makes use of a well-known technology like Pretty Good Privacy (PGP) to offer these new services, avoiding the excessive complexity for establishing a Public Key Infrastructure. The project has been funded by the Basque Government and the private enterprise SARENET S.A.*

1 Introducción

Entre los diferentes servicios que actualmente se ofertan por parte de los proveedores de servicios de Internet (en adelante, ISPs) se puede establecer la siguiente división: en primer lugar, un ISP comercializa el acceso a Internet. Para ello habitualmente disponen de diferentes posibilidades dependiendo de la capacidad y tipo de enlace. Muy ligado al acceso está el servicio de correo electrónico, demandado y utilizado por prácticamente todos los usuarios y para el que éstos disponen de varias modalidades o tipos de cuentas. Además, y destinado principalmente a las empresas u organizaciones que quieran poseer sus propias páginas Web (aunque cada vez es más habitual que los usuarios residenciales dispongan de este tipo de servicios), cada vez es más común que los ISPs ofrezcan servicios de diseño, además del albergue de páginas Web.

Por otro lado, además de realizar todos los procedimientos administrativos necesarios para el registro de dominios, direcciones, etc., realizan labores de asesoría e ingeniería en muchas ocasiones (para implantar la solución de conectividad y servicios que mejor se adapte a la problemática que presenta cada empresa), y también facilitan las tareas de diseño, implementación y mantenimiento de los servidores que soportan las distintas aplicaciones según diferentes modalidades de servicio (web hosting, housing, etc.).

Además de estos servicios comunes prácticamente a todos los ISPs, éstos realizan grandes esfuerzos por comercializar y ofrecer a sus clientes servicios innovadores que, haciendo uso de los avances tecnológicos más recientes, permitan a éstos disponer de nuevas herramientas para optimizar y evolucionar en sus propias actividades. Entre estos nuevos servicios, en los últimos tiempos destacan sobre todo los relacionados con el comercio electrónico. Estos servicios de valor añadido, que están experimentando un gran auge y desarrollo actualmente, permiten a las empresas aprovecharse de las nuevas posibilidades que abre el comercio a través de la red, proporcionándoles los mecanismos y medios necesarios para disponer un canal de comercialización para la distribución y venta de productos y servicios, de una forma fiable y segura y que les posibilita el acceso a un mercado global a nivel prácticamente mundial.

Entre los servicios más novedosos en cuyo desarrollo se está trabajando actualmente, destacan: la facturación electrónica a través de Internet basada en estándares, proveedores de servicios de aplicaciones, notaría digital, etc. Muchos de ellos comparten por el momento un servicio común de intercambio de mensajes: el correo electrónico.

2 Servicios de intermediación desde ISPs

Tal y como se describe en el apartado anterior, la evolución que se ha experimentado en los servicios de datos de valor añadido comercializados por empresas pertenecientes al sector telemático, ha sido muy importante, especialmente en los últimos años en los que principalmente gracias al auge y difusión de Internet, tanto el incremento en el número de usuarios, como la demanda y desarrollo de nuevos servicios han sido espectaculares.

En este apartado se describen algunos de los servicios que más interés están despertando actualmente, tanto desde el punto de vista de los usuarios de los mismos (principalmente para el caso de empresas), como de los ISPs que de forma continua tratan de incorporar los últimos avances tecnológicos a sus ofertas para proporcionar a sus clientes las innovaciones más destacables para competir en sus diversas esferas de actividades, así como para poder ellos mismos destacar en un mercado altamente competitivo, y que evoluciona a un ritmo casi frenético.

Algunos de estos servicios, como es el caso del comercio electrónico, en constante evolución, llevan ya varios años siendo objeto de desarrollos e implementaciones, y aún quedan bastantes aspectos relacionados con los mismos que siguen dando lugar a diversos trabajos de investigación, y en los que no hay aún una estrategia en el mercado claramente definida, ya que además presentan distintas posibilidades en función de los muy diversos escenarios a los que pueden ser aplicados.

Otros servicios, sin embargo, como es el caso de la notaría digital, o los servidores de aplicaciones, responden a los últimos avances que se han producido en este sector, y están siendo objeto de investigaciones actualmente en diversos países.

En este trabajo, nos centramos en el servicio básico de correo electrónico, que prácticamente todo ISP proporciona a sus clientes. Está claro que la información intercambiada habitualmente a través de este servicio es inmensa y muy heterogénea, desde mensajes personales de toda índole, hasta notificaciones entre empresas u organizaciones relacionadas con peticiones de información, presupuestos, facturación y producción, o incluso pagos que, si bien están a la espera de sistemas avanzados de facturación electrónica, aún se encuentran lejos del escenario seguro que éstos les proporcionarán.

En este sentido, la legislación actual, aún inmadura y en continuo desarrollo, pretende establecer normas sobre el uso y desarrollo de la firma digital y de las autoridades de certificación basadas en infraestructuras de clave pública (PKI), con el objetivo de otorgarles un valor idéntico al de la firma y notaría manuscrita en el caso de transacciones electrónicas.

Las soluciones que se plantean en este trabajo pretenden dar herramientas técnicamente maduras y avanzadas para que los ISPs ofrezcan servicios de certificación de mensajes de correo electrónico a sus clientes. De esta forma, se impulsan soluciones cuya implantación es relativamente sencilla y que los ISPs puedan adoptar y ofrecer a los clientes que así lo soliciten en un plazo corto de tiempo, impulsando el empleo de nuevos servicios de comercio electrónico y a la espera de lo que el mercado y la legislación pretendan acordar acerca de un único sistema de certificación (probablemente PKI) que proporcione una seguridad total y homogénea a todos los agentes implicados en transacciones electrónicas.

3 Escenarios

Visto el rápido crecimiento del comercio en este nuevo formato electrónico, se trata de proporcionar nuevas facilidades a los usuarios, de forma que el despegue del mismo se produzca definitiva e inmediatamente. Englobado dentro de esta gama de nuevos servicios que puede ofrecer un ISP, se encuentra el servicio de certificación de correo electrónico, que se relaciona directamente con el comercio electrónico, ya que viene siendo empleado para el intercambio de documentos como facturas, contratos, etc.

Dentro del proceso típico de una actividad comercial a través de la red resulta necesario, en la mayoría de las ocasiones, el intercambio de información a través de un medio asíncrono, como es el correo electrónico. No obstante, en la mayoría de las ocasiones, debido a la información que se va a transmitir a través de este medio, se debe tener en cuenta la seguridad de los mensajes que se intercambian, además de todos aquellos documentos que pueden ir adjuntos. Por ello, durante los últimos años se ha prestado especial interés en aspectos criptográficos orientados a garantizar la confidencialidad, integridad de los mensajes, así como la autenticidad de los extremos receptor y emisor. Fruto de este esfuerzo, en la actualidad, se puede contar con sistemas como PGP, S/MIME o PEM, destinados a proteger la información transmitida a través del correo electrónico.

Básicamente, nos centraremos en tres escenarios principales en cuanto a la certificación de mensajes:

- Mensajes entre clientes del propio ISP, que consiste en ofrecer un servicio de certificación de correo cuyo objetivo son los propios clientes del ISP, de forma que se dé un mayor nivel de confianza a los mensajes de correo intercambiados entre ellos.

- Mensajes entrantes a los servidores de correo del ISP, cuyo origen es externo a éste, de manera que se certificaría la llegada de un mensaje para un usuario determinado en un instante concreto de tiempo.

- Mensajes salientes de los servidores del ISP, con destino en el exterior de éste, de

forma que se certifica el paso de un mensaje de correo saliente por el servidor en un instante de tiempo.

4 Certificación de correo

En este punto, se describe un sistema de certificación de correo basado en un servidor de claves PGP, que pretende enmarcarse en el primero de los escenarios planteados. La elección del sistema PGP se justifica basándonos en la experiencia que le respalda, lo que hace que se trate de un sistema más que probado, así como su amplia utilización entre los usuarios de la red.

4.1 Diseño del sistema

Un esquema global de la descripción de la solución se puede observar en la figura 1, en la que cada bloque representa un módulo funcional. En el esquema se pueden diferenciar a grandes rasgos las entidades involucradas, así como los bloques que presenta la parte del sistema que se encuentra en las instalaciones del ISP.

Dada la filosofía en la que se basa PGP, conocida como telaraña de confianza, se puede llegar a tener serias dudas sobre la veracidad de la relación entre una clave y la dirección de correo a la que está asociada. Para solucionar este inconveniente, se propone aplicar una filosofía similar a la de las PKIs, pero sobre las claves públicas PGP. En este sistema, el propio ISP actuará como entidad certificadora (CA) y entidad de registro (RA) asegurando, mediante su firma, que las claves públicas que presenta en su servidor de claves tienen una relación de garantía con la dirección de correo a la que están asociadas.

Así, aquellos usuarios que deseen intercambiar correo de forma protegida mediante PGP podrán disponer de las claves públicas de los posibles destinatarios en este servidor, ya que el servicio de consultas de claves será abierto a todo el público, dando a conocer las claves a toda persona interesada. Sin embargo, el servicio que permite añadir claves al servidor estará restringido a aquellos usuarios que tengan contratado el servicio. Este servicio ofrecerá la posibilidad de solicitud de firma de la clave pública por parte del ISP, lo cual dotará a la misma de la garantía mencionada.

A continuación se va a describir cada uno de los bloques que compone el sistema para poder entender mejor el mismo, y la relación entre cada uno de ellos.

4.2.2 Servidor WWW seguro

Puesto que el acceso a este servicio de adición, firmado y obtención de claves se va a producir vía web, deberá realizarse sobre un protocolo que proteja la transmisión de dicha información. Para ello se ha pensado en emplear una capa software intermedia como es SSL, la cual va a garantizar que la información no es observada y/o alterada en el medio, además de la veracidad de la identidad del servidor, proporcionando así al usuario la garantía de que el servidor es quien dice ser.

Las páginas de búsqueda de claves estarán disponibles para toda persona que lo desee, ya que se trata de dar a conocer las claves públicas al mayor número de personas posibles. Sin embargo las páginas de solicitud de firmado de claves e incorporación al servidor estarán restringidas únicamente a aquellos usuarios que hayan contratado el servicio.

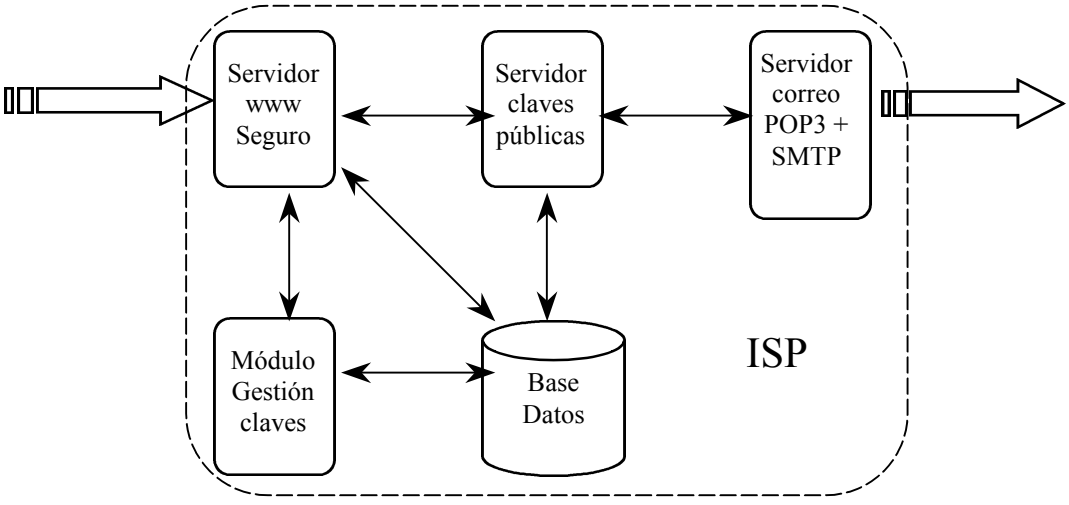


Fig. 1: Diagrama de bloques del servidor de claves públicas

4.1.2 Módulo de gestión de claves

Este módulo será el encargado de supervisar las claves que se añaden al servidor, y de decidir si el ISP debe firmar o no una clave. Los diferentes módulos funcionales que constituyen este bloque se representan en la figura 2. Se describe cada módulo de forma individualizada para comprender mejor el funcionamiento del sistema.

- Submódulo de aceptación de solicitudes

Las páginas HTML a través de las cuales los usuarios podrán añadir sus claves estarán protegidas mediante un login y un password, que cada usuario recibirá *offline* al contratar el servicio. Cuando el usuario consigue acceder a esta página tras introducir los datos correspondientes, se descargará un applet JAVA firmado por una entidad certificadora reconocida, como Verisign, Entrust, etc.

- Submódulo de gestión de solicitudes

Una vez que el usuario haya introducido sus datos, el applet se conectará con el módulo de gestión de solicitudes de firmado del mismo servidor, el cual decidirá si se debe firmar o no, consultando la información de identificación de cada usuario almacenada en la base de datos.

En el caso en que se decida no firmar la clave, se devolverá la misma al usuario, indicándole a través de una página HTML el motivo por el cual se ha rechazado la solicitud de firmado. Por el contrario, en el caso en que se decida firmar la clave, se pasará la misma al módulo de firmado.

- Submódulo de firmado

Este módulo será el encargado de aplicar la firma del ISP a la clave pública que el usuario ha introducido vía WWW, siguiendo las instrucciones del módulo anterior en la cadena. Posteriormente se almacenará la clave ya firmada en la base de datos propia, para que el servidor de claves públicas sea capaz de acceder a ella cuando reciba las solicitudes de claves, y presentar las claves actualizadas. Para su implementación, nos podemos basar en la librería de desarrollo de funciones criptográficas PGP disponible en lenguaje C. Además, de forma paralela, el servidor devolverá la clave pública al usuario en cuestión, para que éste pueda actualizarla en su anillo local, y pueda hacer uso de la misma a la hora de encriptar o firmar mensajes, así como enviar su propia clave a otros servidores de claves ubicados en la red.

- Submódulo de mantenimiento de claves

Por último, dentro del módulo de gestión de claves se puede observar el submódulo de mantenimiento.

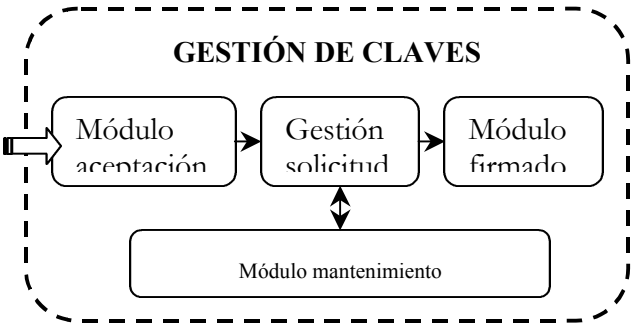


Fig. 2: Módulo de gestión de claves

Este módulo va a desempeñar labores muy diversas, todas ellas relacionadas con el mantenimiento del sistema, para que el funcionamiento del mismo sea el adecuado, y tratando de que se empleen los menores recursos posibles, especialmente en cuanto a espacio de disco se refiere.

Periódicamente, se realizarán revisiones para comprobar la validez de las claves almacenadas en la base de datos. Se trata de decidir si una clave ha dejado de ser válida, para lo cual deberemos tener en cuenta ciertos criterios como el periodo de validez de la clave pública, es decir, la fecha de caducidad de la clave, el periodo del contrato del usuario con el ISP, o comprobar si la seguridad de la clave privada ha sido comprometida. Todas estas claves que dejen de ser válidas se almacenarán en una lista similar a un CRL, para que los usuarios puedan comprobar aquellas claves que pierden su fiabilidad, y tener de este modo una comprobación *on line* siempre que lo desee, simplemente accediendo a las páginas de consulta.

Adicionalmente, éste submódulo deberá ser capaz de ofrecer al administrador del sistema una interfaz web a través de la cual llevar a cabo ciertas operaciones de mantenimiento del sistema, como la introducción en la base de datos de nuevos usuarios o la eliminación de los mismos. Para ello, se hará uso de applets de JAVA que transmitan la información sobre una capa de SSL para proteger la información, que puede ser confidencial, y modificar así las bases de datos de forma segura.

5 Notaría de mensajes

El servicio de notaría digital de mensajes se enmarca dentro del segundo y tercer escenario planteados, donde el objetivo es la propia certificación y sellado en el tiempo de los mensajes y archivos adjuntos que manejan los servidores de correo electrónico de un ISP.

En el apartado anterior se ha descrito una solución a los problemas de integridad, confidencialidad y confianza extremo a extremo en el envío de correos electrónicos. Sin embargo, en ningún momento se deja constancia de si los propios correos han sido

enviados o no, ni tampoco del instante de tiempo en el que ha sido realizada la entrega.

Este vacío de seguridad permite que un usuario del servicio de correo del ISP pueda negar la recepción o envío de un determinado correo electrónico e incluso permite también que éste pueda justificar una demora en el tiempo de recepción delegando la responsabilidad en el servidor de correo.

Para evitar este tipo de situaciones y para tener un mayor control sobre los correos electrónicos que manejan los sistemas servidores de correo se propone un sistema de notaría digital de mensajes.

5.1 Especificaciones

Este sistema de notaría de correo electrónico debe almacenar de forma segura el contenido del propio correo y posibles archivos adjuntos, garantizar su integridad de forma permanente en el tiempo y sellar en el tiempo los instantes en los que el servidor de correo recibe dicho correo entrante, así como los instantes de tiempo en que los correos abandonan el servidor de correo saliente.

A su vez, este sistema deberá ofrecer a los administradores la posibilidad de validar dicha certificación en cualquier instante de tiempo posterior a la certificación, permitiendo probar ante terceros que el correo que en su momento fue certificado sigue siendo válido.

El carácter de la certificación que va a realizar el sistema sobre los correos debe ser permanente. Una vez que un correo y su attach es certificado, ya no puede dejar de serlo. Debido a que la certificación se basa en el almacenamiento de los correos, este sistema debe garantizar el almacén permanente de los correos que certifica. Por tanto, se le debe dotar de los mecanismos necesarios para garantizar que un correo que ha sido certificado y almacenado no se corrompa, modifique, borre o pierda su carácter de certificación nunca.

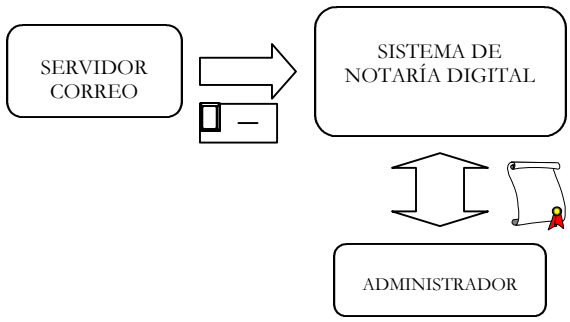


Fig. 3: Sistema de notaría digital

Toda la información que almacene el sistema de notaría (contraseñas, claves de usuario, los propios mensajes, etc.) deben ser ininteligibles incluso para el propio administrador del sistema de certificación.

Asimismo, el sistema debe crear los mecanismos necesarios que permitan controlar y auditar al propio sistema de certificación, para evitar y detectar cualquier actuación de mala fe por parte del propio sistema de notaría.

5.2 Funcionamiento

En el momento en que un mensaje de correo electrónico llegue al servidor de correo entrante, el demonio del módulo de certificación se encargará de todo el proceso de certificación.

Este módulo, en primer lugar, almacenará el correo en un módulo de almacenamiento, procediendo seguidamente a su certificación. El algoritmo de certificación empleado se basa en la modificación de una cadena de certificaciones que a su vez será almacenada en otro módulo de registro de datos garantizando la integridad de toda la cadena de sucesivas certificaciones de correos realizadas.

Cuando el administrador del sistema pretenda validar un correo previamente almacenado y certificado, deberá realizar una petición al servidor de validación. Éste procesará dicha petición y comprobará, consultando el módulo de almacenamiento y el módulo de registro de datos, si dicho correo existe y si sigue cumpliendo con la integridad de la cadena de certificaciones. En caso afirmativo dará por válido el mensaje y entregará un registro firmado por el propio sistema donde se certifica la validez del mensaje, los instantes de tiempo en que fue recibido y entregado, así como la huella del correo, pudiendo servir como prueba ante terceros. Si ha sido requerido, se procederá a la devolución del correo.

5.3 Algoritmo de certificación

Es la parte clave del sistema de notaría ya que debe garantizar la integridad permanente de los correos certificados. Para ello, se basa en el uso de funciones Hash.

Una función Hash H es una transformación que toma como entrada un mensaje m y que devuelve una cadena de bytes de tamaño fijo, llamado valor Hash h ($h= H(m)$). Estas funciones tienen unas propiedades que las hacen idóneas para su uso en criptografía, principalmente por su carácter unidireccional.

Cuando decimos que es unidireccional, nos referimos a que es muy difícil de invertir, es decir, a partir de la salida es computacionalmente imposible obtener la entrada. El valor Hash de un registro digital es como la huella digital de dicho registro, ya que lo identifica unívocamente. En este caso, la huella Hash de los mensajes que se quieren certificar y que se envían al servidor de certificación, identifican unívocamente a dichos archivos.

Si el sistema de notaría almacena en un lugar seguro dicha huella Hash, junto con el instante de tiempo en que se realizó la entrega de dicho correo, se puede certificar que en determinada fecha, un usuario concreto envió un determinado correo electrónico.

La función del sistema de notaría digital es almacenar los valores Hash de los correos que se quieren certificar junto con el sello en el tiempo, usuario, y resto de información relativa a dicha certificación, de una forma permanente que garantice su integridad.

Sin embargo, se podría pensar que el sistema tiene un punto débil, ya que esta forma de realizar la certificación facilita la posibilidad de que posibles intrusos modifiquen este registro de valores Hash, sin que sean detectados. Como solución, se utiliza una técnica de concatenación de valores Hash.

Esta técnica consiste en concatenar el valor Hash del archivo que queremos certificar con el valor Hash Raíz de la certificación inmediatamente anterior. Si calculamos el valor Hash de esta concatenación de dos valores Hash obtenemos el valor Hash Raíz de esta nueva certificación. Este nuevo valor Hash Raíz se utilizará para la concatenación en la siguiente certificación.

De esta forma construimos una cadena de valores Hash Raíz, en la que cada valor Hash Raíz resultante de una nueva certificación depende de todos los valores Hash Raíz de todas las certificaciones anteriores. Por lo que si un intruso pretende modificar los valores Hash referentes a la certificación de un correo, sería fácilmente detectable ya que no coincidiría con el resto de valores Hash de la cadena, violaría la integridad de toda la cadena de certificaciones y por tanto no sería validado.

5.4 Diseño del sistema

5.4.1 Módulo servidor de certificación

Este módulo, ante la llegada de un correo entrante, en primer lugar almacena dicho correo en el módulo de almacenamiento. A continuación, aplica una función Hash sobre el archivo y procede a certificarla.

Para ello accede al módulo de registro de datos, y toma el valor Hash Raíz de la última certificación. Concatena este valor Hash Raíz con el valor Hash calculado previamente del archivo que vamos a certificar y calcula de nuevo el valor Hash de esa concatenación; éste será el valor Hash Raíz de esta nueva certificación.

A continuación, almacena este nuevo valor Hash Raíz en el módulo de registro de datos así como otro conjunto de datos: un identificador de operación, el sello de tiempo del instante en que el correo el recibido, identificador de usuario de correo, el asunto del correo,... que nos van a permitir sellar en el tiempo y gestionar las posteriores validaciones de dichos correos.

5.4.2 Módulo servidor de validación

El módulo de validación atiende las peticiones de validación del administrador. Éste módulo realizará la función Hash de nuevo sobre el correo que

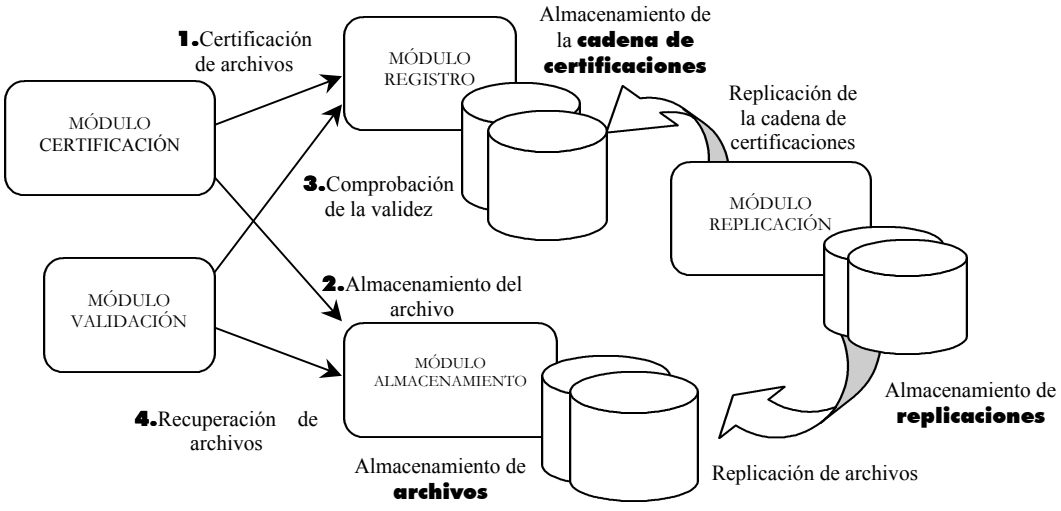


Fig. 4: Arquitectura funcional del sistema de notaría

pretende ser validado, y comprobará que coincide con su correspondiente lugar en la cadena de valores Hash almacenada en el módulo de registro de datos.

En primer lugar, debe recuperar del módulo de almacenamiento el mensaje que fue almacenado en el momento de la certificación. Para comprobar que coincide con la cadena de certificaciones deberá realizar el mismo proceso que en la certificación. Se obtendrá, de la misma forma, la huella digital de dicho correo y concatenando las huellas digitales de dicho correo y el Hash Raíz de la certificación anterior, obtendremos el valor Hash Raíz de dicho correo. Si éste coincide con el almacenado en el registro de datos en el momento de la certificación significa que dicho correo es el mismo que el que en su día fue certificado, por lo tanto queda validado.

A continuación, entregará un certificado firmado por el propio sistema de notaría con diferentes datos descriptivos del propio correo validado, como su huella Hash que lo identifica unívocamente, el sello de tiempo en el que el correo fue recibido por el servidor de correo, el remitente, asunto,... que permita probar ante terceros la validez del correo.

5.4.3 Módulo de registro de datos

Es el encargado de almacenar la cadena de certificaciones y todo el conjunto de datos necesarios para garantizar el correcto funcionamiento tanto de las certificaciones como de las validaciones.

Se usará un sistema de gestión de bases de datos, en el que se almacenará, para cada operación de certificación, un número de identificación de operación, el usuario de correo, la fecha en la que el correo es recibido (sello de tiempo), el valor Hash del archivo que se certifica, el valor Hash Raíz de esa certificación, el asunto del correo, etc. También almacenará datos relativos al estado de la misma base de datos, de los usuarios de correo, y de los correos almacenados.

5.4.4 Módulo de almacenamiento

Este módulo recibe las peticiones tanto de almacenamiento como de recuperación de los módulos de certificación y validación respectivamente.

Tiene la principal función de garantizar el correcto almacenamiento de todos los correos certificados en los sistemas de almacenamiento dedicados. Realiza todas las funciones de redundancia y de supervisión en las operaciones de almacenamiento que prevengan cualquier posible error o mal funcionamiento de bajo nivel. Garantiza la integridad permanente de los datos mediante mecanismos como las copias de seguridad, chequeos periódicos de consistencia entre copias, ...

A su vez debe garantizar que un correo que ya ha sido almacenado no va a poder ser modificado ni borrado. Para ello dispone de funciones específicas para detectar y evitar cualquier intento de modificación de un correo ya almacenado. Únicamente permitirá almacenar nuevos correos, nunca escribir sobre correos ya almacenados.

5.4.5 Módulo de replicación

Esta parte del sistema se encarga de realizar copias de seguridad tanto del módulo de almacenamiento donde se encuentran almacenados los correos, así como de la base de datos del módulo de registro de datos.

El módulo de replicación permite mantener copias exactas de todo el sistema de notaría, dando mayores características de seguridad y garantizando la integridad total del sistema. Las copias de seguridad se pueden almacenar dentro de la misma red física donde se encuentre el servidor de correo y el sistema de notaría, así como en localizaciones en redes físicas diferentes.

6 Conclusiones

Las empresas comercializadoras de servicios telemáticos han sufrido una gran evolución desde el comienzo de sus actividades en los años 80. Al principio, los servicios ofrecidos consistían básicamente en acceso a informaciones almacenadas en bases de datos, en diversos formatos, que en muchas ocasiones se ofrecían por parte de terceros, a través de las redes de transmisión de datos desplegadas.

Sin embargo, con el paso del tiempo, y a medida que las empresas aumentaban el grado de automatización de sus actividades, las necesidades de transmisión y de integración de sus diversos procesos fueron haciendo cada vez más sofisticadas y exigentes sus demandas, de forma que se produjo el despegue de nuevos servicios como por ejemplo EDI (Electronic Data Interchange).

Estos hechos, junto con el explosivo crecimiento experimentado en la utilización de Internet, dieron lugar a que se produjera una transformación en este mercado de empresas de servicios telemáticos de intermediación, dando lugar a la aparición de los denominados ISPs, cuyas actividades consistían básicamente en un principio en la comercialización del acceso a Internet. Sin embargo, el aumento de las demandas en el proceso de incorporación de las nuevas tecnologías a las formas de funcionamiento de sus clientes, hizo que los ISPs aumentaran rápidamente sus ofertas incorporando diversos

productos que trataban de dotarles de elementos diferenciadores frente a los demás, y que comenzaron siendo relativamente sencillos, como es el caso de albergue de buzones de correo, albergue de páginas web, diversos servicios de gestión e implantación de redes LAN, para su conexión a Internet, etc.

Sin embargo, y en un mercado tan dinámico como el que nos ocupa, un reto al que se enfrentan este tipo de empresas, consiste en atender las demandas cada vez más exigentes y sofisticadas de las diversas empresas, que también hacen uso de estos sistemas para competir en la mejor posición posible en sus diversas esferas de actividad, lo que da lugar a que los ISPs estén en una situación de continua búsqueda de soluciones para sus demandas.

En este contexto es en el que se sitúa este proyecto, en el que se ha tratado de dar una visión general del panorama actual en este sector, mencionando algunos de los servicios y campos que están dando lugar actualmente a un mayor número de desarrollos y trabajos de investigación en esta área. Algunos de ellos como es el caso del comercio electrónico, llevan varios años comercializándose según diversas fórmulas, aunque todavía parece no haberse encontrado una solución definitiva que satisfaga por completo a todos los agentes de este mercado, por lo que aún sigue dando lugar a numeroso esfuerzos por parte de distintos actores.

Frente a la excesiva complejidad tecnológica y económica que plantea hoy en día la implantación de una infraestructura de clave pública (PKI) para ofrecer servicio a sus clientes, y dentro del actual marco legislativo en continuo desarrollo en cuanto al entorno del comercio electrónico (autoridades de certificación, firma digital, etc.), la solución propuesta pretende ser una arquitectura de transición, basada en sistemas de certificación suficientemente probados y robustos, que se adapta perfectamente al entorno de funcionamiento en cuanto a transacciones electrónicas y, en concreto, intercambio de mensajes de correo electrónico, de los clientes de un proveedor de servicios de Internet.

El diseño planteado debe ser considerado como una solución a corto plazo, pensada como paso intermedio para proporcionar un mejor servicio hasta la llegada de otras posibles soluciones, basadas en PKIs, que en la actualidad dependen del desarrollo de los estándares definitivos, y en las que nuestro grupo de investigación también se encuentra trabajando de forma paralela al presente trabajo. Se trata de una solución relativamente fácil y rápida de implementar, lo que vendría las necesidades de clientes que demandan soluciones a corto plazo para esta nueva forma de realizar sus actividades comerciales.

Otros servicios de intermediación, sin embargo, son más novedosos, de forma que todavía es relativamente complicado encontrar empresas que los comercialicen, ya que se encuentran en plena fase de investigación, y de hecho son campos en los que se están realizando diversos estudios y proyectos por parte del grupo de investigación que se encarga del desarrollo de este mismo proyecto.

En cualquier caso, este proceso de evolución de las empresas de intermediación no se puede considerar a punto de finalizar, ni mucho menos, ya que sigue evolucionando de forma imparable, de modo que los ISPs no pueden considerarse en ningún momento establecidos con sus servicios actuales, sino que deben seguir escuchando las necesidades y demandas de las empresas de prácticamente cualquier sector para adecuar su oferta de la forma más estrecha posible a dichas demandas, para poder seguir compitiendo en un mercado tan competitivo y dinámico como el de los servicios telemáticos.

Agradecimientos

El trabajo presentado ha sido realizado gracias a la ayuda recibida del Gobierno Vasco dentro del proyecto OD00UN57, y a la colaboración de la empresa SARENET S.A.

Referencias

[1] Bruce Schneier. “Applied Criptography”. John Wiley & Sons. Octubre 1995.

[2] Marie Buretta. “Data Replication Tools and Techniques for Managing Distributed Information”. John Wiley & Sons. Febrero 1997.

[3] William Stallings. “Cryptography & Network Security: Principles & Practice”. Prentice Hall. Diciembre 1996.

[3] The International PGP Home Page.
http://www.pgpi.org

[4] Open Key Server.
http://www.highware.net/openkeyserver/oks.html

[5] Surety: Internet’s Digital Notary and Timestamping Service.
http://www.surety.com

[6] e-TimeStamp: Electronic Internet Notary.
http://www.e-timestamp.com

[7] RSA security.
http://www.xcert.com

Esquema de Votación Electrónica en Entorno Universitario

L. Moraga Ruiz de la Muela, A. Peinado Domínguez
Depto de Ingeniería de Comunicaciones,
E.T.S. Ingeniería de Telecomunicación, Universidad de Málaga.
Campus de Teatinos - 29071 MALAGA
Teléfono: 952 13 71 86 Fax: 952 13 20 27
E-mail: lmoraga@ic.uma.es, apeinado@ic.uma.es

Abstract. *This paper describes a practical voting scheme that making use of several cryptographic tools allows an election to take place entirely over a computer network. An implementation of anonymous channel by a combination of Proxy servers and SSL connections is proposed in order to assure voters' privacy. This solution seems to be the most suitable in a massive campus election where mobility should be preserved and changes in computer networks already installed or special hardware elements are not possible.*

1 Introducción.

La disponibilidad actual de equipos informáticos conectados a todo tipo de redes de propósito general y el desarrollo de herramientas y protocolos criptográficos hace posible afrontar el problema de la votación electrónica. Con este propósito han surgido diversos esquemas [2][3], cuya utilización presenta ventajas evidentes sobre el esquema tradicional de votación. Más concretamente, estos sistemas eliminan la necesidad de desplazarse al centro electoral, aumentando así el grado de participación en los comicios. Por otra parte, el formato electrónico de la papeleta de voto hace más fácil su posterior recuento, permitiendo una automatización del proceso, y evitando posibles indeterminaciones a la hora de interpretar su contenido. Además, se facilita el almacenamiento y la consulta con vista a posibles reclamaciones.

En este punto, la mayor dificultad con la que cuenta un sistema de votación electrónica es la misma que sufren actualmente los sistemas de comercio electrónico: la reticencia y desconfianza por parte de los usuarios a que sus datos personales se vean expuestos y se haga mal uso de ellos. La solución pasa por el desarrollo de esquemas robustos y seguros que permitan su implantación progresiva como complemento a los sistemas de votación tradicionales. Si, además, el esquema incluye la posibilidad de revisar de forma independiente el resultado del proceso de votación [2][4], se habrá conseguido aumentar la confianza de los votantes.

En este documento se presenta el diseño y la implementación de un esquema práctico de votación electrónica que puede ser usado para el desarrollo de una elección o encuesta. Mediante este esquema cada votante puede comprobar de forma independiente que no se haya producido fraude en la elección, e incluso posee pruebas suficientes que permiten probar si su voto ha sido manipulado.

La sección 2 describe las propiedades y requisitos que deben satisfacer los esquemas de votación electrónica. En la sección 3 y 4 se describen el esquema propuesto y la implementación particular de dicho esquema, respectivamente.

2 Propiedades de un Esquema de Votación Electrónico

La implementación de un sistema de votación electrónica no es sencilla. En la mayoría de los casos, no es suficiente un solo protocolo criptográfico y/o autoridad para satisfacer las propiedades deseadas; es por eso por lo que se habla de esquema de votación y no simplemente de protocolo de votación.

De acuerdo con [1], un sistema de votación electrónica puede ser definido como una aplicación distribuida, constituida por un conjunto de mecanismos y protocolos criptográficos que, unidos, permiten que un proceso electoral tenga lugar, por completo, en una red de ordenadores de forma segura, incluso asumiendo que los legítimos participantes actúen de forma maliciosa.

Existen una serie de características deseables en un sistema de votación comúnmente aceptadas por diferentes autores [1][2][3]. Estás características pueden resumirse como:

Precisión. Un esquema de votación es preciso si:

- Ningún voto puede ser alterado
- Ningún voto válido puede ser excluido del recuento
- Ningún voto no válido es tenido en cuenta en el recuento

Democracia. Un sistema es democrático si:

- Solo los votantes censados pueden votar

- Cada votante puede votar sólo una vez

Privacidad. Un esquema es privado si:
- Nadie puede asociar un voto a un votante
- Ningún votante puede demostrar que ha votado en un determinado sentido
- Todos los votos permanecen secretos hasta que la votación concluye

El esquema de votación electrónica aquí propuesto está basado en el modelo Sensus desarrollado por Kranor y Cytron [2], que consta de dos autoridades: un Registro y una Urna. El Registro es el encargado de validar los votos y la Urna se encarga de recogerlos y realizar el recuento. El Votante, mediante sucesivas conexiones con ambas autoridades, consigue tomar parte en el proceso electoral.

Sensus no trata el seguimiento de la conexión entre el Votante y la Urna, que puede conducir a violar la primera condición de privacidad. Sensus asume que ningún voto se puede asociar con su votante a partir del recorrido de los paquetes en la red. Dada la facilidad con que se puede realizar un análisis de tráfico en la red, el modelo presentado en este documento trata de subsanar esta debilidad incluyendo otro elemento, denominado servidor Proxy, entre el Votante y la Urna, con la intención de construir un canal anónimo y seguro entre ambos, basado en la idea de "mezclador" de Chaum [5].

Las características arriba expuestas, a menudo, son difíciles de satisfacer conjuntamente. Es el caso de la característica de precisión, que se consigue entregando al votante una prueba de validez de su voto, y la segunda condición de privacidad, referida a la posible venta de votos o a la coacción, que es posible realizar si el votante posee una prueba del voto. En [6] se establece que no es posible compaginar ambas propiedades, a menos que se recurra a disminuir la movilidad de los votantes (una de las principales ventajas de los sistemas de votación electrónicos), o se recurra a elementos hardware especiales en el lado del votante.

El esquema aquí desarrollado sacrifica la segunda condición de privacidad en aras de que el votante posea una prueba que le permita demostrar que ha votado de una determinada forma, en caso de que su voto no haya sido contabilizado adecuadamente durante el recuento. El sacrificio es admisible si el objetivo es aumentar la confianza de los electores en los sistemas de votación electrónicos sin limitar su movilidad y si no se quiere complicar el esquema con la inclusión de dispositivos hardware especializados.

3 Modelo del Esquema de Votación

El modelo del esquema de votación propuesto en este documento consta de una serie de elementos o entidades que actuando de forma conjunta consiguen llevar a cabo el proceso electoral.

Los elementos que forman parte del esquema son:

- El Votante (A), cuyo principal objetivo es tomar parte en la votación y mantener su identidad y su voto en secreto.

- El Registro (R), cuya misión es la de sólo permitir el voto a aquellas personas pertenecientes al censo electoral y evitar que una persona autorizada pueda votar más de una vez.

- La Urna (U), que tiene como misión recoger los votos, asegurar su validez y realizar el proceso de recuento.

- El servidor Proxy (P), que actúa a modo de canal anónimo entre el Votante y la Urna.

Los distintos elementos que forman el esquema de votación electrónica intercambian mensajes entre sí durante el proceso electoral. Este intercambio puede descomponerse en dos fases diferenciadas:

Fase de Registro. Intervienen sólo el Votante y el Registro. Los objetivos durante esta fase son:

- Autenticar al Votante, y asegurar así que solo las personas autorizadas completan el proceso de votación.
- Evitar que un mismo Votante pueda participar más de una vez en el proceso electoral.

Fase de Votación. Intervienen el Votante, la Urna y el servidor Proxy como intermediario entre ambos. Los objetivos en esta fase son:

- Asegurar que el Votante deposita su voto y este es contabilizado correctamente en el recuento final.
- Ocultar en todo momento la identidad y ubicación del Votante de forma que no sea posible asociar el voto con la persona que lo emite.

Para una fácil comprensión de los mensajes intercambiados este documento hace uso de la siguiente terminología:

- A -> B: m. El elemento A envía al elemento B el mensaje M
- m_1,m_2. Concatenación de los mensajes m_1 y m_2
- *Se*. Clave de sesión utilizada para cifrado simétrico
- $E_{Se}(m)$. Mensaje m cifrado con la clave *Se*
- *V*a. Clave privada del elemento A
- *B*a. Clave pública del elemento A
- $S_{Va}(m)$. Firma del mensaje m con la clave privada del elemento A.
- H(m). Resumen del mensaje m

3.1 Fase de Registro

Los pasos desarrollados durante esta fase (Fig. 1) están encaminados a que el Votante pueda conseguir una prueba del Registro que certifique la validez del voto que está intentando emitir como paso previo a su envío a la Urna.

Los pasos a seguir durante esta fase son:

Paso 1: El Votante (A) codifica el voto como una cadena de bytes (m)

Paso 2: A genera una clave de sesión (*Se*) que le permite cifrar m obteniendo el Voto Cifrado ($x=E_{Se}(m)$).

Paso 3: A genera el par (k,ki) siendo $ki=k^{-1}$ mod n.

Paso 4: A exponencia k con la clave pública del Registro (*Br*) y ciega el voto obteniendo el Voto Cegado ($x'=H(x)\cdot k^{Br}$).

Paso 5: A envía a R su identidad (a), x' y la prueba de votación ($S_{Va}(x')$).

$$A\text{->}R:\ x',\ S_{Va}(x').$$

Paso 6: R comprueba la firma $S_{Va}(x')$, y verifica que A está censado y no ha emitido ningón voto previamente. Si todo es correcto exponencia x' con su clave privada *Vr*, obteniendo $x'^{Vr} = S_{Vr}(x)\cdot k$ que envía a A

$$R\text{->}A:\ S_{Vr}(x)\cdot k$$

Paso 7: A obtiene la firma de x ($S_{Vr}(x)= S_{Vr}(x)\cdot k\cdot ki$)

De la necesidad del Paso 2 se habla en la Fase de Votación. En cualquier caso, lo que se persigue es, mediante un protocolo de no repudio de entrega entre el Votante y la Urna, que esta última no pueda negarse a aceptar el voto, por ejemplo, mediante negación de servicio selectiva en función del contenido del voto recibido.

En la Fase de Votación, el Votante entrega x y $S_{Va}(x)$ a la Urna para demostrar que se trata de un votante censado. $S_{Va}(x)$ es la prueba que entrega el Registro para dotar de validez a un voto. La dificultad de la Fase de Registro estriba en que el Registro debe firmar x sin saber su contenido, ya que entonces, en coalición con la Urna, dispondría

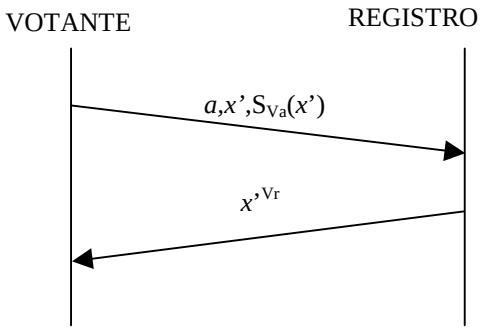


Figura 1: Fase de Registro

de información suficiente para asociar la identidad del Votante con su Voto, incumpliendo la primera condición de privacidad.

La utilización de las firmas ciegas propuesta por primera vez por Chaum [7], permite afrontar esta dificultad; de esta forma, el Votante obtiene la firma del voto cifrado (x) sin que el Registro pueda almacenarlo. El proceso de cegado (Pasos 3 y 4) transforma x obteniendo como resultado el voto cegado (x'), de forma que, mediante sencillas operaciones sobre este último se puede obtener la firma de x (Pasos 6 y 7).

Junto con x', el Votante entrega en el Paso 5 lo que se denomina una prueba de votación. Esta prueba es necesaria si se quiere evitar que el Registro firme con su clave privada votos que él mismo, u otro elemento, pudiera generar. La prueba consiste en la firma digital de x' ($S_{Va}(x')$) con la clave privada del Votante (*Va*). De esta forma, se limita la capacidad para crear pruebas de votos: sólo un Votante puede generar una prueba de votación. Al final del proceso de votación, el número de votos en la Urna será igual o inferior al número de pruebas de votación que posee el Registro comprobándose que sólo exista una prueba de votación por cada Votante. Para ello el Registro marca al Votante en una base de datos para asegurar que no se firma ningún voto más a dicho Votante, en la misma base de datos guarda x' y su firma $S_{Va}(x')$ como prueba de votación.

Si todos los pasos de la Fase de Registro se han realizado de la forma correcta el Votante posee la firma de su voto del Registro, lo que constituye una prueba de la validez del mismo; y todo esto

manteniendo su voto secreto de forma que el Registro no es capaz de asociar votante y voto.

3.2 Fase de Votación

Una vez el Votante posee su voto firmado por el Registro puede, si lo desea, completar el proceso. Durante la Fase de Votación, el Votante envía a la Urna el voto y la firma del Registro, la Urna comprueba si la firma es auténtica y contabiliza el voto.

Las dos mayores dificultades durante esta fase para el Votante son:

a) Que la Urna realice una denegación de servicio selectiva en función del valor del voto, rechazándolo o no contabilizándolo en el recuento.
b) Que la Urna conozca la identidad del Votante con lo que el voto deja de ser secreto.

Las dos formas de solventar estas dificultades pasan por:

a) Que el Votante obtenga un recibo de entrega antes de que la Urna conozca el voto y que pueda posteriormente comprobar que su voto ha sido contabilizado correctamente. En caso contrario, dispone de la prueba necesaria para demostrar el fraude de la Urna.

b) Que el Votante no desvele en ningún momento su identidad. El Votante no necesita hacerlo porque dispone de la firma del Registro que demuestra que el voto es válido.

El núcleo de esta fase es el protocolo de no repudio de entrega entre el Votante y la Urna (Fig. 2) que ofrece una solución a la denegación de servicio selectiva en función del voto.

Paso 1: A->U: $x,S_{Vr}(x)$

Paso 2: U->A: $id,S_{Vu}(id,x)$

Paso 3: A->U: *id,Se*.

En el Paso 1 el Votante entrega a la Urna lo que se denomina una promesa del voto (x) para obtener de la Urna una prueba de aceptación del voto antes de conocer su valor (Paso 2). De esta forma se evita que la Urna pueda negarse a aceptar votos dependiendo de su valor, porque no conoce *Se* que permite obtener el Voto (m) a partir de *x.*

En el Paso 2 la Urna comprueba que el voto es válido verificando la firma del Registro. Una vez comprobada la validez del voto la Urna asegura que no haya otro voto igual en su base de datos, con esto se evita que un votante trate de participar más de una vez en el proceso. La probabilidad de dos votos iguales procedentes de votantes distintos es casi nula, depende de que hayan votado lo mismo y que la clave de cifrado (*Se*) elegida por los dos votantes sea igual, por tanto, hay que dimensionar adecuadamente el tamaño de la clave de cifrado Se

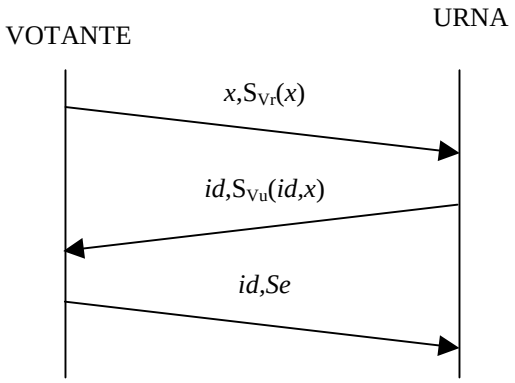


Figura 2: Protocolo de No repudio

para reducir al máximo esta probabilidad.

La Urna almacena el voto cifrado y la firma del Registro en un registro interno e identifica el voto con un número (*id*).

El Votante comprueba la prueba de aceptación verificando la firma de la Urna. Ahora, en caso de disputa, dispone de una prueba de que la Urna aceptó su voto.

El Votante puede ahora enviar la clave de sesión (*Se*) (Paso 3) que permita a la Urna descifrar m y poder contabilizar el voto en el recuento.

La Urna, con el número de identificación (*id*), localiza el voto cifrado (x) y lo descifra obteniendo el voto original (m). Tanto x, $S_{Vr}(x)$, *Se* como m, son expuestos para que todos los Votantes o autoridades que lo deseen puedan comprobar que el recuento ha sido correcto.

En esta primera aproximación desarrollada no hay datos en la información transmitida por el Votante a la Urna que permita su identificación, ni siquiera en el caso de que exista una coalición con el Registro. Este protocolo está diseñado para implementarse sobre una red de ordenadores. En cualquier red los equipos terminales están identificados por una dirección de red, en el caso de una red IP sería la dirección IP del equipo. Por lo tanto en la práctica es posible conocer la identidad del Votante a partir de la dirección del equipo desde donde ha enviado el voto mediante el seguimiento de la conexión.

Existen diferentes formas de afrontar el problema. En este esquema se opta por la creación de un canal anónimo entre la Urna y el Votante mediante la inclusión entre ambos de un elemento denominado Proxy (Fig. 3) en cuanto a que oculta la dirección del equipo desde el que se envía el voto.

Si se quiere que el Proxy tampoco sea capaz de asociar votante y voto, se debe poner especial cuidado en que la información que circula a través

suya sea cifrada convenientemente. La comunicación hacia la Urna se cifra con su clave pública (*Bu*), y los datos que la Urna envía al votante se cifran con una clave de sesión (*Seu*) que el votante genera y envía cifrada con *Bu* a la Urna.

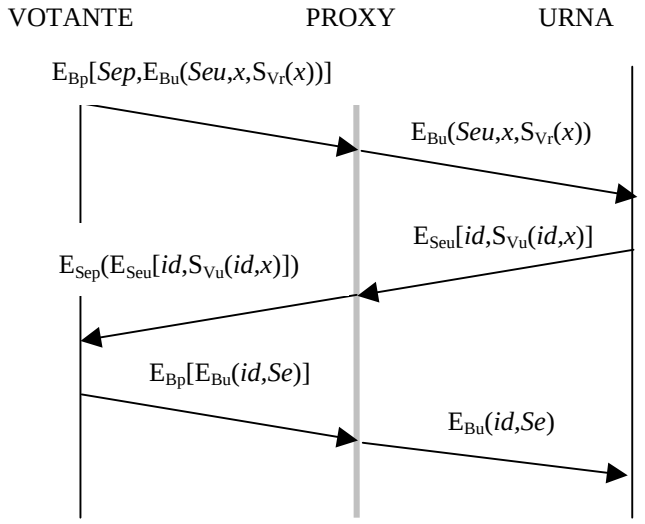


Figura 3: Fase de Votación con un solo Proxy

El canal entre el Votante y el Proxy debe ser seguro para que la Urna no pueda monitorizar la entrada de datos en el Proxy y así relacionar la identidad del Votante con el voto. Por eso se utiliza la clave pública del Proxy (*Bp*) para cifrar la comunicación hacia el Proxy. Para el sentido inverso, el Votante genera una clave de sesión *Sep* que comunica al Proxy cifrada con *Bp*.

En esta configuración existe la posibilidad de un acuerdo entre el servidor Proxy y la Urna que ayude a identificar al Votante que envía el voto. Una solución sería la implementación de una cadena de servidores Proxy (Fig. 4), el mensaje que el votante enviado a la Urna a través de la cadena de Proxies es (en negrita se encuentran los datos correspondientes al protocolo de no repudio):

1) A->P_i->P_j->...->Pn->U: E_{Bpi} (*i*, E_{Bpj}(*j*,...E_{Bpn}(*u*,E_{Bu}(***Seu,x,*****S_{Vr}****(x)**)...))

Donde *j*,*k* y *n* están formado por un número que identifican unívocamente al siguiente Proxy en la cadena y una clave de sesión que utilizará posteriormente el servidor Proxy para cifrar la respuesta de la Urna al Votante (*i*=id_j,Sep_i , *j*=id_k,Sep_j, *...).*

Para el proceso de envío de la firma de la Urna al Votante cada servidor Proxy utiliza la clave de sesión que el Votante generó *ex profeso* para él:

2) U->Pn->...->Pj->Pi->A:
$E_{Sepi}(E_{Sepj}(...E_{Sepn}(E_{Seu}(\mathbf{id,S_{Vu}(id,x)})...))$

Para el envío de la clave de sesión (*Se*) que permite a la Urna descifrar el voto se utiliza el mismo proceso que se realizó en el primer punto.

3) A->Pi->Pj->...->Pn->U:
$E_{Bpi}(E_{Bpj}(...E_{Bpn}(E_{Bu}(\mathbf{id,Se})...))$

Como se puede ver, la única forma de poder asociar Voto y Votante es mediante una coalición de todos los Proxies que intervienen en la cadena. Si además la cadena es configurada por el propio Votante de forma aleatoria y se incluye en cada Proxy un retardo aleatorio desde que recibe el mensaje hasta que lo envía, el proceso de rastrear la comunicación desde la Urna hasta el Votante se vuelve casi imposible.

El Votante ha de crear un número elevado de claves de sesión, de ahí la importancia de tener un buen generador de números aleatorios que hagan difícil la obtención de una clave de sesión a partir de claves anteriores.

4 Implementación

Las conexiones que implican la Fase de Registro y la Fase de Votación se adaptan a una arquitectura cliente-servidor, por tanto la implementación de este esquema incluye la creación de dos servidores que actúen como Registro y Urna respectivamente. Ambos servidores pueden residir en la misma máquina ya que el esquema se ha construido de forma que aunque exista coalición entre ambos elementos el fraude es imposible. En cualquier caso es recomendable la ejecución en máquinas diferenciadas debido a que las operaciones criptográficas necesarias son costosas computacionalmente, limitando el número de usuarios que pueden ser atendidos simultáneamente por el mismo servidor.

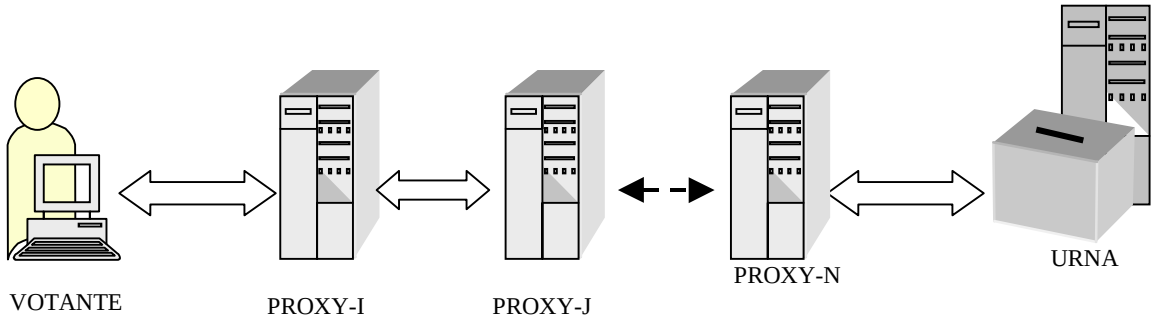


Figura 4: Cadena de Servidores Proxy

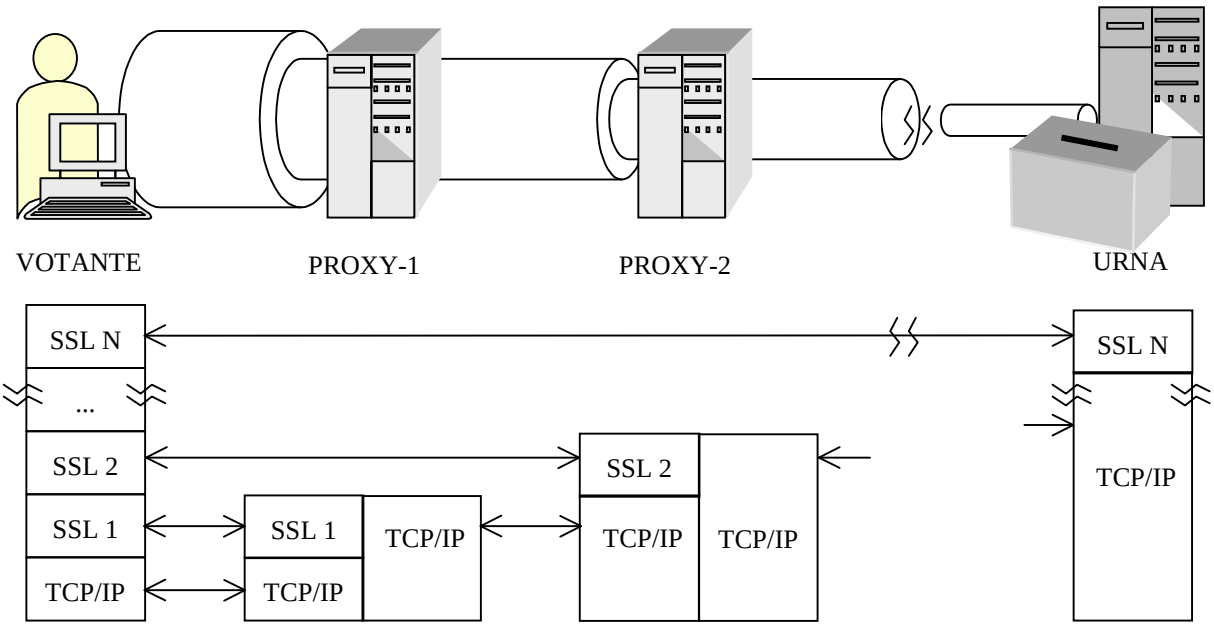
En la Fase de Registro es necesaria una autenticación mutua entre Registro y Votante y el protocolo SSL (*Socket Secure Layer*) se adapta a esta necesidad. SSL proporciona un canal seguro entre ambos elementos y en el *handshake* previo al establecimiento de la conexión es posible la autenticación en los dos extremos de la conexión.

El uso de firmas digitales y SSL implica como paso previo el establecimiento de una infraestructura de clave publica (PKI) encargada de la generación, distribución y mantenimiento de certificados. Para llevar a cabo el proceso de votación es necesario que cada votante posea previamente un certificado válido proporcionado por una Autoridad Certificadora (AC) que debe ser controlada por la junta u organismo encargado de velar por el correcto desempeño del proceso electoral. Este mismo organismo será el encargado de resolver las posibles disputas que pudieran surgir.

El protocolo SSL también puede ser usado para la Fase de Votación, sólo que en este caso la autenticación solo se produce en el lado del servidor, recordemos que el votante debe mantener su identidad en secreto durante esta fase. La ventaja del uso de SSL en esta fase reside en la facilidad de implementar la cadena de servidores Proxy mediante la superposición de canales SSL (Fig. 5). SSL proporciona un canal seguro entre dos puntos, de esta forma el votante establece una conexión segura con el primer servidor Proxy de la cadena, en esta conexión como en las siguientes existe autenticación solo en el lado del servidor con lo que el votante permanece en el anonimato. Encapsulado en el canal formado se crea otro canal desde el votante al siguiente servidor Proxy y así sucesivamente hasta llegar a la Urna. Esta arquitectura proporciona un intercambio de información similar al propuesto en la Fase de Votación.

La ventaja de esta arquitectura es clara si se utiliza una API para la programación de conexiones SSL, en nuestra implementación la API proporcionada por *el OpenSSL Group* (www.openssl.org), debido a que la generación de las claves de sesión requeridas entre cada servidor Proxy y el votante, así como su intercambio se realizan de forma automática durante el *handshake* previo a la conexión SSL. Otra ventaja importante de SSL es que puede utilizarse sobre cualquier protocolo de transporte que proporcione una conexión fiable extremo a extremo, en la implementación se escogió una red TCP/IP por ser el protocolo más utilizado permitiendo la participación en el proceso electoral desde cualquier lugar que posea una conexión a Internet.

La API de OpenSSL además se utilizó para programación de las funciones cirptográficas necesarias en ambas fases como son: creación y comprobación de firmas digitales, realización de resúmenes, realización de las operaciones matemáticas sobre grandes números necesarias para la implementación de firmas digitales ciegas, etc.



Conclusiones

Las elecciones universitarias son un buen campo de experimentación para los esquemas de votación electrónica. En este entorno universitario existe elevado número de personas con acceso a equipos conectados en red y que hacen un uso frecuente de ellos. Los esquemas de votación electrónico deben por tanto adaptarse a los protocolos y elementos disponibles, no solo por el coste que implica la instalación de equipos nuevos sino porque siempre existe una mayor aceptación al ser usados comúnmente.

El esquema que se ha presentado cumple estas premisas: TCP/IP es la base de la mayoría de las redes de ordenadores universitarias y SSL uno de los protocolos de seguridad más usados en aplicaciones con arquitectura cliente-servidor y comúnmente utilizado en el acceso a sitios web seguros.

Agradecimientos

Los autores quieren agradecer a los miembros de la lista de correo del *Openssl Project* su inestimable ayuda.

Referencias

[1] A. Riera i Jorba. *Design of Implementable Solutions for Large Scale Voting Schemes.* Universidad Autónoma de Barcelona, Diciembre 1999

[2] L.F. Cranor, R.K. Cytron. "Sensus: A Security-Conscious Electronic Polling System for the Internet". *Proceedins of the Hawaii International Conference on System Sciences*, January 7-10, 1997, Wailea, Hawaii, USA.

[3] A. Fujioka, T. Okamoto, K.A. Ohta, "A practical secret voting scheme for large scale elections", *Advances in Cryptology - AUSCRYPT '92*, 1993, (LNCS 718), Springer-Verlag, pp. 244-251.

[4] M.J. Radwin "An untraceable, universally verifiable voting scheme", December 12, 1995 Seminar in Cryptology. Professor Phil Klein. http://www.radwin.org/michael/projects/voting.html

[5] D. Chaum. "Untraceable Electronic Mail, Return Addresses, and Digital Pesudonyms", *Comunications of the ACM*, **24** (5), Febrero 1981

[6] A. Riera. "An Introduction to Electronic Voting Schemes", Unitat de Combinatòria i de Comunicació Digital Universitat Autònoma de Barcelona.

Mecanismos de seguridad en redes activas sobre arquitectura SARA

Marcelo Bagnulo[1], María Calderón[2], Bernardo Alarcos[3],David Larrabeiti[4]
[1,2,4] Departamento de Ingeniería Telemática. Universidad Carlos III de Madrid
Av. Universidad 30 - 28911 LEGANES (MADRID)
[3] Área de Ingeniería Telemática, Universidad de Alcalá de Henares
28871 Alcalá de Henares (MADRID)
{marcelo,maria,dlarra}@it.uc3m.es, bernardo@aut.alcala.es

Abstract. *Active network technology enables fast deployment of new network services tailored to the specific needs of end users, among others features. Nevertheless security issues still are a main concern when considering the industrial adoption of this technology. In this article we describe SARA (Simple Active Router-Assistant) architecture, an active network platform deployed in the context of the IST-GCAP project, and then consider security requirements detected in this architecture, concerning confidentiality, integrity, authentication, no repudiation and retransmission. Later, we present the security protocol proposed which intents to cover all imposed requirements, and finally we will address implementation perspectives using available technologies such as IPSec and SSL.*

1 Introducción.

Como evolución de los modelos de red tradicionales, la comunidad científica ha propuesto un nuevo modelo identificado por el término *redes activas* [6], [9]. La idea fundamental es añadir programabilidad a las redes. Las redes activas constituyen una arquitectura de red en la que los nodos de la misma pueden realizar procesamiento "a medida" sobre los paquetes que los atraviesan. Las redes activas producen un cambio en el paradigma de red: de nodos capaces exclusivamente de transportar octetos de forma pasiva, a nodos capaces de procesar los paquetes en cualquier capa de la pila de protocolos.

Las redes activas introducen el concepto de procesamiento específico de los paquetes en base a código móvil que se ejecuta en los nodos de la red. Esto quiere decir que los nodos de la red no son sistemas de procesamiento especializados en un protocolo de red, como sucede en la actualidad, sino que son plataformas de ejecución genéricas en las que se puede descargar dinámicamente código específico para el procesamiento de los distintos tipos de paquetes que se desee definir.

Enmarcado en el contexto del proyecto europeo IST-GCAP [3] (2000-2001) se ha desarrollado en la Universidad Carlos III, una plataforma básica de redes activas denominada SARA que opera sobre redes IPv4 e IPv6.

En esta plataforma, la descarga de código de una aplicación activa (AA) en un nodo activo se hace de forma dinámica cuando llega el primer paquete activo que hace referencia a dicha AA, y se realiza desde uno o varios servidores de código administrados por el proveedor de red. Esta aproximación asegura que en la red activa solamente se ejecutarán aquellas AA que han sido previamente validadas antes de su habilitación por el proveedor o administrador de la red activa, y no el código inyectado por cualquier usuario anónimo como preconizan otras plataformas [4].

Otro elemento importante en esta arquitectura es el concepto de Router-Asistente (Fig. 1), desarrollado por primera vez en la plataforma SARA. En esta arquitectura el router delega las funciones de procesado activo en un asistente presente en una red local de alta velocidad como si de un coprocesador externo se tratara. Esta decisión de diseño tiene implicaciones sustanciales, en primer lugar, permite que la ejecución de aplicaciones activas en la red tenga un impacto mínimo en las prestaciones ofrecidas por el router que debe poder transportar también paquetes convencionales (no activos) con el máximo rendimiento, por otro lado permite dimensionar la capacidad de proceso del nodo activo sustituyendo únicamente el Asistente si fuera preciso. Por último, permite que el router esté blindado frente a potenciales errores en la programación de las AA que podrían afectar al Asistente pero no al router.

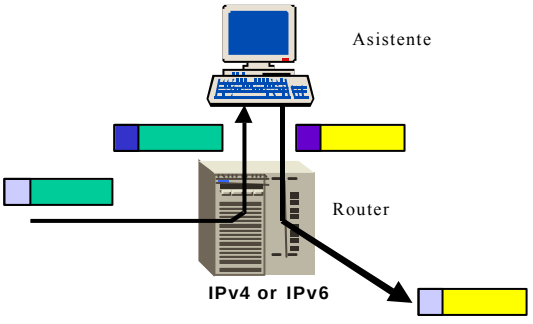


Figura 1. Arquitectura Router-Asistente

La plataforma desarrollada ofrece servicios activos transparentes. Este concepto tiene varias vertientes; por un lado es totalmente transparente para los paquetes tradicionales (paquetes pasivos) el hecho de que coexistan en la red con los paquetes activos y que existan en la red determinados nodos con soporte de redes activas. Por otro lado la topología de red activa es transparente para los sistemas finales, esto significa que estos últimos no están obligados a conocer la ubicación física de los nodos activos para poder hacer uso de sus servicios. Serán los propios nodos activos los que se encarguen de capturar los paquetes activos que los atraviesan. Esta funcionalidad se ha implementado haciendo uso de la opción *router alert* de IP.

En la actualidad SARA es una plataforma de redes activas de alto rendimiento consistente en un prototipo de experimentación que soporta las funcionalidades básicas de una plataforma de este tipo (disponible en: http://matrix.it.uc3m.es/~sara).

Si bien el esquema de funcionamiento de las redes activas ofrece una importante flexibilidad, que permite ofrecer una diversa gama de servicios de forma dinámica, presenta también, por su propia estructura de funcionamiento, una serie de riesgos de seguridad, algunos de los cuales pueden ser muy graves y poner en riesgo el correcto desempeño de toda la red, por lo que intentaremos proponer una solución a los mismos.

En el presente artículo se describirá el intercambio básico de paquetes activos en una red activa con arquitectura SARA para luego analizar los riesgos existentes y los requisitos de seguridad que estos imponen así como las condiciones de borde impuestas por restricciones de otro orden como ser la escalabilidad del sistema. En una segunda instancia se presentará una solución que pretende cumplir con los requisitos detectados para luego evaluar la implementación de la misma utilizando tecnología existente y así dar paso a las conclusiones extraídas.

2. Entorno de trabajo.

2.1 Intercambio básico de paquetes activos en SARA.

2.1.1 Elementos que participan en el intercambio.

Origen: ordenador del usuario de la red activa que genera tráfico en la misma y utiliza las facilidades activas que esta ofrece.

Destino: Ordenador al cual está dirigido el tráfico generado por *Origen*.

RACT: router activo (router + asistente) capaz de interpretar paquetes activos que circulan en la red y obtener el código necesario para procesarlos.

Servidor de código: Repositorio del código ejecutable utilizado por los routers para procesar las distintas clases de paquetes que circulan en la red.

2.1.2 Mecanismo básico.

Para hacer uso de las prestaciones ofrecidas por la red activa para el procesamiento particular de una clase de paquetes, *Origen* debe solicitar dicho servicio a la red (ver Figura 2). Esta solicitud se realiza mediante el envío de un paquete activo (ACT[1]), dirigido hacia el destino final deseado (*Destino*), y que adicionalmente indica a los routers activos del camino el código necesario para el procesamiento de los paquetes de esta clase.

Cuando un router activo recibe un paquete de este tipo, verifica la disponibilidad local del código solicitado. En caso que no posea el código necesario para procesar los paquetes, solicita el mismo al Servidor de Código (CODREQ [2]). El Servidor de Código envía entonces la información solicitada al router activo (COD[3]), quien puede ahora procesar el paquete activo y encaminarlo hacia el próximo salto (ACT[4])).

Los siguientes routers activos del camino realizarán un procedimiento análogo hasta que el último lo encaminará hasta *Destino*, quien ignora la información concerniente al tratamiento del paquete y extrae la información para las capas superiores.

Una vez que se ha establecido el camino y todos los routers contienen el código necesario para procesar los paquetes como lo ha solicitado *Origen*, este debe informar a la red la intención de continuar utilizando este código de procesamiento de paquetes. Para ello, *Origen* debe enviar periódicamente paquetes de refresco (Refresh) del código en uso. Los routers activos verifican el código solicitado y extienden su tiempo de vida en el router.

3. Análisis de riesgos y requisitos de seguridad.

Resulta claro a partir de la descripción previa, que el principal tema a resolver se vincula al control de acceso. Ya sea el control de acceso de los distintos *Origene*s a los códigos solicitados así como el control de cuales routers activos tienen permisos para acceder a los distintos códigos solicitados. También resulta de principal importancia controlar que servidores de código pueden introducir código en los routers activos. Estos requisitos y otros que se detallarán más adelante redundan en los puntos detallados a continuación.

3.1 Autenticación e integridad.

El riesgo más evidente que parece amenazar a la arquitectura descrita, es la posibilidad que alguna parte no deseada pueda cargar código en los routers

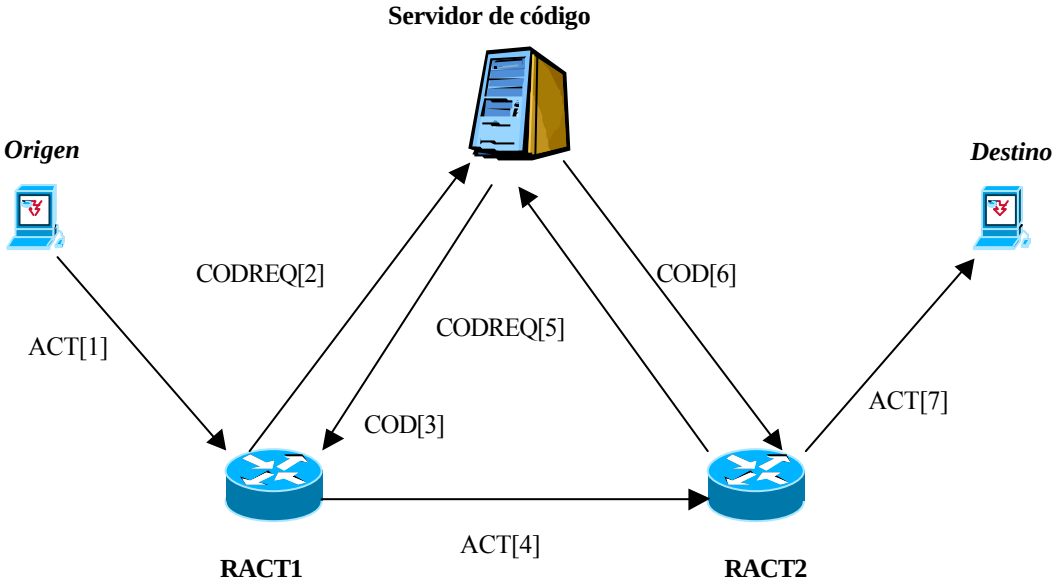


Figura 2

activos de la red, por lo que parece imprescindible que los mensajes que contienen el código (COD) a ser introducido, sean autenticados por parte del router activo de forma de estar seguros que fue el servidor quien generó dichos paquetes. Otro riesgo existente, íntimamente vinculado al anterior, es la posibilidad que un intruso altere el código contenido en el paquete (COD) mientras viaja hacia el router, por lo que también resulta necesario que dicho paquete posea una verificación de integridad asociada a la autenticación.

Adicionalmente, por razones de servicio, resulta deseable que sólo partes autorizadas sean capaces de ejecutar ciertos códigos, ya sea porque los mismos son potencialmente peligrosos para la red, (por ejemplo requieren muchos recursos del router), por lo que también es necesario poder identificar a *Origen* cuando solicita la ejecución de un código particular (ACT). Análogamente al caso anterior, resulta necesaria una verificación de la integridad de la solicitud realizada.

Finalmente, podemos pensar que algunos de los códigos existentes en el servidor no sean de dominio público por lo que sería deseable que sólo routers activos autenticados pudieran acceder al mismo.

3.2 Confidencialidad.

Como ya se ha mencionado en el párrafo anterior, parecería interesante que el código intercambiado no pudiera ser accedido por terceras partes que tienen acceso a la red, por lo que dicha comunicación (COD) debería ser confidencial. Los otros intercambios podrían ser confidenciales por razones menos críticas, como privacía de las intenciones de los clientes de los distintos servicios pero creemos que este es un requisito opcional para el resto de los paquetes.

3.3 No repudio.

Es posible imaginar que los operadores de las redes activas deseen cobrar de forma diferenciada en función de la forma de procesamiento de paquetes que sea solicitada a los routers activos mediante los paquetes ACT, por lo que dichos paquetes tendrán implicaciones comerciales y contractuales, específicamente jugará el rol de solicitud de servicio. Por lo tanto es interesante poder garantizar el no repudio de dichos paquetes por parte de *Origen*, ya que la tarificación del servicio se realizará en función de esto.

3.4 Retransmisiones.

El proceso de carga de código en los routers activos, es exigente en memoria y en capacidad de procesamiento, por lo que es posible imaginar un ataque a los routers de la red simplemente retransmitiendo paquetes válidos de un origen auténtico, lo que aumentaría la demanda en el router e incluso podría saturarlo. Por ello es importante que los paquetes que pueden cargar código (ACT) posean una validez temporal de forma que luego de un lapso ya no sean más válidos y sean descartados por todas las partes. Asimismo sucede con las solicitudes de código al servidor (COD) ya que se podría imaginar un ataque al mismo solicitando múltiples veces código ya pedido.

3.5 Requisitos adicionales.

3.5.1 Conocimiento nulo de usuarios por parte de los routers activos.

Resulta imprescindible diseñar el sistema de forma escalable, considerando una cantidad importante de potenciales usuarios, por lo que no resulta posible considerar una administración de permisos de usuarios local en cada router activo. Es necesario por consiguiente, la utilización de un servicio de directorio que contenga toda la información de usuarios y permisos, donde no sea necesaria la información de usuarios en los routers.

3.5.2 Transparencia frente a los routers activos que forman el camino.

Resulta importante que los distintos *Origenes* no deban conocer los routers que formen el camino hasta el destino, para brindar transparencia de la red e independencia de la topología de la misma, así como asegurar la escalabilidad, por lo que los paquetes activos enviados (ACT) no deben depender de los routers por los que deban transitar. Este requisito es particularmente fuerte parta un sistema de seguridad donde se requiere autenticación y confidencialidad con una parte a la cual no se conoce.

3.5.3 Eficiencia y velocidad en el procesamiento.

Recordemos que el objetivo principal de la subred es la transmisión de la información por lo que la velocidad en el procesamiento de los paquetes debe ser óptima y si bien los requisitos de seguridad son relevantes, la eficiencia en el uso es la razón de ser.

4. Arquitectura de seguridad.

4.1 Infraestructura necesaria.

Para el funcionamiento de la solución propuesta, resulta necesario contar con los siguientes elementos:

Origen: un par de claves asimétricas, pública (PubO) y privada (PrO) y un certificado digital que asocie dichas claves a *Origen* (opcional; ver 4.5).

RACTs: par de claves asimétricas, pública (PubR1 y PubR2) y privada (PrR1 y PrR2) y certificado digital que asocie dichas claves al router correspondiente (opcional; ver 4.5).

Servidor de Código: par de claves asimétricas, pública (PubSC) y privada (PrSC) y un certificado digital que asocie dichas claves al Servidor de Código (opcional;ver 4.5).

4.2 Definiciones.

Firma digital: calculamos el HASH de lo que deseamos firmar y luego encriptamos el mismo con la clave privada del firmante.

Verificar firma digital: calculamos el HASH de la información firmada y luego lo comparamos con el resultado de desencriptar con la clave pública del fimante el HASH recibido con la información recibida. En caso que coincidan, decimos que la firma ha sido verificada; en caso que no coincidan el paquete será descartado.

4.3 Protocolo de seguridad en el intercambio básico de paquetes activos.

ACT[1]: Como ya hemos dicho, este paquete es especialmente crítico y debe ser autenticado, no repudiable, íntegro y con validez temporal para evitar retransmisiones, por lo que se propone que el mismo contenga un sello temporal (timestamp) que establezca la hora de creación del mismo y se firme digitalmente el contenido activo del paquete con la clave privada de *Origen* (PrO) de forma de asegurar autenticidad, integridad y no repudio. Adicionalmente, este paquete debe transmitir una clave simétrica (K), previamente generada por *Origen*, que será luego utilizada por los routers para autenticar los mensajes de Refresh. Dicha clave debe ser transmitida de forma secreta y como *Origen* solo conoce al Servidor de Código (no puede conocer a los routers por requisito establecido previamente) K se encripta con la clave pública del servidor (PubSC).

Contenido de ACT
- Dirección origen: *Origen*
- Dirección destino: *Destino*
- Identificación de código deseado
- Clave simétrica K
- Timestamp

Firmado por *Origen*
Confidencialidad: clave simétrica K encriptada por PubSC

CODREQ[2]: El router activo analiza el paquete activo, pero por el requisito impuesto de no conocimiento de *Origen* por los routers, este no puede verificar si el paquete no ha sido alterado. Verifica si posee o no el código solicitado y reenvía el paquete recibido al Servidor de Código encapsulándolo en otro paquete e indicando si posee o no el código solicitado. En caso de que el router ya posea el código, este intercambio es utilizado para obtener autorización de acceso al código por parte de *Origen*.

Contenido de CODREQ
- Dirección origen: RACT1
- Dirección destino: Servidor de Código
- ACT
- Indicación si desea o no el código

COD[3]: Cuando el Servidor de Código recibe la solicitud (CODREQ), este verifica la firma de *Origen* utilizando su clave pública (PubO) para luego comprobar si *Origen* posee permisos para realizar la presente solicitud. En caso afirmativo, comprueba el timestamp, de forma que el momento de creación del paquete esté dentro de un entorno aceptable del momento presente. Adicionalmente verifica que el router activo solicitante posea los permisos para ejecutar el código solicitado. Una vez realizadas las verificaciones, desencripta la clave K utilizando PrSC. Genera entonces el paquete COD en el cual incluye el código en caso que el router activo así lo solicite y un timestamp. Firma luego la información utilizando PrSC para luego encriptarlo con la clave K. Finalmente adjunta la clave K encriptada con la clave pública del router (PubR1) de forma que sólo éste pueda leer la información enviada.

Contenido de COD
- Dirección origen: Servidor de código
- Dirección destino: Router activo 1
- Código solicitado
- Clave simétrica K encriptada con PubR1
- Timestamp

Firmado por SC
Confidencialidad para Router Activo 1 (encriptado por K, clave K encriptada con PubR1)

ACT[4]: Una vez que el Router activo recibe COD, este desencripta la clave K utilizando su clave privada (PrR1) para luego desencriptar el resto del contenido del paquete utilizando la clave K. Posteriormente verifica la firma digital del servidor utilizando PubSC. Una vez superadas estas corroboraciones verifica que el tiempo de generación del paquete (timestamp) se encuentre dentro de un entorno admisible del instante actual. En caso afirmativo, obtiene el código del paquete en caso que lo hubiera solicitado y asocia la clave K a dicho código, para luego cursar el paquete activo hacia el próximo salto.

Los siguientes routers activos del camino realizarán un procedimiento análogo al descrito anteriormente hasta que finalmente el último de los routers encaminará el paquete hacia *Destino*. Este simplemente descartará la cabecera de seguridad, ya que no es capaz de comprenderla y la información contenida no le presenta interés alguno.

Paquetes de Refresh: Una vez que se ha notificado a todos los routers activos del camino el código a utilizar, es necesario enviar paquetes que indiquen la extensión de la validez del código utilizado en el tiempo de forma que este permanezca en los routers. Dichos paquetes son generados por *Origen* y serán dirigidos a *Destino*; contienen la identificación del código deseado, una marca temporal y están firmados por *Origen*. Para que el proceso de verificación de firma sea menos exigente para los routers, se utiliza la clave K para autenticar los paquetes de refresh, adjuntando el hash del contenido del paquete concatenado con K.

Contenido de Refresh
- Dirección origen: *Origen*
- Dirección destino: *Destino*
- Identificación de código deseado
- Timestamp

Autenticación: hash del contenido del paquete concatenado con la clave simétrica.

Cuando los routers activos reciben el paquete de Refresh, verifican la firma comparando el hash contenido en el paquete con el hash resultante del contenido del paquete concatenado con la clave asociada al código, la cual obtienen de sus bases internas. Si además la marca temporal del paquete es correcta, extienden el tiempo de vida del código dentro del router y envían el paquete hacia el próximo router del camino. Una vez concluido el trayecto, *Destino* recibe el paquete, quien descarta la cabecera de seguridad.

4.4 Funcionalidades opcionales.

4.4.1 Confidencialidad.

Es posible modificar el protocolo propuesto para que todos los mensajes intercambiados sean confidenciales. Si bien esto puede ser deseable en ciertas situaciones, normalmente creemos que es una tarea muy exigente en cuanto a procesamiento, por lo que debe ser opcional su implementación dentro del protocolo. A continuación detallaremos como deberían encriptarse los paquetes intercambiados:

ACT: Como se ha descrito anteriormente, este paquete transporta una clave K encriptada con la clave pública del servidor de código (PubSC). El resto de información se encripta con la clave K para mejorar la velocidad de encriptado ya que los algoritmos de encriptado de claves simétricas son menos exigentes que los de claves asimétricas. El problema que surge en este punto es la imposibilidad del router de comprender la identificación del código solicitado por *Origen*. Las posibilidades entonces son o bien enviamos dicha información en texto en claro o bien el router no lo comprende y el servidor de código siempre envía el código solicitado independientemente de que el router ya lo posea. La primera opción presenta el inconveniente que la identificación de código no es confidencial y el encriptado entonces protege poca información y poco relevante por lo que es dudoso que justifique el esfuerzo. La segunda opción aumenta el overhead del protocolo de seguridad pero puede ser compatible con una red donde el tiempo de vida del código en los routers es corto.

Refresh: Los paquetes de refresh pueden ser encriptados por la clave K que ya es conocida por todos los routers del camino y corresponde a un algoritmo de clave simétrica.

4.5 Certificados digitales.

Hasta el momento hemos supuesto que el Servidor de Código conocía todas las claves públicas de todos los orígenes posibles y que además la administración de los permisos de acceso a los distintos códigos residentes en el servidor para cada usuario era realizada también en el Servidor de código. Si bien esto es posible, existen dificultades en el momento de escalar al solución, en particular cuando los usuarios pertenecen a diversas esferas administrativas.

Una solución posible es la utilización de certificados digitales para los orígenes. Para ello, cada vez que se da de alta un nuevo usuario, se debe generar un certificado digital que asocie la identificación de usuario con la clave pública del mismo. Ahora cuando dicho usuario desee enviar un paquete firmado, debe además de encriptarlo con su clave privada adjuntarle el certificado. Para verificar la firma del paquete, el servidor deberá verificar la autenticidad del certificado para luego verificar la firma. En este escenario, el Servidor de código solo debe conocer la clave pública de aquellos emisores de certificados digitales autorizados para usuarios del sistema. Adicionalmente, se podría incluir dentro de los certificados los permisos de usuario que informarían al Servidor de Código si el usuario posee o no permiso para ejecutar el código solicitado, de forma que estos permisos también fueran administrados por el emisor de certificados y no por el servidor de Código. El problema que puede presentar esta solución es la dinámica de los permisos de usuarios. Para ello se podría elaborar un sistema de certificados de atributos, es decir certificados que se generen dinámicamente a pedido en un servidor de certificación central presentando el certificado inicial.

5. Consideraciones sobre la implementación del protocolo.

Una vez definidos los requisitos y el protocolo de seguridad deseado, debemos evaluar las posibles implementaciones del mismo. Para ello, lo más recomendable es el estudio de las diversas implementaciones de protocolos de seguridad existentes, como elementos funcionales sencillos sobre los que construir la solución propuesta. Los protocolos que nos resultaron más interesantes, por su amplia difusión, fueron SSL e IPSec, por lo que se evaluarán a continuación.

5.1 Secure Socket Layer.

SSL es un protocolo de capa de aplicación utilizado para el establecimiento de una sesión segura entre dos partes. Incluye una etapa de handshake en la que se intercambian claves de forma segura, basándose en que al menos una de las partes posee un certificado. Este protocolo parece un posible candidato para el transporte seguro de código entre el servidor y los routers activos. Sin embargo, cabe notar que la negociación de una sesión de seguridad antes mencionada puede añadir mucho overhead, tanto en el procesamiento como en el ancho de banda necesario.

5.2 IPSec.

IPSec es un conjunto de protocolos diseñados por el IETF para brindar seguridad a IP. Originalmente fue definido para IPv6 pero luego se extendió de forma que pueda ser utilizado con IPv4. El objetivo de IPSec es la comunicación segura entre dos partes remotas. Cabe notar en esta instancia que el objetivo de IPSec es proteger los paquetes intercambiados de los dispositivos intermedios, como pueden ser los routers, es decir exactamente lo contrario de lo que deseamos hacer, es decir, dar indicaciones a dichos dispositivos. Es razonable esperar entonces que se deban realizar al menos ciertos cambios en el espíritu de IPSec. En particular los mecanismos de seguridad son aplicados a todo lo que contiene el paquete IP, incluyendo toda la información de usuario, por lo que, para aplicarlo en nuestro modelo, además de los routers, *Destino* también debe poder comprender los paquetes. Para ello, se podría establecer un intercambio de claves inicial entre *Origen* y *Destino* de forma que ambos conozcan la clave K utilizada, este intercambio podría realizar utilizando Diffie-Hellman o haciendo uso de certificados digitales.

IPSec está compuesto esencialmente de AH, ESP y IKE. Estudiaremos a continuación los posibles usos de estos para el protocolo propuesto.

AH – Authentication Header, brinda facilidades de firma y control de integridad. AH podría utilizarse para todos los paquetes firmados del protocolo. En particular parece un formato idóneo para los paquetes de solicitud (ACT) y para los paquetes de refresh que son simplemente firmados.

ESP – Encapsulating Security Payload, brinda facilidades de firma, control de integridad y confidencialidad, por lo que resulta interesante su utilización para los paquetes que transportan código desde el servidor a los routers activos.

IKE – Internet Key Exchange, posibilita el intercambio de claves entre dos partes para el posterior uso de las mismas con AH y/o ESP. Este protocolo podría ser utilizado para el acuerdo de la clave K entre *Origen* y *Destino* previo al comienzo del protocolo en sí de forma que *Destino* pueda comprender los paquetes.

5.3 Timestmap

En lo que refiere a las marcas temporales incluidas en los paquetes para evitar ataques de retransmisión, cabe notar que existe un compromiso entre los requerimientos de sincronización entre los distintos elementos involucrados y la seguridad del sistema. Los paquetes serán válidos si se han generado dentro de un entorno del momento en el que se recibe y verifica. A medida que el entorno aceptable es mayor , menores son los requerimientos de sincronización y menor es la seguridad del sistema. En el momento de implementación, será necesaria una sintonización de los dispositivos de forma de obtener una solución operativa y que brinde un nivel aceptable de seguridad.

6. Trabajos relacionados.

Existen diversas arquitecturas de redes activas desarrolladas en distintos proyectos y todas ellas, en mayor o menor grado, intentan resolver los diferentes problemas de seguridad resultantes de la propia estructura. Cabe notar que los riesgos detectados en cada caso dependen fuertemente de la arquitectura de redes activas utilizada por lo que las soluciones propuestas pueden diferir considerablemente de un proyecto a otro. En el presente trabajo se han estudiado principalmente requisitos de tipo "dinámico" [6], es decir requisitos que deben realizarse durante el procesamiento de ciertos paquetes críticos. Las herramientas criptográficas utilizadas en la solución, es decir criptografía de claves asimétricas y simétricas, funciones de hash y certificados digitales son técnicas conocidas y utilizadas en diversas soluciones de seguridad en redes tanto activas como convencionales, IPSec por ejemplo. Las diferencias relevantes con otros esquemas de seguridad se encuentran principalmente en el protocolo de seguridad planteado el cual se ha diseñado para cumplir con los requisitos definidos como relevantes en el presente entorno de trabajo y que el mismo ha sido diseñado de forma de beneficiarse de las características particulares de la arquitectura SARA. Cabe destacar que el protocolo propuesto cumple naturalmente con los requisitos referentes a transparencia frente a los routers activos que forman el camino y al conocimiento nulo de usuarios por parte de los routers activos, basándose esencialmente en el papel desempeñado por el Servidor de código y no exige una negociación de una sesión de seguridad entre las partes ya que toda la información necesaria se encuentra en el paquete activo (ACT). Otras arquitecturas como SANE [1], al carecer de una figura central que tome el papel de administrador, requieren una negociación entre el origen de los paquetes y cada uno de los nodos activos del camino, dificultando así la escalabilidad de la solución así como la introducción de nuevas facilidades como agentes móviles. Soluciones alternativas para estos problemas conocidas como "single packet authentication", son costosas y presentas diversos inconvenientes [1]. Otras arquitecturas como ANTS, basan su esquema de seguridad en la restricción del entorno de ejecución de código de forma de un paquete activo no sea capaz de consumir todos los recursos de red [9] y no consideran requisitos adicionales como autenticación de origen o no repudio. Cuán idónea resulte dicha solución dependerá del ambiente de trabajo en el que se pretenda implantar el sistema.

7. Conclusiones.

En el presente trabajo se ha presentado un protocolo de seguridad cuyo objetivo es solucionar los riesgos existentes en la implementación SARA de redes activas, previamente analizados. La solución propuesta al cumplir con los requisitos identificados, particularmente con los que hemos llamado conocimiento nulo de usuarios por parte de los routers activos y transparencia frente a los routers activos que forman el camino, resulta escalable ya no requiere una negociación entre el origen y cada uno de los nodos activos del camino, permitiendo una autenticación mediante un solo paquete, a saber el paquete activo que se desea enviar. La escalabilidad está basada también en un esquema centralizado de administración de permisos de control de acceso, el cual es posible gracias a la arquitectura de SARA. Estas características implican que el protocolo impone una carga relativamente baja a los routers activos. Adicionalmente parece posible una implementación de la solución utilizando en parte tecnología disponible, IPSec y/o SSL. En particular es especialmente interesante la evaluación del nivel de exigencia de procesamiento que impone el protocolo al router, así como el overhead generado por el esquema de seguridad (especialmente cuando se utilizan certificados digitales).

Agradecimientos.

El presente trabajo se ha realizado en el marco de los proyectos TEL99-0988-C02 y GCAP IST-1999-10 504 de CICYT (Comisión Interministerial de Ciencia y Tecnología)

Referencias.

[1] D. Scott Alexander, William A. Arbaugh, Angelos D. Keromytis y Jonathan M. Smith, "A Secure Active Network Environement Architecture", IEEE Network, [1998]

[2] Doraswamy, Naganand, "IPSec : the new security standard for the Internet, Intranets and virtual private networks", Upper Saddle River (New Jersey) : Prentice Hall PTR , [1999]

[3] Servidor www proyecto IST GCAP (Global Communication Architecture and Protocols for new

QoS services over IPv6 networks). http://www.laas.fr/GCAP/

[4] D. Wetherall, J. Guttag, and D.L. Tennenhouse. "ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols". IEEE OPENARCH'98, San Francisco, CA, April 1998.

[5] S. Kent, R. Atkinson, "Security Architecture for the Internet Protocol", Request for Comments: 2401, Network Working Group, [1998]

[6] Konstantinos Psounis, "Active networks: applications, security, safety and architectures", IEEE Communications Surveys, [1999]

[7] Schneier, Bruce, "Applied cryptography : protocols, algorithms and source code in C", New York, John Wiley & Sons , [1996]

[8] Urueña, Manuel, "Diseño de una plataforma de redes activas basada en arquitectura router-asistente", Facultad de Informática, UPM, [2001].

[9] David Wetherall, Ulana Legedza y John Guttag, "Introducing new Internet services: Why and How", IEEE Network Magazine [1998]

Introducción de Aplicaciones UDP en Redes Privadas Virtuales

Jorge Dávila[1], Javier López[2], Rodrigo Román[2]
[1] Facultad de Informática, Univ. Politécnica de Madrid
jdavila@fi.upm.es
[2] E.T.S. Ingeniería Informática, Univ. de Málaga
{jlm, roman @lcc.uma.es}

Abstract: *Virtual Private Network (VPN) solutions mainly focus on security aspects. However, when security is considered the unique problem, some collateral ones arise. VPN users suffer from restrictions in their access to the network. They are not free to use traditional Internet services such as electronic mail exchange and audio/video conference with non-VPN users, and to access Web and Ftp servers external to the organization. In this paper we present a new solution, located at the TCP/IP transport layer and oriented to UDP applications that, while maintaining strong security features, allows the open use of traditional network services. The solution does not require the addition of new hardware because it is an exclusively software solution. As a consequence, the application is totally portable.*

1 INTRODUCCIÓN

Las necesidades actuales de las empresas en el ámbito de las telecomunicaciones ha aumentado de forma ostensible debido a diversos factores, como la globalización de la economía, que aumenta la necesidad de intercomunicación entre sedes de una misma empresa, y a los acuerdos con otras empresas, que imponen la utilización de recursos compartidos. Por lo tanto, reducir el costo de la infraestructura de telecomunicaciones es algo prioritario.

Normalmente, y hasta hace poco, una empresa alquilaba líneas de comunicaciones para intercomunicar los ordenadores de su empresa. Un ejemplo de esto pueden ser las líneas Frame-Relay [1, 2]. Pero esta solución es inconveniente, debido a diversos factores (falta de competitividad en el sector, coste de las líneas, latencia en la comunicaciones,...).

Sin embargo, ahora se pueden utilizar *Redes Privadas Virtuales* (RPVs), mediante las que se construyen canales privados de información sobre un canal de comunicación público (por ejemplo, Internet). La forma de lograr ese canal privado es a través de la utilización de mecanismos criptográficos. Así, una RPV permite enviar información privada entre equipos informáticos usando una línea de comunicación pública y, por lo tanto, sin el elevado coste asociado al alquiler de las líneas dedicadas.

Además, una RPV posee otros beneficios, como la reducción de costes en el mantenimiento, debido a que está montada sobre una red de comunicación pública, con lo que el soporte técnico y la competencia es mayor que en una red alquilada. También es más fácilmente escalable, pues el equipamiento y el trabajo necesarios para añadir un ordenador suele ser menor que en una red dedicada.

No obstante, el uso de las RPVs tiene algunos inconvenientes, como por ejemplo, la obligación de compartir el tráfico privado con el tráfico de otros usuarios de la red pública, introduciendo así esperas indeseadas en el envío de la información. Otro gran inconveniente es la mayor probabilidad de que la información sea interceptada durante su transmisión. Además, si el proveedor que ofrece el servicio a la red pública tuviera problemas, la parte de la RPV asociada a ese proveedor quedaría inoperante.

Estos inconvenientes pueden ser solucionados mejorando los aspectos de seguridad (utilizando un sistema de seguridad bien diseñado), velocidad y fiabilidad (utilizando un acceso rápido y fiable a la red Internet). No obstante, si la seguridad es el objetivo principal de la RPV, existe otro gran inconveniente, y es que los usuarios sufren restricciones para acceder a los recursos de Internet que no pertenezcan a la RPV.

Este trabajo presenta una solución a ese problema. Hemos desarrollado una nueva solución de seguridad para RPVs localizada en el nivel de transporte y totalmente software. Al funcionar en el nivel de transporte, permite usar los mismos servicios tradicionales de Internet. Y al ser software, es una solución muy flexible. Todo ello, manteniendo los criterios de seguridad necesarios. La solución, *SEC-Insel* [3], ya ha sido desarrollada para ser utilizada por los servicios TCP dentro de la RPV. Ahora nos centramos en el desarrollo de un sistema, *SEC-Insel UDP*, que permita utilizar los servicios UDP.

El resto del trabajo está estructurado de la siguiente forma: en la sección 2 se argumenta por que es

necesario la incorporación de los servicios UDP en el ámbito de las RPVs. La sección 3 presenta los posibles mecanismos para implementar una RPV, y la solución propuesta en este trabajo. En las secciones 4 y 5 se expone la arquitectura de los dos módulos, denominados *Secsockets UDP* y *RPV-Insel UDP* sobre los que se basa la nueva solución. En la sección 6 se muestra un escenario de la solución propuesta, y la sección 7 finaliza con las conclusiones.

2 LA IMPORTANCIA DE UDP

En el nivel de transporte de la pila de protocolos TCP/IP, TCP proporciona un servicio confiable orientado a la conexión, es decir, los paquetes llegan sin error y en el orden en el que se envían. Por otro lado, UDP es un protocolo que proporciona un servicio orientado a datagramas, no asegurando que los paquetes lleguen a su destino, y si llegaran, no garantizando su orden.

UDP es un protocolo más simple que TCP, y mucho menos fiable, aunque más rápido. Es útil para aplicaciones que sean simples, que no necesiten de una transmisión fiable de datos, o incluso que necesiten que sus datos sean transmitidos lo más rápidamente posible. Un ejemplo de aplicaciones que utilizan UDP son aquellas aplicaciones que realizan tareas simples, TIME [4], que sincronizan y monitorizan redes usando SNTP [5] o SNMP [6], o que realizan transmisión de audio/video, usando RTP [7] y RTSP [8]. Estas últimas merecen una especial atención.

Los protocolos de audio y video están adquiriendo una gran importancia en la actualidad, debido a la gran ayuda que pueden prestar en el ámbito académico y empresarial. Así, mediante videoconferencia y audioconferencia, los miembros de diversas sucursales de una empresa pueden comunicarse, y grupos de investigación de varios países pueden compartir sus opiniones y resultados de trabajos utilizando para ello la red pública Internet, con el consiguiente abaratamiento en los costes.

Además, utilizando el mecanismo de *streaming* (envío de flujo de información por demanda de un servidor a muchos clientes), se pueden escuchar o visualizar contenidos previamente grabados o que están siendo filmados en tiempo real. Un ejemplo de esto son las Webcams.

Las aplicaciones de audioconferencia y videoconferencia utilizan UDP porque no necesitan de los mecanismos de comunicación fiable que TCP ofrece. Estos mecanismos minimizarían el tamaño de ventana, aumentarían el número de paquetes a enviar durante la comunicación, y no se adecuarían al envío de información en tiempo real. Es decir, TCP perdería el tiempo tratando de retransmitir paquetes que no llegasen a su destino.

Además de utilizar UDP, estas aplicaciones necesitan de un protocolo que, utilizando los servicios que UDP nos ofrece, gestione todo el proceso de comunicación, ya que la transmisión de audio y video no consiste sólo en enviar imagen y sonido. También es necesario controlar una serie de parámetros, tales como la calidad de la información enviada, los codecs empleados, un nº de secuencia, y otros aspectos de la comunicación. El protocolo más utilizado para este fin es RTP, que utiliza los servicios de UDP para proporcionar funciones que construyan aplicaciones multimedia.

Por lo tanto, UDP es un protocolo que está adquiriendo cada vez mayor importancia en el campo de las tecnologías multimedia sobre Internet. Por ello, en este trabajo se plantea la necesidad de añadir seguridad a servicios que utilicen este protocolo en el entorno de una RPV.

3 COMUNICACIÓN PRIVADA PARA SERVICIOS UDP

3.1 OTRAS SOLUCIONES

El objetivo que nuestra solución persigue es dotar a las aplicaciones UDP de la RPV de mecanismos de seguridad, pretendiendo que ésta sea lo suficientemente flexible como para acceder a equipos externos a ella, manteniendo, por ejemplo, una videoconferencia privada y una pública al mismo tiempo. Además, ha de ser fácil de mantener y con bajo coste.

Para lograr tal objetivo podríamos basarnos en algún esquema existente. Pero realmente no se ha diseñado ningún mecanismo estándar específico para lograr un entorno seguro para aplicaciones UDP. Tan sólo existen protocolos propietarios de compañías de software, o "parches" en los programas. Sin ser específicos para UDP, existen dos mecanismos que proporcionan un entorno seguro: son los sistemas hardware del nivel de enlace y el protocolo IPSEC [9] del nivel de red.

Más concretamente, una de las soluciones existentes son los dispositvos hardware, denominados *cifradores en linea*. Son unos dispositivos físicos que disponen de dos puertos, ambos de entrada/salida. Cada uno de ellos tiene una función especifica: uno cifra la información, y otro la descifra. De esta forma, cuando la información entra a través de un puerto, es modificada (cifrada o descifrada) y enviada al otro puerto.

Su funcionamiento es el siguiente: Cuando los mensajes salen del sistema, entran en el dispositivo, el cual los cifra y los envía a la red. Cuando los mensajes llegan a otro sistema, ya sea un terminal o una pasarela, deben atravesar el dispositivo, el cual los descifra. De esta forma, los mensajes viajan siempre cifrados a través de la red.

Esta solución no es adecuada para nuestros objetivos, ya que implica un alto coste pues es necesario tener un cifrador para cada equipo que acceda al exterior de la LAN. Además, la información se traslada siempre cifrada, impidiendo acceder a equipos externos a nuestra red, con lo que la flexibilidad es prácticamente nula.

Otra solución existente que se puede aplicar para UDP es IPSEC [9], extensión que se encarga de añadir seguridad a las tramas enviadas a través de IP. Combinando IPSEC con routers o cortafuegos, podemos conseguir todo lo necesario para que una RPV funcione con capacidad de autenticación, cifrado e integridad de los datos, y generación y gestión automática de claves.

IPSEC utiliza un mecanismo denominado *asociación de seguridad* (SA), por el cual los equipos deben acordar una serie de parámetros de seguridad para todas sus comunicaciones (claves, métodos criptográficos,...). Una vez acordados estos parámetros, IPSEC proporciona diversos mecanismos para añadir seguridad a IP: autentificación, creando una "huella digital" de los mensajes, y privacidad, mediante el cifrado de los paquetes. Esto se logra con dos cabeceras especiales que se añaden a IP: AH (Authentication Header) y ESP (Encapsulation Security Payload).

Sin embargo IPSEC tiene ciertas desventajas que le impiden ser viable para realizar una RPV flexible. Una de ellas es que no puede crear una asociacion de seguridad para procesos o usuarios concretos (debido a que se trabaja a nivel de red), cuando son esos procesos o usuarios los que queremos intercomunicar. Otra desventaja consiste en que el sistema aplica la protección de forma automática según las asociaciones de seguridad, con lo que no permite a los usuarios tomar la decisión sobre cuándo aplicar los mecanismos de seguridad. La consecuencia de lo anterior es que los usuarios no pueden acceder al exterior de la RPV, objetivo que deseamos alcanzar.

En el nivel de transporte no existe ninguna solución estándar para UDP. Para TCP sí existe (protocolos TLS [17] y SSL [16]), pero estos protocolos no pueden ser utilizados con servicios UDP. La solución que buscamos podría utilizarse como un complemento a estos protocolos, por lo que tendríamos completamente asegurada la capa de transporte (TCP y UDP)

Si se diseñara un mecanismo de seguridad para UDP a este nivel, se añadirían ventajas a IPSEC, como la identificación de distintos procesos o usuarios (característica del nivel de transporte), y aplicar el cifrado cuando fuera conveniente. Por supuesto, esta solución debería incorporar también mecanismos para lograr las ventajas de IPSEC (autentificación, manejo automático de claves...).

Respecto a las posibles soluciones en el nivel de aplicación, sería cada aplicación la que habría de proporcionar los mecanismos de seguridad. Esto sería sencillo de cara al usuario final, pero tendría ciertas desventajas pues dejaría de ser una solución homogénea y cada aplicación tendría sus propios mecanismos de cifrado y control de claves. Además, sería necesario controlar que cada uno de los usuarios actualizara sus aplicaciones con aquellas que proporcionan seguridad, hecho éste difícil de conseguir. Más aún, si fuera necesario modificar algún aspecto de la RPV deberíamos cambiar cada uno de los programas que forman parte de la RPV.

3.2 NUEVA SOLUCIÓN

El análisis de estas posibles soluciones, existentes o por diseñar, para ser utilizadas en dotar de seguridad al protocolo UDP, plantea la pregunta de cuál se ajusta más a nuestras necesidades.

El diseño que este trabajo presenta es una solución realizada en el nivel de transporte, totalmente software, que adopta las ventajas del IPSEC, sin arrastrar ninguna de sus desventajas. Este diseño permite a los usuarios decidir cuando acceder a la RPV, y comunicarse (de forma totalmente segura) tanto con miembros de la RPV como con miembros externos a ésta. Es, por lo tanto, muy flexible y fácil de mantener (existe un control centralizado), con bajo coste por basarse en software y sin necesidad de adquirir ningún equipamiento adicional. Este mismo diseño, aunque con ciertas diferencias dependientes del protocolo, fue utilizado para usar servicios TCP, con resultados óptimos [3].

El escenario genérico en que nos hemos basado para el desarrollo de la aplicación se representa en la figura 1. Existe una *entidad principal*, que centraliza el control de la empresa o entidad donde se va a implantar la RPV, y que puede representar la oficina principal de la misma. También hay varias *entidades secundarias*, cada una con su red de área local.

El diseño se divide en dos subniveles, denominados *Secsockets UDP* y *RPV-Insel UDP*. La interfaz Secsockets es el subnivel inferior. Este interfaz trabaja por encima del interfaz sockets tradicional que proporciona acceso a TCP-UDP/IP.

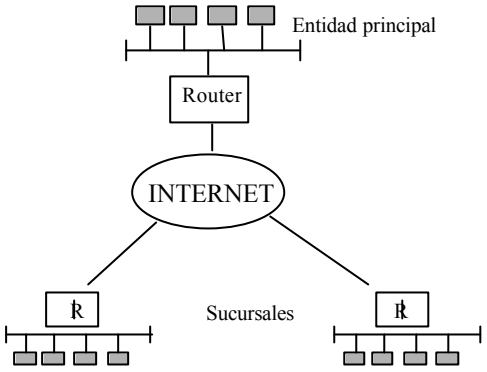


Figura 1. Escenario para la RPV

El servicio ofrecido por la interfaz Secsockets es la transmisión de datos de un proceso origen a un proceso destino a través de un canal seguro y autenticado. El módulo RPV-Insel trabaja sobre Secsockets y, haciendo uso de los servicios que este subnivel proporciona, da soporte de seguridad a las aplicaciones proveyéndolas de los servicios necesarios.

4 EL INTERFAZ SECSOCKETS UDP

4.1 INTRODUCCIÓN

El interfaz Secsockets es una extensión del interfaz socket tradicional, al cual le añade las funciones de seguridad que necesitamos para establecer nuestra RPV: autenticación (la información proviene de alguien que pertenece a la RPV), privacidad (quien no pertenezca a la RPV no puede acceder a los datos) e integridad (la información del mensaje no puede ser alterada).

El objetivo de esta interfaz es proporcionar un servicio de envío de datos seguro utilizando paquetes UDP, siendo fácil de utilizar de cara al usuario. Esta interfaz se basa en un modelo cliente-servidor, y su funcionamiento conlleva dos fases, una de conexión, en la que se ponen en marcha las bases para la comunicación segura, y otra de comunicación, donde los usuarios pueden ya enviar datos seguros de un extremo a otro mediante UDP.

4.2 FASE DE CONEXIÓN

La fase de conexión se realiza de forma automática, y en ella se produce la clave secreta necesaria para realizar una comunicación segura. Debido a que ambas partes (cliente y servidor) no disponen de esa clave hasta el final de la negociación, es necesaria la utilización de certificados digitales de clave pública para lograr la autenticación mutua y el intercambio seguro de los parámetros de seguridad.

Esta fase de conexión se realiza utilizando TCP, y no UDP. Esto es debido a que necesitamos un servicio de entrega fiable para el intercambio de los parámetros de comunicación. Al finalizar esta fase, se devolverá un socket UDP, a través del cual procederemos a realizar la comunicación UDP segura.

Los pasos que se siguen son los siguientes:

1. El cliente obtiene el certificado del servidor y le manda un primer mensaje con el algoritmo de clave pública a utilizar y su dirección electrónica.

2. El servidor recibe el paquete, solicita el certificado del cliente, y le envía a éste un mensaje de aceptación o rechazo. En este momento, tanto el cliente como el servidor poseen los certificados digitales del otro. De esta forma, ambos pueden enviarse mensajes cifrados utilizando el mecanismo de clave pública – clave privada.

3. El cliente envía cifrado con la clave pública del servidor una petición de conexión incluyendo los parámetros de seguridad. Estos parámetros son el algoritmo de cifrado simétrico a utilizar para cifrar la futura comunicación (DES [10], IDEA [11], Blowfish [12] o RC4 [13]), la función hash a utilizar para verificar si un mensaje es íntegro (MD5 [14] o SHA [15]), la compresión a utilizar (ninguna o GZIP) y un valor aleatorio que se usa para el cálculo posterior de la clave secreta.

4. Tras recibir y descifrar el mensaje de petición, el servidor envía al cliente un mensaje de aceptación o rechazo, cifrado con la clave pública del cliente. Además, si la negociación es aceptada, incluye otro valor aleatorio que también se usará para el calculo de la clave y el puerto UDP en el que el servidor escuchará las peticiones.

5. Ambos extremos calculan la clave secreta usando la función hash *H* y los valores intercambiados:

H(H(aleatorio_Cliente) Å H(aleatorio_Servidor))

Además, el servidor cierra la conexión TCP con el cliente y abre la conexión UDP esperando los mensajes del éste.

Cabe reseñar que el algoritmo de cifrado simétrico más aconsejable para Secsockets UDP es el RC4. La causa es que este algoritmo es unas 10 veces más rápido que algoritmos como el DES, y esto es algo vital teniendo en cuenta que pretendemos trabajar con servicios que necesitan de recepción de datos en tiempo real.

4.3 FASE DE COMUNICACIÓN

Los pasos que se realizan durante la fase de comunicación (o de transmisión de datos) son, para cada mensaje: 1) compresión de la tramas, si así se

ha decidido durante la negociación inicial; 2) cálculo del valor hash de cada trama; 3) firma y cifrado de los datos; 4) envío de los datos.

La figura 2 muestra la composición de cada una de las tramas UDP. El tamaño máximo del mensaje depende de la longitud de los paquetes que acepte la red, por lo que es un parámetro configurable. Tras la recepción del mensaje, el receptor debe deshacer las operaciones que se han realizado sobre la trama recibida: descifrar (utilizando la recién conseguida clave simétrica), calcular y comparar su valor hash (para ver si ha sido modificado), descomprimirse (si así fue indicado en la negociación) y finalmente pasar los datos al nivel superior.

4.4 FUNCIONES DE LA INTERFAZ SECSOCKETS UDP

Ya se ha visto anteriormente que el servicio que ofrece la interfaz Secsockets es enviar datos de un proceso origen a un proceso destino a través de un canal seguro y autenticado. Este servicio se ofrece a través de un conjunto de funciones.

Tales funciones se comportan, en su interfaz, como un protocolo orientado a la conexión siguiendo el paradigma cliente/servidor. Esto es así debido a que las aplicaciones seguras que utilizan UDP necesitan un interfaz que les apoye a crear sistemas cliente/servidor paralelos. Además, de esta forma se puede elegir si utilizar para la comunicación TCP o UDP utilizando un simple byte de parámetro.

Las funciones de inicialización del interfaz Secsockets son las siguientes:

- *sec_init( )*: Crea un punto final de conexión en el extremo del servidor. Lo inicializa asignándole un puerto local de comunicación, y deja al servidor en espera de posibles peticiones de conexión de clientes a través de ese puerto.

- *sec_accept ( )*: El servidor llama a esta función para reconocer una conexión entrante, tras la cual empieza una negociación bajo TCP. Luego se devuelve un socket UDP con su dirección correspondiente, además de los parametros de seguridad.

- *sec_connect( )*: El cliente llama a esta función para crear un punto final de comunicación para el cliente. Tras una negociación con TCP (autentificando e intercambiando los parámetros de seguridad), se devuelve un socket UDP y su dirección, además de los parámetros de seguridad.

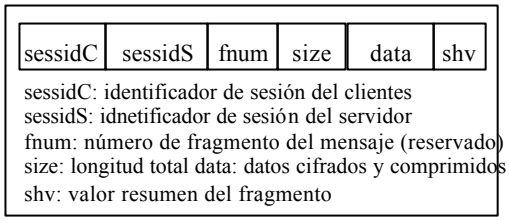


Figura 2. Formato tramas UDP

Con posterioridad a estas llamadas, y utilizando los sockets devueltos y los parámetros de seguridad, es posible establecer una comunicación segura. El intercambio de información y el cierre del canal se realizaría con las siguientes funciones:

- *sec_recv( )*: Esta función permite la recepción de datos a través de una conexión socket segura UDP, los descifra, chequea su autenticidad y los descomprime si es necesario.

- *sec_send( )*: Esta función es simétrica a la anterior. Comprime los datos si así se negoció previamente, calcula el valor hash, cifra los datos y, finalmente, los envía a través del socket UDP. Estas operaciones se realizan de acuerdo a la negociación de parámetros inicial entre cliente y servidor.

- *sec_close( )*: Esta función cierra el socket seguro UDP.

5 RPV-INSEL UDP

5.1 ESQUEMA RPV-INSEL

El módulo RPV-Insel utiliza los servicios proporcionados por la interfaz Secsockets para proporcionar seguridad a las aplicaciones UDP de la RPV. Además, se encarga de gestionarla, ya que inicializa todo lo que ésta necesita de forma automática, gestiona a los usuarios que forman parte de ella, y permite la instalación de servicios UDP exclusivos para sus miembros.

La RPV se basa en la existencia de 4 tipos de procesos: el *servidor principal* (S1), los *servidores secundarios* (S2), los *servidores auxiliares* (SAux), y los *clientes*, como se puede observar en la fig. 3.

Los clientes (C1). Utilizan los servicios que nos ofrece la RPV. Estos equipos forman parte de una LAN (C1).

El servidor principal de la RPV se encarga de: controlar a los demás servidores, mantener una base de datos con la información relativa a éstos (p. ej. sus direcciones IP), y servir información referente a la RPV.

Los servidores secundarios de la RPV. Existe un S2 por cada LAN de la RPV, y se encarga de controlar las peticiones de comunicación de los equipos de la LAN, además de controlar qué equipos de la LAN están conectados a la RPV. Cada S2 posee un

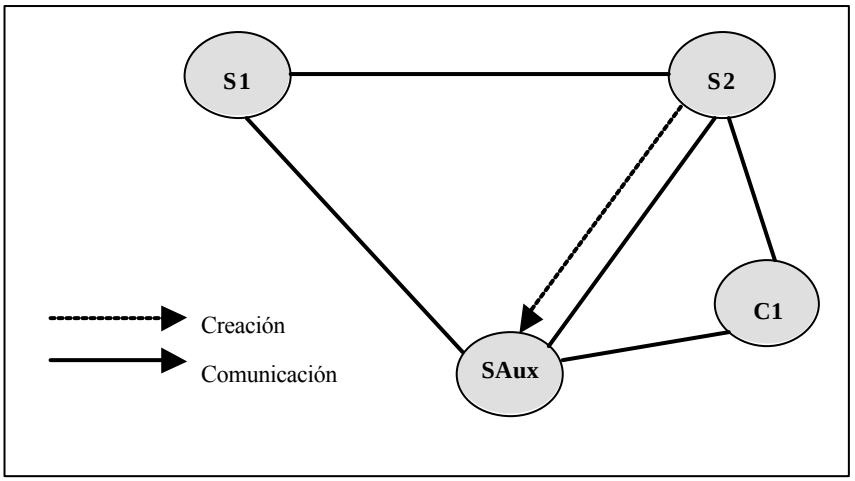


Figura 3. Esquema RPV-Insel UDP

certificado que se utilizará para la negociación de los canales seguros con Secsockets.

Los servidores auxiliares de la RPV son creados por los servidores secundarios, y su función consiste en llevar a cabo las comunicaciones seguras que han sido pedidas por los usuarios de la RPV. Existe un Saux por cada petición de conexión de un cliente.

De esta forma, cuando un cliente desea comunicarse con otro equipo de la RPV, contacta con su servidor secundario, el cual crea un servidor auxiliar que servirá de "puente" entre el cliente y el otro equipo de la RPV. Así, todo el tráfico cifrado pasará a través de los servidores auxiliares cifrado, mientras el cliente cree que el tráfico está siendo enviado como siempre.

5.2 PROBLEMAS DE DISEÑO AL UTILIZAR UDP

Durante la realización de este diseño y su puesta en práctica bajo UDP, han surgido diversos problemas, todos debidos a las características intrínsecas del protocolo UDP. A continuación se describen los problemas y las soluciones que se han adoptado.

Como las aplicaciones UDP basadas en transmisión de audio y video se basan en una arquitectura cliente-cliente (nadie es un servidor predefinido), existe el problema de controlar qué servidor secundario se encarga de actuar como servidor de una petición de comunicación. Por usar un símil, podríamos pensar que siempre hay alguien que llama por teléfono y hay alguien que recibe la llamada y responde, pero cualquiera de los dos puede manejar el rol de emisor/receptor de la llamada.

Para solucionar este problema, hay que pensar que siempre existe un cliente que realiza primero la comunicación, y un servidor que la recibe. Por lo tanto, el esquema es el siguiente: Ambos servidores secundarios están preparados para actuar como servidores (esperan una llamada), pero en el momento que uno mande una petición, se convierte en cliente, y convierte al otro en servidor.

Otro problema consiste en que se debe realizar una arquitectura cliente/servidor paralela (los servidores secundarios crean los servidores auxiliares), y el protocolo UDP no está preparado de antemano para lograr esta tarea, ya que está orientado a una arquitectura petición/respuesta.

La solución consiste en “simular” el protocolo TCP, en el aspecto de proporcionar un nuevo canal de comunicación cliente/servidor. De esta forma, se reserva un puerto temporal UDP en la máquina servidor, se comunica ese puerto al cliente, y se hace que toda la comunicación que el cliente vaya a enviarnos a partir de ese momento fluya por ese puerto. Una vez se deje de utilizar, ese puerto se cierra para permitir su reutilización.

6 ESCENARIO DE PRUEBA

La base de todo el sistema RPV-Insel es que las aplicaciones UDP conectan con los servidores auxiliares (en vez de con el host destino) y a través de ellos envían/reciben las tramas. Estos servidores se encargan de “puentearse” las tramas entre sí utilizando canales de comunicación seguros, de tal forma que la información viaja por la red sin ninguna posibilidad de ser alterada.

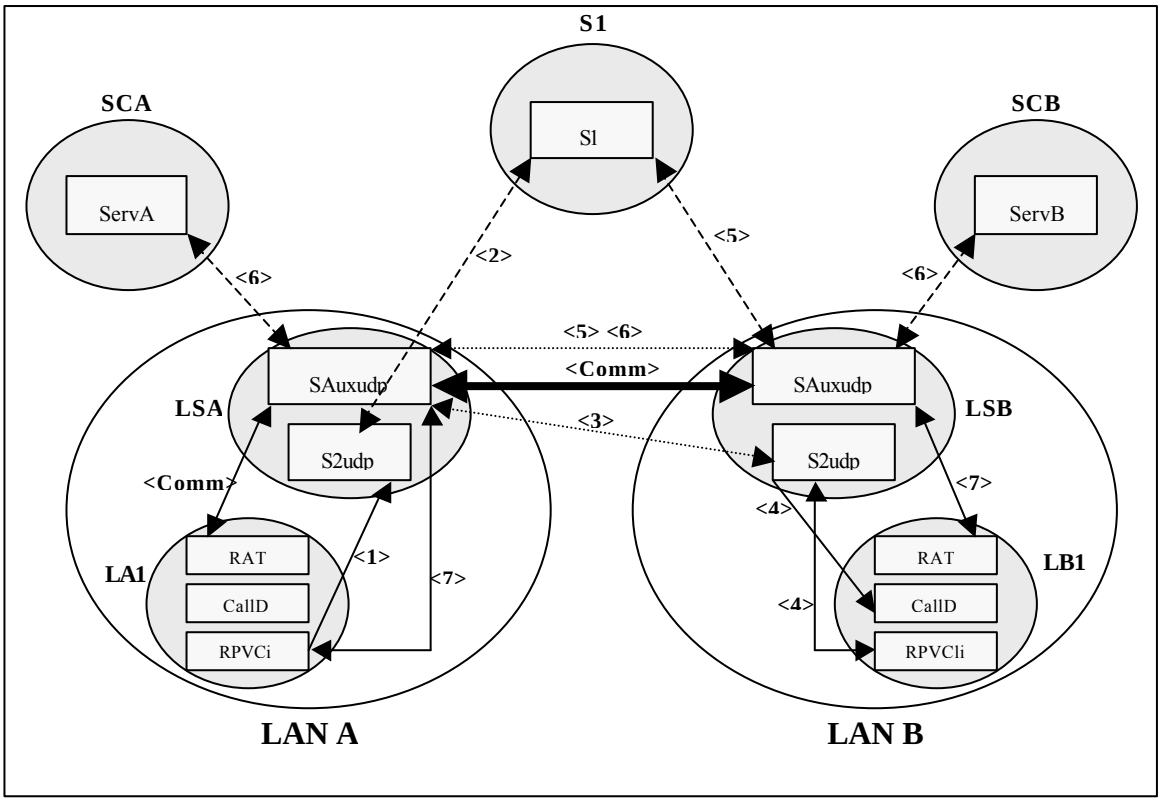


Figura 4: Escenario de prueba

Para la explicación de este sistema de vamos a proceder a la exposición de un escenario (Figura 4), el cual fue testeado bajo máquinas Linux. En él tenemos dos máquinas (LA1 y LB1) que quieren comunicarse entre sí usando el programa RAT (audioconferencia bajo RTP [18]), además de dos servidores secundarios (LSA y LSB, uno para cada LAN), dos máquinas servidoras de certificados (SCA y SCB, una para cada LAN), y un servidor principal (S1).

Al iniciarse la comunicación por petición del usuario, LA1 avisa a su servidor secundario, LSA, que quiere empezar una comunicación con LB1. A partir de ese momento la siguiente secuencia de eventos toma lugar:

1. LSA comprueba que la petición es para realizar una llamada y procede de un usuario de la RPV, por lo que pasa a modo servidor.

2. El servidor secundario envía una petición a S1 para que le diga en que LAN se encuentra LB1 y cual es el S2 de esa LAN. S1 responde a la comunicación (devuelve la dirección IP de LSB).

3. LSA crea un servidor auxiliar, que se encargará de ahora en delante de todo el proceso. Este servidor envía una petición a LSB (mediante un canal TCP), y espera su respuesta.

4. LSB recibe la petición, guarda el puerto del canal TCP, y avisa a LB1 que esta siendo llamado. A continuación LB1 responde a LSB, y LSB (tras comprobar que LB1 pertenece a la RPV) pasa a modo cliente.

5º) LSB crea un servidor auxiliar para que se encargue de la comunicación. Este, tras localizar a LSB vía S1, responde al auxiliar de LSA mediante el puerto TCP anteriormente guardado, comenzando la negociación.

En este momento LSA y LSB saben la dirección del otro, y mantienen una comunicación abierta.

6º) LSA y LSB realizan la negociación de dos canales seguros UDP (RTP necesita de dos canales UDP para funcionar). Es en este momento cuando se accede a las máquinas SCA y SCB.

7º) LSA y LSB comunican a sus respectivos clientes que la comunicación segura puede empezar.

Tras recibir este aviso, los clientes llaman al programa RAT. Estos se comunicarán con los respectivos servidores auxiliares, ya que se se habrá indicado a éstos como destino de las comunicaciones. Se encargarán de enviar los datos a través de los sockets seguros, y de esta forma la información viajará por el exterior de la LAN de forma totalmente segura.

Una vez que las comunicaciones se cortan, se liberan los sockets y los recursos que estaban funcionando, y los servidores secundarios terminan.

7 CONCLUSIONES

El presente trabajo propone una nueva solución para implementar una RPV que cubre las necesidades de seguridad que implican las comunicaciones internas de una organización distribuida. La propuesta que realizamos se basa en dos subniveles: la interfaz Secsockets y RPV-Insel, que se localizan sobre el nivel de transporte de la torre de protocolos TCP/IP y por debajo de las aplicaciones.

Esta solución presenta algunos inconvenientes respecto a las soluciones existentes. Un inconveniente es que no ofrece servicio de no repudio de origen, debido a la sobrecarga que sufriría el sistema. Otro inconveniente consiste en que, al permitir que la comunicación entre distintas LANs este cifrada o no lo esté, estamos creando un posible punto de ataque al sistema (es el precio de la flexibilidad, obliga al administrador a tomar un mayor cuidado al establecer la topología de nuestro sistema).

No obstante, las ventajas merecen ser tenidas en cuenta. La primera ventaja que ofrece este diseño es que es una solución exclusivamente software por lo que su instalación no conlleva la modificación del hardware existente ni la adquisición de dispositivos nuevos, y además, es portable.

La segunda ventaja es la alta seguridad (ya que las comunicaciones internas de una organización se autentican y aseguran), con lo que se consigue la misma funcionalidad que en una red realmente privada. Además, se logra una gran flexibilidad ya que se sigue permitiendo el acceso genérico a cualquier punto de Internet.

Por último, se mantiene el uso de los mismos servicios, por lo que no hay que modificar el software UDP existente, siendo compatible con cualquier software que cumpla los estándares correspondientes.

De esta forma, obtenemos una RPV segura y flexible que, aunque nos exiga un poco más de control al crear nuestra red, nos permite ciertas funcionalidades muy necesarias en estos tiempos (acceso al exterior, flexibilidad,...)

Referencias

[1] U. Black, “Frame-Relay: Specifications and Implementations”. McGraw-Hill, 1994

[2] R. Harbison, “Frame-Relay: Technology for our Time”. LAN Technology, Diciembre 1992

[3] J. Dávila, J. López, R Peralta, “Implementation of Virtual Private Networks at the Transport Layer”, Information Security Workshop, LNCS 1729, Springer 1999.

[4] J. Postel, K. Harrenstien, “Time Protocol”. RFC 868, May 1983.

[5] D. Mills, “Simple Network Time Protocol Version 4 for Ipv4, Ipv6 and OSI”, RFC 2030, Oct. 1996.

[6] SNMP Working Group, SNMP Documents, RFCs 1905, 1906, 1907, 2576, 2578, 2579, 2580.

[7] H. Schulzrinne, S. L. Casner, R. Frederick, V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications”, RFC 1889, Jan. 1996.

[8] H. Schulzrinne, A. Rao, R. Lanphier, “Real Time Streaming Protocol (RTSP)”, RFC 2326, April 1998.

[9] R. Atkinson, “Security Architecture for the Internet Protocol”. RFC 2401, Nov. 1998.

[10] National Bureau of Standards, “Data Encryption Standard”. U.S. Department of Commerce, FIPS pub. 46, Jan. 1977.

[11] X. Lai, J. Massey, “Hash Functions Based on Block Ciphers”. Advances in Cryptology – EUROCRYPT '92, Springer-Verlag, 1992.

[12] B. Schneier, “Description of a New Variable-Lenght Key, 64-Bit Block Cipher (Blowfish)”, Fast Security Workshop Proceedings, Springer, 1994.

[13] R. Rivest, “The RC4 Encryption Algorithm”, RSA Data Security, Mar 1992.

[14] R. Rivest, “The MD5 Message Digest Algorithm”. RFC 1321, April 1992.

[15] National Institute of Standards and Technology, NIST FIPS PUB 180. “Secure Hash Standard”. U.S. Department of Commerce, May 1993.

[16] Netscape Communications, “SSL 3.0 Specification”. *http://home.netscape.com/eng/ssl3/*

[17] T. Dierks, C. Allern, “The TLS Protocol version 1.0”. Internet Draft, November 1998.

[18] Varios, "RAT: Robust Audio Tool" *http://www-ice.cs.ucl.ac.uk/multimedia/software/rat*

Autenticación en la Red: ACerO y JCCM.* Java Card Certificate Management

Ignacio Díaz Asenjo, Arturo García Ares, María Celeste Campo Vázquez, Andrés Marín López, Carlos Delgado Kloos, Carlos García Rubio, Peter T. Breuer

Universidad Carlos III de Madrid.
Avd. Universidad 30 28911 Leganés

e-mail:eticket@it.uc3m.es

Abstract

Public-key cryptosystem are the best option to reinforce systems security, because they solve the interchange of keys problem. Certification Autorities and digital certificate management use these cryptosystems and offer a trust model to build applications on top of. This is the last step to provide secure systems, integrating these concepts in the procedures of the organizations. This integration brings with itself technical and legal problems, which are broached jointly in public keys infrastructures.

In this article we move at the level of Certification Authorities and certificate management. We want to show the two more important results of our work in this area: ACerO, in the server side, and JCCM, in the client side. Both developments will be soon available in the Web. Our motivation is to offer a Web based Certification Authority related to the use of smartcards javacards for certificate storage. All has been developed for Linux and Netscape. In the server side we emphasized the use of servlets and libraries of native code; in the client side, JCCM offers a flexible certificate management and independence with respect to the smartcard manufacturer. JCCM is available for their use with Netscape for secure browsing, ciphering and signing of e-mails.

1 Introducción

Desde que Alicia hizo pública su clave, enviarle información privada o comprobar su autoría se facilitaron enormemente. El problema de la distribución de claves se facilitó enormemente gracias a los algoritmos de cifrado asimétricos. Las iniciativas de desarrollo de software criptográfico basado en este nuevo paradigma, como openssl (`http://openssl.org`), ponen a disposición de los usuarios finales estas nuevas capacidades.

El siguiente paso en la cadena es la construcción de autoridades de certificación y las facilidades de gestión de certificados digitales. También hay software para este fin, de hecho el propio openssl lo hace. El problema es que para que sea realmente utilizable es necesario portarlo a su uso en el web, y para esto hemos desarrollado ACerO. Similar a otras iniciativas como la de la Universidad de Murcia (con el proyecto UMGina), o la OpenCA. En la sección 3 exponemos las ventajas de nuestro enfoque con la OpenCA.

La segunda parte de este artículo se sitúa en la parte de cliente. El uso de aplicaciones criptográficas empieza a popularizarse a partir de la inclusión de capacidades criptográficas en los navegadores. Hoy en día la mayoría de los usuarios del web han accedido alguna vez a páginas *seguras* (utilizando el protocolo [HTTPS]), y poco a poco se empiezan a enviar correos electrónicos firmados o cifrados. La piedra angular de estas nuevas capacidades radica en la utilización de certificados digitales ([X.509]) que establecen un mecanismo para comprobar la identidad de un usuario o de un sitio, para dar fe de que un usuario ha firmado un mensaje y garantizar la integridad del mismo, o bien para proteger una clave de sesión mediante la cual cifrar un mensaje. Estos certificados encapsulan distintas claves, entre ellas la más importante es la clave pública. El punto más débil del sistema está en el almacenamiento de claves (en concreto de la clave privada). Las tarjetas inteligentes son una alternativa muy segura para el almacenamiento y gestión de dichas claves. Las tarjetas actuales son dispositivos inviolables y además tienen una considerable capacidad de cómputo y almacenamiento.

Las aplicaciones que utilizamos, por ejemplo nuestro navegador favorito, incluye las capacidades necesarias para gestionar estos certificados digitales. Por ejemplo, el navegador de Netscape, tiene su propia biblioteca, la *Netscape Security Library* (NSL) para estos fines.

*Este trabajo ha sido desarrollado dentro del proyecto E-TICKET CYCYT Nº2FD1997-1269-C02-01(TEL)

El estándar [PKCS#11] define un interfaz que proporciona a las aplicaciones que desean utilizar servicios criptográficos una visión lógica común de los dispositivos capaces de ofrecer dichos servicios: dispositivos con capacidad de almacenamiento, proceso, y específicamente, capaces de proteger de manera efectiva los elementos sensibles (claves de cifrado) almacenados. Las aplicaciones que utilizamos suelen utilizar este API (por ejemplo, la NSL de Netscape sigue este estándar).

La alternativa más segura es utilizar módulos que implementen PKCS #11 para las aplicaciones que lo admitan, y utilizar tarjetas inteligentes para compartir la gestión y para almacenar de forma segura claves y certificados. Sin embargo, los desarrollos que se han realizado hasta la actualidad de módulos PKCS #11 ([SMARTSIGN], [GPKPKCS#11], [SLBCBPKCS#11]) con tarjetas inteligentes emplean tarjetas no programables o bien no utilizan la facilidad de programación de las tarjetas que sí son programables. Estas implementaciones se limitan a adaptar la funcionalidad criptográfica predefinida en las tarjetas a la visión lógica que debe ofrecer un módulo PKCS #11, recortando la semántica definida por el estándar para adaptarla a la que ofrece el fabricante en cada tarjeta.

Nuestro sistema, Java Card Certificate Management (JCCM), emplea una tarjeta inteligente programable mediante tecnología Java Card como dispositivo criptográfico, trasladando parte de la implementación del estándar PKCS #11 a la tarjeta, resultando en una mayor flexibilidad, e independencia del dispositivo.

Firmar mensajes de correo electrónico mediante algoritmos de clave pública requiere un complejo sistema en el que participan (1) una autoridad de certificación y mecanismos de gestión y distribución de claves en la organización, que en nuestro caso será ACerO, (2) agentes de usuario habilitados para generar y verificar firmas digitales (Netscape), y (3) dispositivos capaces de ejecutar algoritmos criptográficos y de almacenar datos sensibles de manera segura que llamaremos *tokens criptográficos* o simplemente *tokens*. JCCM abarca las partes (2) y (3), dos piezas complementarias que se materializan en:

- Una biblioteca de enlace dinámico (DLL) que sirve de interfaz entre la tarjeta inteligente y la biblioteca de seguridad (NSL) de Netscape –el agente de usuario.
- Un *cardlet*[1] que implementa las funciones requeridas.

La estructura de este artículo es la siguiente: la sección 2 explica el estándar PKCS #11 y la arquitectura de seguridad de Netscape, punto integrador del sistema. La sección 3 describe la parte de servidor de nuestro sistema: ACerO, las tecnologías que utiliza, estándares soportados, arquitectura y funcionamiento. La sección 4 describe la parte de cliente: JCCM, Java Card la arquitectura y una comparativa del rendimiento de las dos tarjetas en que hemos implementado JCCM.

2 Netscape y PKCS #11

PKCS #11 es uno más del creciente grupo de estándares PKCS, acrónimo de Public-Key Cryptography Standards (estándares de criptografía de clave pública) y publicados por RSA Labs. Estos estándares abarcan aspectos del sistema que acabamos de esbozar tales como:

- Definición de los algoritmos criptográficos (tanto de clave pública como de clave simétrica) y algoritmos de apoyo.
- Procedimientos para aplicar los algoritmos criptográficos a flujos de bytes y para producir firmas digitales.
- Definición de tipos de datos (certificados, claves, ...) y sus correspondientes sintaxis de transferencia o formatos de representación binaria.
- Procedimiento para efectuar transacciones HTTP seguras
- Definición de formatos para transmisión de correo electrónico cifrado e inclusión de firma digital.
- Interfaz de programación para aplicaciones (API) que permite hacer uso de los algoritmos, procedimientos y tipos de datos.

La biblioteca descrita por el estándar PKCS #11 recibe el nombre de Cryptoki, abreviatura de "cryptographic token interface". Cryptoki aisla a la aplicación de los detalles del dispositivo criptográfico. En terminología de Cryptoki, estos dispositivos se llaman *tokens*. Estos tokens pueden ser de carácter portátil (p.e. una tarjeta inteligente) o fijos (p.e. una tarjeta PCI insertada en la placa madre del ordenador). Los tokens portátiles se insertan en su correspondiente *slot*, nombre asignado por Cryptoki para la abstracción del terminal lector. Un token almacena *objetos* y es capaz de ejecutar *primitivas criptográficas* empleando esos objetos, posiblemente tras un paso inicial de autenticación del usuario/aplicación al token mediante la presentación de un PIN. Toda comunicación con el token se efectúa dentro del contexto de una *sesión*, que representa un canal de comunicación establecido con un token presente en un slot.

Netscape incorpora una arquitectura de seguridad que permite tráfico HTTP seguro y gestionar correo cifrado y firmado. Esta arquitectura de seguridad, conocida como "Netscape Security Library" (NSL), hace uso de las primitivas presentes

[1] Programa residente en la tarjeta inteligente (o token) escrito acorde con la especificación Java Card

en PKCS #11 para ofrecer la funcionalidad de alto nivel a la que nos referimos. Netscape se distribuye con un módulo PKCS #11 interno que constituye una implementación bastante completa del estándar: incluye tanto los mecanismos basados en RSA como algunos mecanismos de clave simétrica. La versión y la distribución (con o sin restricciones de exportación) de Netscape imponen los mecanismos que podemos utilizar y las limitaciones de los mismos en cuanto a longitud máxima de claves. Este módulo PKCS #11 interno ofrece una visión lógica del propio ordenador como dispositivo criptográfico: utiliza el sistema de archivos como almacenamiento persistente para almacenar los objetos de Cryptoki y la capacidad de proceso de la CPU para efectuar las operaciones de cifrado. Gracias a este módulo interno Netscape es plenamente capaz de ofrecer las capacidades mencionadas, pero adolece de un problema: la utilización del sistema de archivos del ordenador para almacenar certificados y claves rompe la premisa inicial de almacenamiento seguro de datos sensibles. La solución adoptada por Netscape es permitir la incorporación de módulos PKCS #11 externos que sirvan de interfaz a hardware criptográfico especializado, como es el caso de nuestro módulo JCCM.

Tanto las claves como los certificados generados por ACerO serán recogidos en última instancia por la NSL de Netscape dando la posibilidad de esta forma de almacenarse según elija el usuario o bien en el sistema de archivos tradicional o bien en una tarjeta inteligente gracias a la presencia dee este nuevo módulo.

3 ACerO

A lo largo del artículo se hace referencia a una serie de objetos personales e intrasferibles como son claves privadas y certificados, que deseamos almacenar dentro de tarjetas inteligentes. De esta forma conseguimos un medio cómodo de transportar nuestra identidad digital y en cierta forma un entorno protegido frente a intrusos que quieran apoderarse de esta información para hacerse pasar por nosotros.

Lo que se persigue en todo momento es una forma de disponer de una especie de "DNI digital"que nos permita identificarnos ante un determinado sistema preparado para tales fines.

Resulta evidente pues, la necesidad de disponer sistema capaz de generar y administrar esos certificados digitales, y que sea además un entidad en la que podamos confiar, tal y como confiamos en el Ministerio de Interior como emisor de nuestros DNIs tradicionales, para tener la certeza de que ese certificado corresponde realmente a la persona adecuada y no a un intruso.

OpenCA[2] de Massimiliano Pala se puede considerar como uno de los primeros y más conocidos esfuerzos de llevar a cabo la implementación de una infraestructura de clave pública (PKI) abierta y de libre distribución. En gran medida ha sido fuente de inspiración de ACerO, es por ello que la filosofía principal en la que se basa este producto es muy parecida: Interfaz Web y OpenSSL para realizar las funciones criptográficas

OpenCA está realizado en su totalidad por medio de scripts y CGIs en Perl, y utiliza OpenSSL realizando llamadas directamente a un ejecutable *openssl* resultante de la instalación de dicho paquete. Es una implementación pensada para un tráfico de peticiones muy bajo debido a que este tipo de diseño carga demasiado el servidor Web donde reside. Por cada una de las operaciones en el servidor se abrirán dos procesos, uno propio del CGI y otro como resultado de llamar a un ejecutable, *openssl*, de nuestro sistema desde un CGI.

ACerO mejora en gran medida esta enorme carga utilizando soluciones tecnológicamente más avanzadas. Los servlets de Java sustituyen a los CGIs en Perl, y las llamadas al ejecutable *openssl* por invocaciones a métodos nativos. Este cambio desde la base es uno de los principales motivos por los que se prefirió comenzar un nuevo diseño en lugar de contribuir a mejorar y ajustar a nuestras necesidades la OpenCA.

Otras ventajas que ofrece ACerO frente a OpenCA es su mayor escalabilidad. OpenCA da la sensación de estar pensado únicamente para contemplar una única RA ya que no realiza una repartición de los certificados dependiendo de las RAs de las que proviene la petición, sino que envía todos los certificados a todas. ACerO suple esta deficiencia enviando a cada una de las RAs los certificados que ellas solicitaron.

ACerO es una implementación de libre distribución de un Sistema de Gestión de Certificados cuyos pilares han sido concebidos en un principio para funcionar sobre plataforma Linux, pero utiliza tecnología que facilita su posterior portado a otras arquitecturas como puede ser Windows. Esto significa que en la actualidad la parte correspondiente a la Autoridad de Certificación debe instalarse en dicha plataforma.

Su diseño totalmente orientado hacia el navegador Netscape desemboca en una utilización de cierta funcionalidad proporcionada por su módulo interno y la total compatibilidad con el módulo PKCS#11 descrito con anterioridad.

Se trata de una serie de aplicaciones Web que se apoyan en servlets de Java para proporcionar la funcionalidad que el sistema ofrece a través del navegador. Para realizar los operaciones criptográficas que se llevan a cabo se utilizan llamadas a métodos nativos que se basan en el código de implementación de OpenSSL.

3.1 Estándares de PKI admitidos

La cuestión principal de tener un PKI es poder generar certificados y llevarlos donde sea necesario.

[2] `http://www.openca.org`

Existen estándares de seguridad abiertos para los componentes de un PKI que han sido desarrollados con la finalidad de promover la interoperabilidad entre sistemas. Entre las organizaciones dedicadas a esta tarea se encuentran el W3C (World-Wide Consortium), IEFT (Internet Engineering Task Force) e ITU (International Telecommunication Union).

ACerO se ajusta a los siguientes estándares:

- X.509 versión 3 encargado de definir el formato y el contenido de los certificados digitales
- CRL versión 1 encargado de establecer el formato y el contenido de las listas de revocación de certificados
- Familia PKCS encargado de definir el formato y el comportamiento del intercambio y distribución de claves públicas.

De manera que como resultado obtenemos un producto compatible con múltiples herramientas existentes en la actualidad.

3.2 Arquitectura Interna

Internamente el sistema se apoya en una serie de componentes bastante conocidos y que podemos obtener gratuitamente de la red: Apache 1.3.x, Tomcat 3.2, Netscape 4.x o sup, Java2, y OpenSSL 0.9.6.

ACerO utiliza *servlets* de Java, proporcionando una sustitución potente y eficiente de los CGI (*Common Gateway Interface*). Sus ventajas son:

- La **portabilidad** debido a que los servlets de Java son independientes de plataformas y protocolos.
- El **rendimiento** ya que tienen un ciclo de vida más eficiente que los programas CGI o FastCGI. A diferencia de los programas CGI, los servlets no crean un nuevo proceso en cada petición entrante. En su lugar, utilizan diferentes hilos de ejecución (*threads*) en el servidor.
- La **seguridad** porque al ser una derivación del lenguaje Java, heredan todos sus mecanismos de seguridad además de algunas características propias. La *sand-box* del servlet proporciona un entorno controlado en el que el servlet puede funcionar y utiliza un controlador de seguridad para vigilar la actividad del servlet y prevenir operaciones no autorizadas.

Por medio Netscape además de la funcionalidad intrínseca de navegación e interfaz de las aplicaciones se proporcionará la generación de claves y firma de mensajes que pasamos a la CA.

Se han desarrollado una serie de librerías en C, actualmente dependientes de arquitecturas tipo *Linux ix86 (.so),* en las que se implementan con la ayuda de OpenSSL operaciones de: generación de un par de claves RSA para la CA, autofirmado de la petición de certificado de la CA, comprobación e interactuación con el fichero de configuración de la CA, verificación de un firma PKCS#7, firma de una solicitud de certificado, revocación de un certificado, generación de la lista de revocación, exportación de certificados en formato PKCS#12, o mostrar información de un certificado X509.

3.3 Arquitectura Lógica

La arquitectura del sistema de certificación se divide en tres partes, y conceptualmente está diseñada para operar en varias máquinas distintas.

- Usuario (*Operaciones de Usuario)* : En esta parte del sistema se sitúa la Persona o Servicio que solicita y recoge un certificado.
- RA (*Autoridad de Registro*) : La idea de la RA responde a una estrategia para aislar al usuario del núcleo del sistema. La CA delega en la RA las labores de comunicación con el usuario, recopilación de sus datos e identificación del mismo. En cada RA existirá la figura del Operador de la RA. Su misión será la de notario digital, comprobando que los datos que aparecen en la petición son correctos y corresponden realmente con el ente que realizó dicha petición.
- CA (*Autoridad de Certificación*) : La CA es el corazón del sistema, y como tal debe aislarse lo máximo posible ya que si su integridad fuese violada desmoronaría todo el entramado de certificados. En esta parte se situará la figura del Operador de la CA o persona que supervisa la información que la RA validó y firma digitalmente los certificados confirmando de esta forma la relación existente entre la clave pública contenida y la identidad del propietario.

3.4 Funcionamiento

De una manera muy general podremos resumir un escenario típico de funcionamiento de la siguiente manera:

1. Un usuario desea obtener un certificado de nuestra Autoridad de Certificación. Primero deberá enviar sus datos y su clave pública a una Autoridad de Registro dependiente de la Autoridad de Certificación. El par de claves en este caso es generado por el navegador, siendo ésta la forma más segura de realizar esta operación, ya que la clave privada se guarda en todo momento.
2. Los datos del solicitante y su clave pública se añaden a la lista de espera de certificados pendientes de la RA, a la espera de ser aprobados por un *operador.*

3. En la RA únicamente puede operar un usuario de confianza de la CA. Él se encargará de comprobar que los datos recibidos en la petición corresponden a quien dicen ser, y firmará esa información para enviársela a la CA. La firma se realiza por medio de un certificado del operador disponible en el navegador o en su tarjeta inteligente, utilizando el módulo criptográfico del Netscape para de esta forma obtener una salida firmada en formato PKCS#7

4. Cuando la RA desee mandar los datos a la CA, se realizará un envío de información por medio de un paso de exportación de la RA y otro de importación por parte de la CA.

5. Al recibir la CA una serie de solicitudes de certificados aprobados por una o varios RAs, deberá comprobar por cada una de ellas la validez de la firma plasmada por el operador de la RA correspondiente. En el caso de que la verificación sea correcta, la CA firmará la solicitud del certificado devolviendo un certificado con un formato basado en el estándar X.509.

6. La CA enviará a la RA, por medio de un proceso de importación y exportación de datos, el/los certificado/s aprobado/s (*emitidos*), y se notificará al usuario correspondiente que su certificado ya está listo para ser recogido.

7. Una vez que el usuario ha recibido la notificación podrá obtener su certificado personal, puede por medio del número de serie indicado en la notificación, recoger su certificado aprobado por la CA. El certificado se incluirá en el dispositivo donde se creó la correspondiente clave privada, o bien en la base de datos del Netscape (por defecto) o bien en la tarjeta inteligente en el caso de utilizar JCCM.

4 JCCM: Tarjetas inteligentes y Java Card

Una tarjeta inteligente contiene un pequeño ordenador que consta de un microprocesador, memoria volátil (RAM) y memoria persistente (EEPROM). Las capacidades típicas de estos elementos son: microprocesador de 8 bits a 4MHz, RAM de 512 bytes, 32Kb de EEPROM y 16Kb de ROM. Los contactos presentes en la superficie de la tarjeta aportan la tensión de alimentación, la señal de reloj y un canal de comunicación serie (a 9600bps[3]). Estos contactos son gobernados por el terminal lector en el que esté insertada la tarjeta, por lo tanto el microprocesador sólo estará activo cuando la tarjeta esté insertada en un terminal. Extraer la tarjeta del terminal tiene el mismo efecto que apagar un ordenador: se interrumpe la actividad de la CPU y se pierde el contenido de la RAM, pero el almacenamiento persistente permanece; en un ordenador este almacenamiento persistente está constituido por dispositivos magnéticos (discos duros), mientras que en una tarjeta inteligente es la memoria EEPROM. Otra característica importante de este tipo de memoria es su relativamente corta esperanza de vida, unos 100000 ciclos de escritura, y su elevado tiempo de acceso para ciclos de escritura, del orden de 10ms.

La tarjeta inteligente es un dispositivo pasivo: recibe un comando proveniente del ordenador a través del terminal en el que esté insertada, lo procesa, genera una respuesta que es devuelta al ordenador y espera el siguiente comando. Estos comandos se denominan APDUs (Application Protocol Data Unit) y su estructura está definida en [ISO/IEC 7816-4]. Existen dos tipos de APDU, las que codifican un comando enviado por el ordenador y las que codifican la respuesta generada por la tarjeta.

Existen tarjetas no programables, que aceptan un juego de comandos predefinido, y tarjetas programables capaces de incorporar dinámicamente a su memoria persistente programas provenientes del exterior y enriquecer así el conjunto de comandos que son capaces de ejecutar. Una de las tecnologías de tarjetas programables es Java Card. Una Java Card es una tarjeta inteligente capaz de incorporar y ejecutar programas escritos en un subconjunto del lenguaje de programación Java. La principal ventaja que aporta esta tecnología es la independencia de la plataforma, que nos permite desarrollar aplicaciones Java Card que se pueden ejecutar sobre cualquier tarjeta acorde a la especificación, independientemente del fabricante de la misma.

Debido a las restricciones de memoria y procesamiento de las tarjetas inteligentes el lenguaje Java Card es una versión reducida del lenguaje Java [JCADG 2.1]: sólo existen los tipos primitivos `boolean`, `byte`, `short`, carece de la clase `Thread` y `String` y no tiene recolector de basura, por lo tanto todo objeto que es instanciado en memoria persistente permanece en ella hasta que se elimine el cardlet de la tarjeta, siendo ésta una de las limitaciones que más a condicionado nuestro desarrollo.

4.1 Arquitectura de JCCM

Los desarrollos que se han realizado hasta la actualidad de módulos PKCS #11 con tarjetas inteligentes emplean tarjetas no programables o bien no utilizan la facilidad de programación de las tarjetas que sí son programables. Estas implementaciones se limitan a adaptar la funcionalidad criptográfica predefinida en las tarjetas a la visión lógica que debe ofrecer un módulo PKCS #11. La semántica definida por el estándar se recorta para

[3]El estándar ISO7816-3 permite una velocidad de hasta 115Kbps

adaptarla a la que ofrece la tarjeta para la que se realiza la implementación, que además queda ligada a una tarjeta de un determinado fabricante.

La característica distintiva de nuestro desarrollo es la implementación por parte del cardlet presente en la tarjeta de parte de la funcionalidad definida en el estándar PKCS #11. Los objetos se almacenan en la tarjeta con los mismos atributos definidos en el estándar, y es el propio cardlet quien procesa las operaciones de gestión de objetos de Cryptoki, soportando así el almacenamiento de cualquier tipo de objeto e implementando la semántica completa de búsqueda, copia, creación y borrado.

En los siguientes apartados se explican las diferentes partes de que consta la aplicación Java Card desarrollada en este proyecto.

4.1.1 Gestión de memoria dinámica en tarjetas Java Card

La memoria en una tarjeta inteligente es un recurso limitado y es una de las restricciones más importantes a tener en cuenta cuando se realizan aplicaciones para este tipo de sistemas. Para almacenar datos de forma persistente en una tarjeta inteligente, es necesario que residan en memoria EEPROM. El estándar ISO 7816-4 definió para ello una estructura de sistema de ficheros similar a la que tenemos en un ordenador y estandarizó APDUs para poder gestionarla. Las primeras versiones de Java Card, hasta la 2.0, imitaban este modelo de gestión de la memoria persistente: existían clases que replicaban la funcionalidad de las APDUs de acceso a ficheros definidas por ISO y que trataban de imitar el modelo de *streams* de Java. A partir de la versión 2.1 se abandonó este modelo de acceso a memoria persistente para sustituirlo por el método nativo en Java de instanciación de objetos: para disponer de un bloque de bytes en el almacenamiento persistente se instancia un objeto de la clase deseada; el acceso estructurado a la EEPROM, a través de los miembros del objeto instanciado, es así más directo que el efectuado a través de streams. Cuando en Java Card se habla de almacenamiento persistente no se bromea; los objetos instanciados no serán destruidos ni tampoco será liberado el espacio que ocupen por ningún recolector de basura, pues el estándar Java Card no contempla esta facilidad. Por lo tanto, todo objeto instanciado permanecerá durante la vida del cardlet: hasta que éste sea eliminado de la tarjeta.

Las funciones de Cryptoki permiten la creación dinámica de objetos y nosotros deseábamos que nuestra implementación no limitase esa flexibilidad. Podíamos haber diseñado una implementación capaz de almacenar un número predeterminado de objetos Cryptoki, p.e. un certificado X.509 y una clave privada RSA, pero eso limitaba la funcionalidad de nuestra implementación y por lo tanto las aplicaciones potenciales que podrían beneficiarse de ella. Topamos con el problema de desarrollar un cardlet capaz de crear, destruir y modificar dinámicamente un número indeterminado de objetos en un esquema de asignación de memoria que nunca libera los bloques asignados. La solución a este problema consiste en desarrollar un módulo de asignación de memoria dinámica que ofrezca funcionalidad similar a las tradicionales `malloc()` y `free()` de la biblioteca estándar del lenguaje C. La asignación de bloques se hace sobre un array de bytes Java cuyo tamaño se especifica en tiempo de compilación y que es reservado en memoria persistente en el instante de instalación del cardlet en la tarjeta.

Se ha adoptado un sencillo esquema para gestionar los bloques libres y ocupados: todo bloque tiene una cabecera de dos bytes utilizado para la gestión de la memoria y el resto disponible para almacenar datos. La cabecera puede considerarse como una palabra de dos bytes, primero el byte de mayor peso, en la que el bit más a la izquierda es un indicador de si el bloque está libre (0) u ocupado (1) y los 15 bits restantes indican el tamaño del bloque en bytes excluyendo la cabecera. Por tanto, el tamaño máximo de bloque y de memoria dinámica que podemos manejar con este esquema es de 2^{15} bytes (32Kb), suficiente para abarcar toda la memoria persistente disponible en tarjetas inteligentes de tecnología actual. Otra implicación es que las direcciones son de dos bytes. El tamaño del array empleado en la implementación actual es de 4Kb, suficiente para albergar dos certificados propios, es decir, para los que se almacena también una clave privada asociada, (cada pareja certificado,clave privada ocupa algo menos de 2Kb en nuestra estructura de almacenamiento persistente) y soportar además búsquedas, que requieren para su ejecución disponer de almacenamiento temporal.

En cualquier instante el array estará compuesto por una serie de bloques contiguos libres y ocupados mezclados, dependiendo de la secuencia de asignaciones y liberaciones previa. Inicialmente tenemos un bloque libre que abarca el array completo. Cada vez que se solicita un bloque se recorren los bloques en secuencia desde el primero (que siempre está en la dirección 0, índice `[0]` en el array) buscando un bloque libre de tamaño suficiente. Cuando ésto ocurre, si el bloque encontrado es de mayor tamaño que el bloque solicitado y suficientemente grande como para poder hacer un nuevo bloque de la parte sobrante, se divide en dos: un fragmento ocupado de tamaño exacto para albergar el tamaño solicitado y el fragmento restante que se marca como libre. Durante este recorrido de búsqueda se fusionan todos los bloques libres contiguos que se vayan encontrando. Si alcanzamos el final del array sin encontrar un bloque suficientemente grande se devuelve la dirección 0, que es una dirección no válida pues por definición apunta a la cabecera del primer bloque. Las clases que solicitan memoria reciben la dirección del campo de datos del bloque, no de la cabecera.

4.1.2 Almacenamiento y gestión de objetos de Cryptoki

En PKCS #11 los objetos están definidos como un array de atributos, donde cada atributo es una estructura de tamaño fijo con tres campos: un tipo de atributo, un puntero al valor y la longitud del valor. Las estructuras empleadas en el cardlet siguen un esquema parecido que describimos a continuación. La estructura de los objetos, es de longitud variable y consta de los siguientes campos:

- Enlace al siguiente objeto, 2 bytes. Todos los objetos PKCS #11 que se crean en la tarjeta se mantienen en una lista enlazada.
- Número de atributos del objeto, 1 byte. Este campo puede deducirse en función del tamaño del bloque en memoria dinámica donde se aloja la estructura y el tamaño de la parte fija, pero se ha optado por incluir el número de atributos en la estructura del objeto porque el ahorro en consumo de memoria que hubiese supuesto su no inclusión es despreciable: típicamente la tarjeta almacenará únicamente dos objetos, un certificado y una clave privada asociada por lo que el ahorro total en este caso sería de 2 bytes.
- Un número variable de estructuras de atributos, tantas como número de atributos tenga el objeto.

La estructura de un atributo consta de los siguientes campos:

- Tipo de atributo, 4 bytes. Es el valor binario definido en el estándar.
- Puntero al valor del atributo, 2 bytes. El campo valor, al ser de tamaño inherentemente variable, se almacena en su propio bloque de memoria dinámica. No almacenamos la longitud del valor en la estructura del atributo para ahorrar memoria. La longitud de este campo se obtiene de la cabecera del bloque de memoria dinámica en el que se aloja.

El almacenamiento de estas dos estructuras se hace sobre la memoria dinámica descrita en el apartado anterior. Existe una clase Java para ayudar en el manejo de cada una de estas estructuras, son la clase `ObjPatr` para los objetos y la clase `AttrPatr` para los atributos. Estas clases no pueden instanciarse debido al problema de no liberación de memoria mencionado en el apartado 4.1.1, por lo que únicamente poseen métodos y miembros estáticos. Se utilizan para mapear porciones del array de memoria dinámica sobre la estructura correspondiente y para acceder cómodamente a los distintos campos de cada estructura: poseen métodos para establecer/obtener el valor de cada campo y para liberar toda la memoria dinámica asociada a cada una de estas estructuras. Estas dos clases extienden la clase `Patr`, que aporta los métodos básicos para acceder a campos de tipo multibyte. Antes de emplear una de estas clases para acceder a una estructura alojada en memoria dinámica es necesario establecer primero la dirección inicial de la estructura mediante una llamada a `setAddr(short addr)`, lo que establece la dirección de referencia empleada por todos los métodos que acceden a miembros de una estructura (`get...()`, `set...()`).

4.1.3 Implementación del cardlet en tarjetas comerciales

Con el objetivo de demostrar la independencia de dispositivo de JCCM, en este proyecto se han empleado dos kits de desarrollo Java Card para entorno Linux pertenecientes a diferentes fabricantes:

- **GemXpresso RAD 211is de Gemplus**, [GemXpresso RAD 211 UG], estas tarjetas sólo implementan algoritmos de clave simétrica, por lo que el cardlet que se desarrolló en este caso, se limita al almacenamiento de claves y certificados, que se transfieren al ordenador (incluida la clave privada) para realizar las operaciones criptográficas soportadas por nuestra implementación.
- **Cyberflex for Linux Starter's Kit 2.1**, [Cyberflex SDK], estas tarjetas implementan RSA por lo que el cardlet desarrollado realiza operaciones criptográficas. La principal limitación con la que nos enfrentamos es que las tarjetas son Java Card 2.0 y en esta especificación todavía no se incluía un paquete de seguridad (`javacard.security` en Java Card 2.1). Así, para el acceso desde Java Card a las capacidades criptográficas, es necesario utilizar la extensión proporcionada por Schlumberger `javacardx.crypto`, por lo que la solución desarrollada sólo es válida para este tipo de tarjetas.

En la tabla 1 se muestra una comparativa entre las prestaciones proporcionadas por cada una de las tarjetas empleadas en nuestro desarrollo. Se observa que las tarjetas GemXpresso tienen unas velocidades de almacenamiento y lectura mucho mayores que las Cyberflex, tanto en escritura como en lectura. Estas diferencias creemos que son debidas a la mejor implementación de la máquina virtual Java de las tarjetas de Gemplus respecto a las de Schlumberger.

Tarjeta	Tamaño del Cardlet (Bytes)	Almacenamiento (ms)			Lectura (ms)
		Clave privada	Clave pública	Certificado	Certificado
GemXpresso	6437	20312	12998	16228	8934
Cyberflex	3992	38122	28180	34036	16155

Tabla 1: Comparativa entre tarjetas

5 Conclusiones

En este artículo hemos presentado la base de la PKI que hemos desarrollado, con la parte de cliente representada por JCCM, y la de servidor por ACerO.

JCCM es una alternativa flexible para gestionar certificados de cualquier tipo en tarjetas inteligentes, frente a otros trabajos que emplean tarjetas no programables o bien no utilizan la facilidad de programación. Nuestro enfoque no se limita a adaptar la funcionalidad criptográfica predefinida en las tarjetas a la visión lógica que debe ofrecer un módulo PKCS #11, ni recorta la semántica definida por el estándar para adaptarla a la que ofrece el fabricante en cada tarjeta. En la actualidad JCCM soporta dos tipos de tarjetas inteligentes: las GemXpresso RAD 211, y las Cyberflex Access de Schlumberger, y está disponible para su uso con Netscape para navegar de forma segura, cifrar y firmar correos.

ACerO representa nuestra apuesta por seguir utilizando un software tan probado como es openssl, a través del uso de servlets para exportarlo al mundo web de forma robusta y escalable.

JCCM y ACerO son partes de un mismo sistema, en el que faltan definir el soporte legal para poder denominarla PKI. En cualquier caso dejamos esta tarea a nuestros futuros usuarios, y abordamos los trabajos futuros en la línea de la certificación de atributos, mediante la integración en el sistema de módulos de autenticación PAM y listas de control de acceso.

Referencias

[ISO/IEC 7816-4] "ISO/IEC 7816-4: Integrated circuit(s) cards with contacts. Part 4: Interindustry commands for interchange", ISO/IEC, 1995.

[JCADG 2.1] "Java Card Applet Developer's Guide. Java Card Version 2.0", SUN Microsystems, Agosto de 1998.

[JCADG 2.0] 'Java Card Applet Developer's Guide. Java Card Version 2.1", SUN Microsystems, Agosto de 1999.

[GemXpresso RAD 211 UG] "GemXpresso RAD 211 User Guide Version 1.0", Gemplus, Octubre 1999

[GemXpresso RAD 211 CRM] "GemXpresso RAD 211 Card Reference Manual Version 1.0", Gemplus, Octubre 1999

[Cyberflex PG] "Cyberflex Access Developer's Series. Programmer's Guide", Schlumberger, Septiembre 1999.

[Cyberflex SDK] "Cyberflex Access Software Developer's Kit 2 - Release Notes", Schlumberger, Noviembre 1999.

[FAQ Schlumberger] Schlumberger, http://cyberflex.com/Support/support.html

[HTTPS] "HTTP Over TLS", Rescorla, E., IETF RFC 2818, Mayo 2000.

[X.509] "Internet X.509 Public Key Infrastructure Operational Protocols: FTP and HTTP". R. Housley, P. Hoffman. IETF RFC 2585, Mayo 1999.

[CORE00] "Core Servlets and JavaServer Pages". Hall, Marty. Prentice Hall 2000.

[SC SDK] 'Smart Card Developer's Kit", Scott B. Guthery, Timothy M. Jurgensen. Macmillan Technical Publishg. 1998.ISBN 1-57870-027-2.

[SC APP. DEV. JAVA] 'Smart Card. Application Developement Using Java", Uwe Hansmann, Martin S. Nicklous, Thomas Schack y Frank Seliger, Springer, 2000. ISBN 3-540-65829-7.

[PKCS#11] "PKCS #11 v2.10: Cryptographic Token Interface Standard", RSA Laboratories Inc., Diciembre 1999 (003-903052-210-000-000).

[SMARTSIGN] "Smart Sign", Tommaso Cucinotta, http://sourceforge.net/projects/smartsign

[GPKPKCS#11] "GemSAFE Products", Gemplus, http://www.gemplus.com/products /software/gemsafe/index.html

[SLBCBPKCS#11] "Cyberflex Access SDK", Schlumberger, http://www.cyberflex.com/Products

"Estudio comparativo de políticas de revocación de certificados: *OCSP vs. Overissued-CRL"*[*]

José Luis Muñoz, Juan Carlos Castro, Jordi Forné Muñoz
Departament d'Enginyeria Telemàtica. Universitat Politècnica de Catalunya.
Jordi Girona 1 y 3. Campus Nord, Mód C3, UPC. 08034 Barcelona
E-mail: {jlmunoz, jcastro, jforne}@mat.upc.es

Abstract*. The emergent applications of electronic commerce require security services. The authentication is the most difficult to archive, and the employment of public key infrastructures (PKI) is required. The revocation of the certificates is the major cost in the whole PKI. Up to the date, different revocation policies have been proposed. In this paper we evaluate two of the main proposed policies: OCSP and Overissued-CRL.*

1 Introducción

En los entornos de interconexión de usuarios abiertos, tal y como es Internet, se necesita una forma eficaz de proporcionar los servicios básicos de seguridad como son la autenticación de usuarios, la confidencialidad, la integridad de la información transmitida y no repudio (irrenunciabilidad).

Para proporcionar los servicios de seguridad básicos, necesarios para la construcción de aplicaciones seguras para los usuarios finales, en el entorno que estamos tratando, es necesario el empleo de criptografía de clave pública. Los orígenes de la criptografía de clave pública se remontan al estudio realizado por Whitfield Diffie y Martin Hellman en 1976 [7], donde se propuso por primera vez el uso de una pareja de claves, una pública (conocida) y otra privada, para la realización de las operaciones criptográficas.

Para la distribución de las claves públicas en entorno abierto, la solución que más ampliamente se ha adoptado, consiste en recurrir a una tercera parte confiable, llamada Autoridad de Certificación (AC), propuesta por primera vez en 1978 por Kohnfelder [16]. Las funciones de una AC consisten en verificar la identidad de los solicitantes de certificados, crear los certificados y proporcionar los mecanismos necesarios para comprobar la validez de los certificados emitidos. En la Figura 1 se muestra un esquema general de la AC.

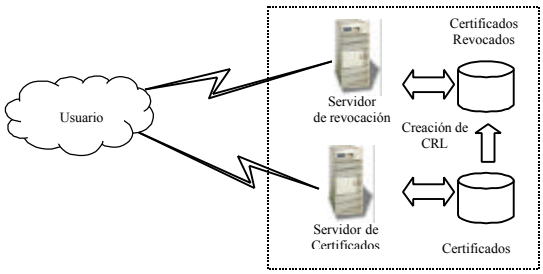


Figura 1 Esquema general de una AC

La AC se divide en dos partes: servidor de certificados, es el encargado de generar y administrar los certificados digitales, y el servidor de revocación, encargado de proporcionar la información del estado de los certificados.

El formato y las funciones de un certificado digital se encuentran definidas en el estándar X.509 definido por la ITU [12] y [13], dicho estándar se basa en el conjunto de entradas de directorio X.500 [4]. Para el uso de certificados digitales se está definiendo una infraestructura adecuada para asegurar la autenticación de las entidades involucradas en una transacción; a esta infraestructura se le conoce como PKI o Infraestructura de Clave Pública, que genéricamente se define como "el conjunto de Hardware, Software, personas, políticas y procedimientos necesarios para crear, administrar, almacenar, distribuir y revocar certificados de claves publicas basados en criptografía de clave pública"[2].

En la actualidad se ha generado una gran expectativa respecto a una nueva generación de aplicaciones que incorporen los mecanismos de seguridad necesarios para ofrecer determinados servicios que de otra forma no serían posibles. Sin embargo y a pesar de la madurez de la tecnología de clave pública, sigue sin producirse un despliegue masivo de este tipo de servicios, esto se debe en gran medida a la todavía inmadura PKI. La PKI presenta varios problemas, sobretodo cuando se quiere extender sus soluciones a una gran cantidad de usuarios, es decir, presenta problemas de escalabilidad. Entre estos problemas uno de los más graves es la revocación de certificados. A este respecto, el tema de la revocación ha sido objeto de diferentes estudios [5], [6], [15], [17], y [19]. Cada uno de estos estudios propone una política de revocación diferente, aunque no proporcionan un modelo general para evaluar dichas políticas. En este artículo se presenta un estudio comparativo de las dos principales políticas de revocación. Para

[*] Este trabajo ha sido desarrollado dentro del proyecto ACIMUT CICYT TIC2000-1120-C03-03

poder evaluar políticas de revocación en primer lugar se define y se justifica que parámetros son considerados relevantes y finalmente se evalúan dichos parámetros en las políticas estudiadas. El principal parámetro evaluado es la velocidad de transmisión necesaria.

2 Políticas de Revocación de Certificados

Las políticas de Revocación de Certificados definen la forma en la cual un usuario puede obtener información relativa al estado de un determinado certificado digital. El análisis de la forma en la cual debe ser proporcionada dicha información al usuario se ha realizado en los estudios mencionados anteriormente, de los cuales podemos concluir que las políticas de revocación de certificados se pueden dividir en dos grandes grupos:

Políticas de revocación basadas en distribución de listas. La primera de estas políticas se basa en la generación de listas de revocación de certificados a la cual se irán añadiendo los certificados revocados a medida que se van produciendo [12]. Este método fue introducido en 1988 en la versión 1.0 de CRL definido en el estándar X.509, dicha lista de certificados revocados hace uso de una base de datos centralizada, a la cual cada usuario debe acceder cada vez que desee emplear un certificado para verificar si el certificado dado ha caducado. El contenido de la CRL consta de diferentes campos: Versión, Emisor de la lista, Algoritmo empleado para firmar la CRL, Fecha de expiración de CRL, Información de los certificados revocados etc. La primera mejora a emplear sobre este método es mantener la CRL en una memoria caché del usuario, de esta forma el usuario no ha de solicitar al servidor de revocación el envío de una nueva CRL si ya dispone de una CRL no caducada en su caché. Como principal problema de esta política tenemos que todos los usuarios tienen el mismo tiempo de expiración para las CRL almacenadas en sus caches, esto hace que todas las CRL caduquen en el mismo tiempo y que el servidor de revocación tenga una carga elevada entorno a este instante. Para solucionar este problema en [5] se propone cambiar el periodo de emisión (creación) de CRL's en el servidor de revocación. Se sugiere un valor de emisión de CRL's inferior al tiempo de expiración, de esta forma se reparte la carga hacia el servidor de revocación en el tiempo.

Otra nueva mejora es el empleo de la política basada en delta CRL, la cual fue introducida en el estándar X.509 en 1994. Esta política consiste en generar una lista de revocación de certificados base (CRL_base) y después de un periodo de tiempo delta, menor que el tiempo de expiración de la CRL_Base, se genera una nueva CRL denominada CRL_Delta que únicamente contiene las revocaciones producidas desde la emisión de la CRL_Base. Con este método el usuario ya no necesita obtener toda la CRL si dispone de una CRL_Base no caducada, sino simplemente necesita la última CRL_ Delta. De esta forma la CRL_Delta al tener un tamaño menor respecto a la CRL_Base, disminuye los costes de transmisión [11]. Una mejora a esta política se propone en [6].

Otra mejora que se puede aplicar en general a todas las políticas es la utilización de diferentes puntos de distribución de CRL, este método se introdujo en 1997 [12] para X.509 V.3, cada CRL contiene las revocaciones de un determinado grupo de certificados. Los criterios para crear estos grupos pueden ser: geográficos, de nivel de importancia, de ámbito de uso, de motivo de revocación.

Políticas de revocación On-Line. En este tipo de políticas se caracteriza por el envío de información de un determinado certificado o certificados que el usuario solicita en un instante en particular. La primera política a destacar es el Sistema de Revocación de Certificados (CRS) la cual fue propuesta en 1996 por Silvio Micali [17]. Esta política se basa en un sistema de firmas Off Line/On Line [8]. Una mejora a este sistema, propuesta por Aiello en 1998 [1], fue denominada Sistema de Certificados Revocados Jerárquicos (HCRS), este método asigna a cada usuario un conjunto de cadenas Hash en lugar de una sola.

La siguiente política fue la propuesta hecha por Paul Kocher [15] en la cual presenta un modelo de Arbol de Certificados Revocados (CRT). Posteriormente Moni Naor y Kobi Nissim presentaron un modelo [20], basado en el árbol de certificados revocados presentado por Kocher [15].

El grupo de trabajo del IETF (Internet Engineering Task Force) ha desarrollado una propuesta para emitir el estado de certificados llamada On Line Certificate Status Protocol (OCSP). Este protocolo nació con la base de dos borradores propuestos [3] y [18], actualmente se encuentra en discusión una nueva propuesta [19], dicha propuesta expira Agosto 2001. Mediante este protocolo el usuario recibe el estado del certificado o grupo de certificados que necesita. En [10] Fox y La Macchia presentan una alternativa a OCSP basada en la emisión de un nuevo certificado X.509 como respuesta a las solicitudes del estado de un certificado dado, dicho certificado indicara si el certificado en cuestión se encuentra revocado o no.

3 Evaluación de políticas de revocación

Se han mencionado diferentes técnicas de revocación de certificados en las secciones anteriores, dichas técnicas se pueden agrupar en consultas Off-Line y consultas On-Line [10]. En la presente sección se expondrá un modelo de evaluación basado en teoría de colas que nos permitirá encontrar las diferentes características que presentan dichas técnicas. Un primer modelo se

puede encontrar en [9], en donde se hace un análisis de una Autoridad de Certificación sin el empleo de CRL observando la relación existente entre el número de usuarios y tiempo de servicio del servidor de revocación.

3.1 Parámetros de evaluación

En este apartado se determinan justificadamente los parámetros que se van a utilizar para la evaluación de las diferentes políticas de revocación. La parte crítica del sistema de revocación es el servidor de revocación, por lo que los parámetros de evaluación que vamos a considerar serán parámetros del servidor de revocación. En un estudio más exhaustivo habría que tener en cuenta también parámetros de evaluación en los usuarios (como costes de caché) y en la entidad emisora de certificados (coste de emisión de un certificado). En la Tabla 1 podemos encontrar los parámetros que consideramos relevantes en la evaluación del servidor de revocación:

B	Velocidad binaria de transmisión de salida del servidor
$T_{validación}$	Tiempo medio de validación de certificado.
T_{cpu}	Tiempo de procesador que consume una validación de certificado.

Tabla 1 Parámetros críticos en el diseño del servidor de revocación

A continuación se justifica porque cada uno de estos parámetros se considera crítico:

- La velocidad binaria de transmisión de salida del servidor B es un parámetro primordial ya que determinará el caudal del flujo de salida de datos mínimo que ha de tener el servidor de revocación hacia los usuarios, ya que la velocidad de transmisión sentido usuario-servidor es menos crítica. La velocidad de transmisión esta directamente relacionada con el ancho de banda necesario para poder realizar la transmisión, por lo que en adelante se utilizará indistintamente velocidad de transmisión o ancho de banda necesario.
- El $T_{validación}$ es un parámetro a tener en cuenta por lo que respecta a los *timeouts* de las conexiones TCP establecidas para realizar la consulta, normalmente los protocolos de consulta suelen ser LDAP o http que son protocolos de aplicación sobre TCP. Si el servidor de revocación nos proporciona un $T_{validación}$ muy elevado es posible que se puedan perder consultas por temporización de conexiones TCP o que los usuarios abandonen la consulta.
- El valor de T_{cpu} tiene mayor o menor importancia dependiendo de la política de revocación, ya que hay políticas que han de firmar los resultados en el mismo instante de la validación con lo que se puede convertir en un parámetro crítico.

3.2 Modelo de evaluación

En este apartado se hallarán los parámetros de evaluación que han sido considerados relevantes en el apartado anterior para el servidor de revocación. Para ello se va a emplear un modelo basado en teoría de colas. En

Figura 2 se muestra el esquema general del modelo de servidor de revocación que vamos a utilizar.

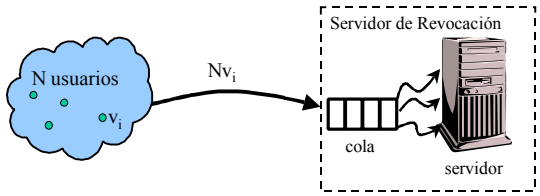


Figura 2 Modelo basado en teoría de colas para el servidor de revocación

N es el número de usuarios de la población que maneja el modelo. Suponiendo N relativamente grande y por las características del servicio de validación de certificados, podemos aproximar el tiempo entre peticiones de validación de certificados que genera la población del modelo mediante una estadística exponencial, tal y como hacen otros modelos de tráfico, como por ejemplo los modelos de tráfico para las redes de voz.
La expresión de la densidad de probabilidad del tiempo entre peticiones consecutivas para la estadística exponencial viene dada por ve^{-vt}. Donde v es la tasa agregada media de peticiones de validación al servidor de revocación de la población del modelo. Si v_i es la tasa de peticiones de validación de uno de los miembros de la población, la del sistema sería $v = Nv_i$.

En segundo lugar aproximaremos el tiempo de servicio T_s de una petición de validación por un valor constante (determinista), es decir, todas las consultas una vez en el servidor tardan el mismo tiempo en procesarse. En general T_s se puede expresar como $T_s = T_{tx} + T_{cpu}$. Donde T_{tx} es el tiempo de transmisión de los datos de la validación y T_{cpu} es el tiempo que precisa la CPU del servidor de revocación para preparar los datos de la validación.
Los datos a transmitir como resultado de una petición de validación los denominaremos genéricamente como CRL (lista de certificados revocados), aunque un caso particular es que la lista este compuesta por un solo certificado.
En general el tamaño de la CRL se puede expresar como la suma de una cabecera CRL_H más cierta información correspondiente a cada certificado revocado CRL_C.

$$CRL = CRL_H + c\ CRL_C \qquad (1)$$

En la CRL_H se puede encontrar dependiendo de la política de revocación parte o la totalidad de la siguiente información:

- Versión de la CRL.
- Algoritmo de firma.
- Firma digital.
- Fecha de emisión.
- Fecha de actualización.
- Extensión de X.509 para la política de validación.

En la CRL_C (información de cada certificado revocado) se puede encontrar parte o la totalidad de la siguiente información:

- Número de serie del certificado revocado.
- Fecha en la que fue revocado el certificado.
- Extensión de entrada de CRL.

Definiremos M como el número total de certificados emitidos por la entidad certificadora, si consideramos que cada usuario perteneciente a la población que puede realizar peticiones de validación dispone de un certificado tendremos que M = N. Por otra parte, el número de certificados revocados M_R es en general una función de M, del tiempo y de parámetros de la política de distribución de certificados que se aplique

$$M_R = f(M, t, ...) \qquad (2)$$

Para nuestro modelo utilizaremos una fórmula sencilla para M_R, suponiendo que es proporcional a M y que no varia con el tiempo

$$M_R = pM \qquad 0 \le p \le 1 \qquad (3)$$

Por otra parte, la velocidad binaria de transmisión de salida del servidor de revocación se puede expresar como

$$B = \frac{CRL\ [\text{bits}]}{T_{tx}[s]}\ [\text{bps}] \qquad (4)$$

Por último supondremos que el servidor dispone de una cola lo suficientemente grande como para poderla considerar infinita, de esta forma podemos aplicar al servidor de revocaciones un modelo clásico de teoría de colas [14] M/D/1. Se ha justificado poder aplicar un modelo de teoría de colas, ya que éste permite la obtención del valor de los parámetros de evaluación sin necesidad de realizar simulaciones, puesto que disponemos de expresiones analíticas para dichos parámetros.

Se define r como la carga del servidor de revocación mediante la siguiente expresión $r = vT_s$. De (1) y (4) se obtiene una expresión general para la velocidad binaria de transmisión B

$$B = \frac{CRL_H + \mathrm{c}\ CRL_C}{\frac{r}{Nv_i} - T_{cpu}} \qquad (5)$$

Finalmente el tiempo de validación se puede hallar según la siguiente expresión

$$T_{validacion} = \frac{T_s(2 - vT_s)}{2(1 - vT_s)} \qquad (6)$$

Substituyendo Ts por las expresiones en (4) se obtiene una expresión para (6) en función del tamaño de los datos CRL, de B y del tiempo de CPU consumido

$$T_{validacion} = \frac{(\frac{CRL}{B} + T_{cpu})\left[2 - v(\frac{CRL}{B} + T_{cpu})\right]}{2\left[1 - v(\frac{CRL}{B} + T_{cpu})\right]} \qquad (7)$$

3.3 Aplicación del modelo de evaluación

Dependiendo del escenario en el cual se ha de aplicar el esquema de revocación de certificados, resulta adecuado el empleo de una determinada política de revocación u otra. Esto es así porque cada política intenta optimizar diferentes parámetros, como por ejemplo, las tasas de petición de validación contra el servidor de revocación, el ancho de banda de transmisión, reducir el tamaño de la CRL, bajar los requerimientos en el servidor de revocación (RAM, CPU…) etc.

En este apartado se mostrará la forma de aplicar el modelo descrito en 3.2 para evaluar la bondad de dos de las principales políticas de revocación de certificados:

- OCSP, la principal política de tipo On-Line.
- *Overissued-CRL*, la política de distribución de listas básica.

3.3.1. Aplicación a la política OCSP

La aplicación del modelo a OCSP, es directa, únicamente determinar los parámetros concretos de la misma. En primer lugar la tasa de peticiones será la que nos proporciona la ecuación $v = Nv_i$.

Por lo que respecta a T_{cpu} consideraremos que la lista de certificados revocados se encuentra almacenada al completo en la memoria RAM del servidor de revocación y que el tiempo de búsqueda de un certificado revocado es despreciable frente al tiempo de la firma de la respuesta. Se ha estimado como parámetro típico de firmado para los datos respuesta a una petición OCSP un valor de 20ms.

Consideraremos que se realizan peticiones de estado de revocación para un solo certificado a la vez. Teniendo en cuenta este hecho podemos expresar B como

$$B = \frac{CRL_H + CRL_C}{r/v - T_{cpu}} \qquad (8)$$

Los valores de CRLH y CRLC para certificados X.509 se hallaron utilizando el ejemplo en [19]. Dichos valores se pueden observar en la Tabla 2.

Parámetro	Valor [Bytes]
CRL_H	130
CRL_C	28

Tabla 2 Valores de CRL para OCSP

3.3.2. Política *Overissued-CRL*

Overissued-CRL es una política que requiere de una memoria caché en los usuarios para almacenar la CRL durante un cierto tiempo. De esta forma, el servidor de revocación emite una CRL que tiene validez desde el momento de su creación hasta su tiempo de expiración T_{exp}.
La ventaja de este tipo de política es que los usuarios realizarán una petición de validación únicamente si en su caché no tienen una CRL no caducada, produciéndose por tanto, una disminución de la tasa de peticiones al servidor de revocación. Basándonos en [5] estudiaremos la forma de aplicar la política a nuestro modelo.

Supongamos en primer lugar que el servidor de revocación crea una CRL, válida durante su periodo de expiración y que no crea otra CRL hasta que la CRL anterior deja de ser válida. Si el servidor de revocación actúa de esta forma tendremos que todas las copias en los caché de los usuarios de CRL que se habrán creado al mismo tiempo y que por tanto también expirarán en el mismo instante. Esto generará un tráfico de pico alrededor del tiempo de expiración con una tasa pico de peticiones de validación hacia el servidor de revocación igual a: $v_{pico} = v = Nv_i$. Esto implica que en los instantes cargados del servidor (alrededor del tiempo de expiración) se mantiene una tasa de peticiones de validación igual a la de la política OCSP, teniendo tamaños de CRL mucho mayores, por lo que sino se hace algo esta política no resulta viable. Para que el caché tenga los deseados efectos positivos sobre las peticiones de validación, se ha de aplicar una técnica de sobre-emisión o sobre-creación de CRL (*Overissued*-CRL) en el servidor de revocación.

En síntesis se trata de repartir las peticiones de validación de los usuarios en el tiempo y evitar la concentración de tráfico. Para conseguirlo, se hace crear CRL's al servidor de revocaciones a un ritmo mayor de lo que estas listas tardan en expirar, de esta forma se consigue que los usuarios tengan copias de CRL con diferentes tiempos de expiración y por tanto se logra repartir la carga de validación.

Si definimos O como el número de CRL emitidas durante el tiempo de expiración T_{exp}. Podemos expresar la nueva tasa de pico mediante la siguiente expresión

$$v_{pico} = \frac{Nv_i}{(O-1)(1-e^{-\frac{v_i T_{\exp}}{O}})+1} \qquad (9)$$

Emitiendo CRL's continuamente se puede conseguir una tasa de peticiones de validación de pico de

$$\lim_{O \to \infty} v_{pico}^{\min} = \frac{Nv_i}{v_i T_{\exp} + 1} \leq v \qquad (10)$$

La tasa de pico hacia el servidor depende del número de intervalos de sobreemisión O. Para simplificar la ecuación (9), podemos expresarla en función de un parámetro F, el cual denominaremos factor de sobreemisión y que nos indicará lo próxima o lejana que está la v_{pico} que estamos manejando respecto al mínimo teórico

$$v_{pico} = Fv_{pico}^{\min} = \frac{Nv_i(1+F)}{1+v_i T_{\exp}} \qquad (11)$$

Donde F

$$F = f(O) \text{ y } 0 \leq F \leq v_i T_{\exp} \qquad (12)$$

Finalmente, considerando que el tiempo de preparación de la CRL para ser enviada T_{cpu} es despreciable para esta política (firma *Off-Line* y datos en RAM) podemos obtener una expresión para B aplicando a la expresión (5) la expresión

$$B = \frac{(CRL_H + \text{pN } CRL_C)(Nv_i)(1+F)}{r(v_i T_{\exp} + 1)} \qquad (13)$$

Los valores de CRLH y CRLC para certificados X.509 se han hallado utilizando el ejemplo en [11]. Dichos valores se pueden observar en la Tabla 3.

Parámetro	Valor [Bytes]
CRL_H	170
CRL_C	17

Tabla 3 Valores de CRL para distribución de listas

4. Resultados

A continuación se analizan las políticas de revocación de certificados OCSP y *Overissued-CRL* para algunos escenarios, evaluando la influencia de parámetros como el ancho de banda, el tiempo de validación de un certificado en el servidor, el número de usuarios del escenario y el tiempo de expiración de una CRL en caché de un usuario.

4.1 Estudio del ancho de banda y tiempo de validación asociado

En primer lugar se realiza un estudio del comportamiento del ancho de banda y del tiempo de validación respecto a la tasa de peticiones de validación de los usuarios.

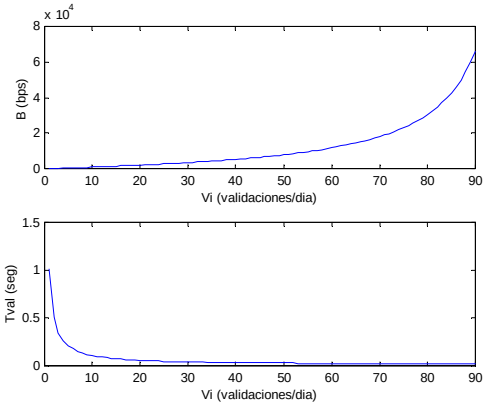


Figura 3 B(Vi) y Tval(Vi) para política OCSP.Escenario con 30,000 usuarios y una carga del 70% del servidor de revocación.

En la Figura 3 se observa un comportamiento de tipo exponencial ascendente por lo que respecta al ancho de banda necesario en función de la tasa de peticiones, a su vez el comportamiento del tiempo de validación tiene un comportamiento también exponencial pero de forma descendente. Los resultados obtenidos eran esperados, a mas validaciones por usuario, manteniendo la carga del sistema, se necesitara más ancho de banda y al tener un ancho de banda mayor tendremos menos tiempo de transmisión de los datos de la consulta y por tanto menos tiempo de validación.

En la Figura 4 se muestran los resultados de un escenario semejante para la política *Overissued-CRL*.

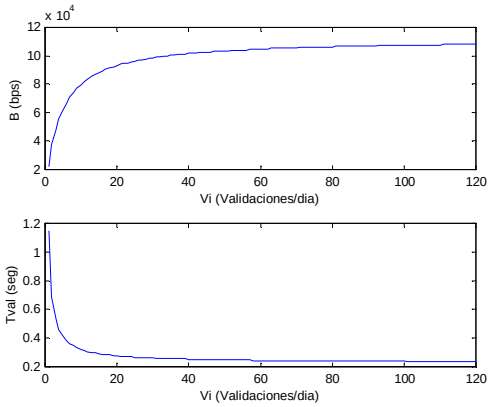


Figura 4 B(Vi) y Tval(Vi) para política Overissued-CRL. Escenario con 30,000 usuarios, porcentaje de certificados revocados 10%, F=0.1, tiempo de expiración de CRL de 6 h y carga del servidor de revocación del 70%.

Las conclusiones en este caso son parecidas al caso de OCSP con algunos matices. A medida que aumentamos la tasa de validaciones también aumenta B y disminuye el tiempo de validación. Sin embargo, la forma de crecimiento de B con v_i ya no es exponencial como en OSCP, sino que es lineal con v_i, para v_i pequeñas y tiende a un valor constante que denominaremos velocidad de transmisión umbral B_{umbral} para v_i grandes. Esta circunstancia era de esperar ya que cuando v_i crece es cuando los usuarios empiezan a sacar provecho de la copia de CRL de que disponen en su memoria caché, llegando a estabilizarse el ancho de banda necesario.

4.2 Estudio de la influencia del número de usuarios

En este apartado se realiza un estudio comparativo del ancho de banda para las políticas de revocación en cuestión, para diferentes tamaños de población del modelo.

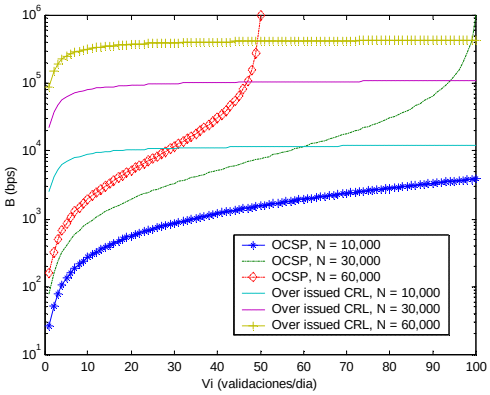


Figura 5 B(Vi) para diferentes poblaciones para políticas OCSP y Overissued-CRL. Escenario con un porcentaje de certificados revocados de 10 %, un factor F de 0.1, un tiempo de expiración igual a 6 horas y un nivel de carga del servidor de revocación de 70 %.

En la Figura 5 se observa que la política OCSP presenta un menor ancho de banda para tasas de validación pequeñas, conforme dicha tasa aumenta el ancho de banda aumenta hasta volver al sistema inestable, esta inestabilidad se produce a tasas más bajas conforme la población crece. Por otra parte, el ancho de banda para la política *Overissued-CRL* tiende a un valor constante (B_{umbral}) para valores altos de validación. A mayor población obtenemos un mayor B_{umbral}.

4.3 Estudio de la influencia del tiempo de expiración

En la Figura 6 se observa que al aumentar el tiempo de expiración, disminuye el ancho de banda necesario, lo cual implica un compromiso entre riesgo en las operaciones con certificados y el ancho de banda utilizado.
Por otra parte, podemos apreciar que las curvas para los diferentes tiempos de expiración alcanzan el valor B_{umbral} para diferentes v_i. Esta circunstancia se explica porque para tiempos de expiración grandes con tasas bajas ya podemos sacar el máximo partido a la memoria caché y por tanto se estabiliza el ancho de banda necesario para estas tasas bajas, mientras que para tiempos de expiración pequeños son necesarias tasas v_i mayores para llegar a sacar partido de la memoria caché necesitando de esta forma una v_i mayor para llegar a B_{umbral}.

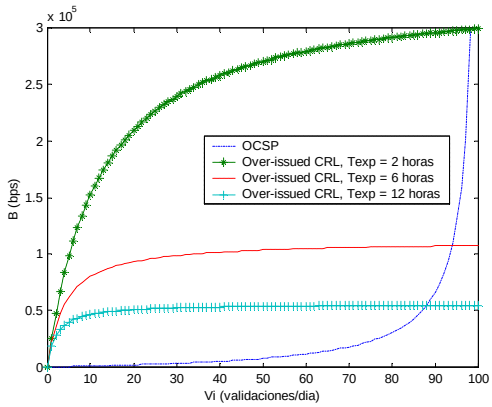


Figura 6 B(Vi) para diferentes tiempos de expiración para políticas OCSP y Overissued-CRL. Escenario con un porcentaje de certificados revocados de 10 %, un factor F de 0.1 y un nivel de carga del servidor de revocación de 70 %.

A pesar de que la curva $B(v_i)$ para OCSP no depende del tiempo de expiración, se ha incluido en la Figura 6 para tomarla como referencia en la evaluación de la conveniencia de una política u otra por lo que al consumo de ancho de banda se refiere. A este respecto, se observa que conforme el tiempo de expiración se hace más restrictivo (pequeño) el cruce de ambas curvas, que nos determina cual es más conveniente, se produce a tasas de validación por usuario más altas.

4.4 Estudio de un escenario

En este apartado se va a definir un escenario concreto para el que se va a evaluar la política más conveniente en función de los parámetros que se han establecido a lo largo del presente documento.

En primer lugar se define el escenario de evaluación, vamos a suponer un escenario de tipo mediano con una población de 30,000 usuarios.

Se tomará un porcentaje de certificados revocados del 10%. En el caso de la política *Overissued*-CRL se considera un factor de sobreemisión de 0.1. Por otra parte, usualmente en los sistemas prácticos, normalmente el ancho de banda disponible, es un parámetro fijado Además, se va a evaluar la política más conveniente en dos supuestos, asumiendo que el servidor posee un ancho de banda de bajada hacia los usuarios de 50Kbps y 100Kbps respectivamente.

Política	B **[Kbps]**	V_i^{MAX} **[val/dia]**	T_{exp}^{MIN} **[h-m]**	$\overline{T_{val}}$ **[ms]**
OCSP	50	124	0''	12
	100	133	0''	11
***Overissued* CRL**	50	∞	9h23'	512
	100	∞	4h41'	512

Tabla 4 Parámetros de evaluación escenario

En la Tabla 4 se observan los diferentes parámetros[1] de evaluación (tasa de peticiones de validación por usuario y día máxima, tiempo de expiración mínimo y tiempo de validación medio) que nos proporciona cada política, para el escenario propuesto.

En función de los datos proporcionados por la Tabla 4, el administrador del servicio de revocación puede elegir la política a emplear en cada caso. Por ejemplo, para el caso del ancho de banda de 50Kbps, si tenemos tasas de validación inferiores a 120 validaciones por día y usuario, la política más conveniente es OCSP, ya que el tiempo de expiración es nulo, no necesitamos ningún tipo de memoria caché en los usuarios y los tiempos de validación medios son un orden de magnitud menor que en el caso de la otra política. Sin embargo, si la tasa de validaciones comienza a incrementarse por encima de 120, el servidor de revocación se vuelve inestable para OCSP, en ese caso es más adecuada la política *Overissued-CRL*. Se ha de recordar que para utilizar esta política es necesario utilizar

[1] Servidor de revocación trabajando al 100% de su capacidad.

memoria caché en los usuarios y tenemos un cierto tiempo de expiración, por lo que no podemos estar seguros de la validez de un certificado en tiempo real. Análogas conclusiones se pueden observar para el caso del ancho de banda de 100Kbps.

Conclusiones

En este artículo se presenta un estudio de las políticas de revocación *OCSP* y *Overissued*-CRL, observando para cada una de estas el comportamiento de los parámetros considerados críticos para la entrega del estado de revocación de un certificado. Los principales parámetros de evaluación son el Ancho de Banda, Tiempo de Validación, Tiempo de Expiración y Tasa de Validación Máxima para una población dada. Se ha estudiado el efecto de cada uno de estos parámetros individualmente y en conjunto para un escenario particular. Finalmente se han alcanzado las siguientes conclusiones:

- En general, para tasas de validación altas el sistema necesita anchos de banda elevados. En el caso de OCSP, el sistema se satura alcanzando un valor de ancho de banda infinito; mientras que el Overissued-CRL el sistema alcanza un valor umbral de ancho de banda, por lo que el sistema es capaz de soportar tasas de validación infinitas, debido a que las tasas de validación reales son constantes.
- A mas validaciones por usuario, manteniendo la carga del sistema, se necesitara más ancho de banda y al tener un ancho de banda mayor, tendremos menos tiempo de transmisión de los datos de la consulta y por tanto menos tiempo de validación (Figuras 3 y 4).
- En CRL el ancho de banda necesario es proporcional al cuadrado del tamaño de la población (Figura 5).
- En CRL existe un compromiso entre ancho de banda y el nivel de riesgo. Un nivel de riesgo alto (tiempo de expiración grande), exige un ancho de banda pequeño y a la inversa (Figura 6).
- Para tasas de validación por usuario altas, la política CRL parece ser la más adecuada, mientras que para tasas pequeñas, parece más apropiada la política *OCSP* (Figura 6 y Tabla 4).

Referencias

[1] Aiello, W.; Lodha, S.; Ostrovsky, R; *"Fast Digital Revocation"*. Advances in Cryptology. Crypto 98. Lecture in Computer Science. Springer-Verlag, Nº 1462. August 98. Pags. 137-152.

[2] Arsenault A.; Turner, S.; *"Internet X.509 Public Key Infrastructure PKIX Roadmap"*. IETF Internet Draft, Octubre 1999. Draft-ietf-pkix-roadmap-04.txt

[3] Branchaud, M.; *"Internet Public Key Infrastructure": Caching the Online Certificate Status Protocol"*, Internet Draft, 1998. draft-ietf-pkix-ocsp-caching-00.txt

[4] CCITT. "*Recommendation X.500: The directory-overview of concepts, models and services."* 1988.

[5] Cooper, A. David. "*A Model of Certificate Revocation*". Proceedings of the Fifteenth Annual Computer Security Applications Conference, December 99, Pages 256-264.

[6] Cooper, A. David. *"A more efficient use of Delta-CRLs"*. Proceedings of the 2000 IEEE Symposium on Security and Privacy. Computer Security Division National Institute of Standards and Technology. May 2000. Pages 190-202.

[7] Diffie, W.; Hellman, M. "*New directions in cryptography"*. IEEE Transactions on Information Theory., IT-11(6): November 1976. Pages. 1644-1654.

[8] Even, S.; Goldreich, O.; Micali, S. *"On-Line/Off Line Signatures"*. Journal of Criptology. Vol. 9. 1996, Pages. 35-67.

[9] Castro, J.C.; Forné, J.; *"A Model to Evaluate Certificate Revocation"*; 4th World Multiconference on Systemics, Cybernetics and Informatics (SCI2000) and the 6th International Conference on Information Systems Analysis and Synthesis (ISAS2000), Orlando (Florida). 2000.

[10] Fox, B.; LaMacchia, B; "*Online Certificate Status Checking in Financial Transactions: The Case for Re-issuance"*. Financial Cryptography-FC 99, Lecture Notes in Computer Science, Springer-Verlag, Vol. 1648, 1999 Pages. 104-117

[11] Housley, R.; Ford, W.; Polk, W; Solo, D.; *"Internet X.509 Public Key Infrastructure Certificate and CRL Profile"*. RFC 2459, Enero 1999.

[12] ITU /ISO "*Recommendation X.509. – Information technology – Open Systems Interconnection - The Directory: Public Key and Attribute Certificate Frameworks"*. Agosto,1997.

[13] ITU /ISO "*Recommendation X.509. – Information technology – Open Systems Interconnection - The Directory: Autentication Frameworks. Technical Corrigendum"* Marzo, 2000.

[14] Kleinrock,L *"Queuing Systems. Volume I: Theory"*. John Wiley & Sons, Inc. 1975.

[15] Kocher, Paul C, "*On Certificate Revocation and Validation"*, Financial Cryptography-FC 98, Lecture Notes in Computer Science, Springer-Verlag, Vol. 1465, 1998 Pages. 172 – 177.

[16] Kohnfelder, L.M.; *"Towards a practical public-key cryptosystem"*. Master's thesis, MIT Laboratory for Computer Science, May 1978.

[17] Micali, S. *"Efficient Certificate Revocation"*. Technical Memo MIT/LCS/TM-542b Laboratory for Computer Science. Massachusetts Institute of Technology. USA. 1996.

[18] Myers, M; Ankney, R.; Malpani, A.; Galperin, S.; Adams, C.; *"X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP"*. RFC 2560, June 1999.

[19] Myers, M.; Ankney, R.; Ankney, R.; Adams, C.; Farrell, S.; Covey, C.; *"Online Certificate Status Protocol, Version 2"*. March, 2001. Draft-ietf-pkix-ocspv2-02.txt

[20] Naor, M., Nissim K., "*Certificate Revocation and Certificate Update"*, Department of Applied Mathematics and Computer Science, 7th USENIX Security Symposium, 1998.

Compresión de vídeo empleando segmentación espacio-temporal jerárquica adaptativa

Cristina Urdiales, Antonio Bandera, J.A. Rodríguez y Francisco Sandoval
Departamento de Tecnología Electrónica, ETSI Telecomunicación
Universidad de Málaga, Campus de Teatinos, 29071 Málaga
Teléfono: 952 132 757 Fax: 952 131 447
E-mail: cris@dte.uma.es

***Abstract**. This paper presents a control mechanism to stabilize the frame rate of a video sequence in reception so that delay sensitive vision applications may work in a remote terminal. This goal is achieved by adapting the video flow to the channel state. If delay is increased, the data volume of each frame is reduced by mapping only its areas of interest at the highest resolution available, while the rest of the scene presents a decreasing resolution profile. A hierarchical spatiotemporal segmentation technique is proposed to extract areas of interest from the sequence in a coherent and efficient way. Besides, if available bandwidth is too reduced, uninteresting areas of the scene may be transmitted at a lower frame rate than interesting ones. The system has been tested with several video sequences captured under different conditions and for changing delays. The system offered good results, specially when the channel bandwidth was very reduced.*

1 Introducción

En visión artificial existe un importante conjunto de aplicaciones, como supervisión de tráfico o seguimiento de objetos, en las que resulta necesario disponer simultáneamente de un campo visual amplio y de un buen nivel de detalle. Una imagen de estas características implica la generación de un gran volumen de datos, lo que dificulta su transmisión y/o procesado a través de un medio de ancho de banda reducido. Mientras que el problema del procesado se ha resuelto empleando técnicas de visión activa, que permiten procesar selectivamente sólo determinadas áreas de la imagen en tanto que éstas puedan ser detectadas, el de la transmisión se ha solucionado comprimiendo la secuencia de vídeo antes de su transmisión. En las últimas dos décadas se han llevado a cabo importantes esfuerzos para estandarizar la compresión, surgiendo estándares como el MPEG-1/2 [1] o el ITU-T H.263/H.263+ [2] que, mediante la aplicación de determinados mecanismos de muestreo en el tiempo y el espacio y cuantificación en brillo o color, han conseguido unos factores de compresión bastante altos con unas pérdidas de calidad aceptables.

No obstante, cuando se trabaja con escenarios reales, la calidad de las imágenes resultantes tras aplicar estos esquemas de compresión puede llegar a degenerarse con rapidez [3]. Esta degeneración se debe, principalmente, a que los fotogramas que forman una secuencia pueden presentar grandes cambios tanto espaciales como temporales. Si este es el caso, cuando se aplica un determinado proceso de compresión con unos parámetros fijos, la calidad resultante no es constante en el tiempo. En estas situaciones sería necesario disponer de un algoritmo que permitiera adaptar el muestreo a las condiciones del entorno, pero hoy en día aún no existe ningún algoritmo de este tipo [3]. Por otro lado, las estructuras en que se basan estos estándares de compresión para llevar a cabo el muestreo espacial y temporal, como los cuadrados usados en MPEG-2 [1], son igualmente una fuente de degradación, ya que no se adaptan correctamente a la geometría irregular de las distintas entidades existentes en las imágenes. Así, si el nivel de compresión sube por encima de un determinado umbral, dichas estructuras son perfectamente apreciables en la imagen.

Las técnicas de compresión de vídeo han evolucionado posteriormente hacia técnicas más eficaces de representación de las imágenes que constituyen la secuencia. Así, las técnicas de compresión de esta segunda generación, como el estándar MPEG-4 [4], representan la imagen como un conjunto de regiones que se tratan de hacer corresponder con los distintos objetos reales presentes en la escena. El tratamiento de cada uno de estos objetos puede así independizarse del resto, y los niveles de compresión aplicados a cada uno de ellos diferir en función de su importancia en la escena (*dynamic coding*). La mayor dificultad de este tipo de técnicas radica en que es extremadamente difícil segmentar una imagen real en regiones correspondientes a objetos individuales. Por ello, si bien los estándares están disponibles desde hace algún tiempo, no se ha llegado a establecer un mecanismo previo de descomposición en objetos para su posterior compresión.

Este artículo presenta un nuevo sistema de compresión de secuencias de vídeo enfocado a aplicaciones en que es necesario disponer al mismo tiempo de un amplio campo visual y de una resolución elevada. En un gran número de

aplicaciones, las cámaras que captan la secuencia a procesar no están directamente conectadas al equipo que lleva a cabo el procesado, sino que la secuencia debe transmitirse previamente. Para reducir el enorme volumen de datos de este tipo de secuencias y acelerar su transmisión y procesado se recurre al uso de imágenes de resolución no uniforme, frecuentemente utilizadas en visión activa. En estas secuencias, sólo las áreas de interés de cada fotograma, que habitualmente se relacionan con los objetos presentes en la escena, se transmiten a máxima resolución, mientras que el resto de la escena presenta un perfil de resolución decreciente en función de su proximidad a los focos de interés. Las imágenes multiresolución presentan la ventaja de ofrecer un volumen de datos mucho menor que sus equivalentes a resolución uniforme. Adicionalmente, son geometrías flexibles capaces de reconfigurarse en tiempo real para presentar un factor de compresión mayor o menor según las necesidades de la aplicación sin que las áreas de interés presenten degradación alguna. Por último, es necesario indicar que las geometrías que se proponen en este trabajo son perfectamente compatibles con todos los algoritmos clásicos de procesado de imagen cuando se almacenan en una estructura jerárquica de datos que se presenta más adelante.

A priori, la única ventaja del sistema propuesto frente a las técnicas de compresión comentadas reside en que las áreas de interés de la imagen no presentan pérdida alguna, por lo que cualquier algoritmo de visión activa es capaz de procesarlas de la misma forma que procesaría una de imagen uniforme de alta resolución. Sin embargo, el problema de la detección de los distintos objetos de la escena sigue presente. Para efectuar dicha detección, en este trabajo se propone un nuevo algoritmo de segmentación espacio-temporal, que extrae objetos significativos de la escena a partir de su color, distribución espacial y movimiento. El hecho de utilizar el movimiento como uno de los descriptores fundamentales de la segmentación para la codificación basada en objetos ya ha sido propuesta en otros trabajos [5] y obedece al hecho de que es prácticamente imposible descomponer una imagen real en regiones significativas basándose únicamente en criterios espaciales. No obstante, una buena parte de las técnicas de segmentación basadas en movimiento no lo utilizan en combinación con criterios espaciales. Esto puede ocasionar discontinuidades en la detección de los objetos de la escena y, por tanto, una codificación incoherente de éstos. El algoritmo propuesto lleva a cabo la segmentación en el tiempo y en el espacio simultáneamente, de forma que los resultados son coherentes a lo largo de la secuencia. Este algoritmo podría usarse también para detectar objetos en escenas a efectos de codificarlos mediante algoritmos tipo MPEG-4, pero en este trabajo se utiliza simplemente para codificar a alta resolución las áreas cuyo desplazamiento presenta diferencias significativas respecto al resto de la escena [5]. Cuando la cámara se encuentra cuasi-estática o se desplaza lentamente, lo que ocurre en la mayor parte de las aplicaciones de visión artificial, dichas áreas son además las más rápidas de la escena. Así, en la aplicación que se presenta en este trabajo las áreas que presentan mayor velocidad se codifican a alta resolución, mientras que el resto presenta un perfil de resolución decreciente en torno a éstas.

Por último, es importante señalar que en muchas aplicaciones es imprescindible recibir una tasa estable de imágenes por segundo para que el procesado sea fiable. Si bien la geometría no uniforme propuesta tiene un parámetro de control que permite reducir el volumen de datos hasta un cierto límite, en determinados casos el estado del canal impone restricciones aún más severas. De acuerdo al esquema de compresión propuesto, las áreas a baja resolución son aquellas que casi no presentan variación en el tiempo. Así, puede asignarse una prioridad menor de transmisión a éstas áreas, dando máxima prioridad a las de mayor resolución. Las zonas de máxima prioridad se transmiten entonces a la tasa de imágenes por segundo establecida, mientras que el resto se transmiten con mayor o menor frecuencia en función del estado del canal. En recepción, la imagen completa se compone a partir de la información recibida y el fotograma más reciente disponible. Es inmediato constatar que si realmente no se han producido variaciones importantes en las áreas no transmitidas, éstas son casi iguales a las equivalentes en el fotograma anterior. Así, el extremo receptor no debería verse excesivamente afectado por esta técnica. Este mecanismo de control de flujo de datos adaptado a las condiciones del medio permite estabilizar la tasa de imágenes por segundo siempre y cuando el canal no se encuentre colapsado.

El sistema propuesto se ha desarrollado para una aplicación de seguimiento de vídeo. Los detalles más importantes del sistema, así como su implementación, se describen en la sección 2. La sección 3 presenta varios experimentos para distintas secuencias de vídeo sobre un canal con retardo variable. Por último, en la sección 4 se resumen las conclusiones del trabajo, así como futuros desarrollos.

2 Descripción del sistema

La Fig. 1 muestra un esquema del sistema propuesto, que consta de cuatro módulos principales. Los tres primeros módulos se ubicarían en el extremo donde se captura la secuencia de vídeo, mientras que el cuarto sería la interfaz de entrada del bloque donde se procesan las imágenes recibidas. El módulo de segmentación y seguimiento de objetos se encarga de extraer y caracterizar los objetos que componen la escena a

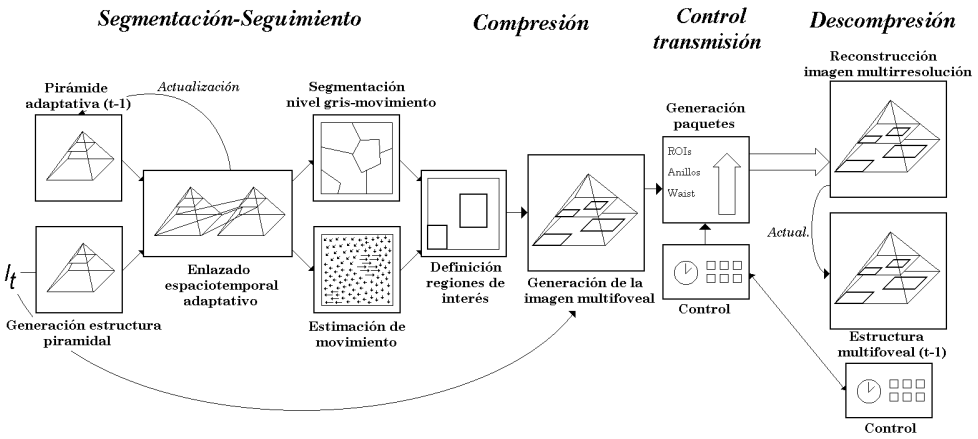


Figura 1: Esquema modular del sistema implementado

partir de una segmentación espaciotemporal de la secuencia. El módulo de compresión usa la información proporcionada por el primer módulo para determinar la posición de las regiones de interés presentes en la secuencia en cada instante de tiempo. Así, el módulo genera la imagen multiresolución que presenta dichas regiones de interés a alta resolución mediante una estructura multifovea de geometría adaptativa y desplazable [8]. Finalmente, el tercer módulo es el encargado de construir el paquete de datos que se transmite por el canal. Dicho paquete incluye inicialmente la totalidad de la imagen multiresolución construida, aunque, si las condiciones de retardo del canal empeoran, se pueden dejar de enviar temporalmente las regiones de menor resolución. Para estimar el ancho del canal en todo momento se mide el retardo asociado al mismo. El módulo de descompresión sirve de interfaz de entrada para el bloque de procesamiento. Dicho modulo lleva a cabo la reconstrucción de la imagen multiresolución, a partir de la información más reciente de que dispone para cada una de las áreas que integran la imagen completa. En las áreas de interés esa información siempre se recibe del canal. En el resto, si es necesario, se usa información disponible a partir de fotogramas anteriores.

A continuación se describen en mayor profundidad las características de los módulos que constituyen el sistema.

2.1 Segmentación y seguimiento de objetos

El objetivo de este módulo del sistema de compresión es segmentar correctamente una imagen de la secuencia de vídeo en regiones de características uniformes, así como determinar la velocidad de éstas. En aplicaciones reales, debido al ruido, cambios de iluminación y variaciones en las condiciones de captura, resulta difícil discernir entre objetos y fondo. Para dividir la imagen en regiones de características homogéneas, se propone el empleo de un algoritmo de segmentación jerárquica. La segmentación se basa en criterios espacio-temporales, teniéndose en cuenta en la agrupación de los píxels su proximidad espacial en el fotograma actual además de la constancia temporal de dicha agrupación en distintos instantes de tiempo. El hecho de trabajar de forma jerárquica permite acelerar la velocidad del proceso, ya que las técnicas de segmentación no jerárquica tradicionales suelen ser muy lentas. Este algoritmo devuelve el conjunto de regiones que forman la escena, así como sus características de movimiento y color. Las regiones de interés son aquellas que se desplazan a mayor velocidad.

En los siguientes subapartados se describen la generación de la estructura piramidal empleada (epígrafe 2.1.1) y su estabilización para conseguir la deseada segmentación espacio-temporal (epígrafe 2.1.2).

2.1.1 Generación de la estructura

Para llevar a cabo el proceso de segmentación propuesto es necesario que para cada fotograma de la secuencia se genere una estructura piramidal. La pirámide es una estructura jerárquica de datos, cuyos niveles presentan la misma imagen a distintas resoluciones. En la Fig. 2 se muestra una pirámide y seis de los niveles que la forman. Esta estructura se genera empleando el siguiente algoritmo:

1. Sea l=0. El nivel 0 de la pirámide es el fotograma de entrada.

2. Se crea un nivel l+1 con un cuarto de la resolución del nivel l. Para ello, por cada conjunto de 2x2 nodos (hijos) del nivel l se genera un nodo (padre) en el nivel l+1. El color de este nodo padre será la media del color que presentan sus cuatro hijos.

3. Cada uno de los nodos hijo del nivel l se enlaza con su nodo padre, para mantener la información topológica entre nodos.

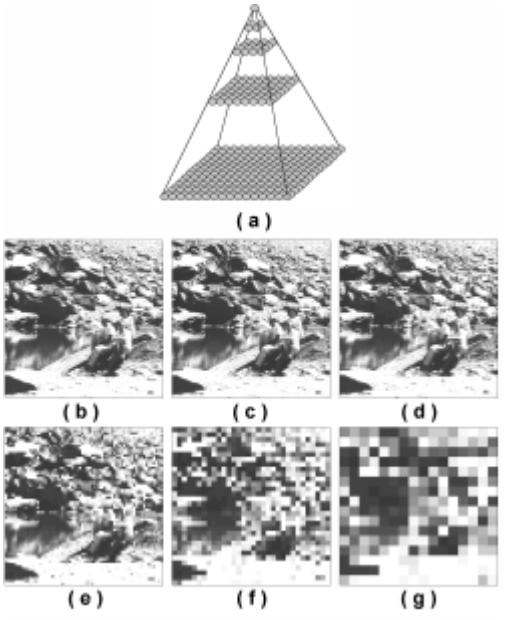


Figura 2: a) Estructura piramidal, y b-g) niveles de la pirámide construida sobre la imagen b).

4. Sea $l=l+1$. Volver al paso 2 hasta que l presente el número de nodos con que se desee trabajar (habitualmente 8x8 ó 16x16).

Se puede observar que cualquier nivel de la estructura piramidal ofrece una segmentación burda de la imagen: si se selecciona un nivel L determinado, cada uno de los nodos que lo forman estará conectado a una región de forma rectangular en la base. Esta segmentación no es válida, ya que no se corresponde con la distribución real de colores de la imagen de entrada sino que, para construir un nodo de los niveles superiores de la estructura se fuerza la agrupación de nodos de colores distintos en la base de la misma.

2.1.2 Estabilización de la estructura

La estructura piramidal descrita en el epígrafe 2.1.1 se emplea tradicionalmente para diezmar la imagen de entrada y así acelerar cualquier tipo de procesado sobre dicha imagen.

La estabilización adaptativa de una única pirámide como método de segmentación fue propuesta originalmente por Burt y otros [6], y consiste en cambiar los enlaces entre nodos hijos y padres de niveles consecutivos de forma iterativa, para que los nodos de color similar se agrupen en regiones homogéneas de forma irregular. De esta manera, cualquier nivel de la estructura está enlazado a un conjunto de regiones de color homogéneo en la base, cuyo número coincidirá con el de nodos en el nivel seleccionado. Obviamente, los resultados de esta segmentación están condicionados por el hecho de que el número real de clases y el de nodos del nivel seleccionado son el mismo. Sin embargo, este problema se soluciona fácilmente seleccionando un nivel con un número de nodos mayor que el de clases potencialmente presentes en la imagen. Un paso posterior de fusión de regiones, que una todas aquellas regiones en contacto que presentan un color similar, proporciona el número correcto de clases de forma no supervisada.

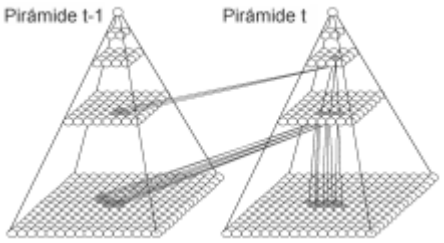


Figura 3: Segmentación espacio-temporal usando dos pirámides adaptativamente reenlazadas.

Esta técnica de segmentación jerárquica se puede emplear para llevar a cabo el seguimiento temporal de una determinada región, pero ello implicaría asociar correctamente cada una de las regiones en todas las pirámides mediante técnicas de *matching*. El principal problema de este sistema radica en su lentitud. Además, las secuencias reales plantean problemas pues sus fotogramas difícilmente se dividen en el mismo número de regiones, incluso en condiciones de iluminación relativamente estacionarias en el tiempo, y, adicionalmente, dichas clases pueden cambiar de forma e incluso de tamaño si los objetos se acercan o alejan de la cámara. Por todo ello, establecer correspondencia entre regiones de forma aislada siempre resulta difícil.

A continuación se propone un algoritmo de estabilización adaptativa capaz de trabajar con dos imágenes consecutivas. De esta manera, las regiones que aparecen en ambas imágenes son las mismas incluso cuando las condiciones de captura sufren variaciones, ya que ambas imágenes tienen la misma influencia en el proceso global de segmentación. Así, dadas dos pirámides asociadas a las imágenes capturadas en los instantes de tiempo t-1 y t, el algoritmo de segmentación propuesto enlaza cada nodo de la pirámide t con una región de forma irregular y color homogéneo en la base de la pirámide t y, además, con la región correspondiente en la pirámide t-1 (Fig. 3).

Los pasos del algoritmo de segmentación adaptativa ejecutado sobre dos pirámides consecutivas asociadas a los instantes de tiempo t-1 y t, son:

1. Sea $l=0$.

2. Se enlaza cada nodo hijo $i_l(t)$ del nivel l de la pirámide t con su nodo-padre de color más parecido, seleccionado entre el conjunto de nueve nodos padres situados inmediatamente sobre dicho nodo-hijo en el nivel $l+1$ de la pirámide t (Fig. 4.a).

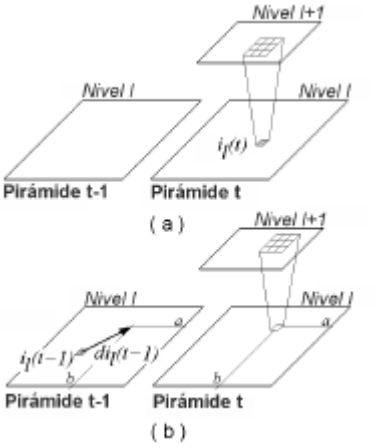


Figura 4: a) Enlazado entre el nodo $i_l(t)$ y su nodo-padre, y b) enlazado entre el nodo $i_l(t-1)$ y su nodo-padre.

3. Se enlaza cada nodo hijo $i_l(t-1)$ del nivel l de la pirámide t-1 con su nodo padre más similar seleccionado, en el nivel l+1 de la pirámide t, entre el conjunto que forman los nueve nodos ubicados sobre el nodo $i_l(t-1)+di_l(t-1)$ en la pirámide t (Fig. 4.b). El valor $di_l(t-1)$ refleja el desplazamiento en el nivel l de la región enlazada al nodo i_l entre los instantes de tiempo t-2 y t-1. Si no existe estimación de movimiento disponible, el valor de $di_l(t-1)$ se fija a cero.

4. Una vez que todos los nodos del nivel l de ambas pirámides han sido enlazados, se calcula el color asociado a cada nodo padre del nivel l+1 de la pirámide t como la media de todos sus nodos hijos, tanto en la pirámide t como en la t-1.

5. Si los cambios de color que sufren los nodos padres están por debajo de un umbral, el nivel l está estabilizado. Se hace l=l+1, volviendo al paso 2 hasta que el nivel l presente tantos nodos como clases, previas a la fusión, se deseen (generalmente 8x8 nodos). En caso contrario, se repiten los pasos 2, 3 y 4.

Una vez que el proceso de segmentación ha finalizado, el último nivel estabilizado de la pirámide tiene cada uno de sus nodos enlazado a dos regiones, una en la pirámide t y otra en la t-1. Si el proceso se lleva a cabo correctamente, ambas regiones son la misma.

2.2 Compresión

El proceso de segmentación espacio-temporal permite que cualquier región de la imagen sea identificada y rastreada a través de la secuencia mediante la estructura de enlaces entre pirámides consecutivas. El desplazamiento de las distintas regiones a lo largo del tiempo coincidirá con el que sufran sus centros de masa asociados, siendo seleccionadas como áreas de interés aquellas regiones que presenten una mayor movilidad. Así, el módulo de compresión identifica fácilmente las

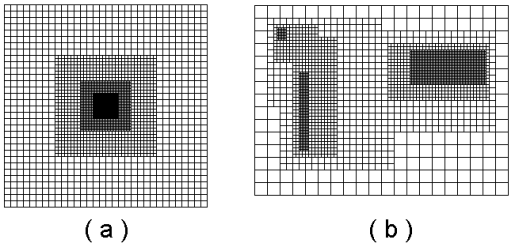


Figura 5: a) Geometría cartesiano-exponencial básica, y b) geometría multifoveal con fóveas desplazables y de tamaño adaptativo.

áreas de interés, y construye una imagen multi-resolución en la que dichas áreas de interés conservan su resolución original, mientras que la resolución de las regiones circundantes se reduce progresivamente para así minimizar el volumen de datos de la estructura a transmitir.

Para construir esta imagen multi-resolución de forma eficiente, y conseguir que sea posteriormente procesable por cualquier algoritmo de visión, se ha recurrido al empleo de las geometrías cartesiano-exponenciales, propuestas inicialmente por C. Bandera [7]. Esta geometría consiste en un grid simétrico que presenta una región central de alta resolución conocida como fóvea, rodeada de un conjunto de anillos de resolución decreciente (Fig. 5.a). La simetría de la geometría está optimizada para su almacenamiento en una estructura jerárquica denominada polígono foveal, que permite el posterior procesamiento de la imagen multiresolución. El módulo de procesado jerárquico no trabaja con dicha estructura, sino que sólo la construye antes de su posterior transmisión.

El principal problema que presenta la geometría propuesta en [7] es que presenta una única área de interés a máxima resolución, con forma cuadrada y centrada en el campo visual. Este problema ha sido resuelto mediante la geometría multifoveal [8], que no solo permite ubicar varias fóveas, sino que además ofrece la posibilidad de desplazar dichas foveas por la imagen, cambiando su tamaño para adaptarlas al de las áreas de interés, como se muestra en la Fig. 5.b.

La transformación de fotogramas de resolución uniforme en imágenes multifoveales, en las que sólo las regiones de interés permanecen a máxima resolución, implica una importante reducción del volumen de datos de la secuencia. Por ejemplo, un fotograma de 256x256 píxeles se puede reducir a sólo 3328 rexeles si se transforma en una imagen de fóvea centrada de 32x32 píxeles y tres anillos de resolución. Estos datos equivalen a una reducción del volumen de información de aproximadamente un 95 %. La reducción del volumen de datos es tanto mayor cuanto menor es el tamaño y número de las foveas. No obstante, cuando éste no puede disminuirse, aún se puede reducir información incrementando el número de anillos de la imagen.

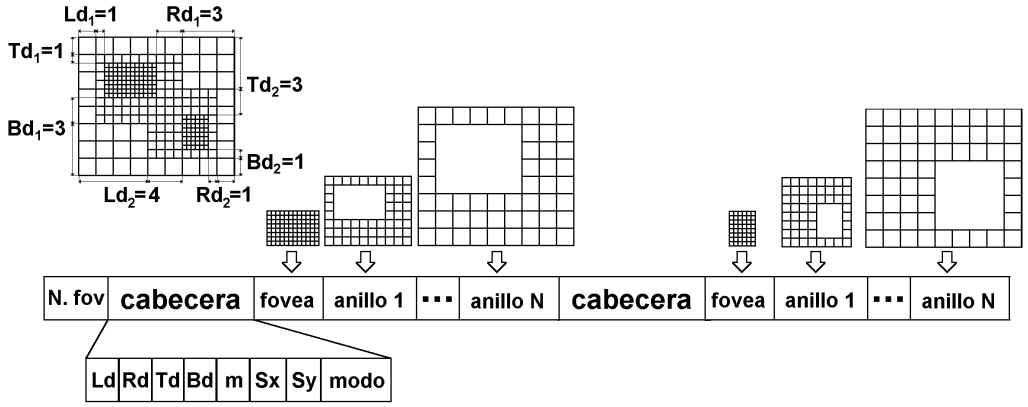


Figura 6: Contrucción del paquete a transmitir para una imagen multifoveal de dos fóveas.

2.3 Control transmisión-descompresión

En entornos en los que las cámaras permanecen estáticas y por ello, el fondo presenta poca variación en el tiempo, las regiones más cercanas a los objetos móviles son, junto con éstos, las áreas de la imagen que presentan cambios temporales más significativos. De acuerdo al esquema del epígrafe 2.2, los objetos móviles se representan a máxima resolución, existiendo una serie de anillos en torno a ellos que presentan unas resoluciones progresivamente menores.

Muchas aplicaciones requieren la recepción de fotogramas a una tasa estable. El uso de las imágenes multiresolución permite que dicha tasa de transmisión pueda aumentar considerablemente de ser necesario pero, aún así, podrían darse situaciones en las que el ancho de banda del canal impida la transmisión de una tasa estable de imágenes multi-resolución. En estos casos, las regiones que no son de interés de cada fotograma podrían dejar de transmitirse durante varios fotogramas. Tal y como se comentó anteriormente, se puede asignar la prioridad más alta a las regiones de mayor resolución y la más baja a las de menor resolución. Si el ancho de banda del canal obliga a no transmitir la totalidad de la imagen multi-resolución, este orden de prioridades es empleado por el algoritmo de compresión para transmitir las distintas regiones a distintas frecuencias. De esta forma, el sistema sólo transmitirá la imagen multi-resolución completa si las condiciones de retardo del canal lo permiten. Cuando la velocidad de transmisión disminuye, en función del tamaño de las distintas áreas de resolución se estima el volumen de datos de cada una y se determina qué áreas transmitir. El módulo de descompresión sabe qué regiones de la imagen multi-resolución recibe en todo instante de tiempo y combina la información recibida con datos almacenados de instantes anteriores para disponer de una imagen multi-resolución completa.

La Fig. 6 muestra una configuración multifoveal genérica y los parámetros que la caracterizan [8]. Dicha figura también muestra como se construye el paquete a transmitir por el canal. La variable *modo* determina el número de regiones de distinta resolución transmitidas por cada fóvea.

3 Experimentos y resultados

La eficiencia del sistema de compresión propuesto ha sido evaluada para secuencias de vídeo diferentes y canales con distintos retardos y anchos de banda, obteniéndose unos resultados especialmente satisfactorios especialmente en aquellas situaciones en las que el ancho de banda del canal de transmisión era extremadamente reducido. En este apartado se pueden comprobar las tres características más importantes del sistema, que son: i) la transmisión sin pérdidas de las áreas de interés de la escena, al tiempo que se mantiene un factor de compresión elevado; ii) la consecución de una tasa estable de imágenes en recepción; y iii) el correcto tratamiento y rastreo de múltiples objetos.

La Fig. 7 muestra una de las imágenes foveales obtenidas al analizar una secuencia de vídeo en la que existe un único objeto móvil. En todo fotograma, el móvil es correctamente detectado, enviándose por el canal a la máxima resolución disponible. Si se comparan la calidad del objeto móvil en recepción para el esquema propuesto y para el algoritmo MPEG-2, para un factor de compresión equivalente, como se muestra en las Figs. 7.b y 7.c, se puede apreciar la degeneración que producen los bloques cuadrados que usa el estándar MPEG. Por ello, aunque la pérdida de calidad en el fondo de la imagen es mayor si se emplea el algoritmo propuesto, éste puede resultar muy interesante para aplicaciones de visión activa, en las que sólo debe evitarse la pérdida de calidad en ciertas áreas de la imagen.

En la Fig. 8 se muestra un ejemplo típico de aplicaciones en las que la calidad del fondo no es de interés: la vigilancia de una determinada zona. Las Figs. 8.a y 8.c presentan dos de las imágenes uniformes de la secuencia de vídeo capturada en las que aparece un objeto móvil correctamente separado del fondo. El entorno en el que se lleva a cabo el experimento permite, además, evaluar la

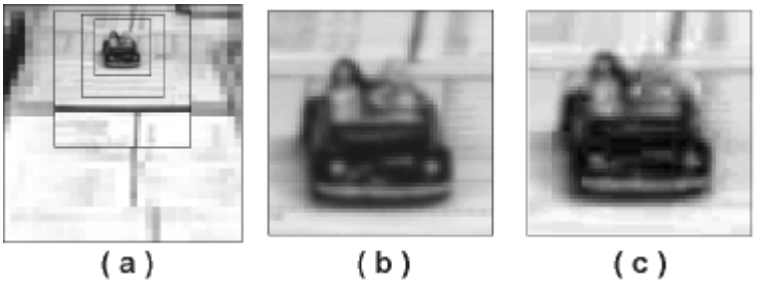


Figura 7: a) Imagen multi-resolución, b) región de interes transmitida usando el sistema propuesto, y c) aspecto de la región de interes al comprimir la secuencia con MPEG-2 (CBR, 0.4 Mbps).

robustez del método de segmentación, pues la escena incluye una escalera mecánica en movimiento, así como sombras y cambios de iluminación que podrían afectar a la correcta detección de los objetos. Las imágenes multi-resolución recibidas al otro extremo del canal se muestran en las Figs. 8.b y 8.d, y permiten confirmar que el móvil se recibe a alta resolución a lo largo de la secuencia a pesar de la complejidad de la misma. La Fig. 8.e presenta el tamaño de las diferentes regiones que forman la imagen multi-resolución para un fragmento de la secuencia de vídeo que consta de 358 fotogramas. El móvil aparece en la escena en torno al fotograma 240, pudiendo apreciarse que, antes de este instante, apenas existen cambios en los tamaños de las regiones, exceptuando pequeños picos debidos a detecciones espúreas. Igualmente, se aprecia como en el intervalo entre los fotogramas 240 y 325 el objeto es correctamente rastreado, existiendo cambios en el tamaño de la fóvea que se deben a variaciones del propio tamaño del objeto por la deformación intrínseca al movimiento de éste.

La Fig. 9 muestra el comportamiento de la secuencia presentada en la Fig. 8 bajo las condiciones de retardo impuestas extemamente que se muestran en la Fig. 9.a. Para conseguir una tasa constante de 10 imágenes por segundo, no siempre se puede transmitir la imagen multifoveal completa. La Fig. 9.b muestra qué regiones de la imagen multifoveal son transmitidas en cada instante de tiempo. Se puede apreciar cómo, generalmente, la imagen multifoveal es enviada completa, especialmente cuando no se detectan objetos móviles en la escena, el retardo no es excesivo o las fóveas son muy pequeñas. Cuando el móvil ha sido detectado, y sólo cuando las condiciones de retardo lo imponen, se dejan de transmitir algunas de las regiones que forman la imagen. Si se tiene en cuenta que se priorizan las regiones cambiantes, es inmediato constatar que se dejan de enviar las que no varían, lo cual no debe tener una importancia excesiva.

Finalmente, en la Fig. 10 se muestran imágenes multirresolución asociadas a fotogramas de distintas secuencias analizadas, en los que se observa la correcta detección de varios móviles.

4 Conclusiones

En este trabajo se propone un método para la compresión de una secuencia de vídeo en función

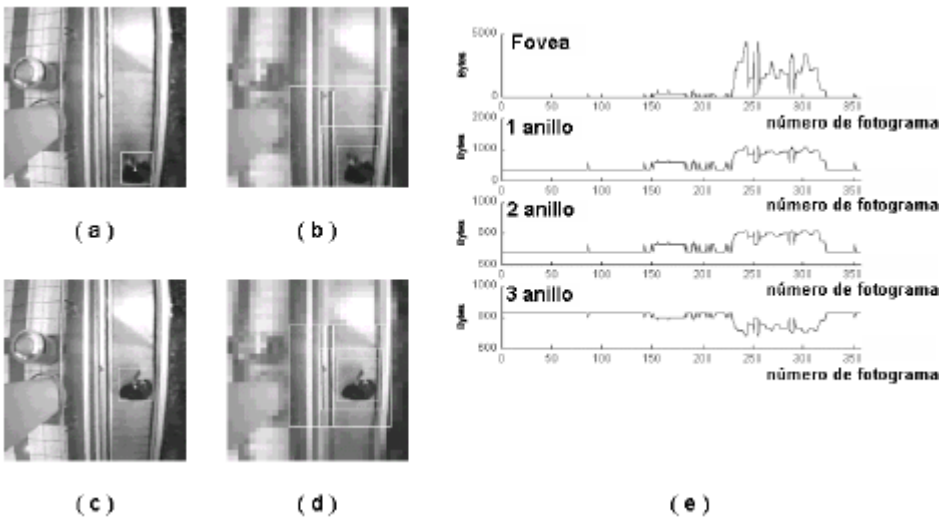


Figura 8: Datos asociados a una secuencia típica: a) Fotograma 242, b) imagen multifoveal recibida (fotograma 242), c) fotograma 266, d) imagen multifoveal recibida (fotograma 266), y e) tamaño de las diferentes regiones de las imágenes multifoveales durante toda la secuencia.

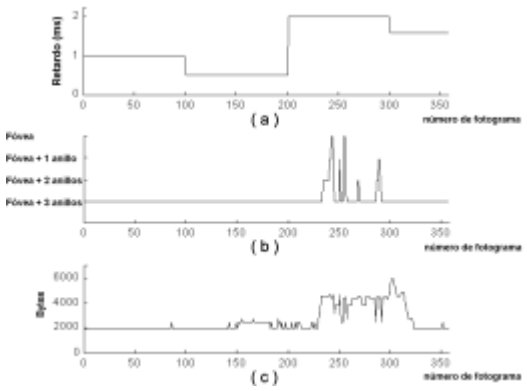


Figura 9: Funcionamiento del sistema: a) retardo forzado en el canal, b) esquema de transmisión, y c) flujo de datos a ser transmitido.

de las condiciones del canal y de la escena. Dicho sistema se caracteriza por su rapidez y eficiencia. El método es válido para aquellas aplicaciones en las que interesa que las áreas de interés de la escena sean transmitidas sin ningún tipo de pérdidas, mientras que la calidad del resto de la imagen puede degradarse. Este tipo de método puede ser empleado, por tanto, en aplicaciones de vigilancia, supervisión o detección de objetos.

Aunque, en principio, la movilidad de las regiones ha sido el factor determinante en el proceso de selección de las áreas de interés, el sistema permite incrementar fácilmente el conjunto de descriptores asociados a cada una de las regiones en que se divide la imagen. De esta forma se conseguiría que el sistema se adapte a la aplicación en particular.

La principal novedad del método descrito radica en la segmentación espacio-temporal en regiones que se aplica a la secuencia de vídeo. El método de segmentación propuesto permite que el seguimiento de cada objeto en el tiempo sea muy robusto, permitiendo trabajar con objetos deformables, lo que resultaría complicado para otros tipos de métodos basados en *matching* de regiones. Este método podría aplicarse también para detección de objetos en técnicas de compresión basadas en objetos como MPEG-4. También es de destacar el uso de imágenes de resolución no uniforme y la transmisión selectiva de regiones en función del estado del canal. Esto permite estabilizar la tasa de imágenes por segundo que se recibe a través de un canal de retardo variable.

Como trabajo futuro, se plantea la posibilidad de utilizar técnicas de compresión orientadas a objeto en combinación con el mecanismo de segmentación espacio-temporal propuesto. Igualmente, sería interesante establecer un conjunto de canales de diferente prioridad para las regiones resultantes de la segmentación en función de su velocidad. Cada uno de estos canales podría así trabajar con un mecanismo de control distinto en función de los

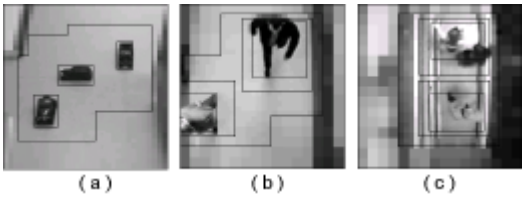


Figura 10: a-c)Imágenes multifoveales.

recursos disponibles para optimizar el proceso de recepción.

Agradecimientos

Este trabajo ha sido parcialmente financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT), en su proyecto nº. TIC098-0562.

Referencias

[1] ISO/IEC, "Information technology - generic coding of moving pictures and associated audio information: video", 13818-2, 1995

[2] ITU-T, "Video coding for low bit rate communication", recommendation H.263, version 2, 1998

[3] T. Ebrahimi y M. Kunt, "Object-based video coding", en Al Bovik (Ed.), Handbook of Image and Vídeo processing, pp. 585-595, Academic Press: San Diego, 2000.

[4] MPEG-4 Video Group, "Coding of audio-visual objects: vídeo", ISO/IEC JTC1 /SC29/WG11 N2202, 1998

[5] M. Kim y J. Kim, "Moving video object segmentation using statistical hypothesis testing", Electronics Letters, 36 (2), pp. 128-129, 2000.

[6] P. Burt, T. Hong y A. Rosenfeld, "Segmentation and estimation on image region properties through cooperative hierarchical computation", IEEE Trans. on Systems, Man and Cybernetics, 11 (12), pp. 802-809, 1981.

[7] C. Bandera y P. Scott, "Foveal machine vision systems", Proc. of the IEEE Int. Conf. on Systems, Man and Cybernetics, Cambridge-EEUU, pp. 596-599, 1989.

[8] P. Camacho, F. Arrebola y F. Sandoval, "Multirresolution sensors with adaptive structure", Proc. of the 24th Annual Conf. of the IEEE Industrial Electronics Society, 2, pp. 1230-1235, Aquisgram-Alemania, 1998.

Transmisión de flujos multimedia para el aprendizaje de idiomas utilizando técnicas de vídeo-on-demand

Carlos Turró Ribalta y Miguel Ferrando Bataller
Universidad Politécnica de Valencia.
Camino de Vera 27 46022 Valencia
E-mail: turro@cc.upv.es, mferrand@dcom.upv.es

Abstract. *La Universidad Politécnica de Valencia dispone en la actualidad de dos aulas dedicadas al autoaprendizaje de idiomas. Para ello se utiliza un sistema multimedia en red que permite al usuario escuchar en su ordenador las pistas de audio correspondientes a las lecturas programadas en los métodos didácticos. Se desea ampliar este sistema para incluir películas de vídeo subtituladas, así como ampliar el número de usuarios a los que se da servicio, pero la estructura actual que utiliza un servidor unicast de vídeo on demand exige grandes requisitos de ancho de banda en el servidor. Para resolver este problema se ha realizado un estudio de los sistemas de distribución de vídeo on demand basados en flujos multicast, así como una validación de los datos teóricos con los datos reales mediante un estudio estadístico y una simulación. Este estudio demuestra que la técnica conocida como stream tapping es la que proporciona el mejor rendimiento para el entorno considerado.*

1. Introducción

En los laboratorios de Idiomas de la Universidad Politécnica de Valencia los alumnos disponen de sistemas informáticos que les permiten seguir cursos multimedia, así como escuchar las pistas de audio de numerosos métodos de enseñanza de idiomas. Para ello disponen de equipos PC multimedia conectados en red utilizando enlaces Ethernet a 10 Mbps. Estos enlaces se interconectan con la troncal de red de la Universidad, con enlaces de Fast Ethernet (100 Mbps) y Gigabit Ethernet (1 Gbps). En esta troncal hay ubicado un servidor de media on demand conectado a un enlace Fast Ethernet, que es el encargado de servir dichas pistas de audio y ficheros multimedia a los estudiantes (Figura 1).

Este servidor lleva en producción desde Julio de 2000, y da servicio a 60 puestos divididos en dos aulas de 30 puestos cada una. El método de trasmisión utilizado es unicast, esto es, para cada una de las peticiones individuales de ficheros se genera un stream desde el servidor hasta cada uno de los clientes, independientemente de que algunos, muchos o todos los clientes soliciten el mismo fichero.

Una alternativa directa sería efectuar una trasmisión multicast, en la cual todos los clientes pueden acceder al mismo vídeo utilizando sólo un stream. Sin embargo, y dada la naturaleza de autoaprendizaje de las aulas, este hecho (que se solicite el mismo vídeo) sincronizado en el tiempo es bastante raro.

En la actualidad, al utilizarse sólo pistas de audio de reducido ancho de banda (64Kbps), esto no es un problema. Sin embargo existe el proyecto de ampliar este servicio en número de usuarios y, lo que es más importante, en el uso de películas de vídeo subtituladas. Esta ampliación, como veremos en el apartado 3 obliga a importantes requisitos en el equipamiento de servidor en el caso que se siga utilizando la trasmisión unicast.

Por ello, se ha realizado una evaluación de las distintas opciones que permitan resolver el problema planteado mediante un análisis de los datos disponibles en la actualidad. Esta solución viene de la mano de las tecnologías de vídeo on demand utilizando servicios multicast.

Las tecnologías de vídeo on demand utilizando servicios multicast permiten a los usuarios seleccionar un vídeo de una gran base de datos de los mismos manteniendo la eficiencia global del sistema. En los últimos años se han publicado numerosos artículos describiendo las diversas posibilidades de diseño.

Este artículo se organiza como sigue: En el apartado 2 examinamos las diversas técnicas de distribución de vídeo on demand utilizando servicios multicast publicadas en la literatura en los últimos años. En el apartado 3 se realiza un estudio estadístico de los datos reales con el fin de hallar una modelización válida de los mismos, conforme a los datos teóricos. En el apartado 4 se realiza una simulación directa sobre los datos prácticos para validar las previsiones anteriores. Finalmente en el apartado 5 se obtienen conclusiones a partir de los apartados anteriores.

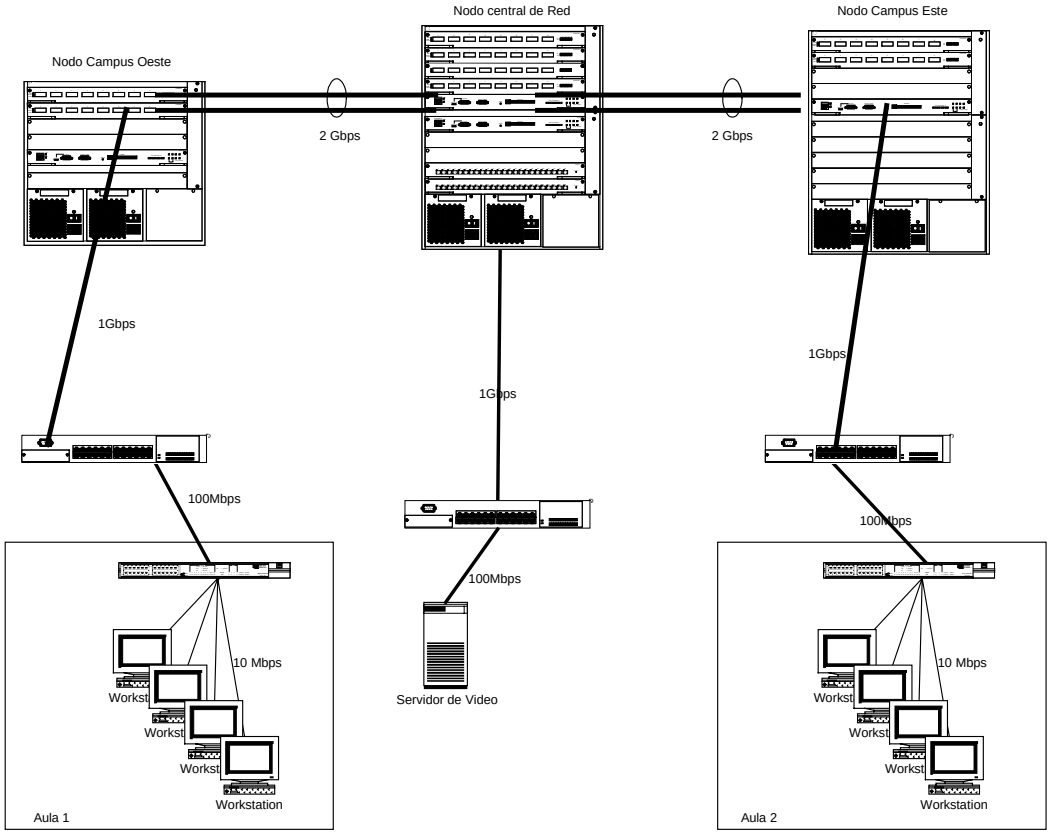


Figura 1 - Conexiones Multimedia

2. Técnicas de distribución de Vídeo on demand utilizando servicios multicast.

Batching

Batching [1] permite al servidor encolar las peticiones durante un cierto tiempo, y después dar servicio a todas ellas trasmitiendo por multicast el vídeo a todos los clientes. Si hay M peticiones en cola, el servidor ahorra M-1 veces el ancho de banda del vídeo.

Hay que notar que si sólo existe una petición en la cola, no se ahorra ningún ancho de banda. De la misma forma, cuanto mayor sea el tiempo de espera, mayor será el ahorro de ancho de banda.

Una estrategia similar es Delayed Batching [2]. En este caso, cuando llega la primera petición al servidor, la hace esperar un cierto periodo de tiempo, que se calcula en función del ritmo de llegada de los clientes. En Staggered Broadcasting o Broadcast regular [3], se efectúa un broadcast regular del vídeo en intervalos fijos de tiempo, con lo que la demora es dependiente del momento en que llegue la petición.

El problema general de las técnicas de batching es que introducen una latencia bastante elevada desde la petición del vídeo hasta el servicio. De hecho lo que se garantiza es que la latencia es mayor que cero en todos los casos. También hay que notar que estas técnicas sólo son útiles con los vídeos más populares, ya que en el caso de los vídeos de uso esporádico se introduce una latencia segura a cambio de ninguna mejora en ancho de banda.

Broadcast segmentado

Las técnicas de broadcast segmentado incluyen las siguientes: Pyramid Broadcasting [4] [5], Skyscraper broadcast [6] [7] y Harmonic Broadcasting [8]. Estas técnicas se basan en la división de los vídeos en diferentes trozos, a continuación los vídeos se trasmiten en streams, de forma que el stream i incluye los trozos de índice i de todos los vídeos. Los trozos no son del mismo tamaño sino que siguen una progresión de menor a mayor, de forma que se disminuye el tiempo de espera.

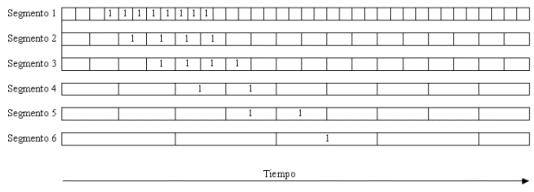


Figura 2 - Técnicas Proactivas

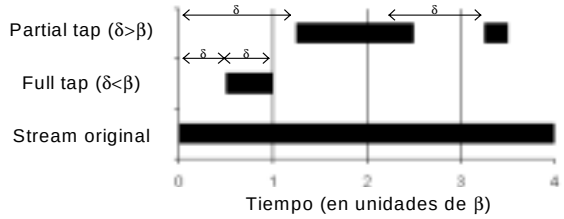


Figura 3 - Stream tapping

Para recibir los vídeos, los clientes deben de sintonizar simultáneamente más de un stream (típicamente 2) y almacenar en un buffer local la parte del vídeo que han recibido antes de su reproducción. Esto obliga a ciertos requisitos de ancho de banda de recepción en el cliente y de espacio de buffer que dependen de la progresión de tamaños utilizada, que es distinta en cada una de las distintas técnicas.

En general podemos concluir que las técnicas de Broadcast segmentado reducen la latencia frente a las técnicas de batching, a costa de aumentar los requisitos de buffer y ancho de banda en la parte cliente. Adicionalmente hay que hacer notar que estas técnicas siguen siendo válidas sólo para el conjunto de vídeos más populares, ya que en otro caso se está reservando ancho de banda para vídeos que nadie solicita. Por ello, una solución completa debe incluir además un método de servicio unicast para los vídeos más raramente accedidos.

Piggybacking

En piggybacking [9] [4], se modifica la velocidad de reproducción de los vídeos de los clientes en un +-5%, lo que es supuestamente indetectable para los observadores, para unir pares de streams cercanos en el tiempo.

Esta técnica tiene dos inconvenientes fundamentales: El primero es que es necesario bastante tiempo para unir efectivamente dos streams, y el segundo es que el servidor de vídeo debe de ir cambiando de velocidad de trasmisión en respuesta a las peticiones de los clientes.

Multicast asíncrono

La técnica de multicast asíncrono [11] permite a los clientes unirse a un stream multicast solicitado por otro cliente después de comenzada la transmisión. Para conseguir esto se dividen los vídeo es una serie de segmentos de longitud L, que se envían a velocidad N veces superior a la velocidad de reproducción. En este caso, la transmisión sólo dura un tiempo L/N, con lo que se pueden enviar N segmentos en el tiempo de uno. Los clientes reciben el stream y almacenan en un buffer los segmentos de vídeo, siempre que lleguen un tiempo (N-1)L más tarde, como máximo.

Esta técnica requiere elevados anchos de banda, y un buffer bastante importante, por ejemplo, si N=3 y L=6 [Kal96] el buffer del ciente debe de almacenar 18 minutos de datos para alcanzar vídeos que comenzaron sólo hace 12 minutos.

Stream tapping/patching

La idea fundamental que hay detrás de las técnicas de stream tapping [12] [13] es que cada cliente no está restringido a su stream, sino que puede recibir información de streams de otros clientes. Si hay streams del vídeo solicitado que ha solicitado previamente algún otro cliente, se solicita sólo un stream por la diferencia entre ambos vídeos, y se almacena en buffer. En este caso, cada cliente debe disponer de un ancho de banda doble del ancho de banda del vídeo transmitido.

Para tener un límite máximo de ancho de banda, cuando la petición llega n<B segundos más tarde del comienzo de un stream original, se trasmite un full tap stream de n segundos. Si la petición llega más tarde de B, se transmite un partial tap stream de B segundos, y a continuación se reciben por el partial tap stream los picos que no puedan ser cubiertos desde el stream original (Figura 2).

Esta técnica tiene como puntos fuertes, por una parte, el que no requiere una clasificación a priori de los vídeos, ya que se puede aplicar a ellos independientemente de su popularidad. Por otra parte, es sencilla de implementar, y sólo requiere 2 veces el ancho de banda de vídeo en el cliente, y un buffer limitado a B segundos.

Clasificación de las técnicas

Siguiendo a [13] podemos clasificar en la Tabla 1 las diversas técnicas con valores representativos de un entorno real. Aquí podemos apreciar que, desde el punto de vista del cliente, una técnica sin latencia es mejor que una con latencia, que las técnicas proactivas requieren un conocimiento previo de la popularidad de los vídeos, mientras que las reactivas no, y que el número de streams a recibir en el cliente y el tamaño del buffer son parámetros de diseño fundamentales en cualquier aplicación práctica.

Sistema	Latencia	Tipo	Ancho de banda en el cliente (streams)	Tamaño buffer (minutos)
Unicast	No	-	1	0
Batching	Sí	Reactiva	1	0
Delayed Batching	Sí	Reactiva	1	0
Staggered Broadcast	Sí	Proactiva	1	0
Pyramid Broadcast	Algo	Proactiva	2-3	30
Skyscraper Broadcast	Algo	Proactiva	2	5-40
Piggybacking	No	Reactiva	1	0
Multicast Asincrono	No	Reactiva	3 o más	10-30
Stream Tapping	No	Reactiva	2	10-30

Tabla 1 - Comparación de técnicas

3. Estudio estadístico de los datos

Se dispone de datos correspondientes al periodo de julio a diciembre de 2000, con un total de 19920 ficheros servidos. Estos ficheros son, en general, pistas de audio con un ancho de banda de 64 Kbps y algunos vídeos de un ancho de banda de 150 Kbps con una duración media de 3 minutos. Estos accesos se distribuyen según se indica en la Figura 4.

Los ficheros no son accedidos de forma uniforme. De forma consistente con lo que se muestra en [14] y [15], se comprueba que la distribución de tráfico se ajusta a procesos fractales o autosimilares [16]. Sin embargo, y en intervalos cortos de tiempo, se verifica que el ritmo de llegada de peticiones se ajusta a una distribución de Poisson, con lo que se puede, en cierto modo, modelizar un intervalo no excesivamente largo de tiempo como una sucesión de distribuciones de Poisson con velocidades fijas en cada intervalo [14].

Los ficheros no son accedidos de forma uniforme. Numerosas evidencias sobre la distribución de los accesos a bases de datos dan como resultado que se sigue una distribución de Zipf modificada, en la que las peticiones del vídeo n en orden de popularidad son proporcionales a 1/(n^α). Esto se aplica tanto a de películas de vídeo [20] como a los accesos a páginas en servidores web y servidores proxy [18] [17] y a otros muchos campos, como la frecuencia de ocurrencia de palabras en texto, campo original del estudio de Zipf, como la bioingeniería, la economía y todo tipo de ciencias aplicadas. Un resumen de los campos en los que se aplica la ley de Zipf se puede consultar en [19].

Se ha llevado a cabo una simulación sobre los datos, y se obtiene que éstos se ajustan con una distribución de Zipf modificada con parámetro alfa entre 0.6 y 0.9, como se observa en la siguiente tabla de resultados (Tabla 2):

Tabla 2 - Parámetro de Zipf

Mes	*Parámetro α*
Julio	0,60
Septiembre	0,61
Octubre	0,69
Noviembre	0,81
Diciembre	0,89
Global	0,59

Sin embargo hay que hacer notar que, aunque los datos por mes y en su conjunto siguen esta distribución, los ficheros más solicitados cada mes y cada día son distintos (tabla 3), factor éste importante, ya que dificulta la predicción de los vídeos más solicitados en el futuro y el uso de técnicas proactivas.

Finalmente, y con respecto a la superposición de peticiones, hay que hacer notar que depende en gran medida de la longitud temporal de los ficheros, ya que a ficheros más largos corresponde un factor de superposición mayor. En cualquier caso, y dado que en la actualidad se sirven por unicast, este valor experimental es un límite superior de ancho de banda en el servidor frente al que vamos a comparar las estrategias de distribución multicast de vídeo on demand.

En las figuras 5 y 6 se muestran los streams simultáneos a servir en media y pico para longitudes de fichero de 5, 30 y 120 minutos.

Con estos datos se observa que, en pico, si se quieren trasmitir películas de 120 minutos codificadas en MPEG-1 a 1,5 Mbps, tendremos un flujo de streams simultáneos de más de 500, con una ocupación de ancho de banda en el server de 750 Mbps de flujos unicast. Sin embargo, vamos a comprobar cómo utilizando técnicas de multicast podemos limitar el ancho de banda a valores muy inferiores.

Orden	*Julio*	*Septiembre*	*Octubre*	*Noviembre*	*Diciembre*	*Total*
1	55001000.rm	B0001003.rm	77001000.rm	55013000.rm	C1016000.rm	55013000.rm
2	55007000.rm	46010000.rm	v0009.rm	69001000.rm	07003003.rm	77001000.rm
3	55011000.rm	v0002.rm	07001000.rm	C1008000.rm	77006001.rm	v0009.rm
4	55002000.rm	B0000000.rm	77002000.rm	69002000.rm	07001004.rm	07001000.rm
5	55004000.rm	77001000.rm	77001001.rm	70001000.rm	C0008000.rm	69001000.rm

Tabla 2 - Ficheros más solicitados

A continuación, una vez verificada la validez de la distribución Zipf para el modelado de los datos, y obtenido un perfil horario de los accesos vamos a extrapolar los datos para evaluar el aumento del número de usuarios del sistema, aumento que sería lineal con técnicas de unicast, pero que no va a ser así con técnicas de multicast.

4. Aplicación de las estrategias a los datos

Una vez modelados los datos, vamos a aplicar las técnicas mostradas en el apartado 3 a los datos reales, mediante una simulación con MATLAB según los algoritmos propuestos en la literatura, y a comparar estos resultados con los valores teóricos para simulaciones con ritmo de acceso Poisson y distribución de ficheros Zipf.

En el caso de los modelos de broadcast segmentado, constatamos la dificultad de seleccionar a priori el conjunto de "vídeos más populares" que forman parte de los streams proactivos. A expensas de un estudio más detallado se han simulado tres estrategias: "los N vídeos más vistos el anterior día laborable" (predicción 1), "el vídeo más visto cada día laborable durante N días" (predicción 2) y un escalado de vídeos más vistos los días laborables. Con el fin de tener un valor de referencia, se ha tomado el "valor óptimo" que corresponde a toma cada día los N vídeos más vistos ese mismo día, valor que corresponde a la mejor predicción posible para un día completo. Para la simulación se han tomado como valores de muestra N=10 y longitudes de vídeo de 60 minutos, y los resultados de pico se muestran en la figura 6.

En el caso de los sistemas reactivos se realiza una simulación directa sobre los datos para un sistema de stream tapping como el propuesto por Carter y Long [Car97]. En este caso, el parámetro de ajuste es β, el tiempo máximo de buffer en el cliente. En la gráfica siguiente se muestran los resultados de pico para valores de β de 1 y 5 minutos. Los resultados de esta simulación se muestran en la figura 7.

Finalmente, para las técnicas de batching. Realizamos una simulación directa sobre los datos con valores de espera β de 30 segundos, 1 minuto y 2 minutos, ya que en esta técnica la espera media del cliente es β/2, y no se considera procedente alargarla más de estos valores. Los resultados de esta simulación para valores de pico se muestran en la figura 8.

5. Conclusiones

De los resultados obtenidos en el estudio estadístico de los datos, se puede constatar primeramente respecto de los ficheros servidos que la distribución de los mismos no es estable en el tiempo, sino que varía continuamente. Este resultado se adecua claramente a un entorno docente en el cual los alumnos acceden progresivamente a los ficheros según van superando las distintas partes de una materia.

Por ello, la predicción de ficheros más vistos, como se ha observado en el apartado 4, no produce grandes resultados en cuanto a ahorro de ancho de banda. De hecho se consume más ancho de banda en servidor y en cliente utilizando las técnicas proactivas que en el caso unicast.

La solución pasa pues, por un conocimiento previo de los ficheros que van a ser utilizados en cada sesión, conocimiento que además debe de ser refrescado con periodicidades menores al día, ya que incluso en el caso de una predicción diaria perfecta los resultados tampoco son buenos, debido sin duda al escaso número de clientes (Figura.6).

Por el contrario, las técnicas reactivas muestran un ahorro claro de ancho de banda desde el principio. Utilizando batching se obtiene un 75% de ahorro de ancho de banda (de pico) para un tiempo de espera de 2 minutos, y utilizando stream tapping se obtiene una ganancia similar con un buffer de cliente de 5 minutos. Esto, para un entorno MPEG-1 a 1,5 Mbps, da una cifra de 56,25 Mbytes, valor perfectamente asumible para PCs de bajo coste, incluso para equipos tipo set-top-box.

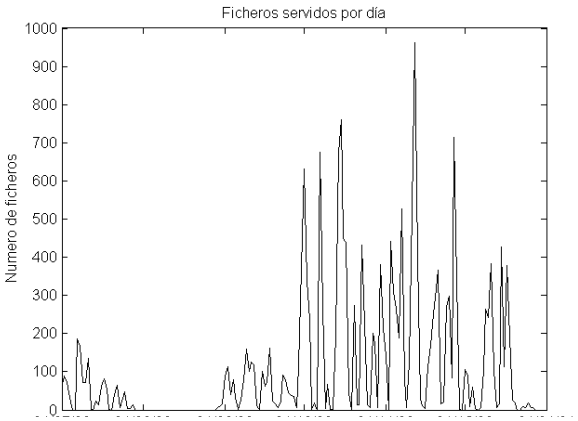


Figura 4 - Ficheros servidos

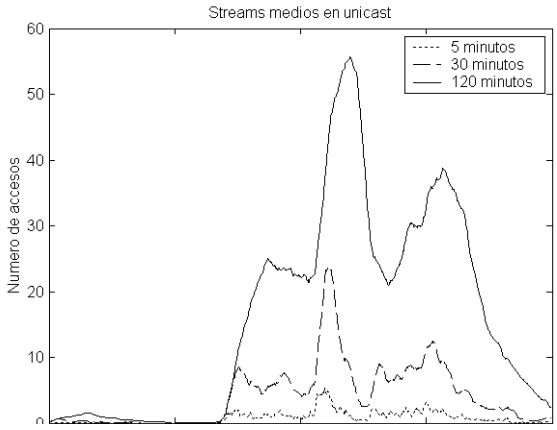


Figura 5 - Streams medios

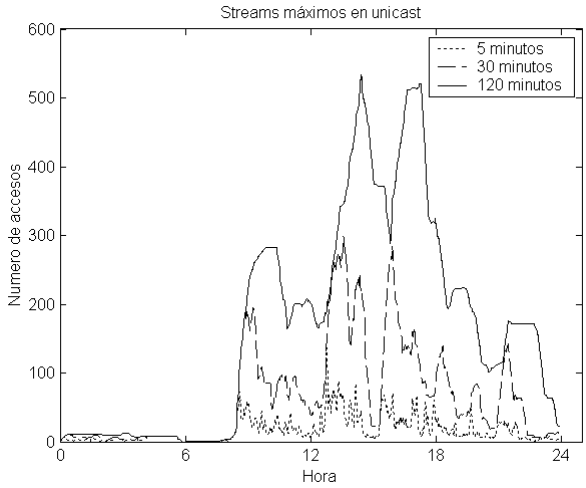


Figura 6 - Streams máximos

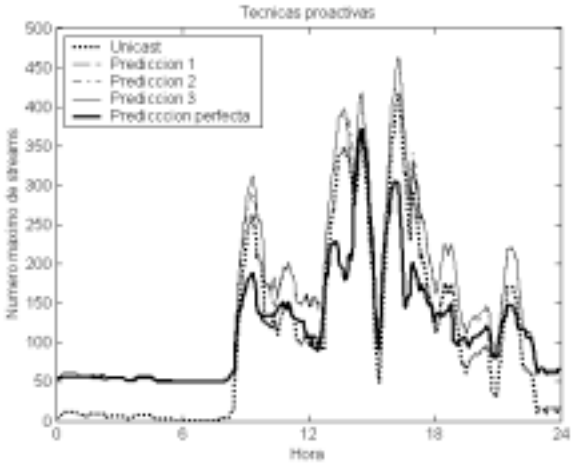


Figura 7 - Técnicas Proactivas

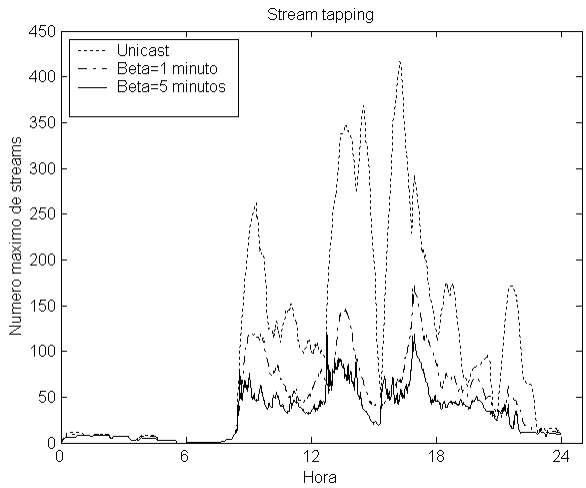


Figura 7 - Stream tapping

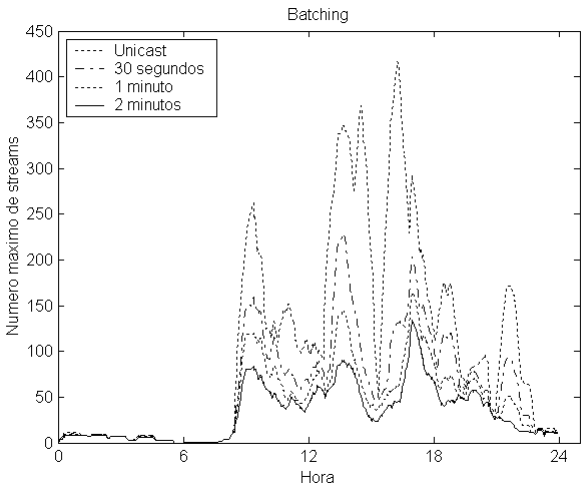


Figura 8 - Batching

Referencias

[1] A. Dan, D. Sitaram, y P. Shahabuddin. *Scheduling Policies for an On-Demand Video Server with Batching*. Proceedings of ACM Multimedia'94, pp 391--398, Oct. 1994

[2] *Shachnai, H. y Yu, P. S. (1995). The role of wait tolerance in effective batching: A paradigm for multimedia scheduling schemes.Technical Report RC20038 IBM Research division T.J. Watson Research Center*

[3] K. Almeroth y M. Ammar. *The Use of Multicast Delivery to Provide a Scalable and Interactive Video on -Demand Service*. IEEE Journal on Selected Areas in Communications, 14(6):1110--1122, Agosto 1996.

[4] C. C. Aggarwal, J. L. Wolf, and P. S. Yu. *A permutation-based pyramid broadcasting scheme for video-on-demand systems*. In Proc. of the IEEE Int'l Conf. on Multimedia Systems '96, Hiroshima, Japan, Junio 1996.

[5] S. Viswanathan y T. Imielinski, *Metropolitan area video-on-demand service using pyramid broadcasting*. Multimedia Systems, 4(4):197--208, 1996.

[6] K. Hua y S. Sheu, "*Skyscraper broadcasting: A new broadcasting scheme for metropolitan video-on-demand systems*," Proc. ACM SIGCOMM, Sept. 1997.

[7] D. Eager y M. Vernon, "*Dynamic skyscraper broadcasts for video-on-demand*," Proc. Inter. Workshop on Advances in Multimedia Information Systems, , Sept. 1998.

[8] L. Juhn y L. Tseng, *Harmonic broadcasting for video-on-demand service*. IEEE Transactions on Broadcasting, 43(3):268--271, Sept. 1997.

[9] L. Golubchik, J. Lui, y R. Muntz. *Adaptive Piggybacking: A Novel Technique for Data Sharing in Video On Demand Storage Servers*. ACM Multimedia Systems Journal, 4(3):140--155, 1996.

[10] C. Aggarwal, J. Wolf, y P. S. Yu, "*On optimal piggybacking merging policies for video-on-demand systems*," Performance Evaluation Review, vol. 24, pp. 200--209, Mayo 1996.

[11] Heekyoung Woo y Chong-Kwon Kim. *Multicast scheduling for VOD services*. Multimedia Tools and Applications, 2(2):157--171, Marzo 1996.

[12] S. W. Carter y D. D. E. Long. *Improving video-on-demand server efficiency through stream tapping*. Proceedings of the 6-th International Conference on Computer Communication and Networks (ICCCN '97), 200-207, 1997.

[13] S. W. Carter y D. D. E. Long, "*Improving Video-on Demand Server Efficiency Through Stream Tapping*", Proc. ICCCN'97, Las Vegas, NV, Sept. 1997.

[14] Vern Paxson y Sally Floyd. *Wide area traffic: The failure of Poisson modeling*. Proceedings of the ACM SIGCOMM Conference on Communications, Architectures, Protocols and Applications, pages 257--268, London, England UK, Agosto 1994.

[15] W. Willinger, M. Taqqu, R. Sherman, y D. Wilson. *Self-similarity through highvariability: statistical analysis of Ethernet LAN traffic at the source level*. ACM Sigcomm '95, pages 100--113, 1995.

[16] W. Leland, M. Taqqu, W. Willinger, y D. Wilson, "*On the Self-Similar Nature of Ethernet Traffic (Extended Version)*", IEEE/ACM Transactions on Networking, 2(1), pp. 1-15, Febrero 1994.

[17] L. Breaslau, P. Cao, L. Pan, G. Phillips, y S. Shenker. *Web caching and Zipf-like distributions: Evidence and implications*. IEEE INFOCOM'99, pages 126 -- 134, Marzo. 1999.

[18] ME Crovella, A Bestavros (1997), "Self-similarity in world wide web traffic: evidence and possible causes", IEEE/ACM Transactions on Networking, 5(6):835-846.

[19] “Zipf’s Law” Rockefeller University, N.Y. http://linkage.rockefeller.edu/wli/zipf/

[20] Video Store Magazine, 13 diciembre 1992

Desarrollo de Servicios Avanzados de Voz sobre redes de Paquetes

Mª Carmen Bartolomé[1], Raquel Panadero[1], Carlos García[1], José Ignacio Moreno[1], David Fernández[2]

[1]Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid
Avda. de la Universidad 30
28911 Leganés (MADRID)

E-mail: {cbc, rpa, cgarcia, jmoreno}@it.uc3m.es

[2]Departamento de Ingeniería Telemática
ETSI Telecomunicación
Universidad Politécnica de Madrid
Ciudad universitaria sn
28040 Madrid

e-mail: david@dit.upm.es

Abstract. *During last years, voice transport over packet-switched networks has experimented great growth due to the development of standards as well as to the appearance of products based on IP technology. In this scope, this article will introduce the different technologies of Voice over IP (VoIP) transport emphasizing the H.323 protocol because of its popularity. Later, PISCIS project is described and we will present our group experience implementing new services based on a key component of the H.323 protocol, the gatekeeper, based on free licence software OpenH323 which have been fully tested in a pilot H.323 network provided by PISCIS.*

1 Introducción

La transmisión de tráfico de voz sobre redes de paquetes ha experimentado grandes avances en los últimos años tanto por el desarrollo de estándares como por la aparición de productos basados en tecnología IP. A medio plazo esta tecnología se vislumbra prometedora motivada por su utilización en redes móviles de tecnología UMTS. La evolución de la release 99 ***[1]*** hacia la release 4 y 5 ***[2]*** incluye el salto de tecnología de transmisión de voz tradicional hacia tecnologías de transmisión de Voz sobre IP (VoIP) basadas en el despliegue de una única red de paquetes integradora de todos los servicios.

En este ámbito, el presente artículo introduce las distintas tecnologías de transmisión de VoIP actualmente estandarizadas, así como la experiencia del grupo en el desarrollo de servicios para el caso de utilizar el protocolo H.323 ***[3]***, protocolo que por motivos históricos, la primera versión apareció en el año 1996, presenta un mayor numero de productos en el mercado. Este trabajo ha sido desarrollado dentro del proyecto de investigación *PISCIS:"Plataforma Piloto de Servicios de Comunicaciones sobre Internet de Servicios Integrados*". ***[4]***

La integración de servicios en una única red de paquetes permite el desarrollo de nuevos servicios que permiten integrar aspectos que anteriormente estaban segmentados por motivos de la tecnología de red sobre la que se basaban. Ejemplos de estos servicios son servicios de localización, grupos de salto, transmisión de vídeo, integración de buzón de voz y correo electrónico, fax, autenticación control de acceso y seguridad

Durante los primeros pasos de esta tecnología se fijó como aspecto diferenciador, respecto a la tecnología clásica de transmisión basada en circuitos, los aspectos relacionados con el coste, especialmente en entornos corporativos donde existe una red de datos, típicamente propiedad de la propia organización y una red telefónica, normalmente contratada a uno o varios operadores con facilidades de grupo cerrado de usuarios, numeración reducida, etc. Sin embargo la reducción de los precios del mercado motivada por la aparición de la competencia en el mismo, junto con la falta de soluciones que garanticen de un modo eficiente calidad para la transmisión sobre redes de datos, han provocado que esta tecnología no haya tenido el éxito esperado por las primeras previsiones. Hasta ahora existen distintos ejemplos tanto de operadores que han apostado por esta tecnología para ofrecer el servicio de voz como de organizaciones privadas. Sin embargo, en ambos casos, estas entidades han debido realizar un sobredimensionado de la red para garantizar aspectos de QoS. Por otro lado, existe una falta de soluciones técnicas para permitir el encaminamiento de tráfico telefónico a través de la pasarela más adecuada (menor coste) de un modo dinámico y adaptativo. Las previsiones de despliegue según un estudio de Lucent-Dataquest muestran que existirá una importante demanda provocada especialmente por entornos corporativos (figura 1).

Los escenarios de aplicación de VoIP permiten la comunicación de usuarios de tres modos distintos en función del terminal utilizado:

?? PC-PC: en el caso de utilizar terminales tipo PC o equivalente interconectados mediante una red de datos

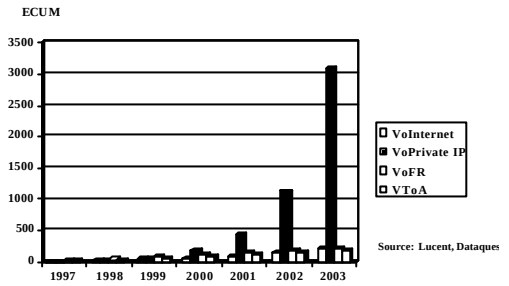


Figura 1: Previsiones de evolución de VoPKT en Europa

?? Teléfono-Teléfono: en el caso de utilizar terminales tradicionales. En este caso, para la interconexión de los mismos se utiliza un backbone IP y dispositivos que permiten interconectar las centralitas tradicionales con la red IP (Gateways).

?? Teléfono-PC: en el caso de interconectar usuarios conectados a redes de datos y redes telefónicas tradicionales.

Estos entornos se diferencian en la complejidad, el coste y el equipamiento necesario. La más fácil y barata es la comunicación PC-PC. Cada PC necesita una tarjeta de sonido, un micrófono y un altavoz. Si utilizamos teléfonos tradicionales, la complejidad y el coste son mayores porque se necesita una gateway, además de la harmonización de direcciones IP-E-164. La tercera opción es la más compleja. Se necesita una gateway y además un software compatible con ella.

En todos los casos, la necesidad de transmisión en tiempo real, obliga a la utilización de protocolos no fiables (UDP/IP), con lo que es necesario garantizar aspectos de QoS (retardo, ancho de banda, etc.) bien mediante el sobredimensionado o bien mediante técnicas basadas en los modelos de DiffServ ***[5]*** e IntServ ***[6]*** sobre los que se está trabajando actualmente.

En este ámbito, la siguiente sección muestra los distintos protocolos de señalización utilizados en VoIP centrándose en H.323 por las razones antes mencionadas. A continuación, se describe el proyecto PISCIS, junto con la plataforma piloto del mismo y los servicios desarrollados en esta plataforma indicando los entornos de desarrollo sobre los que se han trabajado.

2 Protocolos de señalización en VoIP

Diversos organismos trabajan en la normalización del servicio de VoIP. Los más importantes son el ITU-T y el IETF. El ITU-T fue pionero en este sentido, produciendo en 1996 la primera versión de la recomendación H.323, que es considerada un paraguas de normas que aglutinan distintos mecanismos de señalización para la transmisión de tráfico multimedia sobre redes de paquetes (H.225.0, H.245, T.120, ...). El IETF a través de grupo MMUSIC, estandarizó tres años más tarde otro protocolo de señalización denominado SIP (Session Initial Protocol) ***[7]*** que actualmente está siendo debatida su adopción como estándar para la transmisión de voz en redes UMTS (release 5). pensado específicamente para VoIP. La estructura de un escenario SIP es prácticamente la misma en cuanto a los elementos funcionales a la ofrecida por H.323. La principal diferencia con es su simplicidad. SIP hace en una transacción lo que H.323 hacía mediante cuatro o cinco intercambios de mensajes, cada uno de ellos especificado en un documento distinto del ITU-T. Por esta razón tiene un tiempo de establecimiento menor.

Junto a estas dos soluciones, existe una tercera denominada MGCP, MEGACO o H.248. Esta nueva norma establece el protocolo de señalización entre una pasarela o Media Gateway en terminología MGCP y un servidor de llamadas o Media Gateway Controler. La norma originalmente propuesta por el IETF en 1998 (MGCP) ***[8]*** y que integraba soluciones de distintos fabricantes, ha evolucionado hacia el protocolo MEGACO ***[9]***, definida por el IETF en septiembre de 1999 y adoptada por el ITU-T en la norma H.248 en Febrero de 2000 ***[10]***.

2.1 H.323

Los fabricantes de productos de comunicación se han visto atraídos por esta tecnología desde 1996 cuando se generó la H.323v1. Empresas como Lucent Technologies, Cisco, Teldat, NetSpeak, y NetPhone, han introducido productos VoIP basados en este estándar. El líder del mercado, Microsoft, tiene el producto software más utilizado para el soporte de VoIP (NetMeeting), ya que viene integrado en el paquete de aplicaciones de Windows. Quizás por esta razón sea el estándar para VoIP más utilizado.

La recomendación H.323 del ITU-T define los componentes, procedimientos y protocolos para ofrecer comunicaciones multimedia en redes de paquetes sin QoS garantizada. Ofrece servicios de transporte fiable y no fiable de datos, y no fiable de voz y vídeo, es independiente de la topología de red y ofrece interoperabilidad con terminales de la serie H (H.320, H.322, ...) a través de pasarelas o gateways.

Un sistema H.323 está formado por los siguientes elementos: terminales, gateways, gatekeepers, y MCUs (Unidad de control Multipunto). En los párrafos siguientes procedemos a explicar cada uno de estos componentes con detalle.

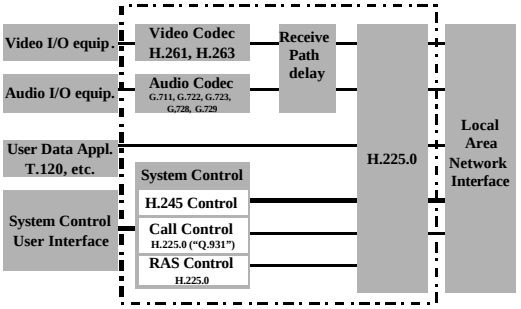


Figura 2: Estructura de un Terminal H.323

El ***terminal*** **H.323** proporciona comunicación bidireccional en tiempo real con otro terminal, gateway o MCU. La información intercambiada consiste en: control, indicaciones, audio, vídeo y datos. Los terminales H.323 deben al menos soportar la transmisión de audio, si bien existen otras combinaciones posibles como: audio más datos, audio más video y audio más video más datos. La estructura de un terminal H.323 se muestra en la figura 2.

Todos los terminales deben tener una unidad de control del sistema, una capa H.225.0, una interfaz de red y un codificador de audio. El codificador de vídeo y las aplicaciones de datos son opcionales.

El ***gateway*** permite la comunicación entre terminales H.323 y terminales ITU conectados a otras redes. Soporta conversión de formatos de transmisión –transcodificación- (por ejemplo H.225.0 a/desde H.221) y procedimientos de comunicación (a/desde H.245 a H.242) además de establecimiento y liberación de llamada en ambos lados. El gateway no es necesario para establecer una comunicación entre dos terminales H.323.

El ***gatekeeper*** soporta traslación de direcciones, control de acceso, control de ancho de banda y gestión de zonas. Es opcional en el sistema H323 pero si existe es obligatorio utilizarlo. Este componente representa la pieza clave de la arquitectura para el desarrollo de servicios y se tratará en detalle en el apartado 4.

La ***MCU*** soporta la comunicación multipunto. Se compone de un controlador multipunto (*MC)* el cual gestiona el control de las conexiones (obligatorio) y un procesador multipunto (*MP)* que gestiona la mezcla y conmutación de audio y vídeo. Este último es opcional ya que esta funcionalidad puede residir en cada uno de los terminales,.

Los protocolos de señalización más importantes utilizados en el seno de la H.323 son:

?? **H.225.0** ***[11]***: Define la señalización entre terminales/gateways y gatekeeper (RAS). También define la señalización para establecimiento y liberación de la llamada (Setup, Alerting,...) que va por el canal de señalización de llamada. En este caso se utiliza un subconjunto de las funciones proporcionadas por la Q.931.

?? **H.245** ***[12]***: Señalización de control extremo a extremo. La función principal es el intercambio de capacidades entre los terminales H.323 previa a la transmisión de información.

?? **H.235** ***[13]***: trata sobre la seguridad en la comunicación incluyendo autentificación, autorización, control de llamada seguro y privacidad de los canales de voz

?? **H.450** ***[14]***: señalización para el control de todos los servicios suplementarios (desvío de llamada, llamada en espera,...)

La arquitectura de protocolos utilizada por H.323 es la mostrada en la figura 3.

Una llamada H.323 puede dividirse en tres fases en relación con los protocolos de señalización que intervienen en la misma:

- **RAS (Registro Autenticación y Estado)**: Cuando un terminal quiere hacer una llamada, pide permiso al gatekeeper mandando un paquete ARQ (Admission Request). Este mensaje contiene, entre otras cosas, los alias del destino (nombre o teléfono del usuario con el que quiere comunicarse). El GKR puede dar permiso para la llamada con un ACF (Admission Confirm) que contiene la dirección de transporte asociada al alias destino o su propia dirección de transporte si decide encaminar la señalización H.225.0. El GKR puede también denegar la llamada con un ARJ (Admission Reject) dando la razón por la cual la llamada no se ha cursado (por ejemplo, no hay suficiente ancho de banda). Durante esta fase el GKR realiza tres funciones: traducción de direcciones, autorización de llamada y gestión del ancho de banda.

- **H.225.0**: Es un subconjunto de mensajes del protocolo de señalización Q.931 de RDSI.

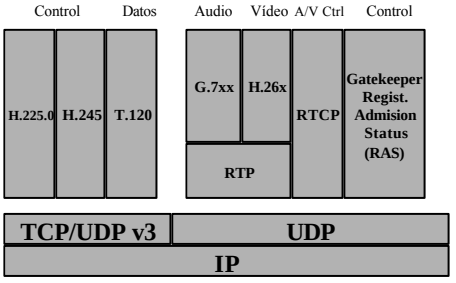


Figura 3. Arquitectura de Protocolos en H.323

Proporciona una conexión lógica entre los dos terminales. En las primeras versiones de la norma, H.225.0 se implementaba sobre TCP pero a partir de la versión 3 existe la posibilidad de utiliza UDP por problemas de retardo.

- **H.245**: Tan pronto como acaba la fase anterior, los dos terminales intercambian sus capacidades. Se ponen de acuerdo en el tipo de información que van a mandar.

Después de estas tres fases, se abren los canales lógicos entre los dos terminales de acuerdo con las capacidades intercambiadas y la comunicación multimedia comienza.

Un ejemplo de una llamada H.323 puede verse en la figura 4.

Los paquetes de audio y vídeo se transmiten sobre UDP, por lo que pueden desordenarse, perderse o retrasarse. Por esta razón se utiliza por encima de UDP el protocolo RTP (Real-Time Transport Protocol). Este protocolo se utiliza para permitir compensar el jitter y el desorden de los paquetes en recepción. El protocolo RTCP (RTP Control Protocol) se usa junto con RTP y permite cierta realimentación sobre la calidad recibida.

Para intentar mejorar la calidad de servicio en los streams de audio y vídeo, se puede utilizar el marcado de paquetes en nodos frontera proporcionado por diffserv. Se elige este mecanismo en lugar de intserv por su sencillez.

La señalización H.225.0 y H.245 puede encaminarse por el GKR o no en función de que se utilice el denominado modelo directo o indirecto. Si el GKR intercepta todos los mensajes de señalización puede realizar gestión de las llamadas manteniendo una tabla con las llamadas activas, el estado de los terminales, etc.

Por tanto, para establecer una comunicación H.323 se abren 3 canales de señalización (H.225.0, H.245 y RAS) más los canales lógicos de audio, vídeo y datos.

Uno de los mayores problemas de H.323 es elevado retardo de establecimiento de llamadas. Con objeto de mejorar la eficiencia, a partir de la versión 2 de la norma se reduce el tiempo de establecimiento de llamada con dos procedimientos: Fast Connect y encapsulado de mensajes H.245 en mensajes Q.931.

3 Proyecto PISCIS

El proyecto PISCIS: Plataforma Piloto de Servicios de Comunicaciones sobre Internet de Servicios Integrados es un proyecto de investigación nacional que trabaja en la problemática de la transmisión de voz sobre redes de paquetes.

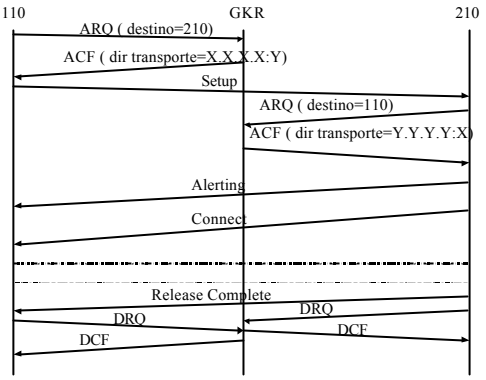


Figura 4. Ejemplo de señalización H.323

El objetivo del proyecto PISCIS es el desarrollo de una plataforma de experimentación de red que permite soportar la prestación de servicios avanzados de voz sobre redes IP con soporte de mecanismos de calidad de servicio (QoS).

El equipo de trabajo está integrado por grupos investigación de la Universidad Politécnica de Madrid, Universidad de Sevilla y Universidad Carlos III de Madrid y cuenta con la colaboración de las empresas Teldat como fabricante de equipos y SuperCable y CyC como operadores de red.

Los objetivos del proyecto sobre los que más se ha trabajado son el despliegue de una red piloto que interconecte las tres universidades utilizando técnicas de transmisión de VoIP que permitan probar la utilidad de los servicios de red desarrollados.

En esta línea, se mantiene activa una plataforma entre las tres universidades. El escenario tipo de cada una de ellas sigue el esquema de la figura 5. Cada escenario cuenta con PCs con software específico (NetMeeting, OpenPhone, ...), una pasarela Teldat para conectar teléfonos tradicionales y una línea telefónica conectada a la misma pasarela para poder cursar llamadas desde/hacia la Red Telefónica Conmutada (RTC).

Durante el año 2000, han montado las tres redes obteniendo una plataforma estable cuyo aspecto es el de la figura 6 y se han realizado pruebas de interconexión entre ambas con unos resultados satisfactorios.

Tras las pruebas de conectividad básica inicial se decidió utilizar la funcionalidad de un gatekeeper para ofrecer servicios de valor añadido como son: la función de directorio, el control de acceso, el soporte de servicios de Help Desk o grupo de salto, la coordinación con mecanismos de calidad de servicio y el desarrollo de pasarelas de buzón de voz-email, funciones que se describirán en los siguientes apartados.

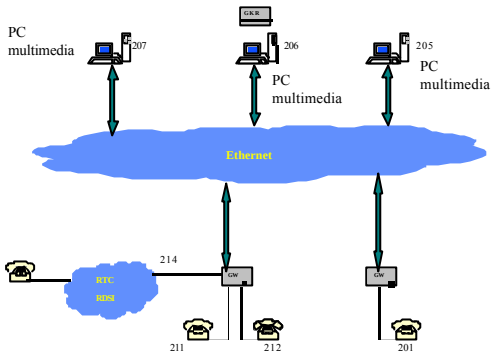


Figura 5. Escenario tipo – pruebas locales

4 Desarrollo de Servicios en redes VoIP

Como ya se ha comentado en apartados anteriores, el gatekeeper (GKR) es un componente que inicialmente se consideró opcional en el modelo, debido a la aparición en el mercado de terminales que podían comunicarse entre sí, sin la necesidad de disponer de dicho elemento, pero que constituye la pieza clave del sistema sin el cual el servicio de VoIP no es más que un juguete. Las funciones que dicho elemento debe realizar son al menos:

- **Función de Registro**: Los terminales al arrancar deben registrase en el GKR (alias, dirección IP, puerto).
- **Traducción de direcciones**: el GKR debe traducir los alias a direcciones de transporte (IP+puerto). La traducción se hace usando una tabla actualizada con los mensajes de registro.
- **Control de admisión**: el GKR controla el acceso a la red mediante los mensajes H.225.0 ARQ (Admission Request), ACF (Admission Confirm) y ARJ (Admission Reject).
- **Control de ancho de banda:** el GKR es notificado con el ancho de banda necesario para el establecimiento de la comunicación entre terminales , en función de los codificadores seleccionados.

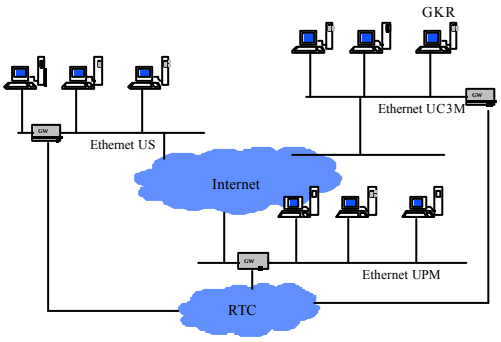


Figura 6. Escenario de interconexión

- **Gestión de una zona**: el GKR debe proporcionar las funciones anteriores a los terminales, MCUs y gateways que se han registrado con él.

Opcionalmente, el GKR debe implementar: señalización de llamada y autorización de la llamada.

En la primera versión de la recomendación, las funciones que debía realizar el GKR eran mucho más reducidas y han ido incluyéndose según las necesidades del sistema H.323. La tarificación es una de las funciones que no se podrían introducir en un escenario H323 sin la presencia de un GKR o de otro componente adicional. El GKR permite que los clientes sean más sencillos a la hora de implementar los servicios suplementarios ya que toda la complejidad reside en él. Podemos implementar nuevos servicios sin necesidad de obligar a todos los usuarios a modificar sus clientes. Otra ventaja es que un usuario puede conectarse en varias máquinas con el mismo alias. De esta forma no necesitas saber la dirección IP o el teléfono del trabajo, de casa,... del usuario con el que quieres hablar, basta con conocer el alias.

A la hora de desarrollar servicios sobre VoIP se vio la necesidad de introducir un GKR en la plataforma del proyecto y se buscaron varios entornos de desarrollo.

4.1 Entornos de desarrollo.

Intentar programar los componentes desde la primera línea de código no parece una solución muy sencilla, además de que nos llevaría mucho tiempo. Por esa razón, se buscaron distintos entornos de desarrollo. El objetivo era encontrar un cliente y un GKR con la funcionalidad básica para luego añadir el código necesario para desarrollar servicios adicionales.

Varios son los fabricantes que ofrecen estos entornos de desarrollo. En el caso de entornos de desarrollo de libre distribución, la selección más adecuada en función de la funcionalidad proporcionada, dinamicidad y disponibilidad de código para plataformas tipo Linux y tipo Windows lo constituyen el proyecto OpenH323 y Opengatekeeper.

Estas dos organizaciones proporcionan código escrito en C++ para el desarrollo de los componentes de un escenario H.323. Actualmente, se han desarrollado clientes, gateways y gatekeeper. El código y los ejecutables están disponibles en ***[15] y [16].***

El proyecto OpenH323 pretende crear una implementación completa del protocolo para teleconferencia ITU H.323 que pueda ser utilizada

sin cargo tanto por desarrolladores como por usuarios comerciales.

OpenH323 se coordina por una compañía australiana, Equivalence Pty Ltd, pero está abierto.

OpenH323 ofrece las librerías PWLib DLL compuestas por una serie de clases que facilitan la programación.

El código de Openh323 incluye clases que implementan los mensajes definidos en las recomendaciones H.225.0, H.245 y H.235 además de proporcionar el soporte para la transmisión de audio y vídeo (codificadores, rtp, ...).

OpenGatekeeper da soporte a todas las características básicas de un gatekeeper H.323 como registro, admisión, traducción de direcciones y control y monitorización de ancho de banda.

Además, da soporte a otras características avanzadas como:

?? Llamadas encaminadas por el GKR (soporte de método indirecto)

?? Soporte de los alias definidos en H.323v2 (party_number, URL, transport_id y email_address).

?? Creación de ficheros log de registro y llamada.

?? Base de datos de GKRs vecinos.

?? Registro del tiempo de vida.

El diagrama de clases de esta distribución se muestra en la fig. 7.

En este sentido los desarrollos realizados en el seno del proyecto PISCIS han sido enviados a estas organizaciones para contribuir como desarrolladores de los componentes H.323.

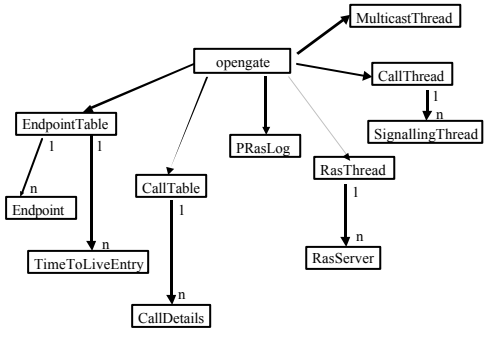


Figura 7. Diagrama de clases del gatekeeper

4.2 Servicios de valor añadido.

Como puede verse, el código implementa las funciones obligatorias del GKR así como alguna de las opcionales.

Tras un periodo de tiempo para tomar contacto con el código. Hemos intentado introducir algunas de las funciones que nos han parecido más interesantes y más útiles en un nuestra plataforma PISCIS.

La funcionalidad introducida ha sido:

- **control de acceso**: permite discriminar los terminales que tienen acceso al servicio a través de la negación de registro en el GKR. De esta forma, podemos controlar que terminal puede o no utilizar el servicio de VoIP.

- **autorización de llamadas**: permite controlar quién puede realizar las llamadas y quién puede recibirlas. Por ejemplo, si no quieres recibir llamadas de un usuario determinado o no quieres recibir llamadas durante un cierto tiempo pero si realizarlas, el GKR permite que sea posible.

- **grupo de salto**: asociamos un alias a distintos terminales de forma que si uno de ellos está ocupado pueda contestar otro terminal libre. Permite implementar, por ejemplo, un servicio de Hot Line o Help Desk

- **buzón de voz - correo electrónico**: permite a los usuarios apuntarse a este servicio de forma que si no contestan a una llamada, se les mandará un e-mail a la dirección de correo electrónico fijada por ellos. Este mensaje contendrá un fichero de voz con una grabación del usuario que ha realizado la llamada. Es un servicio similar al contestador automático.

La principal dificultad encontrada a la hora de introducir nueva funcionalidad al gatekeeper es encontrar el método exacto que hay que extender y el formato de los datos (por ejemplo, las direcciones y los alias) ya que son formatos definidos en las librerías PWLib y Openh323. Por esa razón, se empezó implementando servicios sencillos como los dos primeros que añaden, simplemente, seguridad al escenario.

Una vez implementados los servicios mencionados, se ha instalado el nuevo GKR en la plataforma PISCIS y se han realizado pruebas locales y de interconexión comprobando la utilidad de las nuevas funciones en la misma. Actualmente, la plataforma se encuentra activa con el GKR operativo.

A continuación vamos a comentar como se han implementado cada una de estas nuevas funciones.

4.2.1 Control de acceso

Permite al propietario de la red controlar el acceso a la misma. Impide el registro con el gatekeeper con lo que también impide el uso de cualquiera de sus funciones.

La implementación de esta función consiste en leer de un fichero cada cierto tiempo y guardar el contenido del fichero en una tabla de alias. Esta tabla pasa a formar parte del entorno del GKR. De esa forma, es accesible desde cualquier clase. Cada vez que un usuario se registra, comprobamos que puede hacerlo mirando la tabla de alias.

4.2.2 Autorización de llamadas

Se basa en la existencia de listas que contienen el alias de los equipos. Hasta ahora, se han contemplado las siguientes listas: usuarios a los que no se les permite realizar llamadas, usuarios a los que no se les permite recibir llamadas y parejas de equipos que no pueden establecer una comunicación.

Para implementar estas listas, utilizamos un fichero de texto en el que guardamos las parejas que no pueden establecer conexiones. Igual que para el control de acceso, introducimos los alias en listas que pasan a formar parte del entorno del GKR.

Cuando se recibe una petición de admisión para realizar una llamada (ARQ) se comprueban las listas aceptando o rechazando la misma en función de la información allí contenida.

4.2.3 Grupo de salto.

Cada servicio tiene asociado un grupo de alias que pueden ser un número E.164 o un identificador H.323. Dado que el escenario incluye pasarelas que permiten realizar llamadas con teléfonos convencionales, los alias serán números E.164 para que sean accesibles desde todos los terminales.

La implementación permite que una serie de terminales contesten a la llamada para un mismo alias. Por ejemplo, tenemos un servicio para dar información sobre el tráfico (200). Cuando el usuario llame a ese número el GKR internamente le va a redirigir a uno de los terminales pertenecientes a este servicio que esté libre o le indicará que están todos ocupados.

Además de tener asociado un servicio, los terminales son accesibles desde el exterior independientemente siempre que no estén ocupados.

La implementación actual de este servicio lee periódicamente de un fichero la información sobre las tres funciones anteriores. Sin embargo, se puede modificar el código para incluir bases de datos, solución más apropiada.

Los pasos seguidos para implementar esta funcionalidad son:

- Permitir que dos endpoints en la tabla EndpointTable tengan el mismo alias.
- Hacer rotación de terminales con el mismo alias para encontrar uno libre.
- Controlar si un terminal está libre o no.
- Partiendo de una lista de terminales asociados a un servicio, añadir al array de Alias de un terminal el alias del servicio al que está asociado.
- Servicios privados y públicos: no se permite que ningún terminal se registre con el nombre del servicio privado.

4.2.4 Buzón de voz - correo electrónico

Esta nueva función añadida el GKR es un servicio suplementario no incluido en las recomendaciones H.450.x. Es un servicio que se ofrece a los usuarios similar a un contestador automático o a un buzón de voz.

Para implementar el buzón de voz ha sido necesario modificar los clientes ya que se necesita información adicional a la proporcionada (dirección de correo electrónico). Sin embargo, no se modifica el protocolo de señalización con lo que un cliente que no tenga la modificación puede seguir utilizando el GKR.

El objetivo de este servicio es que un usuario que pertenece a él reciba en la dirección de correo electrónico que decida un e-mail. Este mensaje contiene una grabación con el mensaje dejado por el usuario que realizó la llamada para él.

Cuando alguien llama a un usuario apuntado en este servicio, el GKR indica en el mensaje de ACF que va a encaminar también la señalización H.225.0. Si no se contesta la llamada en un tiempo determinado, el GKR desvía la llamada a un cliente especial (buzón de voz). Para ello, se lanza un timer cuando llega el mensaje Alerting y se sigue la recomendación H.450.3 de desvío de llamada que se muestra en la figura 8.

El buzón de voz reproduce una grabación indicando que se puede grabar un mensaje para el usuario que no contestó a la llamada. Se realiza la grabación, se guarda en un fichero y se manda en un e-mail al usuario destino.

Los usuarios se dan de alta y baja en el servicio realizando una llamada a un número predeterminado. Si el usuario se apunta, el GKR manda un mensaje de petición de información (IRR) al cliente para que éste le conteste con su

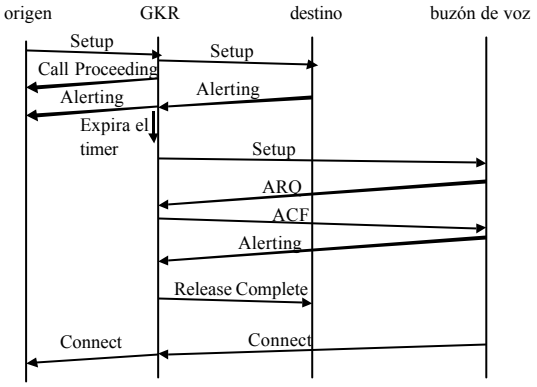


Figura 8. Desvio de llamada

dirección de correo. El GKR guarda esta información en tablas que vuelca periódicamente a ficheros para evitar perder esta información en caso de caída del GKR.

5 Conclusiones

La evolución de la tecnología de transmisión de Voz sobre redes de paquetes presente a medio plazo un ámbito en gran crecimiento motivado por la adopción de este modelo en la evolución de la futura tecnología UMTS, así como la incorporación cada vez mayor en ámbitos corporativos, como solución competitiva en coste, calidad y complejidad frente a las actuales redes conmutadas.

En este artículo se han presentado los protocolos de señalización estandarizados para la provisión del servicio de VoIP, centrándose en el protocolo H.323 por su madurez y disponibilidad de equipos comerciales. De la arquitectura del mismo, destaca el GateKeeper, elemento clave de la arquitectura para la provisión de servicios.

En este ámbito el proyecto de investigación PISCIS ha realizado distintos desarrollos de servicios avanzados que posteriormente han sido probados en la plataforma piloto PISCIS. Para el desarrollo de estos servicios se ha utilizado el entorno de desarrollo de libre distribución OpenGatekeeper.

Agradecimientos

Este trabajo ha sido realizado al amparo del proyecto PISCIS del Plan Nacional de Investigación, programa FEDER.

Los autores agradecen a la empresa Teldat S.A., la colaboración realizada en el desarrollo del proyecto PISCIS, con la cesión de equipamiento de red incorporado a la plataforma PISCIS.

Referencias

[1] 3GPP TS 23.002 v3.3.0: Network Architecture (Release 1999), Marzo 2000. http://www.3gpp.org

[2] 3GPP TS 23.002 v.5.0.0: Network Architecture (Release 5), octubre 2000

[3] ITU-T H.323: “Packet-based Multimedia Communications Systems”, noviembre 2000

[4] http://matrix.it.uc3m.es/?piscis

[5] S. Blake et al, “An Arquitecture for Differentiated Services”, IETF 2475, diciembre 1998, http://www.ietf.org/rfc/rfc2475.txt

[6] http://www.ietf.org/html.charters/intserv-charter.html

[7] M. Handley et al, “SIP: Session Initiation Protocol”, IETF 2543, marzo 1999 http://www.ietf.org/rfc/rfc2543.txt

[8] M. Arango et al, “Media Gateway Control Protocol (MGCP)”, IETF 2705, octubre 1999 http://www.ietf.org/rfc/rfc2705.txt

[9] F. Cuervo, N. Greene, A. Rayhan et al, “Megaco Protocol Version 1.0”, IETF 3015, noviembre 2000, http://www.ietf.org/rfc/rfc3015.txt

[10] ITU-T H.248: “Gateway Control Protocol”, junio 2000

[11] ITU-T H.225.0: “Call Signalling protocols and media stream packetization for packet-based multimedia communication”, noviembre 2000

[12] ITU-T H.245: “Control Protocol for Multimedia Communications”, febrero 2000.

[13] ITU-T H.235: “Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals”, noviembre 2000

[14] ITU-T H.450.3: ”Call diversion suplementary service for H.323”, febrero 1998.

[15] OpenH323 project: http://www.openh323.org

[16] Opengatekeeper project: http://OpenGatekeeper.sourceforge.net

[17] O. Hersent, D. Gurle & Jean-Pierre Petit, “IP Telephony: Packet-base multimedia communications systems”,Addison-Wesley, 2000

[18] D. Collins, “Voice Over IP”, McGraw-Hill, 2001

[19] TELDAT. S.A. http://www.teldat.es

Sistema de Videoconferencia con Calidad de Servicio sobre Redes de Televisión por Cable.

Juan Leal Zubiete, José M. Fornés Rumbao
Área de Ingeniería Telemática. Universidad de Sevilla.
Camino de los Descubrimientos, s/n. 41092 – Sevilla.
Teléfono: 954 48 73 84 Fax: 954 48 73 85
E-mail: {jleal, fornes}@trajano.us.es

***Abstract**. This paper presents the design and development of a videoconference system with quality of service (QoS) guarantees over cable television networks. In order to achieve this, a quality of service system for cable TV networks had to be designed, and Microsoft Netmeeting was modified to perform QoS signaling. The result is a low cost, high quality videoconference system that takes advantage of the high available bandwidth of cable television networks, still making efficient use of bandwidth thanks to the use of QoS techniques. Some of the possible applications of the implemented system include multimedia services related to telelearning, teleworking, telemedicine and video call-centers, besides domestic videoconference and IP telephony.*

1 Introducción

Aunque conceptualmente el servicio de videoconferencia no es nuevo, su despliegue en la actualidad es limitado debido sus requisitos de ancho de banda, mantenimiento del mismo, bajo retardo y pequeña variación de éste (*jitter*).

Por tanto, para la provisión del servicio de videoconferencia sobre redes públicas sin acudir a costosas soluciones basadas, por ejemplo, en circuitos dedicados, es necesario el uso de mecanismos de calidad de servicio [1].

El sistema de acceso mediante modem de cable consiste en la utilización de la infraestructura de los operadores de televisión por cable (CaTV) [2] para la transmisión de datos entre los abonados y la red del operador. Está pensado principalmente para proporcionar acceso a Internet en entornos residencial y de PYMES aprovechando la red de fibra óptica y cable coaxial desplegada por el operador hasta los abonados.

El acceso mediante redes de televisión por cable se beneficia del elevado ancho de banda existente en dichas redes. Además, su despliegue es amplio, es relativamente barato, puesto que aprovecha la infraestructura destinada a la distribución de televisión, y presenta buenas características de retardo. Por tanto, las redes CaTV constituyen a priori una alternativa atractiva para la provisión de servicio de videoconferencia de bajo coste.

Sin embargo, el soporte de QoS en las redes de televisión por cable es limitado. De hecho, aunque la mayoría de los sistemas de modem de cable actuales disponen de mecanismos de reserva de QoS en el nivel MAC, ninguno soporta un sistema de señalización que permita a los usuarios solicitar una cierta QoS a la red cuando la necesitan, sino que el soporte de QoS se utiliza solamente para la definición de distintas modalidades de acceso con distinto precio para el suscriptor [3,4,5].

Teniendo en cuenta todo lo anterior, este artículo presenta un sistema de videoconferencia sobre redes de televisión por cable que implementa mecanismos de calidad de servicio. Para ello se ha desarrollado un sistema que añade capacidad de señalización de QoS a las redes CaTV y una aplicación de videoconferencia que utiliza esta señalización para solicitar la QoS necesaria durante la llamada.

En concreto, el sistema desarrollado trabaja con el sistema de modem de cable propietario del fabricante Com21 [4], que es el utilizado en la actualidad por todos los operadores de cable españoles y uno de los más utilizados a nivel mundial. Sin embargo, el sistema ha sido diseñado con la intención de que pueda ser actualizado al futuro sistema estándar DOCSIS 1.1 [5,6], así como al resto de equipos de la red CaTV.

Las principales aportaciones del trabajo descrito en este artículo son:

- Implementación de una solución de bajo coste para un sistema de videoconferencia, susceptible de formar parte de aplicaciones multimedia relacionadas con teleeducación, teletrabajo, telemedicina, *video call centers*, etc.

- Reconocimiento de la utilidad de las redes de televisión por cable como alternativa de bajo coste para la provisión del servicio de videoconferencia y verificación de la viabilidad

del sistema mediante una maqueta sobre una red real.

- Integración de mecanismos de QoS en el servicio de videoconferencia.

- Desarrollo de un sistema que permite la utilización más eficiente de los recursos de las redes de televisión por cable mediante el uso de mecanismos de calidad de servicio.

El resultado es un sistema de videoconferencia con bajo coste de despliegue ya que utiliza la infraestructura de las redes de televisión por cable y con bajo coste de utilización, puesto que emplea de manera eficiente los recursos de las redes CaTV gracias al uso de mecanismos de calidad de servicio.

Por tanto, el sistema resultante puede ser utilizado en aplicaciones de teleeducación, teletrabajo, telemedicina, además de videoconferencia doméstica, etc.

El resto de este artículo está organizado de la siguiente forma. El epígrafe 2 describe brevemente las redes CaTV y los sistemas de modem de cable. A continuación, el apartado 3 describe el sistema implementado. La maqueta y los resultados de las pruebas del sistema se presentan en el epígrafe 4 y, finalmente, el apartado 5 expone las conclusiones.

2 Las redes de televisión por cable

La tecnología de modem de cable se basa en la multiplexación en frecuencia en la red HFC (red híbrida fibra-coaxial) [7] de los canales de televisión y los dedicados a la transmisión de datos. De esta forma, mediante la utilización de un dispositivo de acceso denominado modem de cable, el abonado puede acceder a los servicios de datos del operador a través del mismo cable por el que recibe el servicio de televisión, con un ancho de banda mucho mayor del que se consigue con el acceso telefónico tradicional.

El operador, a su vez, ha de instalar en sus principales nodos de comunicación (llamados nodos primarios) unos equipos denominados cabeceras de modem de cable o modems maestros, encargados de la comunicación y el control de los modem de cable de los usuarios a través de la red HFC. Dichas cabeceras de modem de cable están conectadas entre sí y con Internet a través de la red troncal del operador (típicamente ATM y/o SDH).

Existe además un nodo especial llamado cabecera de red, que contiene otros elementos importantes como los servidores de correo y web, el segmento de operación y mantenimiento (O&M) de la red y la conexión a Internet. Uno de los equipos conectados al segmento de O&M es típicamente la estación de gestión del sistema de modem de cable.

De entre los sistemas de modem de cable actuales, los sistemas propietarios de Motorola y Com21 [4] se encuentran entre los que tienen una mayor base instalada a nivel mundial. En cuanto a los sistemas estándar, el consorcio CableLabs ha definido un sistema de modem de cable estándar e interoperable llamado *Data Over Cable Service Interface Specifications* (DOCSIS) [5,6]. La versión actual de este sistema es la 1.0 y la 1.1 está en proceso de especificación (*interim status*).

3 Descripción del sistema

Como se ha explicado, el servicio de videoconferencia plantea unos requerimientos de ancho de banda que hacen muy aconsejable la utilización de técnicas de calidad de servicio. Sin embargo, las redes CaTV actuales carecen de soporte de calidad de servicio.

El sistema desarrollado consta por tanto de dos elementos fundamentales: en primer lugar, un sistema de calidad de servicio sobre redes CaTV y en segundo lugar una aplicación de videoconferencia que hace uso de las facilidades de reserva de QoS implementadas.

El sistema de calidad de servicio es responsable de recibir la señalización de QoS de los usuarios y realizar la reserva de recursos sobre los elementos de la red CaTV, en concreto sobre el sistema de modem de cable Com21.

La aplicación de videoconferencia, por su parte, deberá permitir a los usuarios de la red CaTV el mantenimiento de videoconferencias con QoS, realizando para ello las solicitudes de QoS oportunas al iniciarse las llamadas y liberando la reserva cuando la llamada finalice. Los epígrafes siguientes describen cada uno de dichos subsistemas.

3.1 El sistema de calidad de servicio sobre redes de televisión por cable

La función de este subsistema es añadir señalización de QoS a la red CaTV de forma que los usuarios puedan solicitar a la red una calidad de servicio determinada para satisfacer sus necesidades de comunicación, por ejemplo para realizar una videoconferencia.

En este momento, el sistema implementado únicamente trabaja sobre los modem de cable de Com21. Sin embargo, el objetivo es que en el futuro se actualice este sistema para actuar tanto sobre otros sistemas de modem de cable como sobre el resto de elementos de la red CaTV, convirtiéndose de esta forma en un sistema de QoS integrado para toda la red CaTV.

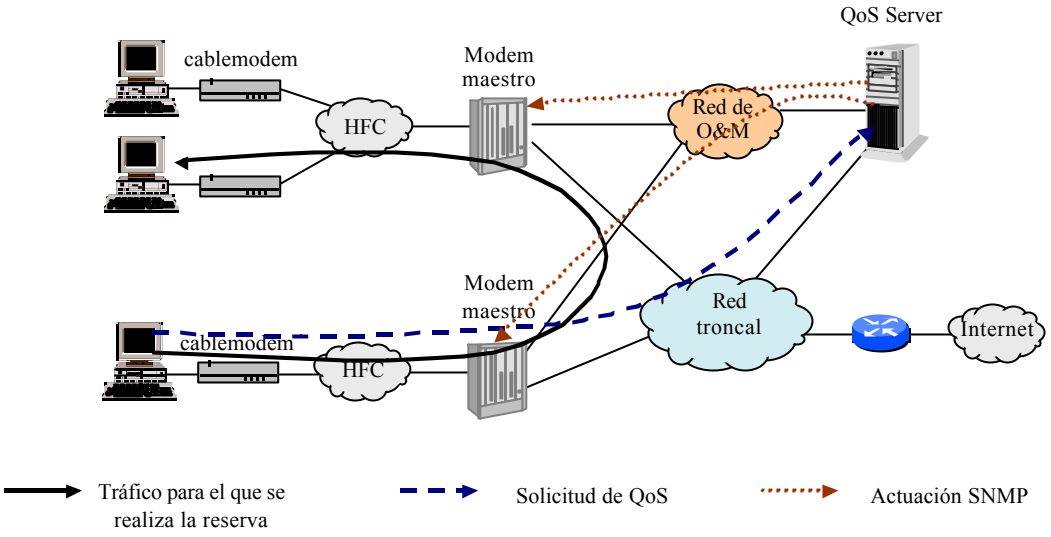


Figura 1: Arquitectura del sistema y proceso de reserva de calidad de servicio.

Los sistemas actuales de QoS para IP (principalmente Servicios Integrados [8] y Servicios Diferenciados [9]) utilizan señalización dentro de banda [10], es decir, la señalización de QoS sigue la misma ruta que los paquetes de datos y es procesada por los mismos elementos de red (*routers*).

Sin embargo, la utilización de esta señalización en el caso de las redes CaTV implica que los modem maestros o los propios modem de cable tendrían que procesar esta señalización. Como el propósito de este trabajo es añadir señalización de QoS a sistemas que no disponen de ella, como es el caso de Com21, no queda otro remedio que recurrir a un servidor centralizado que reciba señalización fuera de banda de los usuarios.

Esto da lugar a la necesidad de usar una arquitectura centralizada con objeto de proporcionar QoS a la red CaTV. El principal elemento de esta arquitectura, mostrada en la Fig. 1, es un servidor de calidad de servicio (QoS Server) que estaría típicamente localizado en la cabecera de red, conectado a la vez a la red troncal y a la red de operación y mantenimiento (O&M).

De esta forma, el funcionamiento de la solución adoptada consiste en que los clientes solicitarán la calidad de servicio al servidor QoS Server, que actuará sobre los equipos apropiados (en este caso, los modem maestros Com21 involucrados) para llevar a cabo la reserva de recursos.

El uso de un sistema centralizado plantea importantes inconvenientes relativos sobre todo a escalabilidad. Resulta evidente que las estructuras centralizadas no son apropiadas para su uso en Internet, donde tienen que interoperar redes administradas por distintas entidades. Por ello, los sistemas de QoS para IP (Servicios Integrados y Servicios Diferenciados) son distribuidos. Sin embargo en redes CaTV, que son administradas por una única entidad, esto no es un problema. De hecho, el uso de un sistema centralizado puede ser la forma más sencilla de integrar las distintas tecnologías (cablemodem, ATM, SDH...) que coexisten en estas redes.

Para llevar a cabo la señalización, en lugar de diseñar un nuevo protocolo sobre TCP o UDP, se ha optado por utilizar CORBA [11]. Esta elección está basada en la simplicidad, ya que CORBA resuelve por sí mismo los aspectos de comunicación e interoperatividad, que podrían resultar engorrosos.

Aunque CORBA sería sin duda inadecuado como sistema de señalización dentro de banda, su uso para la señalización fuera de banda no plantea los mismos problemas. En cuanto a la fundamental objeción que podría esgrimirse contra CORBA, su eficiencia, las pruebas sobre la maqueta, que se describen en este artículo, demostraron que esto no suponía un problema. El resultado, por tanto, es un sistema de señalización de QoS más estándar y fácil de programar que si se utilizara un nuevo protocolo ad hoc.

Los parámetros mediante los cuales el usuario define la QoS deseada por el usuario son los mismos que los usados en Servicios Integrados [8], basados en el modelo del *token bucket*. Con esto se consigue que la especificación de la QoS sea independiente de los formatos usados por los equipos concretos de la red de televisión por cable.

El último problema a resolver es la forma de asignar calidad de servicio sobre el sistema de modem de cable Com21. Este sistema permite a los administradores de la red CaTV asignar a cada modem de cable unos parámetros de operación denominados nivel de QoS. Esta capacidad es

utilizada normalmente por los operadores de cable para ofrecer a sus suscriptores distintos productos basados en distintas velocidades de acceso.

El procedimiento utilizado para asignar QoS sobre el sistema de modem de cable Com21 consiste en emplear la interfaz SNMP [12] de los modem maestros Com21. Al recibir una reserva de QoS, el servidor de QoS implementado modifica temporalmente el nivel de QoS asignado a los modem de cable implicados, restaurando el nivel de QoS contratado al finalizar la reserva.

El formato de especificación de la QoS soportado por el sistema Com21 consiste en la especificación de las tasas de bit máximas (servicio variable bit rate, VBR) o garantizadas (servicio constant bit rate, CBR) en sentido ascendente (usuario a red) y descendente (red a usuario). Ya que este formato es distinto al usado en la señalización del usuario, el servidor debe calcular la QoS Com21 equivalente a la solicitada por el usuario.

Dado que el formato de QoS del sistema Com21 es menos flexible que el empleado en la señalización de usuario (basado en Servicios Integrados), la equivalencia no es biunívoca. La solución adoptada consiste en reservar sobre el sistema Com21 una QoS que, cumpliendo o excediendo la solicitada por el usuario, implica un menor coste al operador de cable.

Por tanto, el proceso de reserva ilustrado en la Fig. 1 es el siguiente. Cuando un usuario desea una cierta QoS, utiliza la interfaz CORBA definida para solicitar dicha QoS al servidor. Éste mediante SNMP busca los modem de cable implicados en todas las cabeceras de modem de cable del operador y modifica su nivel de QoS para satisfacer la petición del usuario. Estas operaciones SNMP se realizan a través de la red de gestión de la red CaTV, inaccesible a los usuarios. Finalmente, cuando el usuario desee liberar la reserva, lo notifica al servidor mediante CORBA y éste restaura mediante SNMP los niveles de QoS contratados.

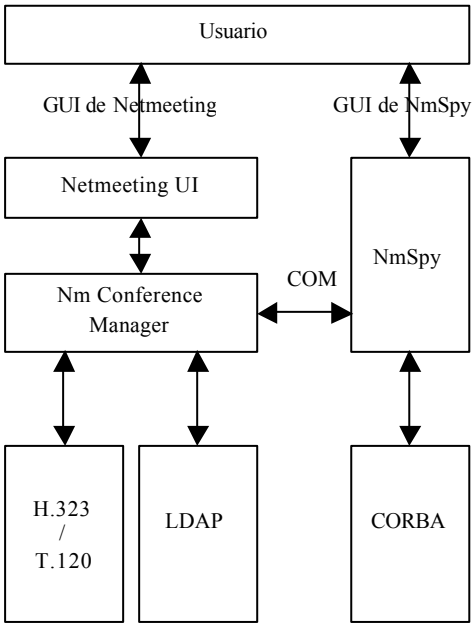


Figura 2: Interacción de NmSpy con Netmeeting.

Para la implementación del servidor se ha elegido una estructura modular basada el lenguaje de programación Java, utilizando las convenciones de programación Java Beans [13]. Esta elección pretende conseguir un sistema que pueda adaptarse para trabajar sobre otros sistemas de modem de cable o elementos de una red CaTV. El principal inconveniente de Java, el rendimiento, no resulta una limitación en este caso al ser la señalización fuera de banda, por lo que el servidor no tiene que cursar el tráfico de los usuarios, sino sólo procesar la señalización. Las pruebas sobre la maqueta del sistema demostraron esta hipótesis.

3.2 La aplicación de videoconferencia

Una vez descrito el sistema de calidad de servicio, el segundo elemento del sistema es una aplicación de videoconferencia que realice señalización de calidad de servicio.

Para ello, en lugar de programar una aplicación completamente nueva, se ha hecho uso del programa Microsoft Netmeeting 3.0, aprovechando la disponibilidad de un software developement kit (SDK) que permite realizar un desarrollo que incorpore a Netmeeting la funcionalidad de solicitar la reserva de QoS.

Microsoft Netmeeting implementa la recomendación H.323 de la ITU-T [14], permitiendo realizar llamadas de voz sobre IP, videoconferencias y otros servicios como pizarra compartida, etc.

Gracias a la popularidad de Netmeeting y al hecho de que se distribuya de forma gratuita a los usuarios de Microsoft Windows, el resultado es un sistema de videoconferencia de bajo coste que proporciona una buena calidad de videoconferencia sobre redes de televisión por cable.

De entre las posibilidades de programación que ofrece el SDK de Netmeeting, se ha elegido la implementación de una aplicación, llamada NmSpy, que realiza la señalización de la reserva de QoS a beneficio de Netmeeting. NmSpy es informada por Netmeeting, a través de una interfaz COM [15], del estado de las llamadas y realiza la señalización de QoS usando CORBA. Esta arquitectura se muestra en la Fig. 2.

Esta aplicación se ejecuta de manera simultánea a Netmeeting (si éste no se está ejecutando al iniciarse NmSpy, es arrancado automáticamente). De esta forma, cuando el usuario inicia una

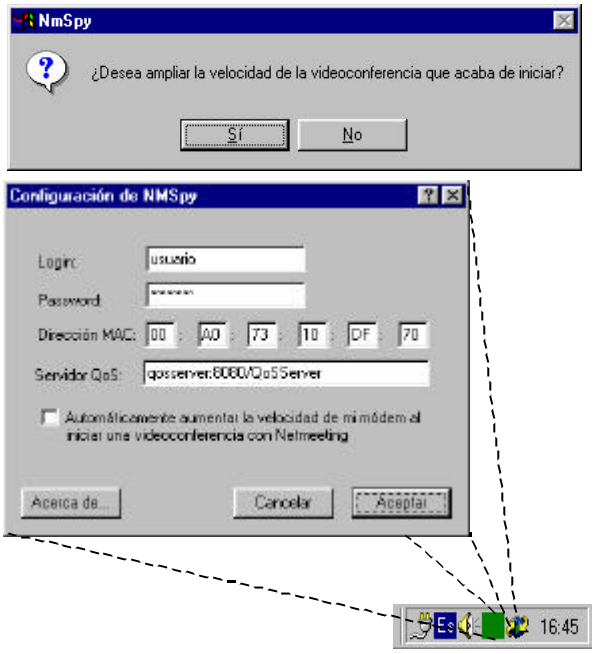


Figura 3: Interfaz gráfica de NmSpy.

videoconferencia, Netmeeting informa a NmSpy de este hecho, y éste solicita al servidor de QoS la calidad de servicio deseada para la videoconferencia. Cuando la videoconferencia finaliza, NmSpy recibe la notificación pertinente y libera la reserva de QoS.

NmSpy muestra al usuario una interfaz gráfica muy simple (presentada en la Fig. 3), consistente únicamente en un icono situado en la barra de tareas indicando el estado de la reserva, un diálogo de confirmación de que se va a realizar una reserva de QoS y una ventana de configuración.

De esta forma, el usuario utiliza la interfaz gráfica estándar de Netmeeting para establecer la videoconferencia o la llamada VoIP, y NmSpy se integra de forma casi transparente al usuario.

Esta aplicación se ha desarrollado en Visual C++, debido a su potencia para el manejo de objetos COM (utilizados para la comunicación con Netmeeting), su flexibilidad para el trabajo con ventanas y a que produce un ejecutable pequeño y eficiente. El *Object Request Broker* (ORB) empleado para la señalización de QoS basado en CORBA fue MICO, versión 2.3.1.

4 Maqueta del sistema y pruebas

El sistema se ha probado sobre la red HFC real del operador Supercable Andalucía S.A., con licencia en la mayoría de demarcaciones de Andalucía. Para ello se montó una maqueta consistente en un modem maestro Com21 dedicado, una estación de gestión del sistema Com21, cuatro modem de cable, un PC con sistema operativo Linux actuando como servidor de QoS y dos PCs multimedia con Windows 95 actuando como clientes. La maqueta utilizó asimismo la red SDH del operador y una porción privada, destinada a pruebas, de la red HFC real.

Los resultados de las pruebas fueron satisfactorios, modificándose adecuadamente los niveles de QoS de los modem de cable pertinentes y realizándose con éxito una videoconferencia de buena calidad con garantías de QoS.

Las mediciones de rendimiento mostraron que el número máximo de peticiones por minuto que el sistema podía servir fue de alrededor de cuarenta, respondiendo esta limitación principalmente al tiempo de respuesta del modem maestro a las operaciones SNMP. De hecho, el tiempo de respuesta del servidor, excluyendo las operaciones SNMP fue de alrededor de 90 ms., lo que permite estimar la carga máxima en unas 600 peticiones por minuto. Esto prueba la hipótesis de que el uso de Java y CORBA no es una limitación importante en cuanto al rendimiento.

La Fig. 4 muestra los resultados de las pruebas de rendimiento en términos de tiempo de servicio de las reservas para distintas frecuencias de llegada de solicitudes. Como puede observarse, cuando la frecuencia alcanza 40 solicitudes por minuto, el tiempo de servicio comienza a crecer y algunas solicitudes comienzan a ser denegadas. Desde una aproximación conservadora, éste punto se toma como el límite de carga del sistema, aunque a priori debería ser el punto en que el tiempo de servicio iguala el intervalo entre peticiones, lo cual ocurre en torno a 80 peticiones por minuto. La estabilización final de la gráfica es debida al incremento en el porcentaje de reservas infructuosas, las cuales se sirven en menos tiempo. Es necesario resaltar que la limitación deriva del número de operaciones SNMP por minuto que el modem maestro Com21 es capaz de procesar.

Como resultado, se consiguió establecer videoconferencias de gran calidad, codificando la imagen y el audio a la máxima calidad soportada por Netmeeting. Además, se sometió al sistema a pruebas consistentes en la solicitud de varias

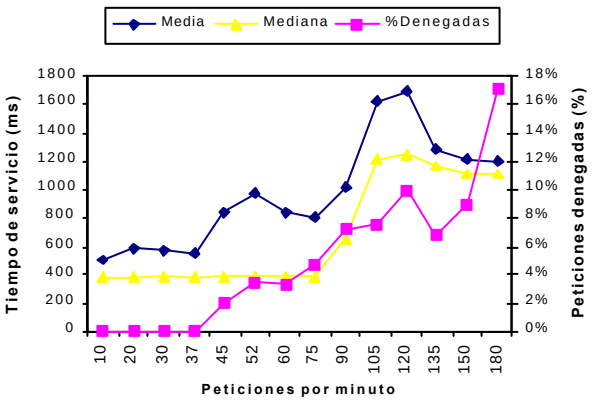


Figura 4. Medidas de rendimiento.

reservas simultáneas, simulación de fallo en la red desconectando al servidor de la misma y simulación de la caída del servidor apagándolo de forma abrupta. El sistema respondió correctamente a todas estas pruebas, restaurando siempre la QoS original contratada por el cliente.

5 Líneas futuras y conclusiones

En sentido estricto, para proporcionar calidad de servicio extremo a extremo, ésta tiene que ser soportada por todos los elementos del trayecto entre los sistemas finales. Por tanto, el objetivo final de este trabajo sería interactuar con todos los sistemas necesarios de forma que las garantías de QoS se extendieran de extremo a extremo. Para ello sería necesario desarrollar gestores para otros dispositivos u otros sistemas de gestión de QoS. Además, serían necesarios nuevos agentes que implementaran protocolos de señalización estándar para la comunicación con los usuarios. Finalmente, sería necesario implementar sistemas eficaces de autenticación y tarificación.

En todo caso, el sistema permite actualmente garantizar QoS en el acceso y, si la red del operador está suficientemente sobredimensionada, como suele ser el caso, puede ser suficiente para mantener videoconferencias con buena calidad entre usuarios de modem de cable del operador.

Un punto especialmente importante que estamos estudiando en este momento es la forma de integrar el sistema con el sistema de modem de cable DOCSIS 1.1, que como se ha dicho aún no ha concluido el proceso de especificación.

Referencias

[1] Stardust.com. "White Paper – The Need for QoS". 1999.

[2] W. Ciciora. "Cable Television in the United States. An Overview". Cablelabs (1995).

[3] R. Rabbat, K. Siu. “QoS Support for Integrated Services over CaTV”. IEEE Communications Magazine. Vol. 37-1 (1999).

[4] Com21 Inc. “ComUNITY Access System – 2.3 Release – Technical Reference Manual”. (1998).

[5] D. Fellows, D. Jones. “DOCSIS Cablemodem Technology”. IEEE Communications Magazine. Vol 39-3 (2001).

[6] Cablelabs. "Data-Over-Cable Service Specifications. Radio Frequency Interface Specification. Version 1.1, revision I05". (7-2000).

[7] C. Bisdikian, K. Maruyama, D. Seidman, D. Serpanos. "Cable Access Beyond the Hype: On Residential Broadband Data Services over HFC Networks". IEEE Communications Magazine. Vol 34-11 (1996).

[8] IETF RFC 1633. “Integrated Services in the Internet Architecture: an Overview”. (6-1994).

[9] IETF RFC 2475. “An Architecture for Differentiated Services”. (12-1998).

[10] IETF RFC 2205. “Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification”. (7-1997).

[11] Object Management Group (OMG). “The Common Object Request Broker Architecture and Specification. Revision 2.2”. (2-1998).

[12] IETF RFC 1157. “Simple Network Management Protocol (SNMP)”. (5-1990).

[13] Sun Microsystems. “The JavaBeans API Specification” (7-1997).

[14] J.Toga and J.Ott. "ITU-T Standardization Activities for Interactive Multimedia Communications on Packet Networks: H.323 and Related Recommendations ". Computer Networks , vol. 31 , 1999.

[15] D. Chappell. “Understanding ActiveX and OLE”, Microsoft Press, 1996.

Evaluación del Transporte de Imagen en Procesos de Visión sobre una Red Industrial Profibus

Víctor M. Sempere Payá (vsempere@dcom.upv.es) DCOM. EPSA. UPV.
Javier Silvestre Blanes (silbla@disca.upv.es) DISCA. EPSA. UPV.
Antonio Cano Morcillo (acano@eln.upv.es) DIE. EPSA. UPV.

Abstract: In this paper we analyze the use of Profibus for image transport that are going to be used in a typical process of industrial computer vision. We evaluate the capabilities of the industrial network for simultaneously transporting this type of traffic with typical control traffic, and the influence that we can obtain in the number of images that we can transport using compression techniques, analyzing the image quality loss in the computer vision process due to image compression. We show developed applications for the capture and transmission of images that we use to take experimental values in Profibus. With different sizes of images and compression rates, we get results that demonstrate the viability of this kind of system under the conditions explained.

1 Introducción.

En los últimos años se ha realizado un gran esfuerzo en la industria para mejorar el control y la supervisión en los procesos de control industrial, implantándose nuevos sistemas de control y de comunicaciones, atendiendo en este último aspecto a los estándares reconocidos internacionalmente. Paralelamente, se ha extendido el uso de aplicaciones multimedia en la industria [2][8], siendo la visión por computador una de las más importantes debido a las ventajas operativas que puede aportar, a la reducción del coste de estos equipamientos, y al incremento de prestaciones debido a los avances software y hardware aplicados a esta tecnología. En este contexto, una de las aplicaciones más interesantes es el procesamiento de imágenes para el control del proceso o producto [7]. El o los sistemas de adquisición de imágenes transfieren las imágenes capturadas al módulo o módulos encargados de procesarlas. Como resultado del procesamiento realizado sobre las imágenes, se pueden tomar decisiones de control que actúen sobre los parámetros del proceso productivo.

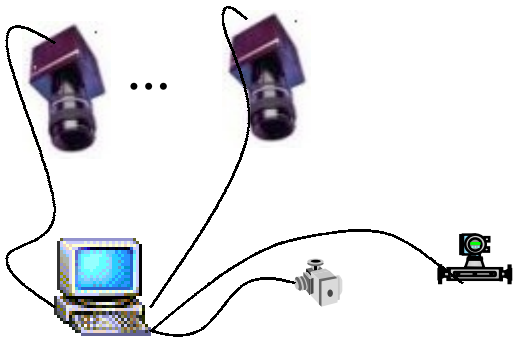
La transferencia de la imágenes a los módulos de proceso se puede realizar de tres formas diferentes:

Tipo 1. Conexión directa entre la cámara y las tarjetas de adquisición colocadas en el ordenador industrial (Fig. 1a). Esta es la forma más habitual de trabajo debido a que el sistema es capaz de trabajar con elevadas tasas de transferencia de imágenes de alta resolución para su procesado. Resulta adecuado cuando el número de bytes por segundo a procesar es muy elevado, (hasta 45 Mbytes/sg) y requiere de potentes sistemas de procesado dedicados.

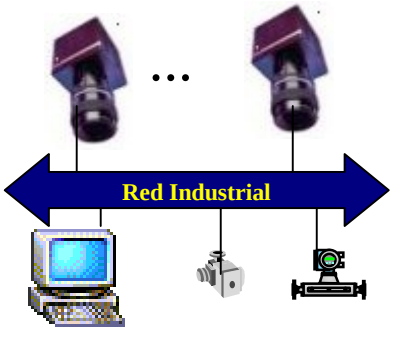
Tipo 2. Procesado local en la cámara. Cada vez es más frecuente encontrar cámaras con procesadores incorporados capaces de hacer un primer tratamiento sobre la señal, e incluso de realizar el procesado completo si este no es muy complejo. Por tanto, puede transmitir la imagen preprocesada al ordenador, para trabajar como en el modelo anterior, o puede comunicar directamente los resultados del proceso realizado, para que el ordenador actúe sobre los controles adecuados.

Tipo 3. Conexión entre la cámara y el módulo de proceso a través de una red industrial (Fig. 1b) [1]. Las imágenes capturadas localmente por la cámara, se transmiten por la red industrial para su procesado remoto. Estas pueden viajar comprimidas o no, dependiendo de la limitación impuesta por el ancho de banda disponible en la red, debiéndose garantizar su coexistencia con el tráfico de control existente.

La alternativa analizada en este artículo es la Tipo 3, realizándose una transmisión de imágenes a través de una red industrial Profibus para procesar éstas en ordenadores conectados a la red



a) Forma habitual de trabajo



b) Alternativa analizada

Figura 1. Métodos de transmisión de la imagen

de forma que, tras el procesado de la imagen, se pueda actuar sobre otros actuadores de la red, integrando por tanto la información extraída de los procesos de visión con los sistemas de control existentes.

En función del ancho de banda disponible y de las características de la imágenes, así como de su frecuencia, los sistemas de comunicaciones industriales serán capaces, o no, de realizar el transporte de las mismas de forma que se cumplan las restricciones temporales sobre los actuadores que imponga el resultado del procesamiento de la imagen.

La compresión incrementará el número de imágenes que el sistema puede transportar, a costa de perder calidad en la imagen, lo que será evaluado en este trabajo mediante un proceso básico convencional de visión artificial que se aplicará en las imágenes recibidas, con el fin de medir la compresión tolerable para un proceso bajo determinadas características.

2 Las redes industriales. Profibus.

Las redes industriales se encuentran en los niveles de proceso y célula de la jerarquía CIM (*Computer Integrated Manufacturing*) [5]. Se trata de sistemas telemáticos que han sustituido al clásico cableado en estrella (típicamente lazos de 4-20mA) entre sensores/actuadores y sistemas de control, por un bus de comunicaciones. Este hecho ha permitido la reducción de costes, la facilidad de instalación y configuración, el uso de diagnósticos para detección de averías, etc. Estas redes permiten descentralizar el conexionado de los elementos que participan en el control, obteniendo sistemas con procesamiento distribuido que satisfacen los requerimientos temporales críticos necesarios en algunas de las aplicaciones que las utilizan.

Dentro de estas redes, Profibus [10] es una solución ampliamente extendida, estándar (EN 50170, IEC 61158) y abierta para las redes industriales de propósito general que operan a nivel de proceso y célula. La arquitectura del protocolo no contempla los niveles 3, 4, 5 y 6 del modelo OSI con el fin de incrementar la eficiencia y reducir el tiempo de procesado. Profibus permite a un amplio rango de aplicaciones industriales hacer uso de sus capacidades, incluyendo alta velocidad de transmisión (hasta 12 Mb/sg) y gestión de procesos complejos o con requerimientos temporales críticos.

Existen tres variaciones de Profibus destinadas a satisfacer diferentes requerimientos de las aplicaciones Profibus-FMS, Profibus-DP y

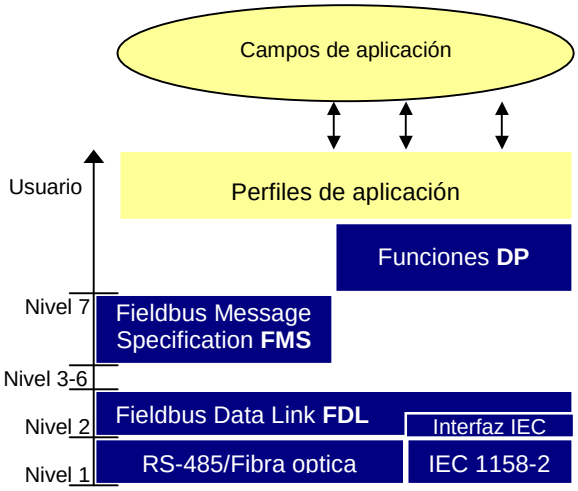


Figura 2. Arquitectura Profibus

Profibus-PA. Profibus FMS se implementa a nivel de aplicación y permite la transferencia de información de forma abierta entre procesos de aplicación. Su principal inconveniente es que disminuyen los tiempos de reacción del sistema por el procesamiento necesario. La velocidad máxima de transmisión es de 1.5 Mbps. Profibus DP (*Distributed Peripherial*) está diseñado para la comunicación entre controladores y periféricos distribuidos por la red a muy alta velocidad. Mapea sus servicios directamente en el nivel 2 (denominado FDL, Fieldbus Data Link), evitando el nivel de aplicación y alcanzando una velocidad máxima de 12 Mbps. Profibus PA sirve para aplicaciones que exijan seguridad intrínseca, el protocolo dispone de la posibilidad de utilizar en el nivel físico el protocolo IEC 1158-2, aunque esto limita la velocidad a 31,25 Kbps.

La aplicación desarrollada para evaluar el comportamiento de Profibus ha sido implementada directamente sobre el nivel de enlace FDL utilizando algunos de los servicios ofrecidos por el mismo, esto hace que pueda coexistir con el resto de tráfico existente en la red. Se utilizan los servicios SDA (*Send Data with Acknowledge*) para abordar todos los aspectos de negociación entre estaciones como las características de las imágenes, resolución y número de bits por pixel, la señalización de final de imagen, y para indicar la finalización de la transmisión y los servicios SDN (*Send Data with No acknowledge*) para la propia transmisión de las imágenes.

El protocolo implementado es orientado a conexión, y dado que se transportan imágenes para ser procesadas en tiempo real y actuar sobre el proceso como resultado de dicho procesamiento no se realizan intentos de recuperación de las mismas en caso de fallos. En el apartado cuatro se detalla con precisión los detalles de este protocolo.

3 Compresión de imágenes.

La compresión de imágenes puede aumentar considerablemente el rendimiento del sistema al aportar una gran reducción del ancho de banda consumido para el transporte de las mismas. Esta reducción puede ser muy útil cuando el ancho de banda disponible, debido al consumo del tráfico de control y el reservado para el tráfico aperiódico, sea escaso para el tamaño y frecuencia de imágenes que se tengan que transmitir.

Los sistemas de compresión de imágenes digitales (con o sin movimiento) se basan en la redundancia espacial y temporal de la/s imágenes ([9]. JPEG, MJPEG, MPEG1, MPEG2, MPEG4, H.261 y H.263). Estas técnicas presentan diferentes propiedades en cuanto a la calidad de la imagen comprimida, la eficiencia espectral, complejidad de codificación y decodificación, etc.

Los parámetros fundamentales a tener en cuenta cuando se han de comprimir imágenes digitales en el contexto de aplicaciones analizado son:

- Tiempo real: dependiendo de los requerimientos de captura, transmisión y procesado en tiempo real, la elección puede estar condicionada por la complejidad del CODEC (codificador /decodificador).
- Compresión con o sin pérdida: la compresión con pérdida de información implica una mayor relación de compresión y la degradación en la calidad de la señal. Si esta degradación es aceptable por la aplicación receptora de la imagen, es conveniente su utilización pues se consigue mayor *Throughtput* en la transmisión de las mismas.
- Inter-frame/intra-frame: En modo inter-frame se comprime cada imagen de una secuencia como si de una imagen estática se tratara, mientras que con esquemas intra-frame se emplean técnicas predictivas, con las que se logran mayores niveles de compresión, si bien la imagen debe tener una continuidad espacial: éste no es el caso a estudiar, por lo que no se analizarán estas formas de compresión.
- Resolución: se refiere al número de píxeles por imagen, que dependerá de la aplicación: en procesos de supervisión y monitorización, se pueden emplear en general resoluciones del orden de QCIF (176x144) a CIF (352x288). En control de calidad, pueden ser necesarias resoluciones 4CIF (720x576) o mayores.

Las técnicas utilizadas para la compresión de imagen en movimiento no se han considerado válidas para la transmisión de imágenes en tiempo real en aplicaciones industriales de visión artificial. Existen dos motivos principalmente. En primer lugar, la captura de imágenes en estas aplicaciones suele estar gobernada por una señal de control (fotocélula, encoder, etc), no necesariamente periódica, encargada de sincronizar la captura de la imagen con el instante de tiempo en que el objeto de estudio se encuentra en el campo de visión de las cámaras, lo que impide la utilización de técnicas predictivas intra-frame. En segundo lugar, las técnicas de compresión de imagen en movimiento utilizan esquemas de control de codificación VBR (*Variable Bit Rate*) o CBR (*Constant Bit Rate*) no utilizables en aplicaciones de este tipo. Ambos esquemas presentan un flujo de información constante cuando la variabilidad de la escena es muy pequeña. Ante cambios importantes en esta variabilidad de la escena, el comportamiento de estos modelos no es adecuado para el tipo de aplicaciones planteadas en este trabajo. El esquema VBR necesita incrementar el ancho de banda que está utilizando para codificar estos cambios en la escena, lo que no es viable en sistemas de tiempo real, ya que se necesita conocer a priori el ancho de banda consumido y el tiempo de transporte para la sincronización de los elementos involucrados. El esquema CBR si proporciona un flujo constante, a costa de reducir la calidad de la imagen, lo que no es aceptable desde el punto de vista del procesamiento de las imágenes, si se está trabajando de forma lógica cerca del limite aceptable para la aplicación. Estos hechos hacen que la técnica utilizada para la compresión de imágenes no explote la redundancia temporal, aprovechando únicamente la información disponible en cada imagen para realizar la compresión.

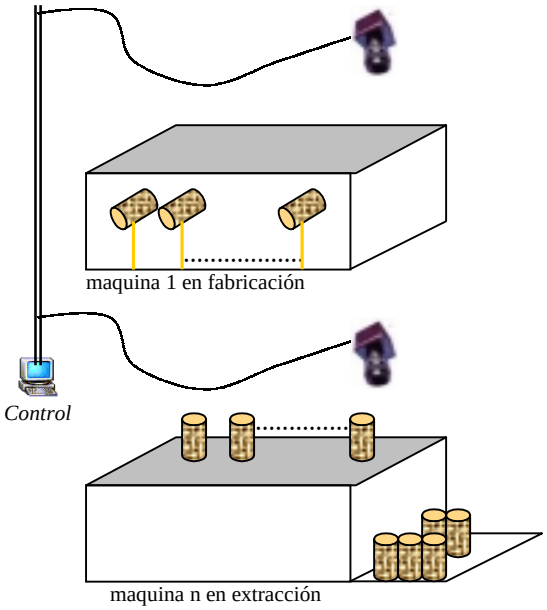
La técnica de compresión de imágenes utilizada en este trabajo ha sido compresión JPEG por software, dada la calidad de la imagen, la menor complejidad del CODEC y las tasas de compresión que se pueden alcanzar. JPEG (*Joint Photographic Experts Group*) es un estándar ISO del año 91 utilizado en la compresión de imagen de tonos continuos (imagen fotográfica) que admite cuatro modos de funcionamiento [4].

4 Escenario y aplicaciones.

4.1 Descripción del sistema

Los sistemas de visión artificial no siempre requieren todo el ancho de banda disponible entre la cámara y las tarjetas de captura. En muchos casos, especialmente en aplicaciones industriales de la visión artificial, la captura de la imagen está gobernada por una señal de sincronización procedente del proceso. Por ejemplo en el control de calidad *on-line* de las bobinas en un proceso de fabricación de hilados, las máquinas que generan las bobinas tardan un tiempo considerable en realizar esta operación (Fig. 3).

En este contexto, la utilización de sistemas de Tipo 1 (conexión directa entre la cámara y la tarjeta de adquisición) supondría un elevado



Cuando finaliza la producción de bobinas en cada máquina se pasa al proceso de extracción para su paletización, y poder así reanudar la producción. Es difícil que coincidan varias máquinas en este proceso de extracción y si lo hacen, el tiempo de espera será despreciable en el cómputo global de la producción.

Figura 3. Maquinaria para la producción de bobinas de hilo.

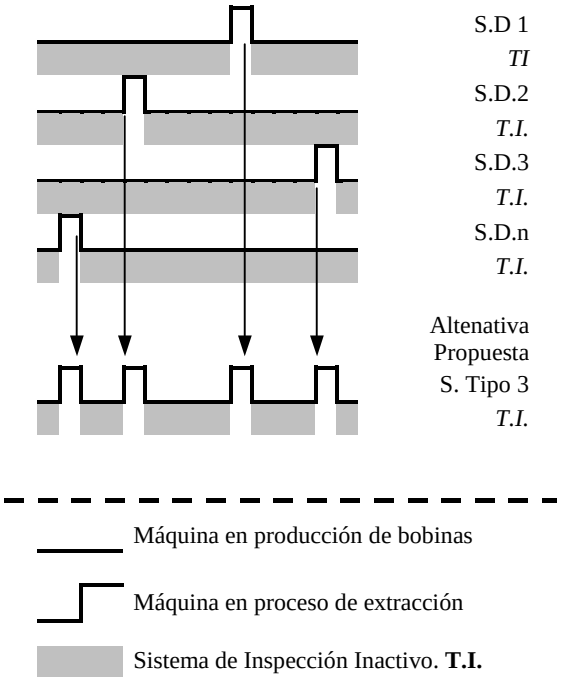


Figura 4. Diagrama de Tiempos de Operación para los sistemas de inspección en un proceso de hilatura

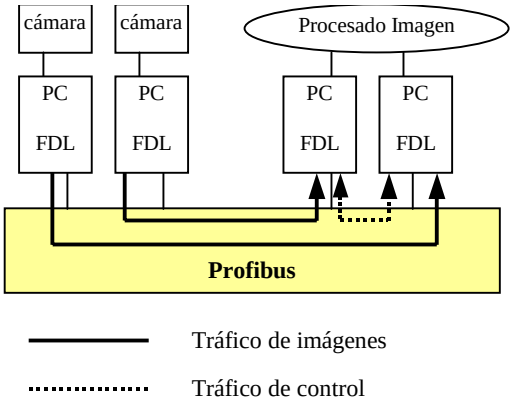


Figura 5. Escenario de trabajo.

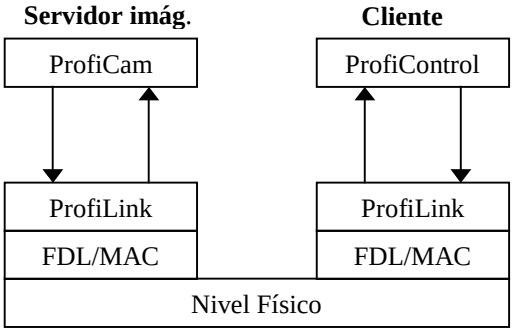


Figura 6. Arquitectura del sistema

desaprovechamiento de los módulos de procesado para control de calidad (Fig. 4) además de suponer un mayor coste.

Otros ejemplos similares se pueden dar en la industria de tejeduría, donde la velocidad de producción es muy lenta en comparación con las capacidades de los sistemas de captura de imágenes, y, en general, la utilización de sistemas de Tipo 3 se puede dar en cualquier sector de producción en donde la cantidad y calidad de la información generada pueda adaptarse al ancho de banda disponible en la red industrial.

El escenario escogido para analizar la viabilidad de sistemas de Tipo 3 implementados sobre redes Profibus se compone de dos equipos servidores de imágenes y dos equipos de visualización y procesado, así como para la generación del tráfico de control típico en una red industrial, como puede verse en la figura 5.

Dado que el sistema diseñado, una vez configurado y en operación sufrirá pocos o ningún cambio en su parametrización, el servidor de imágenes se ha implementado sobre una estación maestra ya que debe funcionar de forma autónoma y lanzar en su tiempo de posesión de testigo imágenes a las estaciones receptoras sin requerir petición alguna. Además un maestro es capaz de operar en modo Multicast (de uno a un grupo), esto puede ser importante para algunas aplicaciones en control que requieran monitorización o procesamiento simultáneo en varios clientes.

4.2 Protocolo de comunicaciones.

En el sistema se ejecutan tres procesos en dos máquinas diferentes (Figura 6). Uno denominado ProfiLink, es el interfaz con el medio de transporte utilizado, en este caso el FDL de Profibus, y se ejecuta tanto en la estación cliente como en la servidora. Los otros dos son el proceso encargado de suministrar las imágenes en el servidor de imágenes, ProfiCam, y el proceso encargado del procesado de las mismas en la estación cliente, ProfiControl, y que son interconectados de forma transparente sobre los procesos ProfiLink y la red Profibus. Para la sincronización del funcionamiento de los procesos concurrentes en la misma máquina se han utilizado objetos CEvent de la Aplication Program Interface de Windows (API) [6]. En cuanto a la información a intercambiar entre estos procesos, se utiliza una estructura *FileMapping* (API de Windows) en memoria compartida en cada una de las máquinas, lo que les permite tener una zona de memoria común donde volcar la información de forma sincronizada a los procesos que se ejecutan en la misma máquina. El nivel de enlace de datos de Profibus proporciona los servicios SDA y SDN necesarios para la transmisión de imágenes analizadas en este trabajo. Las tramas SDA, con el fin de maximizar la capacidad de transmisión de imágenes de Profibus, se utilizan únicamente para realizar operaciones cuya fiabilidad haya de ser garantizada (Tabla 1).

Tabla 1. tipos de tramas SDA modificados.

1	2	2	1	← nº bytes utilizados
código				
0	X	Y	color	negociación
1	-	-	-	fin de imagen
2	-	-	-	fin de la comunicación

Tabla 2. Primitivas de servicio.

Primitiva	Origen	Descripción
negociación (neg)	ProfiCam	Comunicación de los parámetros de la imagen al proceso de comunicación Profibus.
parámetros (par)	ProfiLink	Comunicación de los parámetros de la imagen al proceso de control.
enviar_imagen (env_i)	ProfiCam	Solicitud de transmisión de la imagen
recibida_imagen (rev_i)	ProfiLink	Comunicación al proceso de control de la llegada de una nueva imagen.
liberación (lib)	ProfiCam	Solicitud de liberación de la conexión
liberacion(lib)	ProfiLink	Comunicación de liberación de la conexión

4.2.1 Establecimiento de la conexión.

Dado que se trata de un protocolo orientado a conexión, la primera acción es establecer ésta (ver figura 7). El proceso ProfiCam es el encargado de iniciar la conexión, inicialización que servirá además para comunicar a todos los procesos involucrados las características de las imágenes que va a transmitir. Este proceso se dará siempre que se cambien las características fundamentales de las imágenes a transmitir y procesar, como los tamaños en X e Y, y el número de bits por pixel utilizados. Para realizar esta conexión, utiliza la primitiva *negociación* (***neg***, Tabla 2), iniciando la solicitud del inicio de la conexión. El proceso ProfiLink, al recibir esta primitiva, realiza la transmisión de una trama SDA de tipo 0 (Tabla 1) transmitiendo esta información al proceso ProfiLink receptor. Este proceso mediante la primitiva *parámetros* (***par***) comunica todos los parámetros de las imágenes al proceso ProfiControl, que es el que necesita conocer esta información para poder procesar de forma correcta las imágenes recibidas. Por otra parte, los procesos ProfiLink necesitan esta información para conocer el número de tramas en que han de paquetizar la imagen para su transmisión.

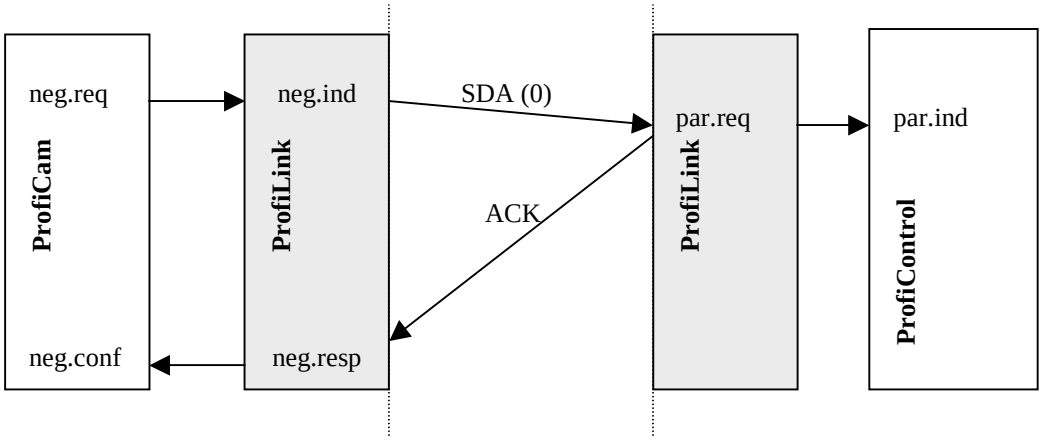


Figura 7. Establecimiento de la conexión

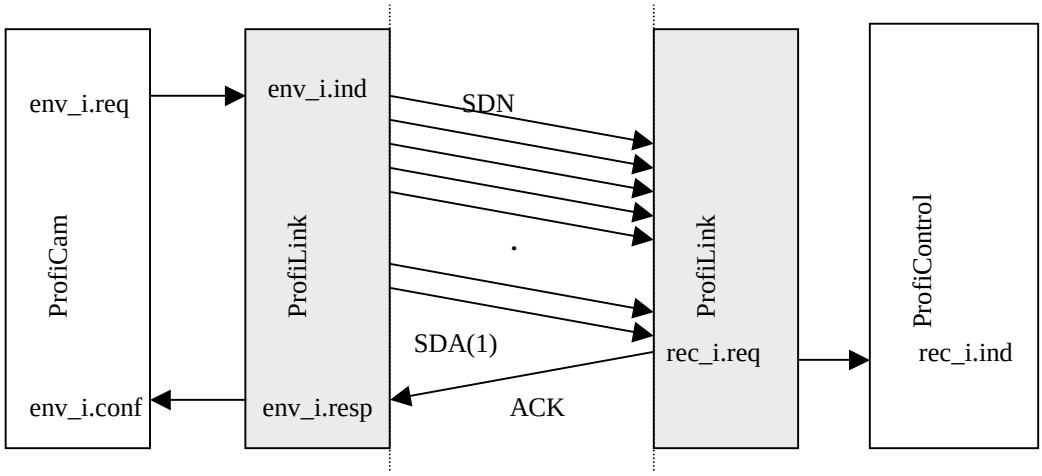


Figura 8. Fase de transmisión de imágenes

A su vez, como consecuencia de la recepción del SDA(0), el FDL del proceso ProfiLink receptor emite un reconocimiento al proceso ProfiLink emisor, lo que permite confirmar el establecimiento de la conexión a ProfiCam, que desde ese momento esta habilitado para iniciar la transmisión de imágenes a la cadencia programada en la estación servidora.

4.2.2 Transmisión de imágenes

Una vez negociados los parámetros de la transmisión, cada vez que la estación generadora disponga de una imagen, la colocará en la memoria común de los procesos ProfiCam-ProfiLink, y señalizará el hecho al proceso encargado de la transmisión mediante la primitiva *enviar_imagen* (***env_i***, Fig. 8). ProfiLink, al recibir esta petición, enviará la imagen por el bus mediante la transmisión de tantos SDN como sea necesario, enviando un SDA de tipo 1 al finalizar la transmisión de la imagen. Este SDA provocará en el ProfiLink receptor la comunicación al proceso ProfiControl de la recepción de una nueva imagen mediante la primitiva *recibida_imagen* (*rec_i*). A su vez, el reconocimiento del envío de la trama SDA provocará en el proceso ProfiLink emisor, la confirmación de la transmisión de la imagen, lo que habilitará al proceso ProfiCam para la captura de la siguiente imagen.

4.2.3 Formato de la trama de datos

Se utilizará un byte del campo de datos para numerar las secuencias de paquetes que forman una imagen y controlar la recepción de todos los paquetes. Esto es importante ya que la aplicación es de procesado de imagen, y la perdida o recepción desordenada de tramas podría suponer una importante distorsión en la imagen y una extracción de información y conclusiones erróneas, lo que podría generar importantes secuencias de acciones de control incorrectas. El formato modificado de la trama SDN utiliza 239 de los 240 bytes disponibles para transmitir información de la imagen, y el primer byte para la numeración de las tramas (Fig. 9)

En el caso de no utilizar compresión, la utilización de esta numeración, garantiza que cada paquete se coloque en el lugar adecuado en el buffer de recepción. Si se pierde algún paquete, se mantendría la información de la última imagen, lo que provocaría un porcentaje de error en el procesado de la imagen prácticamente despreciable, teniendo en cuenta la baja tasa de paquetes perdidos que proporciona una red Profibus y la poca probabilidad de que sea precisamente el paquete perdido el que contenga la información más relevante de la imagen. Si se utiliza compresión, esta numeración permite detectar la falta de tramas, lo que descartaría el procesado de la imagen, ya que la información contenida en una trama, utilizando compresión, puede afectar de forma muy significativa y dependiente del ratio, a la imagen resultante.

nº trama	byte 0	byte 1	byte 2	...	...	byte 238

Figura 9. Trama SDN modificada

4.2.4 Liberación de la conexión

Para finalizar la transmisión de imágenes, el proceso ProfiCam utiliza la primitiva liberación, lo que provoca que ProfiLink genere un servicio SDA con un código 2 (Tabla 1), que garantice la liberación de los recursos consumidos por todos los procesos.

5 Experimentos y Medidas.

Para estudiar la viabilidad del sistema propuesto se han realizado medidas sobre el escenario mostrado en la figura 5.

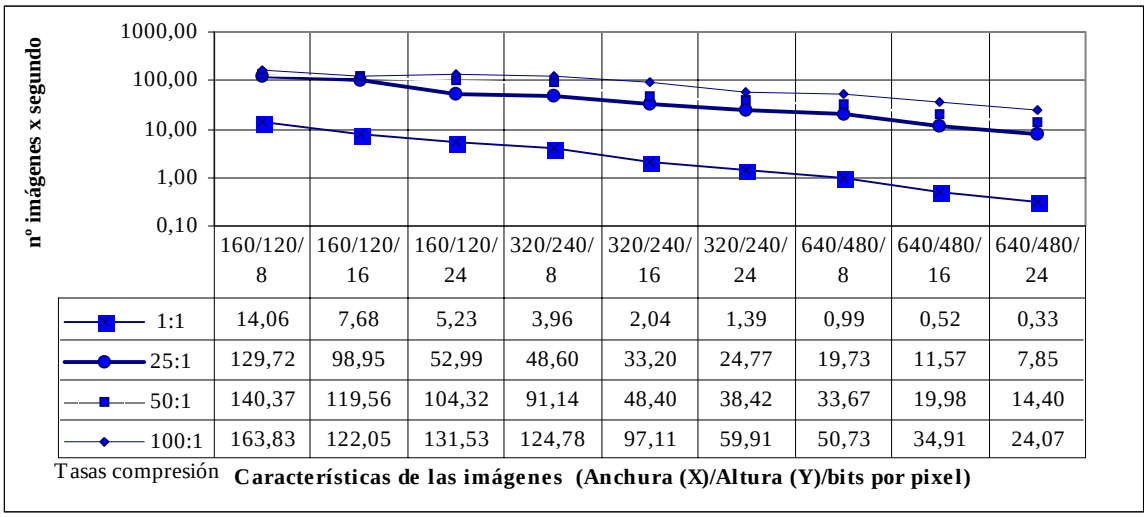


Figura 10. Nº de imágenes por segundo que la red Profibus puede transmitir, con una carga de tráfico de control del 50%, para cada una de las combinaciones de características de imágenes analizadas.

El tráfico de fondo se ha creado estableciendo enlaces entre los clientes e inyectando peticiones al medio tanto periódicas como aperiódicas, hasta alcanzar una tasa de ocupación media del 50 %.

Se han utilizado imágenes combinando los valores de tasa de compresión, tamaño (en X e Y) y número de bits por pixel mostrados a continuación:

Anchura y altura: 160x120, 320x240, 640x480
Bits por píxel: 8 (B&N), 16 (Color), 32 (Color)
Tasas de compresión: 1:1,25:1,50:1,100:1

resultando un total de 36 experimentos diferentes. Estas características son representativas de un amplio abanico de aplicaciones industriales de la visión artificial, cubriendo un rango que va desde imágenes en blanco y negro de baja resolución, cuyo tamaño es aproximadamente de 19200 bytes, hasta imágenes color RGB de media resolución cuyo tamaño es de 921.600 bytes.

La figura 10 representa el número de imágenes por segundo que la red Profibus es capaz de procesar bajo las características definidas anteriormente.

En cuanto a la tasa de compresión utilizando JPEG, el valor máximo aceptable vendrá dado por la aplicación, siendo 25:1 un valor típicamente aceptado para la mayoría de la aplicaciones, y que en este caso vuelve a ser el límite. Como ejemplo (Fig. 11), se han utilizado imágenes de integrados en los que se puede ver que, aplicando el mismo proceso de binarización, una compresión 50:1 empezaría a producir problemas en la detección del tipo de integrado, mientras que una compresión 100:1 introduciría una gran cantidad de errores en el proceso de identificación.

6 Conclusiones y trabajos futuros.

Como resultado de los experimentos realizados sobre la red Profibus, se puede concluir que este tipo de sistemas (Tipo 3) son viables sin utilizar técnicas de compresión, en el caso de ser admisible en la aplicación tasas de una imagen por segundo, siempre que el número de bytes que forman la imagen sea inferior a 300.000 aproximadamente, lo que impide la utilización de este sistema con imágenes de resolución tipo medio. En los casos en que la tasa de llegada de imágenes sea superior a las 5 imágenes por segundo, sólo es posible la transmisión de imágenes de muy baja resolución, aproximadamente de menos de 60.000 bytes.

La capacidad de transmisión de imágenes de la red Profibus, cuando no se utiliza compresión, es por tanto, relativamente baja, lo que limita el número de aplicaciones industriales donde sea viable su utilización.

Siendo aceptable para la aplicación la degradación de la imagen producida por un proceso de compresión 25:1, el incremento del número de imágenes obtenido es muy significativo. En estos casos, el método propuesto es válido para la transmisión de más de 5 imágenes en todas las combinaciones analizadas y permitiendo además, la utilización del sistema en procesos de muy alta velocidad que admitan bajas resoluciones. Bajo estas premisas se puede llegar prácticamente a las 130 imágenes por segundo en blanco y negro, y entre 53 y 99 imágenes en color, dependiendo si la codificación es de 24 o de 16 bits.

El sistema de Tipo 3 se ha simulado mediante cámaras conectadas a ordenadores y estos a su vez a la red Profibus dado que no existen

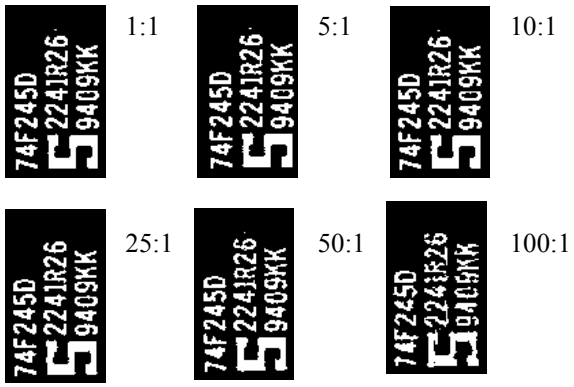


Figura 11. Procesado de la imagen recibida con distintos ratios de compresión

dispositivos en el mercado que se conecten directamente. En este momento trabajamos en el desarrollo de un sensor capaz de capturar imágenes y de realizar todas las funciones de entramado y transmisión en una red Profibus. Paralelamente, es necesario incorporar una etapa de compresión hardware capaz de procesar el flujo de entrada a la velocidad que la red es capaz de transmitir.

Agradecimientos

Este trabajo ha sido desarrollado por miembros del grupo DATA (Desarrollo de Aplicaciones Telemáticas Avanzadas) perteneciente a la Universidad Politécnica de Valencia, en el contexto del proyecto CICYT TIC98-0495-C02-02

Referencias

[1] Cavalieri S.; Di Stefano, A.; Mirabella O. "Impact of Fieldbus on Communication in Robotic Systems". IEEE Trans. on Robotics and Automation. Vol. 13, nº 1, pág. 30-48 February 1997.

[2] Maaref, B; Nasri, S; Sicard, P. "Communication System for Industrial Automation". Procedings of the IEEE International Symposium on Industrial Electronics. ISIE'97, Guimaräes, Portugal. 1997, Vol. 3, pág. 1286-1291

[3] Dalgiç, I: Tobagi, F.A. "Evaluation of 10Base-T and 100Base-T Ethernets Carryng Video, Audio and Data Traffic". IEEE INFOCOM 94, Toronto, Canada, June 1994.

[4] W.B. Pennebaker, J.L. Mitchel. "JPEG Still Image Data Compression Standard". VNR 1992.

[5] Decotignie (J.D.), Pleinevaux (P.) "A survey on industyrial communications networks". Annals of Telecommunications, Nº 9-10 (1993), pp 435-448.

[6] Jeffry Richter. "Programación avanzada en Microsoft-Press 84-481-1160-5

[7] Kaneko, H; Stankovic, A; Sen,S; Ramamritham, K. "Integrated Scheduling of Multimedia and Hard Real-Time Task". Dpto. Computer Science. U. Massachusetts. Technical Report 96-45.

[8] Irwin, J.David. "Emerging Multimedia Communication for Industry Applications". Procedings of the IEEE International Symposium on Industrial Electronics. ISIE'99. Vol. 1, pág 1-6.

[9] Wu, Chwan-Hwa; Irwin, J. David. "Multimedia and Multimedia Communication: A Tutorial". IEEE transactions on Industrial Electronics. 1998, Vol 45 nº1, pag. 4-14.

[10] Profibus, "Profibus Standard DIN 19245 Part I and II". Translated from German, Profibus Nutzerorganisation e.V,1992.

Predicción de tráfico de Internet y aplicaciones

I.Bernal, J.Aracil, D.Morato, M.Izal, E.Magaña y L.A. Díez
Departamento de Automática y Computación. Universidad Pública de Navarra
Grupo de Redes, Sistemas y Servicios telemáticos
Campus Arrosadía - 31006 Pamplona (Navarra)
Teléfono: 948 168904 Fax: 948 168924
E-mail: javier.aracil@unavarra.es

Abstract *In this paper we focus on traffic prediction as a means to achieve dynamic bandwidth allocation in a generic Internet link. Our findings show that coarse prediction (bytes per interval) proves advantegeous to perform dynamic link dimensioning, even if we consdier a part of the top traffic producers in the traffic predictor.*

1 Introducción

Hoy en día estamos asistiendo a un crecimiento imparable del tráfico de Internet. Ante tal demanda es un hecho que las operadoras desean dar calidad de servicio a sus usuarios y para ello es preciso dimensionar los enlaces. Los problemas del tráfico de Internet son muy diferentes de los de otros tipos de tráfico [1] y plantean un escenario de especial complejidad. En concreto tenemos que el tráfico de Internet presenta autosimilitud (*self-similarity*) y no estacionariedad.

Por el contrario, el tráfico telefónico es *de incrementos independientes* y por lo tanto aplican modelos de tipo */G/G/1*. En el entorno de redes de banda ancha el ancho de banda efectivo se calcula con modelos de Markov en varias escalas de tiempo. Pero sin embargo, debido a la fuerte no estacionariedad y autosimilitud del tráfico de Internet, no existe hoy en día una teoría de dimensionamiento de enlaces de Internet.

1.1 Autosimilitud

Para entender bien lo que significa autosimilitud es necesario repasar conceptos básicos de independencia estadística. Sea $X_1, X_2, \ldots X_n$ una muestra de n variables aleatórias independientes con media μ y desviacion estándar σ. Se cumple que:

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \qquad (1)$$

$$var(\overline{X}) = \sigma^2 n^{-1} \qquad (2)$$

Por otro lado, sea el proceso de cuentas de paquetes X_i en intervalos de duración δ, que mostramos en la figura 1.

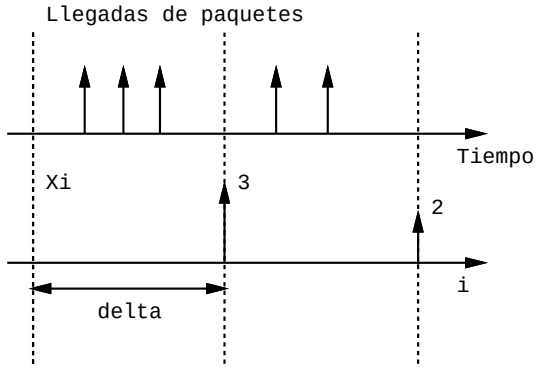


Figura 1: Cuentas por intervalos

Formamos ahora el proceso de agregación de intervalos (media muestral)

$$S_i^m = \frac{X_{im-m+1} + \cdots + X_{im}}{m} \qquad i = 1 \ldots \lfloor \frac{n}{m} \rfloor \qquad (3)$$

y observamos que no cumple el Teorema Central del límite en el caso de tráfico Internet, según el cual la varianza debe decaer con el número de muestras m en una proporción m^{-1}. La figura 2 muestra la varianza frente al nivel de agregación en coordenadas logarítmicas en ambos ejes para una traza real y para un caso de incrementos independientes. Observamos que la varianza decae *lentamente* en comparación con un proceso de incrementos independientes, con la forma:

$$Var(S_i^m) = \sigma^2 m^{-\beta} \qquad (4)$$

con $0 < \beta < 1$.

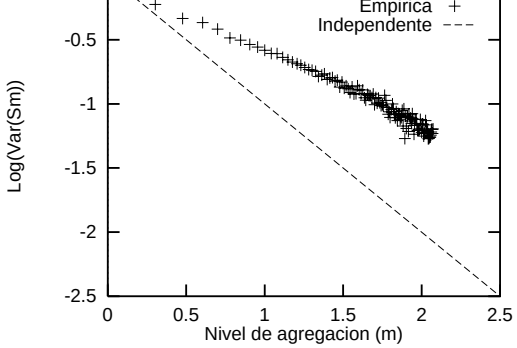


Figura 2: Varianza frente a nivel de agregación

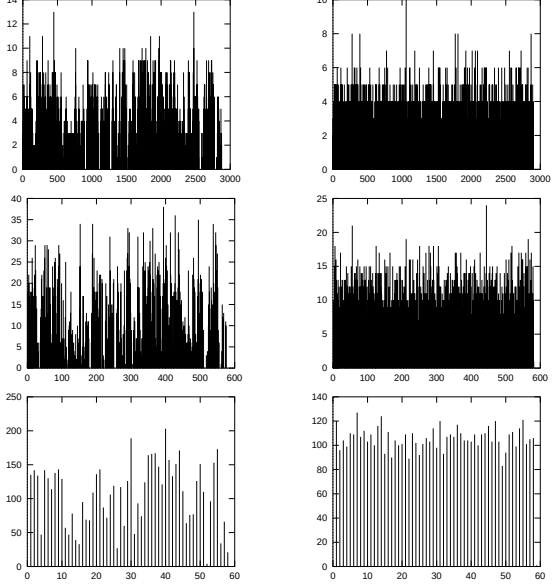


Figura 3: Trafico de Internet frente a Poisson

La varianza decae de forma lenta simplemente porque no se cumplen las hipótesis del teorema central del límite. En concreto, las variables aleatorias que contribuyen a la media muestral (bytes por intervalo) no son independientes. Por el contrario, se cumple que el proceso de llegadas de tráfico es asintóticamente autosimilar de segundo orden. Sea $\rho^{(m)}(j)$ la autocorrelación (*lag* j, $j > 1$) de S_i^m. Tenemos que: Tenemos que:

$$\lim_{m\to\infty} \rho^{(m)}(j) = \frac{1}{2}((j+1)^{2H} - 2j^{2H} + (j-1)^{2H}) \tag{5}$$

Un proceso asintóticamente autosimilar de segundo orden sufre *dependencia a largo plazo*. Esto es, la autocorrelación del proceso decae lentamente y no es sumable, en contraste con procesos Poissonianos/Markovianos. Es importante observar que la dependencia a largo plazo es una propiedad asintótica: no importan los valores *absolutos* de la autocorrelación sino la *forma* de la misma.

En la figura 3 mostramos el efecto de la dependencia a largo plazo. En las gráficas de la izquierda tenemos tráfico de internet frente al tráfico de Poisson en las de la derecha, para varias escalas de tiempo de 10ms., 100ms. y 1 seg (ver ecuación 3). La caida lenta de la varianza provoca *ráfagas en cualquier escala de tiempo*, al contrario que eun proceso de Poisson donde tenemos una suavización hacia la media conforme vamos agregando bytes por intervalo y formadon así el tráfico en escalas de tiempo mayores.

1.2 Causas de la dependencia a largo plazo

La dependencia a largo plazo se produce por el efecto del multiplex de fuentes on-off con varianza infinita [2], como se muestra en la figura 4. Cada una de estas rafagas (conexiones TCP, por ejemplo) introduce correlación en la escala de tiempo de su duración. Estudios experimentales demuestran que las ráfagas que vienen de la transmisión de ficheros en Internet tienen varianza infinita [3].

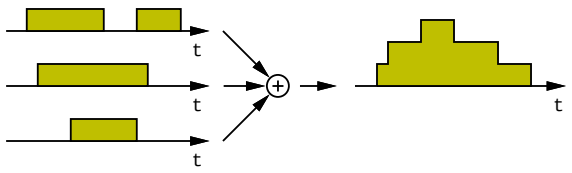


Figura 4: Superposición de fuentes on-off

Para que la varianza de la duración de la ráfaga sea infinita la distribución de la misma debe seguir la forma:

$$P(X > t) \sim Kt^{-\alpha} \qquad 1 < \alpha < 2 \tag{6}$$

que se observa que es el caso para conexiones reales de Internet, como mostramos en la figura 5. En esta figura se muestra la distribución (función de supervivencia) de la duración de conexiones FTP.

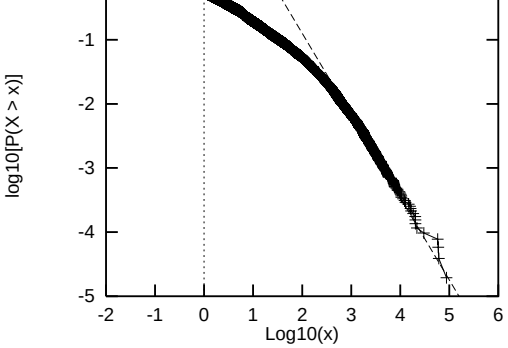
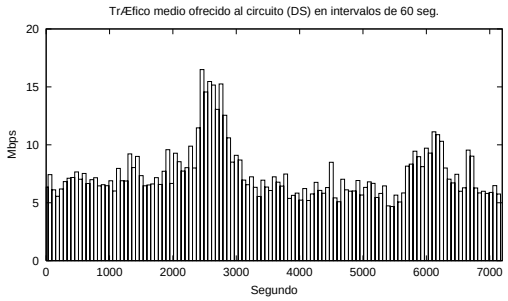


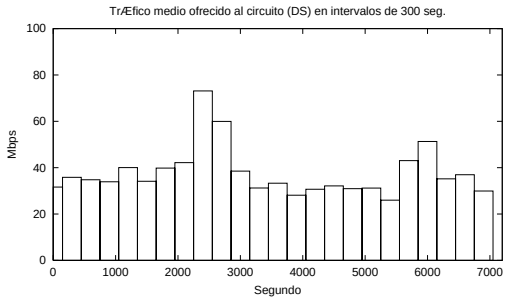
Figura 5: Duracion de conexiones FTP

1.3 No estacionariedad

Por otro lado, el tráfico de Internet adolece de una fuerte no estacionariedad, como se observa en la figura 6, que muestra varias escalas de tiempo de una traza de tráfico real. En conclusión, las características de alta intermitencia (dependencia a largo plazo) y no estacionariedad hacen que el dimensionamiento a-priori de enlaces de Internet sea difícil de realizar en la práctica.



Intervalo 60 segundos



Intervalo 300 segundos

Figura 6: No estacionariedad

2 Planteamiento del problema

De los apartados anteriores observamos que el tráfico de Internet muestra dependencia y no estacionariedad. El modelado solo es posible en estadísticos de primer y segundo orden y eso no es suficiente para una correcta estimación de los recursos a asignar. Ante estas condiciones de tráfico fuertemente dinámico cabe pensar en otros métodos que hagan del problema de la correlación una ventaja. En concreto, los algoritmos de predicción

alta correlación.

Sea δ el intervalo de tiempo de predicción y sean X_k y $\hat{X_k}$ los tráficos (número de bytes) real y estimado en el intervalo. Probaremos una serie de estimadores de implementación sencilla para $\hat{X_k}$. En concreto, utilizaremos el *Método interpolador de Lagrange*, ya que es un estimador lineal simple que se puede usar con intervalos de longitud constante (como es nuestro caso), que sigue la expresión :

$$p(x) = \sum_{i=1}^{n+1} y_i \prod_{\substack{j=1 \\ j \neq i}}^{n+1} \frac{x - x_j}{x_i - x_j} \qquad (7)$$

Particularizaremos para obtener polinomios de orden n con n<3 y obtenemos ecuaciones sencillas lineales, de coste computacional muy reducido para un hipotético asignador de ancho de banda localizado en un router o conmutador que gobierna un enlace.

Para medir la bondad del estimador podemos usar la distribución de probabilidad del error $\hat{X_k} - X_k$. Pero es todavía más interesante el retardo en cola para un servidor con capacidad variable $\frac{\hat{X_k}}{\delta}$. Esta última medida no sólo tiene en cuenta el error instantáneo sino también el acumulado y modela mejor un escenario real de predicción.

Por otro lado, es interesante estudiar no solo el caso de predicción con el total del tráfico sino predicción basada en un subconjunto de usuarios. Esto puede ser muy flexible en el caso de topologías de red donde todo el tráfico no pasa por un solo punto. Las fuentes más activas pueden informar a los routers en el camino extremo a extremo del tráfico que van a enviar, en sintonía con estandares recientes de conmutación por etiquetas [5]. De este modo la predicción no sólo es útil en enlaces de acceso sino en topologías genéricas de red. De hecho la fuerte no homogeneidad de los usuarios ayuda a predecir en base a *parte* del tráfico. La figura 7 muestra el porcentaje de tráfico del enlace frente al porcentaje de usuarios que lo producen.

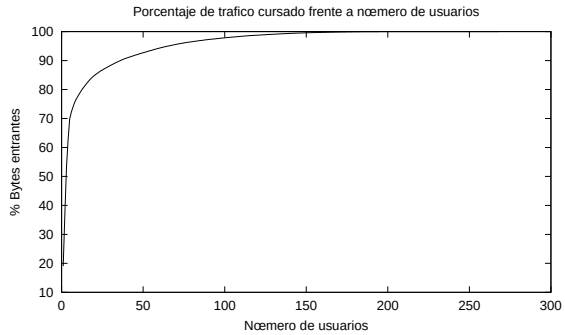


Figura 7: Porcentaje de tráfico frente a porcentaje usuarios que lo generan

Se observa claramente que es posible predecir con un porcentaje pequeño de usuarios y no con

regulares. La figura 8 muestra el usuario más activo de la muestra. Prácticamente transmite a tasa constante.

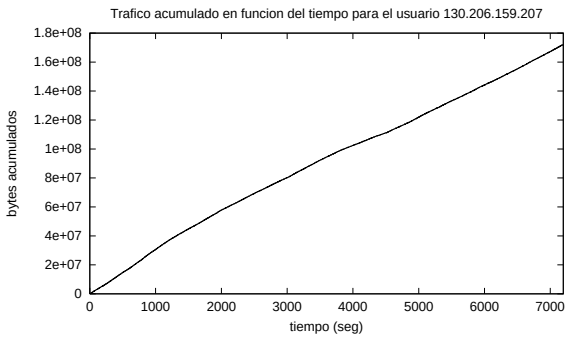
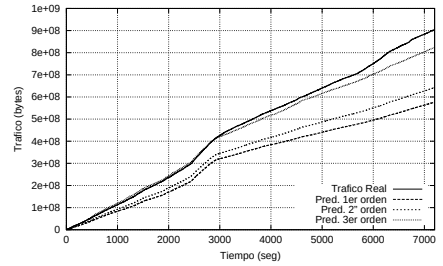


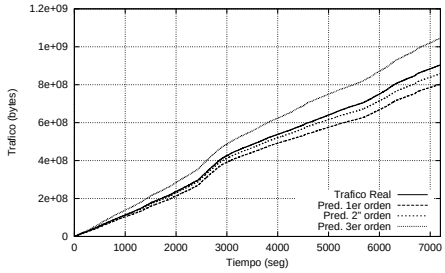
Figura 8: Usuario más activo

3 Resultados

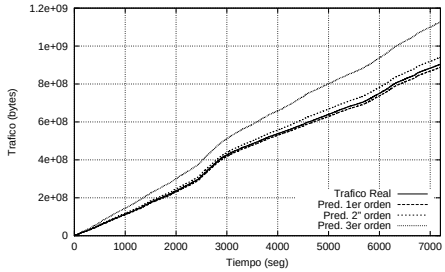
Los resultados preliminares se obtienen con Polinomios interpoladores de Lagrange de primer,segundo y tercer orden en intervalos de 1 segundo. Este intervalo de un segundo es un tiempo superior a RTTs típicos en la Internet y permite que el asignador de ancho de banda (un conmutador ATM con ABR por ejemplo) tenga tiempo suficiente para adecuar las condiciones del circuito a la nueva carga de tráfico. En primer lugar la figura 9 muestra una comparación visual de tráfico real frente a tráfico obtenido mediante predicción.



Predicción con 4 usuarios



Predicción con 30 usuarios



Predicción con 100 usuarios

Se observa que la predicción con un número reducido de usuarios (30 a 100) obtiene buenos resultados. El número total de usuarios en la muestra es de 300. Este resultado se relaciona perfectamente con el resultado observado en la figura 8, en la cual podemos observar como los 30 usuarios más activos, generan más del 80% del tráfico total. De hecho se puede observar como la distribución de probabilidad error de predicción, se estabiliza bastante al utilizar un número de usuarios para la predicción superior a 30, como se muestra en la figura 10

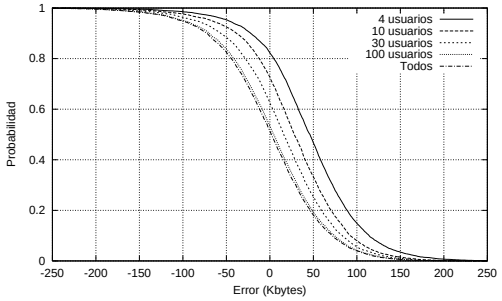
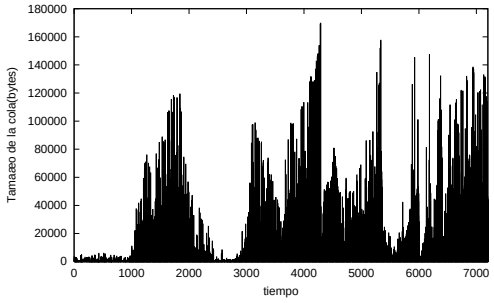


Figura 10: Función de densidad del error de predicción

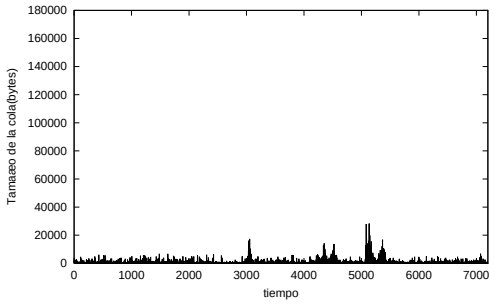
Más interesante todavia es el número de usuarios en cola, obtenido mediante simulación de la ecuación de Lindley:

$$Q_{n+1} = max\left\{Q_n + A_{n+1} - C_{n+1}, 0\right\} \qquad (8)$$

donde Q_n es el número de bytes en cola intervalo n, A_n es el número de bytes que llegan en intervalo n y C_n es la capacidad del servidor en el instante n.



N=10 usuarios



N=30 usuarios

Figura 11: Número de usuarios en cola

En nuestro caso $C_n = A_n$. La figura 11 muestra el número de bytes en cola Q_n en el intervalo de medida. Se observa que con 30 usuarios se consigue estabilizar la cola en torno a valores muy bajos (menores de 30 KBytes), mostrando la viabilidad práctica de la idea.
En la figura 12 comparamos la predicción del tráfico real con un FBM (Fractional Brownian Motion) [4], que es un proceso gaussiano asintóticamente autosimilar de segundo orden, muy utilizado para modelar tráfico de Internet. La figura 12 muestra en este caso la función de supervivencia $P(X > x)$ del retardo en cola.

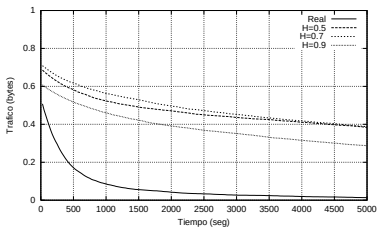


Figura 12: Comparación con un FBM

Observamos que los FBM modelan bien la dependencia a largo plazo *pero no a corto plazo*. Este resultado apunta en la dirección de modelado con fuentes de dependencia a corto plazo, tipo cadenas de Markov para aquellos escenarios donde el parametro de relevancia sea la dependencia a corto y no a largo plazo.

Los resultados anteriores muestran el caso en que se predice el número de bytes que llegan en un intervalo. Es el caso más sencillo y queda por estudiar que ocurre con las características del tráfico dentro del intervalo, que puede presentar escalado multifractal [6]. Posiblemente estas características dentro del intervalo hacen que la predicción sea muy grosera al no tenerlas en cuenta y es necesario introducir más parámetros aparte de los bytes en bruto. Sin embargo existen múltiples escenarios donde *si la red dispone de una estimación de los bytes por intervalo es suficiente para mejorar en gran medida las prestaciones.* Una posible aplicación son los entornos de "Burst Switching" donde, gracias a la predicción, es posible paralelizar el tiempo de paquetización con la reserva de recursos. Al conocer de antemanos los bytes a transmitir, gracias a la predicción, se envia el mensaje de reserva de recuros antes de comenzar la paquetización, como se muestra en la figura 13.

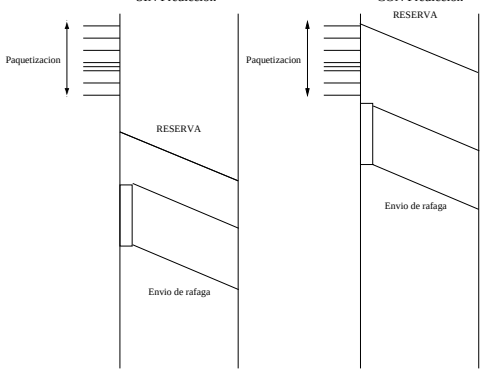


Figura 13: Optical Burst Switching

4 Conclusiones y trabajos futuros

En este artículo hemos presentado métodos de predicción de coste computacional muy bajo que pueden ser utilizados para dimensionar dinámicamente enlaces de la Internet que requieren una estimación de bytes por intervalo. Queda como trabajo futuro el análisis de prestaciones con distintos modelos de tráfico dentro de cada intervalo y la selección de algoritmos de predicción óptimos en este último caso.

Referencias

[1] K. Park and W. Willinger (Editors). *Self-similar Network Traffic and Performance Evaluation.* Wiley Interscience, 2000.

[2] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-similarity through high-Variability: Statistical analysis of ethernet LAN traffic at the source level. *IEEE/ACM Transactions on Networking*, 5(1), Febrero 1997.

[3] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic: Evidence and possible causes. *IEEE/ACM Transactions on Networking*, 5(6):835–846, Diciembre 1997.

[4] I. Norros. On the Use of Fractional Brownian Motion in the Theory of Connectionless Networks. *IEEE Journal on Selected Areas in Communications*, 13(6):953–962, Agosto 1995.

[5] *IEEE Communications Magazine* Special Issue on MPLS, Diciembre 1999

[6] A. Erramilli, O. Narayan and A. Neidhart Performance Impacts of Multi-Scaling in Wide Area TCP/IP Traffic *INFOCOM 2000*, Tel Aviv, Israel

Un modelo de tráfico agregado para redes ATM

Sebastià Galmés y Ramon Puigjaner
Departament de Matemàtiques i Informàtica. Universitat de les Illes Balears.
Cra. de Valldemossa, km. 7.5, 07071 Palma
Teléfonos: 971 172 989, 971 173 288, Fax: 971 173 003
E-mail: {dmisgo0, putxi}@uib.es

***Abstract**. This paper summarizes the work performed by the authors during last years, including the most recent contributions, around on/off models. On/off models have been extensively used in the literature for describing the ATM source traffic, including new observed characteristics in the context of self-similarity, such as long range dependence. This is basically due to their general definition and analytical tractability. This paper introduces the batch-on/off model and describes the main related results. The batch-on/off model is the natural extension of the single on/off source that allows for the generation of more than one cell during active slots. Hence, in addition to the distributions of the busy and idle periods, a batch size distribution needs to be specified. As shown in the paper, batch-on/off models constitute a very powerful tool in research, since they retain the analytical tractability of single sources and to a large extent provide simple, closed-form and exact or very accurate results. Particularly, exact expressions are provided for two types of correlations structures: the correlation of the related counting process and the correlation of the sequence of interarrival times between batches. Also, the response time distribution of a (Batch-on/off)/D/1 queue is presented, which can be decomposed in two terms that are separately described. All these results constitute the basis for developing a software platform to analyze in more detail the relationships between broadband traffic descriptors and performance parameters.*

1 Introducción

En el área de la ingeniería de tráfico, los modelos *on/off* fueron introducidos para describir la naturaleza variable de las fuentes de tráfico. Particularmente, estos modelos fueron frecuentemente utilizados para representar el tráfico de paquetes de voz [1,2] y diferentes formas de comunicación de datos [3-6]. La duración de los períodos activo (*active, busy, on, talkspurt*) e inactivo (*inactive, idle, off, silence*) se suponía distribuida exponencialmente (o geométricamente) y, en consecuencia, la caracterización de estos modelos se basaba solamente en tres parámetros: el período activo promedio, el período de silencio promedio y la velocidad de pico durante los períodos activos. La hipótesis exponencial (o geométrica) permitía el tratamiento analítico de problemas relacionados tales como el análisis del comportamiento de colas o la determinación de otros descriptores de tráfico. Sin embargo, siendo tal hipótesis muy aceptable en el caso de las fuentes de voz, resultaba demasiado restrictiva en la captura del tráfico de datos. Por ello, en [7] se propuso un modelo *on/off* con distribuciones genéricas para los períodos activo y de silencio. Estas nuevas hipótesis no imposibilitaban el tratamiento analítico, ya que en este caso los modelos *on/off* podían ser caracterizados mediante una cadena de Markov que incluyera en su descripción de estados, no solamente el estado de actividad de la fuente, sino también el tiempo en curso en cada estado de actividad. A partir de este momento, se abrió una cierta actividad investigadora en torno a este tipo de modelos de tráfico.

El comportamiento de un multiplexor alimentado por la superposición de fuentes *on/off* genéricas fue también tratado en [7], desde el punto de vista del comportamiento asintótico aproximado de la distribución de la longitud de cola. Desde entonces, han ido surgiendo en la literatura trabajos más completos. En [8] se desarrolla una representación markoviana del estado de ocupación del multiplexor anterior y se propone un algoritmo numérico para resolverlo. Sin embargo, la complejidad computacional de este algoritmo debida al enorme número de estados de la cadena lo convierte a menudo en impracticable. Por ello, en [9] se propone una aproximación más heurística del problema y en [10] se obtienen resultados todavía más aproximados y explícitos.

En el ámbito de los descriptores de tráfico, también se han llevado a cabo interesantes estudios en torno a los modelos *on/off*. En [11] se proponen dos métodos aproximados para determinar el ancho de banda efectivo de los modelos *on/off* con distribuciones genéricas. En [12] se analiza el proceso de conteo asociado desde el punto de vista de la función de autocovarianza, la densidad espectral de potencia y el índice de dispersión de cuentas. El análisis realizado, basado en el método de la función generatriz y la teoría de residuos, más que expresiones explícitas, proporciona algoritmos de cálculo de las funciones anteriores. Sin embargo, en [13] se obtienen expresiones explícitas de la

varianza y la autocorrelación de otro tipo de proceso asociado, el de la secuencia de intervalos entre llegadas consecutivas. Particularmente, se observa que esta función de autocorrelación depende de la distribución completa del período activo y de sólo los dos primeros momentos de la distribución del período de silencio. Más adelante, y en consonancia con este último trabajo, se desarrolla en [14] un algoritmo para ajustar un conjunto finito de coeficientes de la función de autocorrelación de los intervalos entre llegadas de una fuente arbitraria. Este algoritmo es capaz de producir resultados prácticamente exactos y es particularmente útil para ajustar funciones de autocorrelación positivas.

En la segunda mitad de la década de los 90, el área de la ingeniería de teletráfico ha evolucionado notablemente en dos direcciones: por un lado, se ha avanzado cualitativamente en la caracterización del tráfico de banda ancha y, por otro, se han desarrollado descriptores y modelos de tráfico más sofisticados afines a las nuevas características. En este contexto, probablemente la contribución más significativa ha sido el descubrimiento de la naturaleza fractal o de autosemejanza del tráfico de banda ancha. Aunque dicho fenómeno fue inicialmente observado en el tráfico Ethernet [15], los resultados se extendieron rápidamente a otros tipos de tráfico, tales como el tráfico de vídeo a velocidad variable o VBR (*Variable Bit Rate*) [16]. Realmente, la frctalidad del tráfico introduce importantes cambios cualitativos y cuantitativos en la teoría de teletráfico, hasta el punto de convertir en obsoletos gran parte de los modelos de tráfico hasta entonces utilizados. Sin embargo, la generalidad con que se definen los modelos *on/off* les ha permitido "sobrevivir" a estos cambios, y de hecho diversos estudios han demostrado la capacidad de estos modelos para capturar el efecto Noah (distribuciones de caída lenta) y el efecto Joseph (correlaciones fuertes a largo plazo), características típicamente observadas en el contexto del fenómeno fractal. En [17] se demuestra que la superposición de modelos *on/off* cuyas distribuciones de los períodos activo y de silencio presentan una caída lenta, produce tráfico agregado que exhibe el comportamiento fractal. En [18] se establece que para observar el comportamiento fractal en una sola fuente *on/off* es suficiente con que la duración de uno de sus dos estados presente una distribución de caída lenta. Finalmente, varios estudios han mostrado que los modelos *on/off* con distribuciones de caída lenta son especialmente adecuados para describir la naturaleza de algunas transferencias de datos, como las que tienen lugar en las sesiones web [19].

Este artículo se centra en el llamado modelo *batch-on/off*, que no es más que la extensión natural de la fuente *on/off* individual para describir tráfico agregado. El objetivo es ofrecer una descripción global de esta clase de modelos, ubicándolos en el conjunto de los modelos matemáticos para tráfico ATM agregado, y mostrando los resultados más significativos obtenidos en torno a ellos, tanto en el aspecto de la descripción del tráfico generado, como en el de la evaluación de comportamiento. Así pues, este artículo resume el trabajo desarrollado por sus autores durante los últimos años, e incluye las aportaciones más recientes.

En la actualidad, es un hecho ampliamente aceptado que la bondad de un modelo matemático para representar una determinada carga de tráfico, reside en su capacidad para capturar su estructura de correlación. De hecho, las funciones de autocorrelación en sus diversas formas (intervalos entre llegadas o número de cuentas) constituyen un ingrediente indispensable en la definición del grado de ráfaga, lejos ya de aquellas definiciones tan simplistas (cociente entre la velocidad de pico y la velocidad media, o varianza de los intervalos entre llegadas). Así pues, en lo que se refiere a la descripción del tráfico producido por un modelo *batch-on/off*, ésta se centra en las diversas funciones de autocorrelación. Y en cuanto a la evaluación de comportamiento, se considera una cola del tipo (*Batch-on/off*)/D/1, cuyo tiempo de servicio equivale a la duración de una celda ATM. En el apartado 2 de este artículo, se define el modelo *batch-on/off* y se relaciona con otros modelos de tráfico agregado introducidos en la literatura. En el apartado 3, se describen los resultados obtenidos para las diversas funciones de autocorrelación, es decir, la autocorrelación del número de cuentas y la de los intervalos entre llegadas. En el apartado 4, se muestran los resultados obtenidos al evaluar el comportamiento de una cola (*Batch-on/off*)/D/1, y se describen las diferentes contribuciones a estos resultados. Finalmente, en el apartado 5 se exponen las conclusiones más significativas y se sugieren líneas de investigación futuras.

2 El Modelo *Batch-on/off*

En este apartado se introduce el modelo *batch-on/off* y se relaciona con otros modelos matemáticos conocidos para tráfico agregado.

2.1 Definición

Un modelo *on/off* está formado por dos tipos de estados o períodos alternantes, uno activo y otro de silencio. Durante los períodos activos, las celdas se generan a velocidad constante, mientras que durante los períodos de silencio no se generan celdas. Tanto los sucesivos períodos activos, como los sucesivos períodos de silencio, se suponen idénticamente distribuidos e independientes, y ambas distribuciones son arbitrarias. Además, los períodos activo y de silencio se generan también de forma independiente entre sí. En definitiva, un modelo *on/off* se puede ver como dos procesos de renovación independientes que se alternan. En este

artículo, se va a suponer que durante los períodos activos las celdas se generan a la velocidad de pico, es decir, una tras otra.

Un modelo *batch-on/off* generaliza la definición anterior al permitir la generación de un grupo (lote o *batch*) de celdas en cada ranura (*slot*) de tiempo de un período activo. Por tanto, en su definición deberá incluirse la distribución del tamaño de los lotes (*batch size*), que podrá ser arbitraria. La Fig. 1 muestra una posible realización de un modelo *batch-on/off*.

Sean *a*, *s* y *b* las variables aleatorias que representan respectivamente las duraciones de los períodos activo y de silencio (en número de ranuras) y el tamaño de *batch* (en número de celdas). Estas variables aleatorias son discretas sobre un conjunto de enteros no negativos hasta un cierto valor máximo: *A*, para el período activo (máximo tamaño de ráfaga), *S* para el período de silencio (máximo período de silencio) y *B* para el tamaño de *batch* (generalmente el número de fuentes agregadas).

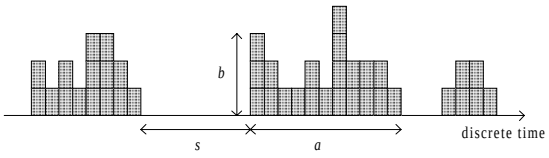


Figura 1: Ejemplo de un modelo *batch-on/off*

Particularmente, cualquiera de estos límites puede ser infinito. Las correspondientes funciones de probabilidad, p_a, p_s y p_b (o las funciones acumuladas de probabilidad F_a, F_s y F_b) serán consideradas como datos de entrada en los análisis que vamos a realizar. La intensidad o carga promedio de tráfico vendrá dada por la siguiente expresión:

$$\rho = \frac{E(b) \cdot E(a)}{E(a) + E(s)} \qquad (1)$$

donde $E(\cdot)$ designa la esperanza o valor medio.

2.2 El Modelo *Batch-on/off* en el Mapa de Modelos Agregados

A fin de ofrecer una perspectiva global, vale la pena situar el modelo *batch-on/off* en el mapa de modelos de tiempo discreto para tráfico ATM agregado. En la Fig. 2 se muestra dicho mapa.

No es el propósito de este subapartado realizar una descripción pormenorizada de todos los modelos que aparecen en la Fig. 2, sino más bien poner de manifiesto la relación del modelo *batch-on/off* con los restantes. Como se muestra en la figura, los modelos D-BMAP (*Discrete – Batch Markovian Arrival Process*) constituyen una clase muy general que comprende la mayor parte de modelos introducidos en la literatura [20]. En un D-BMAP, se asocia una distribución de tamaño de *batch* a cada transición de una cadena de Markov subyacente. Esta genérica definición no impide, sin embargo, obtener resultados exactos o muy aproximados en los problemas de evaluación de comportamiento relacionados, ya que una poderosa maquinaria matemática ha sido desarrollada en torno a esa clase de modelos. No obstante, esta maquinaria se basa en la aplicación de algoritmos notablemente complejos, y por ello la relación entre los parámetros de entrada y las variables de comportamiento no se vislumbra claramente. Además, puesto que la cadena de Markov subyacente carece de significado físico, esta clase de modelos no es la más adecuada para capturar trazas de tráfico real. En cambio, los modelos *batch-on/off* resultan más fáciles de analizar y, tal como se verá más adelante, permiten obtener expresiones matemáticas que reflejan de un modo mucho más explícito la relación entre parámetros de entrada y salida. Además, la cadena de Markov que caracteriza estos modelos está estrechamente relacionada con el estado de actividad del tráfico que se describe, así pues, resulta mucho más viable plasmar las características de una traza real en los parámetros del modelo.

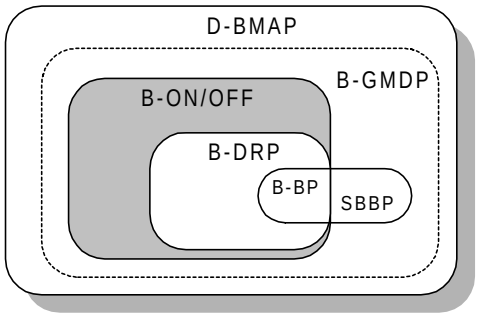


Figura 2: Mapa de los modelos en tiempo discreto de tráfico ATM agregado

En la Fig. 2 puede observarse que entre los modelos D-BMAP y los modelos *batch-on/off*, aparece una clase intermedia, conocida como los B-GMDP (*Batch-General Modulated Deterministic Process*), que podrían constituir la extensión natural para tráfico agregado de los modelos GMDP ya introducidos en la literatura. Obsérvese también en la figura que los modelos *batch-on/off* contienen como casos particulares otras clases de modelos ya conocidos, tales como los B-DRP (*Batch-Discrete Renewal Process*) y los B-BP (*Batch-Bernouilli Process*). Por tanto, los resultados que vayan a obtenerse de forma general para los modelos *batch-on/off* van a ser válidos también para estos últimos. Nótese que los modelos B-BP están en la intersección de los B-DRP y los SBBP (*Switched Batch Bernouilli Process*), una clase especial de modelos B-GMDP. Existen interesantes trabajos en la literatura sobre modelos de tráfico para redes

ATM en los que se describen de forma más detallada los modelos que aparecen en la Fig. 2 [21, 22].

3 Funciones de autocorrelación

En este apartado se describen los resultados correspondientes a dos tipos de funciones de autocorrelación: la función de autocorrelación del proceso de cuentas asociado al modelo, y la función de autocorrelación de la secuencia de intervalos entre llegadas de grupos de celdas consecutivos. Estas funciones constituyen los ingredientes básicos de sendos descriptores del grado de ráfaga, el Índice de Dispersión de Cuentas (IDC) y el Índice de Dispersión de Intervalos (IDI) [1,2,23]. Estas referencias constituyen excelentes trabajos sobre los índices de dispersión y la importancia relativa de cada uno de ellos. No obstante, la literatura no ha discernido hasta el momento la relación existente entre los dos tipos de funciones de autocorrelación mencionados anteriormente, y por tanto entre el IDC y el IDI. Dependiendo del tipo de problema a tratar, puede resultar más eficaz la utilización de un índice de dispersión u otro para describir el tráfico de celdas ATM, pero por lo general la investigación se ha decantado con más frecuencia hacia la caracterización mediante el proceso de cuentas y correspondientemente el IDC.

3.1 Autocorrelación del proceso de cuentas

La evolución en el tiempo de un modelo *batch-on/off*, en lo que a sus estados de actividad se refiere, es la que corresponde a la fuente *on/off* individual (sólo que en el caso del modelo agregado, en cada estado activo puede generarse más de una celda). La fuente *on/off* individual con distribuciones arbitrarias fue descrita por vez primera por Sohraby en 1993 [7]. Para ello construyó una cadena de Markov $\{v(n):n\geq 0\}$, donde $v(n)$ representa el estado de actividad del proceso (activo o silencio) incluyendo el tiempo de vida (tiempo en curso en dicho estado), es decir, $v(n)$ puede ser $a(i)$, i: 1,…, A, si el proceso está en el i-ésimo *slot* (ranura) de un estado activo, o $s(j)$, j: 1, …, S, si el proceso está en el j-ésimo *slot* de un estado de silencio. La Fig. 3 muestra el correspondiente diagrama de transición de estados. Las probabilidades de transición de la cadena fueron también calculadas en [7], quedando expresadas en función de las distribuciones de los períodos activo y de silencio:

$$q_a(i) = p\left[a(i) \to a(i+1)\right] = \frac{1-F_a(i)}{1-F_a(i-1)} \qquad (2a)$$

$$\tilde{q}_a(i) = p\left[a(i) \to s(1)\right] = 1-q_a(i) \qquad (2b)$$

$$q_s(j) = p\left[s(j) \to s(j+1)\right] = \frac{1-F_s(j)}{1-F_s(j-1)} \qquad (2c)$$

$$\tilde{q}_s(j) = p\left[s(j) \to a(1)\right] = 1-q_s(j) \qquad (2d)$$

donde $i \in [1, A]$ y $j \in [1, S]$.

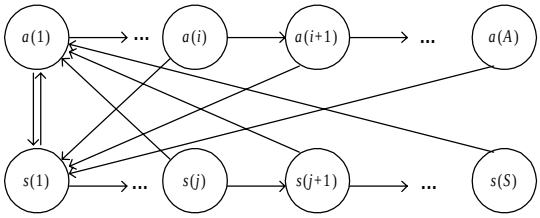


Figura 3: Transiciones de la cadena de Markov que describe una fuente *on/off* individual

Para el vector de estado en régimen estacionario $\mathbf{p} = \left[p_a(1),\ldots,p_a(A),p_s(1),\ldots,p_s(S)\right]$ se obtienen fácilmente las siguientes expresiones cerradas y exactas:

$$p_a(i) = \frac{1-F_a(i-1)}{E(a)+E(s)},\ i = 1,\ldots,A \qquad (3a)$$

$$p_s(j) = \frac{1-F_s(j-1)}{E(a)+E(s)},\ j = 1,\ldots,S \qquad (3b)$$

Si $q(n)$ representa el proceso de cuentas asociado al modelo, esto es, el número de celdas (entre 0 y B) generadas por el modelo a lo largo del tiempo (n), su función de autocorrelación se define de la siguiente manera:

$$R_q(k) = E\left[q(n)\cdot q(n+k)\right] = \\ = R_b(k)\sum_{i,i'=1}^{A} p_a(i)\cdot q_a^{(k)}(i,i') \qquad (4)$$

En esta última expresión, $R_b(k)$ representa la función de autocorrelación de la secuencia de tamaños de *batch*, es decir, de los sucesivos grupos de celdas que el modelo va generando. Dadas las condiciones de independencia, esta autocorrelación se reduce al siguiente resultado:

$$R_b(k) = \begin{cases} E^2(b),\ k \neq 0 \\ E(b^2),\ k = 0 \end{cases} \qquad (5)$$

Así pues, el cálculo de la función de autocorrelación dada por (4) requiere la evaluación del término $q_a^{(k)}(i,i')$, que no es más que la probabilidad de transición del estado $a(i)$ al estado $a(i')$ después de k pasos. Nótese que solamente es necesario evaluar las transiciones entre estados de actividad, ya que son estos los únicos que generan celdas. El análisis de este término es complejo y laborioso y se puede seguir en [24]. Dicho análisis incluye la aplicación de algunos resultados de la

Teoría de Renovación y la propiedad de convolución para la suma de variables aleatorias independientes. El resultado final es exacto y queda expresado de manera más sencilla en el dominio de la transformada-Z, tal como se expone a continuación:

$$R_q(z) = \frac{E^2(b) \cdot E(a)}{E(a) + E(s)} \left[H_{\hat{a}}(z) + E(a) \frac{z G_{\hat{a}}^2(z) G_s(z)}{1 - G_a(z) G_s(z)} \right] \tag{6a}$$

donde

$$H_{\hat{a}}(z) = \frac{1 - z \cdot G_{\hat{a}}(z)}{1 - z} \tag{6b}$$

$$G_{\hat{a}}(z) = \frac{1}{E(a)} \cdot \frac{1 - G_a(z)}{1 - z} \tag{6c}$$

En estas expresiones, $G_a(z)$ y $G_s(z)$ denotan respectivamente las funciones generatrices de los períodos activo y de silencio. La inversión de la expresión dada por (6a) permite obtener la función de autocorrelación en el dominio del tiempo, cuyo resultado debe ser convenientemente modificado para *k*=0 por el factor $E(b^2) / E^2(b)$.

Nótese que el resultado (6a) queda expresado en función de las distribuciones de los períodos activo y de silencio y del tamaño de *batch* promedio, es decir, el número medio de celdas generadas durante los estados activos. Así pues, conocidos estos parámetros de entrada, la expresión (6a) proporciona una procedimiento analítico exacto para obtener la función de autocorrelación del número de cuentas y, por tanto, el IDC. Especialmente interesante resulta dicha expresión en el contexto del tráfico fractal, ya que permite poner de manifiesto con rigor matemático la estrecha relación entre las distribuciones de caída lenta y el grado de dependencia a largo plazo del proceso (función de autocorrelación con perfil de decrecimiento hiperbólico), dado por el parámetro de Hurst (*H*).

3.2 Autocorrelación de los intervalos entre llegadas

Otro tipo de función de autocorrelación considerada en el modelado de tráfico, aunque menos utilizada que la anterior, es la de los intervalos entre llegadas de lotes de celdas (*batches*):

$$R_i(k) = E[i(n) \cdot i(n+k)] \tag{7}$$

donde *i*(*n*) denota el enésimo intervalo entre llegadas de grupos de celdas consecutivos. Puesto que las llegadas de estos grupos en el modelo *on/off* agregado corresponden a llegadas individuales en el modelo *on/off* para una sola fuente, los resultados obtenidos en [13] pueden extenderse aquí sin necesidad de ninguna manipulación adicional. Al igual que en la autocorrelación del proceso de cuentas, se utilizó una formulación markoviana del modelo, pero en este caso para representar la secuencia de intervalos entre llegadas y no la de estados de actividad. Nuevamente, el resultado es exacto y puede expresarse en el dominio de la transformada-Z como se muestra a continuación [13, 24]:

$$R_i(z) = \left[1 + \frac{2E(s)}{E(a)} \right] \frac{1}{1 - z} + \frac{E^2(s)}{E(a)} r_i(z) + \frac{Var(s)}{E(a)} \tag{8a}$$

con

$$r_i(z) = \frac{1}{1 - G_a(z)} \tag{8b}$$

Como puede observarse en las expresiones (8a) y (8b), a diferencia de lo que ocurría con la función de autocorrelación del número de cuentas, que dependía de las distribuciones completas de las duraciones de los períodos activo y de silencio, en la autocorrelación de los intervalos entre llegadas solamente influye la distribución completa del período activo, y la distribución del período de silencio únicamente a través de sus dos primeros momentos, la media y la varianza.

En [24] se ofrece un análisis de esta función de autocorrelación, más sencilla que la del proceso de cuentas. Esta sencillez permite distinguir fácilmente entre dos casos generales: cuando el máximo tamaño de ráfaga *A* es finito, la autocorrelación exhibe un perfil de decrecimiento exponencial, y por tanto el modelo no captura dependencias a largo plazo; por el contrario, si *A* es infinito, el modelo puede mostrar dependencias a largo plazo (perfil de autocorrelación hiperbólico), aunque este aspecto requiere un tratamiento más exhaustivo.

4 Evaluación de Comportamiento

El objetivo principal de la evaluación de comportamiento es el de establecer relaciones fiables entre las características del tráfico de paquetes (recogidas mediante descriptores) y los parámetros de comportamiento de los dispositivos de conmutación (tiempos de respuesta, tasa de pérdida de paquetes, etc.), de modo que permitan un correcto dimensionado de los recursos de red. En el apartado 3 se trataron diversas características del tráfico generado por los modelos *batch-on/off*, pudiendo comprobar cómo estos permitían el tratamiento analítico a pesar de la generalidad con que están definidos. En este apartado se comprueba también esa capacidad en el terreno de la

evaluación de comportamiento. Concretamente, consideramos un modelo de colas del tipo (*Batch-on/off*)/D/1, en el que el servidor representa una línea por la que se transmiten las celdas ATM, que como sabemos tienen longitud fija. Para el análisis de esta cola, se ha seguido el método de descomposición que sugiere la Fig. 4. En esta figura se representa dos trazas de tráfico agregado muy similares. En ambas se señala el tiempo de respuesta que experimenta cada celda al cruzar una cola de capacidad infinita seguida de un servidor determinista. En la traza superior, todas las celdas del primer período activo han sido ya servidas cuando llega el segundo, y por tanto éste se encuentra la cola vacía. En la segunda traza el período de silencio es más corto, y existe un remanente de celdas (nivel de retención o *backlog r*) a la llegada del segundo período activo. Como puede observarse, los tiempos de respuesta de las celdas de ese segundo período activo se incrementan en el valor del remanente. En definitiva, el tiempo de respuesta experimentado por las celdas puede descomponerse en dos contribuciones: la contribución del propio período activo, y la del remanente provocado por el período activo predecesor.

En el caso de los modelos *batch-on/off* estas dos contribuciones son independientes entre sí, y por tanto la distribución del tiempo de respuesta puede expresarse como convolución de las distribuciones asociadas a cada contribución, y en el dominio de la transformada-Z la convolución se puede substituir por un simple producto. Además, las características de dichos modelos facilitan también el análisis de cada contribución por separado. A continuación se exponen los detalles principales del análisis de cada una de ellas. El análisis completo es mucho más extenso y puede seguirse en [25, 26].

4.1 Contribución de ráfagas individuales

Para evaluar la contribución de las ráfagas o períodos activos individuales, se puede construir en primer lugar una caracterización de la cola (*Batch-on/off*)/D/1 mediante una cadena de Markov bidimensional. Dicha cadena se obtiene ampliando la cadena utilizada en el cálculo de la autocorrelación de los intervalos entre llegadas, al incluir en la descripción de estados una nueva variable: el tiempo de respuesta superior. Éste designa el tiempo de respuesta de la última celda de un lote en ser servida, y por tanto representa una cota superior del tiempo de respuesta que se quiere obtener.

La contribución de un solo período activo se obtiene al particularizar y resolver la cadena anterior en el supuesto de un modelo *batch-on/off* ficticio con las mismas ráfagas que el modelo original, pero con períodos de silencio lo suficientemente largos como para que el nivel de retención de celdas sea nulo. Finalmente, el tiempo de respuesta provocado por una sola ráfaga *w'* se obtiene a partir del tiempo de respuesta superior calculando previamente la correlación de este último con el tamaño del lote a que corresponde. El resultado que se obtiene es exacto y queda expresado de la siguiente forma en términos de la función generatriz:

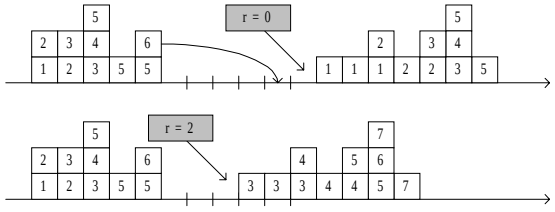


Figura 4: Efecto del nivel de retención o remanente de celdas en el tiempo de respuesta

$$G_{w'}(z) = \frac{z\cdot[1-G_b(z)]}{(1-z)\cdot E(a)}\left\{1+p_a(1)[E(a)-1]+\right.$$
$$\left.+\sum_{i=2}^{A-1}[1-F_a(i)]\cdot\left[\frac{G_b(z)}{z}\right]^{i-1}+\sum_{i=1}^{A-1}p_a(i+1)\cdot\left[\frac{G_b(z)}{z}\right]^{i}\right\} \quad (9)$$

donde $G_b(z)$ representa la función generatriz asociada a la distribución del tamaño de lotes.

4.2 Contribución del nivel de retención

El nivel de retención al inicio de un período activo resulta de sumar el nivel de retención al inicio del período activo predecesor y la variación del nivel de retención introducida por este último. Ésta puede ser positiva o negativa, mientras que el nivel de retención resultante al inicio de cualquier período activo es siempre no negativo. Esta evolución es similar al recorrido aleatorio discreto seguido por una partícula que se mueve a lo largo de la dirección positiva del eje de abcisas, con una barrera reflectante situada en el origen de coordenadas. La posición real de la partícula se correspondería con el nivel de retención, mientras que las variaciones de éste corresponderían a saltos realizados por la partícula a la derecha (positivos) o a la izquierda (negativos). Este recorrido aleatorio admite una caracterización markoviana, ya que los saltos son independientes entre sí (recuérdese que en un modelo *batch-on/off* las ráfagas son independientes entre sí). La matriz de transición de estados **T** es de orden infinito y viene dada por la siguiente expresión:

$$\begin{pmatrix} F_\Delta(0) & p_\Delta(1) & p_\Delta(2) & p_\Delta(3) & p_\Delta(4) & \dots \\ F_\Delta(-1) & p_\Delta(0) & p_\Delta(1) & p_\Delta(2) & p_\Delta(3) & \dots \\ F_\Delta(-2) & p_\Delta(-1) & p_\Delta(0) & p_\Delta(1) & p_\Delta(2) & \dots \\ F_\Delta(-3) & p_\Delta(-2) & p_\Delta(-1) & p_\Delta(0) & p_\Delta(1) & \dots \\ F_\Delta(-4) & p_\Delta(-3) & p_\Delta(-2) & p_\Delta(-1) & p_\Delta(0) & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{pmatrix} \quad (10a)$$

donde p_Δ es la distribución del valor de los saltos (variación del nivel de retención) y F_Δ la correspondiente función acumulada. La distribución de los saltos responde a la siguiente expresión general exacta:

$$p_\Delta(l) = \frac{1}{E(a)} \sum_{j=1}^{S} p_s(j) \sum_{i=0}^{A-1} p_a(i+1)\, p_B(i+1, l+j+1) \tag{10b}$$

donde $l \in [-S, \infty)$.

La distribución del nivel de retención no es más que la solución en régimen estacionario de la cadena de Markov anterior $\mathbf{p} = [p_r(0), p_r(1), \ldots]$ para lo cual hay que resolver las siguientes ecuaciones:

$$\mathbf{p} = \mathbf{p} \cdot \mathbf{T} \tag{11a}$$

$$\mathbf{p} \cdot \mathbf{1} = \mathbf{1} \tag{11b}$$

donde **1** es un vector columna con todos sus elementos iguales a la unidad.

En general, no existe solución analítica para el sistema de ecuaciones (11a)-(11b). Una alternativa consistiría en aplicar algún método numérico para la resolución del régimen estacionario de una cadena de Markov [27, 28]. Una alternativa más interesante a fin de obtener una aproximación analítica consistiría en extender el método de difusión aplicado en la resolución de la cola GI/G/1, que produce muy buenos resultados en condiciones de tráfico elevado [29].

5 Conclusiones

Este artículo recoge el trabajo desarrollado por los autores durante los últimos años, incluyendo las aportaciones más recientes, sobre una clase muy general de modelos de tráfico para redes ATM: los modelos *batch-on/off* (la versión para tráfico de fuente puede considerarse un caso particular de estos). El objetivo ha sido doble:

- Por un lado, poner de manifiesto la vigencia actual de estos modelos que, gracias a la generalidad de su definición, han ido adaptándose a las nuevas estimaciones que sobre la naturaleza del tráfico de banda ancha la investigación ha ido realizando.

- Por otro, poner también de manifiesto la viabilidad de estos modelos como herramienta analítica, tanto en el aspecto de los descriptores como en el de la evaluación de comportamiento.

En cuanto a las líneas de investigación futuras, pueden destacarse las dos siguientes:

- Utilizar los resultados obtenidos como punto de partida para desarrollar un entorno o plataforma software para el análisis de tráfico en el plano teórico y en el de la simulación. Dicho entorno estaría configurado por los tres niveles mostrados en la Fig. 5. El objetivo final sería determinar los parámetros básicos que condicionan la forma de las curvas de respuesta y su relación con los descriptores ATM.

- Otro proyecto sería extender la aplicación de los modelos *batch-on/off* para describir el tráfico Internet.

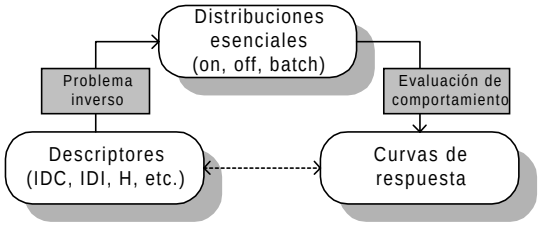


Figura 5: Configuración básica de una plataforma *software* para el análisis de tráfico

Referencias

[1] Heffes, H., Lucantoni, D. M., "A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance", IEEE JSAC, SAC-4, No. 6, 1986.

[2] Sriram, K., Whitt, W., "Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data", IEEE Journal on Selected Areas in Communications, Vol. SAC-4, No. 6, September 1986.

[3] Viterbi, A., "Approximate Analysis of Time-Synchronous Packet Networks", IEEE Journal on Selected Areas in Communications, 4, 879-890, 1986.

[4] Bruneel, H., "Queueing Behavior of Statistical Multiplexers with Correlated Inputs", IEEE Trans. on Communications, 36, 1339-1341, 1988.

[5] Li, S., Sheng, H., "Discrete Queueing Analysis of Multimedia Traffic with Diversity of Correlation and Burstiness Properties", Proc. of the IEEE INFOCOM'91.

[6] Sohraby, K., "On the Asymptotic Behavior of Heterogeneous Statistical Multiplexer with Applications", Proc. of the IEEE INFOCOM'92.

[7] Sohraby, K., "On the Theory of General ON-OFF Sources with Applications in High-Speed Netwroks", Proc. of the IEEE INFOCOM'93.

[8] Elsayed, K., “On the Superposition of Discrete-Time Markov Renewal Processes and Application to Statistical Multiplexing of Bursty Traffic Sources”, Proc. of the IEEE GLOBECOM’94.

[9] Simonian, A., Guibert, J., “Large Deviations Approximation for Fluid Queues Fed by a Large Number of On/Off Sources”, Proc. of ITC 14, 1013-1022, 1994.

[10] Wittevrongel, S., Bruneel, H., “Deriving the Tail Distribution of the Buffer Contents in a Statistical Multiplexer with General Heterogeneous On/Off Sources”, Proc. of the IFIP TC6/WG6.3 & WG7.3 International Conference on the Performance and Management of Complex Communication Networks, Tsukuba, Japan, 1997.

[11] Elsayed, K., Perros, H., “On the Effective Bandwidth of Arbitrary On/Off Sources”, Proc. of the Sixth IFIP WG6.3 Conference on Performance of Computer Networks, Istanbul, Turkey, 1995.

[12] Laevens, K., “(Heavy-Tailed) On-Off Sources (I): Traffic Characteristics”, COST257 2nd MCM Report, 1997.

[13] Galmés, S., Puigjaner, R., “A Source Independent Traffic Model for ATM Networks”, Proc. of the IFIP TC6/WG6.3 & WG7.3 International Conference on the Performance and Management of Complex Communication Networks, Tsukuba, Japan, 1997.

[14] Galmés, S., Puigjaner, R., “On the Capabilities of On-Off Models to Capture Arbitrary ATM Sources”, Proc. of the IEEE INFOCOM’98.

[15] Leland, W. E., Taqqu, M. S., Willinger, W., Wilson, D. V., “On the Self-Similar Nature of Ethernet Traffic (extended version)”, IEEE/ACM Trans. on Networking, vol. 2, no. 1, February 1994.

[16] Garret, M. W., Willinger, W., “Analysis, Modelling and Generation of Self-Similar VBR Video Traffic”, Proc. of the ACM Sigcomm’94.

[17] Willinger, W., Taqqu, M., Sherman, R., Wilson, D. V., “Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level, Proc. of the ACM Sigcomm’95.

[18] Roberts, J., Mocci, U., Virtamo, J., *Broadband Network Teletraffic (Final Report of Action COST 242)*, Springer 1996.

[19] Crovella, M. E., “Performance characteristics of the Worls Wide Web”, Lecture Notes in Computer Science, 1769 (Günter Haring, Christoph Lindeman, Martin Reiser eds.), pp. 219-232.

[20] Blondia, C., “A Discrete-Time Batch Markovian Arrival Process as B-ISDN Traffic Model”, Belgian Journal of Operations Research, Statistics and Computer Science, 32 (3,4):3-23, 1993.

[21] Frost, V. S., Melamed, B., “Traffic Modeling for Telecommunications Networks”, IEEE Communications Magazine, March 1994.

[22] Kuehn, P., “Reminder on Queueing Theory for ATM Networks”, First ATM Traffic Expert Symposium, Basel, Switzerland, 1995.

[23] Gihr, O., Tran-Gia, P., “A Layered Description of ATM Cell Traffic Streams and Correlation Analysis”, Proc. of the INFOCOM’92, pp. 2D.4.1-2D.4.8.

[24] Galmés, S., Puigjaner, R., “Correlation Structure of the Batch-On/Off Model”, aceptado para su presentación en el 2001 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS’01). Pendiente de publicación.

[25] Galmés, S., Puigjaner, R., “Performance Evaluation Based on an Aggregate ATM Model”, aceptado para su presentación en el Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems MASCOTS’01. Pendiente de publicación.

[26] Galmés, S., “Analysis of the (Batch-On/Off)/D/1 Queue”, aceptado para su presentación en el Seventeenth International Teletraffic Congress. Pendiente de publicación.

[27] Stewart, W., Introduction to Numerical Solution of Markov Chains, Princeton University Press, 1994.

[28] Bolch, G., Greiner, S., de Meer, Hermann, Trivedi, K., Queueing Netwroks and Markov Chains (Modeling and Performance Evaluation with Computer Science Applications), John Wiley & Sons, 1998.

[29] Gelenbe, E., Mitrani, I., Analysis and Synthesis of Computer Systems, Ac. Press, 1980.

Modelado de Internet para simulación con modelos de tráfico que incorporan el protocolo TCP

E. González Parada, J. M. Cano García, J. A. Aguilar González, A. Reyes Lecuona y A. Díaz Estrella.
Departamento de Tecnología Electrónica. Universidad de Málaga.

Abstract

In this paper, several models for Internet packet losses and delay are evaluated using a simplified model of an access network. It is determined how the performance of TCP flow control mechanisms is affected by the model election, and its validity is discussed.

1 Introducción

En los últimos años, la popularidad alcanzada por las aplicaciones de Internet ha influido notablemente en el desarrollo y la adaptación de las nuevas redes de telecomunicaciones, lo que es ya una realidad incluso para sistemas de tan nueva implantación como las redes móviles [1, 2].

En muchos casos, el desarrollo de las redes de telecomunicación exige un estudio previo mediante simulación de las prestaciones alcanzadas. Para realizar estas tareas de evaluación de prestaciones mediante simulación, son necesarios modelos de tráfico que se adapten con precisión a las características del tráfico generado por las aplicaciones.

Muchos de lo modelos propuestos para aplicaciones TCP/IP operan sobre lo que se denomina nivel de enlace. Sin embargo, en [3] se concluye que tales modelos no son aplicables para la evaluación de las prestaciones de sistemas que incorporen el protocolo TCP, puesto que éste ofrece un servicio best-effort e incluye un control de flujo basado en las prestaciones que ofrece la red en cada instante, y por lo tanto las características del tráfico son dependientes de las condiciones de la red. Algunos autores señalan incluso que el comportamiento de TCP bajo determinadas condiciones puede ser una de las causas de la autosemejanza del tráfico [4, 5].

La alternativa que parece más viable es centrar las tareas de modelado en la caracterización del comportamiento del usuario en lugar de modelar el proceso de llegada de paquetes. Esto lleva a la necesidad de incorporar el comportamiento de los protocolos encargados de llevar a cabo las comunicaciones requeridas por las aplicaciones, que en el caso de las aplicaciones de Internet es el protocolo de transporte TCP. Por otra parte también es necesario considerar el estado de la red, que viene definido por los parámetros de calidad, retardo y probabilidad de perdida que sufren los bytes de una determinada conexión.

En la literatura existen numerosos trabajos que resaltan la dificultad de modelar las principales características de la red Internet[6, 7, 8, 9]. Por este motivo en muchos estudios sobre redes de acceso se utilizan modelos sencillos [1].

En este artículo se evalúa un conjunto de modelos que caracterizan el estado de la red Internet. Para ello se determinará el grado de afectación de las prestaciones que proporcionan los mecanismos de control de flujo del protocolo transporte TCP, estableciendo el entorno de validez de tales modelos.

El resto de este artículo se divide en las siguientes secciones. La descripción de los distintos modelos de la red Internet propuestos aparecerá en la sección 2. Las condiciones bajo las que se realiza la simulación y los resultados obtenidos aparecen respectivamente en los apartados 3 y 4. Por último, se establecen las condiciones en las que cada uno de los modelos propuestos es más adecuado para simular redes de acceso.

2 Modelo de Internet

En este apartado se describen los modelos considerados para simular el comportamiento de la red Internet. Como se mencionó anteriormente, los parámetros que se consideran son el retardo y la pérdida de paquetes. En los modelos propuestos, dichos parámetros van a ser caracterizados de forma independiente y no se considera la posible correlación que puede existir entre ambos.

2.1 Modelado del retardo

Para modelar este aspecto dentro de la nube Internet se van a considerar tres modelos en orden creciente de complejidad: Retardo constante, retardo gaussiano incorrelado y retardo gaussiano correlado.

2.1.1 Retardo constante

Se trata del modelo más sencillo entre los posibles y establece un retardo constante para todos los paquetes de la misma conexión que atraviesan la nube internet.

2.1.2 Retardo Gaussiano incorrelado

Este modelo supone que el retardo de los paquetes pertenecientes a una conexión dada se ajusta a una distribución gaussiana con una determinada media y varianza. Es un modelo que se encuentra propuesto en la literatura [1] como una forma sencilla de simular el retardo sufrido por los paquetes en el tramo de Internet, y no considera la posible relación existente entre los retardos sufridos por paquetes próximos.

2.1.3 Retardo Gaussiano correlado

Este modelo intenta caracterizar la correlación del retardo que sufren paquetes que pertenecen a una misma conexión y que están próximos entre sí, manteniendo la distribución gaussiana del retardo [10]. Para ello, el retardo que sufre un paquete se calcula en función del retardo del paquete anterior. Esta dependencia con respecto al retardo del paquete anterior disminuye de forma exponencial con el tiempo que separa ambos paquetes:

$$R(t) = a(T)(R(t-T) - \mu) + b(T)w(t) + \mu \quad (1)$$

Donde R es el retardo del paquete actual, R(t-T) es el retardo del paquete anterior, T es el tiempo que los separa, μ es la media del retardo, y w(t) es una variable aleatoria gaussiana con media 0 y varianza 1. Por otra parte:

$$a(T) = e^{\frac{-T}{\tau}} \quad (2)$$

$$b(T) = \sigma\sqrt{1 - a(T)} \quad (3)$$

De esta forma, se puede ajustar la media y varianza del retardo, que mantiene una distribución gaussiana con media μ y varianza σ, y la caída exponencial de la función de autocorrelación, mediante el parámetro τ. Este modelo por tanto ajusta la correlación a corto plazo del retardo de los paquetes de una misma conexión. Sin embargo, existen determinados estudios que señalan la existencia de dependencia a largo plazo (LRD) en el retardo de transmisión [11], como un posible reflejo de la naturaleza autosimilar del tráfico de Internet [12, 13]. No obstante, debemos considerar que el modelo se propone para la realización de simulaciones utilizando conexiones TCP, por lo que no sería tan necesario considerar escalas temporales mayores que el tiempo que duran dichas conexiones.

2.2 Modelado de la pérdida de paquetes

En cuanto a las pérdidas, se pretende comparar dos modelos, un modelo sencillo de Bernouilli, y un modelo más realista que considera la existencia de ráfagas de pérdidas.

2.2.1 Modelo de Bernouilli

En este modelo se fija una probabilidad de pérdida p, que se aplica de forma independiente a cada paquete de una misma conexión. De esta forma, cada vez que un paquete atraviesa la red se realiza un experimento aleatorio con una probabilidad de éxito p que determina si el paquete es perdido o se transmite exitosamente por el tramo Internet.

2.2.2 Modelo de pérdidas a ráfagas

Las pérdidas en Internet no se deben por lo general a errores, sino a la congestión de determinados routers en la trayectoria que siguen los paquetes. Por este motivo, las pérdidas suelen producirse a ráfagas, lo que ha sido comprobado en numerosos estudios [14, 15]. Puede ser por tanto de interés el considerar este hecho en el modelo de la red Internet.

En [15, 16] se analizan las pérdidas de paquetes mediante el envío de paquetes equiespaciados, y se propone, entre otras posibilidades, la utilización de una cadena de Markov de dos estados para modelar el proceso de pérdidas. Siguiendo esta aproximación, aquí se utilizará un modelo de dos estados, considerando una duración exponencial para cada uno de ellos. Todos los paquetes que se transmitan durante el estado de congestión van a perderse, por lo que la longitud de este estado se corresponde con la de la ráfaga de pérdidas y por tanto no tendrá una duración media muy elevada. De acuerdo con las observaciones realizadas por [15], la duración media de la ráfaga de pérdidas se ha considerado de unos 50 ms.

La existencia de pérdidas a ráfagas en Internet se debe, entre otras causas, a que la política de gestión de colas más habitual determina que se descarten aquellos paquetes que no caben en los buffers (*drop tail*). No obstante, las pérdidas tenderán a producirse de forma independiente a medida que se implanten otras políticas como RED (*Random Early Detection* [17]), que implican el descarte aleatorio de paquetes para prevenir la congestión.

3 Simulaciones

En esta sección se establecen las condiciones de simulación y los parámetros empleados en la evaluación de las prestaciones del protocolo TCP al incorporar los modelos de retardo y perdidas de paquetes que caracterizan el comportamiento de la red Internet.

Las pruebas se han realizado utilizando la herramienta OPNET Modeler, considerando un modelo genérico de red de acceso que se describe en la sección 3.1. Para ello se ha desarrollado una librería que emula los principales mecanismos del protocolo TCP. En el apartado 3.2 se detallan en primer lugar los mecanismos de TCP que se han considerado en las simulaciones. Posteriormente

se describirán los escenarios que se han simulado y los parámetros de salida que se van a considerar para realizar la comparación entre los diferentes modelos.

Modelo de tráfico	FTP, Web		
Número de usuarios	40		
Velocidad de los servidores	1 Mbps/usuario, 20 kbps/usuario		
Modelo de Retardo			
Modelo gaussiano correlado	$\mu = 0 - 0.4s$	$\sigma = 0 - 0.25\mu$	$\tau = 0.5$
Modelo de pérdidas			
Bernouilli	4%		
Ráfagas	Distribuciones exponenciales	Duración ráfagas 50 ms	Tiempo entre ráfegas 1.2 s

Tabla 1: Parámetros de las simulaciones

3.1 Descripción de la red de pruebas

Para determinar las características con las que debe contar el modelo de Internet se ha utilizado una red genérica de pruebas. El cometido de esta red es evaluar, en un entorno controlado, el funcionamiento de los mecanismos del protocolo TCP/IP ante los modelos propuestos de Internet.

La red de pruebas es un modelo sencillo de red de acceso, cuyo diagrama de bloques se muestra en la figura 1. Para modelar la red de acceso se utilizan dos colas con servidores de tasa constante, correspondientes a los enlaces ascendente y descendente. Los recursos de estos enlaces son compartidos por varios usuarios para acceder a distintos hosts a través de la red Internet. Las simulaciones se han realizado para dos velocidades de servidor diferentes: 1 Mbps por usuario y 20 kbps por usuario, correspondiendo a una red de acceso rápida y a una red de acceso más lenta. El tamaño de los buffers de la red de acceso no ha sido limitado, por lo que en este tramo no se producen pérdidas.

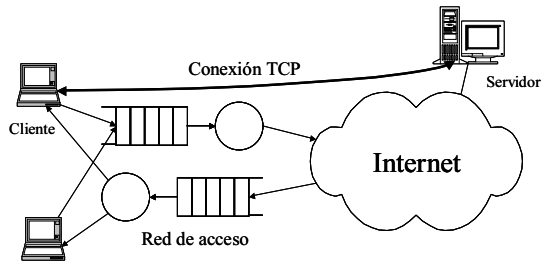


Figura 1: Esquema de la red de prueba

3.2 Modelo de TCP

Puesto que no existe una implementación de TCP que pueda ser considerada estándar, no se ha tratado de realizar un modelo de una implementación concreta del protocolo, sino que se ha intentado reproducir las principales características que debería tener una implementación actual del protocolo, prestando especial atención a los mecanismos relacionados con el control de flujo y la recuperación de paquetes perdidos [18, 19, 20, 21]. La mayoría de las implementaciones de TCP que se incluyen en los sistemas actuales suelen ser variantes del TCP-Reno [22, 23]. Entre los algoritmos que se han considerado, se encuentran los siguientes:

- Slow start de Jacobson [24]
- Congestion Avoidance
- Algoritmo de Karn [25]
- Exponential Backoff
- Delayed ACK
- Fast Retransmit
- Fast Recovery [21]
- Timestamps

Por otra parte, hay que considerar una serie de parámetros, que corresponden a los diferentes mecanismos de TCP, y que se representan en la tabla 2.

Parámetro	Valor
Tamaño del segmento	536 Bytes
Tamaño máximo de la ventana	100 MSS
Tamaño inicial de la ventana	1 MSS
ssthreshold inicial	65535 Bytes
Máximo retardo del ACK	0.2 s
Umbral para FASTRTX	3 DUPS
Timeout Máximo	64 s
Timeout mínimo	200 ms
Timeout inicial	3 s
Granularidad del timer	200 ms
Karn g	0,125
Karn h	0,25
Karn k	4

Tabla 2: Parámetros de TCP

Dentro de las simplificaciones realizadas cabe destacar que en las simulaciones se ha considerado que la ventana del receptor es suficientemente grande como para no interactuar con la ventana de congestión en ningún momento, y no se han modelado los algoritmos de prevención del "síndrome de ventana tonta".

3.3 Modelos de tráfico

En las simulaciones realizadas se han considerado dos modelos de tráfico diferentes. En un primer lugar se ha utilizado un modelo sencillo de FTP, en el que varios usuarios descargan simultáneamente un fichero de tamaño constante (2 MBytes). Además se ha utilizado un modelo de tráfico web estructuralista [26], en el que se caracterizan de forma estadística una serie de parámetros correspondientes a las características propias de la navegación WWW. Estos parámetros se resumen en la tabla 3

Parámetro	Distribución	Parámetros	
Tiempo entre sesiones			
Número de páginas por sesión	Determinista	infinitas	
Tiempo entre páginas	Gamma	$\mu = 46,8s$	$\sigma = 168,6s$
Número de conexiones por página	Lognormal	$\mu = 5,3s$	$\sigma = 12s$
Tiempo entre conexiones	Gamma	$\mu = 2,3s$	$\sigma = 4,5s$
Tamaño de la conexión	Pareto	$\mu = 5616$ Bytes	$\alpha = 1,77$
Tamaño de la URL	Lognormal	$\mu = 364$ Bytes	$\sigma = 101$ Bytes

Tabla 3: Parámetros del modelo de tráfico WWW

3.4 Parámetros de salida

Los parámetros monitorizados considerados para comparar la reacción del TCP ante las distintas condiciones de funcionamiento impuestas por los modelos de Internet son:

- Eficiencia media: definida como la razón entre el número de bytes útiles transmitidos (B_u) y el número total de bytes enviados por el protocolo TCP (B_t).

$$E = \frac{\sum_i B_u^i}{\sum_i B_t^i} \tag{4}$$

- Goodput medio: definido como la razón entre el número de bytes útiles trasmitidos (B_u) y la duración agregada de las conexiones TCP (D).

$$G = \frac{\sum_i B_u^i}{\sum_i D^i} \tag{5}$$

La evaluación de estos parámetros determinará las condiciones en las que cada uno de los modelos propuestos es adecuado.

3.5 Parámetros del modelo de internet

Los modelos de pérdidas y retardo en Internet se aplican de forma independiente a cada usuario, suponiendo que las conexiones establecidas por cada uno de ellos tienen como destino un host diferente, y por tanto, siguen rutas distintas con diferentes estados de congestión.

3.5.1 Modelo de retardo

En primer lugar se han realizado una serie de simulaciones sin incluir el modelo de pérdidas, con el objetivo de comprobar la necesidad de incorporar diferentes refinamientos al modelo de retardo.

En las simulaciones se pretende comprobar el efecto que la desviación del retardo en internet tiene sobre las conexiones TCP. También se pretende establecer si el hecho de que conexiones que comparten un mismo enlace sufran un jitter diferente influye en la forma en la que se reparte el ancho de banda entre ellas. Para ello, en estas simulaciones se consideran dos tipos de usuario. Los paquetes de los usuarios de tipo II sufren un retardo constante al transmitirse a través de la red Internet, mientras que los paquetes de los usuarios de tipo I estarán sometidos a un retardo variable, que vendrá caracterizados por los modelos propuestos. En las simulaciones realizadas, se ha considerado que el retardo medio de los usuarios de tipo II es el mismo que el de los usuarios de tipo I, pero se evalúan diferentes valores de la desviación típica del retardo en estos últimos.

Así pues, los parámetros que van a considerarse en el caso más general de los modelos propuestos son el retardo medio, la desviación del retardo y el parámetro de correlación τ. El valor del parámetro de correlación $\tau = 0.5$ ha sido elegido de acuerdo a observaciones del retardo sufrido por los paquetes. Para ello se estudió la correlación del RTT sufrido por paquetes ICMP de eco enviados cada 20 ms desde la Universidad de Málaga hacia distintos servidores. En estas medidas se constata que la correlación entre paquetes separados más de 2 segundos es prácticamente nula, aunque efectivamente sí existe correlación en el retardo sufrido por paquetes próximos.

Por otra parte, en las primeras simulaciones realizadas con los modelos (Gaussianos correlados e incorrelados) bajo las condiciones de simulación anteriormente expuestas, se puso de manifiesto que los modelos de retardo descritos provocan la llegada de paquetes desordenados al receptor. Este desorden es causado por la diferencia de retardo entre paquetes próximos y por tanto es más acusado en el caso del modelo incorrelado. La llegada de paquetes desordenados al receptor pone en marcha el mecanismo de fast-retransmit, activando los mecanismos de control de congestión aunque en realidad no se producen pérdidas de paquetes en el sistema. El desorden introducido por el modelo de retardo no es realista, puesto que normalmente los paquetes de una misma conexión siguen la misma ruta y no se desordenan, aunque existen estudios que indican que el desorden de paquetes en enlaces con determinadas características no es un hecho infrecuente [27]. Para paliar este efecto, se procedió a modificar los modelos impidiendo que en dos paquetes consecutivos separados T se diese la condición $R_2 < R_1 - T$, donde R_1 y R_2 son respectivamente los retardos asignados por el modelo a los paquetes 1 y 2.

3.5.2 Modelo de Pérdidas

Finalmente, se incluye el modelo de pérdidas en los escenarios de simulación con un doble objetivo. En primer lugar se pretende verificar si el modelo de pérdidas en ráfagas presenta diferencias significativas respecto al modelo de Bernoulli. Y en

segundo lugar la interacción de la presencia de las pérdidas en el modelo de retardo.

Tanto para el modelo de Bernoulli como para el modelo a ráfagas las perdidas consideradas se centran entorno al 4 %, lo que constituye un valor típico de pérdidas en la red Internet [14, 15]. En el caso del modelo a ráfagas se considera que la duración media de la ráfaga de error es de 50 ms, mientras que el tiempo medio entre ráfagas es de 1,2 s.

4 Resultados

En este apartado se muestran los resultados obtenidos al aplicar los modelos descritos en el apartado anterior a la red genérica que se definió en la sección 3.1. Las simulaciones se pueden dividir en dos bloques. En un primer bloque se verán las implicaciones que tienen los diferentes refinamientos del modelo del retardo en la prestaciones del protocolo TCP, comprobando su importancia y la necesidad de incluirlos en el modelo. En el segundo bloque de simulaciones se determinará hasta que punto es necesario considerar en el modelo las pérdidas en ráfagas.

4.1 Evaluación de los modelos de retardo

En las figuras 2-4 se representa el goodput frente a la variación de la desviación típica del retardo para el modelo gaussiano correlado modificado. En ellas se pone de manifiesto que el aumento de la desvación típica típica no tiene demasiados efectos sobre el goodput obtenido por los usuarios. En cambio, el valor medio del retardo de transmisión sí que afecta de forma determinante a las prestaciones obtenidas. En las simulaciones realizadas se ha comprobado que los usuarios de tipo I y II obtienen unas prestaciones similares, por lo que en principio, el hecho de que los usuarios de tipo I presenten un mayor jitter no implica que se vean perjudicados o beneficiados frente a los usuarios de tipo II.

4.2 Evaluación de los modelos de pérdidas

En las figuras 5, 6 y 7 se muestra el goodput obtenido al considerar un modelo de pérdidas de Bernouilli, mientras que en 8, 9 y 4.2, se muestran las prestaciones obtenidas al considerar un modelo de pérdidas en ráfagas. En ambos casos se considera el modelo de retardo gaussiano correlado.

De la observación de este conjunto de gráficas se deduce que existen diferencias importantes entre ambos modelos. Las pérdidas afectan en menor medida a las prestaciones del sistema cuando se producen a ráfagas, mientras que tienen un mayor impacto cuando son independientes. La existencia de pérdidas aisladas dispara con mayor frecuencia el mecanismo de control de congestión, lo que hace

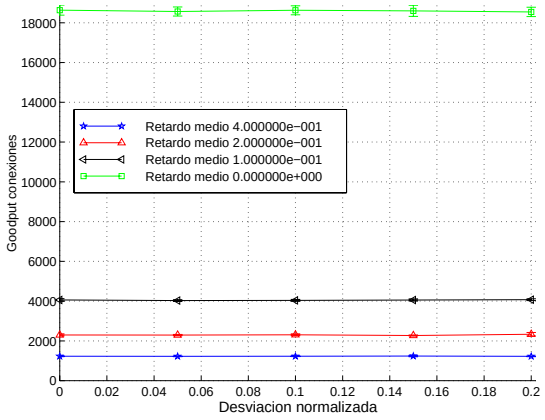


Figura 2: Goodput medio de las conexiones para usuarios de tipo I para tráfico WWW y 1Mbps por usuario

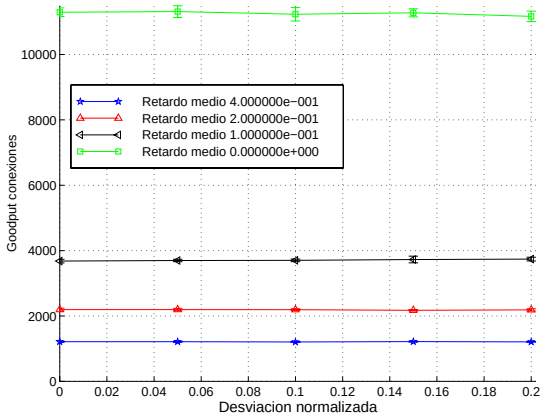


Figura 3: Goodput medio de las conexiones para usuarios de tipo I para tráfico WWW y 20 kbps por usuario

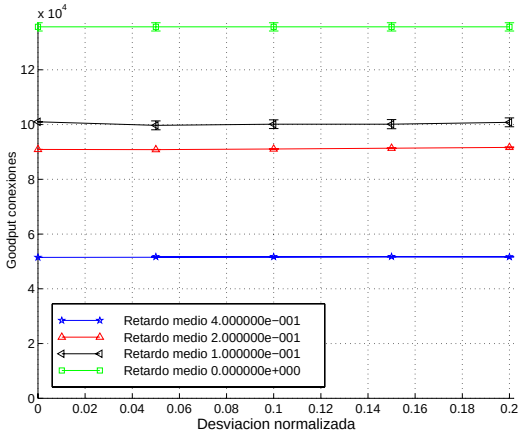


Figura 4: Goodput medio de las conexiones(Bps) para usuarios de tipo I para tráfico FTP y 1Mbps por usuario

disminuir el throughput medio de las conexiones a pesar de que estas pérdidas pueden ser recuperadas por fast-retransmit. Con la versión de TCP utilizada no es posible recuperar mediante fast-retransmit ráfagas de pérdidas, sin embargo, una vez que se produce un timeout, las pérdidas consecutivas son recuperadas sin volver a disparar el control de congestión, a diferencia de lo ocurre en el caso de las pérdidas aisladas.

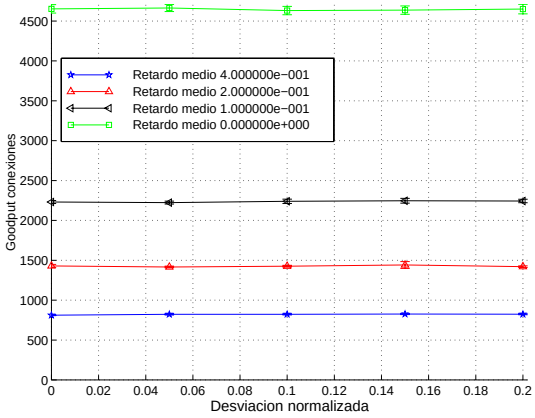


Figura 5: Goodput medio de las conexiones para usuarios de tipo I para tráfico WWW y 1Mbps por usuario. Modelo de errores de Bernouilli

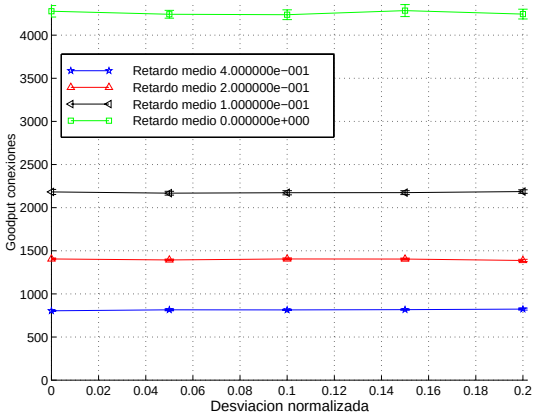


Figura 6: Goodput medio de las conexiones para usuarios de tipo I para tráfico WWW y 20 kbps por usuario. Modelo de errores de Bernouilli

5 Conclusiones

En este artículo se han comparado varios modelos de pérdidas y retardos candidatos para constituir un modelo de Internet. La evaluación de los distintos modelos se ha realizado en base al impacto que tienen los diferentes parámetros introducidos en el funcionamiento y prestaciones del protocolo TCP. Las pruebas se han realizado en una red de acceso genérica, que puede constituir la referencia de otro tipo de redes como las redes de acceso radio. De los resultados obtenidos se puede concluir que el

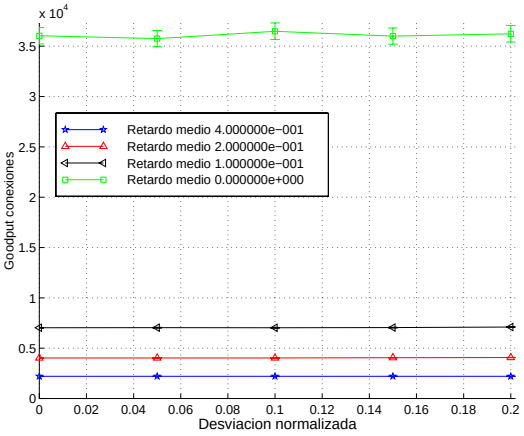


Figura 7: Goodput medio de las conexiones para usuarios de tipo I para tráfico FTP y 1 Mbps por usuario. Modelo de errores de Bernouilli

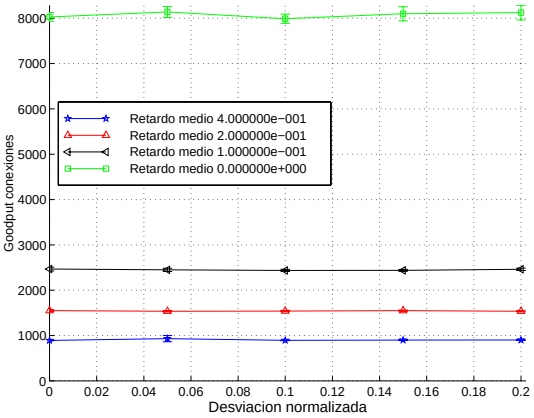


Figura 8: Goodput medio de las conexiones para usuarios de tipo I para tráfico WWW y 1 Mbps por usuario. Modelo de Errores a ráfagas

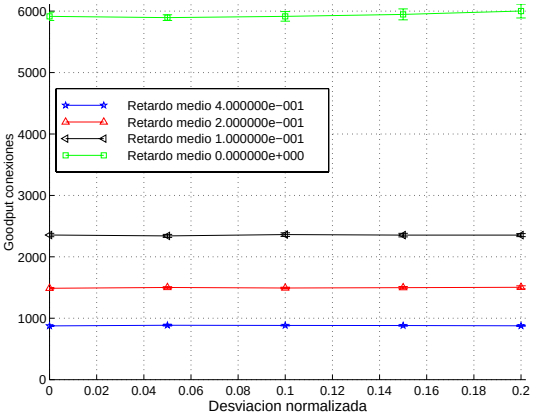


Figura 9: Goodput medio de las conexiones para usuarios de tipo I para tráfico WWW y 20 kbps por usuario. Modelo de Errores a ráfagas

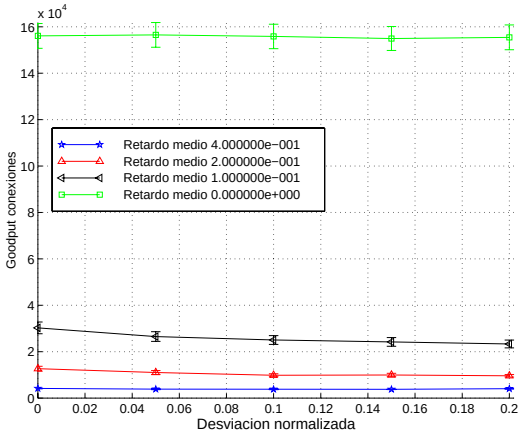


Figura 10: Goodput medio de las conexiones para usuarios de tipo I para tráfico FTP y 1 Mbps por usuario. Modelo de Errores a ráfagas

modelo de Internet debe incorporar las pérdidas a ráfagas y que con un modelo de retardo constante puede ser suficiente para modelar el retardo. Estas conclusiones son validas en sistemas que incorporen los mecanismos del protocolo TCP, para otro tipo de fuentes de tráfico habrá que estudiar si un modelo de retardo más complejo es necesario. En lo referente al modelo de perdidas se ha concluido que este parece ser un elemento critico en el modelado de la red, por ello serán necesarios futuros estudios que revelen las características con las que debe contar.

6 Agradecimientos

Este trabajo ha sido parcialmente subvencionado por la CICYT en el marco del proyecto de investigación ACIMUT (TIC2000-1120-C03-01) y Alcatel España.

Referencias

[1] R. Kalden, I. Meirick, and M. Meyer. Wireless internet access based on GPRS. *IEEE Personal Communications*, pages 8–18, April 2000.

[2] A. K. Salkintzis. Packet data over cellular networks: The CDPD aprroach. *IEEE Communications Magazine*, pages 152–159, June 1999.

[3] A. Arvidsson and P. Karlsson. On traffic models for TCP/IP. *ITC 16*, pages 457–466, July 1999.

[4] Jon M. Peha. Protocols can make traffic appear Self-Similar, 1997.

[5] A. Veres and M. Boda. The Chaotic Nature of TCP Congestion Control. In *INFOCOM (3)*, pages 1715–1723, 2000.

[6] V. Paxson and S. Floyd. Why we don´t know how to simulate the internet. *Proc. 1997 Winter Simulation Conference*, 1997.

[7] V. Paxson. End-to-end internet packet dynamics. *IEE/ACM Transactions on Networking*, 7(3):277–292, June 1999.

[8] S. Floyd and V. Paxson. Difficulties in simulating the internet. *to appear in IEEE/ACM Transactions on Networking*, 2001.

[9] M. Murray and KC Claffy. Measuring the inmeasurable: Global internet measurement infraestructure. 2001.

[10] A. Reyes. Internet delay characterization. Technical report, Departamento de Tecnología Electrónica. Universidad de Málaga, 2000.

[11] Q. Li and D. L. Mills. On the long-range dependence of packet round-trip delays in internet. *Proc. of IEEE ICC'98*, 2:1185–1191, 1998.

[12] V. Paxson and S. Floyd. Wide area traffic, the failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244,, 1995.

[13] W. Leland, M. Taqqu, W. Willinger, and D.V. Wilson. On the self-similar nature of ethernet traffic. *IEEE/ACM Transactions on Networking*, 2(1):1–15, 1994.

[14] Y. Zhang, V. Paxson, and S. Shenker. The stationarity of internet path properties: Routing, loss, and throughput. Technical report, ACIRI, 2000.

[15] M. Yajnik, S. Moon, J. Kurose, and D. Towsley. Measurements and modelling of the temporal dependence in packet loss. *Infocom99.*, 1999.

[16] S. Moon, J. Kurose, P. Skelly, and D. Towsley. Correlation of packet delay and loss in the internet. Technical report, Department of Computer Sciencie, University of Massachusetts, 1998.

[17] S. Floyd and V. Jacobson. Random Early Detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1:397–413, 1993.

[18] J. B. Postel. Transmission control protocol. RFC 793, IETF, 1981.

[19] R. T. Braden. Requirements for internet hosts - communication layers. RFC 1122, IETF, 1989.

[20] W. Stevens. TCP slow start, congestion avoidance, fast retransmit and fast recovery algorithms. RFC 2001, IETF, 1997.

[21] M. Allman, V. Paxson, and W. Stevens. TCP congestion control. RFC 2581, IETF, 1999.

[22] V. Paxson. *Measurements and Analysis of End-to-End Internet Dynamics*. PhD thesis, University of California, Berkeley, 1997.

[23] J. Padhye and S. Floyd. Identifying the TCP behavior of web servers. Technical Report TR-01-002, International Computer Science Institute, 2001.

[24] V. Jacobson. Congestion avoidance and control. *ACM SIGCOMM-88*, Agosto 1988.

[25] P. Karn and C. Partridge. Improving round-trip time estimates in reliable transport protocols. *ACM SIGCOMM-87*, 1987.

[26] A. Reyes Lecuona. *Modelado de Tráfico de clientes WWW*. PhD thesis, Universidad de Málaga, 2001.

[27] J. Bennet, C. Patridge, and N. Schectman. Packet reordering is not pathological network behavior. *IEEE/ACM TRANSACTIONS ON NETWORKING*, 7(6):789–798, 1999.

Análisis y desarrollo de sistemas CSCW mediante *toolkits*: El caso de diagnóstico colaborativo con secciones ecográficas

José Carlos de la Fuente Cortés, Miguel Angel Torribios Fernández,
Yannis A. Dimitriadis y Carlos Alberola López
Departamento de Teoría de la Señal y Comunicaciones e Ingeniería Telemática.
Universidad de Valladolid. Camino del Cementerio s/n, 47011 Valladolid
Teléfono: 983 423 666 Fax: 983 423 667
E-mail: {yannis,caralb}@tel.uva.es

Abstract *Computer supported collaborative work (CSCW) systems are of paramount importance in the plethora of network services currently available. However, due to their enormous complexity, CSCW systems pose serious problems in the field of software engineering. In this paper, we study in detail the modeling and CSCW application development facilities at disposal through collaborative software-development* toolkits. *To that end, we analyze two of the most representative of them, namely, the Java Shared Data Toolkit promoted by Sun, and Groupkit, a popular toolkit in academic environments. This analysis has been performed experimentally, by means of a significative case study proposed in a research and development effort, the purpose of which is to create a virtual navigation environment to ease the complex procedure of the interpretation of fetal ecographic slices and their integration into three-dimensional structures. Despite the already-demonstrated efficiency of* toolkits *for applications development, we found a lack of both standardization and interoperability, which brings up the need of a further study of systems based on components as applied to this domain.*

1 Introduction

Los sistemas telemáticos están teniendo un fuerte impacto en muchas facetas de la emergente Sociedad de la Información/Conocimiento. Además del entretenimiento, formación o administración, importantes servicios telemáticos apoyan y cambian los modos de trabajo. En este sentido, el campo de CSCW (*Computer Supported Collaborative Work* o Trabajo Colaborativo apoyado por Ordenador) estudia y ofrece soluciones a actividades de trabajo colaborativo tales como *workflow*, coordinación, toma de decisiones, etc [1].

Sin embargo, el desarrollo de software de sistemas CSCW es una tarea muy compleja. Por un lado, es necesario conocer profundamente las actividades y dinámicas sociales, típicamente a través de técnicas de análisis y diseño participativo [2]. Por otro lado, se introducen nuevos problemas de interfaces hombre-hombre (no sólo hombre-máquina), que se traducen a tareas de sincronización, control de concurrencia, coordinación, etc., típicas de sistemas distribuidos. Toda esta problemática ha conducido a la propuesta de *toolkits* que faciliten el desarrollo de aplicaciones colaborativas, tanto síncronas como asíncronas. Aquí hay que matizar, que nos referimos a nuevas aplicaciones que incorporen en su diseño explícitamente el carácter colaborativo (transparentes) versus otras que simplemente son extensiones de aplicaciones monousuario existentes (opacas a la colaboración). Ejemplos de *toolkits* colaborativos incluyen a JSDT *Java Shared Data Toolkit* [3], Groupkit [4], para sistemas transparentes, o Habanero [5], y Netmeeting SDK para aplicaciones CSCW opacas a la colaboración. Sin embargo, ninguno de ellos se ha convertido a un estándar debido a la complejidad intrínseca de los sistemas CSCW, así como a la constante evolución de las plataformas distribuidas y de los lenguajes de programación. En este sentido, podemos observar nítidamente la tendencia actual hacia sistemas basados en componentes que pretenden dar una respuesta eficiente y estandarizada a este problema [6].

En este trabajo se presenta un estudio de la problemática relacionada al desarrollo de sistemas CSCW mediante *toolkits* colaborativos. Además de otros estudios similares de nuestro grupo en el dominio de aprendizaje colaborativo [7], aquí nos centramos en el campo de telemedicina. Este sector es de vital importancia para nuestra sociedad y este hecho se refleja ampliamente tanto en las políticas oficiales como en el mundo académico.

Diversos problemas de telemedicina se pueden considerar como casos típicos de CSCW, con máximo exponente el diagnóstico colaborativo entre médicos especialistas situados remotamente. Múltiples sistemas se han propuesto para este problema, pero ninguno de ellos se puede considerar como una solución global, dada la especificidad de las tareas médicas, las diferencias en infraestructuras o la naturaleza de la información a compartir [9].

El caso de estudio empleado en este trabajo está enmarcado dentro del proyecto diSNei, financiado por Fondos FEDER, y trata del diagnóstico en obstetricia a través de secciones ecográficas [8]. Este

calidad de las imágenes ecográficas, así como en la asimetría en el conocimiento entre el médico que realizó las ecografías y el especialista distante. Por ello, surge el problema interesante de cómo establecer el entendimiento común (*common ground*) [13], que permita un diagnóstico eficiente al mismo tiempo que eficaz.

Por todo ello, en el presente artículo se analizan los problemas de ingeniería de software asociados a CSCW, basándonos en el caso de estudio de diagnóstico colaborativo a partir de ecografías. Para ilustrar el problema del uso de *toolkits* colaborativos se evaluarán dos alternativas: la primera utiliza el *toolkit* Groupkit y el lenguaje TCl/Tk, mientras que la segunda el *toolkit* JSDT y el lenguaje Java. Así, podremos analizar importantes propiedades tales como reutilización, estandarización, rendimiento conseguido, etc.

En la sección 2 se ofrece material de apoyo tanto para el proyecto diSNei y las correspondientes tareas de diagnóstico colaborativo, como para el problema general de ingeniería de software en sistemas CSCW. En la siguiente sección, se describen las distintas fases de modelado y desarrollo del sistema que corresponde a nuestro caso de estudio, para poder así pasar a su evaluación y discusión, principalmente desde el punto de vista de la ingeniería de software. Finalmente, en la última sección se presentan las principales conclusiones y se marcan las actuales líneas de investigación en este problema.

2 Fundamentos y contexto del problema

En esta sección se introduce a los distintos aspectos, referidos tanto al caso de estudio (telemedicina, diagnóstico colaborativo), como al problema de ingeniería de software asociado al desarrollo de sistemas CSCW.

2.1 Telemedicina y el proyecto Disnei

Las tecnologías de la información en general, y el avance de las capacidades de procesado y almacenamiento de información en particular, han permitido que el campo de la medicina, digamos, *computacional* haya avanzado notablemente en las dos últimas décadas. Este cambio ha sido orientado fundamentalmente en la dirección de los esquemas de inspección mínimamente invasivos, y de los esquemas de realidad virtual y realidad aumentada.

En fase de desarrollo, y con tecnología enteramente propietaria, se encuentra el proyecto diSNei[1][8], proyecto confinanciado con fondos Feder, el cual está siendo llevado a cabo en colaboración entre las dos escuelas de Ingenieros de Telecomunicación de las universidades de Las Palmas de Gran Canaria

[1] El acrónimo procede de *Diseño Integrado de un Segmentador y Navegador de Estructuras Internas*

construcción de un entorno de navegación virtual por estructuras fetales, el cual ha sido motivado por numerosas razones. Entre ellas, y probablemente la principal, se encuentra el hecho de la enorme dificultad de llevar a cabo una reconstrucción mental de estructuras tridimensionales humanas a partir de secciones sucesivas, tomadas éstas de forma arbitraria, y, asimismo, consistentes en un soporte de imagen altamente ruidoso, con información presente básica (y en cierta manera, únicamente) en los contornos de las estructuras a observar. Bien es cierto que, en la actualidad, un ecografista experimentado tiene esta capacidad y está acostumbrado a tal entorno, pero no es menos cierto, y así hemos sido informados por los especialistas del ramo, que hasta conseguir tal destreza se han tenido que llevar a cabo numerosos ejercicios de integración espacial. Entendemos por tanto que un esquema que consiga, a partir de secciones ecográficas tomadas de forma secuencial (e idealmente quasi-paralela), reconstruir estructuras fetales y representarlas en un espacio 3D, y que, asimismo, permita superponer los datos originales sobre tales estructuras, facilitaría enormemente la tarea de comprensión de la compleja realidad inspeccionada, tanto a expertos en el tema, como, sin duda, a neófitos en el mismo.

Un segunda fuente de motivación, y central cara a este artículo, es el hecho mencionado extraoficialmente entre especialistas del ramo, como el *síndrome del ecografista solitario*. La práctica diaria, y el elevado número de pacientes hace que las inspecciones se conviertan en cuestiones un tanto rutinarias, donde el especialista busca las secciones adecuadas en las que pueda llevar a cabo las mediciones protocolarias oportunas con el objetivo de conocer si el desarrollo fetal es el previsto, de forma acorde con la edad gestacional conocida de antemano. De aquí pueden surgir pérdidas de detalles tal vez importantes, o incluso la necesidad de enviar a la paciente a centros de referencia cuando las exploraciones denotan la presencia de elementos de difícil interpretación.

Ante esta situación, entendemos que el desarrollo de un entorno donde el especialista disponga de un foro de compartición de imágenes y volúmenes, así como de intercambio de información de forma tanto síncrona como asíncrona puede ser enormemente beneficioso para evitar dicho síndrome, así como las molestias causadas a las pacientes en el caso en que los traslados físicos sean necesarios para inspecciones más exhaustivas. Entendemos, asimismo, que el estado actual de las redes de comunicaciones permiten que tal propuesta sea enteramente realista. De hecho, ha habido propuestas en la literatura reciente, las cuales, han sido oportunamente analizadas, y algunas de las mismas se exponen a continuación.

vo

A pesar de que el paradigma de CSCW se remonta a varias décadas, no ha sido hasta mediados de los 90 cuando se ha realizado un estudio más riguroso para la medicina, sin duda propiciado por la mejora tecnológica y el abaratamiento de los equipos.

En 1994, en el Hospital Universitario Rudolf Virchow de Berlín (*German Heart Institute*), se pone en marcha el proyecto *Bermed* [9], la primera implementación real de que tenemos constancia de un sistema de telemedicina colaborativa. Su finalidad es "usar computación avanzada y tecnologías de comunicaciones para mejorar el apoyo informático a los médicos". Los datos de los pacientes se encuentran distribuidos, y el acceso se produce a través de un metarregistro. Éste es uno de los puntos fuertes del proyecto, la integración de datos y la previsión de una posible ampliación del sistema. Ofrece dos tipos de servicio: asíncrono (acceso remoto de expertos a los datos del paciente) y síncrono (facilidades de conferencia en la que dos expertos comparten información de manera simultánea). Dentro de la aplicación se han integrado las siguientes facilidades: canal de audio, canal de vídeo, espacio compartido, manejo del espacio de trabajo, telepunteros, seguridad y adaptabilidad. Como medio de transmisión usan ATM o un acceso primario RDSI (2 Mbps), debido a los grandes requerimientos de ancho de banda del sistema.

En 1998, *Makris et al.* [10] plantean la necesidad de optimizar el uso de los medios existentes para evitar el aumento de coste que supone una aplicación de CSCW para medicina. Su aplicación, *Teleworks*, ofrece a los médicos la posibilidad de "trabajar de forma conjunta en lugares geográficamente distantes". Los objetivos de la aplicación son dar un apoyo al diagnóstico y explotar de una manera óptima las imágenes médicas. Para lograr un buen funcionamiento sobre redes de baja velocidad se prescinde de aplicaciones como la videoconferencia. Sin embargo, aunque según los autores la aplicación funciona bien usando modems de 28.8 Kbps, el entrono de trabajo óptimo es mediante el uso de ATM.

Más cerca de nosotros, debemos mencionar el proyecto europeo ACTS BONAPARTE, en el que *E. Gómez et al.* [11] (1998) plantean un sistema de CSCW para medicina, basado en sistemas de información distribuidos. La colaboración puede hacerse entre especialistas, o bien entre un especialista y un médico general. Las facilidades que ofrece es sistema son: Telerradiología avanzada, diagnóstico cooperativo en tiempo real (mediante el uso de videoconferencia), acceso remoto a archivos de imágenes médicas y telepresencia en una sesión clínica.

Bouillon et al. [12] (1999) plantean una arquitectura colaborativa para trabajar sobre señales médicas de una dimensión (señales neurofisiológicas).

Respecto a aplicaciones colaborativas basadas en imagen ecográfica, podemos mencionar la propuesta la colaboración entre dos especialistas en un entorno ecocardiográfico. En el proceso de adquisición de las imágenes existen parámetros como la localización del transductor o los parámetros del escáner que se pierden. Esta pérdida de información junto al carácter ruidoso de los datos hace que la colaboración entre especialistas sea una tarea deseable pero realmente difícil. Berlage propone el concepto de "artefacto común" (*common artefact*) como elemento central para el entendimiento entre especialistas. El artefacto u objeto común es un metáfora que sirve como mediador entre los cooperantes y permite establecer entre ellos el fondo común o *common ground*. Este objeto sirve como referencia central y puede manipularse por todos los integrantes de la cooperación. En un entorno cooperativo médico como el descrito, el objeto común sería el modelo 3D que sirve como nexo de unión entre los datos "crudos" y la concepción mental que tiene el especialista de los mismos. La conciencia de grupo se ve totalmente respaldada por el uso del concepto de objeto común.

La dificultad que presenta la colaboración con imágenes de ultrasonidos reside principalmente en la pérdida de la información asociada a la inspección del paciente. Éste es un problema que Berlage presenta pero que no indica cómo resolver.

2.3 El problema de la ingeniería de software en CSCW

Como ya se ha comentado en la sección 1, el desarrollador de aplicaciones CSCW se enfrenta a problemas añadidos, ya que debe tener en cuenta aspectos técnicos tales como sincronización, concurrencia, comunicación, coordinación y consistencia. Además, debe entender e incorporar aquellos factores y aspectos humanos necesarios para que el trabajo sea eficaz.

La alternativa de uso de programación a bajo nivel se puede considerar ineficaz, ya que la gestión de las comunicaciones mediante sockets, RMIs, etc es muy costosa tanto para su desarrollo como para su mantenimiento. Por ello, se han considerado alternativas de más alto nivel, en las cuales el desarrollador puede centrarse en los aspectos más abstractos o en las características específicas de su aplicación. Los *toolkits* colaborativos o *kits* de desarrollo para software colaborativo pretenden proporcionar los componentes clave para satisfacer las necesidades habituales de *groupware*, además de reducir el esfuerzo de desarrollo, de permitir una rápida creación de prototipos y de incrementar la calidad de las aplicaciones multiusuario. Si enfocamos a los sistemas síncronos, podríamos destacar unas características que deberían ofrecer los *toolkits* colaborativos [4]:

- Un espacio compartido con múltiples modos de interacción, tales como gestos o anotaciones gráficas, que permita un trabajo fluido. En

terfaces gráficos estandarizados, así como permitir múltiples vistas de los mismos datos.

- Gestión de registro de usuarios y grupos, tanto al inicio como en cualquier otro momento de la sesión, y políticas de control de acceso a las conferencias.
- Gestión de conferencias y de sesiones persistentes, basándose en un mecanismo robusto de comunicaciones.
- Capacidad de integración con otros medios, tales como vídeo o voz, o aplicaciones monousuario.

En la tabla 1 se presenta un análisis más detallado de las características de diversos *toolkits* propuestos en la literatura,

Toolkit	Arq	Conc	Leng	Ind	Web
CCF	Rep	S/B	C	No-SU	No
GroupKit	Rep	Ban	tcl/tk	Sí	No
COAST	Rep	SO	ST	Sí	No
NetMeeting	Mod	Gen	C++	No-SW	Sí
JSDT	Cent	Tok	Java	Sí	Sí
Mushroom	Hib	Gen	Java	Sí	Sí
Promondia	Cent	Der	Java	Sí	Sí
Habanero	Cent	Gen	Java	Sí	Sí

Tabla 1: Comparativa de diversos toolkits. Leyenda: Rep: replicada. Mod: modular. Cent: centralizada. Hib: híbrida. S/B: serie/bloqueo. Ban: banderas. SO: ser. optimista. Gen: genérico. Tok: token. Der: derechos. ST: SmallTalk, SU: sólo Unix. SW: sólo windows.

Hemos tenido en cuenta los conceptos de la arquitectura subyacente (arq), el sistema de control de concurrencia (Con), el lenguaje de programación asociado (Leng), su independencia de la plataforma software (Ind), y la capacidad de integración en entornos Web (Web). Así, se puede observar que existen muchos elementos en común entre los diversos *toolkits*, pero al mismo tiempo hay serias discrepancias, y una incapacidad de comunicación entre aplicaciones creadas con distintos *tolkits*. En este sentido, es necesario analizar hasta qué punto se puede reutilizar el análisis y diseño realizado en alto nivel cuando se emplean distintos *toolkits*, elegir los más estandarizados y con mayor vida esperada, y evaluar la calidad de las aplicaciones desarrolladas con ellos.

Todos estos aspectos son especialmente importantes en el caso de estudio que usamos, donde las tareas son complejas, y se requieren prototipos robustos y estandarizados, que se puedan desarrollar en un período de tiempo corto.

3 Estudio del caso y discusión

Empezamos esta sección con el modelado y el uso del caso de estudio y a continuación presentamos ingeniería de software derivada de nuestro análisis experimental.

3.1 Descripción del caso de estudio

La plataforma *software* objeto de este artículo se ha estructurado acorde con los bloques funcionales que mostramos en la figura 1), y que pasamos a describir a continuación. Intercalamos, en esta descripción, el modelado de los bloques más relevantes basado en el lenguaje UML [14].

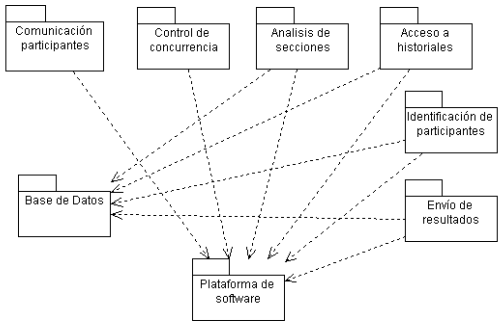


Figura 1: Subsistemas de la plataforma.

Comunicación entre los participantes: (figura 2): La comunicación entre los distintos participantes que constituyen una sesión va a ser de dos tipos. Por un lado se puede establecer una comunicación en tiempo real, a través de un sistema de *chat*, donde los médicos pueden intercambiar sus ideas, opiniones, o cualquier tipo de comentario de manera síncrona (caso de uso *Comunicación síncrona*). Por otro, la plataforma incluirá un sistema de mensajes asíncronos, con la finalidad de poder intercambiarse pequeños comentarios a modo de notas (caso de uso *Mensaje asíncrono*). Además, se intentará ahorrar tiempo en la comunicación incluyendo una serie de frases que los participantes usan normalmente, de modo que no tengan que escribirlas, sino sólo elegirlas con el ratón. Estas frases se podrán editar, y salvar en un fichero para conferencias posteriores (caso de uso *Frases comunes y las tres extensiones de la figura*).

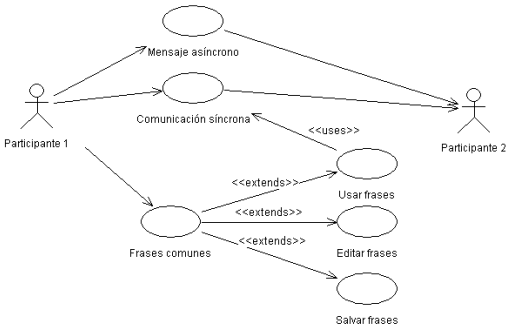


Figura 2: Diagrama de casos de uso: Comunicación entre participantes.

rrencia es necesario para evitar inconsistencias entre los distintos participantes de una sesión. Es necesario proponer algún mecanismo de control que gestione el acceso a los objetos compartidos. Se ha optado por un mecanismo de control de concurrencia basado en turnos, de modo que sólo un participante posee el turno en cada momento. Éste puede realizar todo tipo de operaciones, mientras que el resto tienen un uso limitado. Sólo aquel que posee el turno puede acceder a ciertos recursos comunes. Para la transferencia del turno se sigue una política que consiste en almacenar todas las peticiones en una cola, y asignar el turno al primer elemento de la cola. Se trata, por tanto, de una cola FIFO (*First In, First Out*).

Análisis de secciones ecográficas: Esta parte es la más importante a la hora de realizar el diagnóstico colaborativo por parte de los especialistas, ya que trata del estudio de las imágenes médicas, en este caso ecografías fetales. Los médicos pueden realizar ciertas operaciones (véase diagrama de casos de uso en figura 3) sobre las imágenes, con el fin de obtener un mejor diagnóstico:

- Visualizar las distintas secciones de un volumen ecográfico (*cargar sección*).
- Marcar diferentes regiones y contornos dentro de una sección (*marcado de regiones*).
- Recorrer todo el volumen (*recorrer volumen*).
- Ver un vídeo de todo o parte del volumen (*animación del volumen*).
- Realizar comentarios sobre las secciones (*comentarios de sección*).
- Segmentar las distintas regiones de una sección (*segmentación*).

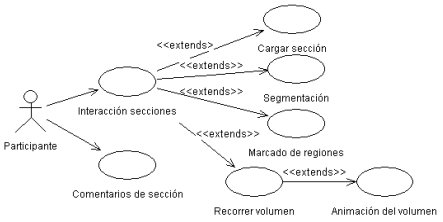


Figura 3: Diagrama de casos de uso: Análisis de secciones

Acceso a historiales médicos: Esta utilidad permite a médicos y especialistas almacenar y consultar los historiales médicos de los pacientes. En nuestro caso, es necesario que el ecografista conozca toda la información clínica del paciente: enfermedades padecidas, antecedentes sufridos, resultados de los análisis realizados, evolución del volumen de líquido a lo largo del embarazo etc, ... También se almacenan las medidas realizadas al feto en las diferentes semanas de la gestación.

Identificación de participantes: Antes de establecer una conferencia, ya sea síncrona o asíncrona, se autentifica al usuario con el fin de proporcionar *login* y *password*, que les será solicitado cada vez que inicie la sesión. Esto es necesario debido a la confidencialidad que requieren los datos médicos. Un servidor de autentificación se encargará de validar a los distintos usuarios.

Envío de resultados: Mediante esta funcionalidad, se permite al participante el envío de información al resto de usuarios; tal envío puede ser llevado a cabo de forma transparente al participante (lo cual se hará siempre que éste lleve a cabo un proceso de segmentación), o bien de forma intencionada por parte del participante. Esta información puede ser imágenes médicas, resultados de la segmentación, ficheros, en suma, cualquier información que un participante considere relevante.

3.1.1 Una ejecución de la aplicación

Como hemos comentado anteriormente, la aplicación dispone de un sistema de autentificación de los usuarios que pretenden hacer uso de la misma. Para tal fin existe por un lado, una tabla en la base de datos con los nombres de entrada y palabras de paso de los médicos autorizados (estas últimas están cifradas) y por otro, un servidor de autentificación transparente a todos los usuarios, que recibe el *login* y *password* de cada médico los cifra, y consulta la tabla anterior para ver si el médico está dado de alta. También, para una mayor seguridad, se ha compilado toda la aplicación de forma que personas ajenas que pudiesen conseguirla, no puedan hacer nada con el código.

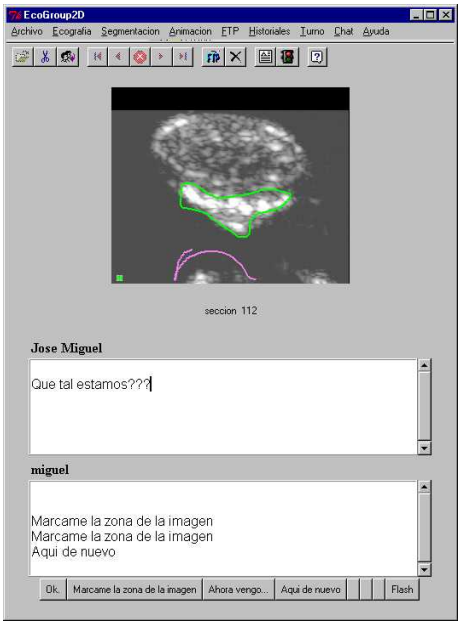


Figura 4: Interfaz principal de la aplicación

Asumiendo que el usuario ha sido capaz de entrar en la aplicación observará en pantalla el llamado *Session Manager*, el cual muestra información relativa a las conferencias que ya hay inicializadas, y los usuarios que están presentes en la conferencia actualmente señalada con el ratón. A partir de

sión (a partir del menú *Conferences*) que puede ser síncrona o asíncrona, o unirse a una de las ya existentes. En ambos casos, aparecerá el interfaz que se muestra en la figura 4.

Este interfaz da pie a comentar alguna de las funcionalidades previstas en la aplicación:

- Todos los usuarios ven la misma ecografía; el usuario que se encuentre en posesión del turno puede avanzar o retroceder en ecografías, o mostrar el vídeo (avance/retroceso rápido) de las ecografías. Para evitar problemas de trasiego de información por la red se ha decidido que todos los especialistas dispongan de las imágenes localmente.
- Cuando un usuario que ha solicitado el turno consigue recibirlo, los iconos del menú de su interfaz se activan; el usuario que acaba de liberar el turno observa el efecto contrario, esto es, desactivación de tales teclas.
- Todos lo usuarios pueden hacer marcaciones sobre las ecografías en cualquier momento. Cada usuario tendrá un color distinto, y existen telepunteros que muestran en todo instante la ubicación y movimientos de todos los participantes en la conferencia.
- Existen ventanas de *chat* (una por usuario, pues se asume que un número reducido de especialistas harán uso de la aplicación), así como frases predeterminadas editables para una escritura más rápida.
- Existe la posibilidad, asimismo, de hacer marcaciones y comentarios que se almacenarán en la base de datos para futura referencia. En este caso, el funcionamiento es similar al chat, aunque en una ventana aparte que se abre de forma simultánea en la pantalla de cada conferenciante.
- Como hemos indicado, existe la posibilidad de segmentar de forma semiautomática los datos bajo estudio. No obstante, dado que la segmentación es una tarea asíncrona, si un conferenciante intenta acceder al menú de segmentación durante una conferencia síncrona, obtendrá un mesaje de error. Cuando concluya la tarea de segmentación, los resultados se enviarán al resto de los usuarios de la aplicación (dados de alta en la base de datos) de forma transparente mediante protocolo ftp.

Asimismo, el sistema dispone de la posibilidad de enviar a cualquier otro usuario todo tipo de información de forma explícita. Para ello, se abre un interfaz donde se puede seleccionar al usuario o usuarios destino del envío, y el fichero que se quiere enviar.

Finalmente, la aplicación permite un acceso asíncrono a los historiales de los pacientes objeto de análisis. Todo usuario puede cambiar los datos allí presentes a voluntad, y tales cambios se reflejarán en el siguiente acceso a la base de datos del resto de

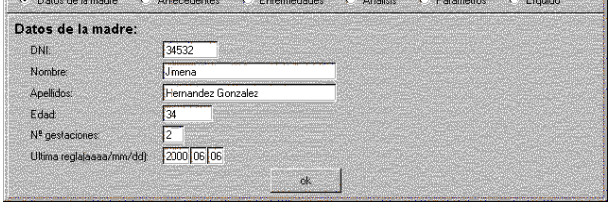


Figura 5: Acceso a historiales

los usuarios. Un ejemplo de historial es el mostrado en la figura 5.

3.2 Evaluación del caso desde la perspectiva de ingeniería de software

En este caso de estudio se pretendía conseguir la funcionalidad adecuada de la aplicación dentro de su contexto real, al mismo tiempo que se querían estudiar aspectos más generales de ingeniería del software en CSCW.

La primera y más importante restricción se refería al análisis y diseño participativo. En este caso, se quería entender la práctica actual de los médicos y al mismo tiempo darles a entender las posibilidades de la solución técnica. Por ello, además de entrevistas periódicas y observaciones, se decidió establecer la comunicación con ellos a partir de prototipos reducidos pero reales. Así, se requería un desarrollo rápido de prototipos, elemento casi sinónimo al caso de *toolkits* colaborativos. Por otro lado, el sistema CSCW debía aprovechar software existente en el grupo de investigación relacionado con el procesamiento de imágenes ecográficas. El uso anterior del *toolkit Vtk* con el lenguaje interpretado Tcl/Tk casi imponía el uso del conjunto de herramientas compuestas por Groupkit, el lenguaje interpretado Tcl/Tk y la interfaz a bases de datos Tcl/SQL para el desarrollo del primer prototipo del sistema. Dado el carácter no orientado a objeto, se optó por un análisis y diseño funcional y estudio paralelo de la usabilidad de UML.

Todas las fases del ciclo de vida hasta el primer prototipo necesitaron un esfuerzo de 8 mes/hombre, hecho que enfatiza la contribución del uso de un *toolkit* en el desarrollo del sistema CSCW. En este momento, ya se podía plantear el uso de otra alternativa que permitiera analizar y comparar varios aspectos. Se optó por el uso de conjunto de herramientas compuestas por el *toolkit* JSDT el lenguaje Java, el interfaz JDBC y el lenguaje de modelado UML como herramienta de desarrollo, dada su amplia difusión actual y su carácter comercial versus el carácter académico de la solución anterior.

En este sentido se pudo llegar al segundo prototipo con funcionalidad suficiente para los médicos con un esfuerzo adicional de 10 mes/hombre. Hay que destacar que a pesar de la experiencia anterior de nuestro grupo en el uso de estas herramientas, los programadores no la conocían. Así, en un pe-

prototipos robustos basados en distintas tecnologías y con prácticamente la misma funcionalidad.

Este éxito en el desarrollo del sistema CSCW se puede atribuir principalmente al hecho de que ambos *toolkits* comparten/ofrecen servicios básicos muy similares (conferencias, sesiones, concurrencia, usuarios etc ...). Así se pudo contrarrestar parcialmente el importante problema que surgió por la inadecuación de UML en el análisis funcional del primer prototipo. Por otro lado, se confirmó que los *toolkits* ofrecían el soporte adecuado por lo menos para aplicaciones colaborativas síncronas y transparentes. Al mismo tiempo, emergió el principal problema del uso de estos *toolkits* que se refiere a su no estandarización y su falta de interoperabilidad. En este sentido, los prototipos generados tenían módulos que no eran ni intercambiables, ni se podían comunicar entre sí. Este hecho sugiere que necesitásemos basarnos en un nivel superior, que fuera independientes de las plataformas software y hardware subyacentes. La solución a este problema se encuentra en los sistemas basados en componentes, que deben aprovechar unas construcciones abstractas a nivel de modelado, y que fueran independientes de las plataformas distribuidas (DCOM, CORBA, I2EE) que gestionan la interoperabilidad. Para ello es necesario cambiar el enfoque y pasar del uso de *toolkits* al diseño de un sistema basado en componentes en este dominio.

Concluiremos esta sección con una reflexión más concreta sobre las características de los lenguajes y *toolkits* empleados, lo cual se traduce de manera directa en las características diferenciales de los prototipos creados. Para llevar a cabo tal reflexión haremos uso de las tablas que se incluyen a continuación:

Java	**Tcl/Tk**
Orientado a objetos	Procedimental
Pseudo-compilado	Interpretado
Multiplataforma	Multiplataforma
Prog. rel. rápida	Prog. rápida
Seguridad	-
Aplicaciones Web	-
Entorno industrial	Entorno académico

Tabla 2: Comparación de Java y Tcl/Tk.

La tabla 2 muestra las diferencias entre el lenguaje instrumental que utiliza las primitivas del *toolkit* correspondiente. En este sentido se ha podido comprobar que el desarrollo con Tcl/Tk era más rápido pero su uso sufre varias restricciones, tales como la falta de integración con entornos Web, su influencia y origen en el mundo académico exclusivamente, así como su base en un paradigma no orientado a objeto. Por otro lado, en la tabla 3 se pueden observar unas diferencias sustanciales entre el Groupkit y el JSDT que no suponen una ventaja cualitativa de ninguno de los dos *toolkits*. El usuario de los *toolkits* debería tener en cuenta las restricciones de su necesidad de integración en el mundo Java. En este caso se podría comentar que el uso del Netmeeting SDK+Visual C++ en otra aplicación de diagnóstico colaborativo llevada a cabo por nuestro grupo [15] mostró las restricciones técnicas de esta solución (aplicaciones opacas a la colaboración), así como su dependencia absoluta del entorno Microsoft.

JSDT	**Groupkit**
Java	Tcl/Tk
Centralizada	Replicada
Token	Banderas
Sockets(canales)	RPC Multicast
Applets	No applets
No soporta entornos	Soporta entornos

Tabla 3: Comparación de *toolkits* JSDT y Groupkit.

Por último, la tabla 4 muestra la calidad y eficiencia de los prototipos conseguidos, donde se pueden destacar los problemas del prototipo Java en velocidad de ejecución y en su uso por un gran número de usuarios. Por otro lado se puede ver que se consiguió casi la misma funcionalidad, aún empleando dos entornos muy distintos.

Característica	**Prot. Java**	**Prot. Tcl/Tk**
Coordinación	WYSIWIS	WYSIWIS
Sincronismo	Turnos (token)	Turnos
Bases de datos	Ses. sínc. y asínc.	Ses. sínc. y asínc.
Animación	Independiente	Conjunta
Marca secciones	PcnU*	sPcnU**
Comunicación	Chat	Talk múltiple
Velocidad	Lento	Relativ. rápido

Tabla 4: Comparación de prototipos Java y Tcl/Tk. * Problemas con número de usuarios. ** Sin problemas con número de usuarios.

4 Conclusiones y futuro trabajo

En este artículo se ha tratado el problema de desarrollo de software en el dominio de CSCW. Para ello, se utilizó un importante caso de estudio que se refiere al diagnóstico colaborativo mediante imágenes de ultrasonidos en obstetricia. En esta aplicación, las tareas de diagnóstico son complejas y la baja calidad de las imágenes requiere una colaboración efectiva entre los médicos especialistas.

Así, a través de este caso de estudio, se pudo analizar el uso de *toolkits* como medio para aliviar la complejidad en las tareas de modelado y desarrollo de sistemas CSCW. La creación de dos prototipos, basados en los *toolkits* JSDT y Groupkit respectivamente, demostró la eficiencia de este modelo de desarrollo, así como la existencia de muchos servicios comunes ofrecidos por ambos *toolkits*. Por otro

rización y la falta de interoperabilidad en estas soluciones, que a largo plazo limita su ámbito de aplicación e impide la coexistencia e interoperabilidad de aplicaciones desarrolladas por distintos *toolkits*. En este sentido, a pesar de las importantes ventajas en el uso de *toolkits*, el camino a seguir consiste en la creación de un sistema basado en componentes para este dominio específico (CSCW). Así, se podrían reutilizar las definiciones abstractas ed los componentes (servicios e interfaces), consiguiendo inedpendencia de las plataformas de los sistemas distribuidos subyacentes.

Nuestro grupo está trabajando actualmente en este camino dentro del contexto de dos proyectos I+D (uno nacional y otro regional) en el dominio de aprendizaje colaborativo apoyado por ordenador (CSCL) que es una especialización del CSCW.

Finalmente, se está avanzando en la mejora de las prestaciones de la aplicación, objeto del caso de estudio. En este sentido, se prevé incluir el uso de modelos 3-D para los fetos a partir de las secciones ecográficas, así como el empleo de artefactos comunes (sondas) que permitan reducir el desequilibrio entre especialistas y mejorar el entendimiento común en el espacio de colaboración.

Agradecimientos

Este trabajo ha sido financido por la Comisión Interministerial de Ciencia y Tecnología y el Fondo Europeo de Desarrollo Regional a través del proyecto 1FD-97-0881-C02-02; los autores quieren expresar su agradecimiento a los facultativos del hospital Río Carrión de Palencia, Dres. Julio Díaz González y Jesús María Andrés de Llano por su consejos y sugerencias durante la realizaciń de la plataforma aquí descrita. Asimismo, debemos hacer mención de las personas que han contribuido a la creación de la versión monousuario del navegador por estructuras fetales, concretamente, los ingenieros D. Raúl San José Estépar, D. Miguel Ángel Martín Fernández y D. Santiago Aja Fernández, de la Universidad de Valladolid, y el grupo del profesor Ruiz Alzola de la Universidad de Las Palmas de Gran Canaria.

Referencias

[1] C. Ellis, S. Gibbs and G. Rein, "Groupware: Some issues and experiences," *Communications of the ACM*, vol. 34, pp. 44–53, January 1991.

[2] M. Kyng, "Designing for cooperation - cooperating in design," *Communications of the ACM*, vol. 34, pp. 65–73, December 1991.

[3] R. Burridge, "Java shared data toolkit user guide," http://developer.javasoft.com/developer/earlyAccess/jsdt/index.html, April 1999.

[4] M. Roseman and S. Greenberg, "Building realtime groupware with Groupkit, a groupware *man Interaction*, vol. 1, pp. 66–106, March 1996.

[5] C.S.A. Chabert, "Java object sharing in Habanero," *Communications of the ACM*, vol. 41, pp. 69–76, June 1998.

[6] D. Krieger and R. Adler, "The emergence of distributed component platfroms," *IEEE Computer*, vol. 31, pp. 43–53, March 1998.

[7] C. Osuna, "DELFOS: Un marco telemático-educativo basado en niveles orientado a situaciones de aprendizaje colaborativo," Tesis Doctoral, ETSIT, Univesridad de Valladolid, Febrero 2000.

[8] Carlos Alberola *et al.*, diSNei: A Collaborative Environment for Medical Images Analysis and Visualization. In Anthony A. DiGioia and Scott Delp (eds.): Medical Image Computing and Computer-Assisted Interventions, Lecture Notes in Computer Science, Springer-Verlag, Berlin Heidelberg, New York 2000, pp.814-823.

[9] L. Kleinholz and M. Ohly, "Supporting cooperative medicine: The bermed project," *IEEE Multimedia Mag.*, pp. 44–53, 1994.

[10] L. Makris, I. Kamilatos, E. V. Kopsacheilis, and M. G. Strintzis, "Teleworks: A cscw application for remote medical diagnosis support and telecomsultation," *IEEE Trans. Inform. Technol. in Biomed.*, vol. 2, pp. 62–73, June 1998.

[11] E. J. Gómez, F. del Pozo, E. J. Ortiz, N. Malpica, and H. Rahms, "A broadband multimedia collaborative system for advanced teleradiology and medical imaging diagnosis," *IEEE Trans. Inform. Technol. in Biomed.*, vol. 2, pp. 146–155, Sept. 1998.

[12] Y. Bouillon, F. wendling, and F. Bartolomei, "Computer-supported collaborative work (cscw) in biomedical signal visualization and processing," *IEEE Trans. Inform. Technol. in Biomed.*, vol. 3, pp. 28–31, Mar. 1999.

[13] T. Berlage, "Augmented-reality communication for diagnostic tasks in cardiology," *IEEE Trans. Inform. Technol. in Biomed.*, vol. 2, pp. 169–173, Sept. 1998.

[14] P. Software and Systems, "Modeling systems with UML", http://www.popkin.com, 1998.

[15] N. Pérez, M de Cabo, Y.A. Dimitriadis, y E. Fernández "Sistema de videoconferencia aplicado en telepatología," *Actas de las Segundas Jornadas de Ingeniería Telemática*, Madrid, Septiembre 1999.

CARMEN, un proyecto internacional de telecardiología

Pedro López[1], David Rincón[1], Miguel Ángel Viñas[2], Enrico Frumento[3], Roberto Fogliardi[4]
[1]Dpto.de Ingeniería Telemática – UPC (Barcelona), [2]Centre de Visió per Computador (Barcelona),
[3]CEFRIEL (Milán, Italia), [4]Aethra srl (Ancona, Italia)
E-mail: david.rincon@upc.es

***Abstract**. This paper presents CARMEN, a teleconsulting application that has been developed by Italian and Spanish partners. This application provides communication between physicians located at different hospitals through the use of TCP/IP and T.120 protocols over LAN, MAN and WAN networks. CARMEN is capable of transmitting hemodynamics films to a remote site and managing the following consultation through an integrated audio-videoconferencing system. The paper describes the application architecture and its impact in the work of physicians. CARMEN is in its early testing phase in Lombardia and soon in Catalunya, and its previous version CAROLIN has been installed in some hospitals in both countries.*

Introducción

Este artículo presenta diversos aspectos relacionados con el proyecto CARMEN (*Co-operative Application for Remote MEdical consultatioN*), enmarcado en el programa TeleRegions [1] de la Unión Europea y desarrollado conjuntamente por centros de investigación de Catalunya (España) y Lombardía (Italia). Orientado básicamente a la telecardiología, pretende proporcionar a los profesionales de la salud una herramienta potente que les ayude en el proceso de diagnóstico y consulta, así como en la gestión y transmisión de los datos clínicos de los pacientes entre hospitales geográficamente alejados.

El texto se estructura en diversos apartados. El primero de ellos introducirá al lector en el marco de la telemedicina, haciendo especial hincapié en los objetivos que dicha disciplina persigue y en las mejoras que la telemática puede introducir en el proceso de diagnóstico. Se pasará a continuación a reseñar los aspectos técnicos más relevantes de una aplicación de telecardiología. Seguidamente se describirá la historia del proyecto CARMEN y se enumerarán sus funcionalidades más destacables. El artículo finaliza con el análisis de los resultados de las primeras experiencias reales de uso de la aplicación, así como los planes de despliegue de la aplicación tanto en Catalunya como en Lombardía.

1. Telemedicina y telecardiología

La calidad de vida de los ciudadanos viene determinada por una serie de factores, entre los cuales el nivel de asistencia médica es un factor de la máxima importancia, por lo que el objetivo prioritario del sistema sanitario es mejorar la cantidad y, sobre todo, la calidad de los servicios ofrecidos.

En este segundo aspecto no cabe ninguna duda que las Tecnologías de la Información y las Comunicaciones (TIC) constituyen una poderosa herramienta.

Dentro de este contexto, surge el concepto de **telemedicina** [2], cuyo objetivo primordial es proporcionar un servicio de salud mejor y más barato mediante la utilización de las TIC. La mejora económica se consigue al eliminar los costes derivados de los desplazamientos tanto del médico como del paciente, mientras que la mejora en el servicio se obtiene al facilitar la consulta de determinadas enfermedades con expertos especializados en la materia.

1.1 ¿Qué puede aportar la telemática a la medicina?

Hoy en día existen equipamientos médicos de adquisición y tratamiento de imágenes muy sofisticados y altamente computerizados, que ayudan a los profesionales de la salud a detectar, prevenir y curar enfermedades de sus pacientes. Entre ellos se encuentran la Resonancia Magnética Nuclear (RMN) y la Tomografía Axial Computerizada (TAC).

Desgraciadamente, no toda la población tiene acceso inmediato a todos estos servicios, ya que son costosos y conllevan una importante dificultad de operación. Por estos motivos tienden a concentrarse en un conjunto muy reducido de hospitales, habitualmente situados en las grandes ciudades, que actúan a su vez como un polo de atracción para los médicos especialistas. Por lo tanto, se tiende de forma natural a un marco de concentración de recursos técnicos y humanos. Tal situación provoca que los enfermos que viven en zonas rurales se vean obligados a desplazarse hacia las grandes metrópolis para resolver las consultas que no pueden ser solucionadas por sus médicos de cabecera, con los consiguientes retardos y costes de desplazamiento.

La telemedicina ayuda a atenuar los efectos de la concentración de expertos y medios. El paciente es atendido en primera instancia por su médico de cabecera en el Centro de Atención Primaria más próximo a su lugar de residencia. Cuando, dada la

gravedad del caso, se juzgue oportuno, establecerá comunicación con el especialista que se encuentra en uno de los grandes hospitales; se llevará entonces a cabo un proceso de teleconsulta con el fin de evitar el desplazamiento del enfermo y los costes asociados.

1.2 Teleconsulta

Una aplicación de **teleconsulta** (también llamadas de "diagnóstico cooperativo") permite a un médico situado geográficamente próximo al paciente efectuar consultas con otros médicos, especializados en la enfermedad del paciente en cuestión pero situados a gran distancia, con el fin de comparar y contrastar opiniones sobre el diagnóstico.

El proceso de teleconsulta puede ser o no interactivo, dependiendo de si el diagnóstico se elabora de forma simultánea y coordinada por los dos médicos (modelo síncrono), o bien se elabora de manera secuencial, estableciendo turnos entre ambos doctores e intercambiando sus resultados (modelo asíncrono; también hay modelos híbridos). Si la interactividad no es necesaria, el proceso se puede realizar mediante correo electrónico. Si el proceso es interactivo, será necesario establecer enlaces de comunicaciones capaces de transmitir imágenes, voz y datos (incluso vídeo a ser posible) entre los dos participantes, con el consiguiente aumento del ancho de banda necesario.

1.3 El caso de la cardiología

A continuación se va a detallar cómo la aplicación de la telemática a la medicina ayuda a paliar los efectos de la concentración geográfica de expertos y medios; en concreto, se detallará el caso de la cardiología y de los Sistemas Públicos de Salud catalán e italiano. De todos modos, dicho ejemplo se puede extrapolar perfectamente a otras especialidades médicas y a otras regiones de Europa.

Las instituciones sanitarias en el ámbito de la cardiología pueden clasificarse en tres categorías diferenciadas, según el tipo y la cantidad de equipamientos médicos de los que disponen. Así, los hospitales del Sistema Público de Salud se dividen en tres niveles [3]:

- Centros de Atención Primaria: disponen de un equipamiento cardiológico limitado.
- Hospitales secundarios: incluyen un departamento de cardiología, y están equipados para realizar angiografías[1].
- Hospitales terciarios: situados generalmente en las grandes ciudades, disponen del equipamiento necesario para cirugía coronaria y pueden realizar angioplastias. Son los hospitales de referencia para el resto de centros.

[1] Una angiografía es una secuencia de vídeo con imágenes de los vasos coronarios y los músculos del corazón. A partir de estas secuencias se pueden detectar lesiones y obstrucciones.

El procedimiento clásico de diagnóstico en cardiología pasa típicamente por diferentes fases, que se detallan a continuación [4]:

- Sospecha: el médico de cabecera detecta síntomas de una enfermedad determinada.
- Comprobación: se realizan pruebas cardiológicas sencillas en el hospital primario para corroborar la sospecha del médico de cabecera.
- Verificación: el paciente se desplaza al hospital secundario donde se le practica una angiografía que ayude a determinar el origen de su dolencia.
- Consulta: si se confirma la gravedad del caso, el paciente se desplaza al hospital terciario, donde el especialista cirujano del corazón le examina y decide una posible intervención.

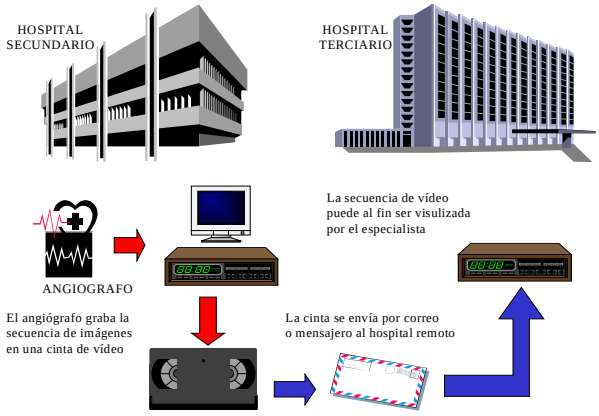


Figura 1: Proceso típico de una consulta en cardiología

El proceso típico es muy costoso, tanto desde el punto de vista económico (material, desplazamientos de paciente y médicos), como temporal (el proceso puede necesitar un tiempo literalmente vital para el paciente, durante el cual la dolencia puede agravarse dramáticamente).

¿Qué puede aportar la telemática en este escenario? El transporte físico de cintas de vídeo puede ser eliminado enviando por red la información que contienen, agilizando el proceso. Además, las herramientas telemáticas permiten el establecimiento de sesiones cooperativas de diagnóstico entre dos o más facultativos. Durante las sesiones interactivas de teleconsulta, médico y especialista contrastarán sus opiniones. Estas sesiones tienen como objetivo principal evitar el traslado del paciente desde el hospital secundario al hospital terciario para ser visitado por el especialista en hemodinámica. El traslado del paciente al centro quirúrgico queda relegado únicamente a los casos en los que la intervención sea estrictamente necesaria. Y, por supuesto, el proceso global se acelera en gran medida, ya que las transmisiones son mucho más rápidas que los viajes del paciente o de las cintas de vídeo que contienen las angiografías.

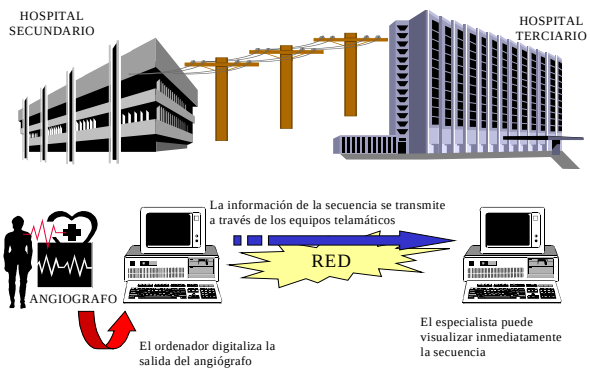


Figura 2: Introducción de la telemática en el proceso de consulta.

En definitiva, la telemática ayuda a reducir tiempos y costes, y facilita el acceso al conocimiento experto, logrando una mejora en la calidad de los servicios de salud. Se consiguen evitar muchos desplazamientos de los pacientes, así como ahorrar cantidades importantes de recursos y, sobre todo, minimizar un tiempo que puede salvar la vida del enfermo.

2 Puntos importantes de una aplicación de telecardiología

A continuación se estudiarán algunos de los puntos más importantes que hay que considerar en el proceso de diseño de una aplicación de telecardiología.

2.1 Volumen de la información

Las secuencias de vídeo utilizadas en cardiología pueden ser relativamente largas, llegando incluso a varios minutos de duración. Esto plantea un grave problema para la transmisión de las mismas, debido al gran tamaño de los ficheros implicados. Sirva de ejemplo el siguiente cálculo: una secuencia de 30 segundos de duración, 25 imágenes por segundo, 512x512 *pixels* por imagen y 8 bits por *píxel* (256 niveles de gris) lleva a un fichero de casi 200 Mbytes de volumen. Si queremos implementar un sistema de telemedicina de uso masivo, con decenas de miles de usuarios potenciales, las cifras son prohibitivas, tanto para su almacenamiento como para la transmisión.

Por tanto, se requerirá un proceso de edición de la imagen así como un algoritmo de compresión de la misma, previos a la transmisión. La edición será llevada a cabo por los cardiólogos, que discriminarán los fragmentos útiles de la secuencia, pero la compresión debe realizarse de forma totalmente transparente al usuario, justo antes de llevarse a cabo la transferencia. Se conseguirá así reducir el tiempo y los costes asociados al envío de secuencias.

La introducción de algoritmos de compresión en imágenes médicas no es un tema trivial, ya que aparecen aspectos legales relacionados con la calidad de las mismas. Si se exige una compresión sin pérdidas (*lossless*), se alcanzarán relaciones de compresión típicas de 2:1 o como mucho de 4:1. Si, en cambio, toleramos unas pérdidas que mantengan la calidad subjetiva de la imagen, podemos llegar a compresiones de 10:1 o incluso 20:1. Las necesidades de la telecardiología basculan entre las dos situaciones, ya que la información se encuentra en el movimiento del corazón (poco sensible a pérdidas), pero también en el análisis de las imágenes de los vasos coronarios (sensible a pérdidas).

2.2 Formato de la información

Un tema importante es decidir cuál será el formato de la información relacionada con el paciente y el diagnóstico, aparte de las imágenes. Se deberá escoger un estándar internacional, con suficiente implantación como para esperar que cualquier máquina moderna de exploración (TAC, RMN, TEP) disponga de una salida de datos en ese formato. Así mismo, sería conveniente que el formato permitiera la integración de la aplicación de teleconsulta y de los sistemas de almacenamiento del hospital. Finalmente debe ser flexible, actual y apoyado por los fabricantes de equipos médicos y la industria informática.

El único estándar que cumple todos estos requisitos es DICOM (*Digital Imaging and COmmunicactions in Medicine*), desarrollado por ACR-NEMA (*American College of Radiology - National Electrical Manufacturers Association*). DICOM [5] integra los aspectos de almacenamiento y comunicaciones. Está estructurado según la filosofía de programación orientada a objetos. El objetivo final de DICOM es permitir interoperabilidad (no sólo interconexión) entre los equipos de diferentes fabricantes, definiendo objetos con información explícita sobre imágenes, estudios, informes, pacientes, etcétera. Por tanto, una buena aplicación de telecardiología debe ser capaz de importar y exportar ficheros DICOM.

2.3 Transmisión de la información

El gran volumen de información asociada a un examen impide que sea utilizada “en directo”; es decir, que la secuencia de vídeo sea transmitida a todas y cada una de las estaciones durante la sesión de teleconsulta, ya que esto implicaría unas velocidades de megabits por segundo. En el ejemplo presentado anteriormente obtenemos un flujo de más de 50 Mbit/s. Con algoritmos como los de la familia MPEG se podría comprimir hasta unas velocidades de entre 2 y 5 Mbit/s, pero aún así es poco realista suponer que se va a disponer de enlaces de estas capacidades en todos los hospitales involucrados.

Por ello se impone el uso de la técnica *store and forward*, que consiste en hacer una transmisión previa, a velocidades menores, antes de operar sobre el examen. Habitualmente esta transferencia se lleva a cabo durante la noche, cuando hay menos tráfico y las tarifas son reducidas. Durante la teleconsulta las estaciones actúan sobre las copias locales de la secuencia, y la única información intercambiada son

las acciones que se ejecutan sobre el examen (incluir una anotación, iniciar la reproducción, hacer pausas, etc). Así se minimizan enormemente las necesidades de ancho de banda (el intercambio de comandos necesita entre 20 y 50 Kbit/s), si bien imposibilita que el sistema sea utilizado en urgencias.

2.4 Red de acceso

¿Qué tipo de red se necesita para desplegar el sistema en todo el territorio cubierto por el Sistema Público de Salud? Estos son los requisitos [4]:

- Se necesita una red con una penetración alta, para llegar a todos los centros sanitarios.
- El ancho de banda debe ser suficiente como para permitir una transmisión rápida de los estudios.
- Si se quiere que la teleconsulta sea realmente interactiva, es necesario disponer de capacidad para establecer canales de audio y vídeo que acompañen a los datos médicos.
- Es importante que el precio del despliegue sea asumible por la Administración sanitaria.

De entre las redes de acceso que se están utilizando hoy en día, sólo las basadas en el aprovechamiento del bucle de abonado telefónico son adecuadas, ya que nos ofrece una penetración de casi el 100% del territorio y ya está instalado (lo que permite minimizar el gasto de despliegue). Sin embargo, el problema es su escaso ancho de banda, lo que redundará en tiempos de transmisión largos y en mermas de la calidad de la videoconferencia durante la sesión de teleconsulta.

Tres son las tecnologías que usan el bucle de abonado: los modems telefónicos (serie V de la ITU-T), la Red Digital de Servicios Integrados (RDSI) de banda estrecha, y las redes ADSL (Asymmetrical Digital Subscriber Line). Sólo los dos últimos ofrecen un ancho de banda suficiente como para permitir una transmisión rápida, y entre ellos RDSI es una tecnología completamente desplegada y probada, característica importante si se quiere implantar un sistema fiable (tal como debe ser la telemedicina).

Por tanto, la mejor elección para las comunicaciones entre los hospitales de referencia y los centros primarios es RDSI, a una velocidad mínima de 128 Kbit/s y recomendable de 384 Kbit/s. Dado que la transmisión de los estudios podría bloquear la línea durante algunas horas, es recomendable que las transmisiones se realicen durante la noche, aprovechando también el menor coste de las llamadas. Esto exigirá que la aplicación de telemedicina prevea la programación de transmisiones y existan mecanismos automáticos de transferencia que se disparen cuando llegue la hora adecuada.

Respecto a las comunicaciones entre los hospitales de referencia, o entre los secundarios y éstos, lo más recomendable es el uso de redes del tipo MAN, que disponen de un gran ancho de banda. Esto permite transmisiones a alta velocidad de gran cantidad de estudios, así como teleconsultas plenamente interactivas.

2.5 Seguridad de la información

Cualquier aplicación moderna de telemedicina debe prever los aspectos relacionados con la seguridad de la información médica que se está transmitiendo entre hospitales. Concretamente, hay que proporcionar las siguientes características:

- *Confidencialidad*: La información (informes e historiales médicos) no debe ser accedida por personas no autorizadas. La confidencialidad de los datos médicos debe ser garantizada por motivos legales y de secreto profesional. Habitualmente la confidencialidad se garantiza mediante técnicas criptográficas basadas en sistemas de clave pública y privada.
- *Integridad*: La información no puede ser alterada maliciosamente durante la vida de la misma. La integridad se garantiza mediante funciones de *checksum* o bien mediante funciones de *hash*. De nuevo, hay que destacar la importancia de la integridad de los datos en el campo sanitario.
- *Autenticidad*: es la capacidad de garantizar que el autor de un documento es quien dice ser.
- *No repudio*: El documento debe poder, mediante la utilización de técnicas criptográficas, identificar inequívocamente a su autor. Tanto la autenticidad como el no repudio deben ser garantizados en los informes clínicos, cuyo contenido es responsabilidad exclusiva de los médicos autores.

2.6 Capacidad de medida y sincronización

La utilización de aplicaciones informáticas permite al médico disponer de herramientas de medida precisas. La medida de distancias y desplazamientos es de vital importancia en el diagnóstico cardiológico; una medida directa sobre la pantalla o sobre papel es sin duda una fuente de ineficacia e inexactitud. Parece razonable integrar en el entorno un sistema de medida preciso. Dicho sistema debe incorporar facilidades para tomar longitudes y ángulos, así como llevar a cabo las calibraciones necesarias. Asimismo, se valorará la funcionalidad de *zoom* y funciones de procesado de imagen, que permitirán observar una imagen o una secuencia con un elevado nivel de detalle, incrementando así la precisión.

Un atractivo que solo pueden ofrecer las aplicaciones multimedia es el de añadir anotaciones de texto, dibujos a mano alzada, o comentarios de voz directamente sobre las imágenes, así como la posibilidad de tener un puntero sincronizado en todas las estaciones que participan en la teleconsulta. Aparecen así conceptos como WYSIWIS (*What you*

see is what I see o "lo que ves es lo que veo"), también conocido como *sincronización de pantallas*, o lo que se conoce como *trabajo cooperativo* o CSCW (*Computer-Supported Cooperative Work*). Una buena aplicación de telemedicina debe incluir estas características.

2.7 Interacción con el usuario

Finalmente, debe cuidarse la manera en que el usuario accede a los servicios de la aplicación, haciendo especial hincapié en el interfaz gráfico, que debe ser amigable y atractivo. Recordemos que los usuarios no serán especialistas en ordenadores, sino expertos en medicina que usan una herramienta informática. Por ello se impone el uso de una plataforma basada en PC con un sistema operativo amigable y extendido, como Microsoft Windows.

3 El proyecto CARMEN

CARMEN nació como fruto del marco de colaboración interregional entre Catalunya y Lombardía [1, 6]. El propósito principal de esta colaboración es unir, mejorar e incrementar en una única aplicación las funcionalidades de algunas aplicaciones telemédicas precursoras: CARE y CAROLIN, desarrolladas por los equipos catalán y lombardo, respectivamente [3, 7]. Son cuatro las instituciones que colaboran en el desarrollo del proyecto: el instituto CEFRIEL de Milán, la Universitat Politècnica de Catalunya (UPC) y el Centre de Visió per Computador (CVC) como desarrolladores; y la empresa italiana Aethra, que juega el papel de socio tecnológico (aporta las tarjetas de videoconferencia y RDSI, y lidera el proceso de reingeniería y comercialización de la aplicación). Todo el proyecto se ha desarrollado en C++ con el entorno Visual Studio de Microsoft para Windows 9x

3.1 Funcionalidades de Carmen

A continuación se destacarán algunas de las funcionalidades más importantes de CARMEN:

Seguridad. La aplicación incorpora potentes herramientas de seguridad para mejorar la confidencialidad de los datos. Se ha escogido el algoritmo de seguridad Anigma [8], que ofrece la posibilidad de encriptar y firmar los documentos relacionados con el diagnóstico. El algoritmo proporciona las características de confidencialidad, integridad, autenticidad y no repudio. La identificación del médico se asegura mediante el uso de discos combinados con *passwords*.

Base de datos. La utilización del estándar de telemedicina DICOM [5], y el almacenamiento de larga duración en soporte CD son algunas de las nuevas funcionalidades que proporciona la aplicación, y que posibilitan la integración con el sistema de información y archivo del hospital. Se pueden exportar e importar ficheros en formato DICOM aptos para ser leídos por otros aparatos, o ser enviados mediante el módulo de correo electrónico integrado. También se pueden extraer copias en papel, a diferentes niveles de detalle, de la información contenida en los exámenes y en los informes de diagnóstico.

Red. CARMEN puede ser utilizada en redes de conmutación de paquetes que utilicen los protocolos TCP/IP (LAN, MAN, RDSI de banda estrecha, RDSI de banda ancha) y ha sido diseñada para funcionar independientemente de la tecnología de red, aunque el escenario típico de uso sea LAN para comunicaciones intrahospitalarias y MAN o RDSI para conexiones entre hospitales [4]. Para permitir la interoperabilidad entre diferentes redes, se ha adoptado el protocolo T.120 de la ITU, sobre el cual se han implementado canales de datos, de control, y de transmisión de ficheros.

En Lombardía, el gobierno regional ha optado por el uso de conexiones RDSI a 384 Kbit/s entre los hospitales secundarios y los de referencia, con transmisiones nocturnas programadas. En Catalunya se ha optado por una red MAN a 2 Mbit/s, lo que permite un envío muy rápido de los exámenes.

Compresión. CARMEN es independiente del formato de compresión de vídeo escogido, ya que puede trabajar con cualquiera de los *codecs* de Windows. La elección depende del hardware de adquisición utilizado. En las primeras instalaciones se están utilizando las tarjetas MJPEG de Aethra, pero la aplicación puede trabajar con otros *codecs*. Existe un compromiso entre la calidad de la imagen, el factor de compresión, la velocidad del proceso y el precio del hardware utilizado. En la actualidad se está evaluando el uso de codificadores MPEG-1 y 2.

Multilenguaje. Dada la dimensión europea del proyecto, todos los módulos que forman CARMEN han sido desarrollados con la capacidad de cambiar fácilmente el lenguaje del interfaz. Inicialmente se desarrollarán versiones en italiano, inglés, español y catalán, pero el código es fácilmente adaptable a otros idiomas. Dos instancias de la aplicación pueden interactuar aunque utilicen idiomas diferentes; un médico español puede conectarse con un cardiólogo italiano manteniendo cada uno de ellos el interfaz en su lengua propia.

Interfaz. Se han implementado técnicas de tratamiento de imágenes para la mejora de la información gestionada por la aplicación, así como la posibilidad de introducir anotaciones de tipo multimedia (dibujos, sonidos, texto). Todo ello manteniendo siempre la filosofía WYSIWIS para sincronizar las estaciones durante la teleconsulta.

Finalmente cabe destacar que al funcionar en plataformas del tipo PC-Windows se asegura un coste asequible para una implantación masiva y la facilidad de manejo de la aplicación, aspectos fundamentales para su aceptación entre la comunidad médica.

3.2 La arquitectura de Carmen

CARMEN se compone de diferentes módulos que actúan sobre los datos correspondientes a los pacientes. La unidad básica de la aplicación es el **examen**. A continuación se enumeran los elementos que componen un examen:

- Secuencia de vídeo: se trata del elemento más importante del examen. La secuencia está contenida en un fichero AVI (MJPEG o MPEG).
- Imágenes: ficheros *bitmap* (BMP) que contienen aquellos fotogramas de la secuencia que el usuario considere de especial relevancia.
- Anotaciones: ficheros que contienen las marcas gráficas (círculos, flechas, distancias, etc.), de texto y voz realizadas sobre las imágenes.
- Fichero de examen (EXM): contiene los datos necesarios para reconocer el examen; en concreto contiene la información del paciente, del estudio y del profesional que llevó a cabo la consulta.
- Informe (*report*): se trata del documento en formato RTF, debidamente cumplimentado y firmado electrónicamente, que refleja el diagnóstico acordado por los dos profesionales durante el proceso de teleconsulta.
- Fichero DICOM: fichero que exporta el examen de CARMEN al formato DICOM, para que éste pueda ser leído por todos los sistemas compatibles con el estándar.

El examen es, pues, el elemento central sobre el que se llevan a cabo las acciones de CARMEN. Un examen pasa por diferentes estados, desde que se adquiere la secuencia de vídeo, hasta que el informe de diagnóstico se firma electrónicamente, se cierra, y los ficheros correspondientes se almacenan en CD.

Los módulos que componen CARMEN son:

Base de datos. Los historiales de los pacientes constituyen la principal información que debe gestionar CARMEN. La base de datos, desarrollada en Microsoft Access, se organiza jerárquicamente en cuatro niveles según con el estándar DICOM: paciente, estudio (asociado a una de las dolencias del paciente), serie (correspondiente al examen) e imagen (unidades básicas de la secuencia de vídeo). Dichos niveles corresponden igualmente a la semántica del mundo real; de este modo, un paciente podrá tener asociado uno o más estudios, un estudio una o más series y una serie (secuencia) una o más imágenes**.**

Browser. El elemento central o "puerta de entrada" de la aplicación es el navegador (*browser*), que controla en todo momento el estado del examen e interacciona con el resto de módulos para llevar a cabo, a instancia del usuario, las operaciones principales que conciernen a dicho examen (introducción en la base de datos, edición, transmisión, teleconsulta, encriptación del informe, grabación en CD, correo electrónico, impresión, etc). El *browser* se diseñó para que se pareciera lo máximo posible al Explorador de Windows; en vez de navegar entre directorios, accede a la base de datos DICOM.

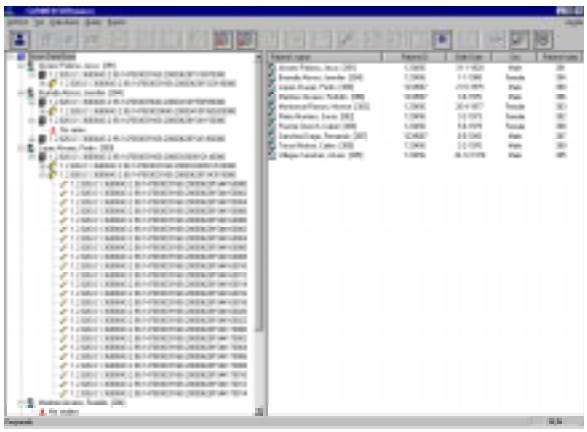


Figura 3: *Browser*. Ejemplo de una base de datos DICOM.

Dicomizer. Es el módulo que efectúa la conversión entre el formato de examen de CARMEN (.exm), con todos sus ficheros asociados, y el formato DICOM. No se ha implementado todo el estándar DICOM, sino sólo un subconjunto de sus funcionalidades (las directamente relacionadas con la cardiología).

Cliente. El cliente es la herramienta con la cual el cardiólogo podrá visualizar, anotar y manipular la secuencia de vídeo y las imágenes a efectos de llegar a un diagnóstico. También será el cliente el encargado de dar soporte a la teleconsulta entre médico de cabecera y especialista. Se distinguen pues dos modos de funcionamiento de cliente: modo local (*stand alone*) y modo teleconsulta (sesión interactiva en el que se necesitará el apoyo del módulo servidor).

El cliente dispone de dos vistas: el VCR virtual y la vista de anotaciones y galería. La primera de las vistas intenta reproducir el funcionamiento de un vídeo (*play, stop, pause, frame-by-frame*, etc).

Figura 4: Cliente. Ejemplo de visualización de una secuencia.

Servidor. En una sesión de teleconsulta colaboran dos médicos sentados frente a sendas máquinas

CARMEN. Para poder llevarla a cabo se necesita un módulo adicional: el servidor, encargado de gestionar y de comunicar a ambos clientes los eventos que se producen durante la teleconsulta.

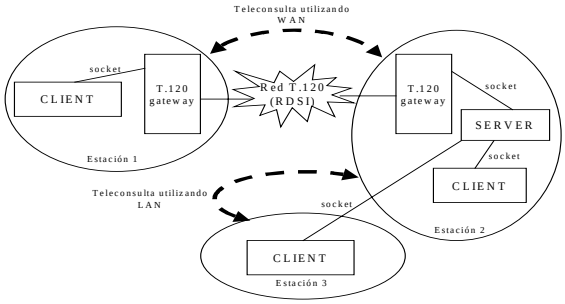


Figura 5: Teleconsulta de varias estaciones, sobre diferentes redes.

Durante la sesión de teleconsulta, todas las estaciones participantes pueden trabajar sobre la secuencia, ya sea mediante acciones relacionadas con el VCR virtual o con la vista de anotaciones. La filosofía de trabajo sigue la técnica maestro-esclavo: sólo una de las estaciones tiene el control, mientras que el resto reproduce los cambios realizados por la estación maestra. Es muy importante que se mantenga la sincronía; no puede permitirse que las estaciones muestren a sus usuarios informaciones distintas.

Scheduler También denominado "programador de transmisiones", es el módulo que concentra la inteligencia del sistema de comunicaciones. Su principal misión es coordinar a todos los módulos que intervienen en la transmisión de datos. Se ubica entre el *Gateway* T.120 y el interfaz de usuario (el *browser* y el cliente). El *Scheduler* es un módulo sin interfaz gráfica de usuario, y se ejecuta permanentemente, a modo de *daemon.*

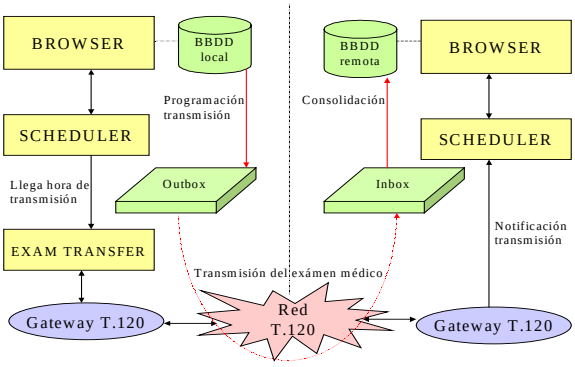


Figura 6: Esquema de una transmisión de un examen CARMEN.

La aportación del *Scheduler* puede ser dividida en dos grandes funciones. Por un lado, prepara y lanza las transmisiones de exámenes médicos entre máquinas CARMEN; por otro, coordina y establece las sesiones de teleconsulta entre dos estaciones. Hay que destacar que el *Scheduler* no es el encargado de llevar a cabo la transferencia de datos propiamente dicha (de eso se encargan los módulos *Gateway* y *ExamTransfer*), pero sí el responsable de preparar los parámetros y procesar los resultados de la misma. El módulo permite programar transferencias durante la noche, sin que el usuario tenga que estar presente, y que pueda responder a situaciones imprevistas como desconexiones, problemas de red, etc.

Gateway T.120 y Exam Transfer. Estos módulos quedan ocultos al usuario final. Son el subsistema de transmisión de datos y ficheros. Proporcionan canales lógicos T.120 transparentes sobre RDSI, LAN o MAN. De esta manera el *Scheduler* no se preocupa de la red subyacente. Estos módulos se han implementado con el SDK de Microsoft NetMeeting.

4 El despliegue de CARMEN

Es importante estimar el impacto que una aplicación de este tipo puede tener sobre el procedimiento cotidiano de diagnóstico en los hospitales. La última palabra la tendrán siempre los médicos que, en definitiva, son los que pueden sacar provecho de las funcionalidades ofrecidas por CARMEN.

La introducción de herramientas telemáticas en el proceso de diagnóstico aporta sin duda enormes ventajas, pero su utilidad se verá limitada hasta que se produzca un *efecto de inmersión* del usuario [9, 10]. Esto significa que el médico debe llegar a pensar únicamente en su trabajo de diagnóstico, sin tener que preocuparse por la faceta informática, que no debe distraer su atención. No se podrá decir que la aplicación ha sido un éxito hasta que ésta forme parte de la vida cotidiana del hospital.

Dado que CARMEN es todavía una aplicación en fase de desarrollo, no ha podido ser difundida para su evaluación por parte de los cardiólogos. Por lo tanto, la valoración se hará basándose en su predecesora CAROLIN, que sí se encuentra ya funcionando a pleno rendimiento en la red CARDNET italiana, compuesta por 8 hospitales de la región lombarda, unidos por conexiones RDSI a 384 Kbit/s [11]. En Catalunya se está desplegando una red similar que alcanza a 4 hospitales públicos, utilizando la infraestructura del proyecto *I2-Cat* (*Internet 2 a Catalunya*) [2], con enlaces de 2 Mbit/s.

Los resultados se pueden perfectamente extrapolar al caso de CARMEN, ya que se trata, en definitiva, de una versión avanzada de la aplicación. CAROLIN está formada, básicamente, por el cliente, el servidor, y un módulo sencillo de transferencia de ficheros.

4.1 Impacto de la aplicación

Los resultados estadísticos que se mostrarán a continuación son fruto de la experiencia piloto [12] entre dos hospitales italianos que utilizan CAROLIN: la Azienda Ospedaliera di Cremona (AOC, hospital secundario con departamento de hemodinámica) y el

[2] http://www.i2-cat.net

Spedali Civili de Brescia (SCB, hospital terciario con departamento de cirugía cardiovascular). Antes de la introducción de CAROLIN, el médico de AOC tenía que llevar físicamente la cinta de vídeo al especialista de SCB para consultar y discutir los casos dudosos.

El departamento de AOC es capaz de examinar una media de 15 pacientes semanales, la mitad de los cuales necesitará una segunda opinión. La frecuencia de estas consultas con el especialista es bisemanal, y por tanto se discuten 15 casos por sesión. Considerando una media de unos 20 minutos por caso, la sesión completa necesita no menos de 6 horas (un día de trabajo) más el tiempo asociado al desplazamiento (una hora en coche). La tabla que se presenta a continuación compara algunos índices de las actividades del departamento de hemodinámica de AOC antes y después de la introducción de CAROLIN.

	Antes	Después
Frecuencia de las consultas	Bisemanal	Semanal
Tiempo de edición para cada examen	Ninguno	3 minutos
Tiempo de consulta para cada caso	20 minutos	15 minutos
Casos discutidos en cada sesión de consulta	15	7
Duración de la sesión de consulta	6h + tiempo de viaje	1 hora
Coste de la sesión de consulta	565 $	Actividad normal
Tiempo medio de espera del resultado de un examen	4-10 días	3 días
Tiempo medio para operar a un paciente	1 mes	15-20 días

Figura 7: Algunos índices relevantes, antes y después de la introducción de CAROLIN.

La primera diferencia significativa radica en el incremento de la frecuencia de las consultas, debido al menor coste del proceso. En segundo lugar, cada consulta se reduce ahora sólo 15 minutos por caso, gracias a la ayuda prestada por las herramientas informáticas (medidas de distancias y ángulos, *zoom*, moviola virtual, etc.) proporcionadas por la aplicación. Con el mismo número de pacientes y porcentaje de casos dudosos, se pueden discutir ahora aproximadamente 7 casos por sesión de teleconsulta, en un tiempo de 1 hora o menos.

La utilización de CAROLIN exige dos actividades adicionales respecto al procedimiento tradicional de consulta: la edición previa de la secuencia de vídeo y la transmisión de la misma. La primera apenas requiere 3 minutos si el usuario está familiarizado con la herramienta y la segunda puede llevarse a cabo durante la noche. Por lo tanto, ninguna de las dos operaciones retrasa significativamente las consultas y, en definitiva, no se aumentan los tiempos de espera ni los costes asociados. Así pues, el uso de una aplicación de Telecardiología como CAROLIN o CARMEN puede tener un gran impacto sobre la rutina de diagnóstico en los hospitales. Los tiempos de diagnóstico se ven considerable reducidos, al igual que los costes de la consulta.

5. Conclusiones

Este artículo ha descrito el diseño y el desarrollo de CARMEN, un proyecto de telecardiología enmarcado en un contexto de proyección europea. Se ha conseguido desarrollar con éxito y poner en funcionamiento una potente herramienta *software* de telemedicina. La aplicación facilita la gestión, la transferencia y la seguridad de los historiales clínicos, y posibilita a los médicos un trabajo cooperativo en la tarea de diagnosticar enfermedades. El objetivo de acelerar el envío de secuencias angiográficas y, en última instancia, de evitar los traslados de pacientes entre la jerarquía de hospitales, a fin de obtener la opinión experta del especialista en hemodinámica, ha sido así ampliamente alcanzado.

Agradecimientos

Los autores quieren agradecer los esfuerzos de los desarrolladores que han participado de una manera u otra en el diseño de la aplicación, desde los pioneros que diseñaron MARC, CARE y CAROLIN hasta los cardiólogos que han colaborado en el desarrollo. Sin su ayuda no se podría haber realizado este trabajo.

Referencias

[1] http://www.teleregions.org, http://teleregions.gencat.es

[2] J. H. Sanders and R.L. Bashshur, *"Challenges to the Implementation of Telemedicine"*, Telemedicine Journal, vol.1, no. 2, 1995, pp. 115-123.

[3] G Valetto. *"CAROLIN – Integrated support to medical GroupWare over wide area networks"*, Proceedings of 3rd International Conference on Networking Entities, 1197, Ancona (Italy).

[4] D. Rincón, E. Frumento, R. Fogliardi, M.A. Viñas, *"Description of a Teleconsultation platform and its interaction with access networks"*. Proc. of the 5th Open European Summer School - EUNICE99 Barcelona, Spain, Septiembre 1999.

[5] *"The Fundamentals of DICOM"*, Kodak Digital Science division, http://www.kodak.com.

[6] E. Frumento, D. Rincón, M.A. Viñas and M. Fregonara, *"CARMEN: an international experience of telecardiology"*, Proceedings of ESEM'99, pp 279-280, Barcelona, 1999.

[7] J. Paradells, J. Casademont, S. Sallent y J. Borràs, *"MARC: A Teleradiology System"*, Multimedia 1994, Japón, 1994.

[8] http://eon.pmf.ukim.edu.mk/~kbajalc/algo/anigma.html

[9] M. Fregonara, E. Frumento, *"On-Field Evaluation of CAROLIN, an Italian Teleconsulting Cardiology Application: Early Results"*, Proc. Of Computers in Cardiology - CIC98, vol.25, pp. 217-220, Cleveland, Ohio (USA), September 1998.

[10] R. Fogliardi, E. Frumento, D. Rincón, M. A. Viñas, M. Fregonara, *"Telecardiology: results and perspectives of an operative experience"*, Journal of Telemedicine and Telecare, 6 (1), pp. 162-164, 2000.

[11] Borghi G., E. Brenna, R. Fogliardi, E. Frumento, L. Luzzi, V. Montericcio, *"Sanità e reti telematiche: il caso della rete CARDNET in Lombardia"*, Progettare per la Sanità, 54: 56-65, 1999.

[12] P. Totaro et al, *"On Line Interactive Dianosis in cardiac Surgery: a Preliminary Experience"*, Proc. of Interational Cardiovascular Surgery Simposium, Zurs am Ariberg (Austria), March 1999.

eDemocracia: Voto por Internet vs. Voto Electrónico

Iñaki Goirizelaia, Koldo Espinosa, José Luis Martín

Departamento de Electrónica y Telecomunicaciones
Área de Ingeniería Telemática
Escuela de Ingenieros de Bilbao
Universidad del País Vasco / Euskal Herriko Unibertsitatea

Email: jtpgoori@bi.ehu.es
Tfno: 34 94 601 42 10
Fax #: 34 94 601 42 59
Alda. Urquijo S/N
48013 Bilbao

Abstract: *A secret and secure ballot is at the core of every democracy. We all feel proud of being able to decide about the future of our countries by making appropriate use of our right to vote in an election. But, how do we improve the efficiency of voting processes? Democratic governments must have mechanisms for polling people that ensure security and privacy of an election process. This paper provides a general overview of some existing systems that fall in these two categories: traditional election systems based on electronic voting equipment and electronic voting systems for Internet. It also shows an electronic secure voting system based on automatic ballot reading that can be used to offer an efficient help to officials and party representatives during governmental elections*

1. Introducción

Votar libremente, sintiéndonos seguros de que nuestro voto es secreto y se cuenta, es sin duda la piedra angular de toda democracia. Todos nos sentimos orgullosos al comprobar que nuestro voto contribuye a decidir sobre el futuro de nuestros países, y convertimos el día en que se celebra una votación en un día especial, con una liturgia y forma de actuar probablemente muy distintas a la de los demás días del año.

Sin embargo, el proceso electoral no ha avanzado en los últimos años a la misma velocidad a la que ha avanzado la tecnología. Ejemplos como el vivido en las últimas elecciones presidenciales en USA han despertado el interés por nuevos tipos de votación, que contribuyan a agilizar y mejorar los actuales procesos electorales. Evidentemente, los gobiernos democráticos deben ofrecer nuevos mecanismos para consultar a la ciudadanía, que asegurando la seguridad y secreto del voto, ayuden a mejorar la eficiencia del proceso electoral. Sí pero, ¿cómo?.

La forma de votar no ha cambiado sustancialmente desde que la civilización griega nos enseñara a votar y lo hicieran por aclamación. Han sido diversos los sistemas empleados, los cuales todavía pueden verse en uso en cualquiera de las distintas formas que tenemos de expresar nuestra opinión en democracia. La utilización de piedritas creando montones para saber cual es la opción más votada, separar una multitud de personas en grupos según su opción, urnas, papeletas escritas, mano alzada,... son algunos de los métodos utilizados y que con algunas variaciones todavía hoy se utilizan con normalidad.

Los intentos por automatizar el proceso electoral no son recientes. Ya en 1869 Thomas A. Edison consiguió su primera patente gracias al diseño de una máquina que posibilitaba el recuento automático de los votos emitidos por los congresistas americanos. Desgraciadamente sus intentos de vender dicha máquina a los responsables del estado de Massachussets fueron baldíos. 132 años más tarde nos encontramos de nuevo ante el que creemos pudiera significar un importante paso que nos lleve a un nuevo proceso electoral más eficiente y de acuerdo con la nueva sociedad en la que vivimos.

Los avances en técnicas criptográficas y la popularización de Internet abren una nueva posibilidad: la realización de todo tipo de consultas al ciudadano y elecciones a través de la red [3]. Sin embargo, el abanico de posibilidades es amplio y son diversos los sistemas que, desde la tradicional urna, pasando por el recuento automatizado de votos, hasta el voto por Internet, pueden ser utilizados.

Sin embargo, sea cual sea el sistema de votación empleado, la "democracia electrónica" debe basarse en sistemas de votación que cumplan todas y cada una de las siguientes propiedades descritas por Cranor [2]:

Fiable

Un sistema de voto será fiable si cumple las siguientes características:

?? Ningún voto emitido por el votante puede ser modificado

?? Todo voto validado por el sistema de votación es utilizado en el recuento definitivo

?? Ningún voto no válido puede ser utilizado en el recuento definitivo

Invulnerable

Un sistema de votación es invulnerable si únicamente permite votar a las personas con derecho a voto, y además asegura que cada persona con derecho a voto vota solamente una vez.

Carácter privado

Un sistema de votación es privado cuando nadie, es decir, autoridades electorales, partidos políticos, representantes del gobierno o cualquier otra entidad pública o privada, puede relacionar el voto con la persona que lo emitió. Así mismo, no debe haber ninguna forma de que el votante pueda probar que votó una determinada opción. Esta segunda característica es de gran importancia para evitar la compra de votos o la extorsión.

Verificable

Un sistema de votación es verificable si cualquier persona puede comprobar que todos los votos emitidos se han contado correctamente.

Cómodo

Un sistema de votación es cómodo si posibilita el que los votantes depositen su voto de forma rápida en una única sesión, y sin necesidad de unas habilidades especiales o equipamiento sofisticado.

Por consiguiente, se trata de definir un nuevo sistema de votación que cumpla todas y cada una de las propiedades listadas. A continuación se comentan algunas de las opciones posibles:

2. No existe un equipo de esas características

Algunas personas afirman que un equipo como el definido no existe. No es posible tener un sistema de votación electrónico que ofrezca todas y cada una de las características citadas sin dar opción a que algún tipo de intervención humana las vulnere. Curiosamente nadie pone en duda la validez de los sistemas de votación tradicionales, usados en la mayoría de los países democráticos, donde necesariamente debemos fiarnos del personal del gobierno que está al cargo de los procesos electorales. Evidentemente existe una confianza mucho menor cuando se trata de utilizar ordenadores en procesos electorales.

3. Sistemas de elección tradicionales basados en el uso del voto electrónico

Los procesos electorales tradicionales se basan en la utilización de una serie de personas en las cuales confiamos, autoridades electorales y apoderados de partidos, que son las que velan por el normal desarrollo de la votación y realizan manualmente el recuento de los votos. La utilización del voto electrónico como ayuda en un sistema tradicional, posibilita automatizar el recuento final de los votos. En este caso, se deben utilizar papeletas electorales similares a las tradicionales, que se lean de forma automática por el equipo de voto electrónico. El sistema es, por tanto, compatible con el tradicional, con la única diferencia de que se posibilita el recuento automático. En caso de que exista alguna duda sobre el sistema de voto deberá ser siempre posible la comprobación manual de los resultados logrados.

4. Voto por Internet

Un sistema electoral basado en el voto por Internet implica el que cualquier votante desde cualquier lugar pueda depositar su voto y que éste sea recontado debidamente. Por tanto, cuando hablamos de voto por Internet nos referimos a un sistema que recupera vía Internet la papeleta del servidor electoral y la presenta al votante en la pantalla de su ordenador. El votante marca su opción y la devuelve también vía Internet al servidor electoral.

Gran parte de los sistemas de voto por Internet se basan en el esquema propuesto por Fujioka, Okamoto y Ohta [1]. Como ejemplo se pueden citar los sistemas Sensus [3] y EVOX [4] los cuales se basan en dicha propuesta. El protocolo puede describirse de la siguiente forma. El votante prepara su papeleta, la encripta con su clave secreta y la blinda. Seguidamente el votante firma la papeleta y la envía al sistema validador. Este sistema verifica que la firma pertenece a un votante registrado, el cual no ha votado todavía. Si el voto es válido, el validador firma dicho voto y lo devuelve al votante. El votante elimina la capa de encriptación blindada obteniéndose una papeleta encriptada firmada por el validador. A continuación el votante envía la papeleta encriptada y firmada al sistema de recuento de votos, el cual comprueba si la firma de la papeleta es adecuada. Si la papeleta es válida, el sistema de recuento almacena el voto para ser contado al finalizar la votación.

Riera et al [5] proponen un protocolo para ser usado en elecciones a gran escala. Su propuesta consiste en una organización jerárquica de centros de votación, basada en el uso de un Servicio de Directorio X.500. Este esquema ofrece mecanismos para resolver cuestiones como la distribución de votantes, coordinación de la apertura y cierre del proceso electoral, distribución de claves públicas, distribución de resultados parciales y finalmente cálculo de los resultados globales.

El informe denominado "Report on the Feasibility of Internet Voting" [7] presenta una propuesta para una arquitectura posible a utilizar en cualquier sistema que pretenda implementar el voto por Internet. La figura 1 muestra la arquitectura propuesta, donde a la izquierda se presentan las máquinas clientes para la votación, las cuales son las utilizadas por los votantes para emitir su voto. Cada cliente se conecta a través de un proveedor de servicios de Internet. A su vez, los diversos proveedores de servicios se conectarán al Centro de Datos Servidor de Votos (CDSV) mediante el proveedor elegido por estos. En todo caso la comunicación de las papeletas entre las máquinas cliente para la votación y los servidores de voto se realiza a través de Internet.

El objetivo del CDSV es principalmente las papeletas electrónicas encriptadas que los votantes envían a través de Internet, almacenarlas de modo seguro, enviar a los votantes de forma inmediata reconocimiento de que su papeleta ha sido aceptada y transmitir las papeletas al Sistema de Recuento, donde una vez verificada la legitimidad de la papeleta, se desencripta sin poder asociar el voto con la identidad del votante, quedando dispuesta para el recuento final.

Es importante destacar las diversas iniciativas de interés que han tenido lugar recientemente. Por ejemplo, la compañía Election.com (http://www.votation.com) anunció la nueva versión de su producto que posibilita una votación por Internet a nivel nacional para más de 200 millones de votantes en un periodo de 15 horas (4.000 votos seguros por segundo). Esta aplicación ha sido utilizada recientemente en Brest (Francia) en el referéndum para decidir si el mandato presidencial debía o no reducirse de 7 a 5 años. También en las elecciones primarias presidenciales en el estado de Arizona el partido democrático puso en marcha el voto por Internet.

Entre las compañías que ofrecen servicios y productos en el campo del voto por Internet se pueden citar Votelink (http://www.votelink.com), Hart Information Services (http://www.worldwideelection.com), LDE (http://www.e-lection.com), SAFEVOTE (http://www.safevote.com) ....

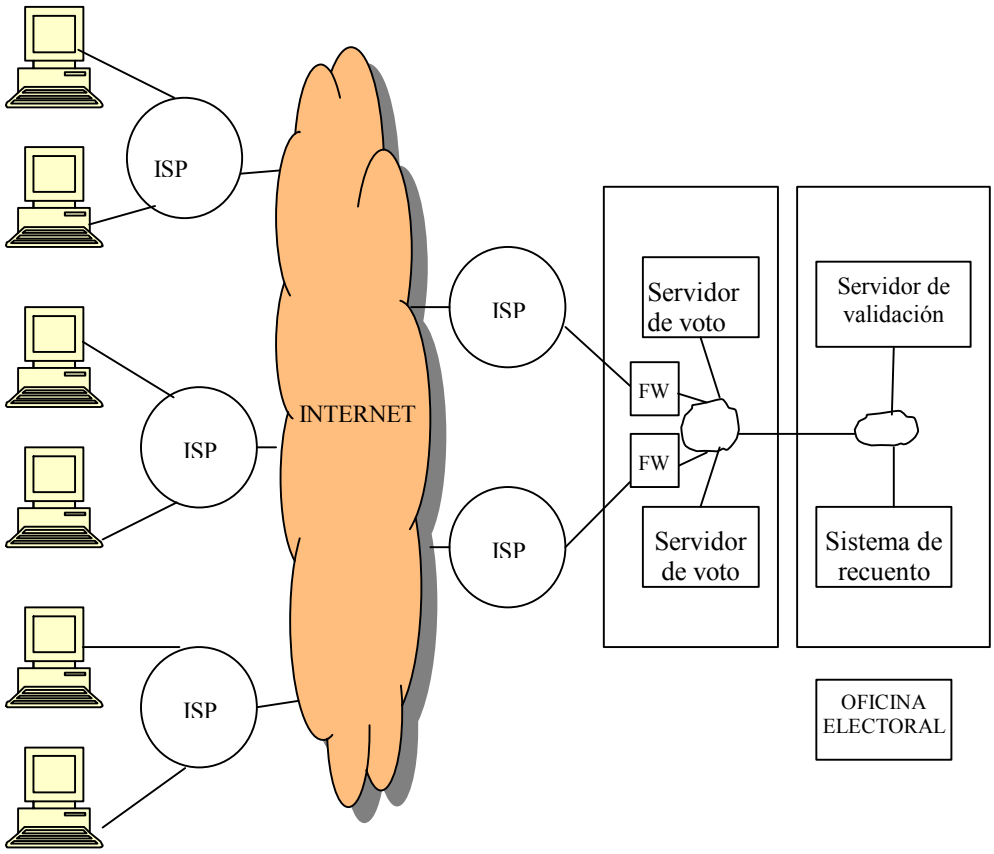


Figura 1: Arquitectura para voto por Internet

5. Sistemas Electrónicos de Votación

Los Sistemas Electrónicos de Votación se utilizan fundamentalmente con el objetivo de mejorar la eficiencia de los procesos electorales. Se presentan a continuación algunos de los equipos electrónicos de votación empleados en procesos electorales.

Optech III

Es un sistema de reconocimiento óptico de marcas (OMR) utilizado para el recuento de votos en la propia mesa electoral. Fue desarrollado por Business Records Corporation, aunque actualmente tienen derechos de fabricación y venta, tanto Election Systems & Software (ES&S) (http://www.essvote.com), como Sequoia Pacific (http://sequoiavote.com).

El sistema Optech III (http://www.ci.detroit.mi.us/electcomm) tiene el tamaño de un maletín, y tiene tres ranuras de salida. Determina por cuál de las ranuras extraer el voto, según su tipo. Por ejemplo, los votos con las marcas correctas los extrae por una ranura, y los rechazados, los extrae por otra diferente. En caso de que se estropee la máquina de recuento automático, se abriría una tercera ranura para que la extracción de los votos permitiese su almacenamiento temporal en una zona separada.

La máquina Optech III (ver figura 2) se coloca sobre una urna especial con tres entradas diferentes, de forma que quedan alineadas las ranuras de salida del sistema de recuento automático con las ranuras de entrada de la urna. Según la máquina procesa los votos, los envía automáticamente al compartimiento adecuado, dependiendo de si el voto se lee correctamente, de si el voto tiene un candidato escrito, o de que sea rechazado por algún motivo.

Para la elección de los candidatos, se rellena una papeleta en la cual se deben completar unas flechas. La cabeza y la cola de la flecha vienen impresas en la papeleta, junto al nombre de cada uno de los candidatos. El votante escoge a su candidato uniendo con una marca la cabeza y la cola de la flecha que apunta al nombre de su candidato.

Figura 2: Optech III junto con la urna dividida en compartimientos

Los primeros modelos de Optech III utilizaban sensores infrarrojos para detectar las marcas hechas por el votante, y para hacer las marcas se necesitaban bolígrafos especiales con tinta que absorbiera la luz infrarroja. Actualmente se utilizan sensores rojos, por lo que se puede utilizar cualquier tipo de bolígrafo o lápiz que no sea de tinta roja.

Optech IV

El sistema Optech IV es muy similar al Optech III, con la particularidad de que está pensado para un recuento automático en una oficina central de recuento, en vez de en el propio colegio electoral en el que el votante deposita su papeleta, y tiene una bandeja de entrada con lo que se facilita la entrada automática. Este sistema de recuento automático también es comercializado por ES&S.

Este sistema utiliza las mismas papeletas que el sistema Optech III, por lo que sirve para hacer recuentos de papeletas que no han podido ser leídas en el colegio electoral porque el Optech III haya dejado de funcionar.

AccuVote Optical Scan System

Sistema de reconocimiento óptico de marcas (OMR) comercializado por la empresa Global Election System, Inc. (http://www.gesn.com). En este sistema, el votante marca en una papeleta especial que se mete en la máquina de recuento automático. Los totales se pueden enviar a un servidor central.

AccuVote-TS (Touch-Screen)

Sistema de Registro Electrónico Directo (DRE) de pantalla táctil de la empresa Global Election System, Inc. Se utiliza tecnología de tarjetas inteligentes (smart-cards), que pueden almacenar información de los colegios electorales y datos de registro de los votantes. Se utilizan las tarjetas para validar a un votante y mostrarle sólo la papeleta electrónica adecuada. Cuando vota, se almacenan los datos en tres medios diferentes y puede enviarse a un servidor central.

eSlate Electronic System

El sistema DRE desarrollado por la empresa Hart Information Systems (http://www.hartis.com). En cada colegio electoral existe un controlador que se encarga de gestionar el proceso de votación de todo el colegio, desde el control del código de los votantes, hasta el envío de los resultados vía modem. Para que el votante elija el candidato se utilizan unos aparatos como el mostrado en la figura 3, que están conectados al controlador para indicarle la opción elegida por el votante.

Figura 3: eSlate 3000 de Hart Information Services

ELECTronic 1242

Este sistema es de Registro Electrónico Directo (DRE) y ha sido desarrollado por Guardian Voting Systems (http://www.controls-online.com/gvs). El ELECTronic 1242 utiliza seis memorias para asegurar la precisión y fiabilidad. Cada máquina realiza un autodiagnóstico después de cada voto.

El sistema tiene una gran pantalla táctil en la que entran todos los candidatos. Se marcan con una luz roja todos aquellos temas para los cuales el votante no ha emitido un voto, de forma que no se olvide de ninguno.

Sistema AVC

Sequoia Pacific Voting Equipment ha desarrollado dos sistemas DRE similares: el AVC Advantage y el AVC Edge.

AVC Advantage Electronic Voting System

Sistema fiable y flexible para adaptarse a los cambios tecnológicos y legales sin necesidad de cambiar el sistema. Se introdujo en 1988 y fue diseñado para mantener el secreto, eliminar el fraude, pérdida o daño de votos, elimina los votos rechazados y asegura que cada voto es tenido en cuenta.

AVC Edge Touch Screen Voting System

Es pequeño, del tamaño de un maletín, y portable. Se coloca sobre unas patas o sobre una mesa. Creado en 1998 para mantener la filosofía de secrecidad y seguridad del AVC Advantage pero con componentes físicos más pequeños (ver fig.4).

Figura 4: AVC Edge Touch Screen Voting System de Sequoia Pacific

Figura 5: Modelo Infinity de Microvote Corporation

Microvote DRE 464 Voting System

Sistema DRE diseñado por la empresa Microvote Corporation (http://www.microvote.com) y certificado por la NASED (National Association of State Election Directors) por lo que puede ser utilizado en elecciones estatales y presidenciales de los Estados Unidos. Utiliza estándares hardware para poder imprimir los resultados en papel o en una pantalla. Se utilizan cinco memorias redundantes, tres en memoria RAM y dos en cartuchos de memoria.

Microvote Infinity

Es el nuevo sistema DRE de Microvote Corporation. Tiene una pantalla de cristal líquido, no táctil, sino que tiene botones junto a la pantalla para elegir los candidatos (ver figura 5).

Como primer paso, se introduce una tarjeta que identifica de forma única al votante en la parte superior derecha. Posteriormente, se escoge el candidato pulsando el botón junto al nombre que aparece en la pantalla. Cuando el votante escoge el candidato puede emitir el voto pulsando el botón situado bajo la ranura de la tarjeta de identificación.

PATRIOT System

El sistema de votación PATRIOT "Touch-Screen" salió al mercado en 1994 comercializado por la empresa Unilect Corporation (http://www.unilect.com/patriot1.html). Es un sistema DRE con pantalla táctil de cristal líquido.

En cada colegio electoral existe un controlador para gestionar los diferentes dispositivos de votación del sistema. Este controlador dispone de una impresora para imprimir en papel los resultados tan pronto como se cierran las urnas. Además, existen varios dispositivos de votación en cada colegio electoral, en los cuales el votante escoge su candidato pulsando sobre el nombre que aparece en la pantalla táctil. Cada uno de estos dispositivos se comunica con el controlador de su colegio electoral para indicarle la opción escogida por el votante.

Los controladores de cada colegio electoral se conectan a una estación central, única en cada jurisdicción, mediante una red suficientemente potente como para recoger la información de cada colegio electoral.

6. Votación electrónica en España

Es importante señalar que en el Estado español, sólo la Comunidad Autónoma de Euskadi cuenta con Legislación Electoral en materia de voto electrónico, (Ley 15/1998, de 19 de Junio) aprobada por el Parlamento Vasco [7].

Los elementos del sistema de voto electrónico previstos en esta Ley son los siguientes: la tarjeta con banda magnética de votación, la urna electrónica, la pantalla de votar, la cabina electoral y el software o programa informático electoral.

Esta norma introdujo un sistema de voto electrónico que no ha encontrado aplicación, aun siendo un sistema totalmente respetuoso con los principios y características exigibles a todo proceso electoral. El sistema garantiza totalmente el secreto y la intimidad en el ejercicio del derecho de voto, permitiendo a su vez la realización del escrutinio con gran rapidez.

Se han realizado varias experiencias piloto en diversas comunidades del estado español. Por ejemplo, en Cataluña se utilizó el sistema de bandas magnéticas en dos colegios electorales para las Elecciones al Parlamento de Cataluña en 1.995. Este mismo sistema fue el utilizado en dos colegios en las Elecciones al Parlamento de Galicia en 1.997, y en las Elecciones Autonómicas de la Comunidad Valenciana en 1.999.

El Gobierno Central ha mostrado su interés por el voto por Internet, interés que queda patente en su intención de modificar la Ley Electoral en la presente legislatura para facilitar el voto por Internet.

7. Una propuesta para un sistema de votación electrónico

El Departamento de Electrónica y Telecomunicaciones de la Universidad del País Vasco ha trabajado en el desarrollo de un nuevo sistema de voto electrónico denominado demotek (http://www.demotek.net). En este proyecto propiciado por la Dirección de Procesos Electorales del Gobierno Vasco, además de nuestro departamento han participado las empresas Euskalnet, Ibermática, Hunolt, Ikusi y los centros de investigación Robotiker e Ikerlan. Las dificultades tanto de caracter técnico como sociocultural nos llevó a proponer un sistema cuya arquitectura se muestra en la figura 6.

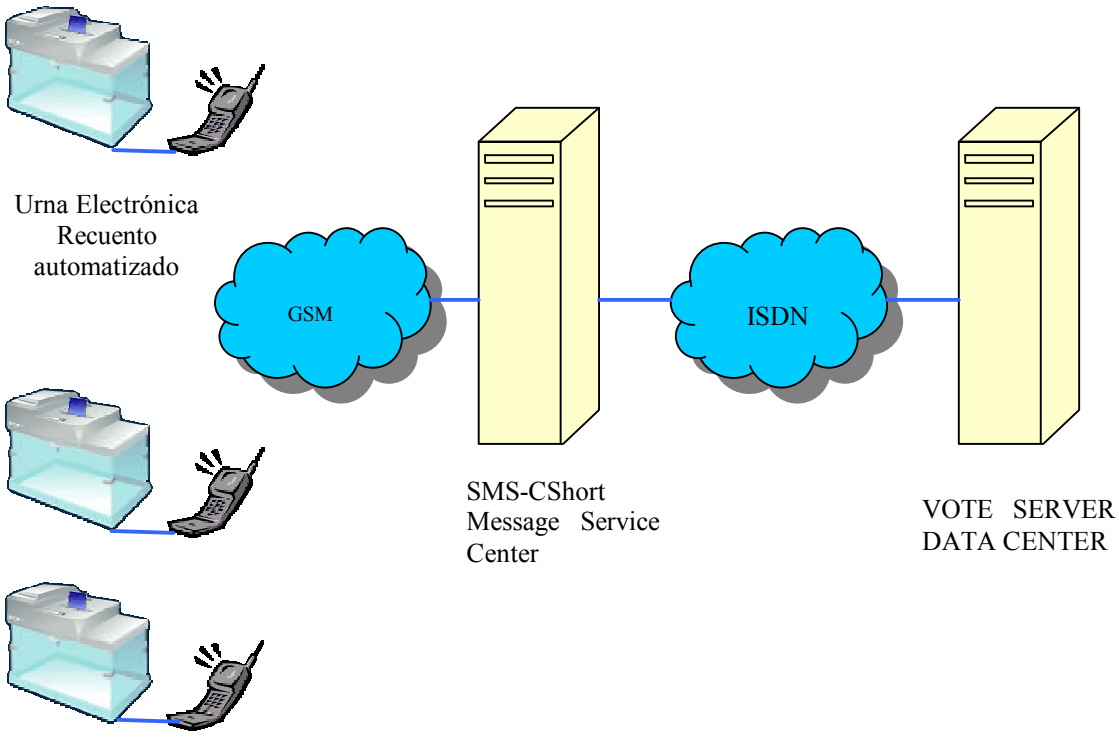


Figura 6: Arquitectura del sistema de votación electrónico propuesto

El Sistema de Votación Electrónica ha sido diseñado para cubrir los siguientes fines:

?? Votación muy similar al sistema convencional de urna y papeleta.
?? Recuento de votos instantáneo de cada urna tras su cierre.
?? Transmisión de datos automática hacia el centro de información a través de redes de telecomunicaciones disponibles, preferentemente mediante el sistema de mensajes cortos SMS de la telefonía móvil.
?? Proceso global de los votos, recibidos vía red de telecomunicaciones, desde las diferentes urnas situadas en los colegios electorales. Se procesa al ritmo de cierre de las urnas.

8. Lectura automática de la papeleta

La papeleta contiene la información a leer oculta, de forma que sólo es visible si se ilumina con luz ultravioleta. Una vez captada la imagen y segmentados todos los caracteres, la lectura de los mismos se realiza mediante la extracción de ciertas particularidades morfológicas del carácter, como concavidades, recintos, etc. [8]. Si dos o más caracteres tienen las mismas características, se hace necesaria la aplicación de máscaras flotantes que recorran el trazo de dichos caracteres para averiguar otras características que consigan diferenciarlos.

Una vez reconocida la papeleta, se determina si se corresponde con una candidatura válida, de tal forma que se pueda introducir la papeleta en la urna y sea incrementado el contador correspondiente.

9. Conclusiones

Todo sistema de votación electrónico o voto por Internet debe tener como objetivo cumplir todos y cada uno de los requisitos definidos en este artículo. Además su puesta en marcha debe facilitar al votante su participación en el proceso electoral contribuyendo de esta manera a incrementar los índices de participación electoral. En este artículo se presenta el dilema Voto por Internet vs. Voto Electrónico. A los autores de este artículo no nos cabe la menor duda de que el futuro de las consultas populares está en Internet. Sin embargo, sólo será posible si encontramos respuestas técnicas para todas y cada una de las siguientes preguntas:

?? ¿Cómo se asegura que sólo votan aquellas personas que tienen derecho a voto?
?? ¿Cómo se asegura que un votante sólo vota una vez?
?? ¿Cómo se asegura que el voto es privado y que la identidad del votante se mantiene en secreto?
?? ¿Cómo se asegura que el voto no se cambia una vez que el votante ha votado?
?? ¿Qué seguridad tiene el votante de que su voto se recuenta?
?? ¿Cómo se protege al votante contra la compra fraudulenta del voto?
?? ¿Cómo se protege al votante para que nadie le obligue a votar contra su voluntad?
?? ¿Cómo se protege el proceso de votación contra un ataque mediante algún virus o hacker informático?

Alguna de estas preguntas tiene hoy en día respuesta técnica mas o menos fiable. Sin embargo, creemos que no es posible dar respuesta a todas ellas de forma adecuada. Por ese motivo, pensamos que la puesta en marcha de elecciones de carácter gubernamental a través de Internet resulta hoy en día arriesgado. Nuestra propuesta se basa en la idea de compatibilizar el sistema actual de voto con el recuento automatizado. Estos son los primeros pasos, el futuro es sin duda la consulta universal por Internet. ¿Estaremos diseñando en este momento el fin de la democracia parlamentaria?

Referencias

[1] Fujioka, A., Okamoto, T., and Otha, K. A practical secret voting scheme for large-scale elections. Jennifer Seberry and Yuliang Zheng editors, Advances in Cryptology – AUSCRYPT '92, volume 718 of Lecture Notes in Computer Science, pages 244-251, Gold Coast, Queensland, Australia, 13-16 December 1992. Springer Verlag.

[2] Cranor, L.F., Cytron, R.K., Sensus: A security-Conscious Electronic Polling System for the Internet. Proceedings of the Hawaii International Conference on Systems Sciences, January 7-10, 1997 Wailea Hawaii USA.

[3] Dictson, D., Ray, D., The modern democratic revolution: An objective Survey of Internet – Based Elections. White paper. January 2000.

[4] Brandon DuRette, Multiple Administrators for Electronic Voting. Bachelors thesis. http://theory.lcs.mit.edu/~cis/theses/DuRette-bhachelors.pdf

[5] Riera, A., Borrel, J. and Rifá, J., 1997 Large Scale Elections by Coordinating Electoral Colleges, in IFIP SEC'97, Information Security in Research and Business (Chapman & Hall), 349-362

[6] California Internet Voting Task Force. "A report on the feasibility of Internet voting". January 2000. California Secretary of State Bill Jones

[7] Ley 5/1990, de 15 de Junio, de elecciones al Parlamento Vasco. Boletín Oficial del País Vasco, nº 134, de 6 de julio de 1990

[8] Bao-Chang P., Si-Chang W., Guang-Yi Y. "A Method of Recognizing Handprinted Characters". Computer Recognition and Human Production of Handwriting, *World Scientific Publ. Co.*, 37-60, (1989).

Planteamientos sobre Sistemas de Voto y Democracia Electrónica

Ana Gómez Oliva[1], Justo A. Carracedo Gallardo[1], Jesús Moreno Blázquez[1] y José David Carracedo Verde[2]
[1]Departamento de Ingeniería y Arquitecturas Telemáticas. Universidad Politécnica de Madrid
Ctra. Valencia km. 7. 28031 Madrid.
Teléfono: 913 367 820. Fax: 913 367 817. E-mail: {agomez,carracedo,jmoreno}@diatel.upm.es
[2]Departamento de Ciencia Política y de la Administración III. Universidad Complutense de Madrid
Campus de Somosaguas. 28223 Madrid
Teléfono: 913 942 664. Fax: 913 942 776. E-mail: jdcarracedo@proyectos.diatel.upm.es

Abstract*. This paper present a preliminary analysis of electronic voting schemes and the requirements of Electronic Democracy as a part of the work carried out by the authors in the VOTESCRIPT project (TIC2000-1630-C02). A summary of the most relevant experiences on this field are discussed and a basic classification of them is pointed out, according to different degrees on process computerization. As it is shown, most of them only take into account a technological perspective, just trying to imitate the conventional voting schemes. A citizen-base bottom-up perspective is proposed to analyze the implementation of electronic voting systems in order to avoid citizen rejection. The paper also hallmarks the new technical possibilities created to be applied to the development of citizen's right realm. Further than conventional voting schemes, the paper proposes the use of advanced security services to extend conceptualization of Electronic Democracy in which citizens have a key role on decision making processes.*

1. Votación y Democracia Electrónica

Estamos asistiendo a una explosión en la demanda de uso de servicios telemáticos para abordar situaciones y problemas que tradicionalmente han venido resolviéndose apoyándose en el intercambio de información sobre papel y en otras formas convencionales de comunicación. La cuestión del voto y de la Democracia Electrónica no podía escapar a esta tendencia y asistimos, así, a una cierta proliferación de sugerencias y propuestas, las más de las veces poco meditadas, sobre la conveniencia de la introducción de la Telemática en estas importantes y sensibles facetas de la vida de los ciudadanos.

Una primera simplificación de la que podemos ser víctimas es la de reducir el riquísimo y controvertido concepto de *democracia* a la categoría de un mecanismo de comunicación como es el caso del *voto*. Puestos a estrechar y empobrecer, se puede reducir también la noción de voto, asimilándolo solamente con los procesos mediante los cuales los ciudadanos eligen entre dos o más alternativas previamente configuradas. El diccionario de la Real Academia, pese a su proverbial laconismo, expande mucho más esta idea, recogiendo entre otras muchas acepciones la de que votar es también emitir parecer o dictamen razonado en una reunión o cuerpo deliberante.

Quizás sea por ese ánimo simplificador por lo que, con frecuencia, cuando se trata de abordar mediante procedimientos telemáticos el asunto del **voto electrónico,** sólo se propone una simple emulación o reproducción de los esquemas convencionales de votación que emplean urnas y papeletas. En estos casos, el reto, nada baladí, consiste en diseñar protocolos que incorporen mecanismos de seguridad robustos para seguir proporcionando las garantías que actualmente tienen los votantes; entre ellas las de que su voto ha sido adecuadamente tenido en cuenta y que no pueda relacionarse su nombre con la opción que eligió.

Es totalmente cierto que la incorporación de servicios de seguridad en las redes telemáticas ha de servir, al menos, para garantizar que los derechos y salvaguardas actualmente reconocidos en las comunicaciones convencionales sean respetados también en la proyección y plasmación que éstos tienen en las Comunicaciones Mediante Computadores.

Pero, siendo esto necesario, no es suficiente: mediante una adecuada utilización de las posibilidades que ofrecen las redes telemáticas y los servicios avanzados de seguridad que sobre ellas pueden establecerse, los ciudadanos podrían disponer de configuraciones y escenarios de comunicación que les permitiesen alcanzar niveles de participación y decisión como jamás antes habían sido vislumbrados (quizás a causa de, o con la excusa de, su difícil implantación en entornos constituidos por comunidades numerosas y complejas).

Otro elemento de capital importancia, que es necesario tener en cuenta a la hora de proponer la implantación de aplicaciones telemáticas de tanta repercusión, consiste en estudiar las posibles repercusiones sociales, políticas y legales que ello conlleva.

La problemática sociopolítica generada de forma directa por la implantación masiva de servicios telemáticos es lo que denominamos **Estratificación Digital** [1]. Se centra en el estudio de los discursos y prácticas asociadas con las desigualdades y diferencias en el acceso a computadores, infraestructura de entrada a la red y adquisición de conocimientos, que se dan entre las distintas clases sociales, dependiendo también de situaciones como etnia, género, nivel educativo, etc. En inglés, este campo de estudios es conocido como *Digital Divide*. Este término es ya centro de una fuerte polémica a causa de su falta de precisión: es vago y no abarca la complejidad del problema. En español, ha empezado a traducirse como "brecha digital", denominación que mantiene las limitaciones y carencias del termino inglés. A nuestro juicio, el término *estratificación* aquí propuesto refleja más claramente la multiplicidad de factores que implica y su jerarquización social.

Por todo ello, los trabajos que se plantean en la presente ponencia están enfocados hacia la perspectiva de abordar el desarrollo de esquemas y arquitecturas telemáticas que faciliten a los ciudadanos los procesos de *votación* y de *participación* en la gestión política de ámbitos y recursos que les son comunes, apoyándose en modelos de Democracia Electrónica, lo que conlleva establecer servicios a los que todos tengan pleno acceso y con las mismas oportunidades.

2. Algunas experiencias previas

En los últimos años existe un creciente interés, principalmente de los gobiernos, en emplear los sistemas de voto electrónico en sus consultas a la población, con la idea de favorecer la participación de un mayor número de ciudadanos en las decisiones que les atañen, reducir los costes ligados a cualquier proceso electoral y minimizar los posibles fraudes que pudieran producirse. En esta sección se examinarán aquellas experiencias sobre voto electrónico consideradas más relevantes y sobre las que existe mayor documentación. Sin embargo, el estudio aquí presentado no pretende ser exhaustivo, ya que constituye el resumen de un documento más completo (donde se realiza una critica sistematizada de cada una de las propuestas) que este grupo de investigación está elaborando, como parte de las tareas del proyecto VOTESCRIPT, que más adelante se comenta.

Estados Unidos ha sido, sin duda, el país pionero en el desarrollo de sistemas de voto electrónico. El uso de los ordenadores en los procesos electorales se remonta a 1964 [2], cuando cinco estados de EEUU hicieron uso de ellos para votar. Su presencia desde entonces ha ido en aumento, calculándose que en elecciones presidenciales de noviembre de 2000, el 69% de los votantes lo hizo por vía electrónica, utilizando diversos y anticuados mecanismos como la tarjeta perforada, el voto óptico y la máquina electrónica de registro automático.

Sin embargo, diversos factores han cuestionado en numerosas ocasiones la validez de estos sistemas: la falta de control administrativo, la confianza ciega que se deposita en los expertos que supervisan los procesos en lugar de que sean los representantes o autoridades electorales los que lo supervisen, los fallos detectados en la programación de los sistemas de votación electrónica, la falta de mecanismos de transparencia, etc.

Existen casos destacables de fallos en los sistemas de votación, siendo el más reciente y clamoroso de todos ellos el que tuvo lugar en las pasadas elecciones presidenciales en el estado de Florida, donde la falta de normativa y control propició que muchos votantes que emplearon el método de tarjeta perforada no pudieran saber con certeza qué opción era la que habían marcado.

A través de **Internet** se han hecho también varios experimentos. El Partido Reformista (1996) y el Partido Democrático de Arizona (2000) ofrecieron la opción de voto por Internet en sus primarias presidenciales, aunque sin poder garantizar el anonimato de los votantes. Esta última elección gubernamental ha sido la única de carácter vinculante realizada a gran escala.

En el estado de California la Secretaría de Estado convocó a la *Internet Voting Task Force* para estudiar la posibilidad de emplear Internet para llevar a cabo las elecciones en California. Se reunieron expertos en el campo de seguridad, legislación y participación ciudadana y elaboraron un informe, publicado en enero de este año [3]. Este informe recoge los requisitos de seguridad exigibles al nuevo sistema de votación y pone de relieve la necesidad de avanzar con cautela en el proceso de introducción del nuevo sistema de votación, ya que la posibilidad de amenazas o pirateo del sistema pondría en peligro el esfuerzo realizado. Sin embargo, afirma que, a pesar de los retos que supone el desarrollo del nuevo sistema, es técnicamente posible utilizar Internet para desarrollar un método de votación, al menos tan seguro como los sistemas actuales. A este respecto, esta Secretaría encargó a la empresa Safevote la preparación y realización de una prueba de un sistema de votación electrónica a través de Internet [4], llevada a cabo en el condado de **Contra Costa**, California, a primeros de noviembre de 2000.

Entre las experiencias no ligadas a elecciones gubernamentales se pueden citar las elecciones a gran escala realizadas sobre Internet en 1999 para elegir a la Junta Directiva de ISOC [5] y la que tuvo lugar al final del año 2000 para elegir a los miembros de la Junta Directiva de la ICANN (Internet Corporation Assigned Names and Numbers). En estas elecciones los votantes habían

recibido por correo ordinario un número de control, emitido anónimamente, de forma que no se podía enlazar el voto con el votante.

También hay que destacar las experiencias llevadas a cabo en **Brasil**. Este país aprobó en octubre de 1995 la Ley Electoral que marca las directrices del voto electrónico con la intención de eliminar el fraude electoral y reducir el tiempo de escrutinio.

El nuevo sistema de votación, basado en urnas electrónicas, se probó en la votación para alcaldes y concejales realizadas en octubre y noviembre del año 1996 en 50 ciudades de Brasil. En 1998 la modalidad del voto electrónico se extendió a 520 ciudades y en el año 2000 se puso en marcha el voto electrónico total que abarca desde la identificación de los electores hasta la publicación del resultado final. En las últimas elecciones celebradas en octubre de 2000 han votado por este sistema 109 millones de electores.

Este proceso de votación se lleva a cabo a través de una especie de cajero automático, dotado de un monitor, en el que van apareciendo los candidatos y donde los votantes pueden realizar su selección oprimiendo un botón. Al finalizar la jornada electoral, se bloquea la urna mediante una clave y automáticamente se imprime una copia de los resultados, a la vez que se obtiene un disquete que se lleva de inmediato a un Centro de Recuento para su cómputo.

Argentina también es uno de los países que está realizando experiencias sobre voto electrónico. En junio de 1999 estableció un convenio de colaboración con Brasil para trabajar en pro de la modernización de las estructuras de los respectivos estados, especialmente en lo concerniente a los sistemas electorales. En esta línea, Brasil ofreció a Argentina el préstamo de las urnas electorales con el objeto de pudiera realizar pruebas piloto en unas elecciones, sin coste alguno. En octubre de 1999 se realizó un simulacro de votación electrónica en varias localidades de Buenos Aires y en la Ciudad de Mendoza, para los que se utilizó un prototipo ideado en Argentina y las urnas desarrolladas por el Tribunal Superior Electoral de Brasil. La prueba fue de carácter voluntario y se hacía después de que el votante hubiera emitido su voto real en la mesa que le correspondiera.

Venezuela también ha incluido en el Reglamento General Electoral las instrucciones para que el proceso de votación, escrutinio y publicación de resultados del proceso de votación se realicen de manera automática. A diferencia del caso de Brasil este Reglamento no especifica el funcionamiento de ninguna máquina de voto en particular.

En las pasadas Elecciones Municipales de 2000 se confió a una empresa española la automatización del proceso de votación. Con este sistema, el elector emite el voto en la urna electrónica y automáticamente se acumula para su recuento y difusión sin intervención humana. Este proceso tiene como característica singular que es auditable por empresas y organizaciones externas al proceso electoral. Sin embargo, las primeras implantaciones de voto electrónico en los procesos electorales venezolanos no han sido muy afortunadas y han estado plagadas de problemas, básicamente motivados por la desconfianza hacia los resultados obtenidos.

En Europa se han realizado también varias experiencias. En **Bélgica,** se iniciaron en 1991, con una prueba piloto en el cantón de Verlaine. El método empleado es el de tarjeta con banda magnética que es entregada a cada elector en el momento de su identificación. Posteriormente, éste graba su opción de voto, utilizando para ello una cabina electoral que dispone de una pantalla, en la que se presentan las distintas opciones, y un lápiz óptico con el que se realiza su selección. Después, acude a la Mesa Electoral donde se introduce su voto en la urna. Como resultado de las pruebas realizadas se ha ido sustituyendo el sistema tradicional de voto mediante papeleta por el de tarjeta magnética. En las últimas elecciones municipales celebradas el pasado 8 de octubre de 2000 el sistema fue usado por el 44% de los electores, no estando todavía extendida su aplicación a todos los electores debido al coste que supone la implantación de este sistema.

En **Holanda** en marzo de 1995 se empleó el sistema del voto electrónico en las Elecciones Municipales de Ede y de Helmand y luego, en noviembre de 1995, en las Elecciones Sindicales en Philips Nijmegen. Estos ensayos realizados en Holanda han sido positivos. La Comisión constitutiva dentro del Consejo Electoral para analizar el sistema de voto electrónico ha emitido un informe favorable sobre el mismo. En la actualidad están modificando la Ley que recoge el proceso electoral para incorporar el *procedimiento del voto electrónico con tarjeta de banda magnética*, que sustituirá al tablero electrónico empleado hasta ahora.

En **Francia** también se han realizado dos ensayos: en junio de 1994 en las Elecciones Europeas al Parlamento de Estrasburgo con 4000 electores y en mayo de 1995 en las Elecciones Presidenciales en Issy-Les Moulineaux. Asimismo, en **Noruega** y **Dinamarca** también se han realizado diferentes pruebas en el periodo 1992-1995.

En **Nueva Zelanda** se ha creado la Fundación de Democracia Electrónica, una autoridad independiente con la finalidad de promocionar el uso de esquemas de voto electrónico. La primera prueba de voto electrónico a nivel nacional tuvo lugar en 1998, con el objetivo de mostrar el

potencial de las elecciones electrónicas mediante la participación ciudadana en temas democráticos.

Japón también ha realizado una prueba piloto de voto electrónico en el municipio de Kawaguchi con un censo electoral de más de 300.000 electores, distribuidos en 78 colegios electorales, 11 de los cuales participaron en la prueba de voto electrónico. El sistema empleado fue el de tarjeta con banda magnética y las opciones se seleccionaban en una pantalla táctil. El resultado de este experimento fue considerado un éxito, con un nivel de aceptación muy elevado por parte de los votantes.

India también desea incorporar el voto electrónico a sus procesos electorales para facilitar el escrutinio de los votos. El método seleccionado es el del tablero electrónico, en el que aparece la lista de los candidatos con un interruptor asociado a cada uno y donde el votante selecciona con este interruptor su candidato preferido. Sin embargo, en este país existen carencias estructurales que cuestionan la implantación de estos métodos a corto plazo (hasta 1998 no se informatizó el censo de votantes).

En **España** se han desarrollado tres experiencias muy restringidas en cuanto a sus objetivos y ámbito de aplicación: en las Elecciones al Parlamento de Cataluña en 1995, en las Elecciones Autonómicas Gallegas en 1997 y al Parlamento Vasco en 1998.

A nivel legislativo, sólo **Euskadi** [6] dispone de una ley propia (15/1998 de 19 de junio), donde se modifica la ley de elecciones al Parlamento Vasco, para regular el *Procedimiento de la votación electrónica*. En esta ley se indican los elementos que componen el sistema de voto: tarjeta con banda magnética, la urna electrónica, la pantalla de votar, la cabina electoral y el software electoral. Este software comprende los programas para permitir la apertura y cierre de la urna, la votación con tarjetas con banda magnética validadas por la Mesa, el control del número de tarjetas con banda magnética registradas en la urna, el escrutinio y la transmisión de los resultados electorales de la Mesa al ordenador central (esto último realizado exclusivamente con fines informativos).

El mecanismo de voto es muy similar al que desde hace varios años se viene utilizando en Bélgica, con ligeras variaciones en el proceso de identificación del votante y selección de los candidatos. Cabe destacar, que en ambos procesos una vez finalizada la votación, se precinta la urna y automáticamente se obtienen los resultados totalizados, que son reflejados en el acta correspondiente (en papel).

Estaba previsto emplear esta máquina en las elecciones al Parlamento Vasco en el año 2002, sin embargo el hecho de que se haya anticipado la convocatoria de estas elecciones a mayo de este año puede haber sido la causa de que este mecanismo no haya sido incluido en la normativa que regulará este proceso electoral.

También a **nivel nacional** existe un creciente interés en favorecer la votación electrónica. El pasado 6 de marzo se aprobó en el Pleno del Senado la creación de una Ponencia formada por la Comisión de la Sociedad de la Información y del Conocimiento y la Comisión Constitucional para estudiar la implantación de los sistemas electrónicos de ejercicio del derecho a voto y recuento, así como la reforma de referente al Régimen Electoral General (5/1985), la Ley Orgánica sobre regulación de las distintas modalidades de referéndum (2/1980), la ley Orgánica sobre la iniciativa legislativa popular (3/1984) y de cuantas otras sean necesarias al respecto.

Aunque la mayor parte de las experiencias desarrolladas se centran en el aspecto de la votación electrónica, también existen otras menos conocidas que abordan el voto electrónico como un concepto más amplio: democracia electrónica. El objetivo de estos ensayos ha sido el de potenciar una democracia local interactiva, que permita a los ciudadanos expresar sus puntos de vista y preferencias como un medio para mejorar el interés y responsabilidad en las instituciones políticas.

Estos experimentos tienen en común que los participantes los ven como un medio de revivir y vigorizar la política democrática, que por una serie de razones han perdido su atractivo y dinamismo; tienen un carácter local o regional y se basan en infraestructuras similares.

Dentro de estas experiencias podemos agrupar aquéllas que combinan un número diferentes de funciones cívicas y de comunicación, tales como deliberaciones, difusión de información pública y, en menor media, soporte a los grupos de "gente corriente". En este apartado se encuentran los proyectos Amsterdam's Digital City (iniciado en 1994), el IperBolE (*Internet per Bologna and Emilia)* (1995) y Santa Mónica Public Electronic Network (1989).

Otras experiencias tienen como objetivo prestar soporte a algún servicio concreto al que proporcionan infraestructura de red para el intercambio de información y el debate. Se puede citar a la iniciativa de la Ciudad de Información de Manchester, creada para promocionar la diseminación de la información relacionada con la regeneración económica y el Neighbourhoods Online**,** proyecto de iniciativa no gubernamental iniciado en 1995, que asiste a los grupos de ciudadanos que trabajan en mejorar las condiciones en comunidades y vecindades.

Por último, cabe citar el proyecto griego Network Pericles, iniciado en 1992 y concebido como un

instrumento de debate y acción política, al estilo de la versión ateniense de democracia, esto es, un proceso activo que implica a los ciudadanos en su autogobierno.

3. Escenarios de votación

A la vista de las experiencias previas enumeradas en el apartado anterior, podemos identificar varios escenarios de votación distintos. Estos escenarios se pueden clasificar en varios niveles dependiendo del grado de automatización del proceso.

En el nivel de partida de esta clasificación está lo que podemos denominar el escenario "clásico" de votación. En este escenario se englobarían tanto las votaciones mediante papeletas, como aquéllas que se sirven de tarjetas perforadas o de lectores ópticos. No podemos considerarlo como un sistema de voto electrónico propiamente dicho, pero hasta ahora, ha sido un referente para los distintos escenarios electrónicos que se han propuesto, ya que, normalmente, lo que se ha intentado ha sido sustituir alguno de sus procesos manuales por un proceso automatizado empleando para ello algún tipo de dispositivo electrónico.

En un segundo nivel se encontrarían los escenarios de votación que, como decíamos en el párrafo anterior, basándose en la forma de operar del método clásico, sustituyen alguno de sus elementos físicos y procedimientos manuales por algún tipo de sistema o de proceso electrónico.

Entre estos posibles escenarios tenemos aquellos que utilizan alguno o varios de los siguientes elementos: tarjetas magnéticas (para autenticar al votante o incluso para emitir el voto), urna electrónica (para la recepción y recuento de votos), pantalla (tablero) de votación (para seleccionar la opción de voto elegida), cabina electrónica (para garantizar la privacidad), software de distintos tipos (para el proceso de escrutinio).

En todos estos escenarios, los procesos a automatizar son los que se realizan comúnmente en el colegio electoral. Estos procesos podemos sintetizarlos en tres: El primero es el de la autenticación del votante, el segundo el de la votación propiamente dicho y el tercero, el que abarca todo lo relativo a la gestión y procesado del contenido de la urna electoral. Todos los componentes electrónicos utilizados en estos escenarios, tratan de automatizar alguno de estos procesos.

Un tercer nivel, y el más interesante desde nuestro punto de vista, sería el de los escenarios que hacen uso de redes telemáticas. Aquí podríamos distinguir dos grupos: Aquellos que utilizan las redes telemáticas (públicas o privadas) para la interconexión de los distintos colegios electorales, o bien los que proponen la votación desde casa (normalmente a través de Internet).

En los escenarios del primer grupo, el elector tiene que desplazarse hasta el colegio electoral (o centro equivalente de votación) para emitir su voto. Una vez allí, puede encontrarse con cualquier escenario de los que hemos considerado de segundo nivel. El uso de redes telemáticas para la interconexión de los colegios electorales y el organismo encargado de la supervisión final (con un papel equivalente al que en España desempeña la Junta Electoral Central) permite una rápida recolección de los datos y publicación de los resultados.

El segundo grupo, votación desde casa a través de Internet, es el más atractivo, desde un punto de vista tecnológico, debido a los retos técnicos y de seguridad que plantea. Pero, a su vez, desde un punto de vista sociológico, plantea interrogantes ya que no todo el mundo tiene las mismas oportunidades de acceso. Esta es una consideración ligada a garantizar el Sufragio Universal.

La idea subyacente en este escenario es la de que cualquier votante, que disponga de acceso a Internet, pueda emitir su voto sin necesidad de desplazarse al colegio electoral. Sin embargo, parece una exigencia política clara (aunque en algunos países y sistemas ni siquiera se plantee), que habría que seguir proporcionando alguno de los otros escenarios para aquellas personas que no dispusieran de acceso a Internet o no quisieran hacer uso de él, pese a tenerlo.

Este escenario posibilitaría también la implantación de otros servicios de lo que hemos llamado democracia electrónica, aparte del de elección de representantes propiamente dicho (todo tipo de votaciones, encuestas de opinión, participación más activa en la toma de decisiones, etc.).

Gran parte de los requisitos de seguridad (autenticación de votantes, privacidad, verificabilidad de los resultados, etc.) y de la implantación de los protocolos adecuados para llevarlos a cabo podrían verse simplificados con el uso de tarjetas inteligentes. Hay que resaltar el hecho de que un dispositivo de este tipo puede mantener información secreta incluso a su poseedor (aspecto interesante para no revelar información antes del final del proceso). Asimismo, permite manejar mecanismos criptográficos que mejoran la calidad del proceso al proteger los datos que se intercambian por la red.

Entre las ventajas más evidentes de este escenario está que el votante no tiene que desplazarse al colegio electoral, la red ya existe y no sería necesaria su implantación, y como aspecto también interesante a tener en cuenta está el hecho de que el derecho a la abstención (o el derecho a no abstenerse) podría garantizarse de una forma más

eficaz. Aunque, como contrapartida, las medidas de seguridad en la red se deben reforzar, se deben incrementar también los mecanismos de autenticación, y los ordenadores de los votantes deberían disponer de un lector para tarjetas inteligentes.

Mientras que este escenario parece idóneo para la mayoría de los servicios de democracia electrónica a escala local, en lo que se refiere a las votaciones a gran escala para la elección de representantes, parece más recomendable un método mixto que integre la votación por Internet con otros escenarios

Estos escenarios mixtos plantean una serie de interrogantes (si existen o no cabinas de acceso libre, cuántos agentes o "terceras partes de confianza (TTPs)" deben intervenir en el proceso, si se usa una tarjeta para cada votación o una permanente, etc.) que se estudiarán detalladamente en el proyecto VOTESCRIPT para encontrar la arquitectura idónea para cada tipo de votación.

También es interesante el análisis de los escenarios emergentes de democracia electrónica [7], basados en el uso de redes, y de las iniciativas que en los distintos países se están llevando a cabo en este campo. Asimismo, se considera de gran interés el estudio de posibles nuevos planteamientos al respecto.

4. Un nuevo planteamiento

Exceptuando los experimentos que tratan de establecer mecanismos de participación ciudadana, en las experiencias de voto electrónico analizadas y en los esquemas de votación que se han resumido en los apartados anteriores, se advierte un denominador común: las propuestas se realizan desde una perspectiva exclusivamente tecnológica, procurando mimetizar los planteamientos existentes en las votaciones con urna y soporte de papel que se han venido estableciendo en los últimos siglos, orientados solamente a la elección de representantes o a la decisión sobre alternativas previamente planteadas [8].

Al centrarse en ese esquema, se han dejado de lado dos aspectos fundamentales. El primero de ellos es que se han dejado sin explorar las riquísimas posibilidades que las nuevas tecnologías ofrecen para ampliar los marcos de decisión que el sistema tradicional acotaba. El segundo es que es necesario tener en cuenta las exigencias y recelos de sus hipotéticos usuarios: de nada sirve desarrollar un sistema telemático técnicamente perfecto que incluya innovaciones notables y use las más avanzadas técnicas, si el entorno social al que va dirigido, es decir, los ciudadanos para los cuales ha sido concebido, no confían en él o no responde a sus necesidades reales.

4.1 Un enfoque multidisciplinar

Partiendo de estas premisas surge el proyecto *VOTESCRIPT*: *Votación Electrónica Segura basada en criptografía avanzada*, subvencionado dentro del Plan Nacional de I+D+I (código TIC 2000–1630-C02). Este proyecto ha dado comienzo en enero del presente año, de tal forma que, a la hora de redactar el presente texto, se encuentra en su situación inicial, estando en fase de ejecución las tareas conducentes a la definición de los distintos escenarios y arquitecturas de votación y participación que vayan a ser contempladas.

Consecuentemente con lo comentado en los párrafos anteriores, consideramos necesario, para un adecuado diseño global de las distintas arquitecturas, tener en cuenta los requisitos tanto técnicos como sociales. Por tanto, en los desarrollos que se lleven a cabo, se considera imprescindible que al mismo tiempo que se realicen los trabajos de ingeniería correspondientes, se hagan análisis sociológicos, politológicos y jurídicos para determinar la viabilidad de los sistemas.

Con esta perspectiva metodológica, el desarrollo del proyecto se ha abordado mediante un equipo interdisciplinar. Está dividido en dos subproyectos coordinados: uno de ellos con sede en el Departamento de Ingeniería y Arquitecturas Telemáticas, DIATEL (Universidad Politécnica de Madrid) y el otro en el Departamento de Ciencia Política y de la Administración III (Universidad Complutense de Madrid).

4.2 Algunos requisitos de seguridad para los procesos de votación

Cuando se trata de detectar los requisitos que los sistemas de votación electrónica deberían tener para emular las garantías que en la actualidad ofrecen los sistemas convencionales, suele presentarse una lista de ellos que, con frecuencia, adolece de falta de extensión y rigor, sobre todo en lo que a su significación social y política se refiere. Así, puede encontrarse como requisito de seguridad la "democracia" rebajada a la simple exigencia de que sólo puedan votar las personas autorizadas y que solamente voten una vez. No es necesario insistir en que *Democracia* (gobierno del pueblo) es un concepto de mayor alcance y complejidad.

No obstante, y aún partiendo de la base de que la definición de estas características (que son objeto de análisis detenido dentro del proyecto) requiere de mayor espacio, precisión y evaluación, adelantamos de forma resumida, las siguientes:

Autenticación: sólo los votantes autorizados pueden votar. Hay que resaltar que, en principio, consideramos aquí el concepto de *voto* y *votante* en sentido amplio, válido también para aquellos

escenarios en los que un voto puede ser una opinión o una propuesta.

Fiabilidad: no se puede producir ninguna alteración fraudulenta de los resultados de la votación. Si se trata de una elección de representantes o de algún tipo de consulta sobre opciones predeterminadas, los votantes no pueden votar más de una vez, restricción que, en principio debería de acotarse de forma distinta en otros escenarios de participación.

Veracidad de la votación, de manera que si se descubre algún defecto en la publicación de los resultados, existan mecanismos para probar el fraude. Esta característica se puede considerar como una prueba global de la fiabilidad.

Anonimato: no se puede relacionar un voto con el votante que lo ha emitido. Este es un requisito que aparece en casi todos los posibles escenarios. Su cumplimiento suele conllevar o bien el concurso de varias TTPs o el uso de mecanismos criptográficos avanzados basados en firmas ciegas, secreto dividido, etc. El uso de tarjetas inteligentes de diseño específico puede aportar soluciones interesantes para escenarios sensibles como son los de elección entre propuestas predefinidas.

Un requisito que es dificilísimo de cumplir en los actuales sistemas de votación con papeletas e interventores es el de un hipotético anonimato en relación con la abstención. Si fuese requerido, conllevaría que se pueda conocer cuántos y qué votan pero no quiénes participan. Este es un caso típico de posibilidades que requerirían una modificación de la normativa electoral, al menos en el caso de España, pero que, en muchos ámbitos representaría una mejora notable de democracia y de libertad individual. Por contra, no sería de aplicación en ambitos en los cuales el voto además de un derecho es una obligación.

Imposibilidad de coacción: ningún votante debe ser capaz de demostrar qué voto ha emitido. De esta forma se impide la compra masiva de votos y la presión sobre los votantes, ya que la persona que desea influir sobre otra u otras no puede obtener garantía del resultado de su acción.

Verificación individual: cada votante deberá poder asegurarse de que su voto ha sido considerado adecuadamente, de forma que el votante pueda obtener una prueba palpable de este hecho.

Definida de esta forma, puede aparecer una cierta contradicción con el requisito de *imposibilidad de coacción*. Cuanto más explícita es la verificación más riesgos de coacción pueden aparecer. No obstante, se pueden diseñar mecanismos no exclusivamente telemáticos, que hagan compatibles ambos requisitos. En el sistema convencional el votante sabe lo que vota, y confía que será contabilizado correctamente cuando comprueba que es introducido en la urna (verificación). Si usa la cabina, conforme a como esta previsto, para cumplimentar su voto, no hay peligro evidente de coacción. Como puede intuirse, un estudio mínimamente riguroso del balance entre los requisitos de *verificación* y *coacción* requeriría la inclusión y análisis de más parámetros dependiendo de los distintos condicionantes sociales.

En escenarios de participación mediante la emisión de votos razonados, la prueba de *verificación* es inmediata al comprobar el participante que su aportación está reflejada y tenida en cuenta en el proceso de discusión.

Neutralidad: todos los votos deben permanecer en secreto mientras no finalice el tiempo de la elección. De este modo, los resultados parciales no afectarán a la decisión de los votantes que no han depositado su voto todavía.

Una expansión del actual sistema de **democracia representativa** es posible merced a la implantación de esquemas telemáticos de voto electrónico. Se abre una plétora de posibles modificaciones dentro del propio sistema representativo (de las que aquí adelantamos tan sólo dos ejemplos), que serían impracticables en los sistemas convencionales debido a la complejidad y coste de gestión que conllevarían.

Este es el caso de las listas abiertas y ponderadas. Este sistema permite combinar en una misma "papeleta" la elección de candidatos de varios partidos. Es similar al proceso de votación para el Senado, sólo que estableciendo jerarquías entre los votados. Por otra parte, en el caso de la elección de representantes, las reglas del juego podrían ser distintas en función del tamaño de la colectividad de que se tratase, de tal manera que, cuanto más reducida sea ésta más coyuntural y condicionada podría ser la elección de un delegado o representante. Se puede implantar un sistema de discusión y evaluación permanente, en la cual se tendría capacidad para revocar cargos que se considere que no cumplen adecuadamente con las tareas que les fueron asignadas.

4.3 Sistemas de Democracia Directa

A la hora de diseñar nuevos esquemas de soporte telemático, una tarea fundamental de partida sería definir los distintos dominios o ámbitos de aplicación del sistema, y llevar a cabo el análisis y diseño de los esquemas de votación y participación que serían demandados [7]. En esta línea hemos de apostar por el diseño y construcción de esquemas de evaluación, discusión y toma de decisiones que tengan su origen en las necesidades concretas de los ciudadanos. Esto puede dar lugar a sistemas telemáticos de consulta que habría que integrar dentro de nuestro actual sistema de Democracia Representativa.

Frente a la relativamente fácil identificación de los aspectos necesarios para desarrollar sistemas de votación convencional a través de la telemática, descritos anteriormente, se ha establecido como prioridad la identificación y definición del tipo de requisitos sociales necesarios para desarrollar los protocolos que permitan una implantación y desarrollo de la democracia electrónica sin que se limiten sus posibilidades.

Una primera consideración que marca las diferencias entre el esquema convencional y el que posibilita el debate y la toma de decisiones estriba en la **interactividad** que permiten las redes. Así, se plantea el desarrollo de protocolos que permitan conocer las opiniones de los ciudadanos, pero más allá de la encuesta, sondeo o referéndum, en las cuales las preguntas le son planteadas al ciudadano sin apenas opcionabilidad. Con este planteamiento, las deliberaciones no siempre deberían funcionar bajo parámetros marcados, con lo que las posibilidades de la democracia electrónica pueden superar en mucho a la estructura del referéndum clásico.

Como líneas de partida de la investigación de los aspectos sociales nos planteamos a consideración los siguientes condicionantes.

- **Interactividad**. La consulta a los ciudadanos debiera basarse en una **interactividad simétrica**. Debería permitirse que la ciudadanía tuviese la posibilidad de preguntar sobre las cuestiones planteadas e incorporar modificaciones al debate. De forma tal que cualquier pregunta planteada en encuesta o refrendo no debería estar cerrada.

- **Condicionalidad**. La ciudadanía debería de estar en disposición de responder condicionalmente. Es decir si *x* ocurre entonces *y*, pero en ausencia de *x*, no se apoya *y* o se propone *z*.

- **Elección Múltiple**. Los sistemas telemáticos podrían permitir un sistema de consulta múltiple escalonada, en las que la ciudadanía fuera capaz de ir perfilando los detalles y eligiendo. Esto podría permitir dilucidar los elementos de consenso y centrarse en buscar soluciones a los problemas que susciten diferencias.

- **Accesibilidad.** La cuestión de la accesibilidad plantea serios problemas. El acceso desde casa, quizás a través de Internet, plantea innumerables ventajas, pero conlleva unos riesgos capitales en lo relativo a la Estratificación Digital. Un sistema de Democracia Electrónica conlleva el derecho de acceso del conjunto de la ciudadanía: sería una proyección telemática del concepto de Sufragio Universal. Una posible solución es la participación a través del establecimiento de kioscos de votación conectados a la red en la que se gestionan las consultas. Otra propuesta, quizás algo utópica, sería la subvención de la compra de ordenadores y acceso desde casa para el conjunto de la población. Si bien resulta tremendamente costosa, sí que constituiría un paso definitivo para acabar con la Estratificación Digital, además de permitir la ampliación de otras facetas de la democracia electrónica.

- **Rechazo o veto.** Quizás debieran instituirse mecanismos sobre la posibilidad de anulación de una consulta si un porcentaje cualificado de la ciudadanía no se sienten representados dentro del arco de opciones planteadas o con la forma en que se organiza el debate. El objetivo sería evitar que algunos participantes se sientan atrapados y sin ser capaces de expresar sus opiniones en la consulta.

5. Conclusiones

Las innovaciones tecnológicas abren y cierran puertas para la mejora de los derechos cívicos. La implantación de sistemas de votación y Democracia Electrónica debe servir no sólo para garantizar que sean respetados los derechos y salvaguardas actualmente reconocidos en los sistemas de votación convencionales, sino para, aprovechando las posibilidades que ofrecen las redes telemáticas, conseguir mayores niveles de participación y decisión.

Los trabajos del proyecto *VOTESCRIPT* están abordando el diseño e implementación de diversos escenarios de votación y participación desde una perspectiva multidisciplinar, teniendo en cuenta los requisitos tanto técnicos como sociológicos y políticos. Para poder ofrecer estas facilidades se emplearán mecanismos criptográficos avanzados, tarjetas inteligentes de nueva configuración y se desarrollarán terceras partes de confianza (TTPs) especializadas.

Referencias

[1] Carracedo, J.D."Attenting to understand the Digital Divide". Civic Collaborative Center. University of California, San Diego. June 2000.

[2] http://www.etcetera.com.mx/pag24n2.asp

[3] A Report on the Feasibility of Internet Voting. http://www.ss.ca.gov/executive/ivote/final_report.htm, January 2000.

[4] http://www.safevoto.com/contracosta/index.html#Report

[5] Internet Society. 1999 ISOC Board of Trustees election. http://www.isoc.org/members/vote/99election/]

[6] http://www1.euskadi.net/botolek/

[7] Tsagarousianov R., editor. "Cyberdemocracy, technologies, cities and civic networks". Routledge, London 1998.

[8] Riera A.,"Design of implementable solutions for large scale Electronic voting schemes". Ph. Doctoral Thesis in Computer Science, University of Barcelona 1999.

Análisis y Caracterización de Tráfico IP en la Red Regional Ciez@net

María Dolores Cano, Josemaría Malgosa Sanahuja, Fernando Cerdán, Joan García Haro
Universidad Politécnica de Cartagena
Departamento de Tecnologías de la Información y las Comunicaciones
Campus Muralla del Mar s/n (Ed. Hospital de Marina)
30202 Cartagena, España
Teléfono: 968 325 953 Fax: 968 325 338
E-mail: {mdolores.cano,josem.malgosa}@upct.es

Abstract*. In this paper, we present the most significant results studying the Internet network traffic measurements obtained in Ciez@net. Ciez@net is a citizen subnet located in the village of Cieza that belongs to the regional network of the Autonomous Community of Murcia in Spain. This subnet is one of the firsts pilot experiences of a Digital City in Europe and the first in the Region of Murcia. The goal is the seamless introduction of the Information Society in a medium-size population. Access to advanced electronic information services is stimulated or subsidized for an effective penetration. These measurements will allow a qualitative and quantitative knowledge of the network traffic in order to achieve a most effective network resource provisioning and Internet traffic forecasting in a real scenario. A suitable dimensioning of the network as well as an adequate provisioning of Quality of Service to users may depend partially on these results. Measurements were taken from a Frame Relay link connecting Ciez@net users to Internet through the main node located in Murcia city. We used a promiscuous network analyzer that avoids interfering with network traffic. We report results of traffic load, network performance, percentage composition of traffic by protocol and type of application, and IP packet size distribution in both up and down communications streams.*

1 Introducción

En los últimos años el incremento del número de usuarios, del volumen de tráfico, de nuevas aplicaciones y de la topología en Internet está causando un tremendo cambio en la naturaleza del tráfico que genera [1]. En este contexto, se hace necesario realizar estudios sobre el tráfico de Internet y sus tendencias en escenarios reales [2].

En este artículo, hemos realizado un análisis y monitorización del tráfico de Internet sobre una subred real de ciudadanos denominada Ciez@net. El proyecto Ciez@net [3] es la primera experiencia piloto de *Ciudad Digital* realizada en la Región de Murcia. Ciez@net provee a sus usuarios con un acceso básico a Internet con tecnología RDSI-BE hasta el ISP *(Internet Service Provider*) y finalmente a través de un enlace *Frame Relay*.

Para realizar la captura y monitorización de datos se utilizó el analizador de redes DominoWAN DA-310. Además hemos desarrollado una herramienta *software* para interpretar los resultados [4].

El resultado de las medidas nos permiten conocer aspectos de la subred como son la carga de tráfico, direcciones web más visitadas, número de usuarios conectados, distribución de tamaño de paquetes IP, composición del tráfico por protocolo y aplicación, y distribución de la longitud de paquetes por servicios.

El resto de este artículo queda organizado como sigue. La sección 2 explica la infraestructura y metodología. Las Secciones 3 y 4 presentan los resultados. Finalmente la Sección 5 destaca los puntos más relevantes de este trabajo.

2 Infraestructura y metodología para el análisis del tráfico

Los usuarios de Ciez@net se conectan al ISP mediante 8 líneas RDSI. El tráfico final se enruta hacia Internet a través de un enlace *Frame Relay* a 512 Kbps (Fig. 1), donde se tomaron las medidas.

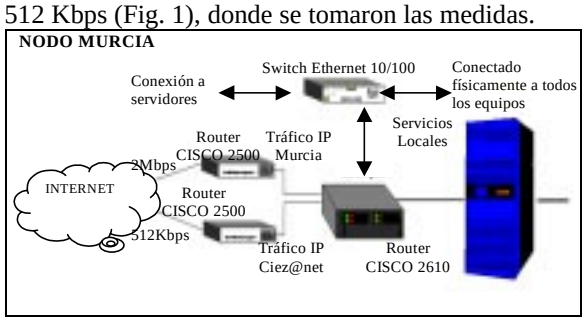


Fig. 1: Equipos de comunicación en el nodo de Murcia.

3 Medidas de carga de tráfico y prestaciones

Como se observa en la Fig. 2, el enlace de subida (de Ciez@net a Internet) está siendo utilizado muy por debajo de su máxima capacidad (512 Kbps). Se han realizado medidas del número de usuarios de Ciez@net conectados de forma instantánea para las muestras anteriores a intervalos de 5 minutos. Ver Fig. 3. Asimismo, una vez calculadas las diez direcciones de Internet más visitadas en promedio distinguiendo entre días laborables y festivos, destaca *www.terra.es*, probablemente por ser la

dirección de conexión por defecto de un gran número de usuarios.

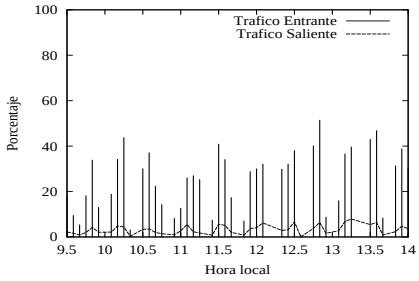


Fig. 2: % Utilización enlace en día laboral (mañana).

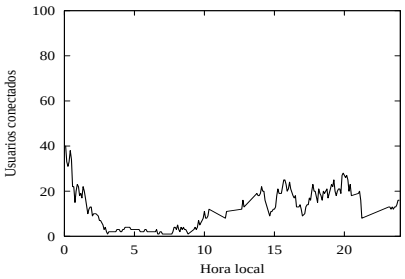


Fig. 3: Número de usuarios conectados en día festivo.

4 Medidas de caracterización de tráfico IP

Básicamente existen tres grupos de tamaños predominantes: paquetes cortos (40-150 bytes), paquetes medianos (500-600 bytes) y paquetes relativamente grandes (1400-1550 bytes). El tráfico de entrada tiene una mayor proporción de paquetes grandes (Fig. 4) debido a las transferencias masivas de datos. En cambio, la mayor parte del tráfico de salida (Fig. 5) se compone de paquetes cortos (peticiones a servidores externos de información).

En cuanto a servicios más utilizados la aplicación dominante en el enlace de bajada es *www*. Además existe un porcentaje de tráfico de control TCP muy elevado en el enlace de subida. Otros servicios importantes son IRC, FTP, POP3 o RTP. El tráfico denominado *Desconocido* (hasta un 16% en el enlace de bajada) engloba servicios cuyos puertos TCP no están estandarizados (juegos *on line*, etc.).

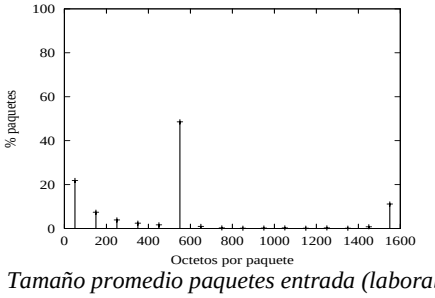


Fig. 4: Tamaño promedio paquetes entrada (laborable)

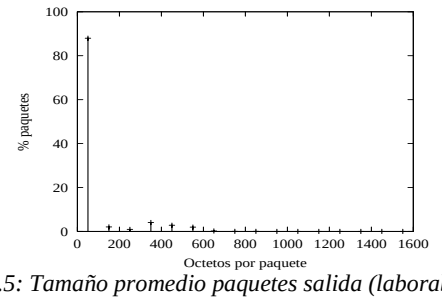


Fig.5: Tamaño promedio paquetes salida (laborable).

5 Conclusiones

En este trabajo se han obtenido un conjunto de medidas de tráfico generado en la subred de ciudadanos Ciez@net. Con ello se ha logrado iniciar un estudio sobre el conocimiento de las prestaciones de la infraestructura de red instalada y en especial, de las características del tráfico que actualmente cursa. Además este estudio puede contribuir a una mejor compresión del comportamiento del tráfico de Internet en escenarios reales, especialmente aquellos conformados por ciudadanos residenciales [5].

Las medidas de carga horaria y diaria de tráfico del enlace *Frame Relay* revelan que la ocupación máxima del enlace es casi del 90% en el sentido de entrada y apenas supera el 10% en el sentido de salida. Lo que supone una clara infrautilización del enlace.

En la mayoría de las gráficas obtenidas queda plasmada la gran dispersión estadística típica de Internet (mayor conforme más nos acercamos al usuario final) en cuanto a patrones horarios, uso de aplicaciones y longitud de los paquetes IP. Esta variabilidad radica fundamentalmente en el hecho de que el colectivo de usuarios es heterogéneo.

La sobrecarga de paquetes de control TCP es debida principalmente a las aplicaciones *www*, ya que requieren múltiples conexiones TCP por página. El uso alternativo de otra tecnología como sería por ejemplo ATM no resolvería el problema [6]. Una posible solución sería replantear el diseño del protocolo HTTP, o bien, sustituirlo por otro más eficiente sin olvidar el problema de compatibilidad con las aplicaciones actuales.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Educación y Ciencia (TIC2000-1734-C03-03) y por la Fundación Integra con el proyecto 0124.

Referencias

[1] S. McCreary. “Trends in Wide Area IP Traffic Patterns”. Monterrey Mayo 2000. <old.caida.org/Papers/AIX00/>

[2] Kc Claffy. “Internet Measurements and data analysis: topology, workload, performance and routing statistics”. NAE ’99 workshop.

[3] “The Ciez@net project”. <www.f-integra.org/projects.htm#ciezanet>

[4] Josemaria Malgosa-Sanahuja, Maria-Dolores Cano, Fernando Cerdan, Joan Garcia-Haro. “TAT, Traffic Analysis Tool For the Statistics Analysis of IP Networks”. Proceedings of IEEE PACRIM ’01.

[5] Maria-Dolores Cano, Josemaria Malgosa-Sanahuja, Fernando Cerdan, Joan Garcia-Haro. “Internet Measurements and Data Study over the Regional Network Ciez@net”. Proceedings of IEEE PACRIM’01.

[6] M. Alvarez-Campana, A. Azcorra, J. Berrocal, J. R. Pérez, E. Vázquez. “Internet traffic measurements over the Spanish R&D IP/ATM Network Backbone”. Proceedings of IFIP ATM ’99.

EvalúaWeb. Una herramienta para la edición de exámenes, evaluación y autoevaluación a través de la Web.

Suárez Rivero, D, Marrero Marrero, D.
Departamento de Ingeniería Telemática. Universidad de Las Palmas de Gran Canaria.
Campus Universitario de Tafira. Edificios de Telecomunicación C.P. 35017
Teléfono: 928 451223 Fax: 928 451243
E-mail: dmarrero@cic.teleco.ulpgc.es

***Abstract**. This paper introduces an assesstment system designed to work over the Web. It is based on the JAVA Programming Language and Applets. The user can realize tests and other examinations. Using the Client-Server Model for the communication, an user in the client side can use the application with a simple Browser. This client side is formed for a GUI (Graphic User Interface) generated with applets in html page. The user can interact with the application like a tutor or student. As tutor, he can edit exams and evaluate them and as a student can realize the exams in form of self-evaluation or evaluation. In self-evaluation mode, the user can consult the correct answer and he can improve his acknowledges. In evaluation mode, the exams will be store in the server with a complete identification user. The tutor will be able to assess it.*

1 Introducción

El incipiente uso de Internet como medio para el intercambio de información, compartir recursos de toda índole y disponer de servicios remotos, invita a utilizar la red para realizar pruebas autoevaluativas o de cualquier otra índole. Con este servicio, los posibles usuarios pueden formarse a distancia, ya sea desde su propio domicilio o desde salas habilitadas para la realización de dichas pruebas, dependiendo, obviamente, de la importancia de las mismas.

Para facilitar esta tarea o como parte de lo que se ha venido en llamar "Universidad Virtual" [1], se pensó en desarrollar una aplicación que permitiera realizar evaluaciones de cualquier materia de manera remota, requiriendo únicamente de conexión a Internet y del Browser apropiado. Asimismo, el tutor o instructor puede elaborar las pruebas desde el browser, previa la autentificación correspondiente. A pesar de existir entornos educativos basados en WWW muy potentes como WebCT ver 1.3 [2], que abarcan un mayor abanico de posibilidades, la aplicación aquí presentada incorpora las características de estar realizada con código Java [3] (Applet), lo cual permite un GUI más elaborado frente al formulario y CGIs [5] asociados, la posibilidad de evaluación remota y, algo muy importante, la disponibilidad de fuentes para su mejora y revisión (no herramienta propietario).

2. El lenguaje JAVA y WWW

Para la realización de este servicio remoto se parte del Lenguaje de Programación JAVA [2] y sus componentes Applet [3] para la inserción de código en páginas Web [4]. Cualquier usuario que disponga de conexión a Internet y de un simple Browser que soporte Applets podría autoevaluarse de determinada materia dentro de unos estudios académicos. Además debería permitir la realización de pruebas mucho más controladas (evaluaciones).

3. Descripción de la Aplicación

La aplicación desarrollada dispone de los siguientes módulos accesibles desde el cliente:

- **Modulo Tutor**
- **Modulo Autoevaluación** y
- **Modulo Evaluación**

En la fig. 1 se muestra la imagen capturada de la portada de la aplicación dentro del entorno del Browser Netscape Communicator [6].

Figura 1. Portada Principal de la Aplicación

➢ A través del **Módulo Tutor**, el profesor Tutor o responsable de la aplicación puede realizar tras su identificación dos operaciones: editar las pruebas que desea que sean almacenadas en el servidor y puede calificar y controlar las pruebas realizadas.

Para la edición de exámenes o pruebas se puede seleccionar entre **objetivas (tipo test)**, **desarrollo o combinadas.** Para pruebas objetivas o tipo test, se indicará la pregunta y una lista de posibles respuestas. De entre estas, se deberá indicar cual o cuáles son las correctas para permitir la corrección

automática desde la misma aplicación sin necesidad de interactuar con el tutor. En el caso de cuestiones de tipo desarrollo únicamente se indicará la pregunta correspondiente.

Dentro del módulo tutor se dispone de lo necesario para la corrección de exámenes. El tutor podrá controlar y evaluar las preguntas tipo desarrollo según desee. Asimismo podrá analizar los exámenes realizados por los diferentes usuarios registrados y generar listados de los mismos. Estos datos podrán ser impresos en formato html para su publicación, para remitirlos por correo, u otros objetivos.

➢ A través del **módulo de autoevaluación** cualquier usuario que acceda al servidor donde se encuentren ubicadas las diferentes páginas html [5] de soporte a la aplicación podrá seleccionar, en el caso concreto del ámbito de la EUITT, el curso académico, la materia y la prueba deseada para que pueda autoevaluarse (ver figura 2).

Fig. 2 Módulo de Autoevaluación.

➢ Por último el **módulo de evaluación**, como su nombre indica, pretende realizar pruebas evaluatorias a través de la red. Para ello, el usuario deberá autentificarse con la adecuada clave de acceso que será provista por el tutor o evaluador. Un ejemplo de la parte de evaluación se muestra en la figura 3. Acabado el examen se procederá al almacenamiento y el cifrado del mismo en el servidor para su evaluación posterior.

Figura 3. Interfaz de examen tipo desarrollo

Si fuese tipo test, como se comentó previamente, se procederá a su corrección automática y se le mostrará la calificación obtenida; si lo desea, el usuario desde su ubicación podrá sacar una copia impresa en formato html [4] como copia para reclamaciones.

Fig. 4 Formato de salida para impresión

4. Conclusiones

El uso de Internet es sumamente atractivo al brindarnos un medio para realizar remotamente cualquier actividad. Concretamente en un entorno académico/docente se pueden confeccionar “sites” o portales de facultades o centros de enseñanza, donde se pueden encontrar apuntes, exámenes resueltos, documentación variada, etc. e incluso “Universidades Virtuales” [1] (en creciente implantación), con servicios de videoconferencia de clases teórico/prácticas, tutorías remotas vía e-mail o chats, entrega de documentación, etc. Además con herramientas como la aquí presentada se pueden superar las barreras físicas para realizar exámenes y autoevaluaciones. Mediante estas últimas, el alumno podrá complementar y mejorar sus conocimientos.

Referencias

[1] La “Universidad Virtual”. http://www.Fernuni-hagen.de, http://www.uoc.es., www.icde.org , http://www.open.ac.uk , http://www.uned.es

[2] Murray W. Golbert and Sasan Salari. “An Update on WebCT World-Wide-Web Course Tools”. Proc. NAUWeb Jun. 97 Arizona. http://www.webct.com

[3][4] El lenguaje de Prog. JAVA. Tutorial Swing. Clases. http://java.sun.com/docs/books/tutorial, http://www.sun.com/java. Los Applets de Java.

[5] HTML. W3 Cons. http://www.w3.org/MarKup http:/www.w3.org/CGI

[6] Servidor WEB Apache. http://www.apache.com

[7] Netscape Communicator. www.netscape.com

Servidor de Consulta Bibliográfica basado en ANSI/NISO Z39.50.

J. Alba Soto, J. M. Vozmediano Torres
Área de Ingeniería Telemática
Departamento de Ingeniería de Sistemas y Automática
Camino de los Descubrimientos, s/n, 41092 Sevilla
e-mail: albas@trajano.us.es, jvt@trajano.us.es

***Abstract**. This article addresses the problem of finding documents among a variety of libraries. Most modern library management applications support the ANSI/NISO Z39.50 protocol for information retrieval. The web-enabled system described in this article allows, by using this protocol, to query for a document in any library. The implementation was made in JAVA, with servlets to improve the performance.*

1 Introducción

El creciente auge de los motores de búsqueda basados en tecnologías Web permite a los investigadores localizar información con facilidad. Sin embargo, los fondos bibliográficos tradicionales siguen siendo un elemento indispensable de consulta. El proporcionar un procedimiento de localización de un determinado documento en una o varias bibliotecas es un aspecto a menudo descuidado en el universo de Internet.

La mayoría de las aplicaciones de gestión bibliotecaria modernas ofrecen una interfaz de comunicaciones que implementa el protocolo ANSI/NISO Z39.50 v.3 [1] para la consulta a los catálogos de fondos bibliográficos. En este artículo se describe una herramienta de búsqueda que, implementando dicho protocolo, permite localizar la biblioteca en la que se halla un determinado documento, agilizándose así notablemente el trámite de poder acceder posteriormente al mismo a través de los diferentes acuerdos ínter bibliotecarios.

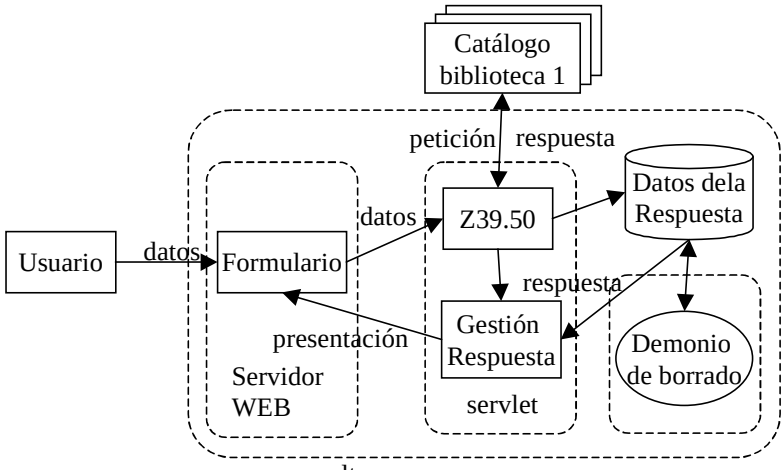
Por su universalidad y sencillez, se ha dotado a la aplicación de consulta de una interfaz que resulta accesible desde cualquier navegador HTTP. Asimismo, y en aras de la portabilidad, la implementación se ha desarrollado en Java [2], gestionando el diálogo y atención de las consultas mediante servlets [3].

En las siguientes secciones se describe la arquitectura de la aplicación y sus componentes principales, su funcionamientoy la forme en que se muestran los resultados.

2 Arquitectura y componentes

La herramienta desarrollada y que se describe en este artículo permite al usuario consultar de forma simultánea los catálogos bibliográficos de varias bibliotecas. El sistema está compuesto por los siguientes bloques básicos (ver figura):

1. El servidor WEB es el encargado de dar la bienvenida al usuario y recoger toda la información posible para pasarla a la aplicación. También se ocupa de presentar al usuario los resultados generados por su consulta.



2. El envío del formulario de recogida de datos desencadena la búsqueda correspondiente. Para ello se pasan los datos de la misma a un servlet. Como ya se ha dicho, se ha usado la tecnología servlet por ser más eficiente que CGI y permitir un mejor aprovechamiento de los recursos del equipo servidor. Por cada petición, el servidor lanza un hilo que la gestiona. En un mismo servidor puede haber servlets diferentes esperando ser invocados. Cuando se lanza un servlet, el servidor WEB pasa como parámetro un manejador para recibir datos del usuario (así es como se recogen los datos de entrada) y otro manejador para enviarle datos (usado para mostrar los resultados).

3. El protocolo ANSI/NISO Z39.50 es la norma internacional para consulta bibliográfica. Sin embargo, no todas las implementaciones comerciales las soportan. Para favorecer la modularidad del código, se ha generado una clase especial, que contiene una determinada librería de funciones. Con ellas se permite codificar el valor de las variables a BER y decodificar un paquete BER para obtener el valor de los diferentes parámetros. A su vez, de esta librería heredan las restantes clases que conforman el protocolo. Los métodos de la clase librería son la base para crear un conversor de código ASN.1 a código JAVA.

4. Gestión de Respuesta: Cuando se han recibido las respuestas de todos los servidores se gestionan conjuntamente. Tras ordenarse, se introducen en un fichero con formato HTML para que la presentación al usuario resulte más legible.

5. Los datos bibliográficos resultantes de una consulta pueden ser muy abundantes. Un demonio de borrado se encarga de limpiar periodicamente la información obsoleta.

Los bloques básicos residen en un equipo servidor donde se ejecutarán todas las operaciones necesarias y donde se guardarán todos los ficheros de datos. Es necesario que dicho equipo contenga un servidor WEB con soporte para servlets, así como la *Java Virtual Machine* para poder ejecutar el código de la Biblioteca Virtual.

3 Resultados

Tras la fase de implementación se han realizado múltiples pruebas de interconexión con todos los servidores comerciales Z39.50. Estas han permitido validar el funcionamiento de la herramienta y, además, detectar algunas deficiencias técnicas de las implementaciones comerciales. En concreto:

- Muchas implementaciones incumplen la norma Z39.50, apareciendo incompatibilidades con aquellas que sí la siguen.

- Los registros bibliográficos intercambiados pueden almacenarse según diferentes formatos (USMARC, IBERMARC, CANDIAN MARC, etc.). Esto ocasiona problemas sintácticos a una aplicación como la desarrollada, que integra respuestas de distintos servidores y por tanto en formatos potencialmente diferentes. La norma Z39.50 define un formato común de mínimos de obligado cumplimiento. Sin embargo, no todos los servidores Z3950 comerciales la soportan. A este respecto existen proyectos financiados por la Unión Europea como USEMARCON [5]. Esta aplicación es un traductor entre diferentes estos diferentes formatos.

La velocidad de consulta depende sobre todo del número de coincidencias encontradas en los catálogos. Se ha llegado a observar tiempos de espera de hasta 3 minutos en consultas que generaron hasta cuarenta ficheros de resultados válidos. El retraso viene impuesto por la capacidad disponible en la conexión entre el servidor Z39.50 de la biblioteca en cuestión y el servidor donde reside la aplicación desarrollada.

4 Conclusiones

Con este proyecto se ha pretendido facilitar el trabajo de aquellas personas que realmente usan Internet como un medio de trabajo y localización de información.

Este proyecto está desarrollado de tal manera que es fácilmente escalable, tanto en la incorporación de nuevos servidores bibliográficos como en la incorporación de nuevos servicios de interés para todos los usuarios de esta aplicación (como por ejemplo el préstamo ínter bibliotecario, consulta documental on-line, reserva de libros, etc.).

Es necesario destacar que las normas que afectan al ámbito de este proyecto no son respetadas por la mayoría de las aplicaciones probadas. Las dificultades encontradas inciden en la importancia que tiene ajustarse a dichas normas.

Agradecimientos

Los autores de este artículo desean expresar su agradecimiento al Comisario del Salón Internacional del Estudiante Andalucía 2000 (SIE 2000), a la Universidad de Sevilla, al C.I.C.A. y a Luis Miguel Rivas Asensio su participación en todas las fases de desarrollo del proyecto.

Referencias

[1] ANSI/NISO. "Information Retrieval (Z39.50): Application Service Definition and Protocol Specification". Julio 1995.

[2] Patrick Naughton, Herbert Schildt. “Java. Manual de Referencia”. McGraw-Hill 1997. ISBN 0-07-882231-9

[3] Sun Microsystem. “Documentación sobre Sevlets”. http://java.sun.com/docs/books/tutorial/servlets/

[4] UIT-T X.208. Especificación de la notación de sintaxis abstracta uno (NSA.1). UIT-T.

[5] USEMARCON. “ftp://ftp.konbib.nl/pub/usemarcon”.

Evaluación de un Modelo Publicación-Subscripción para Aplicaciones de Tiempo-Real en una Red de Área Local.

Juan Manuel López-Soler[1], José L. Castillo Ramírez, Gerardo Pardo-Castellote[2]

[1]**Departamento de Electrónica y Tecnología de Computadores**.
Universidad de Granada. 18071-GRANADA.
Teléfono: 958 243271 Fax: 958 243230. juanma@ugr.es
[2] **Real-Time Innovations, Inc**.
155A Moffett Park Drive, Sunnyvale, CA (EE.UU.) http://www.rti.com.

1 Introducción.

Para facilitar una interfaz universal a los servicios TCP/IP, se han desarrollado una serie de APIs basadas, por lo general, en el modelo de interacción *cliente-servidor*. Recientemente, han aparecido nuevas aplicaciones con requerimientos de tiempo real para las que el modelo *cliente-servidor* no es necesariamente el más adecuado. Como alternativa, se ha propuesto un modelo de interacción basado en un nuevo paradigma denominado *publicación-subscripción* (PS) [1]. En el nuevo modelo, un subscriptor se comunica sólo con el nodo que tiene la información a la que se subscribe, a diferencia de lo que ocurre en los sistemas cliente-servidor, en los que los clientes se comunican a través del servidor.

Este artículo, tiene por objetivo estudiar la viabilidad del modelo PS para la comunicación de aplicaciones de tiempo real en un entorno de área local. Para tal propósito se ha evaluado el "*middleware*" NDDS ("Network Data Delivery Service") [2] para el caso de direccionamiento de uno a uno y multidifusión.

2 Publicación-Subscripción para aplicaciones de Tiempo-Real.

Para facilitar la interacción de *muchos a muchos*, en el modelo PS, unas entidades se "suscriben" a los datos que necesitan y otras "publican" la información que producen. Productores y consumidores son anónimos; ninguno conoce hacia dónde van los datos, o en dónde se originan éstos.

Debido a la posibilidad de disponer de múltiples y simultáneos productores/consumidores, este modelo facilita la replicación y movilidad de las aplicaciones. Además, es predecible una mayor eficacia toda vez que los subscriptores no tienen que solicitar datos por cada transacción.

Una subscripción se describe mediante dos parámetros: *tema* y *tipo*. Además también se caracteriza con una serie de parámetros de *calidad de servicio* como por ejemplo *separación mínima*, *expiración* y *modo de subscripción*.

Los consumidores se basan en un mecanismo de notificación, con dos variantes: *Subscripción Inmediata* o *Sondeada* dependiendo de cómo se llame a la función ("*call-back*") responsable de atender a la notificación. En la Fig. 1, se ilustran cómo se usan los parámetros definidos para regular el ritmo de actualización de los consumidores. Una vez se ha notificado la llegada de una emisión, la subscripción no será notificada de nuevo por lo menos durante tiempo de *separación mínima*. Si no llegan más emisiones durante un tiempo igual a la *expiración*, el subscriptor será notificado de esta contingencia.

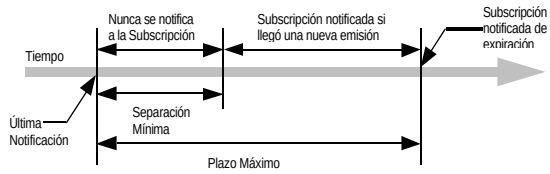


Fig. 1: Cadencia de las actualizaciones

Las publicaciones se describen igualmente mediante el *tema* y *tipo*, además de una serie de parámetros de calidad de servicio tales como *fuerza* y *persistencia*, los cuales permiten que los consumidores sean tolerantes a fallos arbitrando entre distintos productores de la misma publicación.

Se definen tres modos de publicación: *síncrono*, *señalado* y *asíncrono*. El *modo síncrono* indica que la emisión debe ser enviada inmediatamente y además en el contexto de la misma hebra de la aplicación. El *modo señalado* indica que la emisión debe ser enviada inmediatamente, pero por otra hebra distinta. El *modo asíncrono* indica que la emisión no será enviada inmediatamente, sino que será almacenada temporalmente.

3 Resultados Experimentales.

NDDS v2.2 [1] está construido sobre el modelo PS; para su evaluación se han llevado a cabo una serie de medidas experimentales en un entorno de red aislado Ethernet a 10 Mbps con distintas configuraciones, de 1-a-1 y de 1-a-4 unidifusuón y en multidifusión.

El escenario de test consiste en la transmisión de un fichero conteniendo un "stream" de vídeo. Se dispone de un *Productor* junto con una serie de *Consumidores*. Durante las pruebas se ha variado el tamaño de los paquetes y la cadencia de las publicaciones.

Se han considerado tres *temas*: **Control_de_Flujo, Vídeo** y **Respuesta**. El tema **Control_de_Flujo** se utiliza para iniciar la transmisión, sincronizando a los posibles subscriptores, los cuales se subscriben en modo fiable. El tema **Vídeo** es la publicación cuyas características vamos a medir. El *Productor* publica en modo *síncrono* y el *Consumidor* se subscribe en el modo de *subscripción inmediata* para alcanzar la mínima latencia. Para que podamos enviar tantas emisiones como sea posible se fija la *separación mínima* de los *Consumidores* a 0. Todos los paquetes del tema **Vídeo** han sido numerados para poder determinar la pérdida de paquetes. El tema **Respuesta** ofrece la posibilidad de enviar información desde el *Consumidor* hasta el *Productor*. La publicación **Respuesta** comunica el número de bytes recibidos por el *Consumidor* permitiendo al *Productor* determinar el porcentaje de pérdidas y tomar las acciones apropiadas cuando sea necesario. Para simular distintas velocidades de generación del "*stream*" se ha incluido un parámetro denominado *tiempo de espera* que permite de una forma controlada introducir retardos entre publicaciones consecutivas.

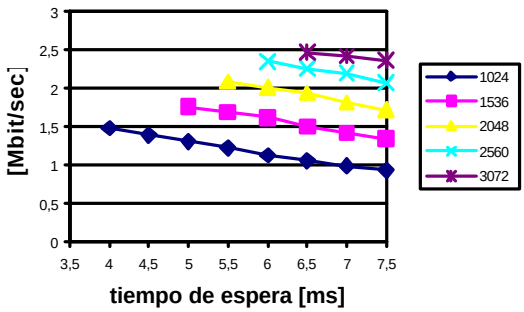


Fig. 2: Resultados de 1-a-1.

En la Fig. 2 se muestran los resultados para el caso de transmisión de 1 a 1. Es interesante observar que la productividad no siempre aumenta al reducir los tiempos de espera. La razón de este hecho reside, según se ha comprobado experimentalmente, en que las fluctuaciones corresponden con aumentos en el número de pérdidas en la recepción. En el siguiente experimento (Fig. 3) se miden velocidades de transmisión desde 1 *Productor* hasta 4 *Consumidores*. Lo valores representados se obtienen como la media de los valores medidos en los cuatro *Consumidores*. Los valores en este caso son mucho menores que los valores para el caso 1-a-1 (Fig. 2) toda vez que cada paquete tiene que ser

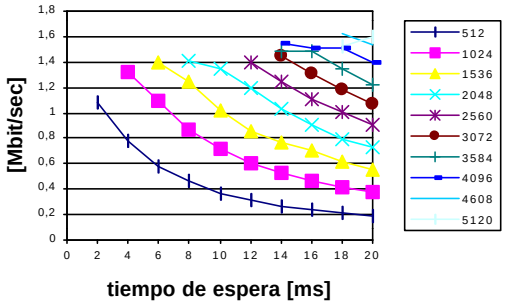


Fig. 3: Resultados de 1-a-4 en unidifusión.

enviado cuatro veces. El periodo de una publicación consiste en el intervalo para enviar (*send*) mas el *tiempo de espera*. Evidentemente, cuando hay más de un *Consumidor*, en modo unidifusión, la función *send* consume más tiempo, mientras que el *tiempo de espera* permanece sin cambios. Por este motivo, la productividad no es cuatro veces más baja que en el caso de 1-a-1.

Las siguientes series se llevaron a cabo para cuatro *Consumidores* usando multidifusión. (Fig. 4).

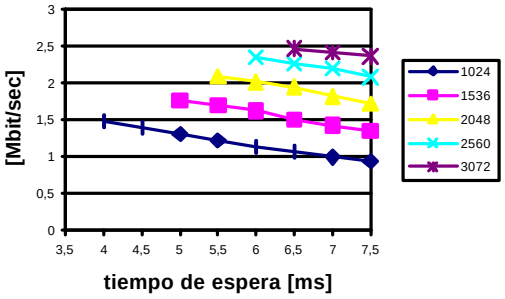


Fig. 4: Resultados de 1-a-4 en multidifusión.

Las prestaciones de los experimentos de la Fig. 4 son ligeramente mejores a los del escenario de unidifusión de la Fig. 3. No obstante, la similitud entre las prestaciones sugiere que el tiempo de transmisión propiamente dicho implica una fracción relativamente pequeña del tiempo total involucrado.

Referencias

[1] Pardo-Castellote, Gerardo, Schneider, Stan, Hamilton, Mark: ""NDDS: The Real-Time Publish-Subscribe Network. White Paper. Agosto 1999. Real Time Innovations, Inc.

[2] Pardo-Castellote, Gerardo, Schneider, Stan: "The Network Data Delivery Service: a Real-Time data Connectivity System". IEEE Proceedings of the International Conference on Robotics and Automation. Mayo 1994.

Aplicación de un método de solución matricial-geométrico al análisis de disciplinas de servicio *

José Ramón Piney
Sebastiá Sallent
Universidad Politécnica de Cataluña, Barcelona
E-mail: {jpiney, sallentg}@mat.upc.es

Abstract We present the application of a matrix-geometric solution method in order to develop an analysis of a discrete GPS (Generalized Processor Sharing) scheduling discipline. The discrete system from which we get the analysis support two kind of services or guarantees. The results of this analysis permit us to compare other disciplines and see how far are from GPS, which is an ideal discipline from the point of view of latency and fairness. We present a example with a simulation of a node that implements a WFQ (Weigthed Fair Queueing) discipline and we compare it with the analysis.

1 Introducción

En la actualidad uno de los desafíos más importantes en el diseño de redes de alta velocidad es el requerimiento de proporcionar garantías de calidad de servicio (QoS). Las redes necesitan gestionar sus recursos (búffers y ancho de banda) para poder soportar diferentes tipos de servicio mediante sistemas de gestión de búffers y algoritmos de planificación (scheduling algorithms). La mayoría de los análisis que existen de disciplinas consideran el tráfico acotado por un leaky bucket, lo cual les permitir englobar varios tipos de tráfico pero sin obtener resultados muy precisos. En nuestro caso lo que se hizo fue considerar un tipo específico de tráfico de entrada, lo cual reduce la complejidad del sistema y nos permite obtener un análisis muy aproximado del funcionamiento de una disciplina.

En este trabajo se presenta la aplicación de un método numérico para el análisis de la disciplina GPS [1], la cual es una disciplina ideal no realizable pero que es óptima desde el punto de vista de latencia (tiempo transcurrido hasta que un mensaje empieza a recibir servicio) y justicia (el servicio recibido por cada tipo de usuario normalizado al valor de su reserva debe ser el mismo para todos). Este análisis es una generalización del trabajo presentado en [3].

*Este trabajo ha sido auspiciado por el proyecto TIC1998-0495-CO2-01.

2 Descripción del sistema

El sistema sobre el cual se realiza el análisis es un sistema discreto con una disciplina de planificación que da servicio a dos colas, a las que nos referiremos como cola uno y cola dos. Los mensajes de llegada a cada cola se describen mediante un proceso de Bernoulli, donde con una probabilidad p puede ocurrir una llegada de un mensaje en una ranura y con una probabilidad $1 - p$ no llegan mensajes. A cada cola se le garantiza una tasa mínima de servicio, lo que se traduce en una determinada garantía de retardo. Suponemos que los mensajes de la cola uno reciben una tasa mínima garantizada de $\frac{n_1}{n}$ y los mensajes de la cola dos de $\frac{n_2}{n}$, donde $n = n_1 + n_2$. Para emular este funcionamiento se hace la suposición de que cada mensaje consiste de n paquetes. Si únicamente una cola tiene mensajes, entonces el sistema transmite n paquetes de esa cola en una ranura, pero si las dos colas tienen mensajes, el sistema transmite n_1 paquetes de la cola uno y n_2 paquetes de la cola dos (esto se puede ver como dos colas con dos servidores donde la tasa de servicio de cada servidor depende del estado de la otra cola). De esta forma obtenemos una cadena de Markov discreta donde cada estado de esta cadena representa el número de paquetes en las colas.

3 Solución matricial geométrica

Los estados de la cadena de Markov discreta que describen el modelo están denotados por $E_{i,j}$, $i \geq 0$ y $0 \leq j \leq n - m$, donde i es el número de paquetes en la cola uno del sistema y j es el número de paquetes en la cola dos del sistema (consideramos esta cola finita con un tamaño de m mensajes porque para aplicar el método de solución matricial-geométrica [2] así se requiere).

Los estados del sistema se reagrupan de la siguiente manera

$$\nu_i = \begin{cases} [E_{ni;0}; E_{ni-1;1}; \ldots; E_{0;ni}] & 0 \le i \le m \\ [E_{ni;0}; E_{ni-1;1}; \ldots; E_{n(i-m);nm}] & i > m \end{cases}$$

Según [2], si una cadena de Markov P es recurrente positiva, entonces

$$\nu_{i+1} = \nu_i R \qquad (1)$$

para $i \ge 0$, donde R es la matriz de tasa. En nuestro caso, esto se cumple para $i \ge m$.

Utilizando la relación (1), obtenemos el sistema $\pi P = \pi$ donde

$$\pi = (\nu_0; \nu_1; \ldots; \nu_{m+1})$$

y

$$P = \begin{pmatrix} F_{00} & F_{01} & F_{02} & \ldots & 0 & 0 \\ M_1 & D_1 & L_1 & \ldots & 0 & 0 \\ 0 & M_2 & D_2 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \ldots & L_{m-1} & 0 \\ 0 & 0 & 0 & \ldots & D_m & L_m \\ 0 & 0 & 0 & \ldots & A_2 & A_1 + RA_2 \end{pmatrix}$$

Para la definición de los bloques se supone que a lo más puede llegar un mensaje de cada usuario por ranura.

Conocidos el vector π y la matriz de tasa R, podemos obtener una expresión cerrada del número medio de paquetes en cada cola.

El número medio de paquetes en la cola uno, $\bar{q}_1$, está dado por

$$\bar{q}_1 = \sum_{i=0}^{nm} \sum_{j=0}^{ni} (ni - j)\nu_i u_{j+1} + \nu_{m+1} nR(I - R)^{-2} \sum_{j=0}^{nm} u_{j+1} + \nu_{m+1}(I - R)^{-1} \sum_{j=0}^{nm} [n(m+1) - j] u_{j+1}$$

El número medio de paquetes en la cola dos, $\bar{q}_2$, esta dado por

$$\bar{q}_2 = \sum_{i=1}^{nm} \sum_{j=0}^{ni} j\nu_i u_{j+1} + \nu_{m+1}(I - R)^{-1} \sum_{j=0}^{nm} j u_{j+1}$$

4 Resultados

Se muestran los resultados al comparar nuestro análisis con un sistema que representa un nodo Di®Serv (de Servicios Diferenciados) según el esquema reconocido por el Internet Network Working Group, que soporta dos grupos de usuarios,

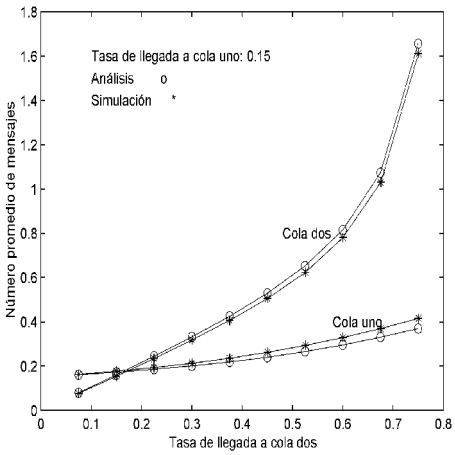


Figura 1: Sistema con reserva de 1=3 para la cola uno y 2=3 para la cola dos. Tasa de llegada a cola uno ¯ja y de valor 0:15.

el EF PHB [4] y el AF PHB [5]. El grupo EF debe tener la más alta prioridad y el grupo AF debe asegurar una cierta calidad. Suponemos que este nodo utiliza una disciplina WFQ. En la ¯gura los mensajes del grupo AF y los del grupo EF corresponden a la cola uno y a la cola dos, respectivamente. En la Fig. 1 se muestra el sistema al considerar un patrón de reserva de 1=3 para el grupo AF y de 2=3 para el grupo EF, bajo diferentes tasas de llegada de mensajes. Como se puede observar el WFQ (emulación del GPS) tiene un funcionamiento muy similar al GPS en cuanto a valor medio.

Referencias

[1] Abhay K. Parekh and Robert G. Gallager. A generalized processor sharing approach to °ow control in integrated services networks: the single-node case. IEEE Trans. on Networking, 1(3):344{357, Jun. 1993.

[2] M. F.Neuts. Matrix-Geometric Solutions in Stochastic Models: An algorithmic Approach. Dover Publications, Inc., 1981.

[3] J. R. Piney and S. Sallent. Analysis of a discrete system with Generalized Processor Scheduling Discipline Aceptado en SCI 2001/ ISAS 2001.

[4] V. Jacobson et al. An Expedited Forwarding PHB. RFC 2598, June 1999

[5] J. Heinanen et al. Assured Forwarding PHB Group RFC 2597, June 1999.

Software convertidor del lenguaje HTML a WML.

Ildefonso Ruano Ruano, Mª Dolores Molina González, Raul Mata Campos, José Antonio Marín Haro
Area de Ingeniería Telemática del departamento de Electrónica. Universidad de Jaén.
E.U.P. de Linares. C/ Alfonso X El Sabio nº 28 23700. Linares, Jaén
Teléfono: 953 026558 Fax: 953 026508
E-mail: alonso@ujaen.es

Abstract. *Nowadays a data transfer in a GSM mobile system (Global System for Mobile Communication) implies the use of WAP (Wireless Application Protocol). In this paper a new program to translate HTML (HyperText Markup Language) files to WML (Wireless Markup Language) files is presented. This software runs with HTML v.4.0 and WML v.1.2, it has been programmed in Java Language and intends to be a new tool to users who want to program or how to program in WML. This converter program has two different versions, one of them runs online inserted as an applet in a HTML page and the other one runs alone in a computer, this papers is about the stand-alone application and explain how it works.*

1 Introducción.

El presente artículo presenta un software realizado en lenguaje JAVA que como característica más singular permite realizar la conversión de código HTML (*HyperText Markup Language*) en WML (*Wireless Markup Language)*. Ambos lenguajes forman parte de lo que ha venido a denominarse tecnologías WEB, que constituyen el principal motor actual de desarrollo de Internet en el mundo. El lenguaje WML, además, esta relacionado con los servicios móviles ya que constituye el formato necesario para la visualización de información web en terminales móviles por medio de la tecnología WAP (*Wireless Application Protocol*) [1].

Este software pretende constituir por un lado una herramienta de ayuda a la hora de adoptar versiones WML de páginas HTML y por otro lado un sistema de ayuda para el aprendizaje del nuevo lenguaje para todo aquel que quiera aprender WML y posea conocimientos previos de HTML.

2 Entorno empleado y versiones.

Se trata de un software desarrollado en Java 1.2.2 que realiza conversión de código HTML a WML, las versiones de lenguaje empleadas son el HTML versión 4.0 [2] y el WML versión 1.2 [3].

El software obtenido posee una característica muy deseable y es la posibilidad de obtener, de manera muy asequible, una versión *applet* del mismo que posibilita su uso *online* a través de Internet por medio de páginas HTML. De esta forma se han obtenido dos versiones del software convertidor:

- Versión independiente, se ejecuta en cualquier PC que posea la JVM (*Java Virtual Machine*).
- Versión *online*, se ejecuta en cualquier PC que posea un *browser* HTML adecuado.

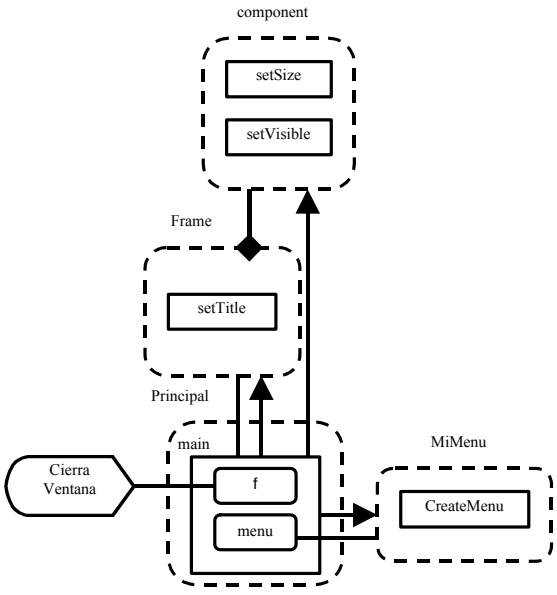
En este trabajo se trata la versión independiente.

3 Estructura del programa convertidor independiente.

La programación ha sido realizada aprovechando parte de las características de Java como es la de ser un lenguaje orientado a objetos, se ha implementado una Interfaz gráfica de usuario (GUI) con el paquete awt de la que parten todas las posibilidades del programa empleando un sistema de menús.

Para explicar su funcionamiento se van a mostrar una serie de diagramas de las partes más significativas del mismo que se han elaborado siguiendo la técnica de modelización de objetos de Rumbaugh adaptada a JAVA [4], la Fig. 1 muestra el diagrama general de la estructura del mismo.

Figura 1. Diagrama general del programa convertidor



Las clases que constituyen el software, agrupadas por las funciones principales que realizan, son:

- Clase Principal del programa: Principal.

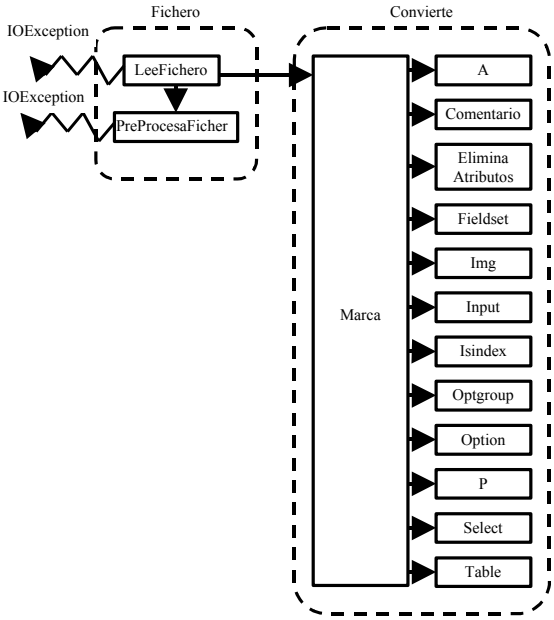
- Clases relacionadas con el manejo de ficheros de texto: Text y Fichero.
- Clases relacionadas con la conversión de HTML a WML: Fichero (nuevamente) y Convierte.
- Clases relacionadas con el entorno gráfico: el resto.

3.1 Conversión: Clases y métodos relacionados

Tras pulsar la opción "convertir" (fichero HTML previamente abierto o editado) el método actionPerformed lanza el método LeeFichero de la clase Fichero y con él comienza el diagrama de clases mostrado en la Fig. 2, realizándose las siguientes acciones por parte de cada uno de los métodos involucrados:

1. LeeFichero: Lee el fichero HTML y lo manda a un procesado previo.
2. PreProcesaFichero: Realiza un preformateo del fichero de entrada HTML abierto en la acción 1 y comprueba errores en su código HTML. En caso de que se encontrase algún error se abre una ventana que da aviso de los mismos, se explica porque se ha/n producido y se da opción a realizar la conversión (teniendo en cuenta que ésta va a encontrarse con errores y que el usuario debería corregirlos en el fichero WML generado) o a no convertir volviendo a la edición del fichero HTML. Detecta errores de instrucciones no cerradas y avisos de etiquetas imprescindibles no encontradas.
3. Marca: Comprueba el fichero en busca de las marcas existentes en el estándar HTML 4.0 (más de 90), dependiendo del elemento que encuentre puede realizar diversas acciones:
   - 3.1 Comentario: ignora las marcas que pudieran existir en su interior escribiéndolo sin cambios en el fichero de salida (los comentarios WML tienen el mismo formato que los de HTML), no convierte.
   - 3.2 Marca HTML con equivalente WML directo: lo escribe en el fichero de salida, conversión directa.
   - 3.3 Marca HTML simulable o que requiere un tratamiento personalizado de los atributos: se invoca al método correspondiente que realizará la conversión adecuada (A, Img, Fieldset, Input, etc...), conversión simulada.
   - 3.4 Marca HTML sin correspondiente WML: es eliminada (método EliminaMarca), no convierte.
   - 3.5 Texto que no se corresponde con ninguna etiqueta HTML: es escrito en el fichero de salida de la misma forma en que se encontró.

Figura 2. Diagrama de clases y métodos de la conversión.



El método marca es el método más importante del programa ya que es donde se realiza realmente la mayor parte de la conversión de formato HTML a WML. Existen otros métodos donde también se realiza conversión de código, son los métodos asociados en la misma clase (Convierte) que son "llamados" por el método Marca tal y como se explicó en el punto 3.3.

Tras ser realizada la conversión se muestra el fichero de salida en una nueva ventana que muestra el código WML generado a partir del archivo HTML suministrado.

4 Resultados obtenidos y conclusiones.

Se ha obtenido un programa con el que se puede editar o abrir un fichero HTML, visualizarlo, realizar una conversión a WML y visualizar el resultado de manera fácil y sencilla. Este programa resulta de gran ayuda para empezar a emplear el lenguaje WML si se tienen conocimientos previos de HTML y competitivo con otros programas gratuitos que realizan conversión HTML a WML y se pueden encontrar en Internet.

Referencias

[1] WAP forum "Wireless Application Protocol. White Papers". http://www.wapforum.org.

[2] World Wide Web Consortium "HTML 4.0 Specification". http://www.w3.org.

[3] WAP forum "Wireless Application Protocol. Wireless Markup Language Specification Version 1.2". http://www.wapforum.org.

[4] Judy Bishop "Java, Fundamentos de Programación". Addison-Wesley. 2ª Edición 1999. ISBN: 84-7829-022-2.

Definición de la seguridad en código fuente*

Jordi Forga Alberich
Departamento de Ingeniería Telemática. Universitat Politècnica de Catalunya.
Jordi Girona 1 y 3. Campus Nord, Mod C3, UPC. 08034 Barcelona.
Teléfono:934 015 999 Fax:934 015 981
e-mail: jforga@mat.upc.es

1 Introducción

En la actualidad JAVA, tanto a nivel de lenguaje de programación como de máquina virtual, es uno de los pocos sistemas que integran los mecanismos que permiten el desarrollo y la ejecución de aplicaciones seguras. A nivel de lenguaje, dispone de clases que permiten verificar la validez y comprobar los permisos de acceso a recursos del código que se ejecuta. En cuanto a máquina virtual, dispone de un código máquina (BYTECODE) con un repertorio simple de instrucciones orientadas a objeto que limita las operaciones posibles que se pueden realizar en tiempo de ejecución.

En JAVA, la política de seguridad se define de forma separada del código de la aplicación. El usuario o administrador de la aplicación debe definir los permisos de acceso a recursos privilegiados para las distintas clases de la aplicación. Así pues, para una correcta definición de la política de seguridad, es necesario que el administrador conozca todos los servicios requeridos por la aplicación.

En aplicaciones cliente ('applets') normalmente los recursos de sistema requeridos son pocos y por lo tanto la definición de la política de seguridad adecuada puede resultar bastante simple. A medida que estas aplicaciones se hacen más complejas, y el requerimiento de recursos privilegiados aumenta, la política de seguridad puede resultar no tan simple. En muchos casos, usuarios poco expertos acaban dando más permisos de los necesarios, sinó todos, con los problemas de seguridad que esto conlleva.

En aplicaciones servidor el problema se hace mucho más evidente debido a la cantidad de recursos privilegiados necesarios y a la complejidad de éstas. Además, en este caso, la separación entre el código y la política de seguridad parece menos adecuada ya que uno y otra están muy relacionados.

En casos en que se disponga de servidores de aplicación capaces de ofrecer servicios a otros proveedores de servicios, el administrador del sistema servidor puede no conocer la política de seguridad de cada uno de los proveedores. En estos casos se hace necesaria una mayor integración de la política de seguridad con el código.

En esta presentación se muestran las construcciones del lenguaje de programación que permiten definir los aspectos de seguridad en el propio código fuente de la aplicación. El nivel de granularidad en la definición de la seguridad es muy fino (hasta el nivel de método), permitiendo definir exactamente las clases que pueden ejecutar determinados métodos privilegiados. En el propio código de las clases se indican también las condiciones de autentificación (certificados, etc.).

*Este trabajo ha sido financiado por el proyecto de investigación "SSADE – Sistema Seguro de Acceso y Distribución Eficiente de servicios multimedia" (CICYT. TEL99-0822).

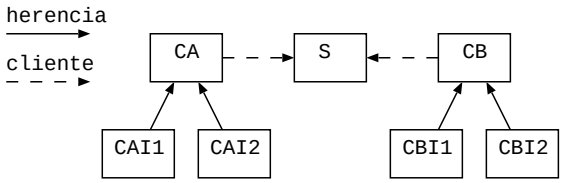
2 Aplicaciones de código distribuido

Supongamos, por ejemplo, un sistema que ofrece servicios para múltiples proveedores. Los proveedores ofrecen las distintas aplicaciones que se ejecutan en el sistema y que interaccionan finalmente con sus clientes.

El sistema ofrece únicamente determinados servicios para cada proveedor (la conexión con un determinado servidor remoto, por ejemplo). Por otro lado, cada proveedor definirá los servicios que desea ofrecer a sus clientes incluyendo sus propios criterios de seguridad.

En este esquema, en donde el código de las aplicaciones proviene de múltiples fuentes, conviene definir con detalle los servicios ofrecidos a cada proveedor. Durante el diseño y/o configuración del sistema se definen estos servicios y las autorizaciones necesarias.

Un lenguaje que permita definir métodos que se exportan únicamente a unas determinadas clases permitiría tener sistemas organizados según el siguiente esquema:



donde se definen determinados servicios (mA y mB) que son accesibles por los proveedores A y B respectivamente. Los métodos mA y mB (correspondientes a los servicios ofrecidos por el sistema) son exportados selectivamente a las clases CA y CB respectivamente. La exportación selectiva de métodos permite definir con detalle las clases cliente desde las que se permite la invocación de los métodos. Si un método es accesible desde una determinada clase cliente, también lo es

desde sus clases heredadas.

En el ejemplo de la figura, el método mA de la clase S sólo es accesible desde la clase CA y sus clases heredadas. De igual forma, mB sólo es accesible desde CB y subclases de CB.

El mecanismo de exportación selectiva de métodos nos permite definir con mucho detalle los aspectos de política de seguridad relativos a permisos o autorizaciones. Además estas autorizaciones se indican en el mismo código del servicio, evitando la necesisdad de ficheros u objetos separados.

Por otro lado es necesario definir unas condiciones en las clases clientes que permitan realizar autentificaciones del código. Estas condiciones de validación o autentificación se indican mediante invariantes de las clases cliente.

En el ejemplo, el invariante de la clase CA definiría las condiciones que deben cumplir todos los objetos creados con esta clase o sus subclases. Entre estas condiciones podrían estar, por ejemplo, las comprobaciones de los certificados con que se ha firmado el código de la clase, origen del código, etc.

El código de las clases servidora S y cliente CA,CB (que podrían ser abstractas) se obtiene como resultado del desarrollo del sistema servidor, y podría ser la parte residente del sistema. Durante este desarrollo se definen los servicios ofrecidos por S y se especifican los permisos y autentificaciones necesarias para el acceso a estos servicios, todo en el mismo diseño.

El código de las aplicaciones ofrecidas por los proveedores A y B serían implementaciones (CAI1,CAI2,CBI1,CBI2) que heredarían las clases cliente CA,CB y por lo tanto tendrían acceso únicamente a los servicios indicados. El método mA sería accesible por CAI1 y CAI2 y el método mB por CBI1 y CBI2.

3 Construcciones propuestas

En el apartado anterior se introducen los dos mecanismos del lenguaje de programación necesarios para la especificación de la política de seguridad en el código fuente: la exportación selectiva de método y los invariantes de clase.

En el Departamento de Ingeniería Telemática de la UPC se ha desarrollado un lenguaje de programación concurrente y orientado a objeto en el que se incluyen estos mecanismos. En cuanto a los aspectos de concurrencia, se trata de una evolución del lenguaje PADD (Paralellism and Abstraction in Dimensional Design)[1] con construcciones de paralelismo jerárquico, mecanismos de paso de mensajes y especificación de precondiciones y postcondiciones[2]. Los aspectos orientados a objeto son similares a los de Eiffel, creado por Bertran Meyer [4].

La clase S del ejemplo del apartado anterior seguiría el siguiente esquema:

```
class S {
  export CA {
    void mA() {
      /* Método accesible desde CA */
    }
  }
  export CB {
    void mB() {
      /* Método accesible desde CB */
    }
  }
  export Nil {
    void mP() {
      /* Métodoño accesible
       * desde clases cliente */
    }
  }
}
```

en donde mediante la construcción **export** se indican cada uno de los métodos que son exportados selectivamente a las clases CA y CB.

La inclusión de invariantes puede verse en el esquema de la clase CA:

```
abstract class CA {
  export Nil {
    void m() {
      /* Método sólo accesible
       * desde subclases de CA */
    }
  }
  invariant {
    /* Comprobación de certificados,
     * origen del código
     * u otras condiciones */
  }  }
```

Referencias

[1] M.Bertran, J.Forga, F.Oller, J.A.Frau. "Integrated Simulation and Design of communication Systems in a PADD Environment". IEEE CAMAD-92. Montevell (Canada), Set-Oct 1992.

[2] M.Bertran, F.Oller, J.Forga. "A Design Environment with Simulation and Formal Verification". IEEE CAMAD-94. Princeton (USA), Apr-1994.

[3] Li Gong. "Java Security Architecture (JDK1.2)" October 1998. Sun Microsystems, Inc.

[4] B. Meyer. "Object Oriented Software construction". 2nd Edition. Prentice Hall PTR, 1997.

[5] M.Pistoia, D.F.Reller, D.Gupta, M.Nagnur, A.K.Ramani. "Java 2 Network Security". 2nd Edition. Prentice Hall PTR.

Propuesta de una arquitectura para una futura Red de acceso radio basada en IP

Tomás Robles[1], Alberto López[2], Héctor Velayos[3]

[1]Departamento de Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid
ETSI Telecomunicación, Ciudad Universitaria – 28040 Madrid
Teléfono: 91 336 7332, Fax: 91 336 7333
E-mail: robles@dit.upm.es

[2]Departamento de Ingeniería de la Información y las Comunicaciones, Universidad de Murcia,
Facultad de Informática, Campus Universitario de Espinardo - 30071 Murcia
Teléfono 968 364607, Fax: 968 364151
E-mail: alberto@dif.um.es

[3]Agora Systems S.A.
C/ Aravaca nº 12 3ºB - 28040 Madrid
Teléfono: 91 533 5857, Fax: fax: 91 534 8477
E-mail: hvelayos@agora-systems.com

1 Introducción

Las tecnologías móviles están revolucionando el mercado de las telecomunicaciones. Desde los primeros sistemas analógicos de los 80, pasando por los sistemas digitales de 2ª generación (GSM, IS-95, IS-54, ANSIU-54, ANSI-136 y PDC), la demanda de servicio por parte de los clientes ha superado continuamente las previsiones más optimistas. Sin embargo, incluso recientemente, los servicios proporcionados han estado restringidos a comunicaciones de voz y servicios de bajo ancho de banda. Al mismo tiempo, la expansión de los servicios y aplicaciones multimedia en Internet, se ha centrado principalmente en las redes fijas.

En ese contexto BRAIN (Broadband Radio Access for IP based Networks) [1] pretende proporcionar un sistema que sirva de base para los sistemas a desarrollar después de la 3G. El sistema propuesto, está basado en la tecnología de Internet (IP), para lo cual ha sido necesario dotar a la arquitectura propuesta de soporte de Calidad de Servicio (Quality of Service, o simplemente QoS) y movilidad como principales características en el entorno de aplicación. La red de acceso del BRAIN será parte de una red de comunicaciones completamente basada en IP. La Figura 1 describe el sistema completo, mostrando la integración de tecnologías existentes y emergentes, incluyendo la convergencia de redes fijas y móviles mostrando además un posible camino de evolución hacía los futuros sistemas de cuarta generación. El núcleo de la Red IP interconecta redes de acceso de diferentes tipos[2], tanto fijas como inalámbricas.

La movilidad de terminales y usuarios se soportará a los niveles micro y macro mediante el uso de mecanismos de hand-over1 que permitan los movimientos tanto dentro de la misma red de acceso, como el cambio a redes de acceso de diferente tecnología. De esta forma un usuario será capaz de acceder a sus servicios habituales con el mismo interfaz, mientras se desplaza entre las áreas de cobertura de diferentes redes y hace uso de diferentes tecnologías de acceso inalámbrico. El usuario no necesita preocuparse de las tecnologías de red subyacentes. El proyecto BRAIN contribuye a esta visión centrándose principalmente en:

1. El diseño de una red de acceso basada en IP que soportará tecnologías móviles no celulares, añadiendo funcionalidad para complementar a los sistemas 3G.
2. Soportar la provisión de servicios de forma transparente – proporcionando adaptación de Calidad de Servicio (QoS de forma abreviada) en el caso de deterioro de la señal de radio o la disponibilidad de bajo ancho de banda durante el hand-over.
3. Definir los requisitos de un interfaz radio (10Mbits/s) para la cobertura de zonas específicas mediante pico-células, proponiendo las modificaciones y mejoras necesarias para a la norma HIPERLAN/2.

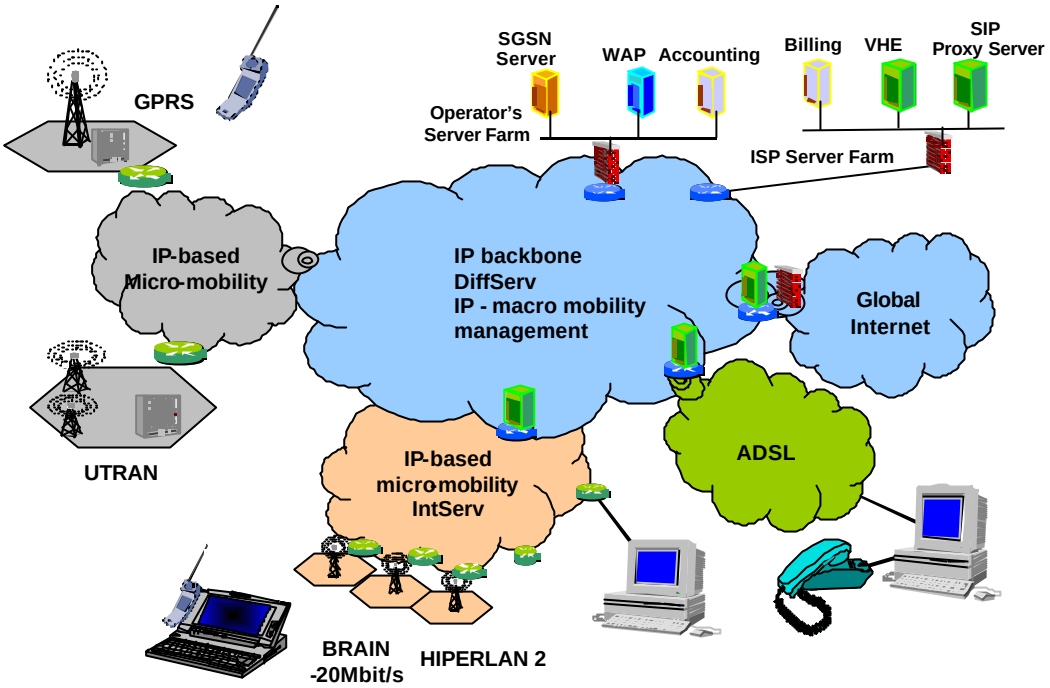
2 Arquitectura de la Red

Uno de los principales objetivos de BRAIN es el diseño de una plataforma que proporcione soporte para QoS, gestión de la movilidad y mecanismos de adaptación para diferentes tipos de aplicaciones. El resultado de este trabajo es BRENTA (BRAIN End Terminal Architectue [4].

La arquitectura BRENTA opera en dos planos principales; el (usual) plano de red, y el plano de gestión de QoS y de recursos. El aspecto de gestión de QoS es opcional, ya que mientras unas aplicaciones y servicios específicos (aplicaciones de tipo D) van a demandar estos servicios, otras aplicaciones gestionarán QoS por ellas mismas.

[1] Entendemos por hand-over el cambio de estación radio con la cual se comunica con el termina de usuario.

Figura 1: La visión del BRAIN de un sistema de comunicaciones totalmente basado en IP



Las aplicaciones y servicios en BRENTA usaran un interfaz de QoS – y movilidad - organizado en diversos interfaces, cada uno de ellos específico para un grupo de aplicaciones.

Las aplicaciones las vamos a clasificar en cuatro grupos (A,B,C y D), según que tipo de interfaz utilicen. Las aplicaciones tipo A utilizan un interfaz típico de IP, mientras que las de tipo D delegan casi todas las funciones en los servicios del terminal.

Los problemas principales en la capa de red son las interacciones entre la Calidad de Servicio (QoS) y la movilidad; la adaptación de las aplicaciones y los protocolos a la gran variedad de interfaces de radio con QoS variable; y la unificación del conjunto tan dispar de protocolos de Internet para formar una red móvil coherente. Los requisitos para una red de acceso en el proyecto BRAIN pueden expresarse de manera simple de la siguiente manera [3]:

> ***El objetivo básico de la Red de Acceso BRAIN es hacer que el acceso móvil a la Internet aparezca como 'normal' a través de la infraestructura inalámbrica.***

En el diseño de la nueva Red se han aplicado los siguientes principios básicos:

- El Principio del Extremo a Extremo y la Transparencia

  En el contexto de una red de acceso móvil este principio puede expresarse mediante los siguientes requisitos: ser independiente de las capas de transporte y aplicación específicas; ser tan independiente como sea posible del tipo de paquetes que están siendo transportados. Y minimizar el número de funciones especiales que proporciona la red de acceso.
- Seguir el modelo en Capas.
- Maximizar la flexibilidad y la evolución futura.
- Minimizar los requisitos del terminal.
- No reinventar la rueda
- Explotar la Funcionalidad Estándar de la Red Troncal IP.
- Mantenerlo simple

Agradecimientos

This work has been performed in the framework of the IST project IST-1999-10050 BRAIN, which is partly funded by the European Union. The authors would like to acknowledge the contributions of their colleagues.

Alberto López agradece en particular a la Fundación Séneca (Comunidad Autónoma de la Región de Murcia) su apoyo y la financiación para llevar acabo este trabajo.

Referencias

[1] BRAIN IST PROJECT: http://www.ist-brain.org/

[2] D.R. Wisely, W. Mohr & J. Urban "Broadband Radio Access for IP-Based Networks (BRAIN)",IST Mobile Summit, Galway, October 2000.

[3] P. Eardley, R. Hancock, "Modular IP Architectures for Wireless Mobile Access" 1st International Workshop on Broadband Radio Access for IP-Based Networks (BRAIN), London, November 2000.

[4] BRAIN deliverable D1.2. Concepts for service adaptation, scalability and QoS handling on mobility enabled networks. March 2001.

Definición de Políticas de Seguridad Homogéneas en IPsec

Antonio F. Gómez Skarmeta, Gregorio Martínez Pérez
Dpto. de Ingeniería de la Información y las Comunicaciones
Universidad de Murcia
Facultad de Informática. Campus de Espinardo, s/n. 30.071 Murcia
Teléfono: 968 364607 Fax: 968 364151
E-mail: {skarmeta, gregorio}@dif.um.es

1 Introducción

Uno de los temas que cuenta actualmente con un mayor interés en el mundo de la investigación, en lo que a sistemas distribuidos de comunicaciones se refiere, es el relacionado con la definición de entornos consistentes que permitan su gestión de manera sencilla y segura. No en vano, uno de los principales inconvenientes que presentan los sistemas distribuidos es la exigencia de gran cantidad de recursos humanos y de computación para poder mantener una cierta consistencia y homogeneidad en su comportamiento, sobre todo cuando las responsabilidades de administración no están bien repartidas.

Para abordar esta situación de una forma simple, y a la vez elegante, uno de los conceptos más importantes que se ha definido en este contexto es el de dominio administrativo. Un dominio es una colección de sistemas agrupados explícitamente por el administrador para un propósito específico, permitiendo delimitar las responsabilidades de gestión a llevar a cabo dentro de dicho dominio desde el punto de vista de las acciones de seguridad, movilidad, etc. Con ello conseguimos un método flexible que permite evitar los inconvenientes comentados anteriormente, al tiempo que proporciona el marco ideal para especificar el comportamiento del sistema a través de un conjunto de reglas, conocido formalmente como política del sistema.

Dichas políticas van a permitir, por lo tanto, controlar el comportamiento del sistema distribuido de comunicaciones dentro de un determinado dominio lógico a la vez que juegan un papel importante a la hora de aportar homogeneidad y definiciones exentas de inconsistencias dentro de cada dominio administrativo.

2 El Entorno Desarrollado

En este sentido, y tomando como base el lenguaje SPSL [1], pero sin olvidar con ello los demás elementos que conforman una arquitectura de políticas, hemos diseñado y puesto en marcha un sistema escalable de gestión de seguridad a nivel IP basado en políticas.

Una de las cuestiones más interesantes del diseño de nuestro sistema es que se ha basado completamente en WEB, con lo que conseguimos que tenga un gran nivel de accesibilidad y que sea intuitivo, seguro y sobre todo muy fácil de utilizar. En este sentido, destacar la existencia de un servidor WEB Apache seguro que hace las veces de punto de acceso a todo el sistema de manejo de políticas y que es el que de manera directa gestiona el acceso al servidor de políticas, mediante una serie de servlets Java diseñados a tal efecto.

En lo referente a seguridad, se ha planteado la utilización del protocolo HTTP junto con SSL para todas las comunicaciones entre los administradores y el servidor de políticas, con lo que conseguimos ya de partida, confidencialidad, integridad y autenticación de servidor.

Estas medidas de seguridad se ven complementadas con una autenticación de cliente y por un control de accesos. La primera de ellas se consigue mediante certificados X.509v3 emitidos a cada administrador del sistema por parte de una Infraestructura de Clave Pública, o PKI, que se constituye en la base de confianza de todo el sistema. Dicho certificado, junto con su clave privada asociada quedan almacenados dentro de una tarjeta inteligente RSA que se entrega a cada administrador, y que hace las veces de "llave de acceso" a todo el sistema. Este tipo de tarjetas están caracterizas por el hecho de que las claves privadas normalmente son generadas por un coprocesador del cual dispone la propia tarjeta, y todas las operaciones que utilizan dicha clave privada (firmado y descifrado) se realizan también dentro de la tarjeta, con lo que la clave privada nunca sale de este dispositivo.

En lo referente al control de accesos, éste es realizado en base a la información contenida en el campo DN (Distinguised Name) que tienen los certificados que presentan los administradores al servidor en la fase de negociación del protocolo SSL. Una vez realizada esta fase, un servlet Java, otorga o deniega finalmente el acceso al servidor de políticas. Con ello se consigue que sólo los administradores acreditados previamente, puedan acceder a la base de datos de políticas para modificar su contenido.

Una vez realizados los pasos de autenticación y control de acceso, el administrador tiene la posibilidad de decidir si desea crear una nueva política (a través de un conjunto de reglas de política), modificar los parámetros de alguna de las

políticas existentes o eliminar alguna que él haya creado o de la cual sea responsable.

Para entender un poco mejor como funciona realmente el sistema, nos vamos a centrar un poco más en las fases, concretamente cinco, que conforman la operación de creación.

La primera de estas fases, consiste en la definición de la información general asociada a la política, a saber, entidad a la que va destinada esta política (ya sea, un equipo final, un gateway seguro o un dominio de seguridad) el máximo tiempo que la política puede estar en caché, comentarios generales, etc. Destacar el hecho de que el nombre de la política será asignado por el servidor con el objetivo de evitar inconsistencias de nombrado dentro de cada dominio administrativo.

La segunda fase, permite definir las condiciones de una regla de política, es decir, permite indicar cuestiones como la lista o los rangos de direcciones IP (con o sin máscara) o puertos origen y destino, protocolos de transporte y dirección del tráfico, propiedades que serán las utilizadas para determinar si a una comunicación se le debe de aplicar esta regla o no.

Una vez definidas las condiciones, necesitamos indicar los parámetros de las acciones que se deben de ejecutar cuando éstas se cumplan. Las acciones que se pueden realizar son las de aplicar seguridad (protocolos AH y/o ESP) y las de aplicar compresión al paquete IP. Para cada una de ellas existe una relación de algoritmos y modos de ejecución (túnel o transporte) que se deben de elegir.

Una vez finalizada esta tercera fase, habremos definido una nueva regla de política. El sistema nos indica entonces dos opciones: volver al paso 2 para definir nuevas reglas (recordemos que una política es un conjunto de una o más reglas) o la de dar por finalizado el proceso de definición de reglas y pasar a la cuarta fase. Esta cuarta fase, permite que el servidor valide las reglas creadas en busca de inconsistencias entre ellas. En caso de existir, estas inconsistencias le serán indicadas al administrador que tendrá la posibilidad de modificar los errores cometidos.

En el caso de que el proceso de validación sea correcto pasamos a la quinta y última fase, la de generación de una especificación neutral de la política en XML y posterior firmado digital de la misma. La generación se produce con respecto a un DTD (Document Type Definition) creado por parte de los diseñadores de esta arquitectura en base a la especificación de SPSL. Una vez creado el documento XML por parte del servidor, éste lo envía al navegador Web del administrador para que éste lo verifique y posteriormente proceda a firmarlo digitalmente usando la clave privada RSA almacenada en su tarjeta inteligente. La firma generada se introduce en el documento XML de forma permanente, aportándole información sobre la entidad firmante, integridad y no repudio de origen. El formato de esta firma digital se corresponde con la última versión de los estándares especificados por el W3C. Una vez firmado el documento XML que contiene la política, éste es almacenado en el servidor de políticas del dominio administrativo.

Una vez definidas, las políticas pueden ser consultadas por los procesos asociados a los interfaces que implementan IPsec en lo equipos finales de una comunicación o en los gateways seguros que existen en su camino. Para ello se emplea un protocolo definido por el IETF y que se conoce como SPP [2] (Security Policy Protocol). Este protocolo permite, con independencia de cualquier sistema de gestión de claves (como IKE) y suite criptográfica, descubrir servidores de políticas, distribuir y negociar políticas de seguridad y resolver conflictos, sobre todo cuando intentamos comunicar equipos que se encuentran en dominios administrativos distintos.

En concreto, cuando un equipo final o un gateway seguro se encuentra con un tráfico IP de entrada o de salida del cual desconoce cual debe de ser el tratamiento de seguridad que le debe de aplicar, procederá a realizar una consulta SPP al servidor de políticas de su dominio. Éste procederá a consultar internamente en su base de datos aquella política (o políticas) que se ajustan a las características de la comunicación que ha aportado la entidad de red a la hora de hacer su consulta. Si el servidor no la encuentra intentará propagar la consulta al servidor del dominio administrativo del nodo destino, o a cualquiera de los que se encuentran en el camino entre los dos equipos.

7 Conclusiones

En este artículo se ha presentado un sistema capaz de gestionar, de forma distribuida y segura, políticas de seguridad IP para un dominio administrativo de comunicaciones.

Agradecimientos

El desarrollo de este sistema escalable de gestión de seguridad en IP basado en políticas ha sido parcialmente soportado por el proyecto de investigación PISCIS (TEL-1FD97-1426).

Referencias

[1] M. N. Condell et al. “Security Policy Specification Language”. IP Security Policy Working Group. Internet Draft. Marzo 2000

[2] L. A. Sanchez, M. N. Condell. “Security Policy Protocol”. IP Security Policy Working Group. Internet Draft. Julio 2000

SISTEMA DE FACTURACIÓN ELECTRÓNICA BASADO EN EDI/XML

Marivi Higuero, Juanjo Unzilla, Asier Murciego, Eva Ibarrola
Departamento de Electrónica y Telecomunicaciones
Universidad del País Vasco / Euskal Herriko Unibertsitatea
Alda. Urquijo s/n. 48013 – Bilbao
Teléfono: 94 601 42 07. Fax: 94 601 42 59
Email: jtphiapm@bi.ehu.es, jtpungaj@bi.ehu.es, jtbmuala@aintel.bi.ehu.es, jtpibara@bi.ehu.es

***Abstract**: this paper presents a flexible solution for data interchange based in XML, focused in billing systems. Data can be retrieved from different types of databases thanks to JDBC abstraction layer which is used in conjunction with a XML template for document generation. These documents are validated against a DTD to check that they are well-formed, and then encrypted before being sent through the Internet attached to electronic mail messages. On the other side, they are decrypted and, after checking the XML document, the database is updated with the new information.*

1 Introducción

Internet ha revolucionado el mercado internacional al poner a disposición de cualquiera la información necesaria para desempeñar diversas actividades en las empresas. El Intercambio Electrónico de Datos (Electronic Data Interchange o EDI) permite realizar transacciones comerciales traducidas en mensajes estandarizados e implementados por guías de referencia avaladas por las Naciones Unidas.

El gran avance experimentado por estas tecnologías en los últimos años, ha permitido el desarrollo de modelos de sintaxis para intercambio de información más sofisticados, como es el caso de XML [1]. Actualmente existen varias iniciativas en marcha a nivel mundial relacionadas con XML, así como proyectos en el campo de EDI/XML [2].

Este artículo presenta un sistema de facturación electrónica basado en XML que permite la interacción entre sistemas EDI tradicionales a través de Internet.

2 Sistemas de comercio EDI-XML

Una de las mayores ventajas de los sistemas EDI es su implantación: las empresas confían en estos sistemas tanto como para procesar información crítica para su propio funcionamiento, como son las órdenes de compra o las facturas.

El sistema de transporte de EDI es bastante rígido y simple, pero su implantación supone un elevado coste debido a la complejidad derivada del formato críptico y comprimido de sus mensajes. XML supera este hecho incluyendo los denominados metadatos dentro de los mensajes de datos [3], con lo que simplifica la transferencia de la información, permitiendo extender las ventajas de EDI a las pequeñas y medianas empresas. Para la compatibilidad entre ambas tecnologías, las empresas pueden ampliar su base de comercio electrónico instalando traductores XML-EDI en sus servidores web.

Básicamente, XML/EDI está formado por cinco elementos principales [4], que son: XML, EDI, las plantillas y los agentes que las interpretan y, por último, los depósitos (repository), que suministran principios semánticos para búsquedas y transacciones. Estos componentes se integran en una arquitectura basada en capas que definen el esquema de trabajo de XML/EDI.

3. Diseño del sistema de intercambio de facturas EDI basado en XML

El diseño del sistema puede dividirse en distintos módulos que se observan en la figura 1, donde se representa el esquema funcional del sistema a la hora de realizar el envío de una factura:

- Recuperación de la información de facturación de la base de datos empresarial,
- Creación de un documento XML a partir de los datos obtenidos en la fase anterior, de acuerdo con un DTD que determinará la estructura de la factura,
- Validación de la factura frente al DTD para comprobar su correcta sintaxis y estructura,
- Envío de la factura al destinatario correspondiente (información que puede haber sido extraída en la primera fase de la base de datos).

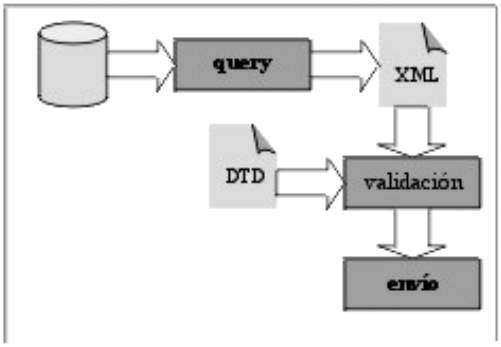


Fig. 1: Diagrama funcional del envío

Al de recibir una factura, el proceso es el inverso: se determina la veracidad de la factura, validándola frente al DTD común y se pasa a actualizar el sistema de facturación con los datos recibidos.

3.1 Módulo de control

Se trata del módulo principal del sistema y se encarga de controlar el resto de módulos, secuenciando y supervisando su funcionamiento, así como de la configuración del mismo.

3.2 Módulo de consulta

En módulo se lanzan las consultas necesarias a la base de datos de la empresa y se formatean los datos obtenidos de las mismas para conformar el documento XML. Para ello se utilizan una serie de plantillas, adaptables en función del cliente, que determinan la estructura de los datos.

3.3 Módulo de validación

Este módulo comprueba que todos los documentos XML entrantes y salientes del sistema son correctos, mediante un validador XML basado en las implementaciones actuales de DOM versión 2, frente a un DTD que, como ya se ha indicado, puede ser diferente dependiendo de la empresa.

3.4 Módulo de intercambio

El flujo de datos en el sistema puede ser bidireccional (entrante y saliente), por lo que puede funcionar de dos modos distintos:

- En **envío**, recoge la transacción en formato XML y la envía a una dirección de correo electrónico indicada en la factura y/o recuperada de la base de datos de clientes de la empresa, como documento adjunto, encriptado utilizando PGP.

- En **recepción**, extrae la factura del correo y la desencripta, pasando el documento XML al módulo central.

Dentro de este módulo, la seguridad está proporcionada por un sistema basado en PGP, del cual se ha implementado un servidor de claves dentro del Grupo de Ingeniería Telemática. El sistema permite la utilización de otros sistemas de mensajería segura como S/MIME.

4. Prototipo implementado

El prototipo está basado en un sistema Windows2000 Server con una base de datos de facturación sobre SQL Server 6.5sp2. En el desarrollo se ha utilizado software de libre distribución, a partir del código fuente, para poder adaptarlo a las necesidades del sistema. El prototipo se ha realizado completamente en lenguaje Java, mediante la versión más reciente del kit de desarrollo de Sun (1.3.0 release 2), y aunque cabe la posibilidad de que el sistema funcione bajo UNIX/LINUX, la elección de Java permite abstraernos de la plataforma a utilizar.

5 Conclusiones

En este artículo se ha presentado una propuesta para crear un sistema de facturación ágil, sencillo, seguro y extensible que pueda ser adaptado a diferentes compañías de forma simple. Para ello se utiliza XML como soporte para los datos, lo que nos permite extender y adaptar el sistema.

Su diseño en base a módulos, y sobre todo, a plantillas fácilmente editables, permiten extender sus posibilidades y actualizar el sistema con nuevos estándares aún por definir, como UN/CEFACT, facilitando su mantenimiento y duración.

Agradecimientos

El trabajo presentado ha sido realizado gracias a la ayuda recibida del Gobierno Vasco dentro del proyecto OD00UN57, y a la colaboración de la empresa SARENET S.A.

Referencias

[1] Tim Bray, Jean Paoli, C.M. Sperberg-McQueen, Eve Maler. "Extensible Markup Language (XML) 1.0 (Second Edition)". http://www.w3.org/TR/2000/REC-xml

[2] David RR Webber. "Introducing XML/EDI Frameworks". Electronic Markets vol.8 no.1, 1998.

[3] Sterling Commerce. "XML y el comercio electrónico: una nueva revolución en el horizonte". Revista “e.comm”. Septiembre 2000, p104-109.

[4] Duane Nickull, Brian Eisenberg. "ebXML Technical Architecture Specification" http://www.ebxml.org/specdrafts/ebXML_TA_v1.04.pdf

Diseño y Evaluación de un Sistema de Telemedicina para Entornos Rurales

José García, Emiliano Bernués, Julián Fernández y Carlos Cajal
Departamento de Ingeniería Electrónica y Comunicaciones
María de Luna, 3. Universidad de Zaragoza. 50015 Zaragoza
Teléfono: 976761962 Fax: 976762111
E-mail: jogarmo@posta.unizar.es

***Abstract**. This work presents the design and evaluation of a Telemedicine system which provides telecommunications facilities for doctors in small-sized. The system permits to share applications between doctors located in different locations. The evaluation of the system showed that transmission of audio and video in real-time without important bandwidth requirements is possible.*

1 Introducción

La Telemedicina ha experimentado durante los últimos años un gran avance debido al espectacular desarrollo de las tecnologías de red que le sirven como soporte. Uno de los entornos en que puede proporcionar mayores logros es el de las zonas rurales, donde no es factible disponer de especialistas en los centros de salud y puede recurrirse a sistemas de teleconsulta entre diferentes médicos [1-2].

El objetivo de este trabajo consiste en desarrollar un sistema que permita la comunicación entre el médico del centro rural y el especialista de un hospital central, para facilitar el acceso de los pacientes en zonas rurales a las consultas de especialistas.

2 Descripción del sistema

El sistema diseñado establece la comunicación a través de una red IP (ver Fig. 1) entre un hospital rural y el centro remoto permitiendo la transferencia de datos de pacientes, pruebas médicas, etc.

La aplicación se ha desarrollado en Visual C++ (clases MFC) y utilizando los componentes de desarrollo de NetMeeeting SDK [3] (basado en las normas H.323, para conferencia, y T.120, para conexiones de datos IP multipunto) y las librerías FreeImage, que permiten el manejo de imágenes. La estructura general del sistema propuesto se divide en los siguientes bloques: interfaz gráfico, gestor de imágenes, gestor de informes, notificación de eventos y conferencia.

Interfaz Gráfico: las funciones de la aplicación son accesibles desde la pantalla principal (ver Fig. 2), dividida en varias zonas.

Gestión de Imágenes: mediante la utilización del asistente AppWizard se implementó una aplicación MDI (interfaz de documentos múltiples) para visualizar varias imágenes simultáneamente, mostrar imágenes representadas a diferente escala, etc.

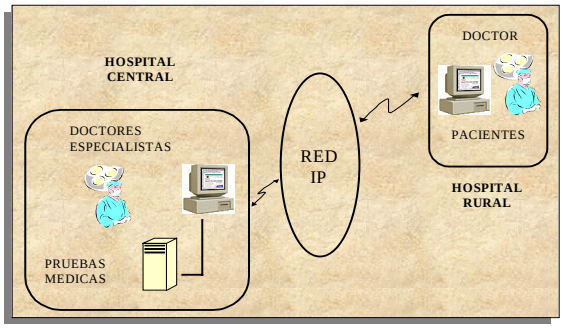


Fig. 1. Entorno de utilización del sistema.

Gestión de Informes: una sencilla aplicación de archivos de texto permite escribir, visualizar e incluso modificar los informes médicos. Incluye funciones de lectura y almacenamiento de archivos en modo texto.

Notificación de Eventos: se incluye una ventana de notificación donde queda constancia de los eventos de relevancia (conexiones y desconexiones tanto locales como remotas) que se han producido durante la conexión. que se vayan produciendo.

Conferencia: Se efectúa el control de llamada (estableciendo la dirección IP asociada), de vídeo y audio, de transmisión de ficheros y de visualización remota de aplicaciones.

3 Evaluación del sistema.

Se han realizado pruebas de comunicación entre dos PCs conectados mediante dos *routers* Nucleox+ WAN-LAN bajo una conexión Frame Relay para evaluar su funcionamiento. Este escenario permite variar el CIR de los PVCs configurados para simular diferentes capacidades de red donde evaluar la aplicación: 16Kbps (en lo que podría simular una conexión a través de Internet), 64Kbps (a través de un único canal RDSI), y 128Kbps (mediante la utilización de los dos canales del acceso básico RDSI o un acceso Frame Relay).

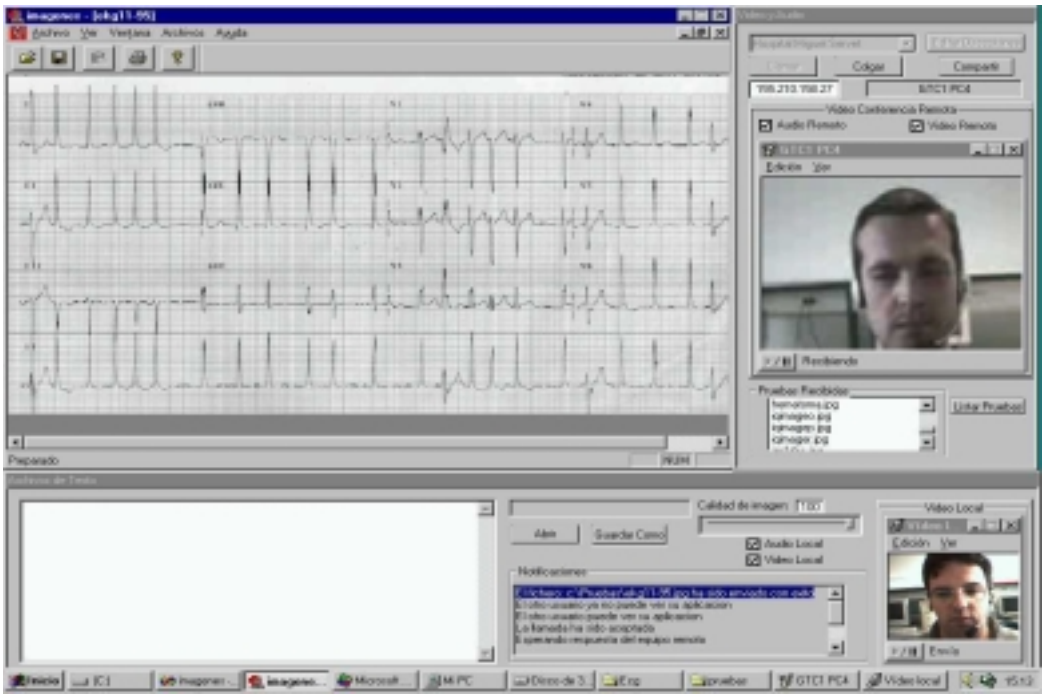


Fig. 2. Vista general de la aplicación.

Los resultados obtenidos se muestran en Tab. 1, donde se representa de forma cualitativa la recepción de señales de audio, vídeo, y la utilización de aplicaciones compartidas bajo diferentes tasas de transmisión. La utilización de los dos canales de un acceso básico RDSI (128Kbps) parece una solución idónea para establecer este tipo de conexiones.

Tabla 1. Resultados de la evaluación del sistema.

	Audio	**Vídeo**	**Aplicaciones**
16Kbps	Inaceptable	Inaceptable	Inaceptable
64Kbps	Buena	Aceptable	Aceptable
128Kbps	Buena	Buena	Buena

4 Discusión

El diseño realizado consta fundamentalmente de una estructura sobre la cual se pueden añadir en el futuro aplicaciones específicamente implementadas de telediagnóstico, procesado de señales biomédicas, etc. Se ha pretendido que el diseño sea sencillo de usar y amigable, teniendo presente que el usuario final de este tipo de servicios es personal médico.

Se ha comprobado que la comunicación se realiza adecuadamente para conexiones a través de canales de 128Kbps. La utilización del sistema con anchos de banda superiores permitirá mejorar la calidad en la utilización de aplicaciones compartidas además de permitir enviar/recibir señales registradas en tiempo real.

5 Conclusiones

Se ha desarrollado una aplicación que puede facilitar la realización de las consultas médicas a especialistas en zonas de poca densidad de población. La aplicación consigue transmitir audio y vídeo en tiempo real sin necesidad de elevados anchos de banda. Puede servir como base para desarrollos posteriores soportados sobre otro tipo de redes como UMTS en comunicaciones tipo ambulancia-hospital.

Agradecimientos

Este trabajo ha sido financiado en parte por los proyectos Fondos Europeos de Desarrollo (FEDER) 2FD97-1070 y 2FD97-1197-C02-01 y de la Diputación General de Aragón (DGA) CONSI+D P40/98.

Referencias

[1] J. Zhang, J.N. Stahl, H.K. Huang, X. Zhou, S.L. Lou, K.S. Song. “Real-time teleconsultation with high-resolution and large-volume medical images for collaborative healthcare”. IEEE Trans Inf Technol Biomed, vol. 4, no. 2, pp. 178-85, 2000.

[2] A. Hassol, G. Gaumer, C. Irvin, J. Grigsby, C. Mintzer, D. Puskin. “Rural telemedicine data/image transfer methods and purposes of interactive video sessions”. Journal of the American Medical Informatics Association, vol. 4, no. 1, pp. 36-37, 1997.

[3] “NetMeeting SDK ver. 2.1”. Documentación digital.

Entorno de Gestión Distribuida para la Monitorización y Correlación de Datos Oceanográficos.

Enric Trullols Farreny[1], Carlos Samitier Otero[2], Antoni Barba Martí[3], Rafael Morillas Varón[3]
[1]Departament de Matemàtica Aplicada IV. [2]DIMAT SA.[3]Departament d'Enginyeria Telemàtica.
Universitat Politècnica de Catalunya.
{enric, telabm, morillas}@mat.upc.es,[2]cso@dimat.es

Abstract. *This paper presents a project for the evolution from the present oceanographic data acquisition systems to a new platform, to be implemented in the Spanish oceanographic ships in the next 2-3 years. The main goal of this project is the development of a technology that allows a virtual distributed research, with Web accessibility. The project is based on a number of existing technologies such as the Java object technology, the CORBA (Common Object Request Broker Architecture) and the mobile agent technology. The applications distributed in the ship's Ethernet network will be accessible from any executed application in a server, as if they would be physically situated in the same machine. The mobile agents' technology allows the applications to move through the network and to be executed from any place. In addition it will be possible to correlate data from different experiments or complement one experiment with other information gathered from the local environment or any other remote site.*

Key words: *Distributed instrumentation system. Oceanographic instrumentation. CORBA, Java, Mobile agents.*

1 Introducción

La evolución tecnológica de la investigación oceanográfica está fundamentalmente ligada a la capacidad de los equipos de medida y a los procesos asociados a ella. Sin embargo, se pueden introducir otros factores que definan nuevos escenarios de trabajo. Compartir los recursos y eliminar los obstáculos introducidos por las redes de comunicación puede ser la clave que abra las puertas a nuevos conceptos en la adquisición y la gestión de datos oceanográficos.

En este artículo presentamos una aplicación distribuida que actúa como plataforma en los sistemas de adquisición, gestión y monitorización de datos oceanográficos. La aplicación será probada e implementada en los buques oceanográficos españoles en los próximos 2 ó 3 años. El sistema está orientado hacia los objetos distribuidos y contará con una interfaz para acceder a estos, que estarán distribuidos en una red VPN (*Virtual Private Network*).

La presentación y el acceso a los datos estará basado en tecnología Web. Gracias a esto, será posible acceder a las herramientas para realizar los experimentos usando cualquier tipo de navegador y por lo tanto integrar diferentes plataformas de investigación oceanográfica en una intranet común, accesible desde cualquier lugar.

2 Tecnologías involucradas en la aplicación

Actualmente, los sistemas de adquisición de datos oceanográficos, instalados en los buques oceanográficos españoles, están basados en un conjunto de sensores y unidades de medida controladas por PC's e interconectados mediante una red Ethernet en tiempo real, *Oceanographic Data Adquisition System* (SADO) [1]. El software de control de este sistema es orientado a objetos. Existen unas clases genéricas de datos, que modelan los sensores y los instrumentos usados en los distintos buques oceanográficos. Los datos adquiridos son almacenados en uno o más servidores dependiendo de la configuración del experimento. Los datos pueden ser consultados localmente mediante la red Ethernet o de forma remota mediante comunicación vía satélite usando protocolo TCP/IP.

Nuestro proyecto desarrolla SADO desde su estado actual hacia una plataforma de objetos y servicios distribuidos. Los datos se obtendrán de los distintos dispositivos físicos y serán ofrecidos como servicios a los diferentes experimentos. También se contempla la posibilidad de desarrollar aplicaciones inteligentes.

Las tecnologías básicas utilizadas son:

- La tecnología de programación de objetos, Java, que permite definir aplicaciones cliente/servidor que pueden residir en cualquier lugar en la red.

- La tecnología de objetos distribuidos, CORBA, que oculta la plataforma de objetos de las aplicaciones. De esta manera las aplicaciones serán transparentes al sistema real, es decir, independientes del tipo de software utilizado.

- La tecnología Web, que junto con Java permite desarrollar interfaces de usuario que tienen la característica de poder ser ejecutadas desde cualquier equipo conectado a Internet y procesar información desde cualquier lugar con conectividad HTTP.

- La tecnología de Agentes Móviles, que permite definir aplicaciones que se muevan a través de la red y se ejecuten desde cualquier lugar apropiado para ofrecer el servicio final.

En una primera fase del proyecto migraremos la actual estructura SADO hacia una aplicación de objetos distribuidos basada en CORBA. Gracias al hecho de que los objetos usados en los buques oceanográficos españoles han sido programados en C++, será fácil migrar los objetos situados en diferentes PC's que ahora controlan las medidas y los sistemas de adquisición. En esta primera fase será necesario recompilar los objetos e incluirlos en la interfaz normalizada IDL (*Interface Definition Language*) [2].

Con la utilización de una red Intranet, basada en Internet2, en la aplicación de gestión tendremos disponibles múltiples niveles de servicio, por lo cual deberemos implementar mecanismos de control de recursos y medidas de seguridad.

También se aprovechan los mecanismos de asignación de recursos que proporcionan los Servicios Diferenciados [3], [4] para trabajar con requerimientos de aplicaciones heterogéneas y diferente QoS [5]. Los Servicios Diferenciados permitirán una gestión sobre el acceso a la red y en el núcleo de la misma. Por lo tanto, será posible ofrecer a los usuarios un *Services Level Agreement* (SLA), considerando los distintos perfiles de tráfico (Internet, Web, aplicaciones críticas para el funcionamiento, voz sobre IP, multimedia, etc.).

La aplicación de gestión utilizará como uno de los principales protocolos, el *Multiprotocol Label Switching* (MPLS) que facilita el encaminamiento. Dicho protocolo permitirá métodos de gestión de tráfico, encaminamiento de QoS y otros aspectos de operación. MPLS asigna a cada paquete IP una etiqueta de 32 bits con información especifica de encaminamiento (por ejemplo una ruta y la prioridad del paquete). Con este método se elimina la necesidad de que los *routers* ejecuten una búsqueda de direcciones para cada paquete. De esta manera MPLS logra que los paquetes lleguen a su destino con gran eficiencia y rapidez.

CORBA, y en general la arquitectura OMA (*Object Management Architecture)* del OMG *(Object Management Group)* constituyen una de las soluciones más extendidas al problema del desarrollo e implementación de aplicaciones en entornos distribuidos heterogéneos. Una de las principales ventajas de CORBA reside en su carácter de especificación abierta. CORBA define las interfaces y los servicios para soportar la inter-operabilidad y la transparencia necesarias en la construcción de aplicaciones distribuidas, permite la reutilización de los objetos, ofrece protección frente a la dependencia de un único proveedor y soporta múltiples plataformas.

Agradecimientos

Los autores agradecen a J.I. Díaz, J. Sorribas y R. Boza del CSIC su participación y colaboración en el proyecto y a los revisores (anónimos) de JITEL'01 sus valiosos comentarios.

Este proyecto ha sido financiado por el Ministerio de Ciencia y Tecnología, TIC2000-1027.

Referencias

[1] R. Boza, "Sistema de Adquisición de Datos Oceanográficos". UGBO-CSIC-INF. Marzo 2000.

[2] OMG: "*The Common Object Request Broker: Architecture and Specification". Technical Report, Revision 2.2*". Febrero 1998.

[3] M. Carlson, W.Weiss, S.Blake, Z.Wang, D.Black, E.Davies. *An Architecture for Differentiated Services*. RFC 2475. Dic. 1998

[4] *A conceptual Model for Diffserv Routers*. Internet Engineering Task Force. Marzo 2000.

[5] QoS en I2 http://www.internet2.edu/qos/