

Arquitectura de Topología Híbrida para Sistemas P2P de Compartición de Ficheros

J. P. Muñoz Gea, J. Malgosa Sanahuja, A. M. Guirado Puerta, J. C. Sánchez Aarnoutse, J. García Haro

{juanp.gea, josem.malgosa, antonio.guirado, juanc.sanchez, joang.haro}@upct.es

Departamento de Tecnologías de la Información y las Comunicaciones,

Universidad Politécnica de Cartagena.

Campus Muralla del Mar, 30202, Cartagena, España.

Abstract- Over the Internet today, there has been much interest in emerging Peer-to-Peer (P2P) network overlays because they provide a good substrate for creating data sharing, content distribution, and application-level multicast applications. There are two classes of P2P overlay networks: structured and unstructured. Structured P2P networks can efficiently locate items, but the search is not user friendly. Conversely, unstructured P2P networks have a easy mechanism to search a content, but the lookup process is inefficient. In this paper, we propose an hybrid structured and unstructured topology in order to take advantages of both kind of networks. In addition, our proposal guarantees that if a content is at any place in the network, it will be reachable with probability one. Simulation results show that the behaviour of the network is stable and that the network distributes the contents efficiently to avoid network congestion.

I. INTRODUCCIÓN

La principal característica de una red *overlay* es que los terminales informáticos que la conforman, aún estando interconectados entre sí mediante routers IP, se organizan de tal manera que definen una nueva estructura de red superpuesta a la ya existente. Se trata por tanto de sistemas puramente distribuidos, y que pueden ser utilizados en muchos ámbitos de interés: por ejemplo, para transmitir tráfico multicast utilizando como soporte una red unicast (como Internet); técnica conocida como Application Layer Multicast (ALM). Aunque quizá las redes *overlay* más conocidas sean las denominadas *peer-to-peer* (P2P), muy utilizadas para la descarga eficiente de grandes volúmenes de información. En este último escenario, cabe distinguir dos tipos de redes *overlay* P2P: las estructuradas y las no-estructuradas.

En el primer caso, la construcción de la red *overlay* se realiza en base a un algoritmo conocido de antemano. Dicho algoritmo identifica a cada terminal mediante la aplicación de una función *hash* a su dirección MAC o IP. Conocido el identificador, el algoritmo es capaz de situar al terminal en una posición concreta de la red, pasando a ser un nodo de la red *overlay*. Por contra, en las redes no-estructuradas los terminales se sitúan en la red *overlay* gracias a la intervención de uno (o varios) terminales *rendez-vous* con funciones de gestión de la red.

Las redes P2P no-estructuradas, aunque requieren la presencia de al menos un controlador (*rendez-vous*), tienen la ventaja de que el proceso de búsqueda de la información a descargar es extremadamente sencillo y eficaz (es una metodología muy similar a la utilizada para buscar información en Google, por ejemplo). Ello no ocurre cuando la red P2P es del tipo

estructurada, aunque en este caso la ventaja es que no se requiere de ningún controlador central y en consecuencia la red *overlay* es más escalable, robusta y también mucho más fácil de reorganizar frente a posibles cambios de contorno.

En este trabajo se trata diseñar un sistema de compartición de ficheros que combine las ventajas de ambos tipos de redes P2P. Se pretende que los contenidos puedan localizarse de una manera no-estructurada. En caso de que esta búsqueda falle, se hará uso de un servicio *multicast* de nivel de aplicación (proporcionado por una red P2P estructurada), para intentar localizar al terminal que posea la información buscada. Hay que indicar que con el sistema que se propone en este trabajo se garantiza la localización de cualquier contenido que esté presente en la red.

El resto del artículo está organizado del siguiente modo. La Sección II describe las principales características de las redes estructuradas y no-estructuradas. La Sección III describe detalladamente el sistema que se propone en este artículo. La Sección IV resume las aportaciones más relevantes de la solución propuesta. La Sección V muestra los resultados de simulación del sistema propuesto, y, finalmente, la Sección VI detallan las conclusiones más relevantes.

II. REDES *overlay* P2P

El significado técnico de red estructurada es que la topología de la red *overlay* P2P está determinada algorítmicamente. Tanto los nodos como los contenidos tienen asignado un identificador (*NodeID* y *Key*, respectivamente) pertenecientes al mismo dominio. Estos sistemas P2P utilizan una función DHT (*Distributed Hash Table*) para generar dichos identificadores. Además, existe un algoritmo de *lookup* capaz de localizar el nodo responsable de un contenido a partir de su identificador. Algunos ejemplos de redes P2P estructuradas son: CAN, Chord, Tapestry, Pastry, Kademlia y Viceroy [1],[2],[3].

Las redes P2P no-estructuradas están compuestas por nodos que se unen a la red sin ningún conocimiento previo sobre la topología de la misma. Los terminales necesitan conocer de antemano la ubicación de un controlador central, también denominado *rendez-vous point*, encargado de registrarlos dentro de la red *overlay* y de almacenar los contenidos de que disponen dichos terminales. Las peticiones de búsqueda son enviadas al nodo *rendez-vous*, y éste resuelve en función de la información de que dispone. Si el contenido no está localizado en el *rendez-vous*, la mayoría de las redes utilizan mecanismos

de *flooding* para localizarlo. Algunos ejemplos de redes P2P no-estructuradas son: Gnutella, FastTrack/KaZaA, BitTorrent y eDonkey 2000 [4],[5].

Para un ser humano la búsqueda de un contenido es más fácil en una red no-estructurada, ya que ésta se realiza utilizando patrones de muy alto nivel (como en una búsqueda en Google). Sin embargo, existe mucha ineficiencia en el proceso de localización de dicho contenido. En las redes estructuradas ocurre justo lo contrario: la localización es cuasi-inmediata, pero el proceso de búsqueda es más tedioso.

III. DESCRIPCIÓN DEL SISTEMA

Nuestra propuesta pretende definir un sistema híbrido. De forma que el usuario pueda realizar la búsqueda mediante la introducción de unos parámetros, más o menos generales, y posteriormente elegir de entre los elementos que satisfagan el criterio de búsqueda, aquel contenido que desea descargar, al igual que en las redes no-estructuradas. Sin embargo, la red se organizará de forma estructurada, lo que facilitará la localización del contenido elegido.

Todos los nodos están inmersos en una red *overlay* estructurada, de cualquiera de los tipos mencionados anteriormente. Además, los nodos se van a agrupar en distintos subgrupos, de forma más o menos uniforme, entorno a un nodo *rendez-vous*, que destacará del resto de nodos de su subgrupo por sus prestaciones en cuanto a CPU, ancho de banda y fiabilidad. Cada uno de los nodos del subgrupo será capaz de establecer una conexión TCP con su nodo *rendez-vous*, y le enviará información sobre los ficheros que desean compartir. Para que cada nodo pueda identificar al nodo *rendez-vous* que le corresponde, se utilizan las características que brinda una red estructurada, de tal forma que a partir de una cierta información que identifique al grupo, pueda localizar a su nodo *rendez-vous*. Esta última propiedad permite definir dinámicamente los nodos *rendez-vous* y garantizar la estabilidad de la red a lo largo del tiempo. A la hora de realizar la búsqueda de un determinado contenido, el usuario enviará a su *rendez-vous* unos parámetros de búsqueda y éste le devolverá información sobre los contenidos ubicados en dicho subgrupo y coincidentes con esos parámetros.

Todos los nodos *rendez-vous* de la red van a formar parte de un grupo multicast definido dentro de la misma red estructurada. De este modo, si la búsqueda falla, el nodo *rendez-vous* reenviará la petición hacia el resto de nodos *rendez-vous* en modo multicast (ALM) La Fig. 1 describe la arquitectura general del sistema.

A. Obtención de Identificadores y Unión a la Red General

Cada nodo necesita obtener un *nodeID*. En este trabajo se propone que este identificador se obtenga al aplicar una función *hash* (MD5 o SHA-1) a su dirección IP pública. De la misma forma, cada nodo también necesita obtener un *subgroupID*, que identifica al subgrupo al cual va a pertenecer el nodo. La idea que se propone es utilizar un servidor conocido de antemano para conseguir este identificador. Cada grupo tendrá un número máximo de nodos, y los nodos se irán asignando por orden a cada uno de los subgrupos hasta completar su capacidad máxima. A medida que se completen los grupos existentes se podrán crear nuevos subgrupos.

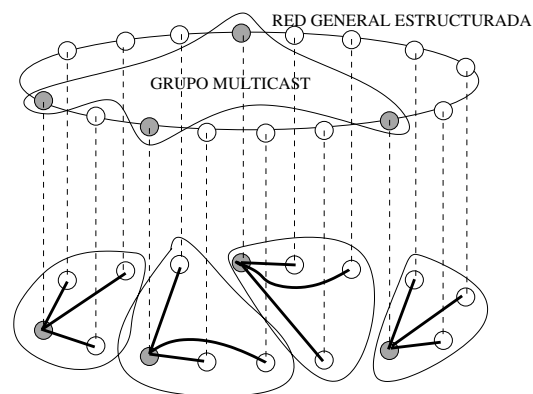


Fig. 1. Arquitectura general del sistema. Se trata de una red estructurada con capacidad de realizar búsquedas en modo no estructurado.

Como es habitual en cualquier red estructurada un nodo también necesita conocer la dirección de otro nodo de la misma red. El servidor anterior también puede proporcionar esta información. Finalmente, el nodo se deberá unir a la red *overlay* P2P, utilizando para ello el mecanismo propio de la red estructurada que se utilice.

B. Unión al Subgrupo

Además de pertenecer a la red general, cada nodo también va a formar parte de un subgrupo. Los miembros de este subgrupo se caracterizan porque todos se conectan al mismo nodo *rendez-vous*. Cuando se incorpora un nodo nuevo hace un *lookup* en la red estructurada utilizando como parámetro de búsqueda el identificador del subgrupo (*subgroupID*) al que pertenece, y que previamente obtuvo. Dicho proceso le proporcionará la dirección IP del nodo *rendez-vous* del subgrupo correspondiente.

C. Gestión de la Jerarquía

Cuando el nuevo nodo encuentra a su *rendez-vous*, le notifica los recursos de CPU y ancho de banda de que dispone. Los nodos *rendez-vous* monitorizan a los nodos que se unen a su subgrupo y presentan buenas características, y forman una lista ordenada de futuros *rendez-vous* candidatos: cuanto más tiempo permanece conectado un nodo y mejores son sus recursos, mejor candidato a *rendez-vous* se convierte. Esta lista se transmite a todos los miembros del subgrupo, y cuando el *rendez-vous* falla, el primer nodo en la lista se convierte en su sucesor. Después debe informar a todos los miembros del subgrupo de que él es ahora el nuevo *rendez-vous*, modificar el archivo en el que se identifica el *rendez-vous* del grupo, e incorporarse al grupo de los *rendez-vous*, es decir, registrarse como *rendez-vous*.

D. Gestión de los Nodos Rendez-Vous

Todos los nodos *rendez-vous* forman parte de un mismo grupo multicast (como ya se ha mencionado, definido a nivel de aplicación). Cuando un nodo se convierte en *rendez-vous*, debe primero unirse a dicho grupo multicast, encargado de difundir las búsquedas fallidas en cada subgrupo. Las redes P2P estructuradas se pueden usar para implementar un servicio multicast de nivel de aplicación, como por ejemplo CAN-Multicast [6], Chord-Multicast [7] y Scribe [8]. Cada uno

usa una *overlay* P2P diferente y pueden implementar el servicio de multicast usando inundación (CAN-Multicast, Chord-Multicast) o la construcción de un árbol (Scribe). Cualquiera de los métodos anteriores proporciona un mecanismo eficiente para identificar y mandar mensajes a todos los miembros de un grupo.

Nuestra propuesta utiliza un Chord-Multicast modificado, que intenta minimizar el impacto de las inundaciones en la carga de la red. En el momento en que un nodo del grupo multicast responde afirmativamente a una petición, se aborta inmediatamente el proceso de inundación.

E. Registrar los Ficheros Compartidos

De forma similar a KaZaA [9], cuando un nodo establece conexión con su *rendez-vous* le envía los *metadatos* de aquellos ficheros que quiere compartir. Esto le permite al *rendez-vous* mantener una base de datos que incluye los identificadores de los ficheros que todos los nodos del subgrupo están compartiendo y la correspondiente dirección IP del nodo que los contiene. La información que envía el nodo incluye: el nombre del fichero, su tamaño y sus descriptores.

F. Búsqueda

Cuando un usuario desea realizar la búsqueda de un determinado contenido envía una solicitud sobre la conexión TCP establecida con su *rendez-vous*. Para cada coincidencia de la búsqueda en la base de datos, el *rendez-vous* devuelve la dirección IP y los *metadatos* correspondientes.

En caso de que esta búsqueda falle, el usuario tiene la posibilidad de solicitarle a su nodo *rendez-vous* que intente contactar con otros *rendez-vous*. La identificación de estos nodos se realizará de forma sencilla, ya que todos pertenecen al mismo grupo multicast de nivel de aplicación.

IV. VENTAJAS DEL SISTEMA

A continuación se van a describir algunas de las aportaciones del sistema propuesto en este trabajo. En primer lugar hay que destacar que los nodos son asignados a un subgrupo y no a un servidor, o a un nodo *rendez-vous* determinado. Los propios nodos, de una forma sencilla, son capaces de encontrar el nodo *rendez-vous* a partir del grupo del que forman parte. Al ser la asignación de los *rendez-vous* dinámica, la definición de los subgrupos es muy robusta ante fallos de estos, porque de forma automática y óptima cualquiera de ellos tomará el relevo del que ha caído.

También hay que destacar que en este sistema se consigue explotar la heterogeneidad de la red. Los nodos más estables y con mejores prestaciones se convertirán en nodos *rendez-vous*, lo que aumentará de forma considerable el rendimiento de la red. El hecho de que los nodos más estables a lo largo del tiempo tenga más posibilidades de convertirse en nodos *rendez-vous* hace que el tráfico de señalización, y las posibles sobrecargas que éste conllevaría se minimicen.

Por otra parte, el servicio de multicast a nivel de aplicación proporciona una forma eficaz de que un nodo *rendez-vous* conacte con el resto de nodos *rendez-vous* de la red. La ventaja que proporciona este tipo de servicios es que los trabajos de mantenimiento de los grupos multicast se realizan de forma prácticamente automática. Además, esto va a garantizar que

cualquiera de los contenidos presentes en la red pueda ser localizado por cualquier usuario.

Por último, las búsquedas por parte de los usuarios se realizarán de una forma sencilla, similar a las realizadas en las actuales aplicaciones de compartición de ficheros.

V. SIMULACIONES

Una de las ventajas de este sistema, comentadas anteriormente, es que cualquier contenido presente en la red podrá ser localizado por cualquier usuario. Sin embargo, las búsquedas de contenidos presentes en el subgrupo del solicitante serán mucho más rápidas y eficientes, en términos de carga en la red, que las búsquedas en otros nodos *rendez-vous* de la red.

Hay varios parámetros de interés que se deben cuantificar. En primer lugar, la probabilidad de que el contenido que se solicita esté ya registrado en el nodo *rendez-vous* del solicitante. En segundo lugar, la evolución del parámetro anterior a lo largo del tiempo, ya que a medida que los usuarios vayan realizando búsquedas sucesivas de contenidos en otros subgrupos de la red, éstos automáticamente pasarán a estar registrados en su propio nodo *rendez-vous*, aumentando el valor de esta probabilidad. Por último, también es de interés contabilizar el número medio de nodos *rendez-vous* que se deberán contactar para localizar un contenido no presente dentro del subgrupo del solicitante.

Para cuantificar los parámetros anteriores se ha programado un simulador en lenguaje C. A continuación, se describen las características más relevantes de este simulador. En primer lugar, los contenidos se clasifican en tres clases en función del grado de interés que pueden despertar en los usuarios del sistema (muy apetecibles, apetecibles y poco apetecibles). En la inicialización del simulador, los contenidos disponibles se reparten de forma aleatoria entre los nodos de la red, de forma que cada contenido se sitúa en un único nodo. También hay que indicar, como ya se comentó anteriormente, que el algoritmo utilizado para implementar el multicast a nivel de aplicación ha sido un Chord-Multicast modificado.

Los resultados que nos proporciona el simulador son la probabilidad de que el contenido buscado se encuentre en el *rendez-vous* del solicitante, y el número medio y máximo de *rendez-vous* consultados hasta localizar el contenido buscado. Todos estos resultados se obtienen en función del número de iteraciones de simulación, en cada una de las cuales se hace que todos los nodos de la red soliciten un contenido que no posean en ese momento.

Los resultados de simulación que se presentan en este trabajo son los correspondientes a una red con 12.800 contenidos distintos y 6400 nodos, de los cuales 128 son *rendez-vous*.

En la Fig. 2 se presenta la probabilidad de que el contenido buscado esté en el mismo subgrupo que el solicitante, tanto para los contenidos más apetecibles, como para un contenido cualquiera. Se observa como esta probabilidad crece a medida que aumenta el número de iteraciones, hasta llegar a converger a un valor de uno, lo que asegura que nuestro sistema es estable. Esto, también nos indica que la arquitectura de nuestro sistema nos va a asegurar que la probabilidad de que un contenido cualquiera se encuentre presente en algún nodo del subgrupo del solicitante es muy alta, una vez transcurrido un corto periodo de tiempo a partir de la inicialización del

sistema. También se puede observar como en ese transitorio inicial la probabilidad de encontrar un contenido más apetecible en el *rendez-vous* aumenta más rápidamente que la de un contenido cualquiera, y converge más rápidamente al valor de uno.

En la Fig. 3 se presenta de forma logarítmica el número medio de *rendez-vous* consultados para localizar un contenido. Se observa como transcurrido el transitorio inicial el número de *rendez-vous* que es necesario consultar disminuye hasta un valor menor de 10, y aumentando el número de iteraciones, este valor converge hasta un valor de uno, que representa al propio *rendez-vous* del subgrupo del solicitante. Esto nos indica que la carga procedente de otros subgrupos que van a tener que soportar los *rendez-vous* es mínima. También se observa como este parámetro decrece más rápidamente en el caso de los contenidos más apetecibles que para cualquier otro contenido.

Por último, en la Fig. 4 se representa en unidades lineales el número máximo de *rendez-vous* consultados para localizar un contenido en cada una de las iteraciones. Se observa como este parámetro oscila mucho en el transitorio inicial, pero transcurrido éste, converge hasta valores cercanos a la unidad, coincidiendo prácticamente con la media.

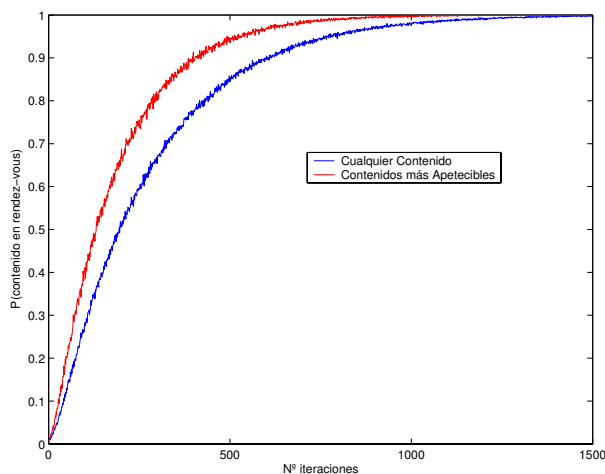


Fig. 2. Probabilidad de que el contenido buscado esté en el *rendez-vous* del solicitante.

VI. CONCLUSIONES

Este artículo presenta una red *overlay* P2P híbrida que facilita al usuario tanto el proceso de búsqueda como la localización de los contenidos. Los resultados de simulación demuestran que en este tipo de redes los contenidos se distribuyen de forma que se minimiza la sobrecarga en los nodos *rendez-vous*.

AGRADECIMIENTOS

Este trabajo ha sido subvencionado por el proyecto nacional ARPAq (TEC2004-05622-C04-02/TCM), y con fondos de la D.G. de Innovación Tecnológica y Sociedad de la Información de la Consejería de Industria y Medio Ambiente de la Región de Murcia, y con fondos FEDER de la Unión Europea.

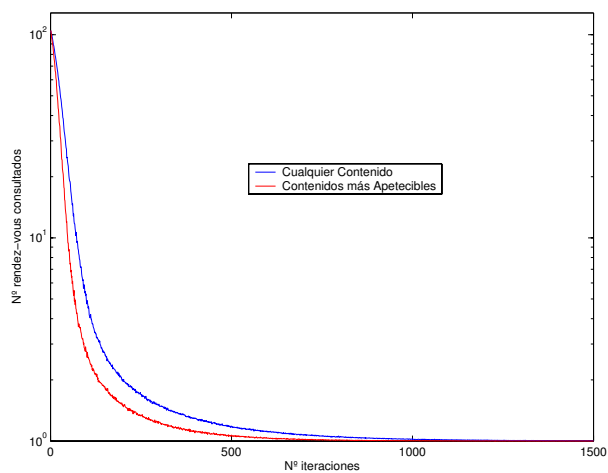


Fig. 3. Número medio de *rendez-vous* consultados para localizar un contenido.

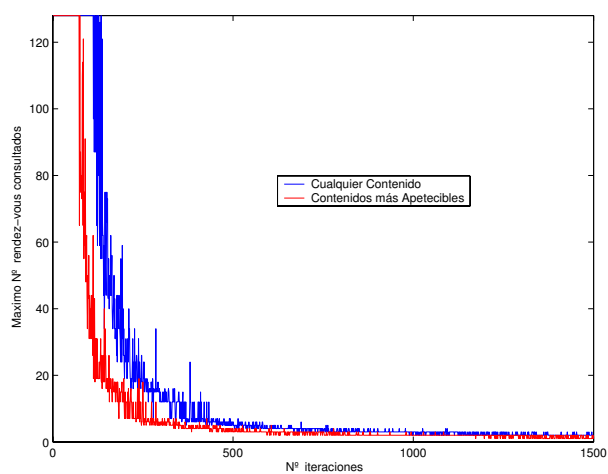


Fig. 4. Número máximo de *rendez-vous* consultados para localizar un contenido.

REFERENCIAS

- [1] Ion Stoica, Robert Morris, David R. Karger, M. Frans Kaashoek, Hari Balakrishnan. "Chord: A scalable peer-to-peer lookup service for Internet Applications". Proceedings of the SIGCOMM'01, San Diego, CA, pp. 149-160, 27-31 August 2001.
- [2] B.Y. Zhao, L. Huang, J. Stribling, S.C. Rhea, A.D. Joseph, J. D. Kubiatowicz. "Tapestry: A Resilient Global-Scale Overlay for Service Deployment". IEEE Journal on Selected Areas in Communications (JSAC), Vol. 22, No. 1, January 2004, pp. 41-53.
- [3] Petar Maymounkov, David Mazières. "Kademlia: A Peer-to-peer Information System Based on the XOR Metric". Proceedings of the First International Workshop on Peer-to-peer Systems, MIT, March 2002.
- [4] "Fastrack Peer-to-Peer Technology Company". Available at <http://www.fastrack.nu/>, 2001.
- [5] "Bittorrent". Available at <http://bitconjurer.org/BitTorrent/>, 2003.
- [6] S.Ratnasamy, M. Handley, R. Karp, S. Shenker. "Application-level multicast usgin content-addressable networks". In Proceedings of NGC, November 2001.
- [7] Sameh El-Ansary, Luc Onana Alima, Per Brand, Seif Haridi, "Efficient Broadcast in Structured P2P Networks", Proc of the 2nd International Workshop on Peer-to-Peer Systems (IPTPS), pp. 304-314, February 2003
- [8] Miguel Castro, Peter Druschel, Anne-Marie Kermarrec and Antony Rowstron. "Scribe: A large-scale and decentralized application-level multicast infrastructure". IEEE Journal on Selected Areas in Communications (JSAC), Vol. 20, No. 8, October 2002, pp. 100-110.
- [9] Jian Liang, Rakesh Kumar, and Keith W. Ross. "Understanding KaZaA". 2004. Available at <http://cis.poly.edu/ross/papers/UnderstandingKaZaA.pdf>.