

# Gestión de Sistemas P2P mediante Redes *Overlay* Estructuradas

Juan Pedro Muñoz Gea, Pilar Manzanares López, Juan Carlos Sánchez Aarnoutse  
Escuela Técnica Superior de Ingeniería de Telecomunicación. Universidad Politécnica de Cartagena  
Campus Muralla de Mar. Antiguo Cuartel de Antigones  
30202 Cartagena, Teléfono: 968 33 8871 Fax: 968 32 5973  
E-mail: {juanp.gea, josem.malgosa, pilar.manzanares, juanc.sanchez}@upct.es

**Resumen.** En este artículo se presenta una nueva red híbrida P2P. Todos los nodos están inmersos en una red *overlay* estructurada, que no se utiliza para localizar contenidos, sino para organizar dinámicamente los nodos en una arquitectura jerárquica, que permita la localización de los contenidos de una forma fácil y eficiente. Las figuras de mérito del sistema propuesto son evaluadas mediante simulación.

## 1 Introducción

Existen dos tipos de redes *overlay* P2P: estructuradas y no-estructuradas. En este trabajo hemos diseñado una red P2P híbrida, mucho más eficaz y segura que las descritas en [1][2] [3], y que comparte las ventajas de ambos tipos de redes. Los nodos del sistema están inmersos en una red estructurada, utilizada para organizar dinámicamente los nodos, mientras que el proceso de búsqueda utiliza una arquitectura no-estructurada jerárquica capaz de soportar solicitudes complejas.

## 2 Descripción del sistema

Los nodos se dividen automáticamente en diferentes subgrupos. Cada subgrupo es gestionado por un *SuperPeer* de nivel 1, que a su vez se agrupan en subgrupos gestionados por *SuperPeers* de nivel 2. Este proceso continúa recursivamente hasta alcanzar el subgrupo de nivel más alto.

Todos los *SuperPeers* de nivel 1 mantienen una base de datos que registra las listas de contenidos de todos los nodos de su subgrupo. El resto de *SuperPeers* no mantienen esta información, pero pueden solicitarla a los miembros de su subgrupo cuando sea necesario. Para buscar contenidos, el nodo del usuario enviará los parámetros de búsqueda a su *SuperPeer* de nivel 1, y éste le devolverá la información sobre quien tiene el contenido en su subgrupo. Si la búsqueda falla, el *SuperPeer* de nivel 1 enviará la solicitud a su *SuperPeer* asociado de nivel superior. Si es necesario, este proceso continuará hasta el *SuperPeer* de más alto nivel.

### 2.1 Inicialización y Unión a la Red

Cada nodo obtiene su identificador (*NodeId*) aplicando una función *hash* a su dirección MAC. El nuevo nodo contacta con un nodo cualquiera existente en la red para unirse a la red estructurada, obteniendo de él su *SubgroupId*, que identifica el subgrupo al cual pertenece este nodo. El nuevo nodo

utilizará este identificador para intentar unirse a ese subgrupo.

### 2.2 Unión a un Subgrupo

A continuación, el nodo debe buscar su *SuperPeer* correspondiente. Utiliza la red estructurada para localizar el nodo cuyo *NodeId* coincide con el *SubgroupId* obtenido previamente. Este nodo conoce la dirección IP del *SuperPeer* del subgrupo. El nuevo nodo contacta con ese *SuperPeer*, y le solicita unirse a su subgrupo. Cada subgrupo tiene un número máximo de nodos. Si hay sitio, se incorpora al grupo y le envía al *SuperPeer* su lista de contenidos.

### 2.3 Asignación de *SuperPeer*

El proceso previo falla cuando el nodo intenta unirse a un subgrupo completo. En este caso, el *SuperPeer* le asigna al nodo un nuevo *SubgroupId*, y se convierte en el *SuperPeer* del nuevo subgrupo. Para ser alcanzable por el resto de la comunidad, el nuevo *SuperPeer* usa la red estructurada para insertar su dirección IP en el nodo cuyo *NodeId* coincide con el nuevo *SubgroupId*. Si este nodo ya tiene un valor, el *SubgroupId* generado está repetido. En este caso, el nodo intenta unirse a ese subgrupo como un nodo normal. Cada *SuperPeer* sólo puede generar un *SubgroupId* adicional, y debe conservar este identificador para responder a las solicitudes futuras.

Cuando el segundo subgrupo de nivel 1 se crea, es necesario crear un nuevo subgrupo de nivel 2. El *SuperPeer* responsable de crear este segundo subgrupo de nivel 1 se convierte en el *SuperPeer* del nuevo subgrupo de nivel 2. Los subgrupos de nivel 2 también tienen un número máximo de nodos, y por lo tanto, el proceso de creación de subgrupos continúa de una forma similar hasta el nivel más alto de jerarquía.

Si un nodo intenta conectar con su *SuperPeer* sin éxito, quizás sea porque éste ha caído. En este caso, el nodo busca al nodo cuyo *NodeId* encaja con su *SubgroupId* y comprueba la dirección IP registrada. Si la dirección IP todavía coincide con la dirección IP

del *SuperPeer* que ha fallado, el nodo inserta su propia dirección IP en este registro y se convierte en el nuevo *SuperPeer* del subgrupo correspondiente. De esta forma, cuando el resto de nodos detecten que el *SuperPeer* previo ha fallado, podrán obtener la dirección IP del nuevo *SuperPeer*. Este mecanismo proporciona fiabilidad frente a caídas no esperadas de los nodos *SuperPeer*.

## 2.4 Búsqueda de contenidos

Para buscar contenidos, el nodo del usuario enviará los parámetros de búsqueda a su *SuperPeer* de nivel 1, y este devolverá información sobre quien tiene ese contenido en su subgrupo. Si la búsqueda falla, el *SuperPeer* de nivel 1 enviará la solicitud a su *SuperPeer* asociado de nivel superior. Si es necesario, este proceso continuará hasta el *SuperPeer* de más alto nivel. Los *SuperPeers* de niveles superiores no mantienen bases de datos de contenidos, sino que tan sólo solicitan información a los miembros de sus subgrupos cuando es necesario.

## 3 Simulaciones

En esta sección, evaluamos nuestro sistema propuesto mediante simulación. Se asume que, dada una distribución de contenidos inicial en la red, en cada paso de simulación todos los nodos realizan una solicitud para encontrar y descargar un contenido nuevo. Además, los contenidos se clasifican en tres tipos, en función del grado de interés que despiertan en el usuario: muy interesantes, interesantes y de poco interés.

El escenario de simulación está formado por 12.500 contenidos diferentes uniformemente distribuidos entre 6.250 nodos. Los tamaños de los subgrupos están establecidos a 50 y 5 nodos para el primer y el resto de niveles de la jerarquía, respectivamente. Por lo tanto, hay 125 *SuperPeers* de nivel 1, 25 de nivel 2, 5 de nivel 3 y uno de nivel 4. La Fig. 1 muestra la evolución temporal de la probabilidad de que un contenido esté en el mismo subgrupo del solicitante. Se puede observar que esta probabilidad crece y converge a la unidad. Cuando un *SuperPeer* sufre un fallo inesperado, se inicia el mecanismo para reconstruir la red. La reconstrucción de la red conlleva un periodo de tiempo y durante ese tiempo, algunos de los contenidos presentes en el subgrupo que ha caído no serán accesibles por el resto de los nodos de la red. La Fig. 2 muestra la probabilidad de que durante el tiempo de reconstrucción, un contenido sea inalcanzable, cuando un número fijo de *SuperPeers* de nivel 1 falla simultáneamente. Esta probabilidad es cercana a cero cuando el *SuperPeer* falla en la ranura de tiempo número 20. Ello es debido a que conforme pasa el tiempo, los contenidos se distribuyen entre el resto de subgrupos, y el fallo de un conjunto de *SuperPeers* tiene un efecto pequeño en el rendimiento de la red entera.

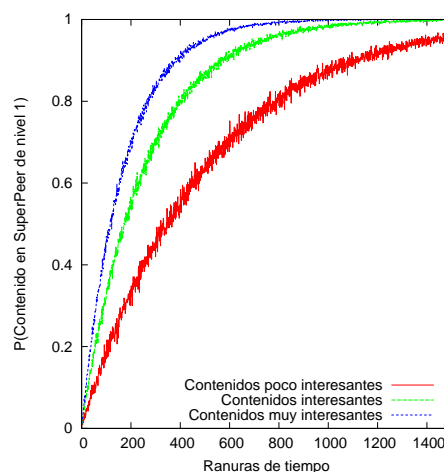


Fig. 1. Evolución temporal de la probabilidad de encontrar un contenido en el mismo subgrupo del solicitante.

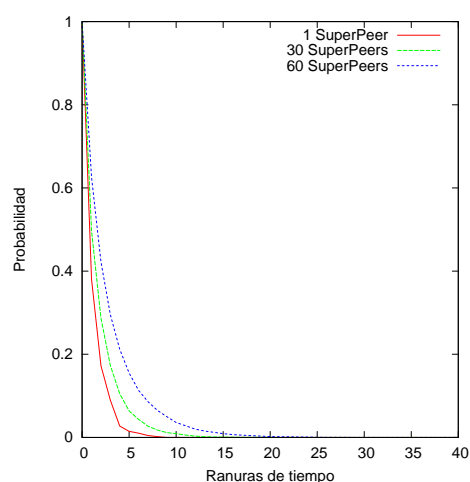


Fig. 2. Evolución temporal de la probabilidad de que durante el tiempo de reconstrucción, un contenido sea inalcanzable cuando varios *SuperPeers* de nivel 1 fallan simultáneamente.

## Agradecimientos

Este trabajo ha sido subvencionado con fondos de la Dirección General 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.

## Referencias

- [1] B. Traversat, et al., "Project JXTA 2.0 Super-Peer Virtual Network". <http://www.jxta.org/project/www/docs/JXTA2.0protocols1.pdf>
- [2] B. Zhao, et al. "Brocade: Landmark routing on overlay networks". Proc. of the 1st International Workshop on Peer-to-Peer Systems, 2002.
- [3] J. Mischke, B. Stiller. "Rich and Scalable Peer-to-Peer Search with SHARK". Proceedings of the 5th International Workshop on Active Middleware Services, 2003.