



Centro de Tecnologías de Información y Comunicación
Universidad Nacional de Ingeniería, Lima

Introducción a Sistemas Peer to Peer

Yudith Cardinale y Jesús De Oliveira
Universidad Simón Bolívar

Marzo 2009



Peer-to-Peer computing: Cómputo intensivo descentralizado

- Sistemas y aplicaciones que emplean recursos distribuidos para realizar funciones críticas de manera descentralizada
 - Cómputo distribuido
 - Almacenamiento y compartimiento de datos (intercambio de archivos)
 - Comunicación y colaboración para crear software
 - Conversación directa en línea
 - Servicios en plataformas heterogéneas



Peer-to-Peer computing: Se presupone que los nodos participantes no tienen control sobre el acceso a los mismos

- Qué ofrecen:
 - Cooperación
 - Independencia de localidad
 - Rodeo de controles
 - Anonimato



Peer-to-Peer computing: Ejemplos de sistemas

● Sistemas para cómputo distribuido:



Proyecto académico para búsqueda de señales provenientes de otros mundos

MAGI

Producto para aplicaciones de negocios colaborativos (comercio basado en p2p) e incorpora múltiples tipos de dispositivos entre peers (PDAs, celulares, chips especializados, comunicación basada en eventos)



Proyecto de software abierto que provee servicios para p2p: encontrar peers, compartir archivos, encontrar contenido en sitios remotos, crear un grupo de peers, monitorear actividades y comunicación segura.

Peer-to-Peer computing: Ejemplos de sistemas

● Sistemas para comunicación:

groove



Provee espacios virtuales compartidos para la interacción de pequeños grupos. Los usuarios que comparten un espacio pueden “chatear”, comunicarse por voz, enviar mensajes instantáneos, actualizar un calendario o plan de eventos, compartir un archivo, etc.

● Sistemas para compartir información:

» Intercambio de música mp3. Creció considerablemente a punto de que fueron demandados por compañías editoras de discos... y perdieron.



- Servidor centralizado para mantenimiento de información de ubicación

El intercambio de información se hace directamente entre clientes

Peer-to-Peer computing: Ejemplos de sistemas

● Sistemas para compartir información: (cont.)



Intercambio de archivos, principalmente mp3. Incluyeron secretamente un módulo de cómputo, hecho por Brilliant Digital para hacer cómputo distribuido.

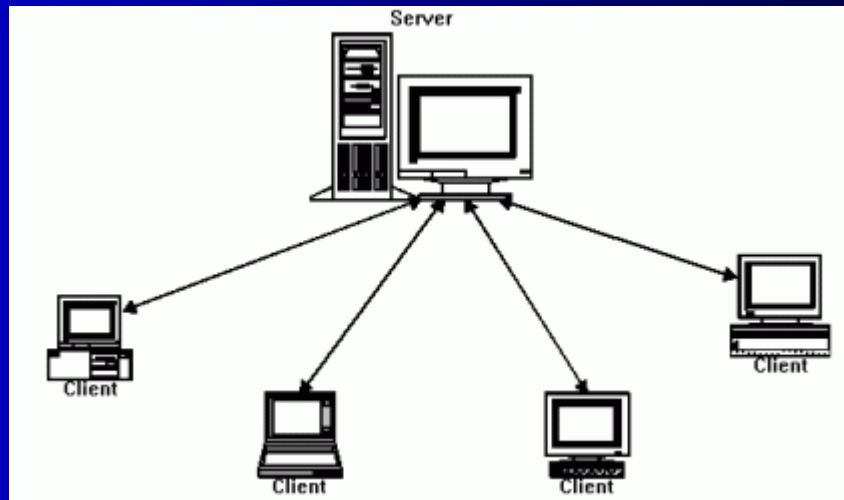


gnutella.com

- ◆ Compartimiento, búsqueda y copiado de archivos entre usuarios en Internet en forma descentralizada
- ◆ Los clientes son servidores al mismo tiempo.
- ◆ Aprende sobre los nodos conectados al vecino
- ◆ Descubrimiento de la red (ping y pong)

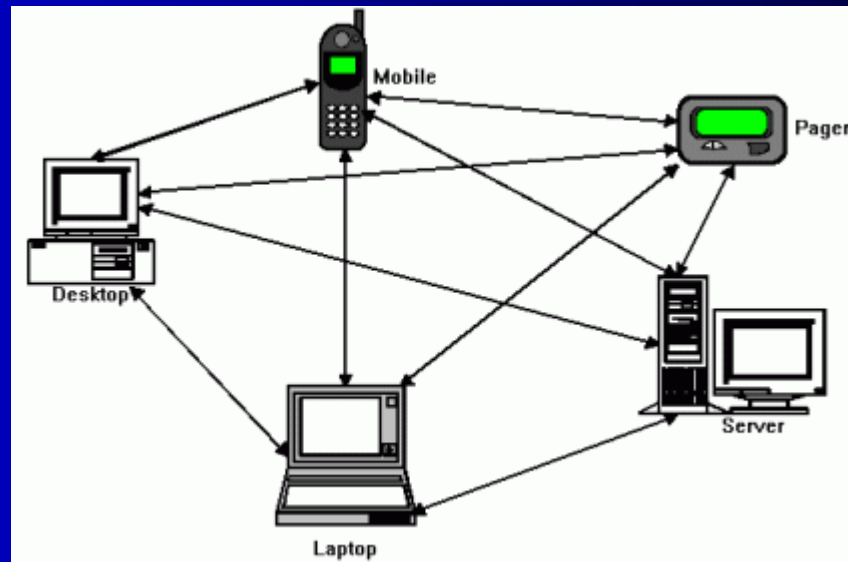
Modelo Cliente/Servidor

- Los clientes se comunican con el servidor
- Pueden comunicarse entre sí *a través* del servidor



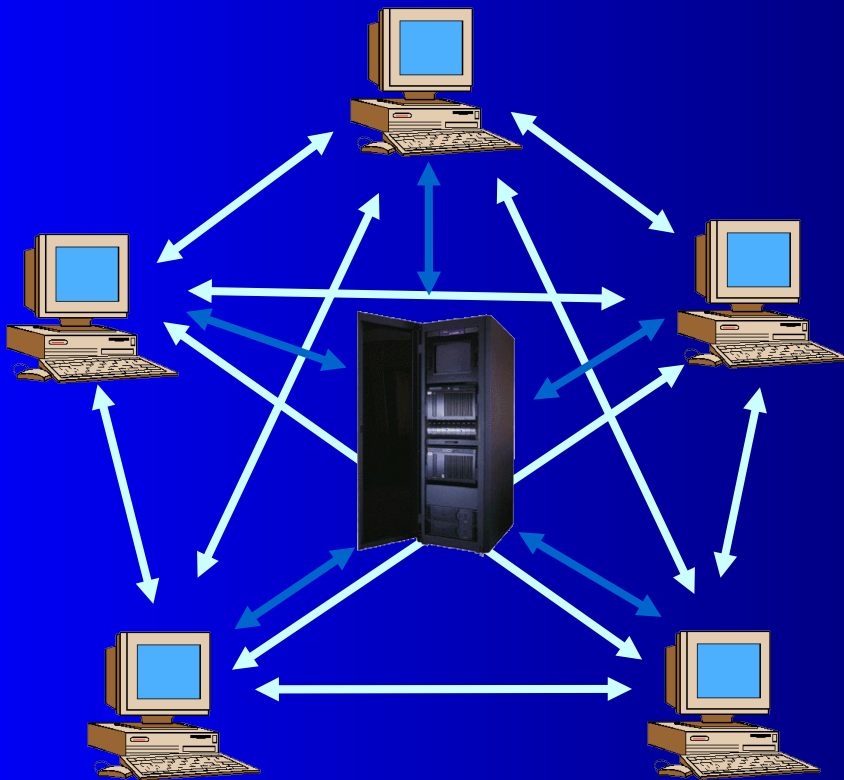
Modelo P2P (punto-a-punto)

- Los clientes se comunican directamente
- Pueden consultar previamente a un servidor con propósitos de localización

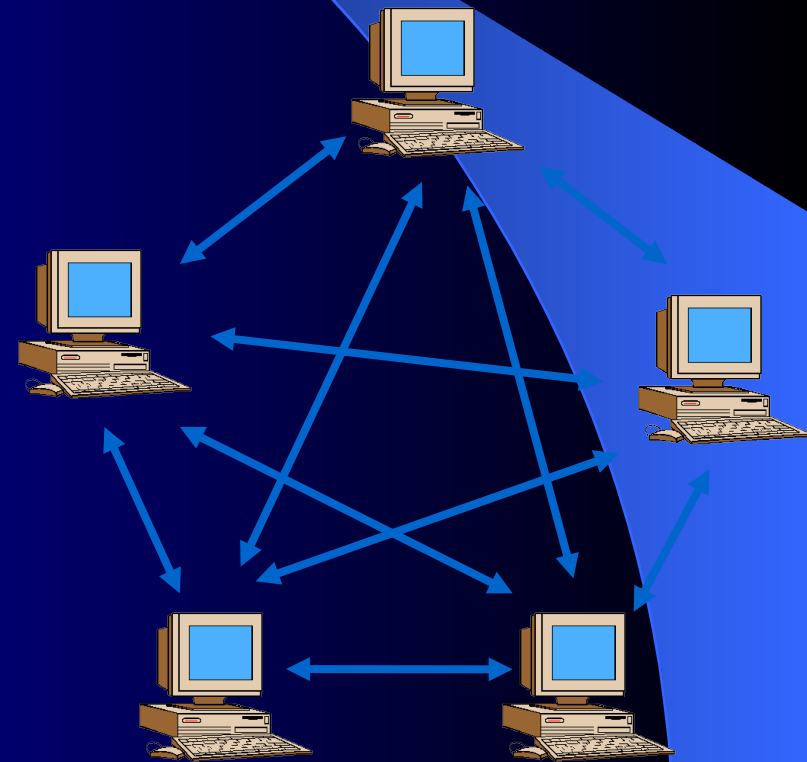


Variantes de P2P

Control central (ej. Napster)



Distribuido (ej. Gnutella)



Espacio de Aplicaciones P2P

- ◆ Las aplicaciones P2P se pueden definir en tres ejes principales:
 - **Descentralización:** puede haber un control centralizado (seti) o ser completamente independiente de un servidor central (freenet)
 - **Cooperación:** puede no haber comunicación entre pares (seti), basarse en la cooperación continua entre pares (freenet) o un esquema intermedio (napster)
 - **Indirección:** puede haber cooperación P2P directamente entre máquinas cliente (napster) o indirectamente, entre servidores (almacenamiento distribuido)

Consideraciones

◆ Interoperabilidad

- ◆ Protocolos
- ◆ Formato de datos

◆ Seguridad

- ◆ Confidencialidad
- ◆ Presencia de cortafuegos, NAT, etc.

◆ Descentralización

- ◆ Independencia del DNS
- ◆ Compartir recursos sin intermediación

◆ Identificación

- ◆ Nombres para los “peers”

Interoperabilidad: protocolos

- Hay muchos de ellos, ejemplos:
 - **HTTP** (Hypertext Transfer Protocol): protocolo para solicitud de información, usado en el Web
 - **SOAP** (Simple Object Access Protocol): protocolo para invocar código usando XML sobre HTTP
 - **GIOP** (General Inter ORB Protocol): protocolo utilizado en CORBA para comunicar ORBs
 - **RTP** (Real-Time Transport Protocol): protocolo para intercambio de datos a tiempo real, como video y audio
 - **Gnutella Protocol**: descubrimiento usado por Gnutella

Interoperabilidad: datos

- **HTML**: conjunto de etiquetas y reglas para definir documentos de hipertexto
- **XML**: conjunto de reglas para definir formatos de datos estructurados
- **Java bytecode**: se pueden ejecutar applets en casi cualquier plataforma
- **¿DOC?**: estamos hasta la coronilla de que nos envíen documentos en formato DOC

Seguridad: autenticación y confidencialidad

- **SSL** (Secure Socket Layer): protocolo para establecer conexiones de sockets seguras
- **HTTPS**: simplemente HTTP sobre SSL
- **PKI** (Public Key Infrastructure): sistema de cifrado de clave pública que se apoya en certificados digitales
- **SSH** (Secure Shell): programa para utilizar un shell en un computador remoto y para hacer túneles seguros

Seguridad: cortafuegos y NAT

- El modelo cliente-servidor ha dado origen a mecanismos que permiten la conexión “hacia afuera” a todos pero limitan la conexión “hacia adentro”
- NAT
 - ◆ Se origina por la carencia de direcciones IP
 - ◆ Impide el acceso a máquinas internas, a menos que haya NAT hacia adentro
- Cortafuegos
 - ◆ Se origina por la necesidad de protegerse de ataques
 - ◆ Intencionalmente se evita el acceso a máquinas internas, con excepción de ciertos servicios
- Se necesitan mecanismos de conexión a servicios que utilicen la conexión TCP hacia afuera

Descentralización: independencia del DNS

- Los servicios de acceso a Internet (por discado o por conexión de banda ancha) en general no incluyen un nombre DNS
- De hecho, los servicios de acceso a internet suelen asignar números IP que cambian continuamente
- Hay servicios de DNS dinámico (solución parcial)
- Se necesitan mecanismos de identificación (*naming*) alternativos, que no se basen en la asociación DNS-IP
- Para asociar un nombre con un destino (IP, Puerto) hace falta
 - Servidor de nombres
 - Mecanismo de descubrimiento

Descentralización: participar sin intermediación

- No necesariamente se cuenta con servidores con DNS oficial
- Posibilidad: establecer redes que se tejen extendiendo las conexiones con vecinos
- Se necesitan protocolos de descubrimiento, tanto de nodos como de otros objetos (archivos)
- Se puede tejer una red de servidores de nombres alternativa al DNS