Introd. a los Sistemas Distribuidos, Arquit. Cliente/Servidor y P2P

Mg. Gabriel H. Tolosa

tolosoft@unlu.edu.ar

“A distributed system is one in which the failure of a computer you didn't even know existed can render your own computer unusable”.

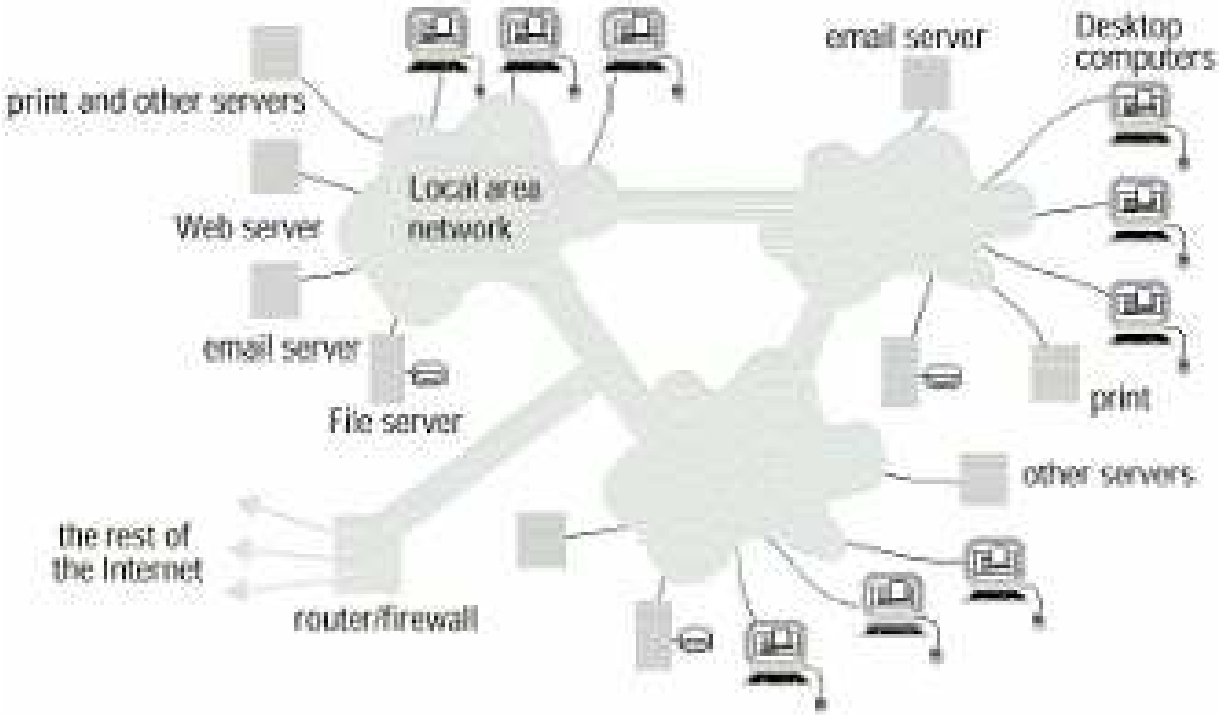
Leslie Lamport

“Time, Clocks, and the Ordering of Events in a Distributed System

Computación Distribuida

Definición

- Modelo donde los procesos ocurren en diferentes ‘lugares’ de la red (nodos)
- Los servicios son provistos por un conjunto de computadoras que colaboran, cada una con un rol o función particular (en un momento dado)



Sistemas Distribuidos

Definiciones

- Colección de computadoras autónomas enlazadas por una red y equipadas con un software distribuido, que cooperan en pos de un objetivo común.

- Un SD es uno en el cual los componentes localizados en una red de computadoras se comunican y coordinan sus acciones solo por pasaje de mensajes. (Colouris, 2002)

- Un sistema distribuido es una colección de computadoras independientes que se muestran al usuario como un sistema único coherente. (Tanembaum)

Sistemas Distribuidos

- **Ejemplos**
  - **WWW, Correo Electrónico**
  - **Sistemas Operativos Distribuidos**
    - Chorus, Mach 9
  - **Computación Distribuida**
    - SETI, Distributed.net
  - **Sistemas de Distribución de Archivos**
    - Gnutella

Pero: Si el procesamiento es centralizado en un equipo al sistema no se lo considera distribuido.
Por ejemplo: La búsqueda remota en una BD centralizada no es un sistema distribuido (Parcialmente distribuido): **y Google?**

Sistemas Distribuidos

Características

- **No hay bus común**
  - => Varios procesadores débilmente acoplados
- **No hay reloj común**
  - => Carencia de tiempo global
- **No hay memoria principal compartida**
  - => Requieren de la red para pasaje de mensajes. La red es invisible al usuario.
- **Paralelismo de componentes**
- **Fallas independientes de los componentes**

Sistemas Distribuidos

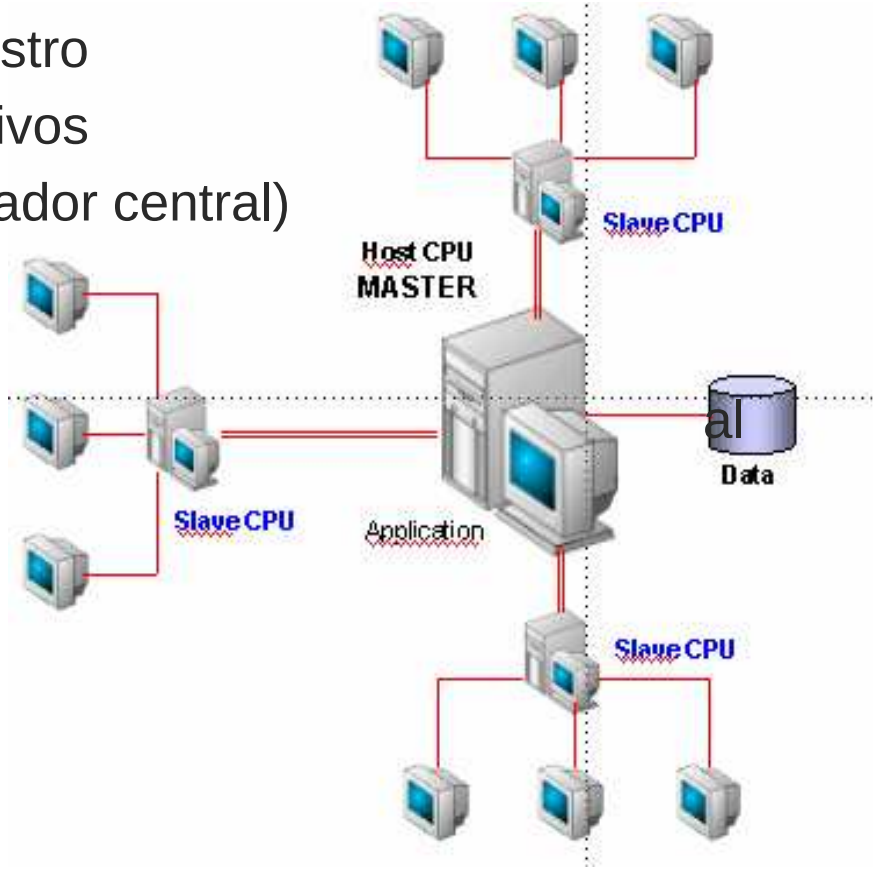
Objetivos y Beneficios

- **Compartir recursos e información**
- **Mejor performance**
  - SETI / NOW
- **Mayor fiabilidad y disponibilidad**
- **Escalabilidad**
  - Crecimiento por incrementos
- **Economía**
  - Clusters vs. Mainframes
- **Distribución geográfica de recursos coherente**
  - Sistema Free-Flow de Akamai

Evolución de Arquitecturas

Maestro-Esclavo

- Un dispositivo o proceso maestro controla a uno o más dispositivos o procesos esclavos (coordinador central)
- La dirección de mando es siempre es del maestro esclavo



Evolución de Arquitecturas

Maestro-Esclavo (Cont…)

- La arquitectura provee una sencillez para programar la concurrencia, pues una máquina solamente se encarga de coordinar a las demás.

- **Consideraciones**
  - a) El maestro representa un único punto de falla.
  - b) Los mensajes son generados siempre desde el maestro al esclavo y éste retorna los resultados.
- **Ejemplo**: SETI

Evolución de Arquitecturas

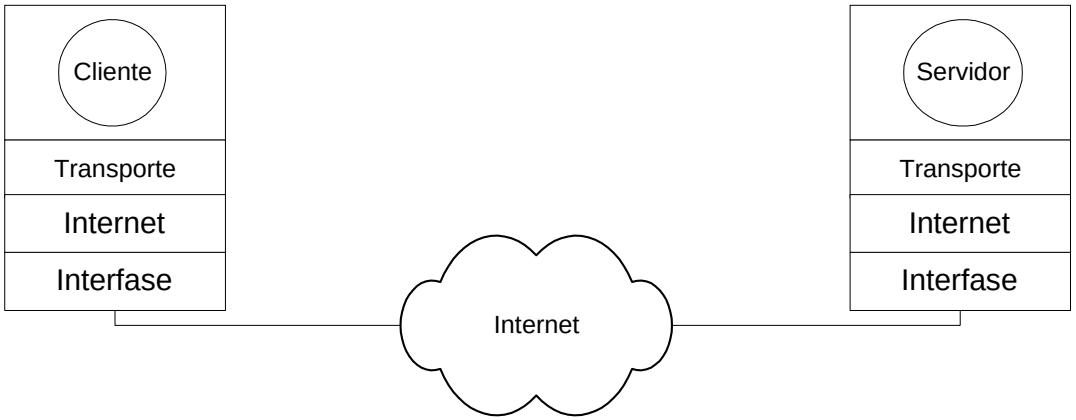
File-Sharing

- Surge con la aparición de las redes locales y las PCs
- Un “Servidor de Archivos” almacenaba los archivos de una organización
- Las aplicaciones solicitaban datos de algún archivos (todo o partes)
- Soportado generalmente por el SO
- **Ejemplos:**
  - WFW y LAN Manager de Microsoft
  - Netware de Novell
  - NFS y Samba para Unix

Evolución de Arquitecturas

Modelo Cliente/Servidor

- **Arquitectura que plantea la división de la funcionalidad para un servicio determinado**



- Clientes y servidores son software (procesos) que pueden correr bajo una misma CPU o no.
- Paradigma request/response
- Protocolos asimétricos

Modelo Cliente/Servidor

Cliente

- Es un **programa de aplicación** que ejecutan los usuarios cuando necesitan acceso a recursos remotos.
- Es **interactivo**, dado que se ejecuta en primer plano.
- **Corre de forma local** en la computadora del usuario.
- **Es activo** dado que inicia las conexiones con un proceso servidor.
- Es un **proceso temporal,** dado que su vida se limita a las necesidades de interacción del cliente
- Un pedido (request) indica **qué** recurso se desea

Modelo Cliente/Servidor

Servidor

- Es un **programa de aplicación** que provee un servicio específico.
- **No es interactivo**, dado que generalmente corre en segundo plano.
- Generalmente se lo activa cuando el sistema se inicializa (demonio/servicio).
- Corre en una computadora con recursos compartidos entre varios procesos del mismo tipo.
- Espera de **forma pasiva** por pedidos de conexiones
- Recibido un request, resuelve **cómo** satisfacelo

Modelo Cliente/Servidor

- **Ejemplos**

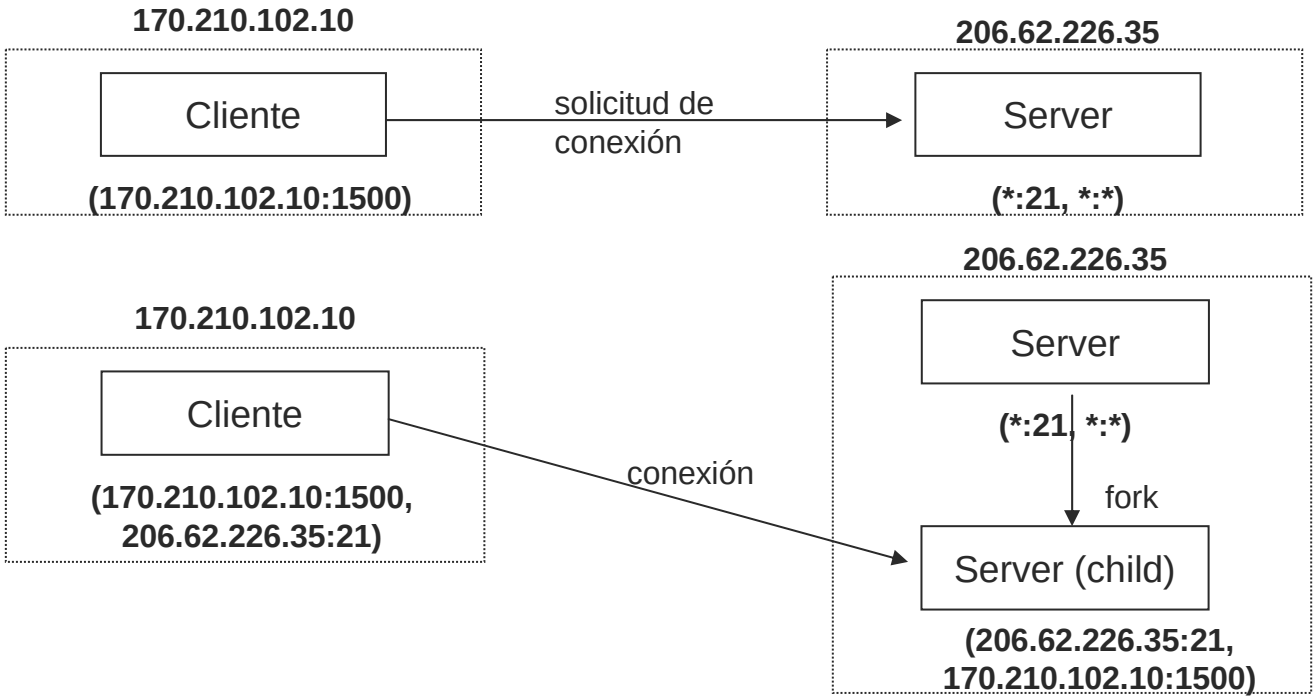
Servidor	Cliente
Graphic server	X Window
DBMS (MySQL)	Clientes SQL/App
Exchange	Eudora/Outlook
DNS (BIND)	Resolver (Cliente), Dig

El modelo de DB con MS-Access es Cliente/Servidor?

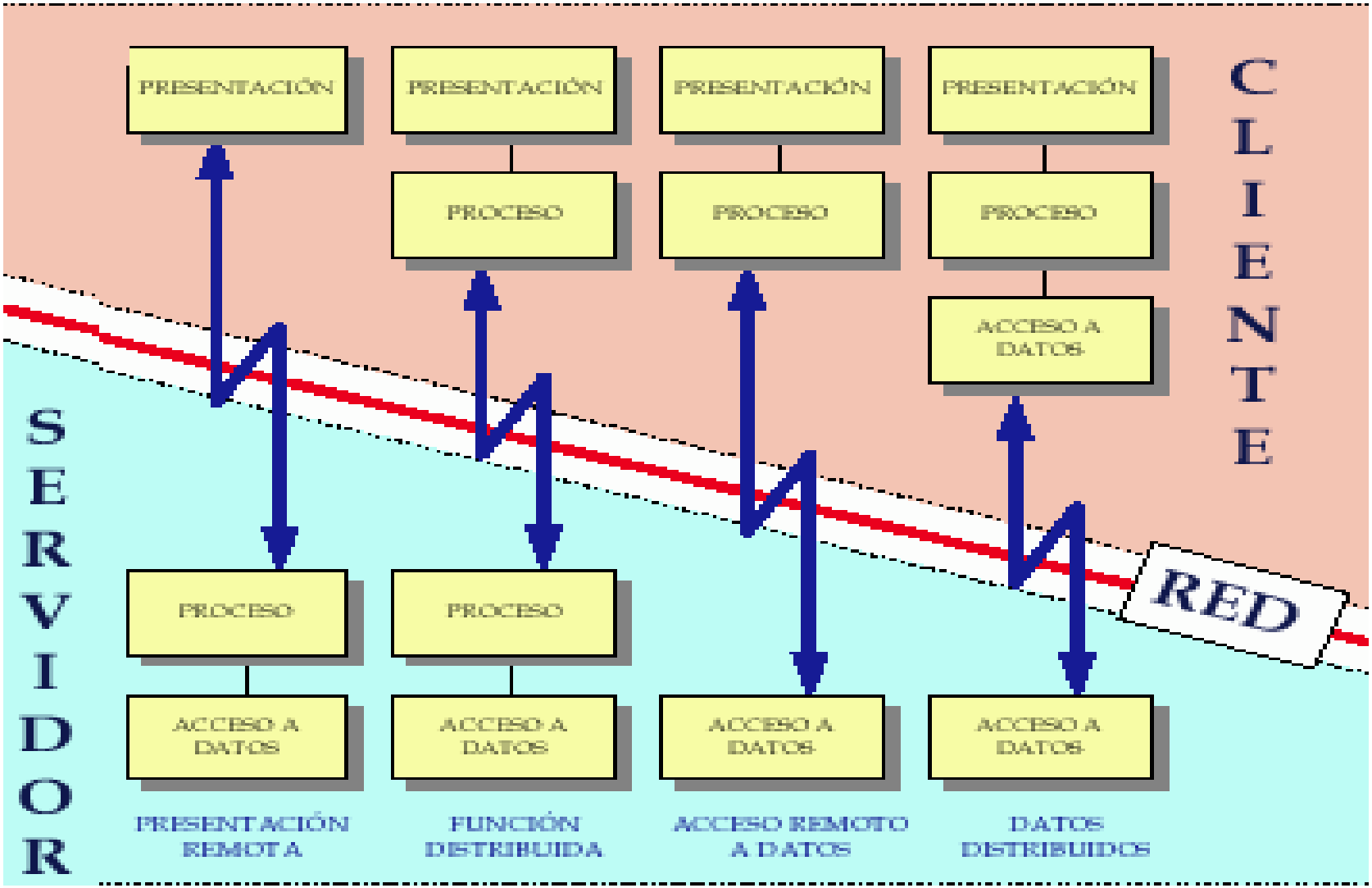
Modelo Cliente/Servidor

Tipos de Servidores

- **Secuenciales o iterativos**
- **Concurrentes (mediante proc. hijos ó multihilos)**



Grados de distribución de funciones en el Modelo C/S



Arquitecturas multi-nivel

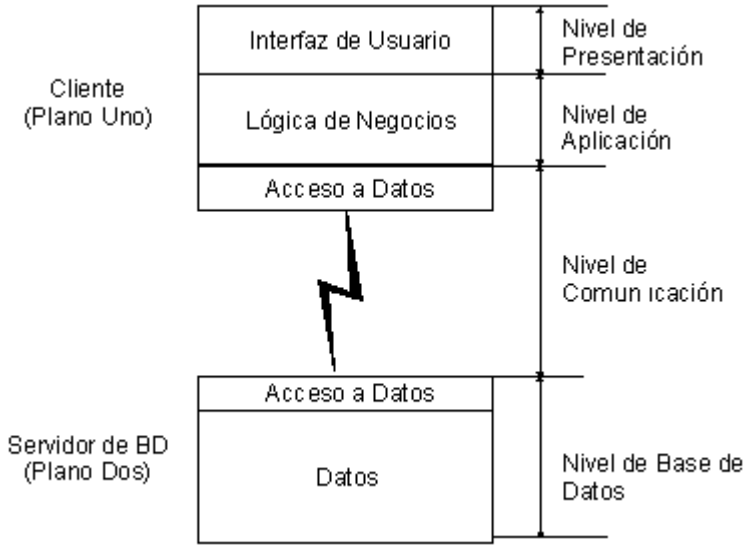
- **Se basa en la separación de funciones**
  - **Nivel de Presentación**: Agrupa a todos los elementos asociados al componente Cliente
  - **Nivel de Aplicación**: Agrupa a todos los elementos asociados al componente Servidor
  - **Nivel de comunicación**: Agrupa a todos los elementos que hacen posible la comunicación entre los componentes Cliente y Servidor
  - **Nivel de base de datos**: Agrupa a todas las actividades asociadas al acceso de los datos

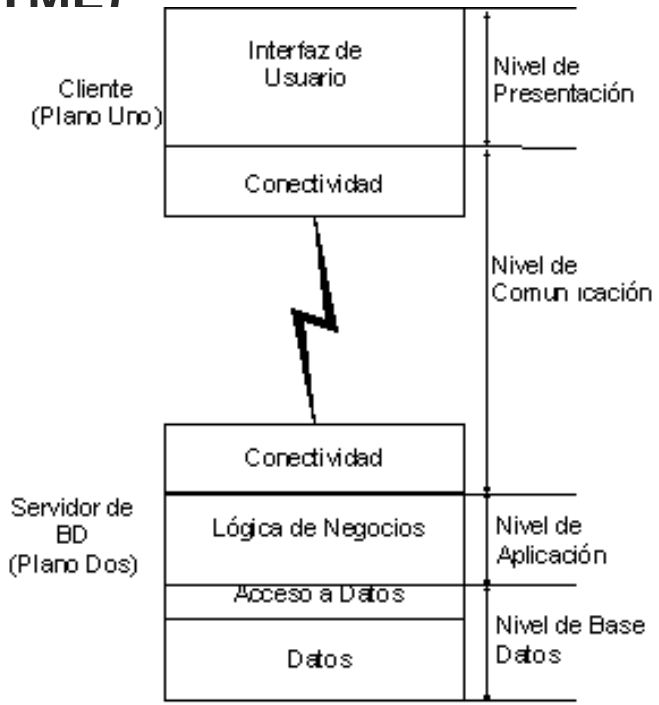
Arquitectura de un nivel (one-tier)

- Todas las operaciones las realiza un equipo central
- La terminal solo servía para E/S de datos
- Ejemplo:
  - Mainframes

Arquitectura de dos niveles (two-tiers)

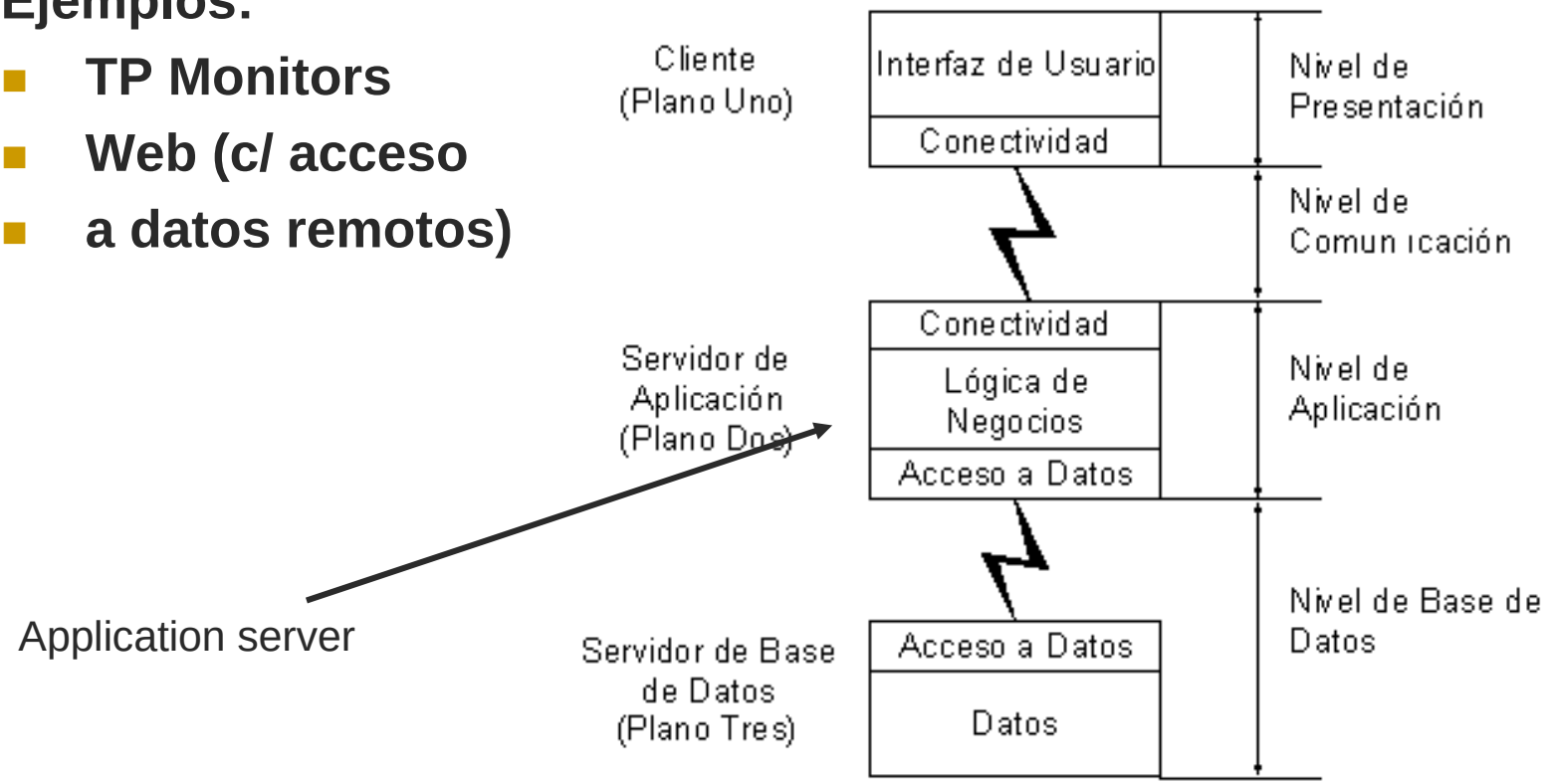
- Hay procesamiento en el C y en el S
- Ejemplos:
  - Aplicación de negocios (C) que consulta una BD
  - Web (servidor de archivos HTML)





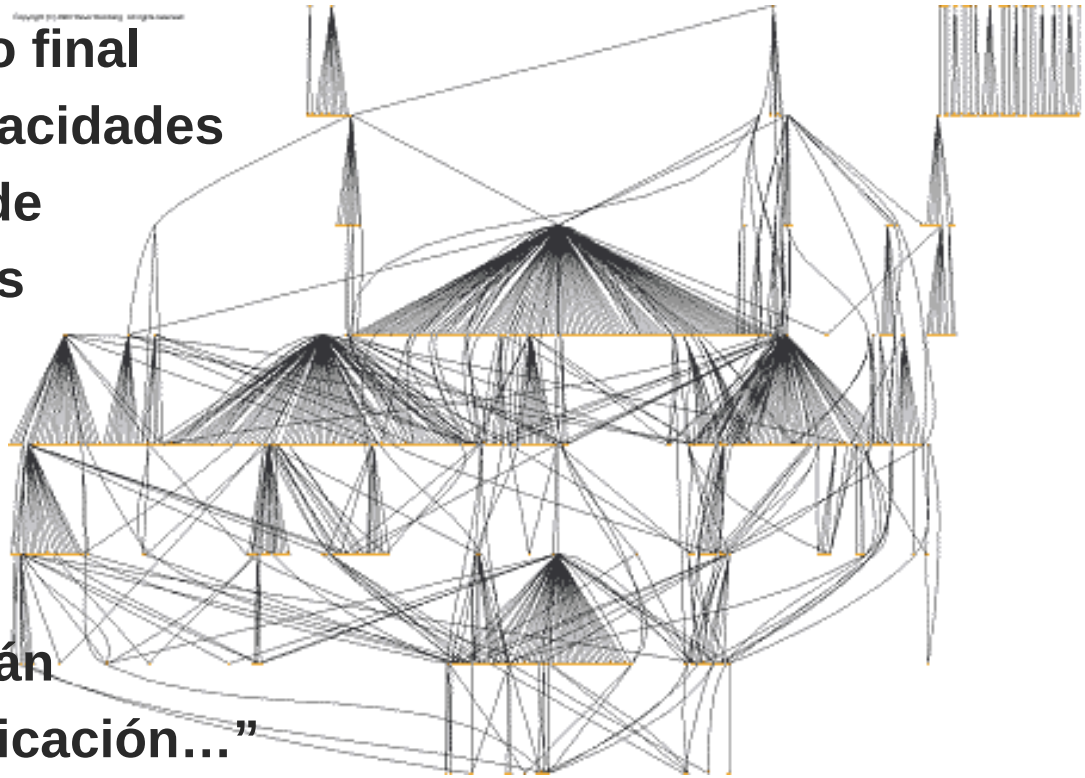
Arquitectura de tres niveles (three-tiers)

- Existen niveles adicionales
- Generalmente cliente ligeros
- Ejemplos:
  - TP Monitors
  - Web (c/ acceso
  - a datos remotos)



El Modelo Compañero a Compañero (peer-to-Peer)

“Las estaciones de usuario final comenzarán a agregar capacidades a la red, como servidores de archivos, cachés ó motores de cómputo…
Como resultado de estas capacidades adicionales, los servicios de la red, como la web, no estarán confinados a una única ubicación…”



John Postel, The GRID: Blueprint for a New Computing Infrastructure, 1997

El Modelo Compañero a Compañero (peer-to-Peer)

“Modelo de comunicación donde todos los nodos poseen los mismos roles, es decir, son todos nodos pares que pueden actuar en el sistema tanto como clientes como servidores (servent)”

- Nodos con capacidades y responsabilidades equivalentes (simétricas).
- Característica fundamental: Intercambio directo entre nodos pares.

Características (deseables) de los sistemas P2P

- Los nodos deben ser autónomos, y regulan su grado de participación en la red y los recursos que brindan
- No necesitan tener una vista global del sistema
- El comportamiento global surge de interacciones individuales
- Pueden operar con esquemas de nombres y direcciones alternativos al DNS
- Los nodos pueden entrar y salir de la red arbitrariamente (el sistema de debe reconfigurar)
- La ubicación de los recursos en la red es dinámica, depende del estado en un momento del tiempo (lookup service)
- El ambiente (la red) es heterogéneo

Surgimiento de los Sistemas P2P

- **En los 2000's**
  - Resurge con aplicaciones de intercambio de archivos y mensajería (Killer app → Napster).

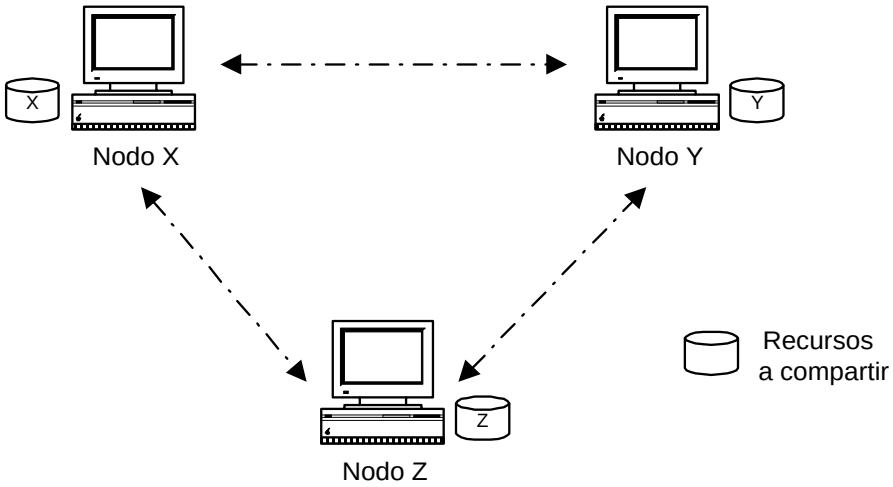
  - Cambio del modelo de comunicaciones
    - **Usuarios**

P2P involucra "una clase de aplicaciones que toman ventajas de recursos - almaceamiento, ciclos de CPU, contenido, presencia humana - disponible en los "costados" (edges) de Internet"

Clay Shirky, The Accelerator Group

Tipos de Redes P2P

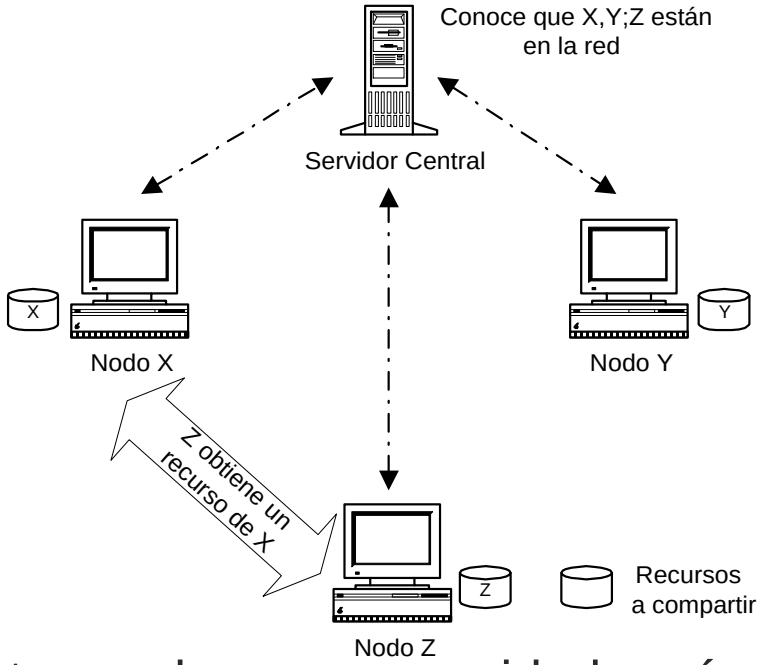
Redes puras



No hay nodos con capacidades funciones especiales

Ejemplos: **Gnutella y Freenet**

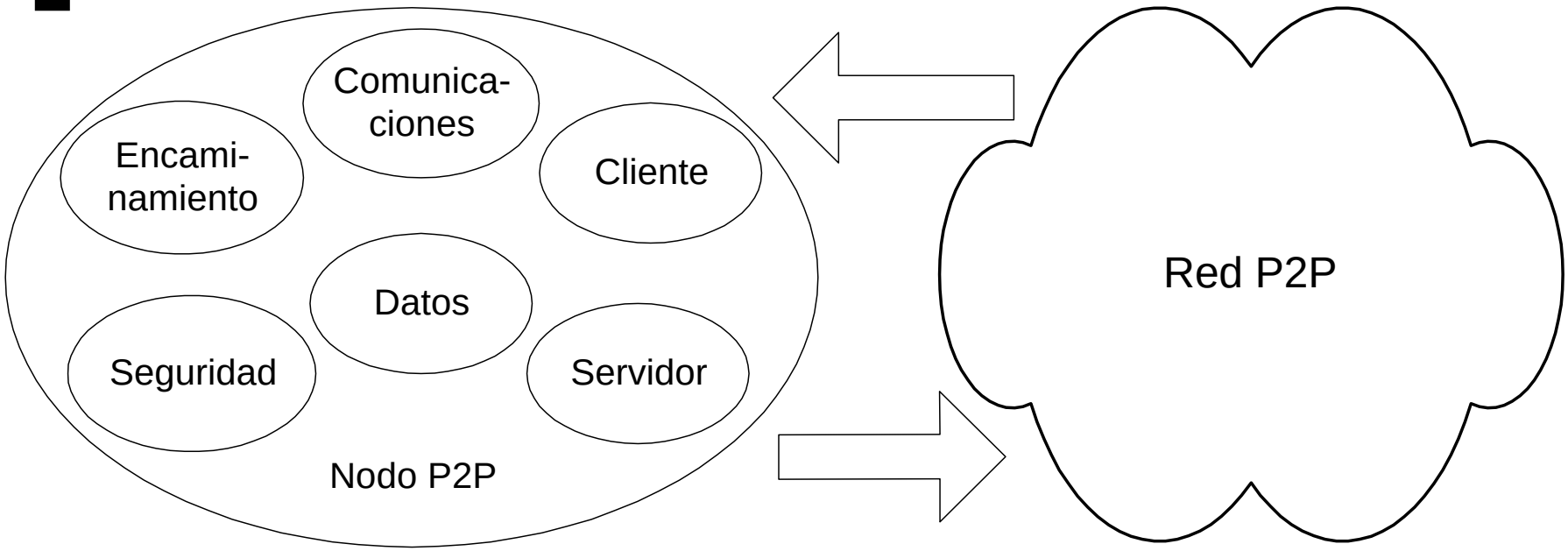
Redes híbridas



Existen nodos con capacidades ó funciones especiales

Ejemplo: **Napster**

Arquitectura de un nodo P2P



Datos: Mantiene info. de estado

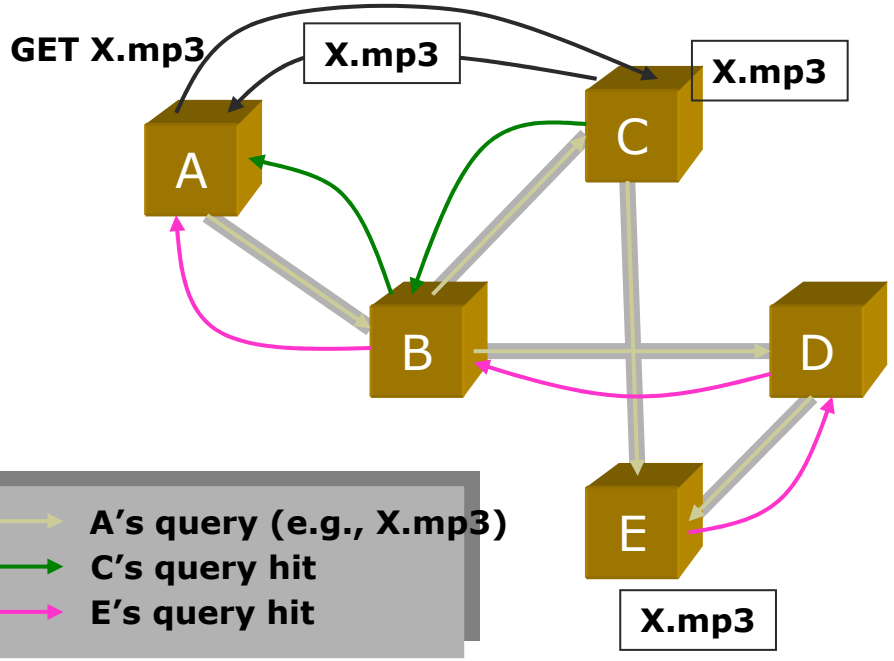
Comunicaciones: Soporte de conexiones entrantes y salientes (mantenimiento en la red)

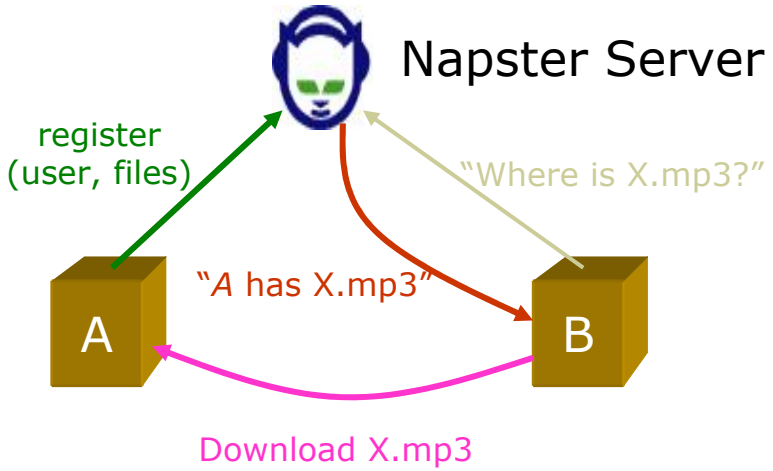
Seguridad: Autenticación, cifrado

Encaminamiento: Propaga mensajes propios y de terceros de acuerdo a una estrategia

Mecanismos de Comunicación

- **Servicio de directorio**
- **Reenvío por difusión**

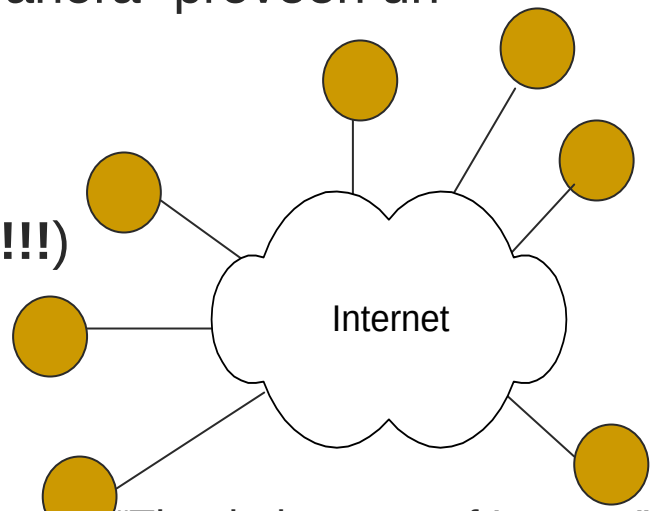




- **Ruteo**
  - **Por recursos (Chord/CAN)**
  - **Por contenido (Neurogrid/JXTA Search)**

P2P en la Internet actual

- **Los sistemas emergentes están destinados a los usuarios “hogareños”**
  - Recursos computacionales importantes (CPU/Disco)
  - Básicamente, son consumidores que “ahora” proveen un recurso/servicio
  - Conectividad variable (relativa)
  - Ancho de banda limitado (**hoy menos!!!**)
  - Direcciones de red dinámicas
  - No figuran en DNSs



“The dark matter of Internet”
Clay Shirky

P2P en la Internet actual

- **Pero...**
  - **Ancho de banda asimétrico** (Ej. ADSL). Si el “caño” de subida se congestiona (por servir datos), afecta al de bajada.
  - La infraestructura de la red (respecto de los usuarios) está optimizada para “clientes”, no “servidores”. Es ineficiente para P2P?
  - **Además: “Generar no es lo mismo que publicar”**
    - Hoy los usuarios no necesitan generar contenido para poder publicar. **Por qué? Napster** era un ejemplo!

P2P en la Internet actual

- **Principo en la Internet original**

“Si un host puede alcanzar a cualquiera en la red, éstos también pueden alcanzar a tal host”

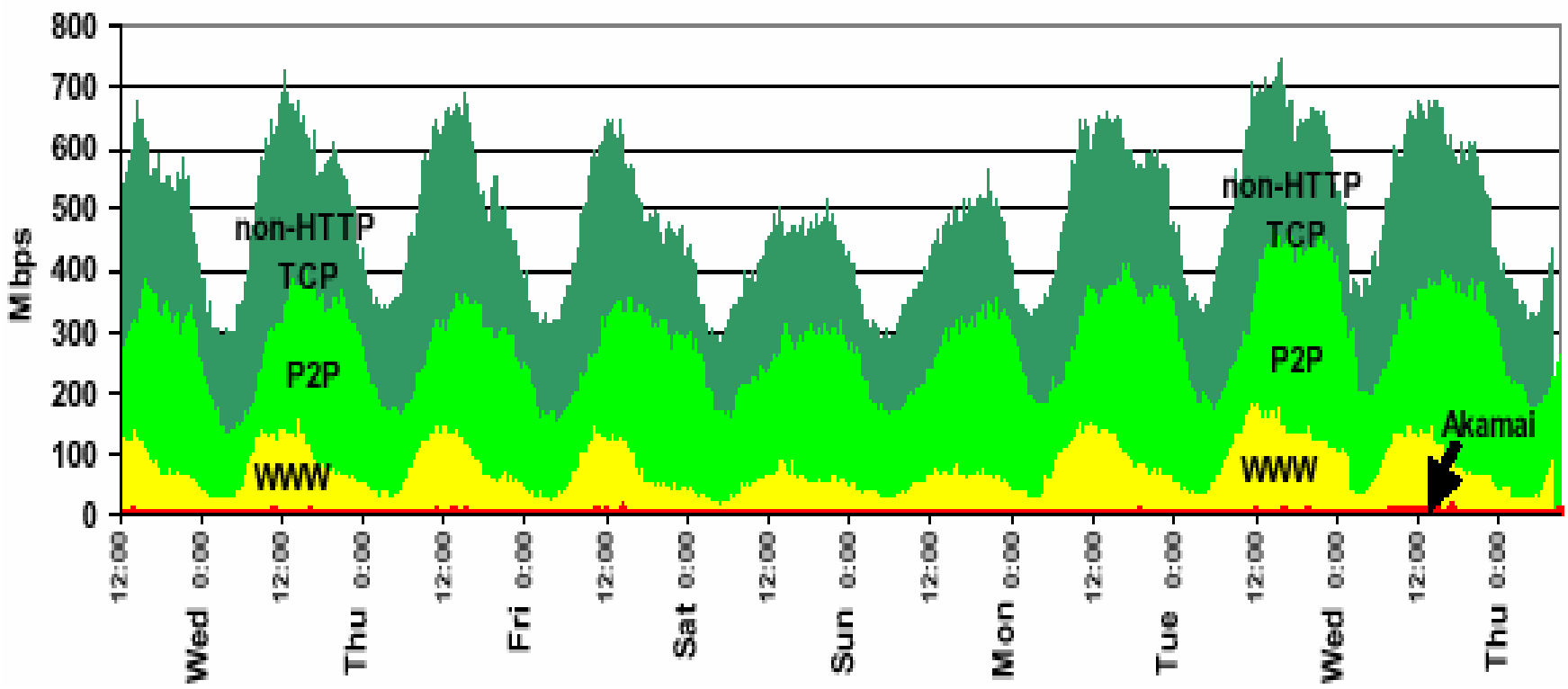
- **En la Internet actual: Esa afirmación ya no es verdadera !!**

Firewalls “parten” las redes – Dirs Dinámicas - Dirs Privadas y NAT

- Se intentan resolver el problema de la seguridad (con millones de usuarios), pero relegaron a un conjunto de hosts a un segundo plano: SOLO Clientes!!!
- Las aplicaciones P2P pueden ayudar a revertir esta situación

P2P en la Internet actual - Estudio:

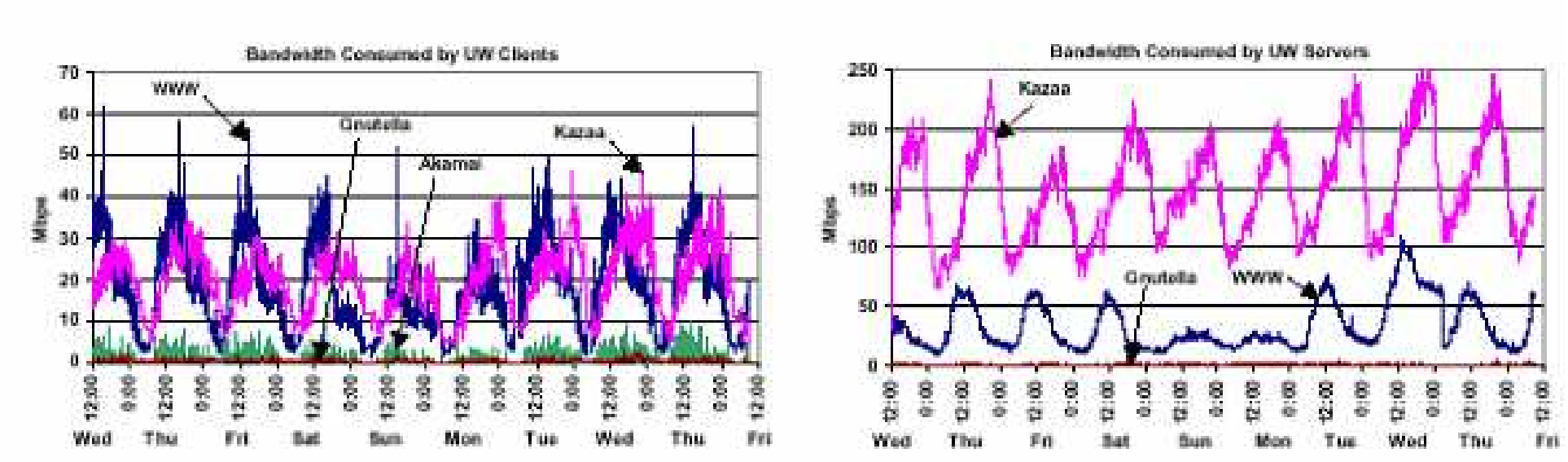
Medición del Tráfico en la Universidad de Washington (aproximadamente 60.000 alumnos) durante 1 semana



"An Analysis of Internet Content Delivery Systems". Saroiu, Gummadi, Dunn, Gribble, Levy

P2P en la Internet actual

Medición del Tráfico en la Universidad de Washington (aproximadamente 60.000 alumnos) durante 1 semana



"An Analysis of Internet Content Delivery Systems". Saroiu, Gummadi, Dunn, Gribble, Levy

P2P en la Internet actual

Conclusiones

- El tráfico P2P excede a HTTP en un factor de 3. Como fracción de todo el tráfico TCP, Kazaa cuenta el 36.9% de bytes transferidos, mientras que la web, el 14,3%
- Un pequeño número de clientes y servidores es responsable de la mayoría del tráfico P2P. Los 200 top (de 4644 clientes) cuentan el 50% de los bytes descargados en Kazaa
- Solo 600 nodos externos a UW proveen el 26% de los bytes transferidos
- Cada nodo P2P consume un BW significativo en ambas direcciones, predominando los uploads. Los 4,644 nodos Kazaa peers proveen 13.573 TB a 611,005 nodos externos, mientras que solicitan 1.78 TB desde 281,026 nodos

“An Analysis of Internet Content Delivery Systems”. Saroiu, Gummadi, Dunn, Gribble, Levy

P2P en la Internet actual

- **Aspectos a desarrollar**
  - **En los sistemas P2P**
    - Sistemas de nombres
    - Ruteo
    - Búsquedas
    - Interoperabilidad
    - Seguridad y confianza
    - Estándares!!!
  - **En la Internet**
    - Modelos de tráfico