

Tema-1-Introduccion-a-la-Ingenie...



fer_luque



Ingeniería de Servidores



3º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada

Ya puedes imprimir desde Wuolah

Tus apuntes sin publi y al mejor precio



Tema 1: Introducción a la Ingeniería de Servidores

1 ¿Qué es un servidor?

Sistema informático (SI)

Conjunto de elementos **hardware**, **software** y **peopleware** interrelacionados entre sí que permite obtener, procesar y almacenar información.

- Hardware: conjunto de componentes *físicos*
- Software: conjunto de componentes *lógicos*
- Peopleware: conjunto de *recursos humanos*

Clasificaciones de los SI

Según paralelismo:

- SISD: *Single Instruction Single Data*
- SIMD: *Single Instruction Multiple Data*
- MISD: *Multiple Instruction Single Data*
- MIMD: *Multiple Instruction Multiple Data*

Según su uso:

- De **uso general**: PC de sobremesa o portátil
- De **uso específico**: Sistemas empotrados (*embedded systems*), servidores (*servers*)

Según la arquitectura de servicio:

La arquitectura de servicio describe cómo se establece la interacción entre el servidor y sus clientes

- **Sistema aislado**: No interactúa con otros sistemas. Arquitectura monolítica en la que no existe distribución de la información
- **Arquitectura cliente/servidor**: Modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Tienen dos tipos de nodos/niveles en la red:
 - Clientes: remiten solicitudes
 - Servidores: reciben solicitudes
- **Arquitectura cliente/servidor de varios niveles**: El servidor se subdivide en varios niveles de microservidores más sencillos con diferentes funcionalidades en cada nivel. Mejora la distribución de carga entre los diversos servidores, es más escalable, pone más carga en la red y es más difícil de programar y administrar.
- **Arquitectura cliente-cola-cliente**: Habilita a todos los clientes para desarrollar tareas semejantes interactuando cooperativamente para realizar una actividad distribuida, mientras que el servidor actúa como una cola que va capturando las peticiones de los clientes y sincronizando el funcionamiento del sistema

Sistemas empotrados (*embebbed systems*)

Sistemas informáticos **acoplados** a otro dispositivo o aparato, diseñados para realizar una o algunas funciones dedicadas, frecuentemente con fuertes restricciones de tamaño, tiempo de respuesta (sistemas de tiempo real), consumo y coste.

Servidores (*servers*)

Son SIs que, formando parte de una red, proporcionan servicios a otros sistemas informáticos denominados clientes.

No tienen por qué ser máquinas de grandes proporciones, pueden ser desde un ordenador de gama baja hasta un conjunto de *clusters* de computadores.

Tipos de servidores

- **Web:** Documentos HTML, imágenes, texto... distribuye contenido a clientes que lo soliciten en la red.
- **De archivos:** Acceso remoto a archivos almacenados en él
- **De base de datos:** Provee servicios de base de datos a otros programas o computadoras
- **De comercio-e:** Cumple o procesa transacciones comerciales, valida el cliente y genera un pedido al servidor de base de datos
- **De correo-e:** Almacena, envía, recibe, re-enruta y realiza otras operaciones relacionadas con e-mail para los clientes de la red
- **DHCP:** Asigna dinámicamente una dirección IP y otros parámetros de configuración de red a cada dispositivo que entra en una red
- **DNS:** Devuelve la dirección IP asociada a un nombre de dominio
- **De impresión:** Controla una o más impresoras y acepta trabajos de impresión de otros clientes

2 Fundamentos de Ingeniería de Servidores

Diseño, configuración y evaluación de un Servidor

Recursos físicos, lógicos y humanos

Placa base, memoria, microprocesador, fuente de alimentación, periféricos, SO, aplicaciones, conexiones de red, cableado, refrigeración, administración...

Requisitos funcionales (los más importantes)

Prestaciones, disponibilidad, fiabilidad, seguridad, extensibilidad, escalabilidad, mantenimiento y coste

Prestaciones (*Performance*)

Medida o cuantificación de la velocidad con la que se realiza una determinada carga o cantidad de trabajo.

Medidas de prestaciones:

- **Tiempo de respuesta** (*response time*) o **latencia** (*latency*): Tiempo total entre que se solicita una tarea y finaliza la misma
- **Productividad** (*throughput*) o **ancho de banda** (*bandwidth*): Cantidad de trabajo realizado por el servidor o por un componente del mismo por unidad de tiempo

Soluciones

¿Qué afecta?	¿Cómo mejorarlo?
Componentes hardware	Actualización de componentes
Sistema Operativo	Ajuste
Aplicaciones	Sintonización

Disponibilidad (*Availability*)

Un servidor está disponible si se encuentra en estado operativo (respondiendo a las peticiones de los clientes). Medidas:

- **Tiempo de inactividad** (*Downtime*): cantidad de tiempo en el que el sistema no está disponible. Puede ser:
 - Planificado: Actualizaciones
 - No planificado: Fallos, anomalías...

Soluciones

SO que permitan actualizaciones sin reiniciar, *hot-plugging*, *hot-swapping*, *RAID 1*, sistemas redundantes de alimentación, sistemas de red redundantes, sistemas completos redundantes con distribución de carga (*load balancing*)

Fiabilidad (*Reliability*)

Es fiable cuando desarrolla su actividad sin errores. Medida:

- **MMTF** (*Mean Time to Failure*): tiempo medio hasta que ocurre un error.

Soluciones

Sumas de comprobación, comprobación de recepción de paquetes de red...

Seguridad

Un servidor debe ser seguro ante:

- La incursión de individuos no autorizados (confidencialidad)
- La corrupción o alteración no autorizada de datos (integridad)
- Las interferencias (ataques) que impidan el acceso a los recursos

Soluciones

Autenticación segura, encriptación de datos, antivirus, parches de seguridad actualizados, cortafuegos...

Extensibilidad-expansibilidad

Hace referencia a la facilidad que ofrece el sistema para **aumentar** sus características o recursos

Ya puedes imprimir desde Wuolah

Tus apuntes sin publi y al mejor precio

Soluciones

Bahías o conectores libres, SO modulares, interfaces de E/S estándar...

Escalabilidad

Facilidad que ofrece el sistema para poder **aumentar** de forma **significativa** sus características o recursos para enfrentarnos a un aumento **significativo** de la carga

Nota: Todos los sistemas escalables son extensibles pero no a la inversa

Soluciones

Cloud computing + virtualización, servidores modulares, arquitecturas distribuidas, *Storage Area Networks* (SAN), programación paralela

Mantenimiento (*Maintenance, support*)

Hace referencia a todas las acciones que tienen como objetivo prolongar el funcionamiento correcto del sistema.

Soluciones

SO con actualizaciones automáticas, *Cloud Computing* (proveedor se encarga de chequeo de componentes y de actualización), automatización de copias de seguridad y de tareas de configuración y administración

Coste

Tenemos que adaptarnos al presupuesto, teniendo en cuenta el coste del hardware, del software y del mantenimiento, personal, proveedores de red, alquiler del local, consumo eléctrico...

Soluciones

Cloud computing, software de código abierto, eficiencia energética...

3 Introducción a la comparación de características entre servidores

Motivación: comparación de prestaciones

El computador de mejores prestaciones (el más rápido), para un determinado conjunto de programas, será aquel que ejecuta dicho conjunto de programas en el tiempo más corto.

En lugar de tiempos absolutos **se expresan en términos relativos**: tantas veces más rápido o tanto por ciento mejor:

Ganancia en velocidad (*speedup* o *aceleración*)

Se define el *speedup* de la máquina A respecto a la máquina B como:

$$S_B(A) = \frac{v_a}{v_b} = \frac{t_b}{t_a}, \text{ siendo } v_a \text{ y } v_b \text{ las velocidades de ejecución de las dos máquinas.}$$

En porcentaje (incremento en velocidad):

$$\Delta v_{a,b}(\%) = (S_B(A) - 1) \times 100$$



Te enviamos los apuntes a casa

Recogelos en tu copistería más cercana



WUOLAH

Relación prestaciones/coste

$$\frac{\text{Prestaciones}_A}{\text{Coste}_A} = \frac{v_A}{\text{Coste}_A}$$

Relación entre prestaciones de diferentes máquinas:

$$\frac{\frac{\text{Prestaciones}_A}{\text{Coste}_A}}{\frac{\text{Prestaciones}_B}{\text{Coste}_B}} = \text{Relación de prestaciones de A respecto a B}$$

Introducción a la mejora del tiempo de respuesta de un servidor

Mejora del tiempo de ejecución de un proceso

Una de las formas más habituales de mejorar el tiempo de respuesta de un servidor consiste en el reemplazo de un componente. Para evaluar la mejora se supone que en el servidor sólo se ejecuta un único proceso monohebra.

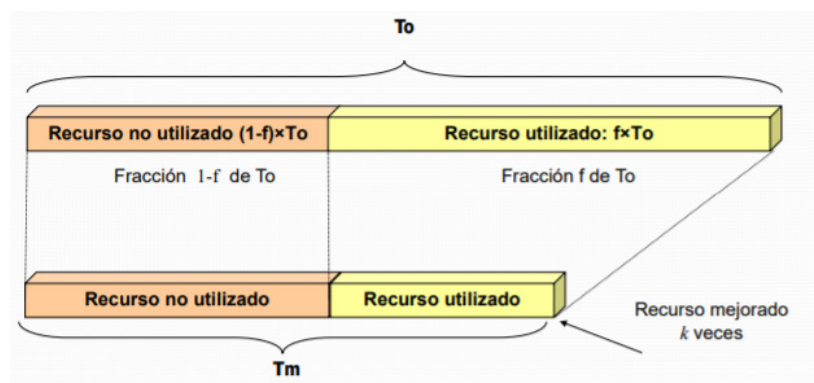
Supongamos que:

- El servidor tarda un tiempo $T_{original} = T_O$ en ejecutar el proceso
- Mejoramos el sistema reemplazando uno de sus componentes por otro k veces más rápido
- Este componente se utilizaba durante una fracción f del tiempo T_O ($f = 1$ supone el 100% del tiempo)
- El nuevo tiempo con el nuevo componente se llama $T_{mejorado} = T_m$

¿Cuál es la ganancia en velocidad del sistema después de mejorar k veces un componente?

Ley de Amdahl

Deducción:

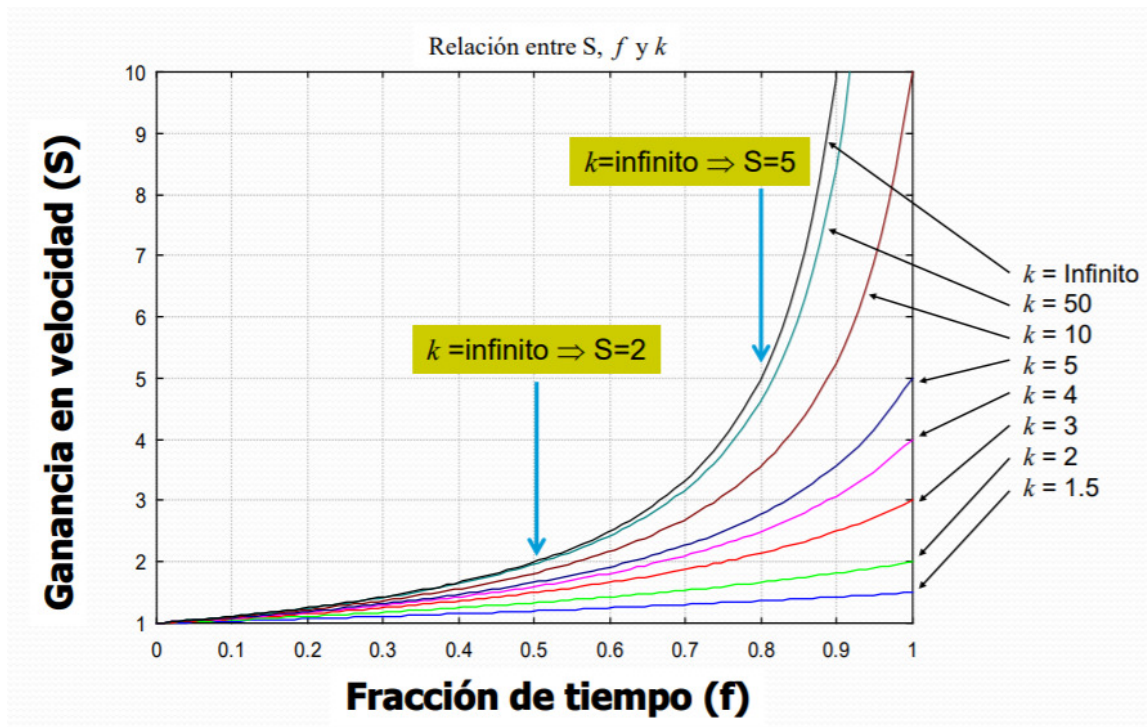


$$S_{original(mejorado)} = \frac{v_m}{v_O} = \frac{T_O}{T_m} = \frac{T_O}{(1-f) \times T_O + \frac{f \times T_O}{k}} = \frac{1}{1-f + \frac{f}{k}}$$

- Si $f = 0 \Rightarrow S = 1$: no hay ninguna mejora en el sistema
- Si $f = 1 \Rightarrow S = k$: el sistema mejora tantas veces como el componente
- Si $k \rightarrow \infty \Rightarrow S \rightarrow \lim_{k \rightarrow \infty} S = \frac{1}{1-f}$

Ejemplo (OneNote):

Relación entre S , f y k



Interesa mejorar el componente cuando la fracción de tiempo sea más alta.

Esta ley estima la ganancia para un sólo trabajo (un hilo).

Generalización de la ley de Amdahl

$$S = \frac{1}{(1 - \sum_{i=1}^n f_i) + \sum_{i=1}^n \frac{f_i}{k_i}}$$

