

# **Tema 1.- Introducción**

**Tecnologías de los Sistemas de Información en la Red**

# Índice

- Definición de **Sistema Distribuido**
  - Formado por distintos nodos interconectados en red
  - Ofrece imagen de máquina única
- Relevancia y áreas de aplicación
- Evolución histórica
  - etapa actual = computación en la nube
- Implementación servidores

# Concepto de Sistema Distribuido

- Conjunto de agentes autónomos. Cada agente;
  - es un proceso secuencial con su propio ritmo
  - tiene su propio estado independiente
- Los agentes interactúan (cooperan). Opciones:
  - intercambio de mensajes
  - memoria compartida
- En la práctica, un sist. distribuido es un sistema en red

# Relevancia de los Sist. Distribuidos

- Área en evolución continua
  - Entorno de computación actual distribuido e interconectado (web)
    - Infinidad de nodos conectados
    - Infinidad de servicios remotos
- Rama de los sistemas concurrentes
  - Asumimos la base proporcionada por CSD

# Relevancia ...

- 👍 Mejora el rendimiento
  - Divide un tarea compleja en subtareas, y asigna cada una a un nodo diferente
- 👍 Mayor disponibilidad
  - Si falla un nodo, otros completan su tarea
- 👍 Compartición de recursos
  - Cada nodo tiene recursos asociados (ej.- impresoras, discos)
  - Otros nodos del sistema pueden acceder a ellos

# Relevancia ...

- Reto
  - conseguir que los ordenadores cooperen entre sí y colectivamente desarrollen trabajo útil
  - Crear subsistemas capaces de proporcionar servicios robustos
- Ejemplos a gran escala (millones de usuarios)
  - Servicio de búsqueda de Google
  - Compartición de ficheros en DropBox
  - Simulación de fármacos contra el cáncer

# Principales áreas de aplicación

1. World Wide Web
2. redes de sensores
3. Internet of things
4. Computación cooperativa
5. Clustes altamente disponibles
6. Cloud Computing

# Aplicación: WWW

- Basada en el modelo Cliente/Servidor
  - cliente = navegador web (solicita/recibe documento)
  - servidor.- responde a peticiones de docs
- Inicialmente diseñado para compartir documentos
  - los documentos pueden formarse a partir de las respuestas de varios servidores
  - Posteriormente extendido para soportar peticiones sobre servicios




# Aplicación: Redes de Sensores

- Aparecen equipos de muy bajo coste
  - Mini-ordenadores de propósito específico
  - Empotrados en disp. de uso cotidiano
    - ej.- en algunos electrodomésticos
  - Contienen sensores (humedad, temperatura, ..)
- Muchas aplicaciones potenciales
  - Vigilancia, detección de desastres, monitorización consumo, ...

# Aplicación: Internet of Things

- Generalización de las redes de sensores
- Facilita la interoperabilidad de los dispositivos
  - Los dispositivos pueden interactuar entre sí
  - *Los dispositivos pueden alterar su entorno*
- Nuevos escenarios
  - Ciudades inteligentes
  - Automatización (construcción, fabricación, ...)
  - Cuidado médico informatizado

# Aplicación: Internet of Things

 imag internet of things

# Aplicación: Comp. Cooperativa

- Recursos en internet infrautilizados
  - Muchos PCs pasan horas inactivos
  - Pueden suscribirse para recibir tareas
    - instala soft. cliente, se registra en un servidor
- El servidor
  - Divide tarea grande en subtareas
  - Distribuye subtareas entre los clientes
  - Recopila los resultados

# Aplicación: Clusters ↑ Disponibles

- Ciertas entornos necesitan ↑ disponibilidad
  - Bancario, Empresarial, Asistencia médica, ...
    - Deben garantizar integridad información
    - Deben mantener disponibilidad servidores
- Hechos:
  - Todo dispositivo falla en algún momento
  - Pero fallo simultáneo es muy improbable
- Solución.- replicación (y reconfiguración tras fallo)

# Aplicación: Cloud Computing (CC)

- Queremos construir y facilitar **servicios de forma sencilla, eficiente, y económica**
- Implantados mediante clusters ↑ disponibles
  - Un centro de cómputo dedicado es caro
    - adquisición, administración, mantenimiento
  - Para ↓ costes compartimos centros de cómputo externos (pago por uso)
    - Acceso al centro externo a través de internet
    - Posible por mejoras en redes y navegadores

# Aplic.: CC modelos de servicio

- **SaaS** (software como servicio)
  - Aplicaciones escalables como servicio
- **PaaS** (plataforma como servicio)
  - Automatiza la gestión de servicios elásticos (ciclo de vida, escalabilidad)
- **IaaS** (infraestructura como servicio)
  - Automatiza gestión de recursos
  - Usa virtualización

# Implementación servidor

- Un servidor recibe peticiones, las procesa, y retorna respuestas
- Para completar una petición puede necesitar pedir a su vez servicios a otros
- Por escalabilidad, un servidor debería aceptar varias peticiones simultáneas
  - Mientras gestiona una petición, debe ser capaz de aceptar otras
- Alternativas para implantar un servidor: **servidor concurrente, y servidor asíncrono**



# Servidor concurrente (multi-hilo)

- Cada petición servida por un hilo distinto
  - Todos los hilos comparten un estado global
  - Hay que evitar interferencias e interbloqueos
- 👍 cada hilo se planifica por separado (si se suspende no afecta al resto del servidor)
- 👎 control concurrencia → suspensión.  
Sobrecarga
- 👎 programación y depuración complejas
- Usado en Java y .NET

# Servidor Asincrónico

- programación asincrónica = dirigida por eventos
  - corresponde al modelo guarda/acción
    - guarda = evento
    - acción = callback del evento
- múltiples actividades, pero no acceden de forma concurrente al estado compartido
  - Las acciones preparadas para ejecución (cumplen la guarda) se encolan en orden FIFO

# Servidor Asíncrono ...

- 👍 No hay estado compartido: ↓ complejidad
- 👍 No soporta threads: ↓ sobrecarga ↑ escalable
- 👍 El funcionamiento real de un S.D. es dirigido por eventos: facilita razonar sobre el sistema
- 👎 Toda acción debe cuidar la gestión el estado
- 👎 No es fácil controlar el orden de encolado
- 👎 Todo el entorno debe ser asíncrono (ej. servicios del S.O. no bloqueantes)
- Usado en NodeJS y Async .NET