

---

# IIC2523

# Sistemas Distribuidos

— Hernán F. Valdivieso López —  
(2025 - 2 / Clase 02)

---

# Introducción a los Sistemas Distribuidos

## ¿Qué son? ¿Cuál es su objetivo? ¿Qué tipos hay?

# Temas de la clase

1. Sistema Distribuido
  - a. Definiciones
  - b. Distinción entre otros sistemas
  - c. Objetivos de un sistema distribuido
2. Tipos de Sistemas Distribuidos
  - a. Sistemas Distribuidos de Alto Rendimiento
  - b. Sistemas de Información Distribuidos
  - c. Sistemas Pervasivos

# Sistemas Distribuidos

Extenderemos la definición

Distinción entre otros tipos de  
sistemas

Objetivos de un sistema  
distribuido

---

# ¿Qué es un sistema distribuido?

*Es una **colección de sistemas informáticos** en red donde los procesos y recursos se extienden a través de **diferentes computadoras**, con el fin de mejorar el rendimiento de un sistema informático individual en términos de **fiabilidad, escalabilidad y eficiencia**.*

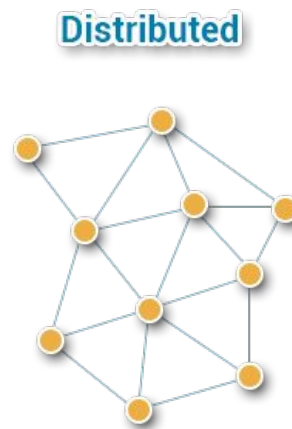
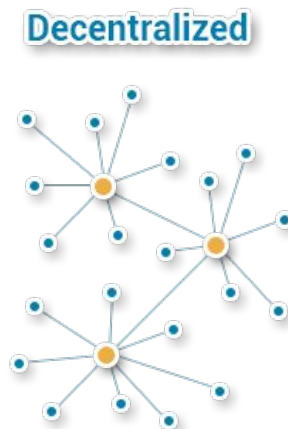
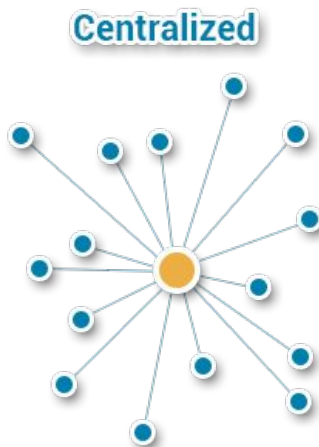
# ¿Qué es un sistema distribuido?

Es una **colección de sistemas informáticos** en red donde los procesos y recursos se extienden a través de **diferentes computadoras**, con el fin de mejorar el rendimiento de un sistema informático individual en términos de **fiabilidad, escalabilidad y eficiencia**.

Un sistema distribuido es una colección de computadoras independientes que **aparecen ante los usuarios del sistema como una única computadora**.

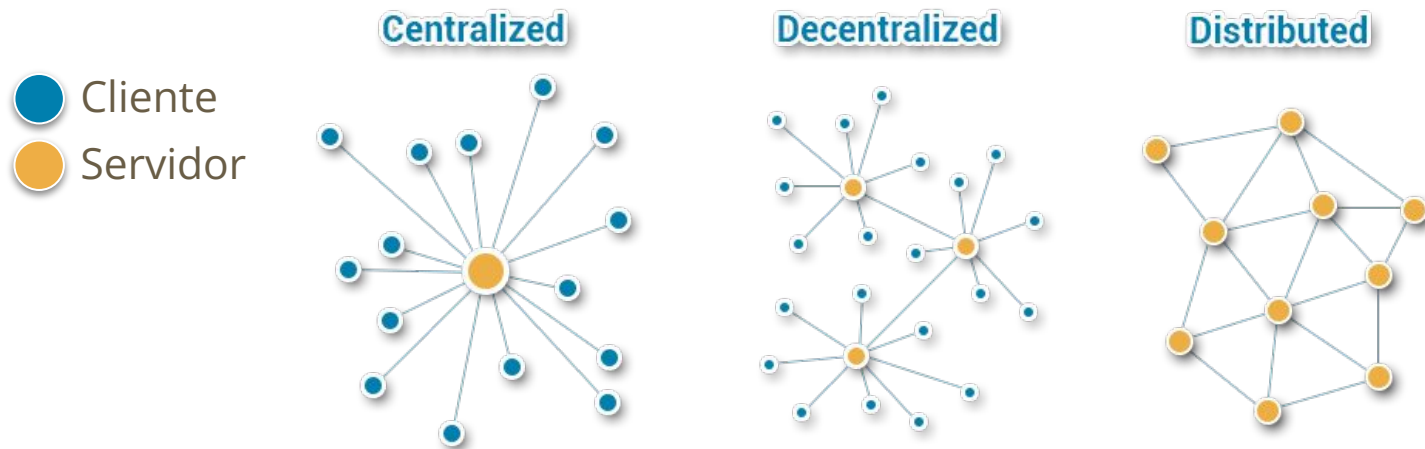
# ¿Cuál es la diferencia con un sistema descentralizado?

- ◆ No existe un consenso universal sobre la diferencia entre estos conceptos.
- ◆ Un enfoque clásico es según la existencia de una **autoridad** permanente.
  - ◆ En el descentralizado, la autoridad se reparte entre varios nodos; en el distribuido, todos los nodos pueden tener dicha autoridad.



# ¿Cuál es la diferencia con un sistema descentralizado?

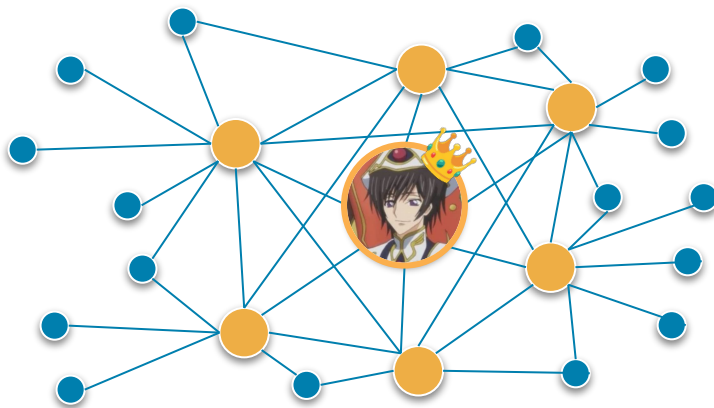
- ◆ No existe un consenso universal sobre la diferencia entre estos conceptos.
- ◆ Un enfoque clásico es según la existencia de una **autoridad** permanente.
- ◆ Otro enfoque es la **robustez** del sistema.
  - ◆ El descentralizado aún depende de ciertos nodos clave; el distribuido puede eliminar completamente los puntos únicos de falla.





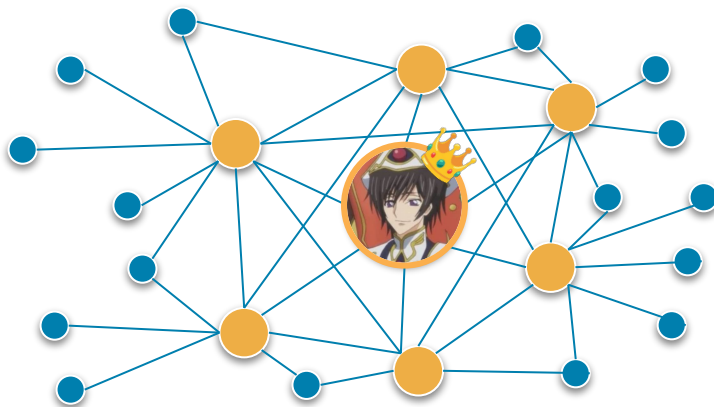
# ¿Cuál es la diferencia con un sistema descentralizado?

- ◆ No existe un consenso universal sobre la diferencia entre estos conceptos.
- ◆ Un enfoque clásico es según la existencia de una **autoridad** permanente.
- ◆ Otro enfoque es la **robustez** del sistema.
- ◆ Sin embargo, un sistema distribuido puede tener diferentes tipos de autoridad o repartir algunas responsabilidades, así como la robustez dependerá de cómo es la implementación.



# ¿Cuál es la diferencia con un sistema descentralizado?

- ◆ No existe un consenso universal sobre la diferencia entre estos conceptos.
- ◆ Un enfoque clásico es según la existencia de una **autoridad** permanente.
- ◆ Otro enfoque es la **robustez** del sistema.
- ◆ Sin embargo, un sistema distribuido puede tener diferentes tipos de autoridad o repartir algunas responsabilidades, así como la robustez dependerá de cómo es la implementación.



- ◆ En este curso nos guiaremos por el libro guía, es decir... la necesidad.

# ¿Cuál es la diferencia con un sistema descentralizado?

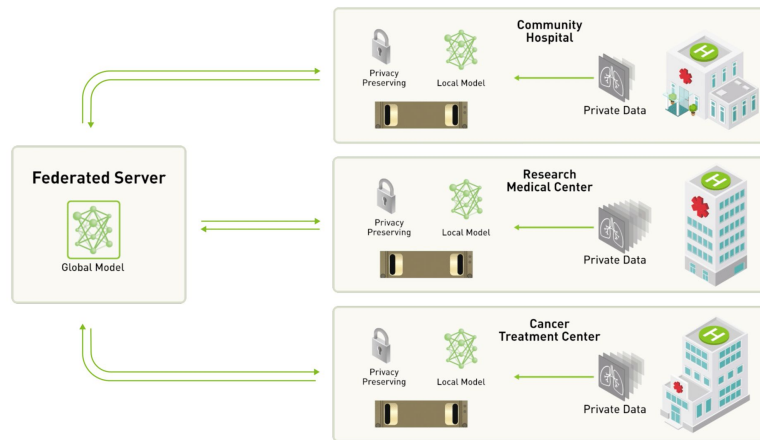
## Descentralizado

- ◆ Surge una **necesidad que obliga** a tener múltiples servidores repartidos.

# ¿Cuál es la diferencia con un sistema descentralizado?

## Descentralizado

- ◆ Surge una **necesidad que obliga** a tener múltiples servidores repartidos.
- ◆ **Necesidad administrativa:** *federal learning* debido a privacidad de datos.
- ◆ **Necesidad de confianza:** *Blockchain*.
- ◆ **Necesidad geográfica:** sistemas de monitoreo de plantas energéticas.



# ¿Cuál es la diferencia con un sistema descentralizado?

## Distribuido

- ◆ Surge una **decisión de diseño** para cumplir requisitos de rendimiento y fiabilidad impuestos por el mismo creador.
- ◆ Por ejemplo, Google con múltiples *Data Centers* repartidos por el mundo y nodos cercanos para reducir la latencia de comunicación.

Google



*Data Centers*

# ¿Cuál es la diferencia con un sistema descentralizado?

## Descentralizado o Distribuido

- ◆ El descentralizado separa los servidores por **necesidad**; el distribuido, por **decisión** de diseño.
- ◆ La autoridad o la robustez depende de **cómo se orquesta** cada sistema, no de su clasificación.
- ◆ Un sistema distribuido puede incluir nodos servidores y clientes, y el nodo cliente puede depender de uno o varios servidores.
- ◆ Con esta distinción, todo lo que aprenderemos en el curso aplica tanto a sistemas distribuidos como descentralizados.

# Objetivos de un Sistema distribuido

Se busca que un Sistema Distribuido cumple los siguientes objetivos:

- ◆ Compartir Recursos (*Resource Sharing*)
- ◆ Transparencia de Distribución (*Distribution Transparency*)
- ◆ Apertura (*Openness*)
- ◆ Confiabilidad (*Dependability*)
- ◆ Seguridad (*Security*)
- ◆ Escalabilidad (*Scalability*)

# Objetivos de un Sistema distribuido

Se busca que un Sistema Distribuido cumple los siguientes objetivos:

- ◆ **Compartir Recursos (*Resource Sharing*)**
- ◆ Transparencia de Distribución (*Distribution Transparency*)
- ◆ Apertura (*Openness*)
- ◆ Confiabilidad (*Dependability*)
- ◆ Seguridad (*Security*)
- ◆ Escalabilidad (*Scalability*)

Acceder a recursos remotos sin preocuparse por su ubicación, Sistema Operativo utilizado, etc.

## Ejemplo

- ◆ *Torrent*
- ◆ *One drive*



# Objetivos de un Sistema distribuido

Se busca que un Sistema Distribuido cumple los siguientes objetivos:

- ◆ Compartir Recursos (*Resource Sharing*)
- ◆ **Transparencia de Distribución** (*Distribution Transparency*)
- ◆ Apertura (*Openness*)
- ◆ Confiabilidad (*Dependability*)
- ◆ Seguridad (*Security*)
- ◆ Escalabilidad (*Scalability*)

Ocultar el hecho de que sus procesos y recursos están distribuidos físicamente.

## Ejemplo

- ◆ URL de un sitio
- ◆ *Google Drive*

# Objetivos de un Sistema distribuido

## Transparencia de Distribución (*Distribution Transparency*) - Existen varios tipos

### ◆ Transparencia de **Acceso**

- ◆ Permite que los recursos locales y remotos se acceden de la misma manera.

### ◆ Transparencia de **Ubicación**

- ◆ Facilita el acceso a recursos sin necesidad de conocer su ubicación física o en la red.

### ◆ Transparencia de **Migración**

- ◆ Permite mover recursos o clientes dentro del sistema sin interrumpir la operación.

# Objetivos de un Sistema distribuido

## Transparencia de Distribución (*Distribution Transparency*) - Existen varios tipos

- ◆ Transparencia de **Replicación**
  - ◆ Oculta la existencia de múltiples copias de un recurso.
- ◆ Transparencia de **Concurrencia**
  - ◆ Garantiza que múltiples nodos compartan recursos al mismo tiempo sin conflictos.
- ◆ Transparencia de **Fallos**
  - ◆ Permite completar tareas, incluso si ocurren errores en los componentes, al reconfigurar o redirigir las operaciones automáticamente.

# Objetivos de un Sistema distribuido

Se busca que un Sistema Distribuido cumple los siguientes objetivos:

- ◆ Compartir Recursos (*Resource Sharing*)
- ◆ Transparencia de Distribución (*Distribution Transparency*)
- ◆ **Apertura (*Openness*)**
- ◆ Confiabilidad (*Dependability*)
- ◆ Seguridad (*Security*)
- ◆ Escalabilidad (*Scalability*)

Ofrecer componentes que pueden ser fácilmente usados o integrados en otros sistemas.

## Ejemplo

- ◆ Protocolos de comunicación
- ◆ Aplicación web con microservicios

# Objetivos de un Sistema distribuido

Se busca que un Sistema Distribuido cumple los siguientes objetivos:

- ◆ Compartir Recursos (*Resource Sharing*)
- ◆ Transparencia de Distribución (*Distribution Transparency*)
- ◆ Apertura (*Openness*)
- ◆ **Confiabilidad (*Dependability*)**
- ◆ Seguridad (*Security*)
- ◆ Escalabilidad (*Scalability*)

Las operaciones realizadas pueden funcionar según lo esperado incluso ante fallos parciales.

## Ejemplo

- ◆ Buzón de tarea de Canvas
- ◆ Google

# Objetivos de un Sistema distribuido

Se busca que un Sistema Distribuido cumple los siguientes objetivos:

- ◆ Compartir Recursos (*Resource Sharing*)
- ◆ Transparencia de Distribución (*Distribution Transparency*)
- ◆ Apertura (*Openness*)
- ◆ Confiabilidad (*Dependability*)
- ◆ **Seguridad (*Security*)**
- ◆ Escalabilidad (*Scalability*)

Los activos del sistema (*hardware, software, datos*) están protegidos y no se alteren sin autorización.

## Ejemplo

- ◆ Permisos explícitos para editar archivo de *Google Drive*
- ◆ Roles de usuarios en canvas (estudiantes vs profesor)

# Objetivos de un Sistema distribuido

Se busca que un Sistema Distribuido cumple los siguientes objetivos:

- ◆ Compartir Recursos (*Resource Sharing*)
- ◆ Transparencia de Distribución (*Distribution Transparency*)
- ◆ Apertura (*Openness*)
- ◆ Confiabilidad (*Dependability*)
- ◆ Seguridad (*Security*)
- ◆ **Escalabilidad (*Scalability*)**

Seguir siendo efectivo (en rendimiento, costo, facilidad de gestión) a medida que crece.

## Ejemplo

- ◆ Reuniones por Zoom (escalabilidad geográfica)
- ◆ *Retails* durante *cyber days* (escalabilidad de tamaño)

# Objetivos de un Sistema distribuido

## Escalabilidad (*Scalability*) - Existen varios tipos

### ◆ Escalabilidad de **Tamaño**

- ◆ Capacidad de añadir más usuarios y recursos al sistema sin una disminución perceptible del rendimiento.

### ◆ Escalabilidad **Geográfica**

- ◆ Los usuarios y los recursos pueden estar geográficamente muy separados, pero los retrasos en la comunicación no se perciben significativamente.

### ◆ Escalabilidad **Funcional**

- ◆ Se pueden agregar nuevas funcionalidades al sistema sin un retraso de rendimiento.



# Tipos de sistemas

# Tipos de sistemas - Introducción

**Clasificaremos según su principal uso:**

- ◆ Computación Distribuida de Alto Rendimiento (*High-Performance Distributed Computing*)
- ◆ Sistemas de Información Distribuidos (*Distributed Information Systems*)
- ◆ Sistemas Pervasivos (*Pervasive Systems*)

Los límites entre estas categorías no son estrictos y las combinaciones son comunes. Lo importante es distinguir el uso principal de la red.

# Computación Distribuida de Alto Rendimiento (HPC)

Se utilizan para **ejecutar tareas de computación que requieren un alto procesamiento** de información en un tiempo acotado.

**Se distinguen 2 grupos en este tipo de sistema distribuido:**

- ◆ Computación en Clúster (*Cluster Computing*)
- ◆ Computación en Grid (*Grid Computing*)

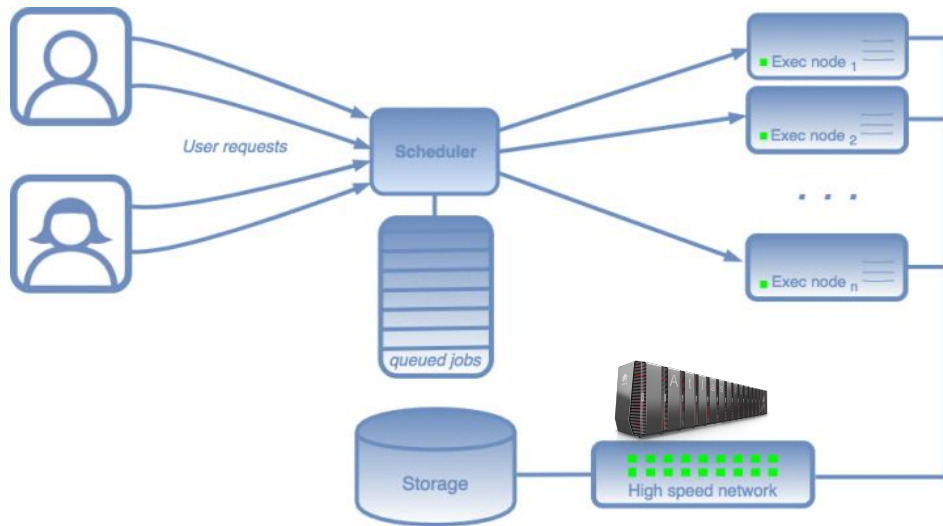
# HPC - Computación en Clúster

**Definición:** Una colección de nodos de cómputo similares interconectados por una red de alta velocidad.

**Uso principal:** Programación paralela, donde un programa intensivo en cómputo se ejecuta en paralelo en múltiples máquinas.

## ¿Cómo se organizan?

- ◆ Existe un responsable de recolectar trabajos de los usuarios y distribuir las tareas asociadas entre los nodos de cómputo.
- ◆ Slurm es un programa comúnmente utilizado para este tipo de situación.



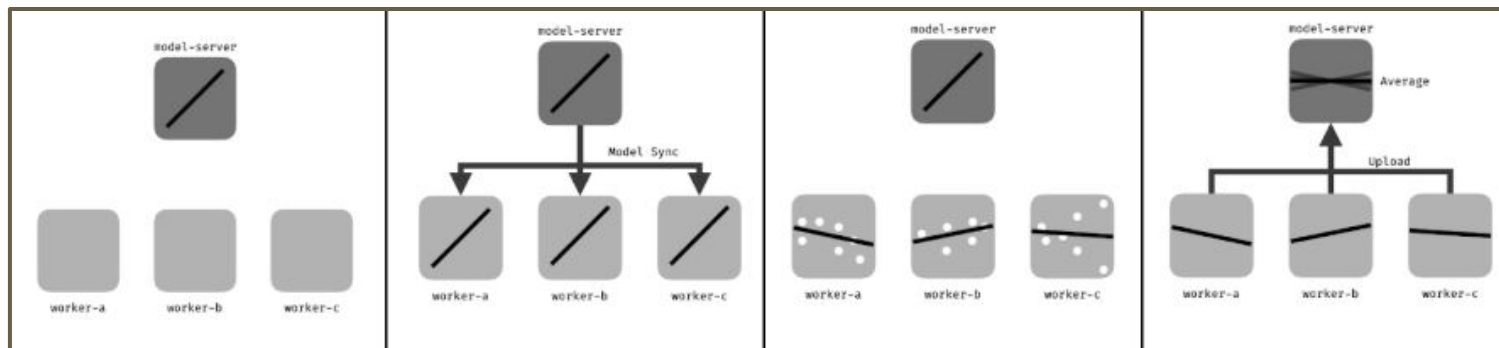
# HPC - Computación en Grid

**Definición:** Una colección de nodos de cómputo (*clusters*) **separados** que se organizan para actuar como una única entidad, es decir, sistema distribuido de *clusters*.

**Uso principal:** Aumento de rendimiento en el entrenamiento y distribución de carga de un *cluster*. Además, se puede utilizar para garantizar privacidad de información.

## ¿Cómo se organizan?

- ◆ Se añade una capa adicional que se comunica con cada responsable del *cluster* y unifica la información.



# Sistemas de Información Distribuidos

Centrado en la **operabilidad entre diferentes sistemas** y la integración de aplicaciones a nivel empresarial dentro de organizaciones con múltiples aplicaciones en red.

Se identifican 2 usos principales no excluyentes:

- ◆ **Integración de Aplicaciones Empresariales:** se facilita un intermediario (*middleware/intermediario*) para integrar las distintas aplicaciones.
- ◆ **Procesamiento de Transacciones Distribuidas:** se garantiza que todas las solicitudes realizadas se ejecuten o ninguna lo haga.

# Sistemas de Información Distribuidos

**Ejemplo:** sistema de mensajería

- ◆ **Integración:** comunicación asíncrona y persistente mediante el uso de una cola. El usuario puede enviar desde cualquier gestor de mensajería y este será aceptado, y no necesita estar conectado para recibir el mensaje.
- ◆ **Transaccional:** si envío un correo (solicitud 1), debe llegar al usuario (solicitud 2). Si no ocurre ambas solicitudes, se indica que el envío rebotó.



# Sistemas Pervasivos

Sistemas compuestos por una variedad de dispositivos que se integran de forma natural en nuestro entorno.

En la literatura, se distinguen **3 grupos no excluyentes** en este tipo de sistema distribuido

- ◆ Sistemas de Computación Ubicua (*Ubiquitous Computing Systems*)
- ◆ Sistemas de Computación Móvil (*Mobile Computing Systems*)
- ◆ Redes de Sensores (*Sensor Networks*)

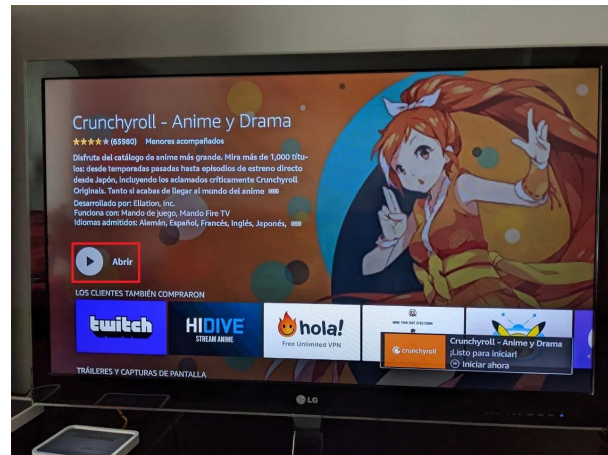


# Sistemas Pervasivos - Sistemas de Computación Ubicua

El sistema está constantemente **presente en el entorno** y con el que el usuario interactúa de manera **implícita**, a menudo sin siquiera ser consciente de que lo está haciendo por debajo.

Algunos ejemplos de estos sistemas son:

- ◆ *Smart TV.*
- ◆ Asistentes de voz como Alexa o Google Home



# Sistemas Pervasivos - Sistemas de Computación Móvil

Son sistemas que involucran dispositivos móviles, típicamente utilizando comunicación inalámbrica para acceder a servicios remotos. Se caracterizan por el **movimiento del dispositivo y su necesidad de mantener la conectividad**.

Ejemplos comunes:

- ◆ Celular
- ◆ Tarjeta inteligentes



# Sistemas Pervasivos - Redes de Sensores

Son colecciones de nodos del tipo **sensor** que colaboran para **procesar datos de manera eficiente**.

Buscan procesar los datos lo más cercano a la fuente, reduciendo el tráfico de red y el consumo de energía.

Por ejemplo:

- ◆ Alarma de humo.
- ◆ Sensores de temperatura.
- ◆ Sistema de riego.



# Poniendo a prueba lo que hemos aprendido

Una empresa desarrolla un servicio de *streaming* de conciertos que promete funcionar 24/7 a los clientes, incluso si algún servidor se cae, el usuario será redirigido a otro para continuar con el servicio. Para esto, se tiene total confianza en que la comunicación entre los servidores será de la más rápida posible, tal que si cae un servidor, alcanzará a enviar el concierto a otro servidor para que el usuario lo siga viendo sin problemas.

¿Qué par objetivo-problemas de los sistemas distribuidos **ilustra correctamente** este escenario?

- a. Compartir recursos - Problemas de latencia y ancho de banda.
- b. Confiabilidad - *Pitfalls* de diseño.
- c. Seguridad - *Pitfalls* de diseño.
- d. Confiabilidad - Problemas de latencia y ancho de banda.
- e. Escalabilidad - Complejidad inherente.

# Poniendo a prueba lo que hemos aprendido

Una empresa desarrolla un servicio de *streaming* de conciertos que promete funcionar 24/7 a los clientes, incluso si algún servidor se cae, el usuario será redirigido a otro para continuar con el servicio. Para esto, se tiene total confianza en que la comunicación entre los servidores será de la más rápida posible, tal que si cae un servidor, alcanzará a enviar el concierto a otro servidor para que el usuario lo siga viendo sin problemas.

¿Qué par objetivo-problemas de los sistemas distribuidos **ilustra correctamente** este escenario?

a. Compartir recursos - Problemas de latencia y ancho de banda.

**b. Confiabilidad - Pitfalls de diseño.**

c. Seguridad - Pitfalls de diseño.

d. Confiabilidad - Problemas de latencia y ancho de banda.

e. Escalabilidad - Complejidad inherente.

# Próximos eventos

## Próxima clase

- ◆ Comunicación entre nodos I
  - ◆ Características de una comunicación: tipos y paradigmas.
  - ◆ Comunicación directa mediante *socket*. Vital recordar esto para la Tarea 1.

## Evaluación

- ◆ Control 1 se publicará mañana y evalúa hasta esta clase.
- ◆ Recomendando hacerlo con tiempo para que sea un buen mecanismo de estudio paulatino.

---

# IIC2523

# Sistemas Distribuidos

— Hernán F. Valdivieso López —  
(2025 - 2 / Clase 02)

---

# Créditos (animes utilizados)

## Naruto



## Dragon Ball



## Evangelion



## Code Geass

