

Informe Técnico Comparativo de Arquitecturas de Sistemas

Este informe contrasta las características clave de las arquitecturas Centralizada, Cliente-Servidor y Distribuida (o P2P/Multinivel), centrándose en su modelo operacional, capacidad de crecimiento y robustez.



Introducción a las Arquitecturas de Sistemas

¿Qué es una arquitectura de sistemas?

Una arquitectura de sistemas es el plan fundamental que define la estructura de un sistema informático, la forma en que sus componentes se interrelacionan y la manera en que el sistema funcionará. Esencialmente, es el blueprint que guía la construcción de un sistema, garantizando que cumpla con los requisitos funcionales y, más importante aún, con los requisitos no funcionales (como escalabilidad, seguridad, y rendimiento). En el ámbito de la informática, el término arquitectura de sistemas engloba tanto al hardware como al software y las redes que los conectan.



La elección correcta de arquitectura determina el **rendimiento**, **escalabilidad** y **seguridad** de cualquier sistema. Exploraremos las arquitecturas centralizada, cliente-servidor, distribuida y descentralizada.



Arquitectura Centralizada: El Control en un Solo Punto

Características

Todo el procesamiento y almacenamiento se concentra en un único servidor central que gestiona todas las operaciones del sistema.

Ventajas

- Administración sencilla y centralizada
- Seguridad controlada desde un punto
- Costos iniciales bajos

Desventajas

- Punto único de falla crítico
- Problemas de escalabilidad
- Alta latencia para usuarios remotos

Ejemplos típicos incluyen redes internas empresariales tradicionales y sistemas bancarios centralizados donde el control absoluto es prioritario.

Arquitectura Cliente-Servidor: Comunicación Controlada



Cliente

Solicita servicios y recursos
específicos



Servidor

Procesa peticiones y envía
respuestas



Red

Facilita la comunicación bidireccional

Ventajas Clave

- Separación clara de responsabilidades
- Mantenimiento centralizado simplificado
- Control de acceso robusto

Limitaciones

- Saturación del servidor con alta demanda
- Latencia dependiente de la red
- Escalabilidad limitada verticalmente

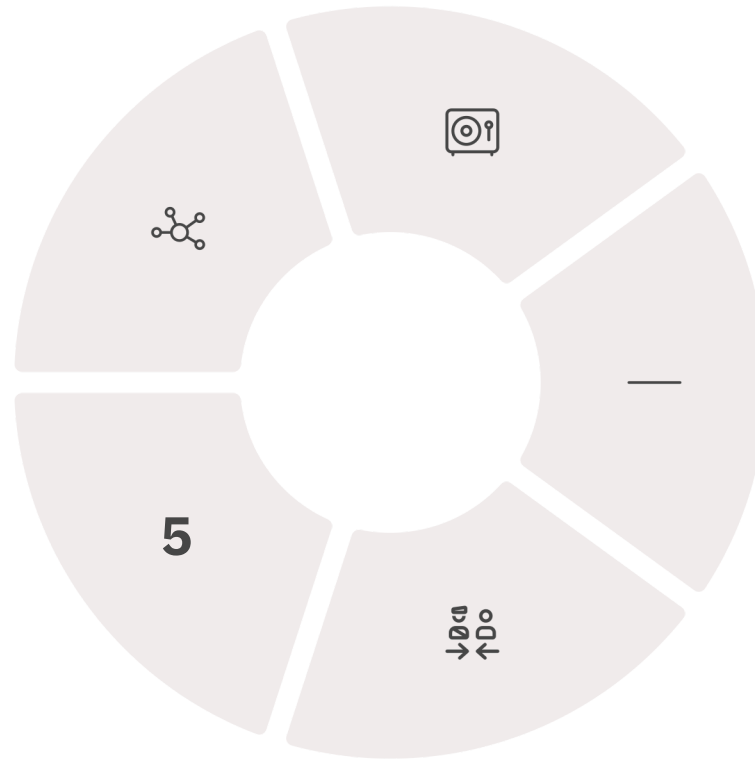
Ampliamente utilizada en aplicaciones web modernas, sistemas de bases de datos empresariales y servicios de correo electrónico.

Arquitectura Distribuida: Recursos en Múltiples Nodos

Alta Disponibilidad

Múltiples nodos garantizan continuidad operativa

Ejemplos actuales incluyen **blockchain, Google Spanner, arquitecturas de microservicios** y sistemas de computación en la nube.



Tolerancia a Fallos

Sistemas resilientes ante fallas individuales

Escalabilidad

Crecimiento horizontal agregando nodos

Complejidad

Desafíos en coordinación y sincronización

Arquitectura Descentralizada: Control Compartido y Resiliencia

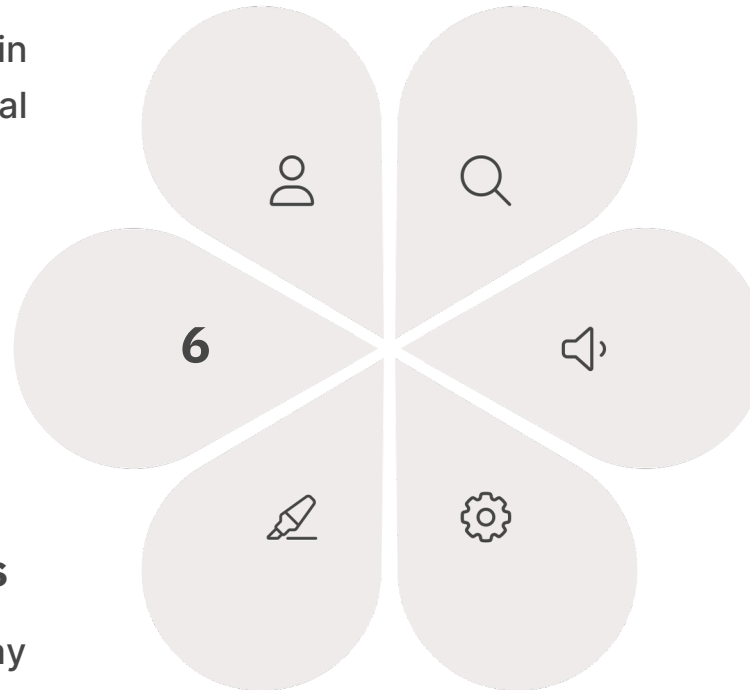
Autonomía

Nodos operan independientemente sin control central

Casos emblemáticos: **redes blockchain como Bitcoin, sistemas peer-to-peer como BitTorrent** y redes de distribución de contenido descentralizadas.

Costos Elevados

Mayor inversión en infraestructura y mantenimiento



Robustez

Resistencia extrema ante fallos y ataques

Baja Latencia

Procesamiento local reduce tiempos de respuesta

Gestión Compleja

Coordinación sin autoridad central

Comparativa Clave: Ventajas y Desventajas

Arquitectura	Ventajas	Desventajas
Centralizada	Fácil gestión, seguridad centralizada	Punto único de falla, baja escalabilidad
Cliente-Servidor	Control centralizado, mantenimiento sencillo Alta disponibilidad, escalabilidad	Saturación del servidor, latencia
Distribuida	horizontal Resiliencia máxima, baja latencia	Complejidad, coordinación difícil
Descentralizada		Costos altos, gestión compleja

ilidad y Seguridad

01

Centralizada

Limitada por la capacidad física del servidor central

02

Cliente-Servidor

Escalabilidad vertical principalmente, con cuellos de botella

03

Distribuida/Descentralizada

Escalabilidad horizontal limitada agregando nodos



Seguridad reforzada en blockchain mediante consenso distribuido entre nodos, eliminando puntos únicos de vulnerabilidad.

Casos de Uso y Tendencias Actuales



Centralizada

Sistemas legacy empresariales y aplicaciones con baja demanda de escalabilidad



Cliente-Servidor

Aplicaciones web tradicionales y sistemas de bases de datos relacionales



Distribuida

Microservicios, cloud computing, Internet de las Cosas (IoT)



Descentralizada

Blockchain, redes P2P, edge computing y DeFi

i Tendencia creciente hacia arquitecturas **distribuidas y descentralizadas** impulsada por necesidades de resiliencia, escalabilidad global y democratización tecnológica.

Conclusión: Elegir la Arquitectura Adecuada

No existe una solución única

Factores de Decisión

- Objetivos del negocio
- Recursos disponibles
- Contexto operacional
- Tolerancia al riesgo

Recomendaciones

Centralizada: simplicidad y control absoluto

Distribuida/Descentralizada: escalabilidad y alta disponibilidad

La evolución tecnológica impulsa la adopción de **sistemas distribuidos y descentralizados**. Evalúe cuidadosamente sus necesidades y riesgos para diseñar sistemas robustos, eficientes y preparados para el futuro.

