

Informe Técnico Comparativo: Arquitecturas de Sistemas

Análisis y evaluación de modelos Centralizados, Cliente-Servidor y Distribuidos.

Presentado por: [Nombre / Institución]

SECCIÓN I

La Arquitectura como Pilar del Sistema

Definición de Arquitectura de Sistemas

Marco conceptual que define la estructura, organización e interconexión de componentes, módulos y relaciones dentro de un sistema informático.

Importancia de la Elección

La decisión arquitectónica impacta directamente en la escalabilidad, el rendimiento, la mantenibilidad y la robustez del producto final a largo plazo.

Este informe técnico se centra en el análisis comparativo de tres paradigmas fundamentales: Centralizado, Cliente-Servidor y Distribuido.

Arquitectura Centralizada: El Control Único

Modelo y Funcionamiento

Todos los procesos de negocio, el almacenamiento de datos y la lógica de aplicación residen en un único host o servidor principal (el *mainframe*).

Los usuarios acceden a través de terminales "tontas" (clientes ligeros) que solo se encargan de la entrada y salida de datos, sin capacidad de procesamiento local.



Ventajas Clave

- Control de seguridad unificado y simplificado.
- Bajo costo inicial de infraestructura.
- Fácil despliegue y mantenimiento de software.

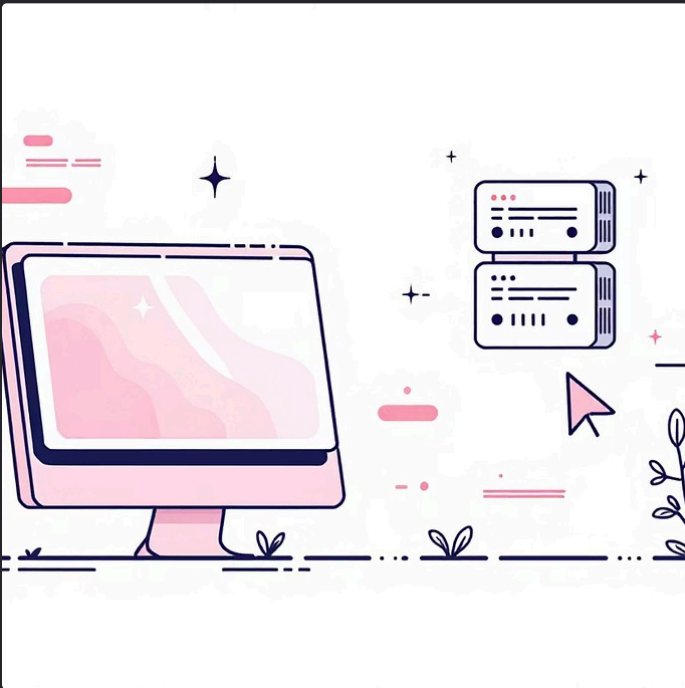


Ejemplos de Uso: Sistemas bancarios antiguos, mainframes corporativos y algunos sistemas de gestión de inventario monolíticos.

- ❑ **Principal Limitación:** El punto único de falla (Single Point of Failure - SPOF) y la baja tolerancia al aumento de la carga de trabajo.

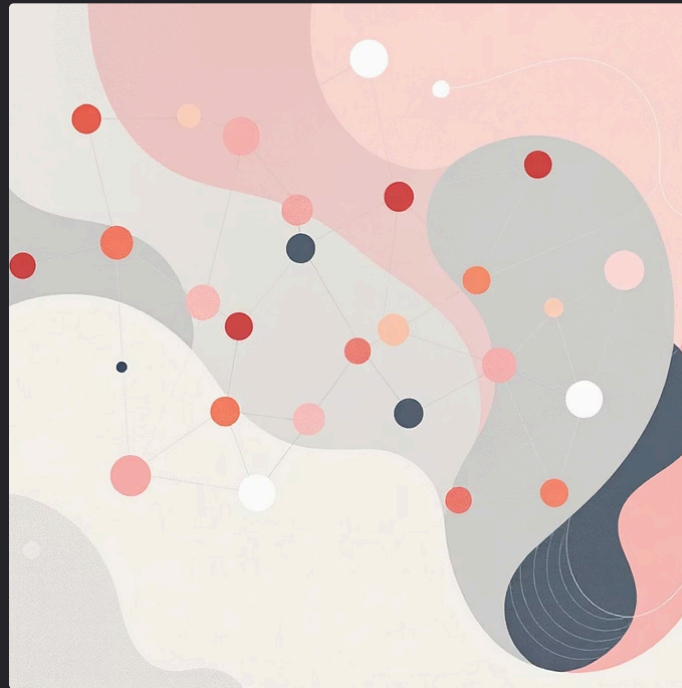
Arquitectura Cliente-Servidor: La División de Roles

Este modelo introduce la especialización, dividiendo el trabajo entre proveedores de recursos (servidores) y solicitantes de servicios (clientes).



Modelo Básico

El cliente inicia la solicitud de servicios (datos, procesamiento, recursos) y el servidor responde a esas peticiones.



Ventajas Operacionales

Permite la distribución de tareas y el procesamiento descentralizado, mejorando el rendimiento y facilitando el mantenimiento modular.



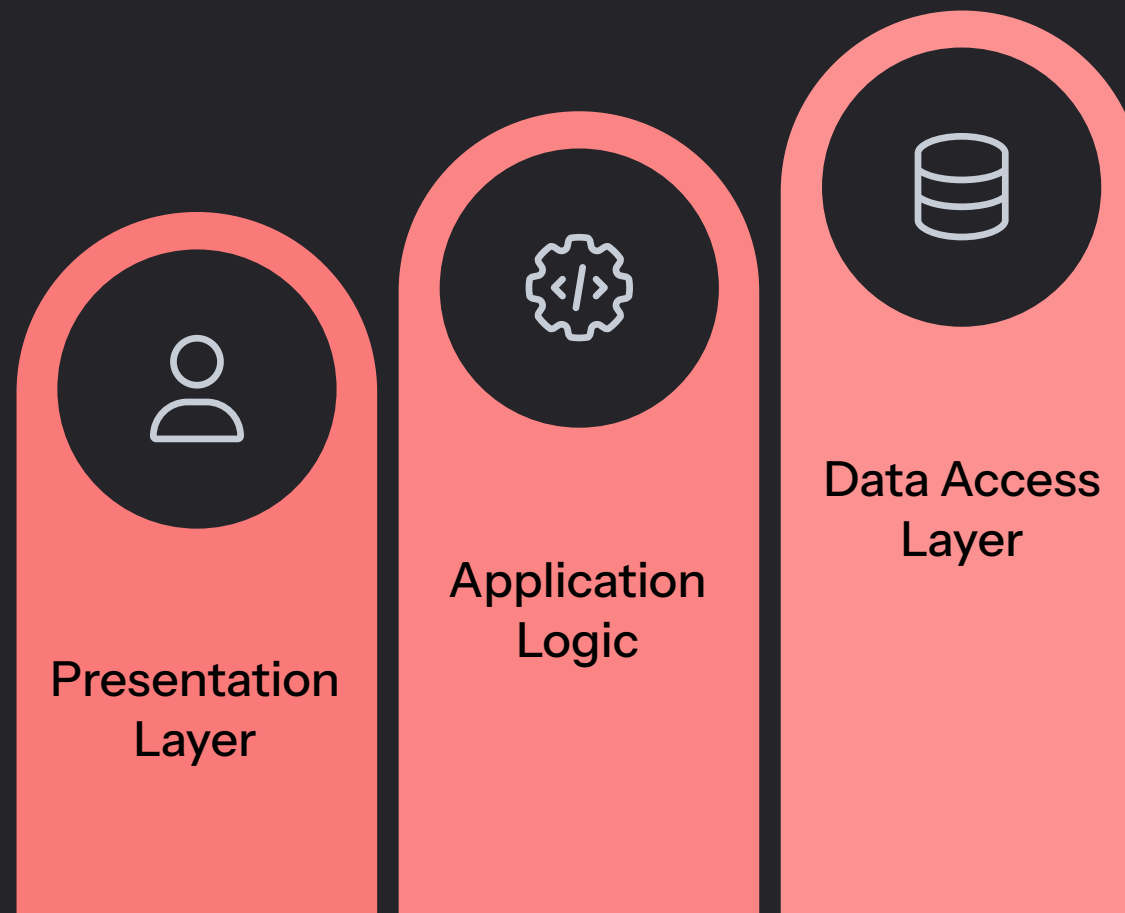
Desafíos

La dependencia total del servidor puede generar cuellos de botella y limitaciones de escalabilidad si el servidor principal se sobrecarga.

El modelo Cliente-Servidor es la base de la mayoría de las aplicaciones web y bases de datos relacionales modernas.

El Modelo de N Capas (Multinivel)

Una refinación del modelo Cliente-Servidor, que separa la aplicación en capas lógicas para una mayor flexibilidad y escalabilidad.



Capa de Presentación

Gestión de la interfaz de usuario (el cliente).

Capa de Lógica (Business)

Contiene las reglas de negocio y el procesamiento centralizado.

Capa de Datos

Gestión del almacenamiento y recuperación de la información.

PARADIGMA 3

Arquitectura Distribuida: Tolerancia y Colaboración

Un sistema en el que los componentes se comunican y coordinan sus acciones únicamente mediante el paso de mensajes, sin una memoria compartida central.



Definición

Conjunto de computadoras autónomas que aparecen ante los usuarios como un único sistema coherente.



Escalabilidad Superior

La capacidad de añadir nuevos nodos horizontalmente sin afectar significativamente el rendimiento del sistema.



Alta Resiliencia

Tolerancia a fallos: el sistema sigue operando incluso si varios componentes fallan (sin SPOF).

Ejemplos de Uso: Microservicios, sistemas de *cloud computing* (AWS, Azure), redes P2P (BitTorrent), y sistemas de registro distribuido (Blockchain).

El Desafío Distribuido: Complejidad

Aunque las arquitecturas distribuidas ofrecen beneficios inigualables en escalabilidad y disponibilidad, imponen desafíos técnicos significativos.

Gestión de la Sincronización

Asegurar la consistencia de los datos y el estado en múltiples nodos que operan de forma independiente (ej. Transacciones distribuidas).

Mayor Complejidad de Desarrollo

El desarrollo, la depuración y el monitoreo requieren herramientas y metodologías especializadas (ej. trazabilidad de microservicios).

Seguridad de la Comunicación

Implementar protocolos seguros a través de la red para proteger la información en tránsito entre todos los componentes.



Tabla Comparativa de Arquitecturas

Una visión general de los puntos fuertes y débiles de cada modelo en métricas clave.

Métrica	Centralizada	Cliente-Servidor	Distribuida
Escalabilidad	Baja (Escala vertical limitada)	Media (Escala vertical y horizontal limitada)	Alta (Escala horizontal ilimitada)
Rendimiento	Variable (Saturación central)	Bueno (Balanceo de carga simple)	Excelente (Paralelización de tareas)
Tolerancia a Fallos	Nula (SPOF)	Baja/Media (El servidor es SPOF)	Máxima (Resiliente por diseño)
Mantenimiento	Simple	Moderado (Modular)	Alto (Orquestación compleja)
Costo Inicial	Bajo	Medio	Alto (Mayor infraestructura y conocimiento)

Recomendaciones de Contexto

La elección debe alinearse con los requisitos no funcionales del sistema (rendimiento, disponibilidad, presupuesto).



Centralizada

Sistemas pequeños, de bajo presupuesto y sin necesidad de alta concurrencia. Donde la seguridad física de un único entorno es crítica.



Cliente-Servidor

Aplicaciones empresariales, bases de datos internas o sistemas web con tráfico predecible. Ideal para la separación de preocupaciones (Front/Back-end).



Distribuida

Sistemas de misión crítica, aplicaciones con picos de tráfico impredecibles o requisitos de disponibilidad del 99.99%. Tendencia actual del Cloud Computing.



Conclusiones y Proyecciones Futuras

La industria se dirige inexorablemente hacia la **descentralización** y la **modularidad** para satisfacer las demandas de disponibilidad global.



El Dominio Distribuido

El auge del *Cloud Computing* y los microservicios valida la arquitectura distribuida como el estándar para los sistemas modernos de alto rendimiento.



Gestión de la Complejidad

El foco se traslada de la gestión del hardware a la orquestación, la observabilidad y la coherencia de múltiples servicios.



Decisión Arquitectónica

El arquitecto de sistemas moderno debe priorizar la flexibilidad, diseñando soluciones que permitan la transición fluida hacia modelos distribuidos según el crecimiento del negocio.

Agradecimiento: Gracias por su atención. Preguntas y debate abiertos.