

Arquitectura e Integración de Aplicaciones Empresariales

José Vicente López López

26/10/2024

1 Comparación de Arquitecturas en Aplicaciones Empresariales

La siguiente tabla proporciona una comparación de cuatro arquitecturas (en capas, orientada a objetos, basada en eventos y basada en datos) en el contexto de la arquitectura e integración de aplicaciones empresariales. Cada arquitectura se evalúa en función de sus características, ventajas y desventajas.

Arquitectura	Características	Ventajas	Desventajas
Arquitectura en Capas	<ul style="list-style-type: none">- Divide el sistema en capas lógicas (por ejemplo, presentación, lógica de negocio, acceso a datos).- Cada capa tiene una responsabilidad específica y comunica con capas adyacentes.- Común en sistemas monolíticos y web.	<ul style="list-style-type: none">- Modularidad: cada capa puede desarrollarse, probarse y mantenerse de forma independiente, lo que facilita el mantenimiento.- Escalabilidad horizontal limitada: se puede dividir la carga entre las capas.- Reutilización: los componentes en una capa pueden ser utilizados en varias partes del sistema.	<ul style="list-style-type: none">- Interdependencia: cualquier cambio en una capa puede requerir ajustes en otras capas.- Baja flexibilidad para responder a cambios en las necesidades del negocio, ya que las capas superiores dependen de las inferiores.- Puede presentar cuellos de botella en capas específicas, afectando el rendimiento general.
Arquitectura Orientada a Objetos	<ul style="list-style-type: none">- Basada en el paradigma de Programación Orientada a Objetos (POO), donde cada componente es un objeto que representa datos y comportamientos.- Facilita la creación de modelos de negocio mediante la identificación de objetos reales (clientes, productos, pedidos, etc.).- Los objetos colaboran entre sí a través de mensajes e interfaces bien definidas.	<ul style="list-style-type: none">- Alta cohesión y bajo acoplamiento, permitiendo que los componentes sean flexibles y adaptables.- Reutilización de componentes y código gracias a la herencia y composición.- Facilita la comprensión y modelado de dominios complejos mediante la representación en objetos del mundo real.	<ul style="list-style-type: none">- Complejidad en grandes sistemas debido a la gran cantidad de objetos y relaciones.- Dificultad en la integración con arquitecturas basadas en datos, ya que se necesita un modelo relacional diferente.- Sobrecarga en el rendimiento debido a la comunicación entre objetos, especialmente en sistemas distribuidos.

Arquitectura	Características	Ventajas	Desventajas
Arquitectura Basada en Eventos	<ul style="list-style-type: none"> - Los sistemas se organizan en torno a eventos que indican cambios de estado en los componentes. - Los eventos desencadenan acciones, y los componentes reaccionan de forma independiente sin requerir comunicación directa. - Promueve un sistema asincrónico, donde los eventos pueden estar desacoplados de los servicios que los procesan. 	<ul style="list-style-type: none"> - Alta escalabilidad y flexibilidad, permitiendo la integración de nuevos servicios sin impactar otros componentes. - Resiliencia: los sistemas son más tolerantes a fallos, ya que las interacciones se realizan mediante eventos asincrónicos. - Buen rendimiento para sistemas con alta concurrencia, ya que los eventos pueden procesarse de manera independiente. 	<ul style="list-style-type: none"> - Complejidad en la gestión de estados y en el manejo de eventos que se generan de manera masiva. - Dificultad para depurar y rastrear errores debido a la naturaleza asincrónica de los eventos. - Dependencia en la infraestructura de mensajes (brokers de mensajería), que aumenta la complejidad del sistema.
Arquitectura Basada en Datos	<ul style="list-style-type: none"> - Se enfoca en la centralización de datos, donde la fuente de datos es el núcleo y los servicios o componentes acceden directamente a ellos. - Los sistemas de información y los servicios de negocio se estructuran en torno a la manipulación de datos y su persistencia. - Común en sistemas de gestión de grandes volúmenes de datos, como almacenes de datos o plataformas analíticas. 	<ul style="list-style-type: none"> - Permite la unificación y centralización de datos, lo que facilita la consistencia y gobernanza de los mismos. - Escalabilidad en el manejo de grandes volúmenes de datos, útil en sistemas de big data y análisis de información. - Simplificación en la gestión de datos, permitiendo un acceso directo a la información en lugar de múltiples intermediarios. 	<ul style="list-style-type: none"> - Menor flexibilidad para cambios en el modelo de datos o en la estructura del sistema. - Problemas de rendimiento si los accesos concurrentes a los datos no son gestionados correctamente. - Riesgos de disponibilidad y seguridad, ya que los datos centralizados pueden ser un punto de falla único.

References

- [1] G. Alonso, F. Casati, H. Kuno, y V. Machiraju, *Web Services: Concepts, Architectures and Applications (Data-Centric Systems and Applications)*. Springer, 2010.
- [2] L. Shklar y R. Rosen, *Web Application Architecture: Principles, Protocols and Practices*. John Wiley & Sons, 2009.