

EVENT SOURCING

-Integrantes:

Luis Lomba

Marcos Méndez

-Introducción:

Event Sourcing es un patrón de arquitectura que define un enfoque para las operaciones de manipulación de datos. La idea es crear una secuencia de eventos, cada uno de los cuales se registra en un store.

-Uso:

En lugar de almacenar simplemente el estado actual de los datos de un dominio, se usa un almacén de solo anexar para registrar toda la serie de acciones realizadas en los datos. El almacén actúa como el sistema de registro y puede usarse para materializar los objetos de dominio

Event Sourcing asegura que todos los cambios en el estado de la aplicación se almacenen como una secuencia de eventos. No solo podemos consultar estos eventos, también podemos usar el registro de eventos para reconstruir estados pasados y como base para ajustar automáticamente el estado para hacer frente a los cambios retroactivos.

-Problema CRUD:

Trabajando con datos el enfoque clásico y habitual es trabajar con CRUD, aunque con ello nos lleve a ciertos problemas que Event Sourcing nos puede resolver.

-Solución Event Sourcing frente a CRUD:

El código de la aplicación envía una serie de eventos que imperativamente describen cada acción que se ha producido en los datos del almacén de eventos, donde se conservan. Cada evento representa un conjunto de cambios en los datos.

Además, las aplicaciones pueden leer el historial de eventos en cualquier momento y usarlo para materializar el estado actual de una entidad al reproducir y consumir todos los eventos relacionados con esa entidad.

-Ventajas de Event Sourcing:

- Reconstrucción completa
- Consulta temporal
- Repetición de eventos
- Auditoría

-Problemas:

- Actualizaciones externas
- Consultas externas
- Cambios en el código

-Cúando usar el patrón:

- Auditoría
- Soporte
- Debug
- Escalabilidad
- Patrones Parallel Models o Retroactive Events

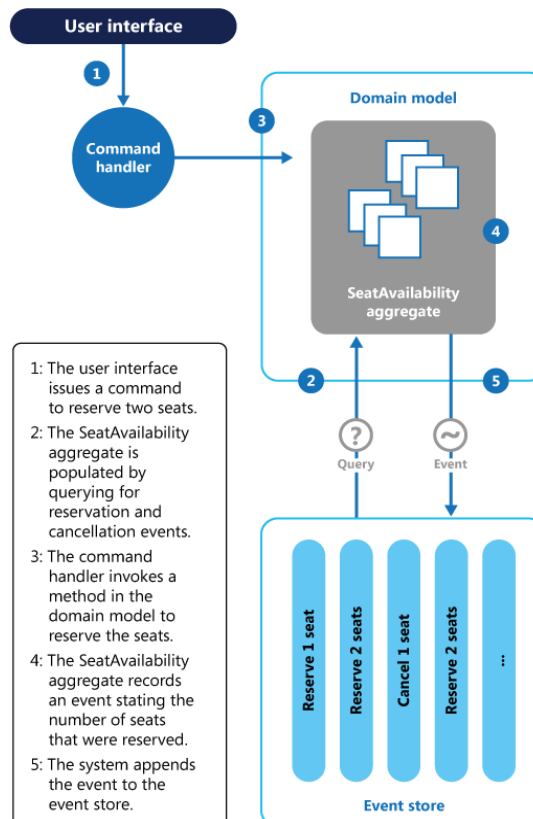
-Ejemplo para entender el patrón:

Ejemplo del ajedrez



-Ejemplo de uso cotidiano:

Ejemplo de gestión de un sistema de administración de conferencias.



-Bibliografía:

<https://martinfowler.com/eaDev/EventSourcing.html>

<https://chriskiehl.com/article/event-sourcing-is-hard>

<https://microservices.io/patterns/data/event-sourcing.html#:~:text=Event%20sourcing%20persists%20the%20state,operation%2C%20it%20is%20inherently%20atomic>

<https://docs.microsoft.com/es-es/azure/architecture/patterns/event-sourcing>

<http://sensetecnic.com/cqrs-and-event-sourcing-for-the-iot/>