

Event Sourcing

In aceasta lucrare voi prezenta ideea de Event Sourcing. Aceasta ideea fundamentala asigura faptul ca orice schimbare facuta la application state, la „starea unei aplicatii” sunt salvate intr-o secventa de evenimente ca si event objects. Aceste „obiecte de evenimente” pot fi folosite, ba mai mult, log-ul de evenimente poate fi folosit pentru a accesa starile anterioare, si pentru a ocupa automate de ajustarile unei stari cu schimbări retroactive.

Sa luam ca si exemplul o companie de transporturi maritime care are mai multe nave pe mare si vor sa vada unde se afla fiecare, ba mai mult fiecare persoana care are un produs transportat peste ocean va vrea sa stie unde produsul comandat este. Compania pune la dispozitia noastra o aplicatie de tracking care ne permite sa stim cand nava pleaca sau ajunge in port.

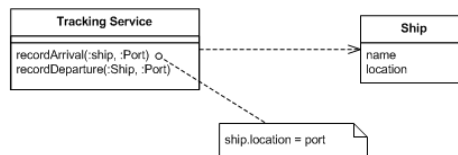
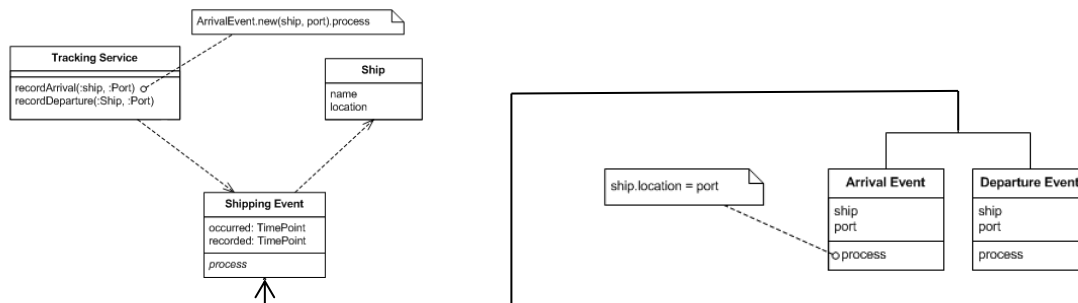


Figure 1: A simple interface for tracking shipping movements.

In aceasta situatie cand serviciul este folosit, ne gasi nava relevanta noua si ii va arat locatia. Obiectul „Ship” inregistreaza starea curenta a navei. Introducand Event Sourcing se va adauga un pas in plus la acest proces.



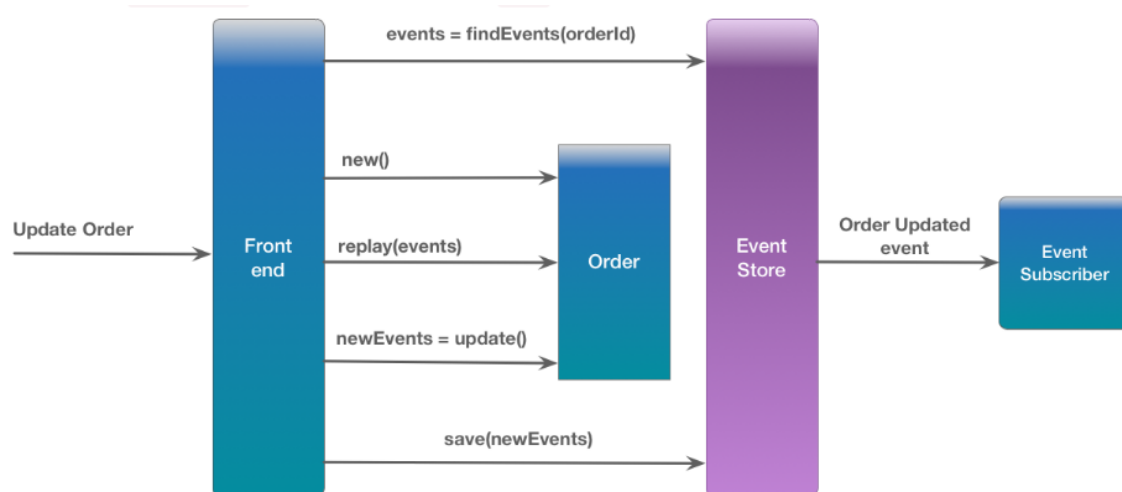
Asta ar parea doar un pas putin necesar, dar partea interesanta va aprea doar dupa ce aplicatia va suferi cateva schimari, deoarece cu serviciul actual se va salva doar ultima schimbare facuta, pe cand folosind Event Sourcing vom putea accesa orice eveniment pe care l-a suferit „Nava nosatra”, adica, nu vedem doar unde aceasta se afla, ci si unde aceasta a fost.

Best Practice:

Cea mai simpla modalitate de a calcula o stare ceruta cu Event Sourcing se realizeaza pronind de la o stare goala prin aplicarea evenimentelor pana se ajunge la acea stare. Acesta poate parea un proces greoi daca exista multe evenimente pana la atingerea obiectivelor.

În multe aplicații este mult mai întâlnită situația de a verifica ultima stare prin care acestea au trecut, dar acest lucru este posibil și prin Event Sourcing, deoarece dacă cineva dorește ultimul eveniment, predeceul nostru oferă capacitatea de built on top.

Folosind Event Sourcing, toate evenimentele vor fi stocate practic ca o bază de date și nu numai, acesta poate fi folosit ca și un API care permite accesarea evenimentului din surse externe. În această arhitectură, requesturile de update pe o entitate, fie ele printr-un HTTP extern sau un eveniment realizat de altă aplicație arată în felul următor:



Avantaje în a folosi Event Sourcing:

- Rezolvă una dintre problemele cheie în implementarea unei arhitecturi bazate pe evenimente, face posibilă publicarea fiabilă a evenimentelor ori de câte ori se schimbă starea.
- Deoarece persistă evenimente în loc de domain object, în cea mai mare parte se evită nepotrivirea impedanței ORM (Object-relational impedance mismatch).
- Furnizează un jurnal al schimbărilor aduse asupra entității de business 100% fiabil.
- Îți oferă posibilitatea de a accesa orice stare la orice moment din prezent sau trecut.

Dezavantaje în a folosi Event Sourcing:

- Este un mod diferit, foarte puțin familiar, deci pentru a fi folosit într-un proiect mare persoanele care lucrează pe el, cel mai probabil vor avea nevoie de un training.
- Pe cât este de eficientă în folosire pe atât este de „ciudată” de injectat într-un proiect, deoarece pentru a putea interoga printre stările prin care a trecut, de exemplu procesul de business, cel mai probabil va trebui să se folosească Command Query Responsibility Segregation pentru a primi datele corecte.

