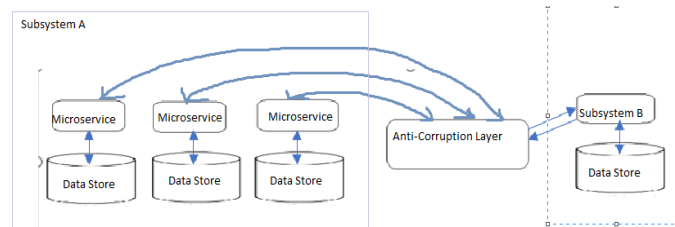


Anti-Corruption Layer

Multe aplicații se bazează pe alte sisteme pentru anumite date sau funcționalități. De exemplu, atunci când o aplicație moștenită este migrată într-un sistem modern, poate fi necesară existența unor resurse vechi. Funcțiile noi trebuie să poată apela sistemul vechi. Acest lucru este valabil mai ales pentru migrația treptată, în care diferite caracteristici ale unei aplicații mai mari sunt mutate într-un sistem modern.

Dacă aplicația noastră are nevoie să se ocupe de o bază de date sau de o altă aplicație a cărei model este nedorit sau inaplicabil modelului pe care îl dorim în propria aplicație, trebuie utilizat **un strat anti-corupție** pentru a traduce în/din acel model și al nostru.



Soluția împotriva coruperii este izolarea diferitelor subsisteme prin plasarea unui strat de anti-corupție între ele. Acest strat **traduce comunicările** între cele două sisteme, permițând un sistem să rămână neschimbat, în timp ce celălalt poate evita compromiterea design-ului și a abordării tehnologice. Pe plan intern, stratul transmite în **ambele direcții** între cele două sisteme, după cum este necesar. Chiar dacă modelul pe care îl folosim este definit de un subsistem extern, cred că este încă logic să existe un strat anti-corupție, indiferent cât de subțire, să restrângem orice schimbări viitoare pe care trebuie să le facem în codul nostru, ca rezultat al modificărilor externe în stratul respectiv.

Cu alte cuvinte, Anti-corruption layer este un strat care ne ajută să păstrăm buna funcționare chiar dacă există o schimbare în sistemele externe de care depinde. De exemplu, imaginați-vă că o aplicație se bazează pe Google Calendar pentru programări și memento pentru evenimente. Pentru a sincroniza datele către și din Google Calendar, va trebui să mapăm datele dintre cele două sisteme, adică aplicația noastră și Google Calendar. Acest lucru se datorează faptului că modelele și structura vor fi diferite în aceste sisteme. Dacă vom dezvolta un ACL ar trebui să asigurăm:

- traducerea modelului și protecția împotriva defectării
- monitorizarea
- îmbunătățirea în cadrul integration testing

Probleme și considerații:

- Stratul anti-corupție poate adăuga latență apelurilor efectuate între cele două sisteme
- Stratul anti-corupție adaugă un serviciu suplimentar care trebuie gestionat și menținut
- Trebuie luată în considerare posibilitatea de a fi necesare mai multe straturi anti-corupție.
- Asigurarea că tranzacția și consistența datelor sunt menținute și pot fi monitorizate