***Event Driven Architecture***

* ***Event sourcing*** adalah pendekatan untuk mempertahankan entitas bisnis. Sebagai gantinya memodelkannya dengan keadaan statis yang dapat Anda ubah seiring waktu, Anda memodelkannya sebagai urutan peristiwa yang tidak dapat diubah. Misalnya ***customer*** tidak akan ada tabel ***customer*** didata tetapi urutan kejadian CustomerChanged.

Misalkan kita mempunyai ***customer*** dengan nama john. Kita mengubah namanya menjadi Jhonas dan kami perhatikan kami melakukan kesalahan sehingga menggantinya lagi menjadi john. Dalam metode persistensi tradisional, jika Anda memeriksa data setelahnya menerapkan perubahan ini, Anda hanya akan melihat nama John, tetapi ketika menggnakan metode ***event sourcing*** maka akan ada urutan kejadian dari perubahan nama tersebut sampai perubahan terakhir.

CustomerChanged -> name: John (created), CustomerChanged -> name: Jhonas (mistake), CustomerChanged -> name: John (correction).

Contoh umum yang biasanya digunakan dengan metode **event sourcing** ini adalah didalam aplikasi perbankan, yang pola ini sangat masuk akal.

* **CQRS (Command-Query Responsibility Segregation)** adalah pola format **query model** (untuk membaca) dan **comman model** (untuk menulis) dipisahkan, sehingga memungkinkan pendekatan membaca yang sangat cepat dan sistem yang jauh lebih kompleks.

Kelebihan dan kekurangan Event-Driven Architecture

* Kelebihan
  + Loose Coupling
  + Transactions
  + Fault Tolerance
  + Orchestration and Monitoring
* Kekurangan