Query or Command

(Short) Introduction to CQRS

Zbyszko Papierski
Solutions Architect @ allegro.pl
Twitter: @ZPapierski
Linkedin: https://www.linkedin.com/in/zbyszko

C.R.U.D.

Command
Query
Responsibility
Segregation

Bertrand Meyer

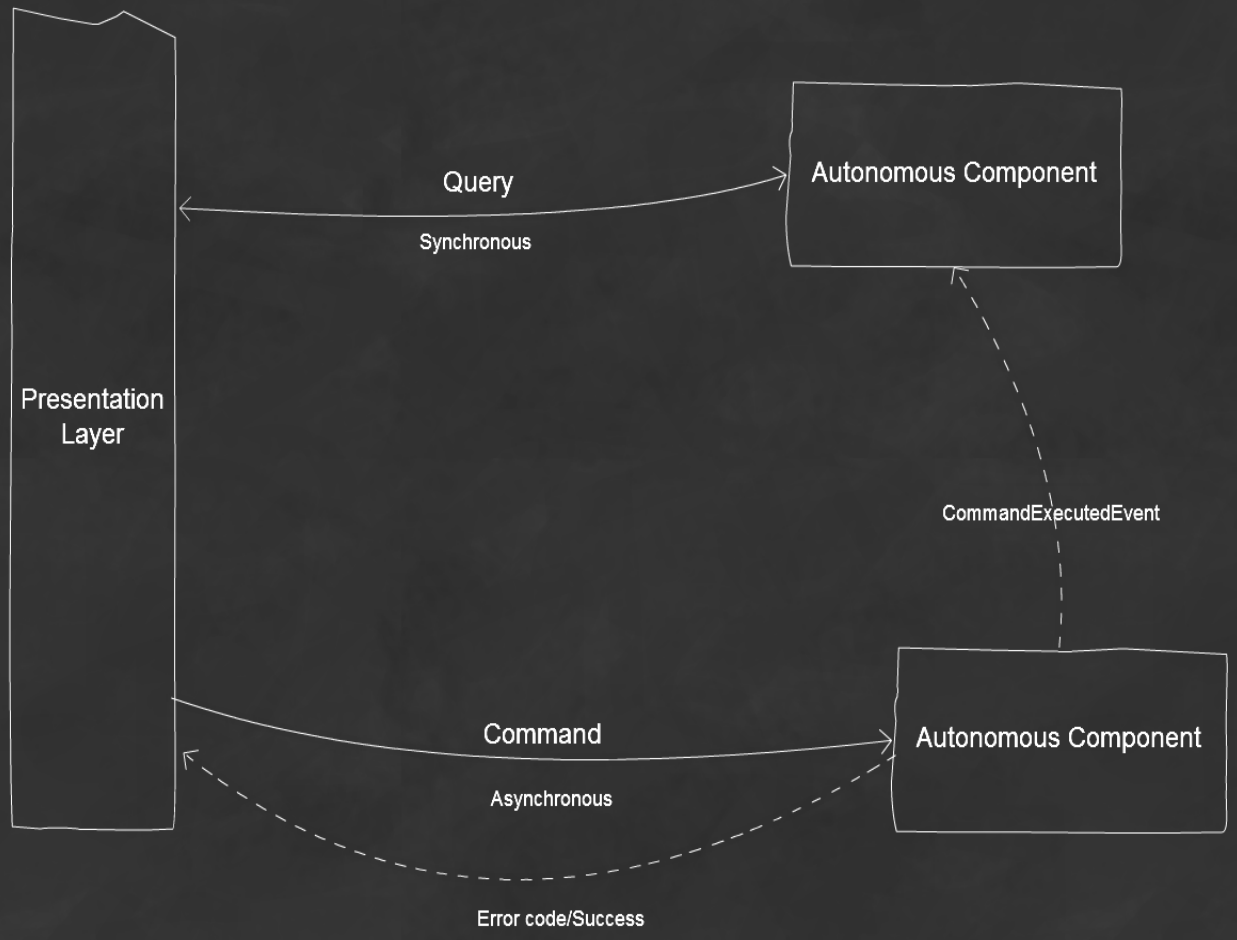
Martin Fowler

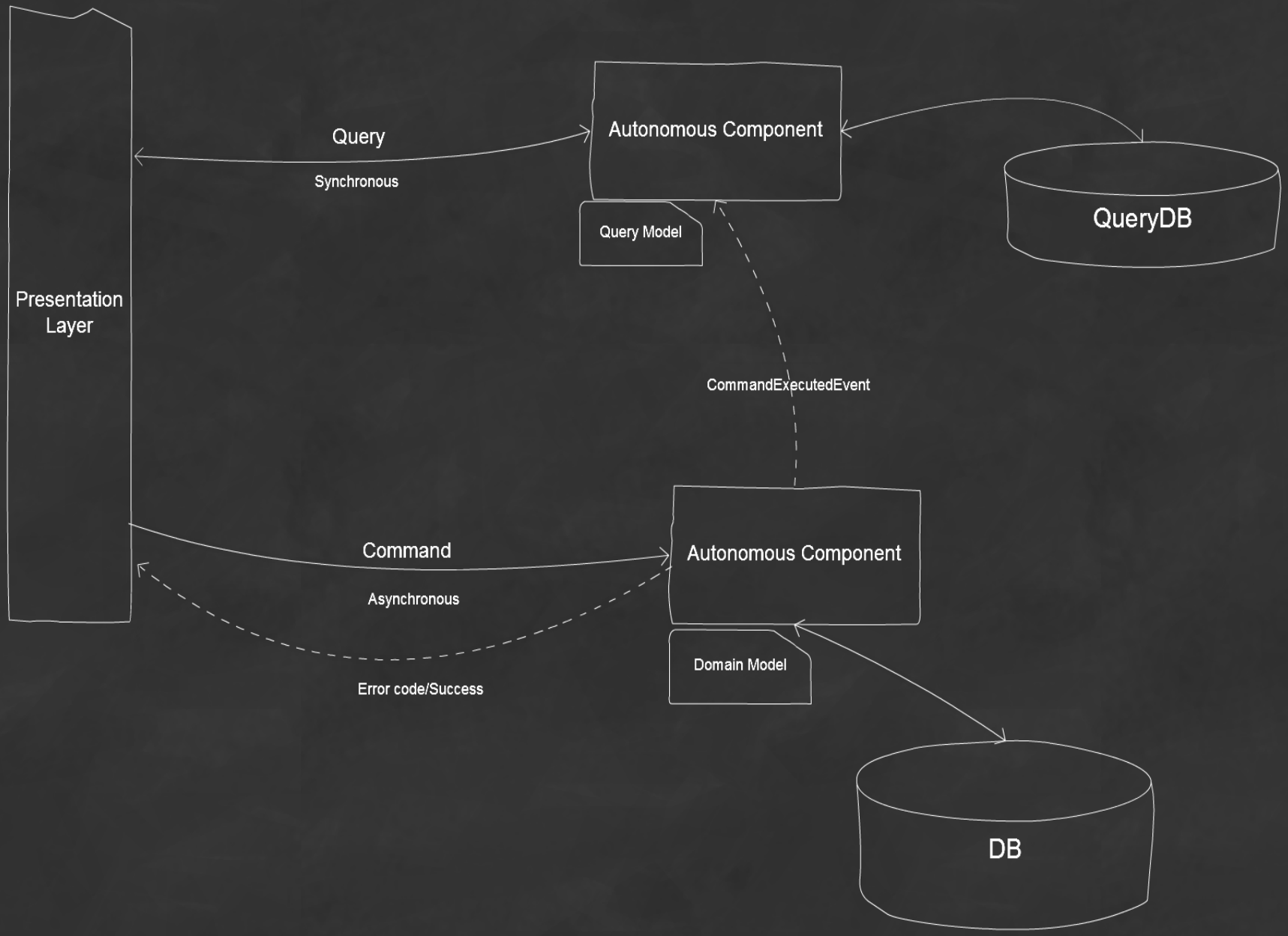
Greg Young

Command (vs) Query

Responsibility Segregation


Presentation Layer
Query
Autonomous Component
Command
Autonomous Component


Presentation Layer
Query
Synchronous
Autonomous Component
CommandExecutedEvent
Command
Asynchronous
Autonomous Component
Error code/Success


Presentation Layer
Query
Synchronous
Autonomous Component
Query Model
QueryDB
CommandExecutedEvent
Command
Asynchronous
Error code/Success
Autonomous Component
Domain Model
DB

NOT

one style to rule them all

scalable
but
eventually consistent

more elastic

but

more (globally) complex

What’s next?

Event Sourcing

Task Based UI

Google says you don’t need that save button…

Technology Stack?

Axon Framework

Jdon

Anything you like

Stack Example - communication

- Kafka - distributed pub/sub
- AngularJS - js framework with good ansychronous mechanisms
- Vert.X - asynchronous web communication
- RxJava - Reactive view changes

Stack Example - db solutions

- any db you need
- Cassandra - extremely fast writes and pretty fast reads
- MongoDB - fast document store
- Akka Persistence - great implementation of Event Sourcing

Key Takeaways

- As always - “right tool for the job”
- Build your views asynchronously
- Consider what is the best model for each query and each command
- Consider your consistency and performance requirements

Mandatory Links Slide

- https://cqrs.wordpress.com/ - not updated, but interesting read
- http://martinfowler.com/bliki/CQRS.html - Martin Fowler on CQRS
- http://www.axonframework.org/
- http://en.jdon.com/

???