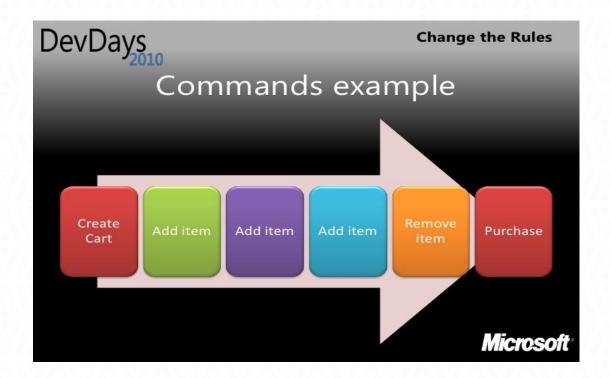


CQRS+ES

example

Command

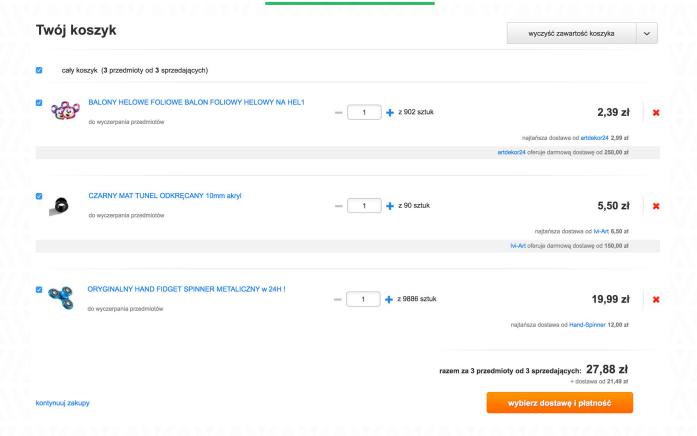


Command

```
// Purchase is a Domain Model, responsible for taking COMMANDS in order to
type Purchase struct {
    discount float64
    shipping Address
    products map[int32]Product
    finished bool
// AddProduct COMMAND
func (p *Purchase) AddProduct(v Product) error { ... }
// ChangeProductQuantity public COMMAND
func (p *Purchase) ChangeProductQuantity(pid int32, quantity int) error { ... }
// GiveDiscount COMMAND
func (p *Purchase) Discount(code string, cp CouponProvider) error { ... }
```



Query

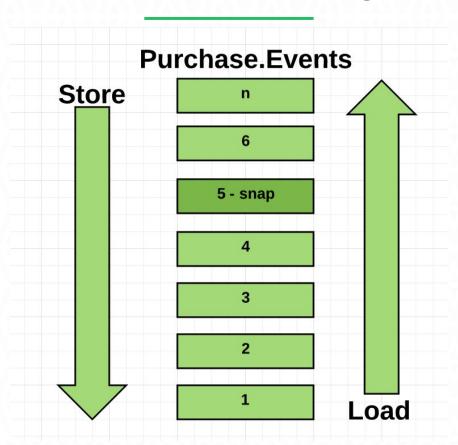




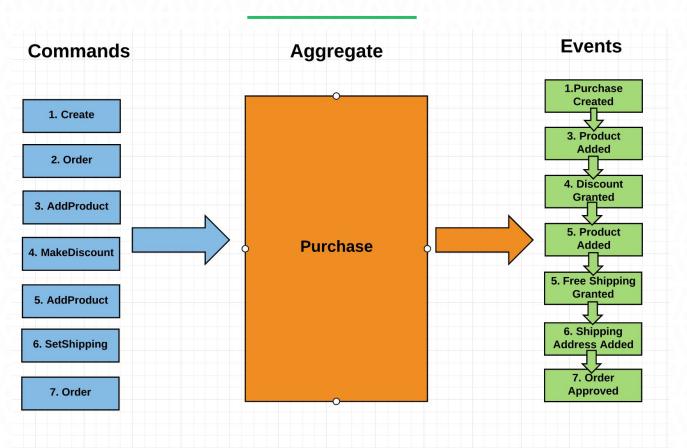
Query

```
// ShoppingCart is a Domain Model, responsible for taking QUERIES.
type ShoppingCart struct {
    total float64
    shipping float64
    products []Item
// TotalPrice QUERY
func (s ShoppingCart) TotalPrice() float64 {...}
// ShippingPrice QUERY
func (s ShoppingCart) ShippingPrice() float64 {...}
// Item QUERY
func (s ShoppingCart) Item() []Item {...}
```

Event Sourcing



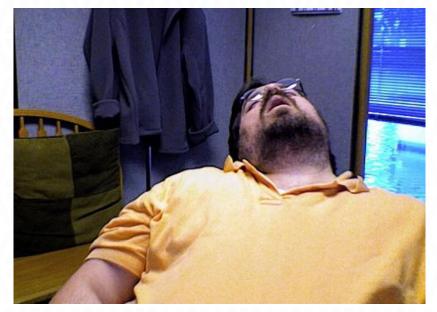
Event Sourcing



Why?

Write data safe

Read data fast



Let's make some code...

Thank you

SILICON VALLEY · NYC · PORTLAND · POLAND · LONDON

TIVIX