



CQRS AND EVENT SOURCING



Metehan Gültekin

Java Developer at

@DefineX

Co-organizer at @Folksdev

Metehan Gültekin graduated in Computer Engineering and has been working as a Java Developer for 3 years. He is currently working at Definex. He enjoys learning new things and sharing his knowledge with the community. Driven by this passion, he shares his experiences through presentation videos and technical presentations on YouTube.

X @metehan_gltk

in n @mgmetehan

Q @mgmetehan

C Q R S

Command
(Komut)

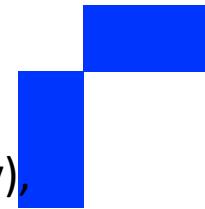
Query
(Sorgu)

Responsibility
(Sorumluluk)

Segregation
(Ayristirma)

CQRS == CQS

??



Bir fonksiyon ya veriyi değiştirmeli (command) ya da veriyi döndürmeli (query),
ancak her ikisini aynı anda yapmamalıdır.

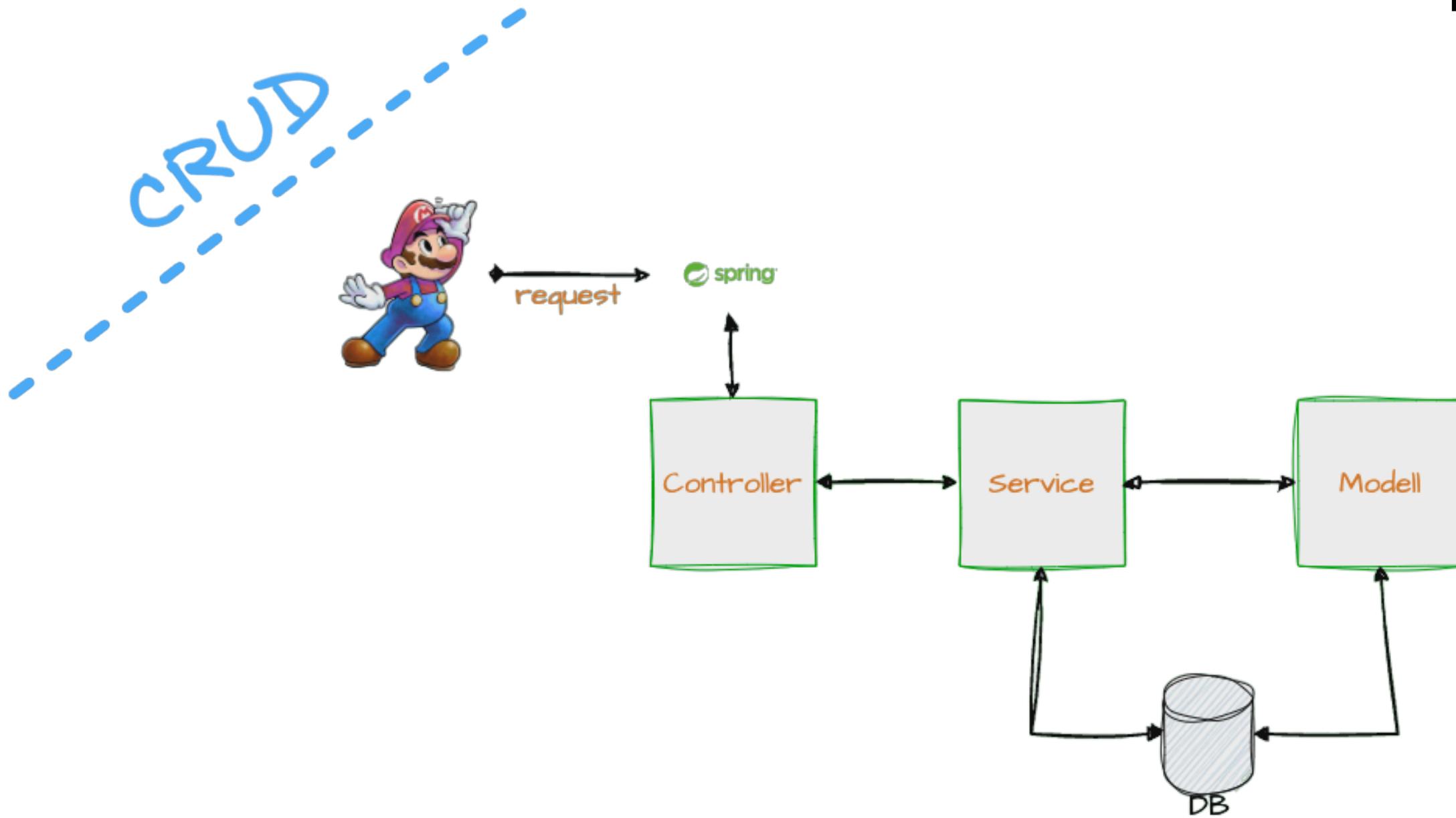
1988 yılında ortaya atılmıştır.

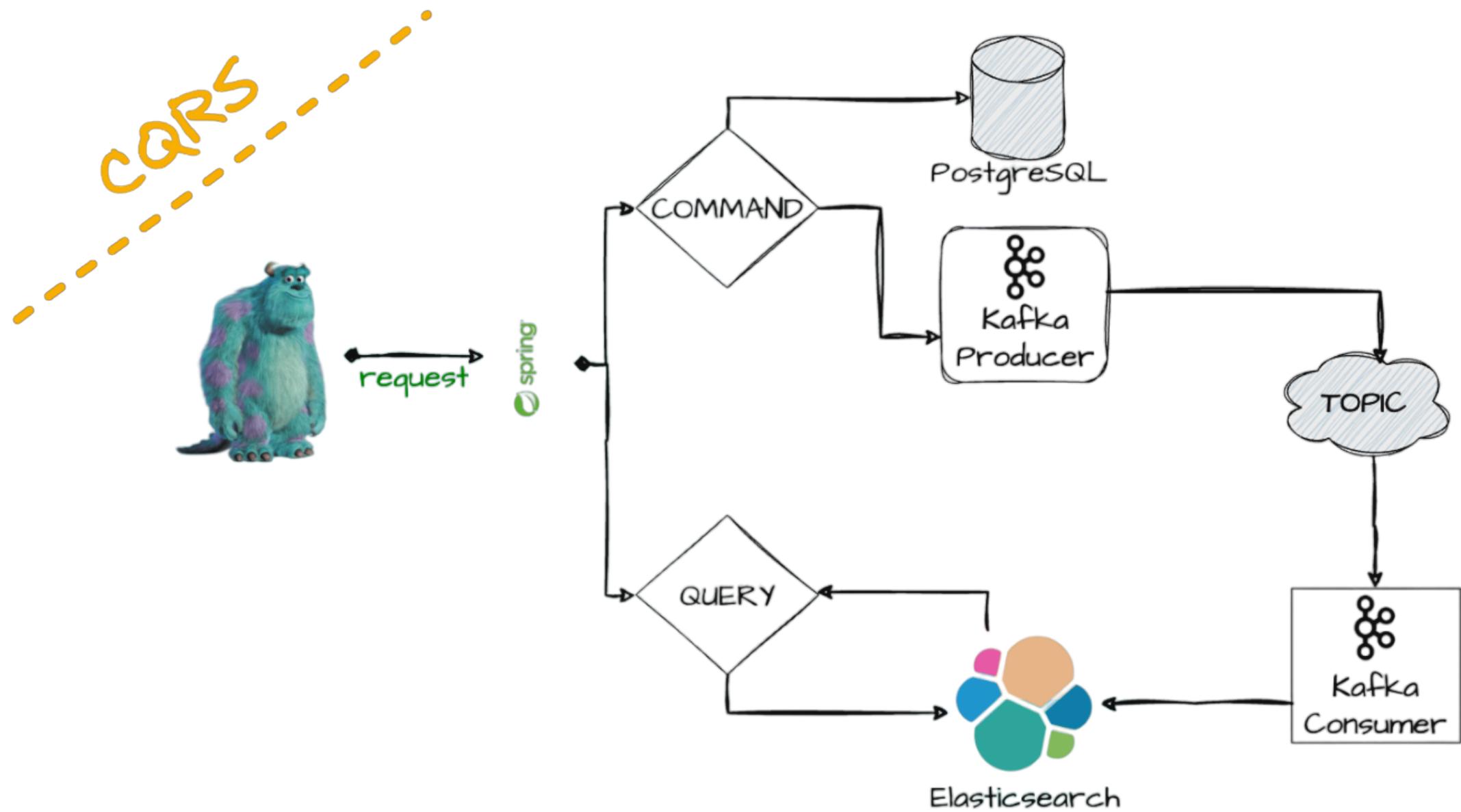


CQRS vs CRUD

read /
Query == /GET
write /
Command == /PUT
/DELETE

Create == /POST
Read == /GET
Update == /PUT
Delete == /DELETE





CQRS'nin avantajları:

- ✖ Çeşitli Çözümler Sunma Esnekliği
- ✖ Hata ve Kesinti İzolasyonu
- ✖ Daha İyi Güvenlik Kontrolü
- ✖ Read Model Optimizasyonu
- ✖ Farklı Domain Logic Üzerinde Çalışma
- ✖ Bakım Kolaylığı
- ✖ Geliştirme Hızı
- ✖ Bekleme Süresi Olmadan İşlemlerin Yürütülmesi
- ✖ Bağımsız Ölçekleme

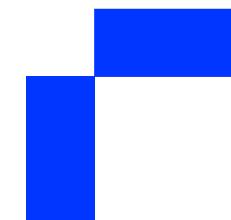
CQRS'nin dezavantajları

- ✖ Karmaşıklık ve Öğrenme Eğrisi
- ✖ Artan Geliştirme Zamanı
- ✖ Veri Senkronizasyonu Zorlukları
- ✖ İki Veritabanı Kullanımı & Maliyet Artışı
- ✖ Eğitim ve Adaptasyon Zorlukları
- ✖ İş Süreçleri Arasındaki Kopukluk
- ✖ Artan İletişim İhtiyacı



CQRS'i

Ne Zaman Kullanmalıyım
ve Kullanmamalıyım?







CQRS Ne Zaman Kullanılmalı ?



- ✖ Karmaşık İş Mantığı
- ✖ Bağımsız Geliştirme ve Dağıtım
- ✖ Farklı İhtiyaçlara Sahip Modeller
- ✖ Hata ve Kesinti İzolasyonu || Uygulama Güvenilirliği
- ✖ Yüksek Trafikli Sistemler
- ✖ Event Sourcing Kullanımı
- ✖ Farklı Veritabanı ve Teknoloji Kullanımı

CQRS'i Ne Zaman Kullanmamalıyız?



- ✗ Proje Karmaşıklığı ve Ölçeği Küçükse
- ✗ Ekip Deneyimi ve Eğitimi Yetersizse
- ✗ Ekstra Karmaşıklık İstenmiyorsa
- ✗ Maliyet ve Kaynak Sınırılamaları
- ✗ Hızlı Geliştirme ve Prototip İhtiyacı
- ✗ Veri Tutarlılığı Mutlak Öncelik Değilse
- ✗ Event Sourcing Gereksinimi Yoksa
- ✗ Performans Kritik Değilse

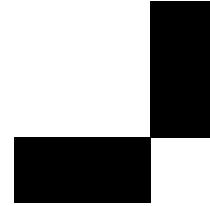
CQRS'in Kullanılabileceği Senaryolar

Netflix

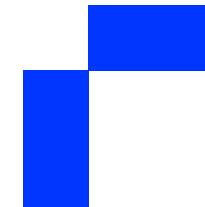
E-Ticaret Platformları

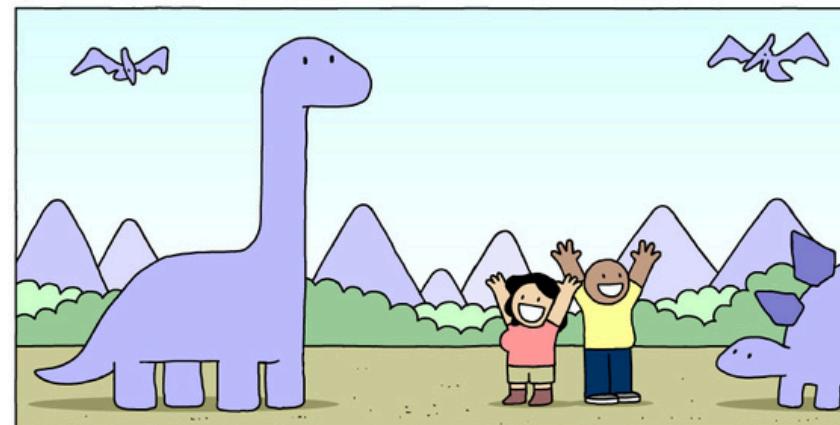
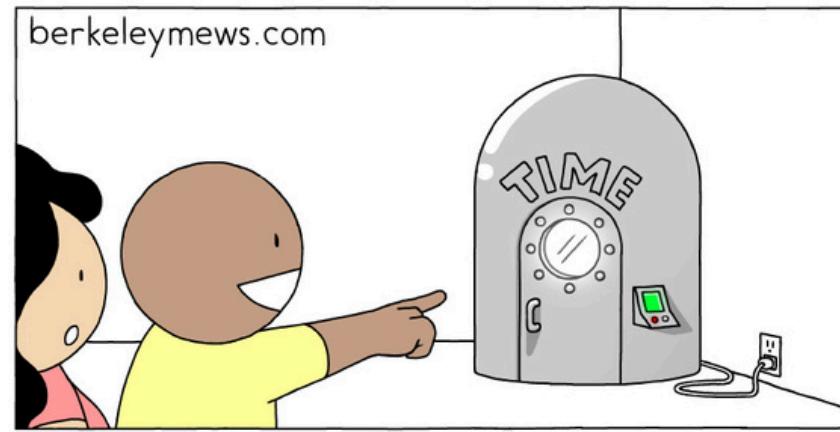
Uber

Twitter

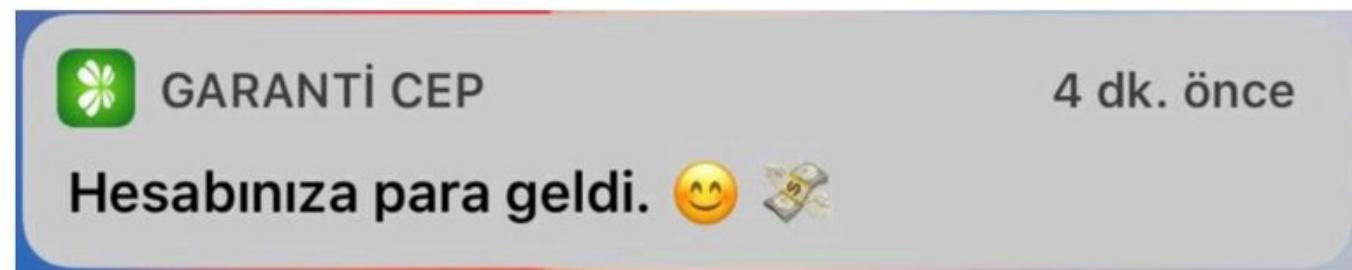


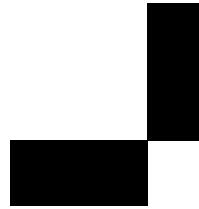
Event Sourcing (Olay Tabanlı Mimari)



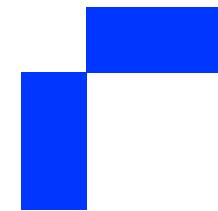


Event?



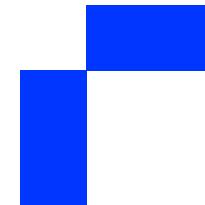


State?





Aggregate?



Total Item Stock
8

Items Created
8

t1

t1
↓



Total Item Stock
7

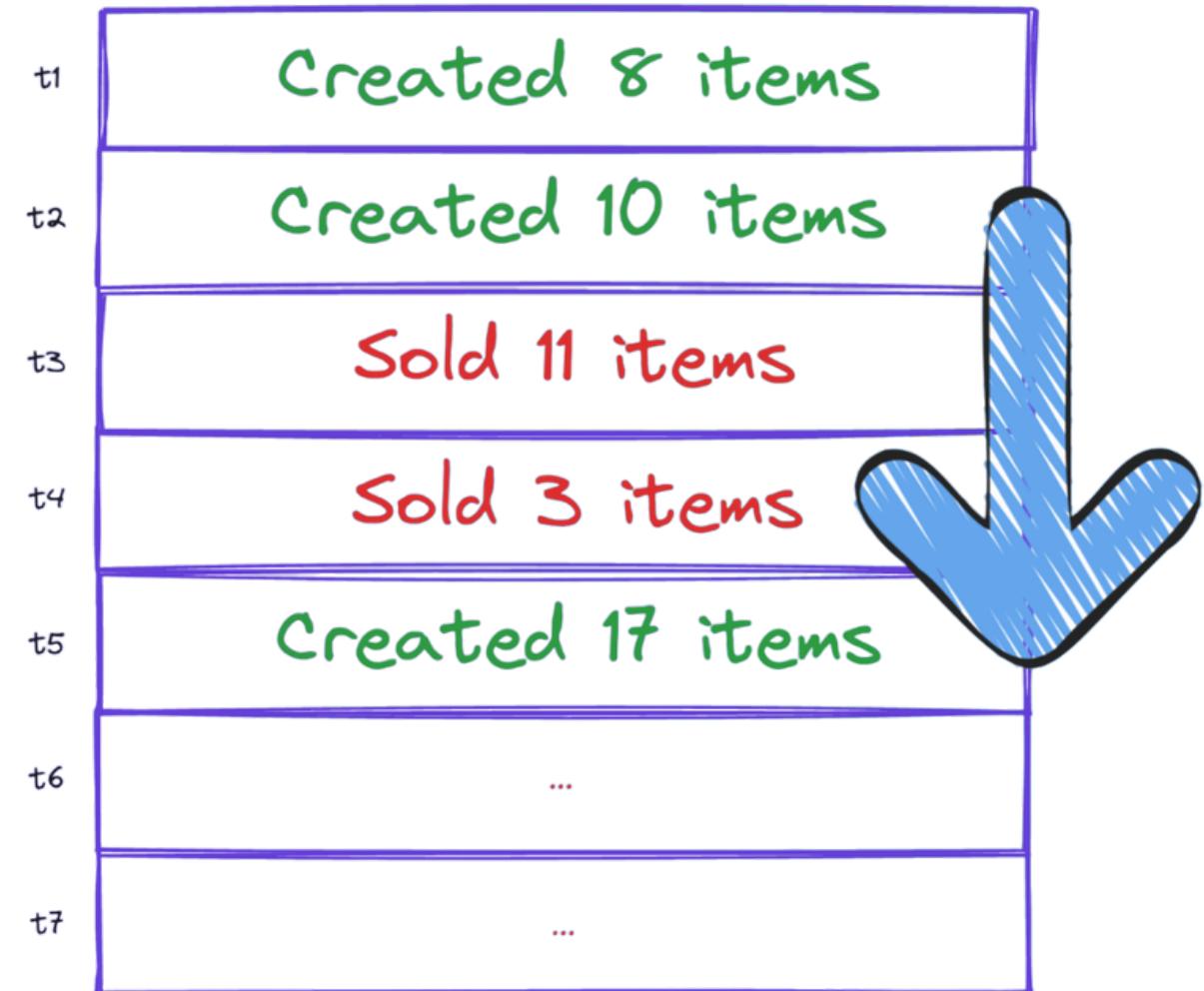
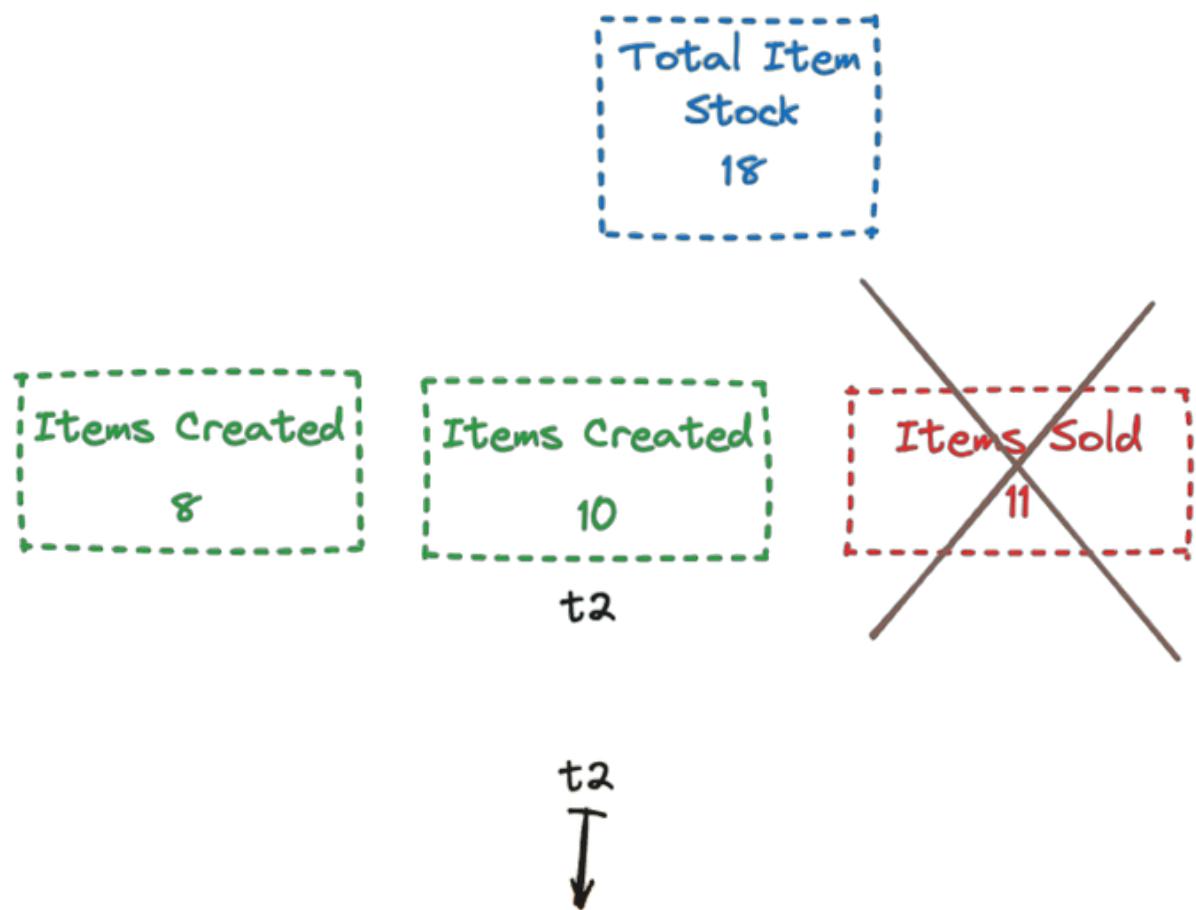
Items Created
8

Items Created
10

Items Sold
11

t3

t3
↓



Event Sourcing == Event-Driven Architecture ??

- ✗ Event Sourcing, uygulamanın durumunu her durum değişikliğini temsil eden event aracılığıyla kaydeden bir mimari desen
- ✗ Event-Driven Architecture ise bileşenler arasında event'lerin yaylanması ve dinlenmesine dayanan bir iletişim prensibidir

Entities vs Events

id	order_name	order_state	create_date	update_date
1	Mac M2	Created	t1	-

t1

Order Created

event_id	event_type	entity_type	entity_id	event_data	event_date
80	Order Created	Order	101	{...}	t1

id	order_name	order_state	create_date	update_date
1	Mac M2	Approved	t1	t2

t2

Order Approved

event_id	event_type	entity_type	entity_id	event_data	event_date
80	Order Created	Order	101	{...}	t1
81	Order Approved	Order	101	{...}	t2

Order Shipped

id	order_name	order_state	create_date	update_date
1	Mac M2	Shipped	t1	t3

t3

event_id	event_type	entity_type	entity_id	event_data	event_date
80	Order Created	Order	101	{...}	t1
81	Order Approved	Order	101	{...}	t2
82	Order Shipped	Order	101	{...}	t3
...
...

Event Sourcing'in avantajları

- ✖ Geçmiş İzleme ve Denetleme
- ✖ Hata Ayıklama Kolaylığı
- ✖ Paralel İşleme Yeteneği
- ✖ Durum Geri Yükleme Yeteneği
- ✖ İş Zekası ve Raporlama
- ✖ İleriye Dönük Uyumluluk

Event Sourcing'in dezavantajları

✗ Karmaşıklık

✗ Öğrenme Eğrisi

✗ Performans

✗ Veritabanı Boyutu

✗ Analistik Zorluklar

✗ Sistem Yeniden İnşası

Event Sourcing'in Kullanılabileceği Senaryolar



- Bankacılık ve Finansal İşlemler
- E-Ticaret Sistemleri
- Lojistik ve Tedarik Zinciri Yönetimi



That's all folks!



That's all folks!