Microservices

Adam Seidel | IT Specialist
Peter Butkovic | IT Architect
Kosice, Slovakia | 04.2021

Overview

Who we are?
Microservices
Demo app Architecture
Microservice Patterns
Microservices in Tax office
Quiz

http://bit.ly/msshowcase
http://sli.do **#248 673**

Who we are?

Adam Seidel

- Studied informatics @ FEI TUKE
- 7 years of experience in IT
  🇸🇰 IBM ☺
- Java | Microservices | Cloud
- Biking, Hiking, Fishing

Who we are?

Peter Butkovic

- Studied informatics @ **FEI TUKE**
- 15+ years experience in IT

  🇩🇪 Atron, Teradata

  🇸🇰 Siemens, **IBM**
- Java | Cloud | Microservices | Architecture | DevOps | OSS
- Hobbies: my 4 kids, bike, garden, Bible reading

Microservices

- Approach of developing a single application as a suite of small services
  - Owned by a small team
  - Loosely coupled
  - Independently deployable
  - Organized around business capabilities
  - Highly maintainable and testable

Source: https://microservices.io

How big is the Microservice?

Two Pizza Rule: *If a team can’t be fed with two pizzas, it’s too big.*

by Jeff Bazos

7 people +/- 2

Source: https://www.axcelerate.com.au/post/a-tale-of-two-pizzas-the-key-to-team-success

Microservices pros and cons

Pros

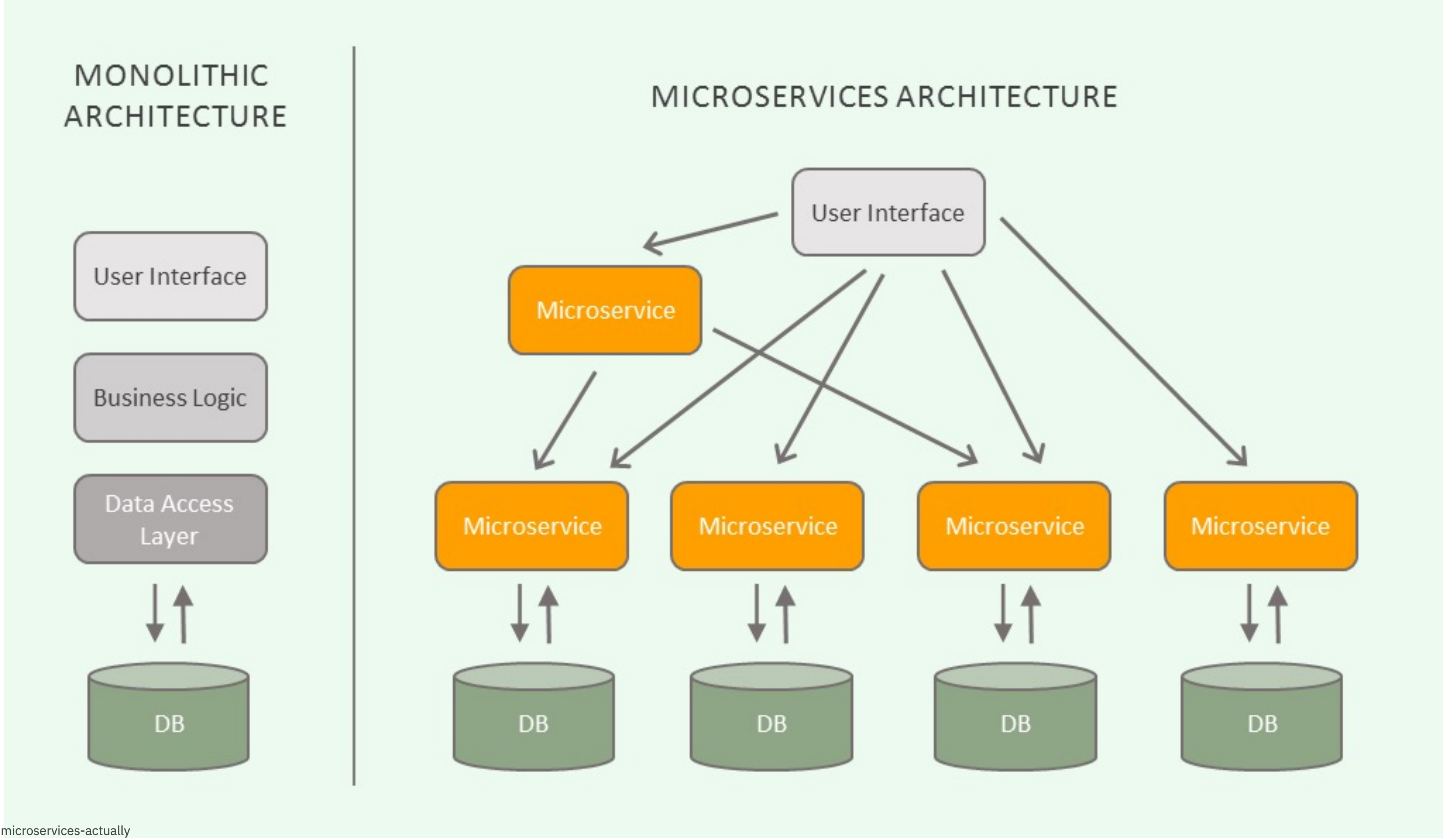
- Better fault isolation
- Easier to understand
- Faster deployments
- Scalability
- Each microservice can have its own technology stack

Cons

- Additional complexity
- May be demanding on HW resources
- Testing
- Debugging
- Faster deployment =/= easier deployment
- Used when not neccessary

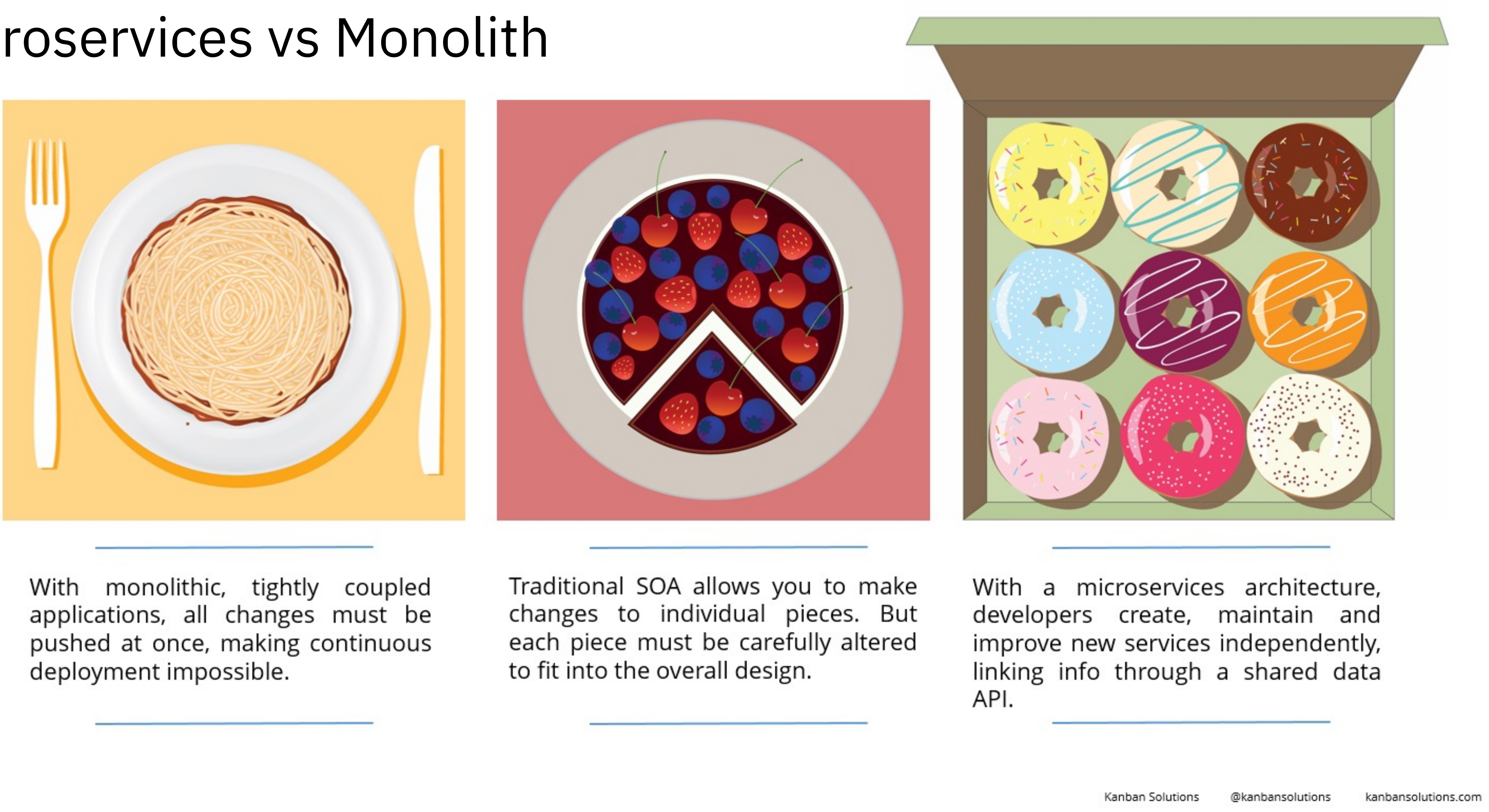
Source: https://martinfowler.com/articles/microservices.html

Microservices vs Monolith



https://dzone.com/articles/what-are-microservices-actually

Microservices vs Monolith



With monolithic, tightly coupled applications, all changes must be pushed at once, making continuous deployment impossible.

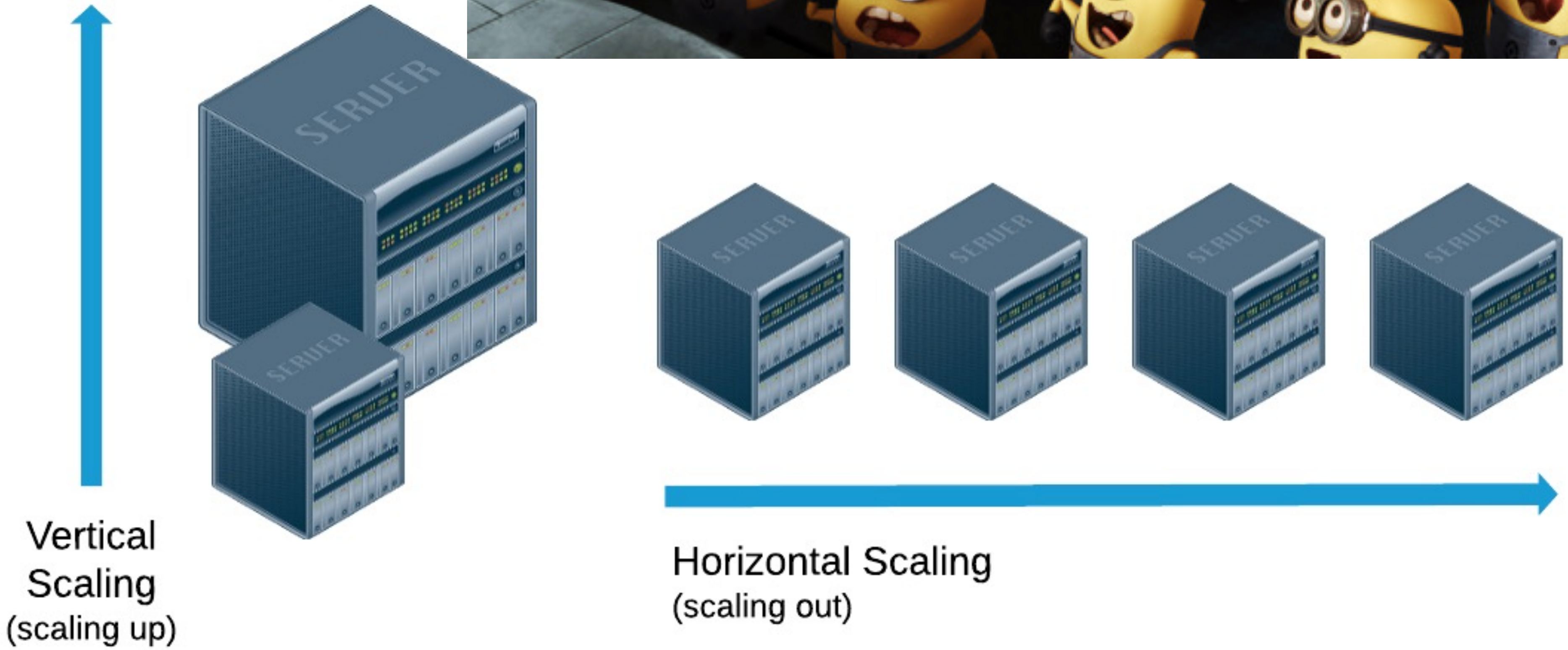
Traditional SOA allows you to make changes to individual pieces. But each piece must be carefully altered to fit into the overall design.

With a microservices architecture, developers create, maintain and improve new services independently, linking info through a shared data API.

Kanban Solutions @kanbansolutions kanbansolutions.com

https://dzone.com/articles/what-are-microservices-actually

Scaling horizontally vs vertical

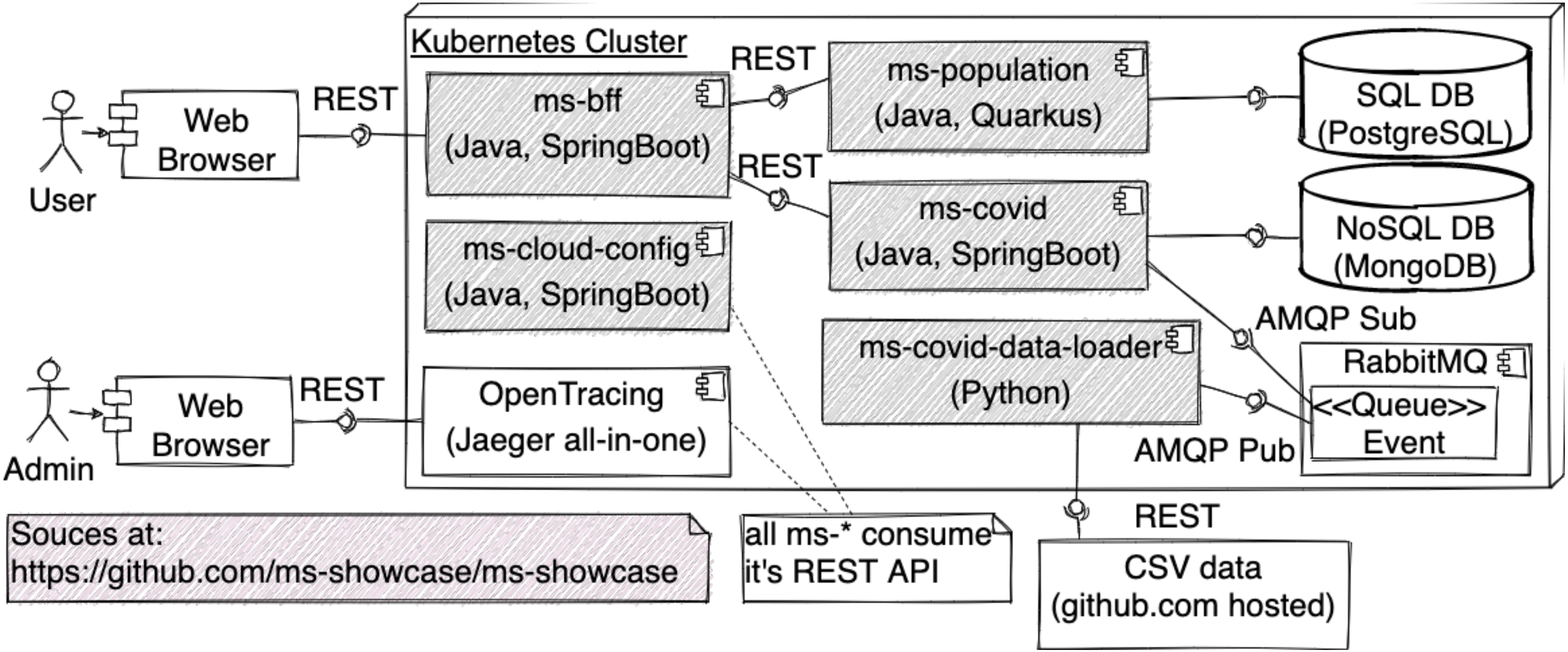


Vertical Scaling
(scaling up)

Horizontal Scaling
(scaling out)

Source: https://www.section.io/blog/scaling-horizontally-vs-vertically/

Demo app (ms-showcase) Architecture



Kubernetes Cluster
User
Web Browser
REST
ms-bff (Java, SpringBoot)
REST
ms-population (Java, Quarkus)
SQL DB (PostgreSQL)
REST
ms-covid (Java, SpringBoot)
NoSQL DB (MongoDB)
ms-cloud-config (Java, SpringBoot)
AMQP Sub
ms-covid-data-loader (Python)
RabbitMQ
<<Queue>> Event
AMQP Pub
Admin
Web Browser
REST
OpenTracing (Jaeger all-in-one)
REST
CSV data (github.com hosted)

Souces at:
https://github.com/ms-showcase/ms-showcase

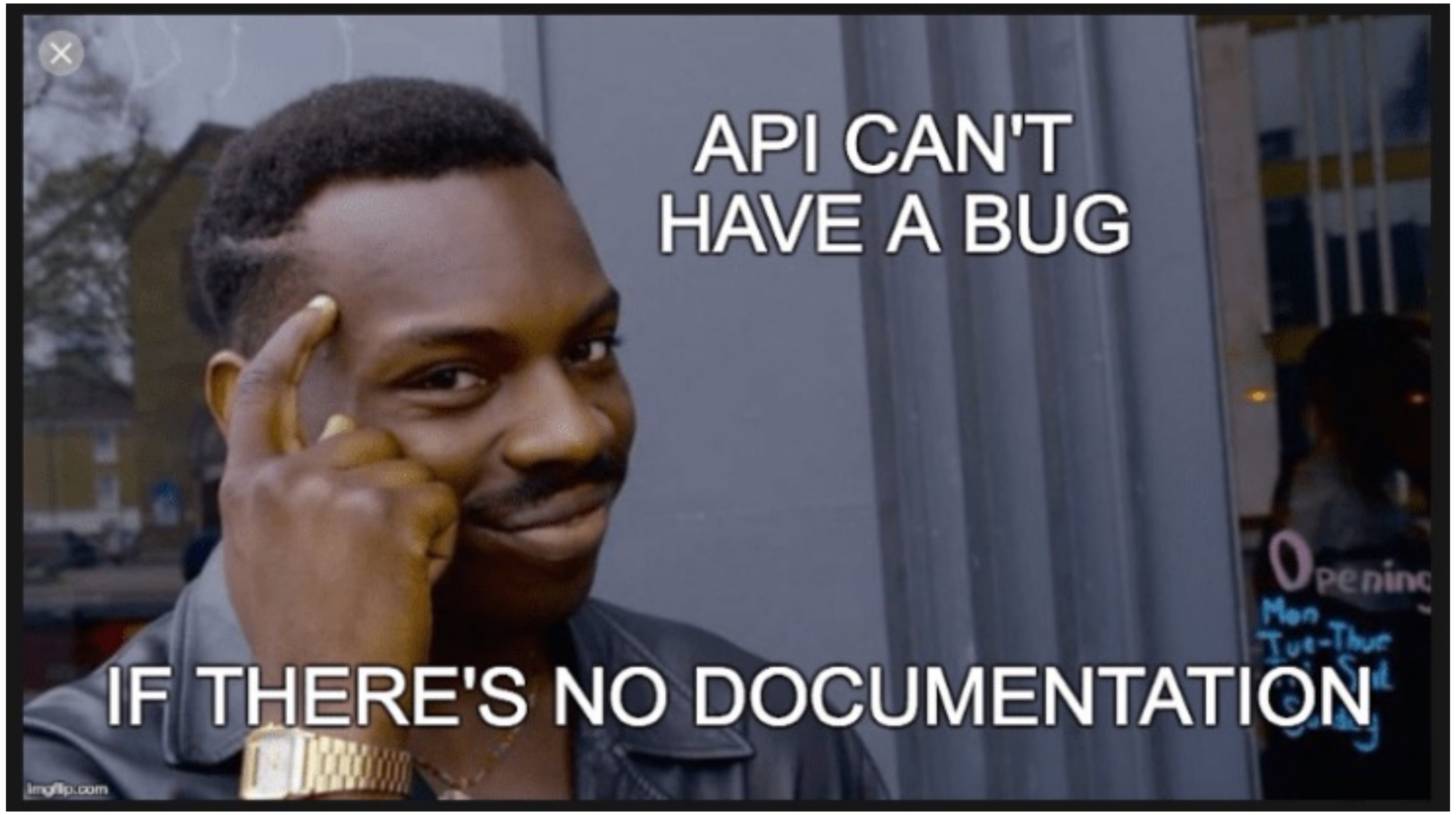
all ms-* consume it's REST API

API

- A **microservice API** is a **contract** between the service and its clients.
- You can evolve a microservice independently only if you do not break its API contract, which is why the **contract is so important**.

OPENAPI INITIATIVE

AsyncAPI



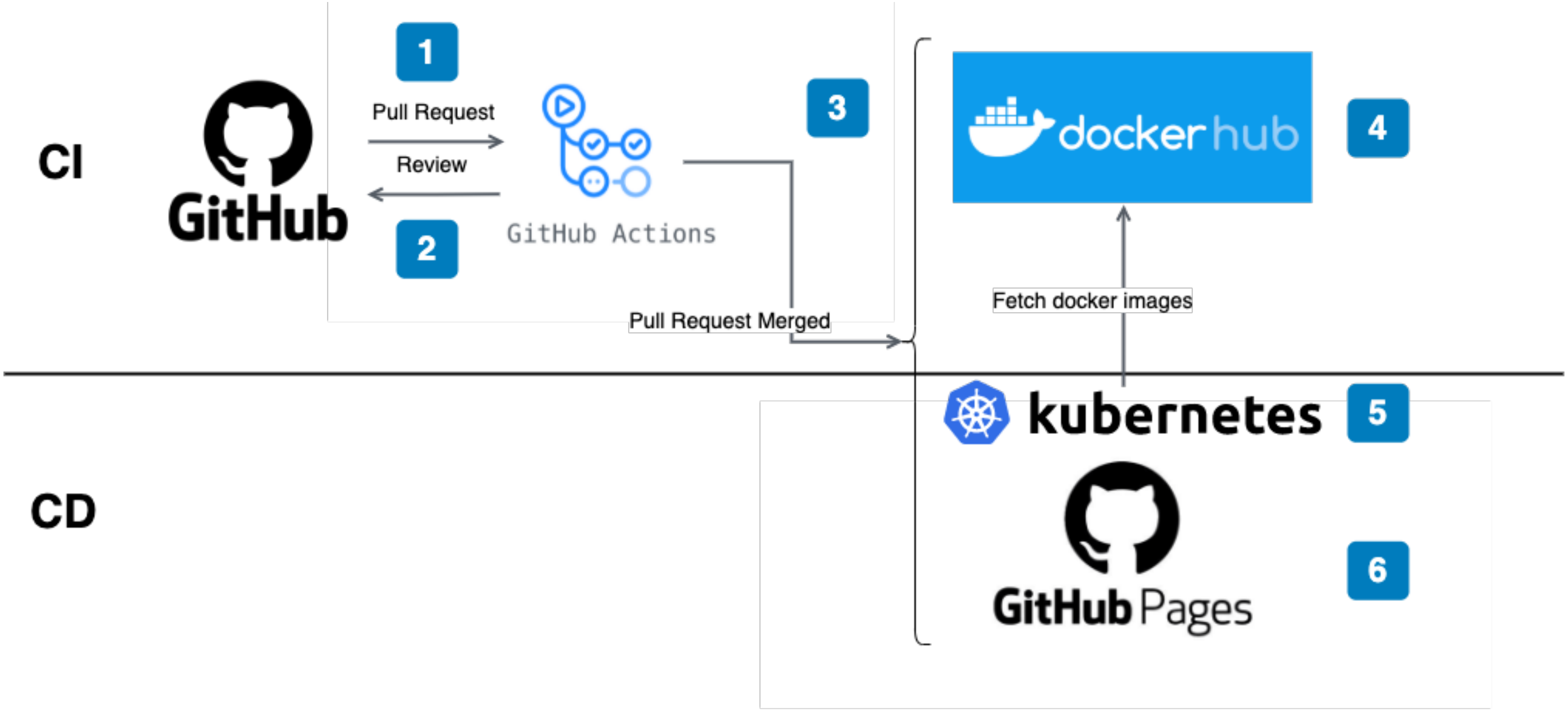
Source: https://docs.microsoft.com/en-us/dotnet/architecture/microservices/architect-microservice-container-applications/maintain-microservice-apis

12 factor apps = cloud native

- https://12factor.net
- Codebase, Dependencies, *Config*, Backing Services, Build, release, and Run, Processes, Port Binding, Concurrency, Disposability, Dev/prod parity, Logs, Admin processes

Source: https://makeameme.org/meme/automate-everything

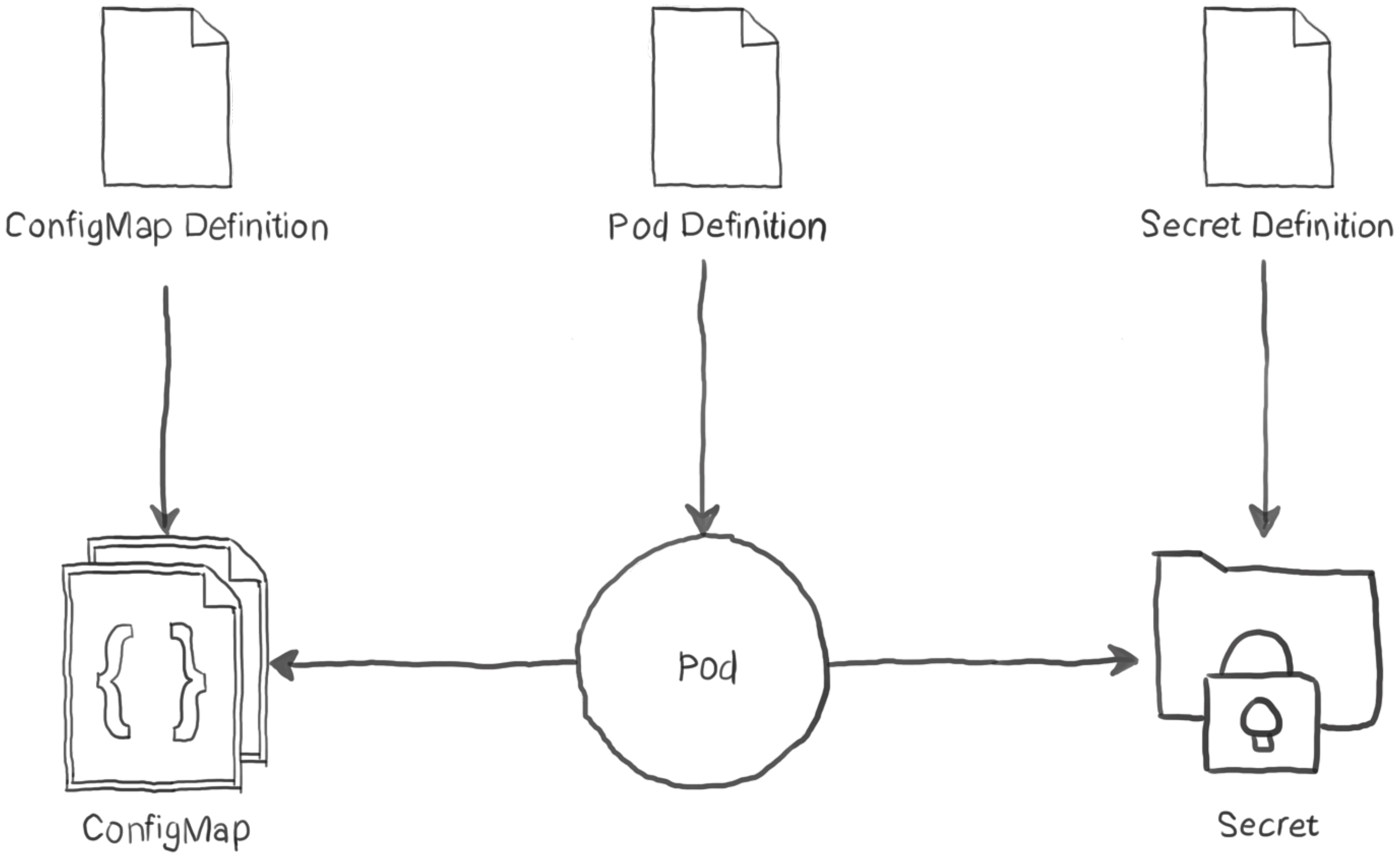
Demo app (ms-showcase) CI/CD



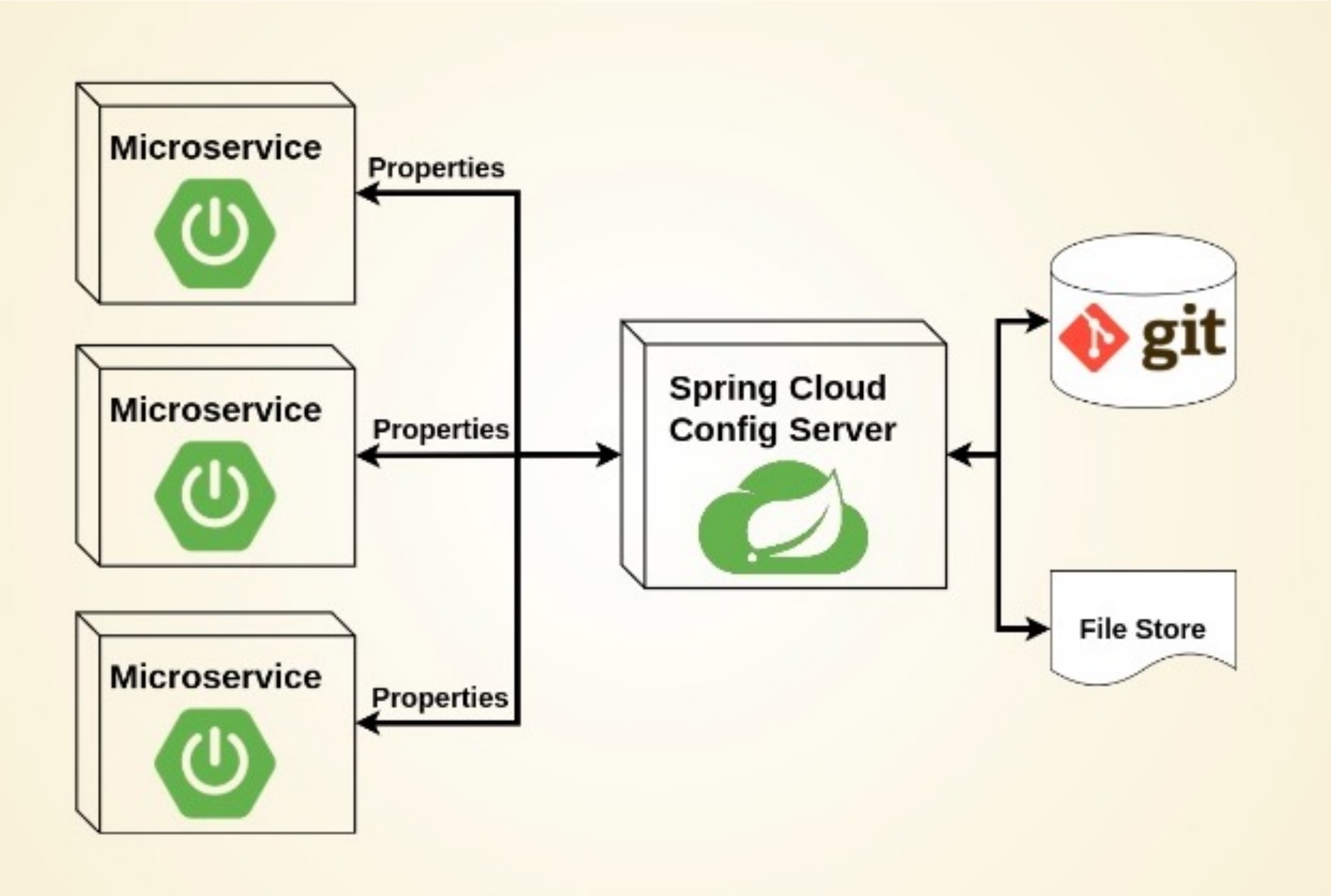
Source:

12 factor apps - Config

k8s config maps / Secrets



Spring Cloud Config

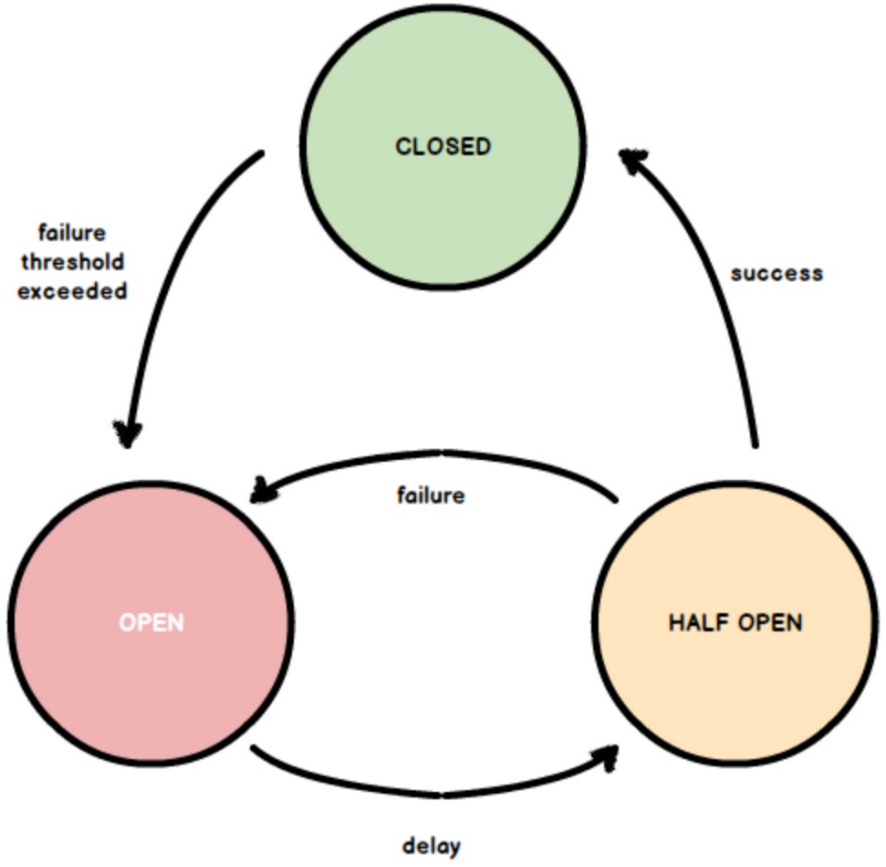


Source: https://www.magalix.com/blog/the-configmap-pattern
https://medium.com/walmartglobaltech/manage-application-configuration-using-spring-cloud-config-80b27ecb34b7

Fault tolerance

***Fault tolerance** is the property that enables a system to continue operating properly in the event of the failure of some of its components.*

- Timeouts
- Retries
- Circuit breaker
- ……



Source: https://medium.com/@jegasingamjeyanthasingam/circuit-breaker-pattern-for-microservices-eb71569dc44d

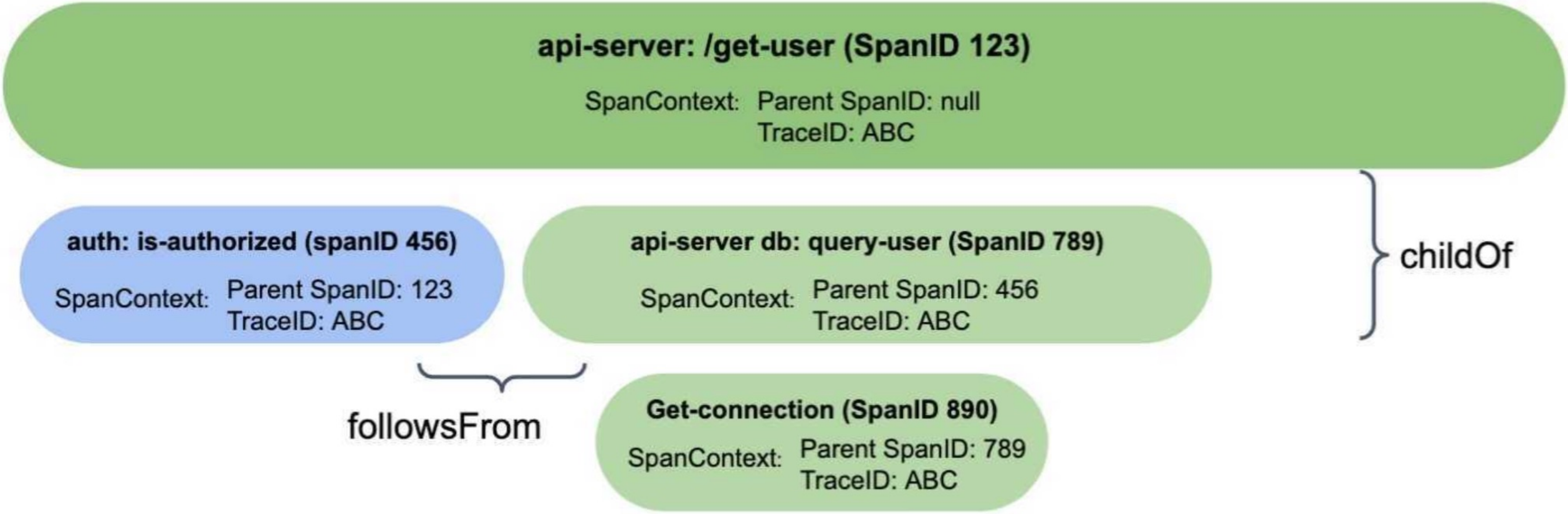
Distributed tracing

- **OpenTracing API** = vendor neutral standard.
- Implementations: **Zipkin** by Twitter, **Jaeger** by Uber
- Unique **Trace ID** propagated across components involved in handling the request.
- Every service adds new **Span ID** to the trace.
- Services involved in the trace can add **metadata** (e.g.: start and stop timestamps, …).

WE REPLACED OUR MONOLITH WITH MICROSERVICES
SO THAT EVERY OUTAGE COULD BE MORE LIKE A MURDER MYSTERY

Source: https://makeameme.org/meme/we-replaced-our

Distributed tracing



api-server: /get-user (SpanID 123)

SpanContext: Parent SpanID: null
TraceID: ABC

auth: is-authorized (spanID 456)

SpanContext: Parent SpanID: 123
TraceID: ABC

api-server db: query-user (SpanID 789)

SpanContext: Parent SpanID: 456
TraceID: ABC

childOf

followsFrom

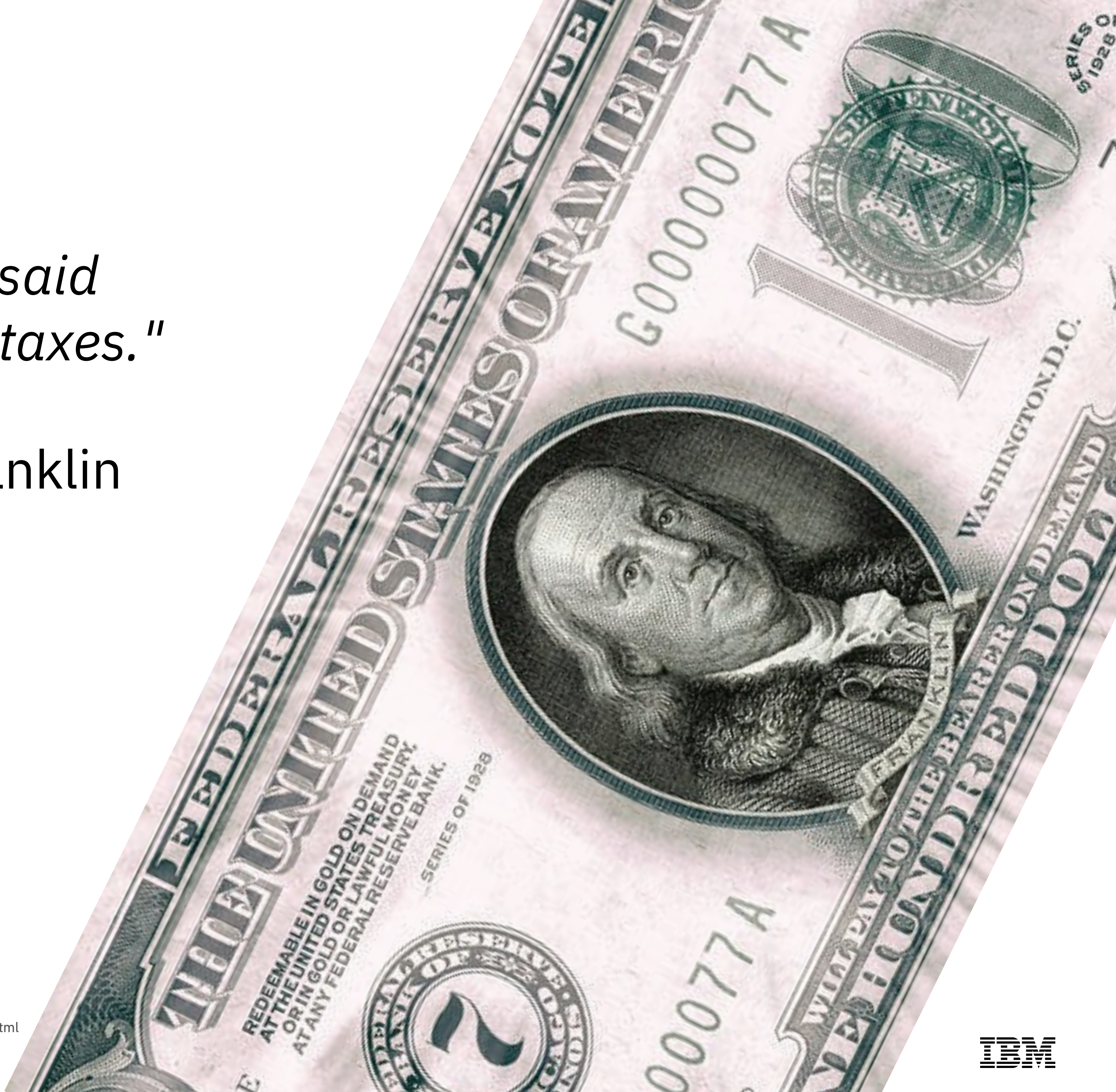
Get-connection (SpanID 890)

SpanContext: Parent SpanID: 789
TraceID: ABC

Microservices in Tax Office

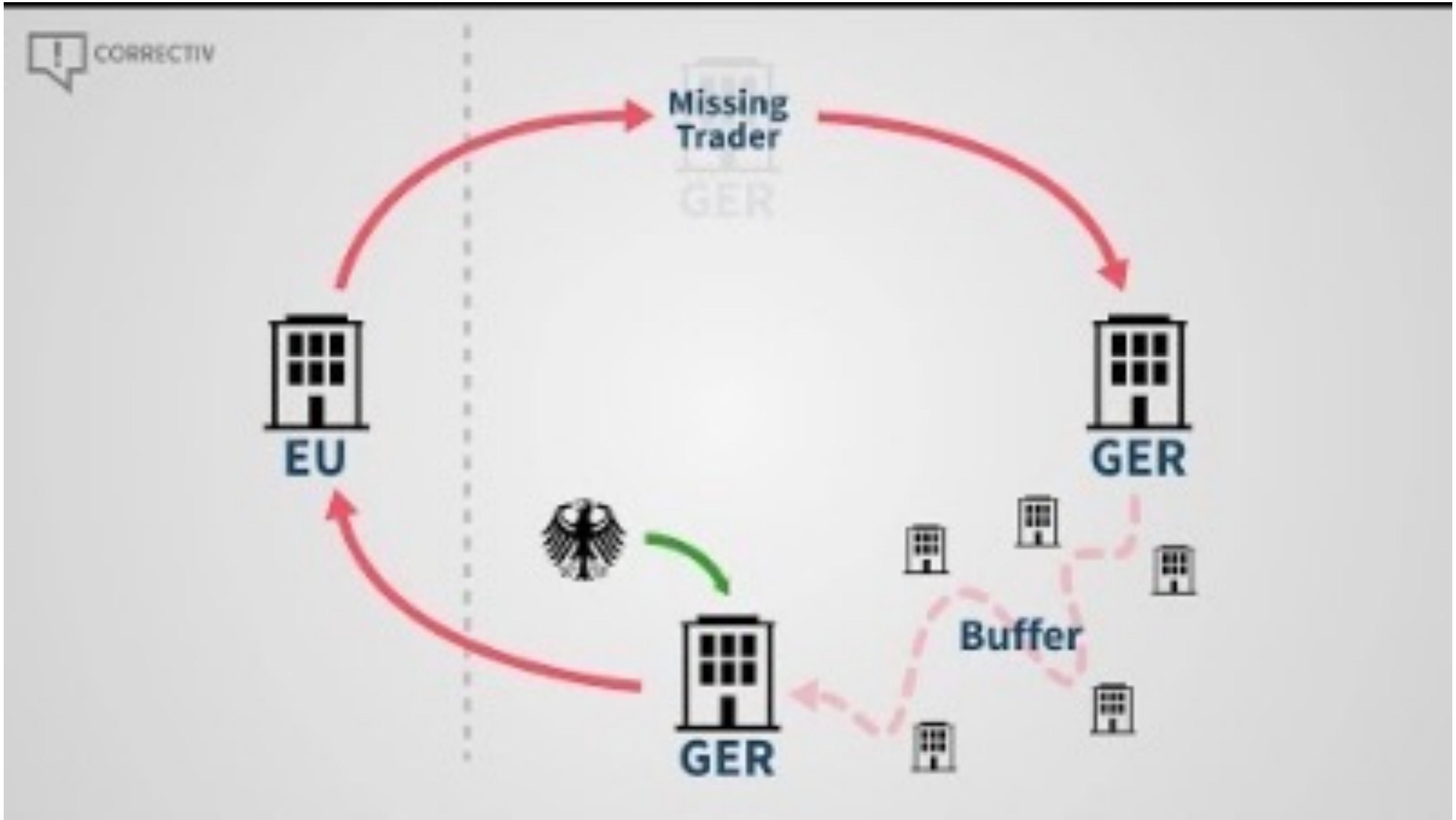
"... in this world nothing can be said to be certain, except death and taxes."

Benjamin Franklin


FEDERAL RESERVE NOTE
THE UNITED STATES OF AMERICA
G00000077A
WASHINGTON, D.C.
FRANKLIN
REDEEMABLE IN GOLD ON DEMAND AT THE UNITED STATES TREASURY, OR IN GOLD OR LAWFUL MONEY AT ANY FEDERAL RESERVE BANK.
SERIES OF 1928
WILL PAY TO THE BEARER ON DEMAND

https://pixels.com/featured/ben-franklin-1928-american-one-hundred-dollar-bill-currency-starburst-artwork-shawn-obrien.html

Missing Trader Fraud



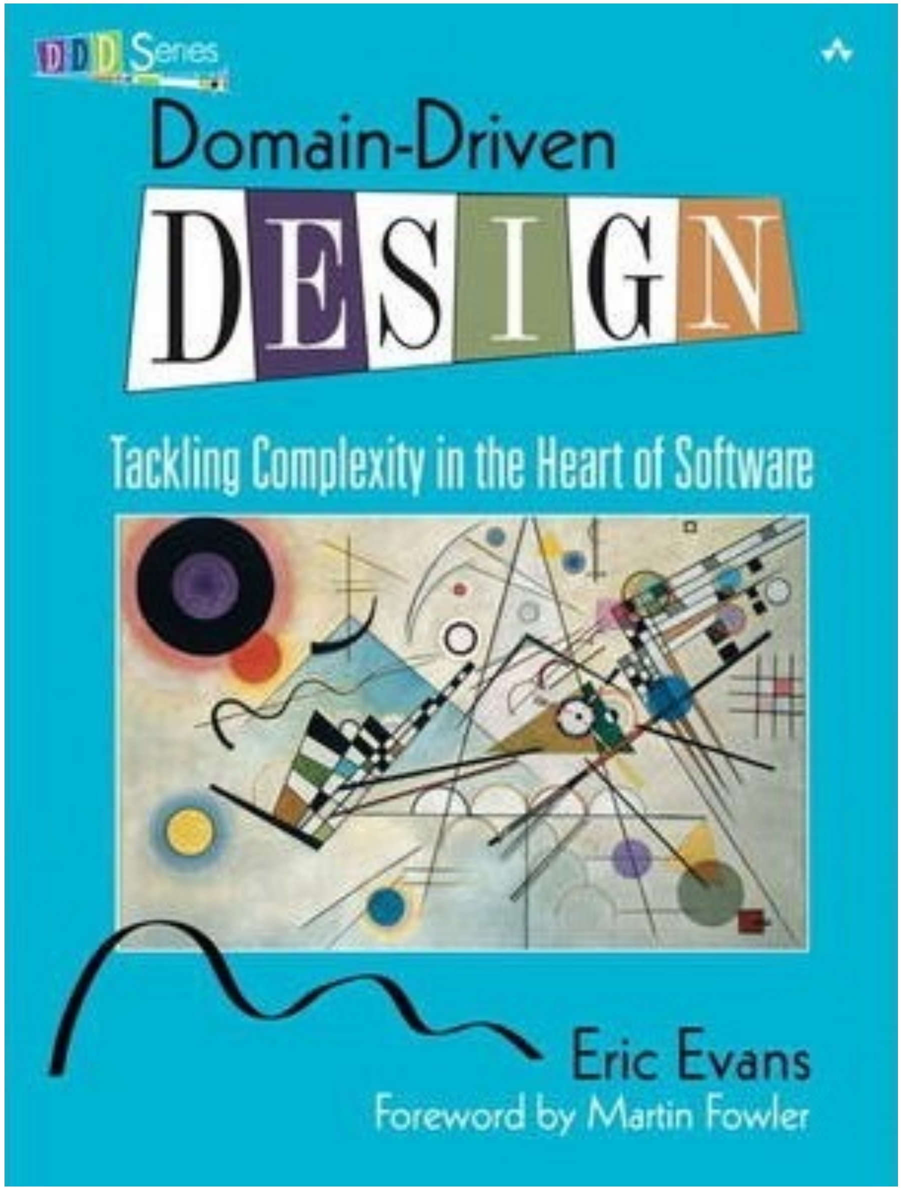
Source: https://www.youtube.com/watch?v=arjmpgA_VQA

Value Added Tax **I**nformation and **E**xchange **S**ystem

- Serves as a **data hub** between EU States, Financial Office and Tax Office
- Replacing existing (mainframe based = **Cobol**) system
- It handles company data (millions of active and historized data) and respective interface and dynamic data.
- Strict **auditing**/protocoling requirements
- circa **20 External systems** to communicate with
- **Complex domain**: In the time of signing the contract: 26 Business processes and 214 Use-cases (and growing)
- **70 developers** on the project (+ Test, Architecture, DevOps, PM)

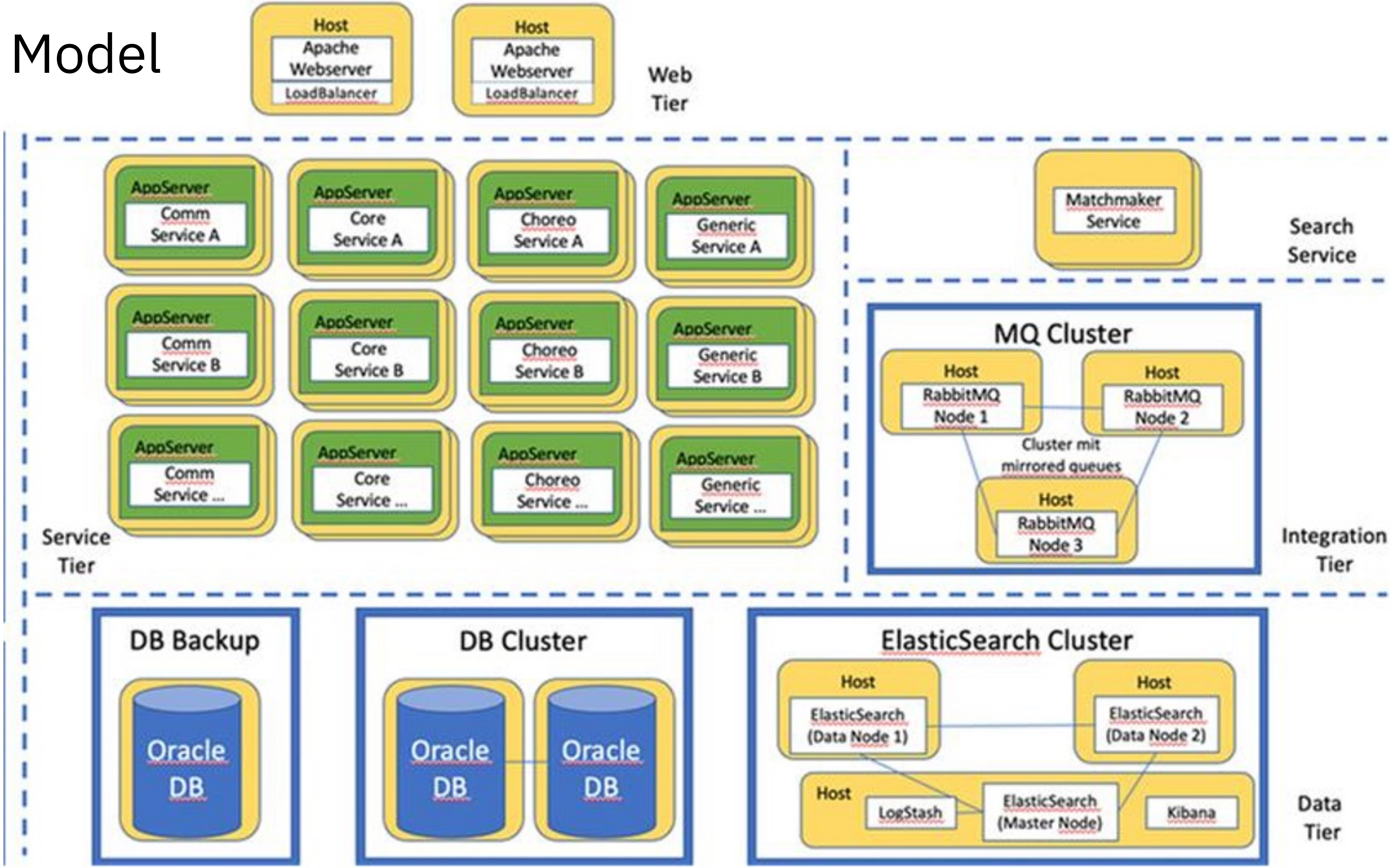
Technical realization

- **Microservice**-based architecture (DDD).
- Following the **Event Sourcing** and **CQRS** Patterns.
- **50+ microservices** and counting (of 3 types: GUI, business logic and communication).
- Tech Stack: Primefaces (JSF), JasperReports, **Spring Boot** (incl.: Spring Batch and Spring Cloud Config), Hibernate (JPA), **Oracle** and **RabbitMQ**.
- Running on JBoss EAP, built by **Maven.**
- DevOps: Terraform, Ansible, Jenkins.


DDD Series
Domain-Driven
DESIGN
Tackling Complexity in the Heart of Software
Eric Evans
Foreword by Martin Fowler

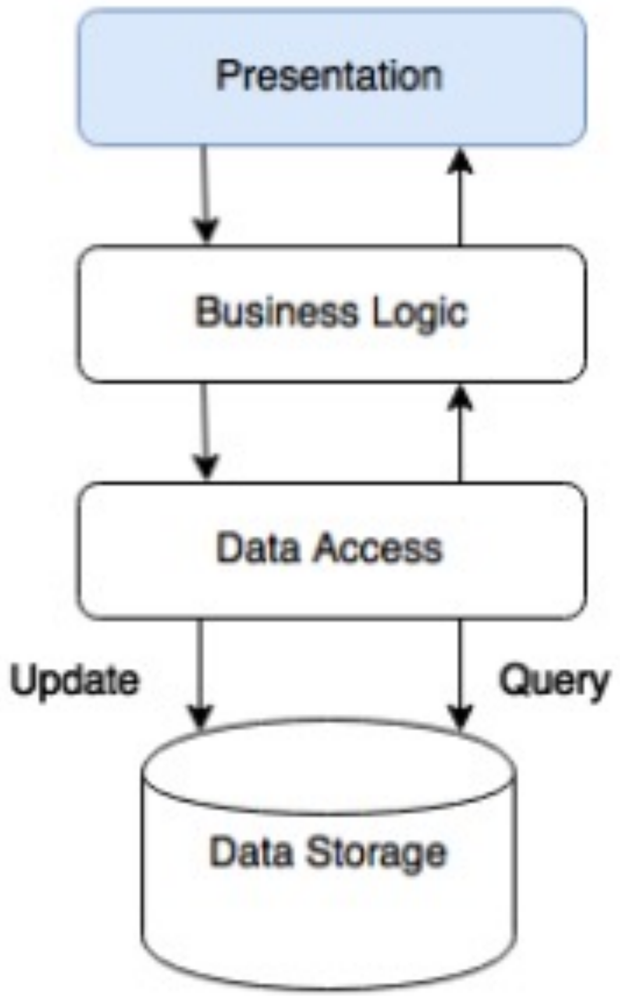
Source: https://www.bookdepository.com/Domain-Driven-Design-Eric-Evans/9780321125217

Operational Model

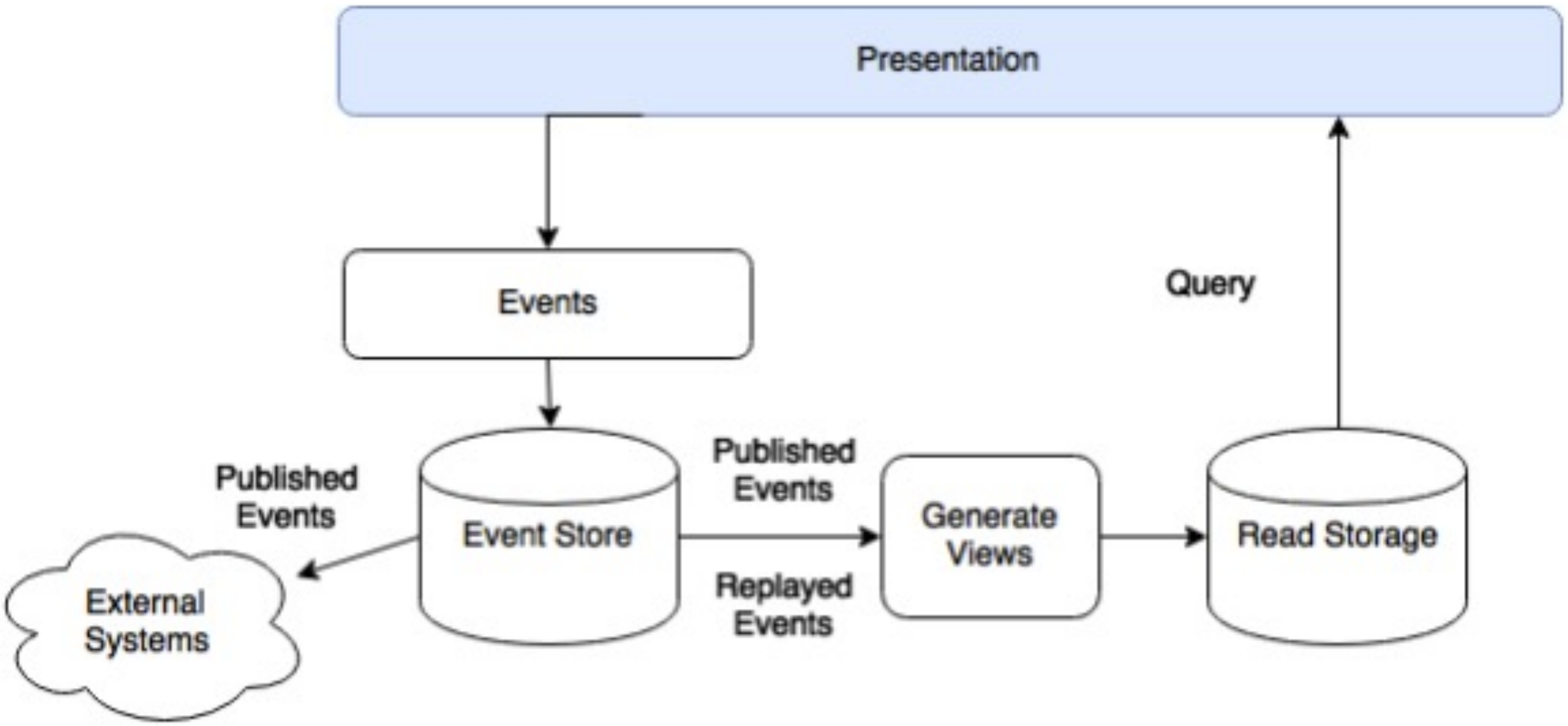


Traditional vs Event Sourcing and CQRS

Traditional



Event Sourcing with **C**ommand **Q**uery **R**esponsibility **S**egregation



Source: http://sensetecnic.com/cqrs-and-event-sourcing-for-the-iot/

Event store

- Events stored in the event store (Oracle CLOB) as JSON/diffs guarantee complete technical history (with timestamps)
- JSON Patch (RFC 6902)

The original document

{ "baz": "qux", "foo": "bar" }

The patch

[{ "op": "replace", "path": "/baz", "value": "boo" }, { "op": "add", "path": "/hello", "value": ["world"] }, { "op": "remove", "path": "/foo" }]

The result

{ "baz": "boo", "hello": ["world"] }

Source: http://jsonpatch.com/

Quiz

- 3x T-Shirt
- 5x IBM Cloud voucher

http://bit.ly/msshowcase
http://sli.do #248 673

More questions?

- Adam Seidel aseidel@sk.ibm.com
- Peter Butkovic peter.butkovic@sk.ibm.com

Source: https://contoh99.github.io/sedang/post/meme-any-questions/

IBM®