

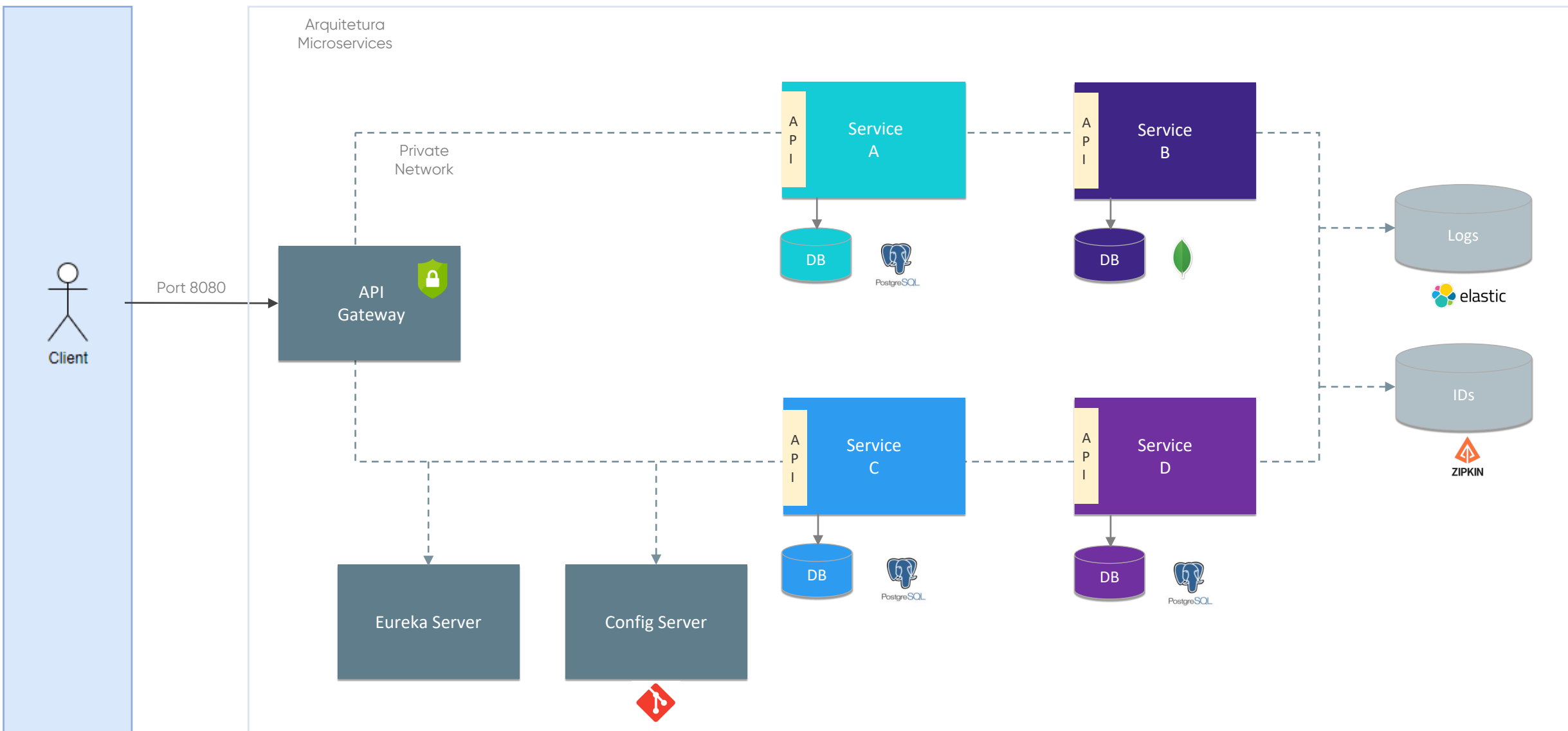
DECODEIR WEEK

Vídeo 2 - 15/03

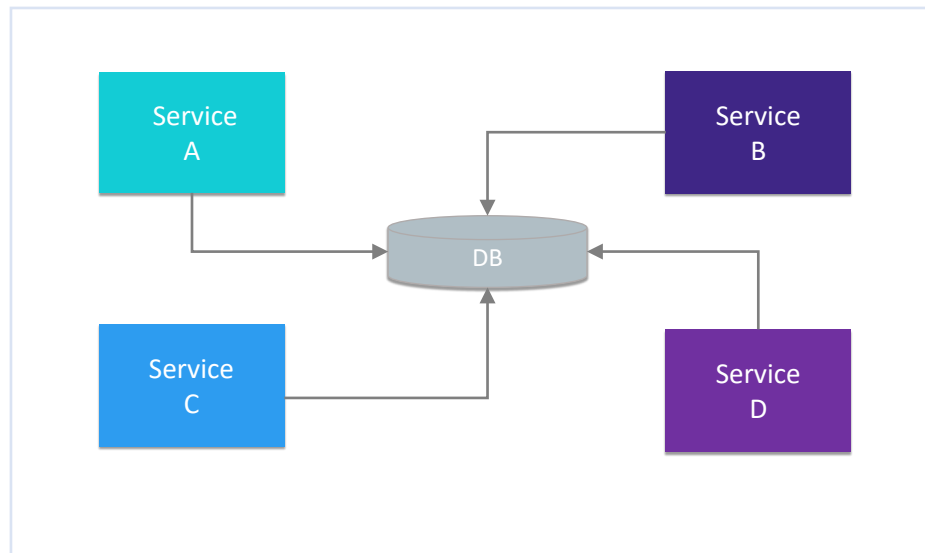
Comunicação entre Microservices



Arquitetura de Microservices – Abstração



Base de Dados Compartilhada

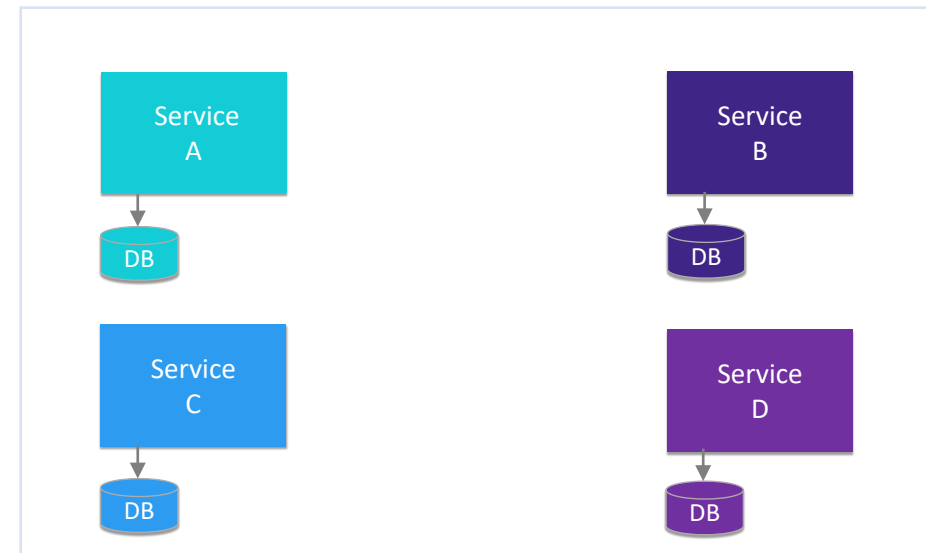


Na migração de Monolítico para Microservices o uso de base de dados compartilhada é comum no início.

Base de dados compartilhada garante forte consistência.

Base de dados compartilhada gera forte acoplamento, sem isolamento da modelagem de dados.

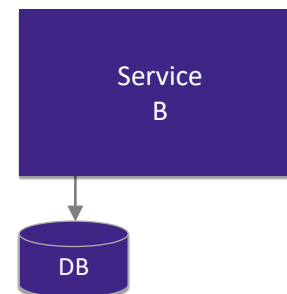
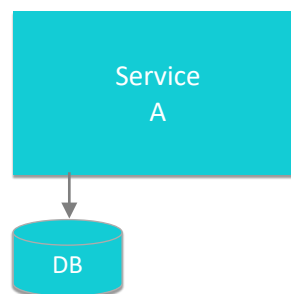
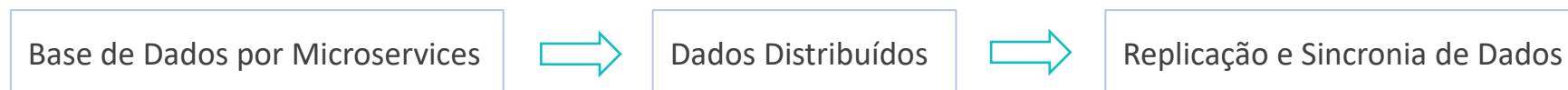
Base de Dados Distribuídas



Com base de dados por serviço é preciso lidar com a consistência eventual e a replicação de dados.

Base de dados por serviço geram menor acoplamento e isolamento da modelagem de dados.

Dados Distribuídos entre Microservices

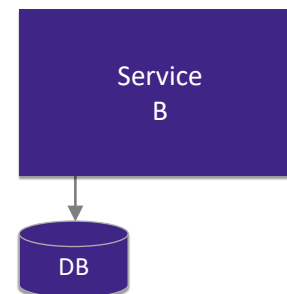
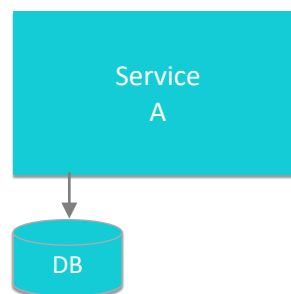
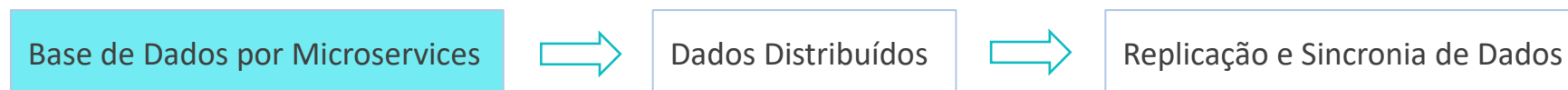


#decoderweek

@brito_michelli
youtube.com/michellibrito



Dados Distribuídos entre Microservices

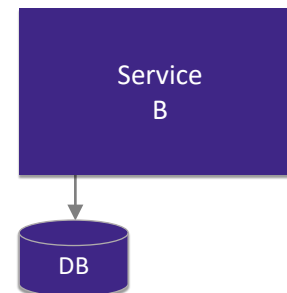
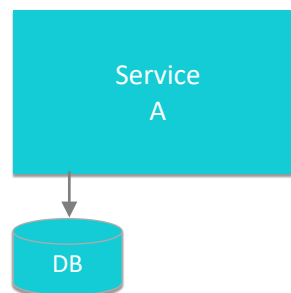
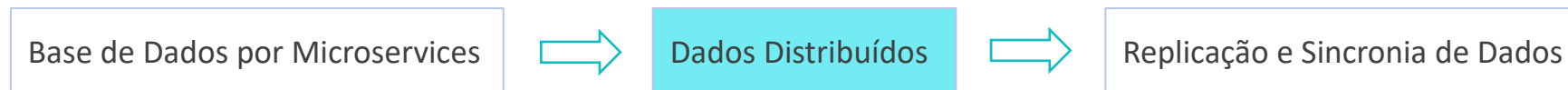


#decoderweek

@brito_michelli
youtube.com/michellibrito



Dados Distribuídos entre Microservices

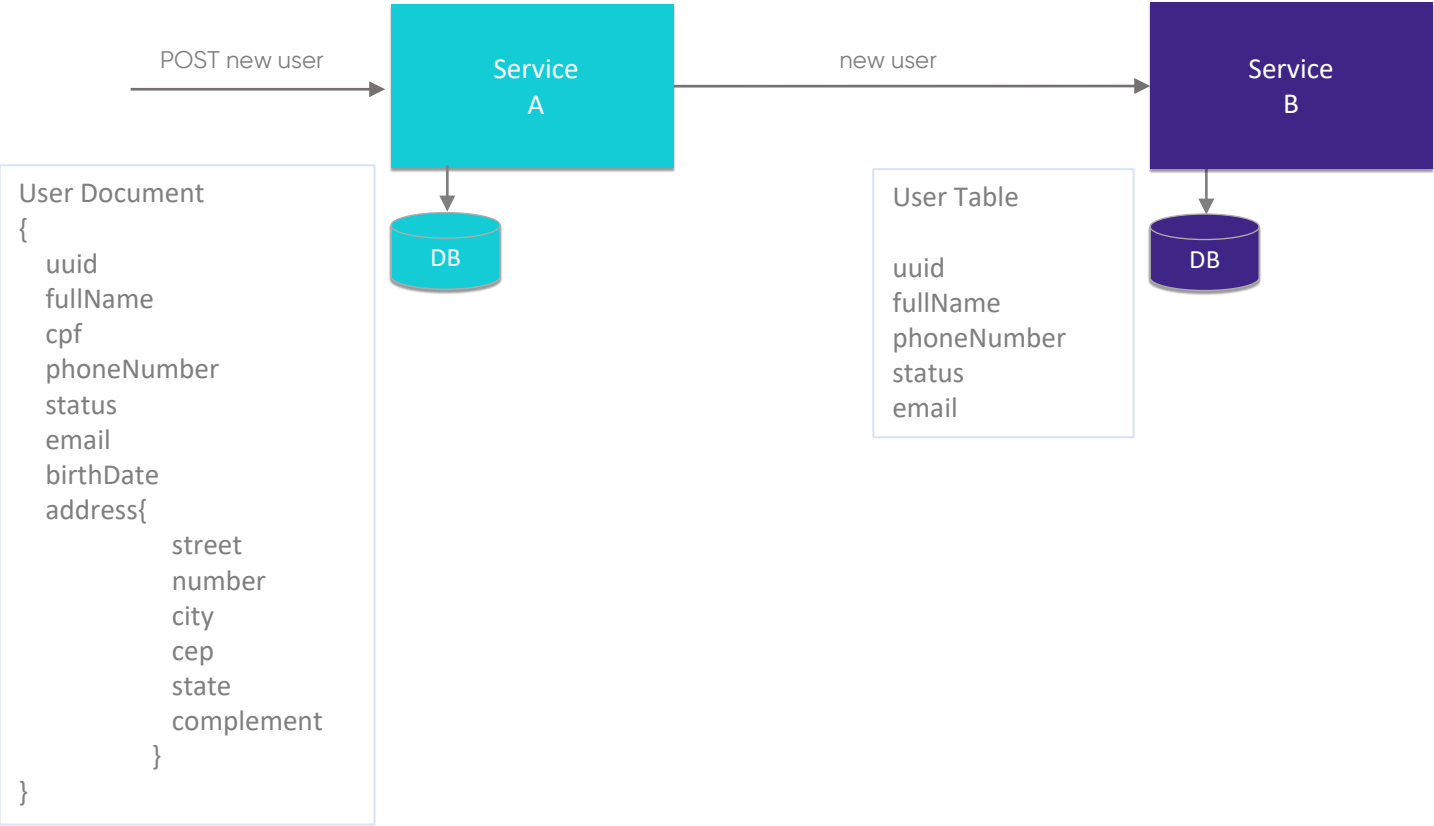


#decoderweek

@brito_michelli
youtube.com/michellibrito

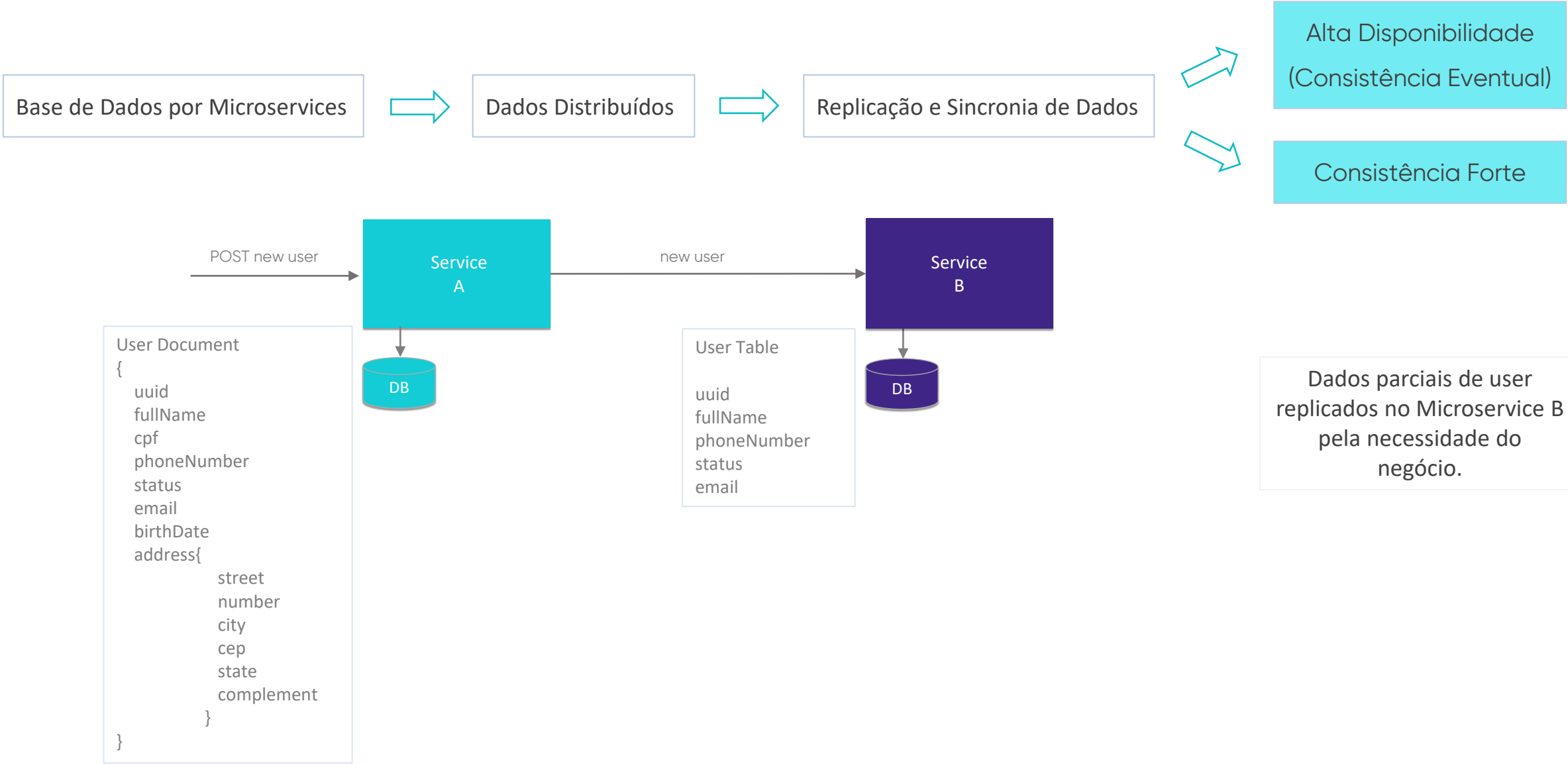


Dados Distribuídos entre Microservices

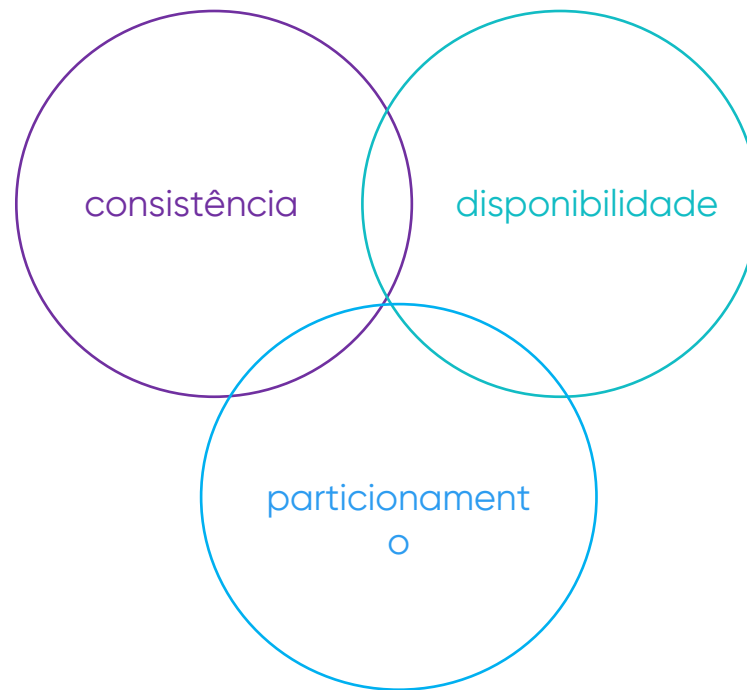


Dados parciais de user replicados no Microservice B pela necessidade do negócio.

Dados Distribuídos entre Microservices



Alta Disponibilidade vs Consistência Forte: Teorema CAP



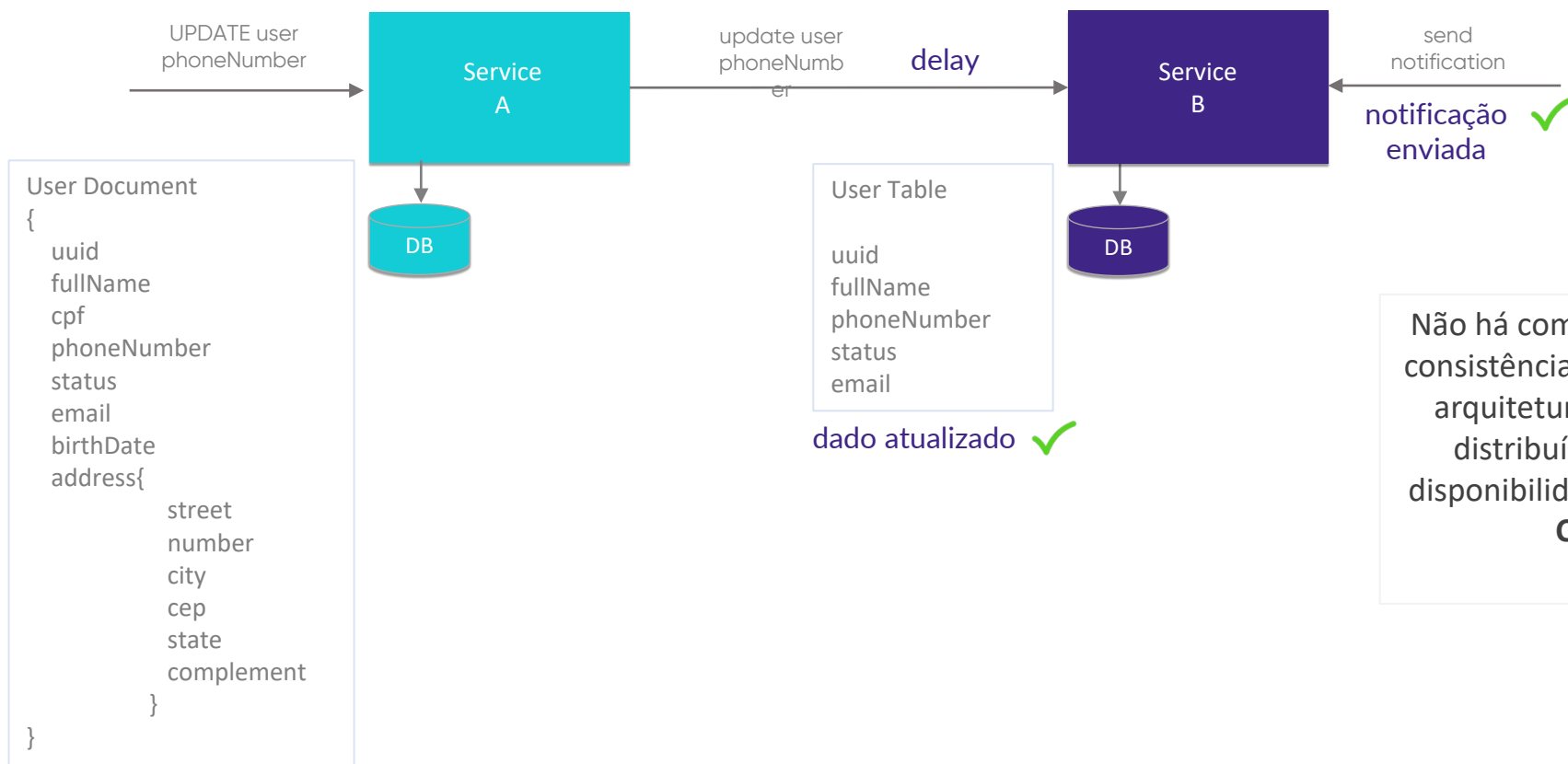
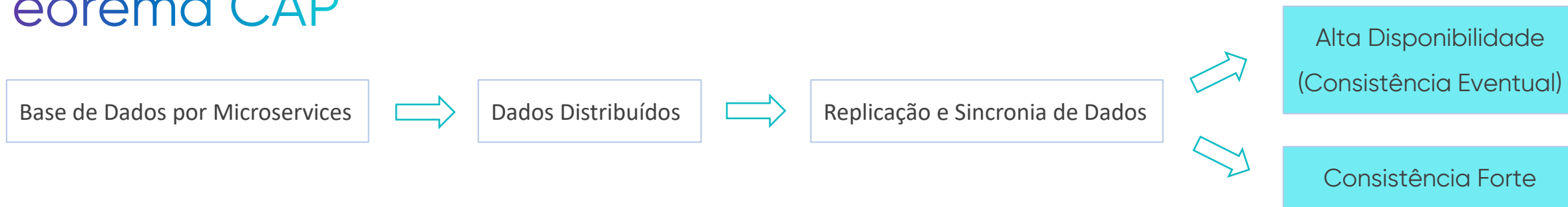
#decoderweek

@brito_michelli

youtube.com/michellibrito



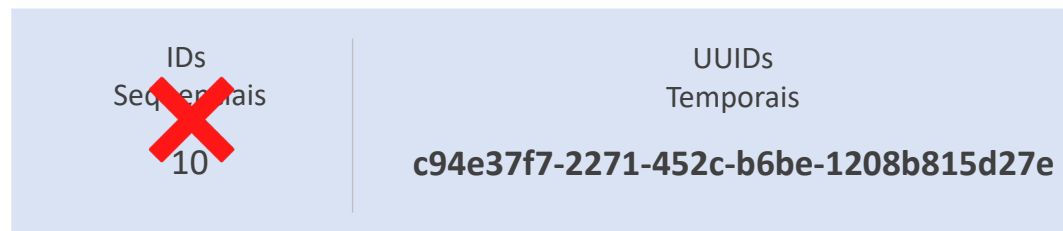
Alta Disponibilidade vs Consistência Forte: Teorema CAP



Não há como garantir alta disponibilidade e consistência forte ao mesmo tempo em uma arquitetura de Microservices com dados distribuídos, ou seja, em favor da alta disponibilidade consequentemente temos a **Consistência Eventual**.

dado atualizado ✓

UUIDs – Identificadores Distribuídos



IDs do tipo UUID são identificadores temporais universalmente exclusivos e essenciais para sincronia de dados distribuídos.

Podem ser gerados em qualquer lugar

Garantem maior manutibilidade

Facilitam a replicação de dados

Únicos em qualquer base de dados



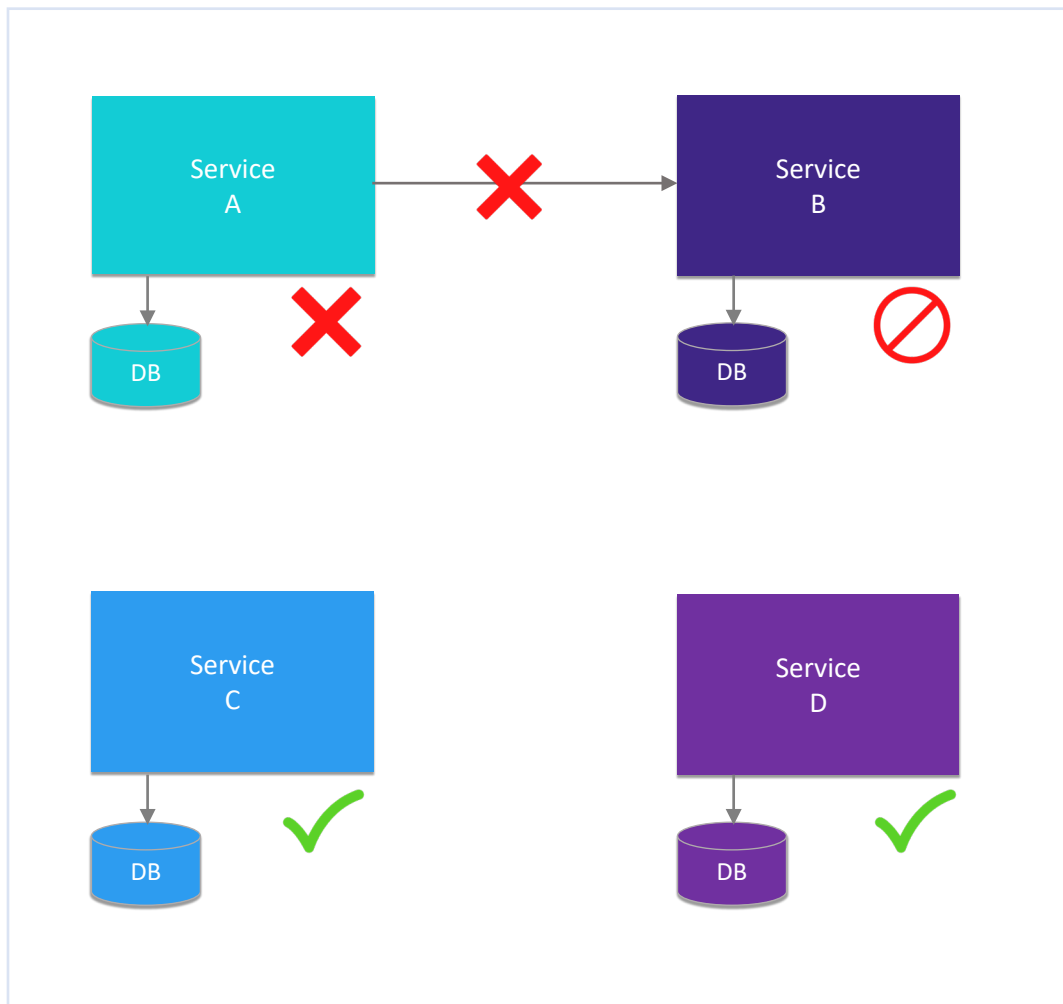
#decoderweek

@brito_michelli

youtube.com/michellibrito



Acoplamento em Microservices



Não existe desacoplamento absoluto entre Microservices.

O forte acoplamento nem sempre é um problema de modelagem arquitetural, mas sim pode ser uma necessidade do próprio negócio.

O entendimento do negócio é essencial na definição da arquitetura.

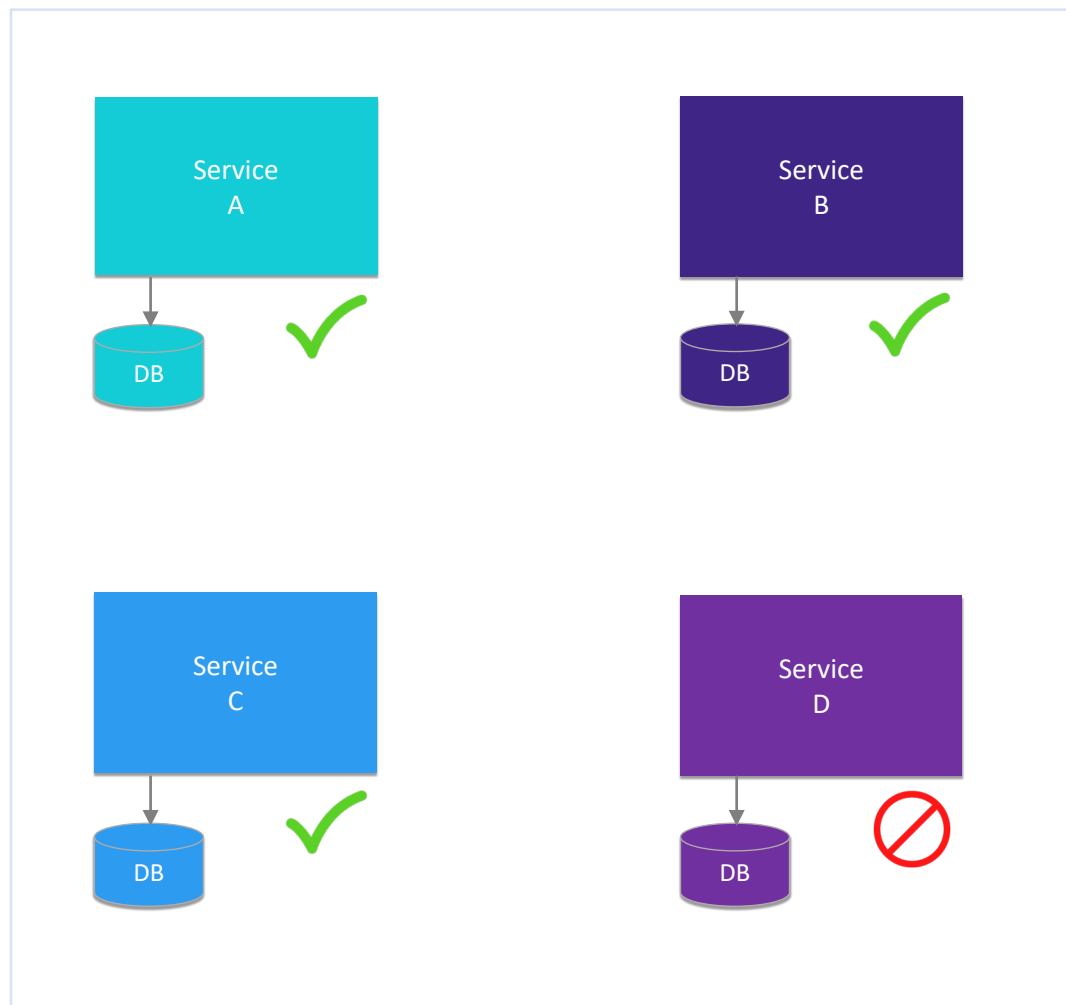
No exemplo, a necessidade de negócio definida obriga uma forte dependência entre A e B.



#decoderweek



Disponibilidade em Microservices



Não criamos Microservices para que qualquer um possa parar em algum momento sem afetar os demais, **mas sim para que alguns possam parar eventualmente e o sistema continuar disponível**. E isso já é muito melhor do que se nenhum pudesse parar.

A busca é sempre pela maior disponibilidade, mesmo não existindo na prática uma disponibilidade de 100%



#decoderweek



Comunicação entre Microservices **via APIs e Mensageria**



#decoderweek

@brito_michelli

youtube.com/michellibrito



Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

5 - Comunicação híbrida via API e Mensageria



#decoderweek



Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

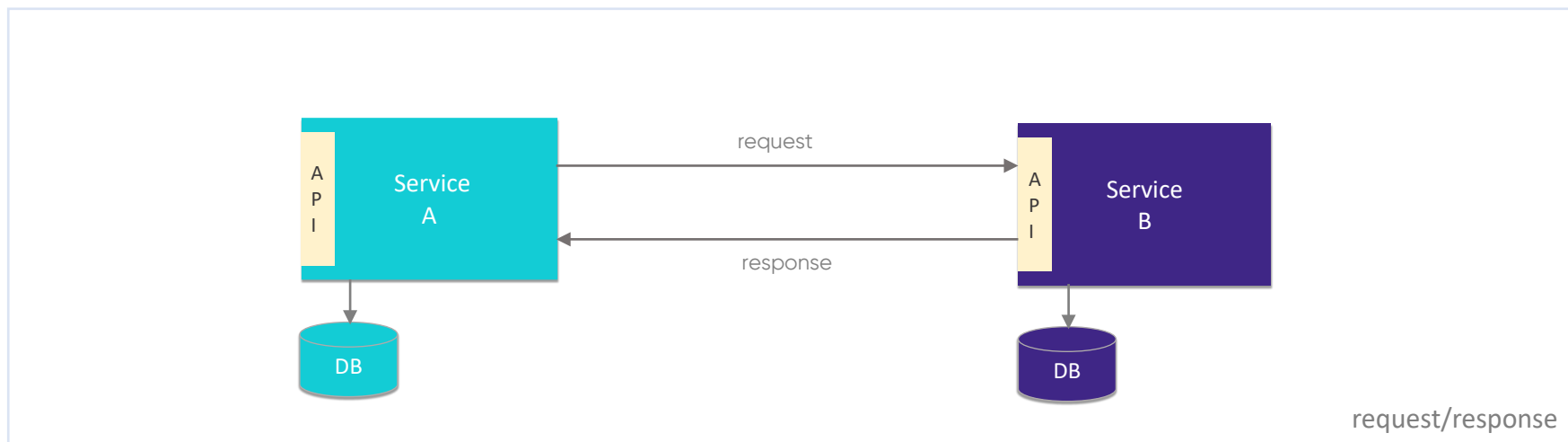
5 - Comunicação híbrida via API e Mensageria



#decoderweek



Comunicação Síncrona via APIs



Comunicação via métodos HTTP

Métodos GET, POST, PUT, DELETE



#decoderweek

@brito_michelli

youtube.com/michellibrito



Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

5 - Comunicação híbrida via API e Mensageria



#decoderweek



Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

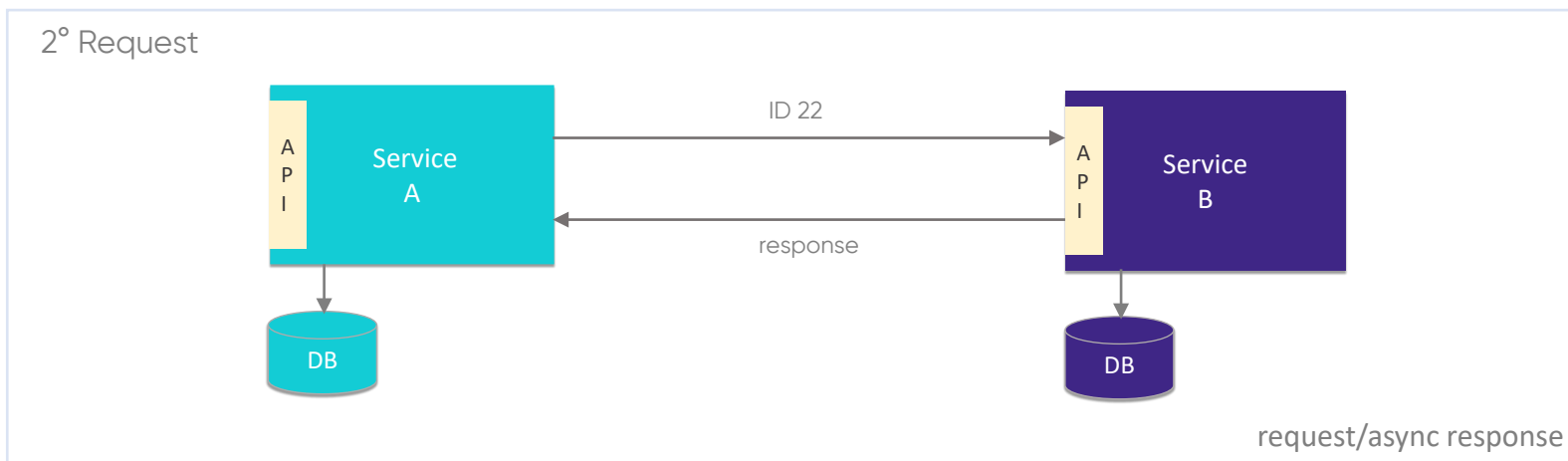
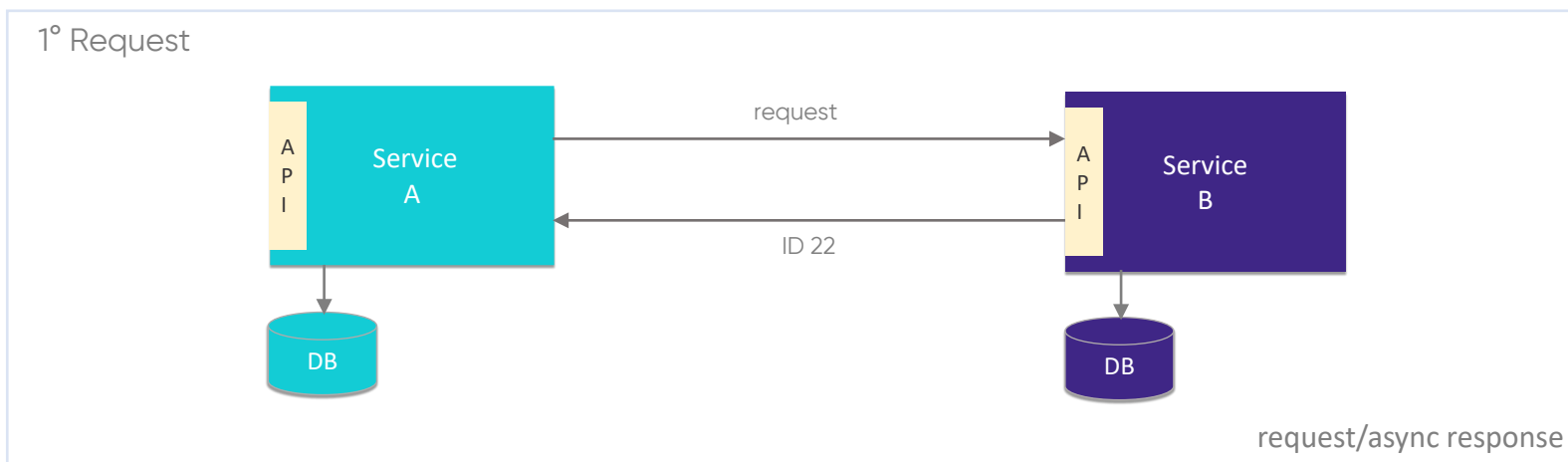
5 - Comunicação híbrida via API e Mensageria



#decoderweek



Comunicação Assíncrona via APIs



#decoderweek

@britto_michelli

youtube.com/michellibrito



APIs com Spring Web MVC

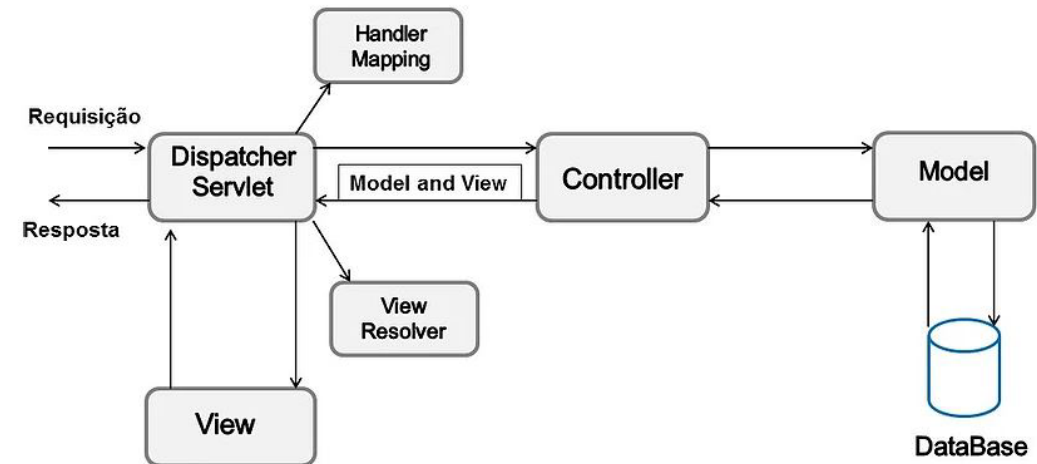


Conhecido como Spring MVC

Módulo fonte: spring-webmvc

Utiliza de um controlador frontal:
Dispatcher Servlet

Servidor Tomcat



```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
```



#decoderweek

@brito_michelli

youtube.com/michellibrito



APIs com Spring Web Webflux



Inserido na versão 5 do Spring Framework

Baseado no projeto Reactor

Criação de APIs Reativas

Totalmente não bloqueante

Servidor Netty

Tipos Flux e Mono

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-webflux</artifactId>  
</dependency>
```



#decoderweek

@brito_michelli

youtube.com/michellibrito



Resiliência com Spring Cloud Circuit Breaker Resilience4J



Disjuntor Resilience4j: Circuit Breaker

Estados: CLOSED, OPEN e HALF_OPEN

Métodos fallbacks e fluxos alternativos

Retry Resilience4j



```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-circuitbreaker-resilience4j</artifactId>
</dependency>
```



#decoderweek



Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

5 - Comunicação híbrida via API e Mensageria



#decoderweek



Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

5 - Comunicação híbrida via API e Mensageria

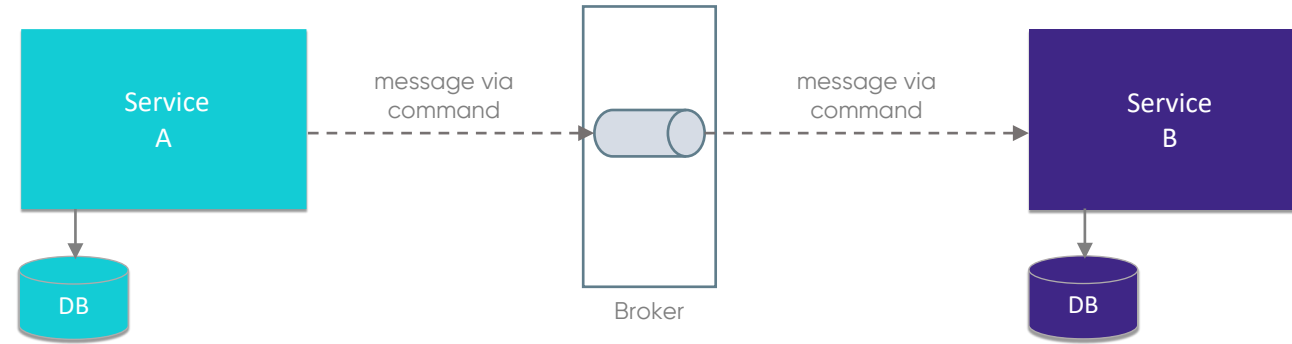


#decoderweek



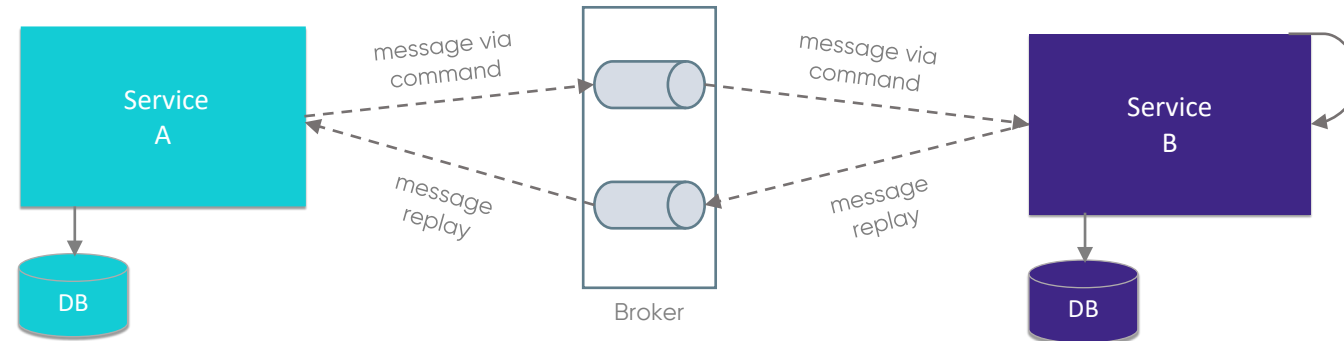
Comunicação Assíncrona via Mensageria - Comandos

1º case



one way

2º case



request/async response



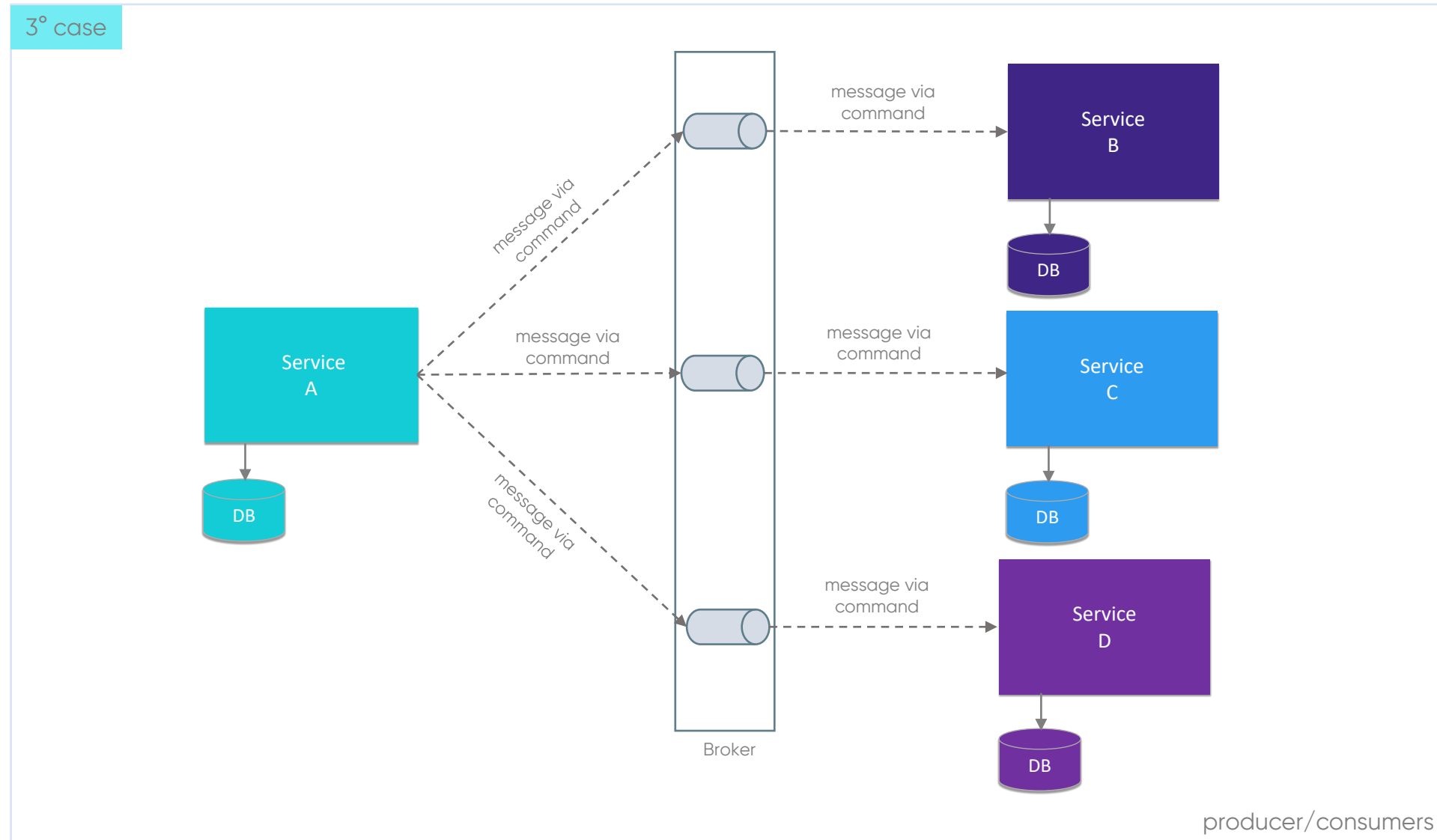
#decoderweek

@brito_michelli

youtube.com/michellibrito



Comunicação Assíncrona via Mensageria - Comandos



Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

5 - Comunicação híbrida via API e Mensageria



#decoderweek

Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

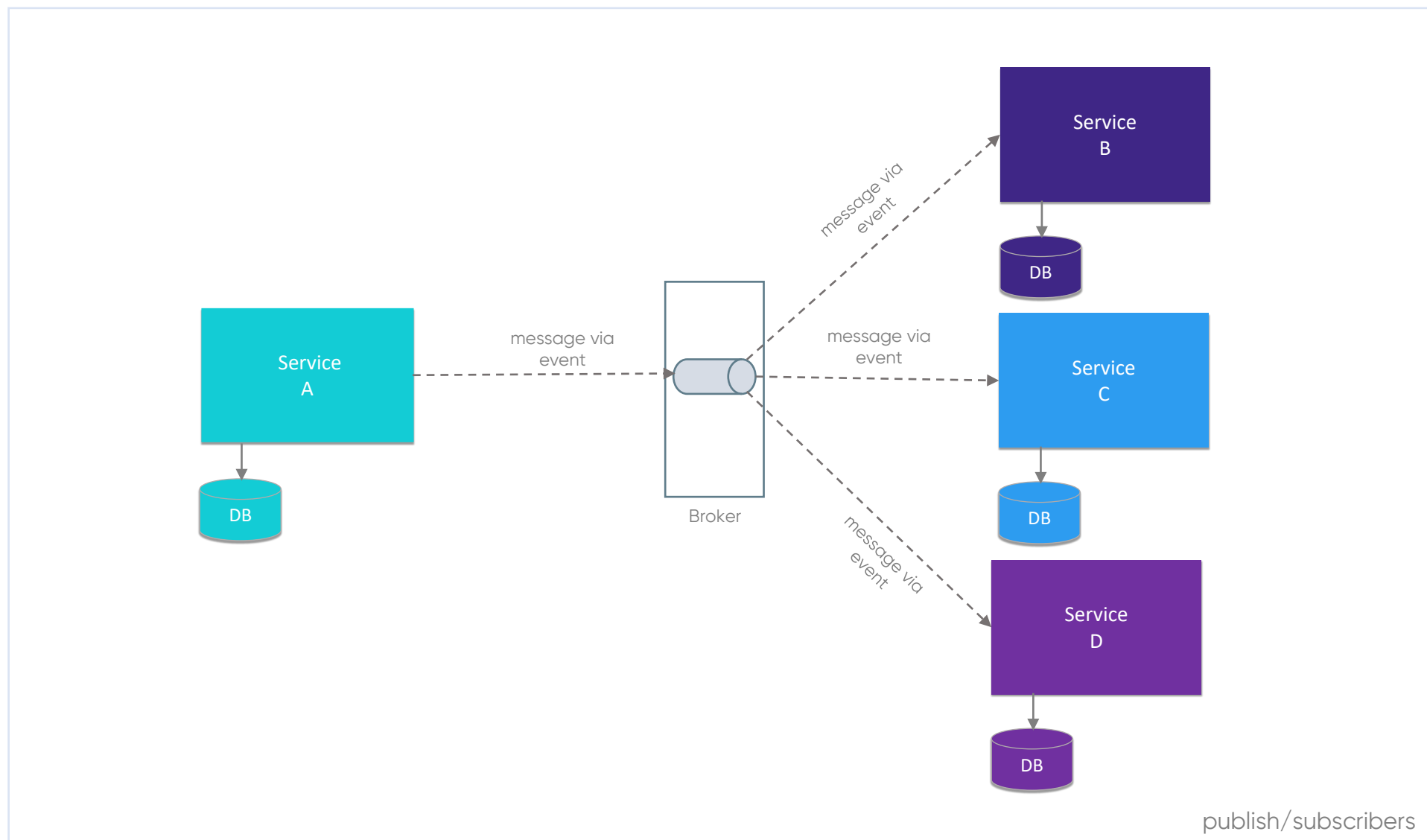
5 - Comunicação híbrida via API e Mensageria



#decoderweek



Comunicação Assíncrona via Mensageria - Eventos



Mensageria com Spring AMQP

Suporte a mensagens utilizando protocolo AMQP

Enviar e receber mensagens

Definir automaticamente queues, exchanges e routing keys

Projeto consiste em duas partes: base spring-amqp e spring-rabbit

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-amqp</artifactId>  
</dependency>
```



#decoderweek

@brito_michelli

youtube.com/michellibrito



Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

5 - Comunicação híbrida via API e Mensageria



#decoderweek

Comunicação entre Microservices via APIs e Mensageria

1 - Comunicação Síncrona via APIs

2 - Comunicação Assíncrona via APIs

3 - Comunicação Assíncrona via Mensageria - Comandos

4 - Comunicação Assíncrona via Mensageria - Eventos

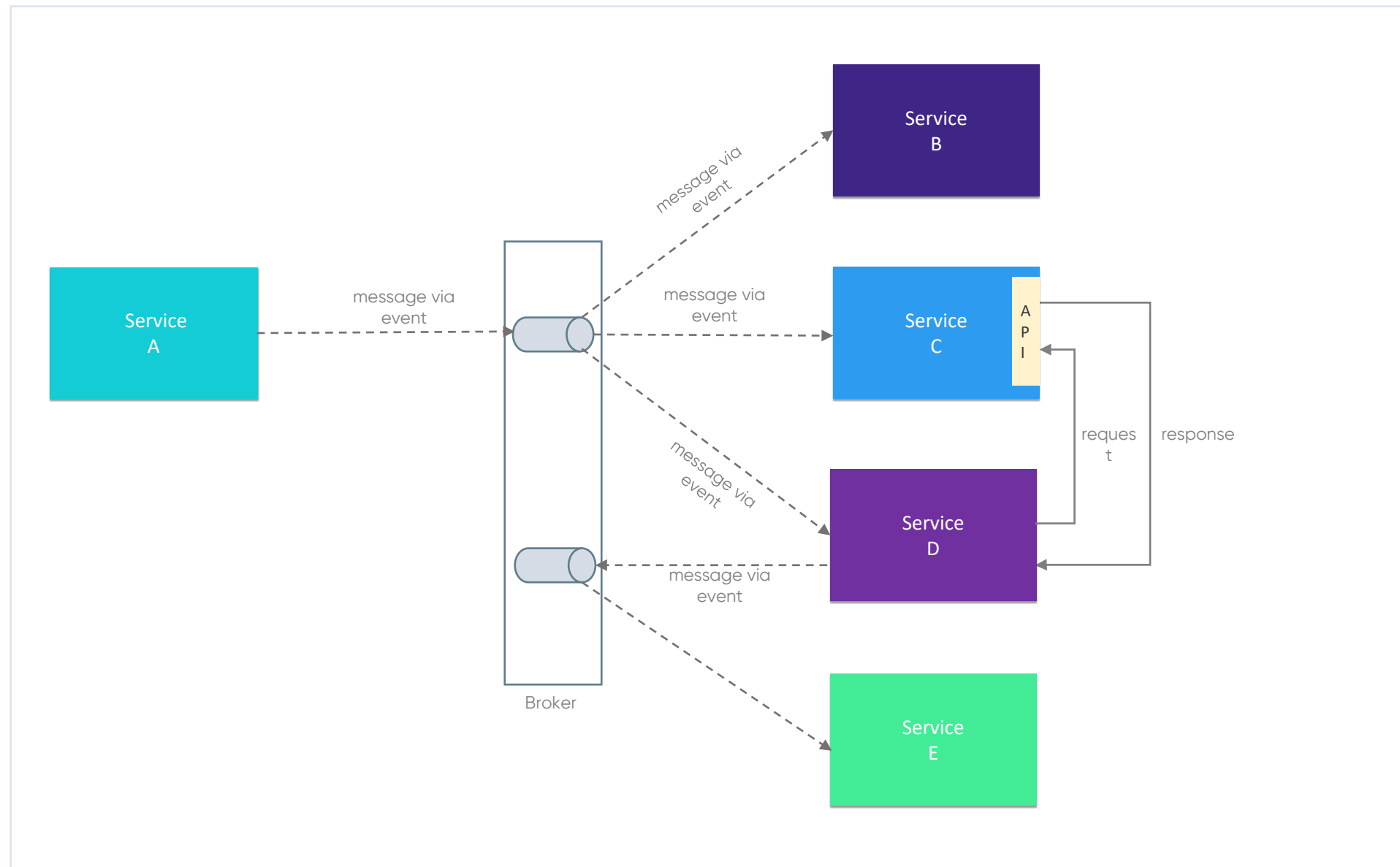
5 - Comunicação híbrida via API e Mensageria

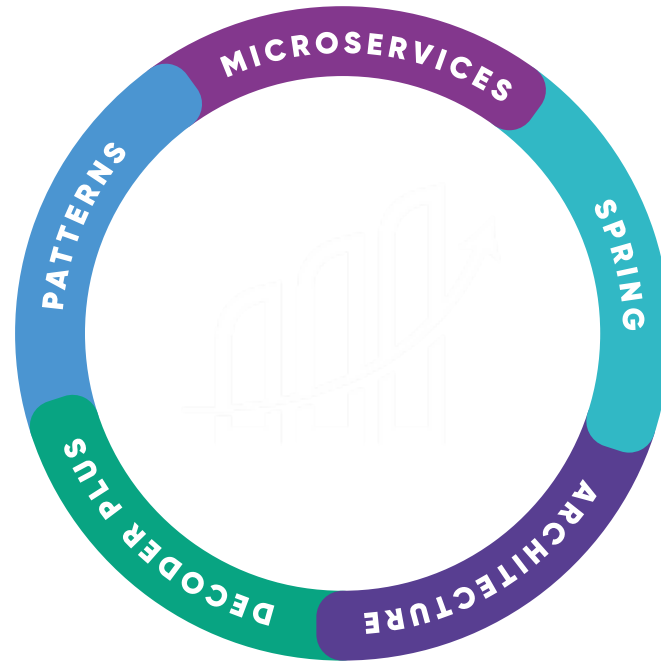


#decoderweek




Comunicação híbrida via API e Mensageria





#decoderweek

@brito_michelli 
youtube.com/michellibrito 