

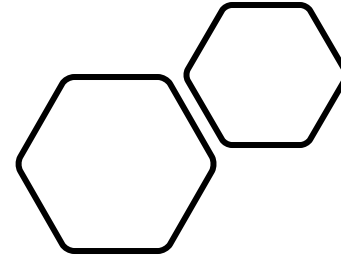
# Arquitetura de Micros Serviços (P2)

Prof. Bernardo Copstein

Prof. Júlio Machado



# Micros serviços: mecanismos de comunicação



Leituras recomendadas:

- Sommerville, Ian. Engineering Software Products – An Introduction to Modern Software Engineering. Pearson, 2019. Capítulo 6.
- Newman, Sam. Building Microservices. O'Reilly Media Inc, 2015.



# Comunicação de micros serviços

- Os micros serviços se comunicam trocando mensagens.
- Uma mensagem enviada entre serviços inclui algumas informações administrativas, uma solicitação de serviço e os dados necessários para entregar o serviço solicitado.
- Os serviços retornam uma resposta às mensagens de solicitação de serviço.
  - Um serviço de autenticação pode enviar uma mensagem para um serviço de login que inclui o nome inserido pelo usuário.
  - A resposta pode ser um token associado a um nome de usuário válido ou pode ser um erro informando que não há usuário registrado.



# Comunicação de micros serviços

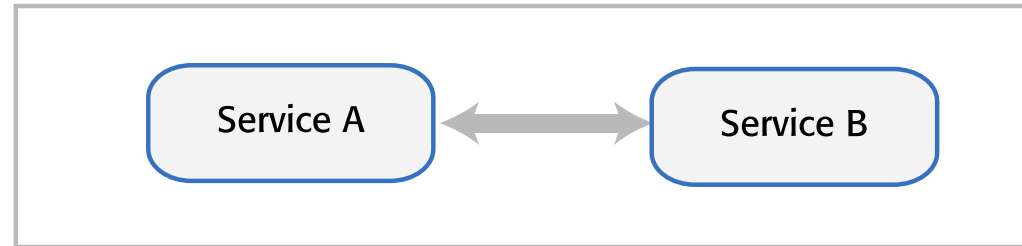
- Os serviços se comunicam trocando mensagens que incluem informações sobre o originador da mensagem, bem como os dados que são a entrada ou saída da solicitação.
- Ao projetar uma arquitetura de micros serviços, você precisa estabelecer um padrão de comunicação que todos os micros serviços devem seguir. Algumas das principais decisões que você precisa tomar são
  - A interação de serviço deve ser síncrona ou assíncrona?
  - Os serviços devem se comunicar diretamente ou por meio do middleware do agente de mensagens?
  - Qual protocolo deve ser usado para mensagens trocadas entre serviços?

# Comunicação direta e indireta

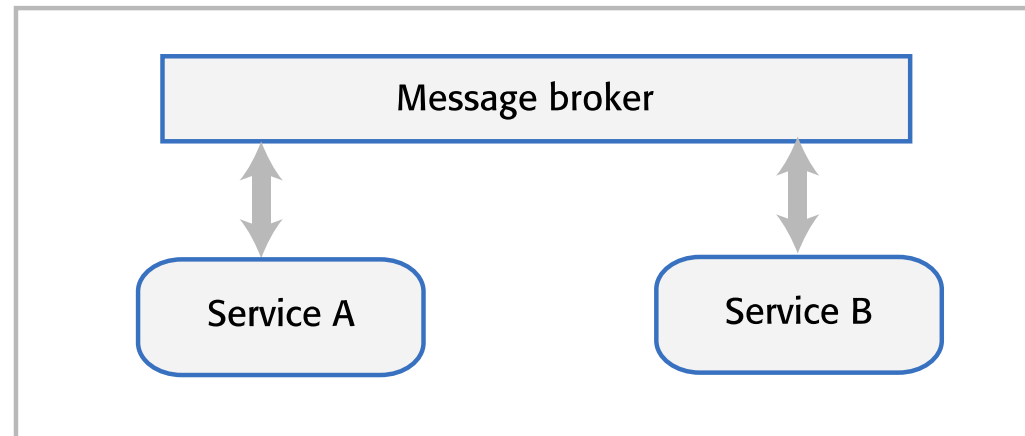
- A comunicação direta de serviço requer que os serviços que interagem conheçam o endereço um do outro.
- Os serviços interagem enviando solicitações diretamente para esses endereços.
- A comunicação indireta envolve nomear o serviço necessário e enviar essa solicitação para um agente de mensagens (às vezes chamado de barramento de mensagens).
- O agente de mensagens é então responsável por localizar o serviço que pode atender à solicitação de serviço.

# Comunicação de serviço direta e indireta

Direct communication - A and B send messages to each other

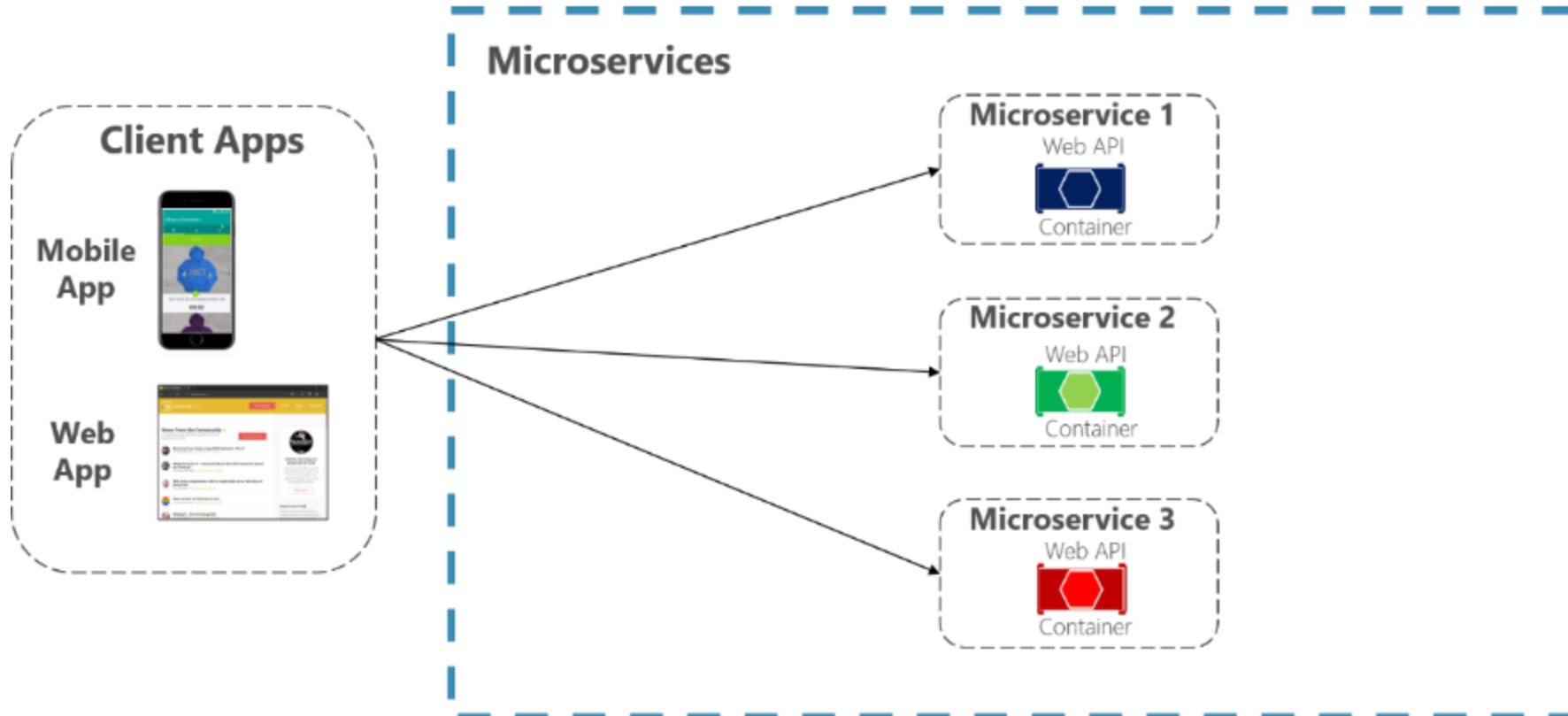


Indirect communication - A and B communicate through a message broker



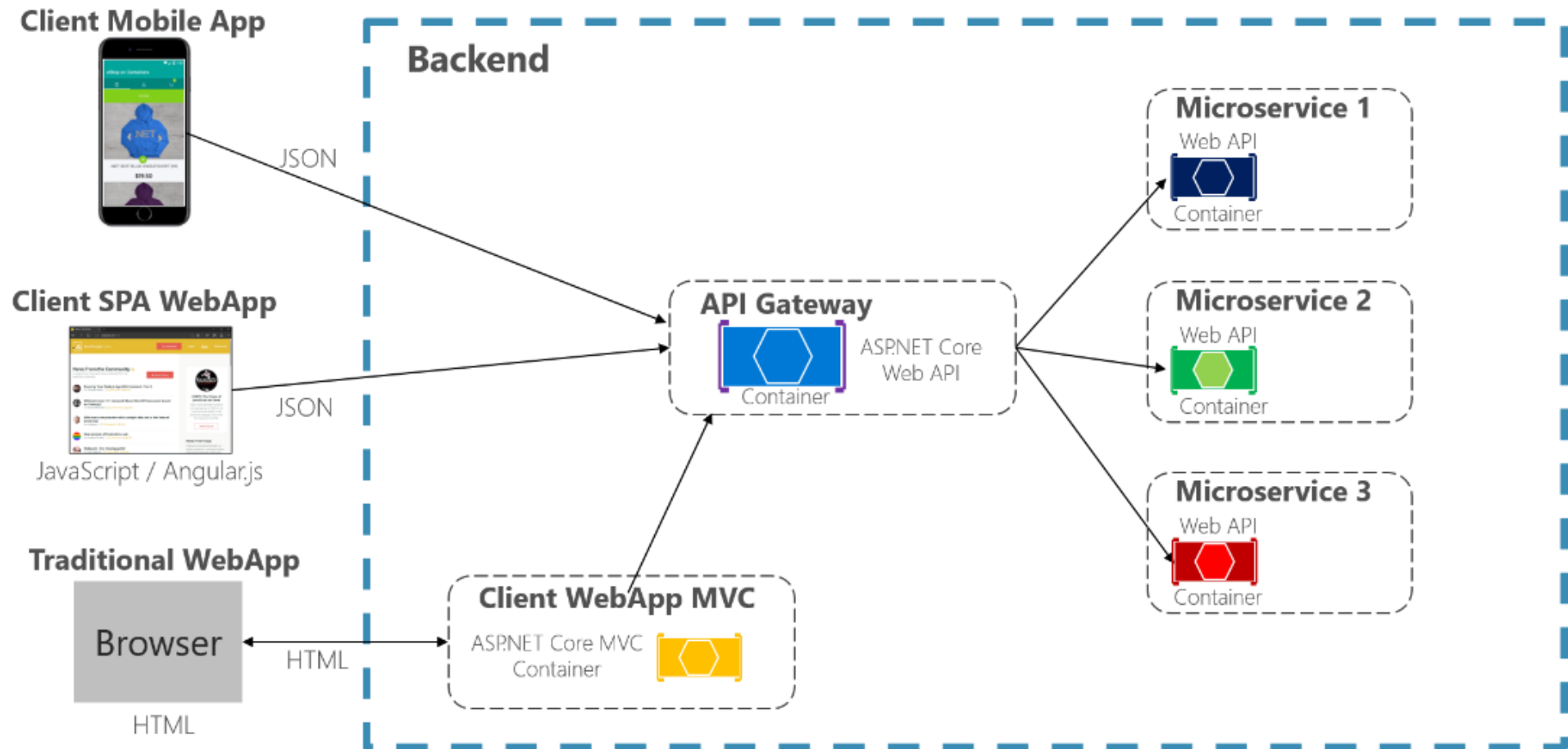
# Comunicação direta e indireta

## Direct Client-To-Microservice communication Architecture



# Comunicação direta e indireta

## Using the **API Gateway Service**



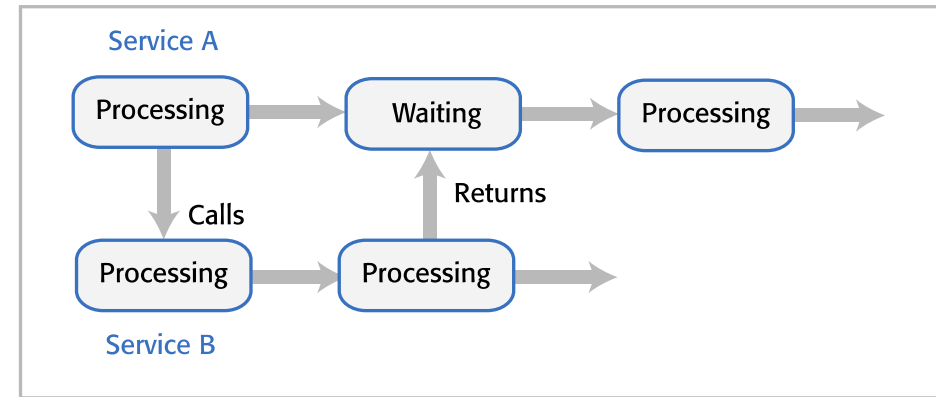


# Interação síncrona e assíncrona

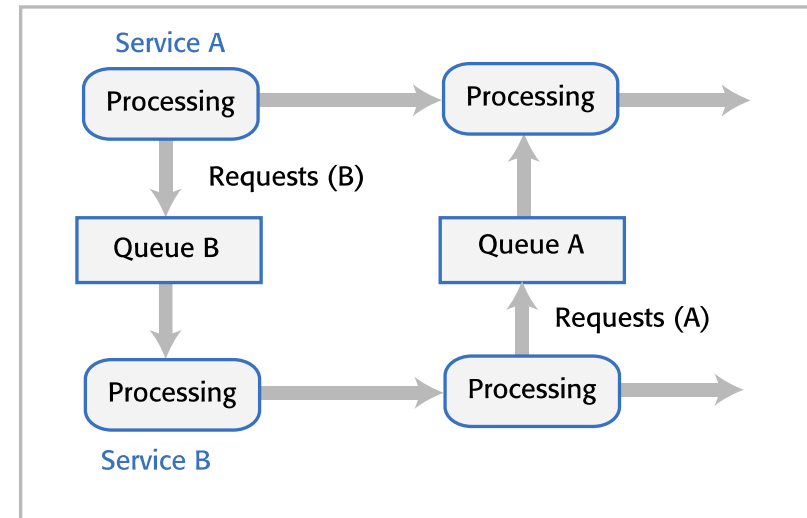
- Em uma interação síncrona, o serviço A emite uma solicitação ao serviço B. O serviço A então suspende o processamento enquanto B está processando a solicitação.
- Ele espera até que o serviço B retorne as informações necessárias antes de continuar a execução.
- Em uma interação assíncrona, o serviço A emite a solicitação que é enfileirada para processamento pelo serviço B. A então continua o processamento sem esperar que B termine seus cálculos.
- Algum tempo depois, o serviço B conclui a solicitação anterior do serviço A e enfileira o resultado a ser recuperado por A.
- O serviço A, portanto, precisa verificar sua fila periodicamente para ver se um resultado está disponível.

# Interação de micro serviços síncrona e assíncrona

Synchronous - A waits for B



Asynchronous - A and B execute concurrently



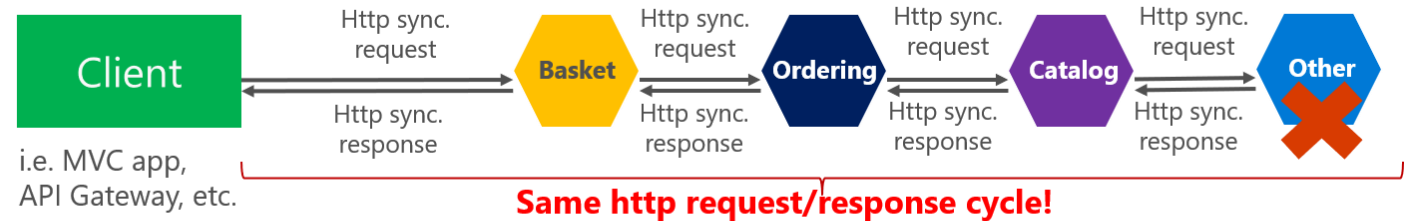
# Interação síncrona e assíncrona

## Synchronous vs. async communication across microservices

### Anti-pattern



**Synchronous**  
all req./resp. cycle



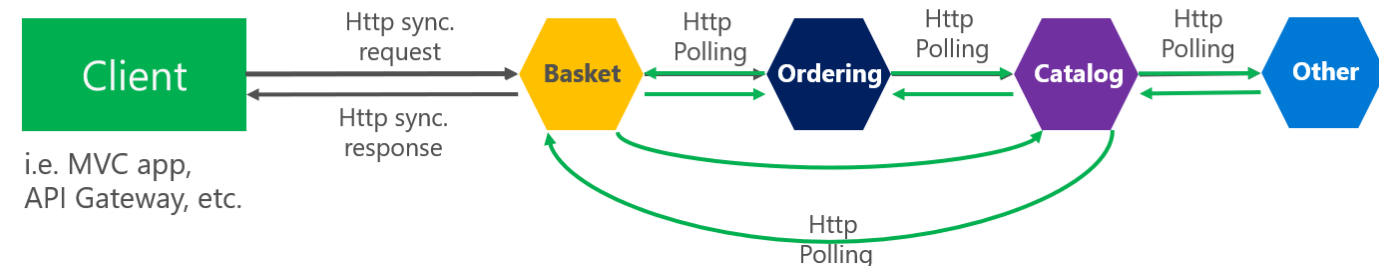
### Asynchronous

Comm. across  
internal microservices  
(EventBus: i.e. **AMPQ**)



### "Asynchronous"

Comm. across  
internal microservices  
(Polling: **Http**)



# Exercícios

- Discuta com seus colegas qual seria o melhor mecanismo para coordenar a comunicação entre os serviços no caso do exercício do restaurante visto anteriormente?



