

Avaliação 2

João Pedro Rodrigues Leite RA: A2487055

Data: 6 de setembro de 2024

**Universidade Tecnológica Federal do
Paraná**

Curso: Sistemas para Internet

Professor: Ivan Luiz Salvadori

Tópicos Avançados em Tecnologia da Informação

1. Explique as diferenças entre comunicação síncrona e assíncrona em serviços web, destacando os principais cenários de uso para cada abordagem. Descreva como cada tipo de comunicação afeta o desempenho, a escalabilidade e a experiência do usuário em sistemas distribuídos. Em seguida, forneça exemplos de tecnologias e padrões utilizados para implementar ambos os tipos de comunicação em aplicações web modernas.

Diferença entre Comunicação Síncrona e Assíncrona

Basicamente na comunicação **síncrona**, o cliente faz uma requisição e precisa aguardar a resposta antes de continuar a execução, já na **assíncrona**, o cliente envia a requisição e continua suas operações normalmente sem aguardar a resposta imediata.

Cenários de Uso

A comunicação síncrona é usada quando a resposta imediata é essencial para o programa, geralmente em sistemas de pagamento ou operações CRUD por exemplo. Já a comunicação assíncrona é ideal para cenários onde o processamento pode demorar, e o cliente não precisa de uma resposta imediata, como envio de e-mails.

Impacto no Desempenho e Escalabilidade

A comunicação síncrona pode se tornar um gargalo, em alguns casos não é recomendada, pois bloqueia o cliente até a resposta, afetando o desempenho em sistemas. Em contrapartida, a comunicação assíncrona melhora a escalabilidade, pois libera o cliente para continuar outras tarefas sem que ele precise aguardar uma resposta, mas pode impactar a experiência do usuário se não houver feedback imediato.

Tecnologias

Para comunicação síncrona, são comuns HTTP/HTTPS e RPC. Para assíncrona, pode-se utilizar Message Queues como o Apache Kafka, ActiveMQ e RabbitMQ

2. Descreva, com suas palavras, a arquitetura de microserviços, destacando suas principais características e benefícios em comparação com a arquitetura monolítica. Em seguida, discuta como a comunicação síncrona e assíncrona é utilizada entre microserviços, explicando as vantagens e desvantagens de cada abordagem. Por fim, exemplifique cenários em que uma comunicação assíncrona seria mais adequada que uma comunicação síncrona dentro de uma aplicação baseada em microserviços

Arquitetura de Microserviços

A arquitetura de microserviços é uma abordagem para desenvolver sistemas como um conjunto de pequenos serviços independentes. Cada serviço é responsável por uma função específica e pode ser desenvolvido, atualizado e escalado separadamente. A ideia é que a alteração em um microserviço não impacte em outro micro serviço, se isso ocorrer, os microserviços não foram bem construídos. Já na arquitetura monolítica tudo está em uma única fonte de código. As vantagens dessa arquitetura estão listadas abaixo:

- **Modularidade:** Cada microserviço faz uma coisa, mas faz bem feita, sempre buscando a perfeição. O que facilita a manutenção e a evolução.
- **Independência:** Mudanças em um serviço não devem afetar diretamente os outros, isso permite melhores atualizações.
- **Tecnologia Diversificada:** Você pode usar diferentes tecnologias para diferentes serviços, o que ajuda a escolher a melhor ferramenta para cada tarefa. Dependendo da função que o microserviço deve executar algumas tecnologias (Linguagem de programação, Banco de dados, etc) serão mais adequadas.
- **Resiliência:** Se um serviço falhar, o resto do sistema continua funcionando.
- **Escalabilidade:** É possível escalar apenas os serviços que precisam de mais recursos.
- **Facilidade de Substituição:** Microserviços pequenos são mais fáceis de substituir ou atualizar sem grandes impactos no sistema.

Comunicação entre Microserviços

Microserviços se comunicam de duas formas principais: síncrona e assíncrona.

Comunicação Síncrona

Na comunicação síncrona, um serviço faz uma solicitação e espera pela resposta antes de continuar o processo. É como uma conversa onde você espera a resposta antes de prosseguir.

Vantagens:

- **Simple de Implementar:** A comunicação é direta e fácil de entender.
- **Resposta Imediata:** Ideal para situações que exigem uma resposta rápida.

Desvantagens:

- **Dependência:** Se o serviço chamado estiver lento ou fora do ar, isso pode impactar o desempenho.
- **Latência:** Pode gerar atrasos se os serviços estiverem distantes, sobrecarregados ou mal construídos.

Comunicação Assíncrona

Na comunicação assíncrona, os serviços trocam mensagens sem esperar pela resposta imediata. É como enviar um e-mail e continuar o trabalho enquanto aguarda uma resposta.

Vantagens:

- **Desacoplamento:** Menos dependência direta entre serviços, o que melhora a flexibilidade.
- **Escalabilidade:** Melhora a performance em sistemas com alto volume de dados.

Desvantagens:

- **Complexidade:** Pode ser mais complicado de implementar e gerenciar.

- **Consistência Eventual:** Os dados podem não estar sincronizados imediatamente. Nesse caso, geralmente essa sincronização é feita a cada um certo tempo conforme foi explicado pelo professor no vídeo disponibilizado.

Quando Usar Comunicação Assíncrona

A comunicação assíncrona é útil em casos como:

- **Processamento de Dados em Lote:** Para tarefas que não exigem uma resposta instantânea, como processamento de pedidos, e-mails ou análise de logs.
- **Alta Disponibilidade:** Quando você precisa de um sistema resiliente, onde falhas em um serviço não interrompem o funcionamento geral.
- **Escalabilidade:** Em sistemas que precisam lidar com grandes volumes de eventos ou dados sem sobrecarregar o sistema.

3. Descreva o funcionamento das filas e exchanges descritos no protocolo AMQP, abordando como cada um desses componentes contribui para o processamento e roteamento de mensagens dentro de um sistema. Explique como as filas garantem a persistência e a entrega das mensagens. Além disso, analise os diferentes tipos de exchanges disponíveis no AMQP, como "direct", "topic" e fanout". Para cada tipo de exchange, discuta suas características específicas, como elas influenciam o roteamento das mensagens e quais são as suas vantagens e desvantagens em termos de flexibilidade e eficiência.

Funcionamento das Filas e Exchanges no AMQP

O AMQP (Advanced Message Queuing Protocol) é um protocolo de mensagens que facilita a comunicação entre sistemas usando um modelo de mensagens. No AMQP, as filas e exchanges desempenham papéis importantes no processamento e roteamento de mensagens.

Filas

As filas são onde as mensagens são armazenadas até serem processadas pelos consumidores. Elas garantem a persistência e a entrega das mensagens através de alguns mecanismos, sendo eles:

- **Persistência:** As mensagens podem ser configuradas para persistir no disco, garantindo que não sejam perdidas em caso de falhas do servidor. Isso é feito marcando a fila e as mensagens como persistentes.
- **Entrega Garantida:** As filas garantem que as mensagens sejam entregues aos consumidores. Se um consumidor falhar, a mensagem permanece na fila até ser processada com sucesso.

Exchanges

As exchanges são responsáveis pelo roteamento das mensagens para uma ou mais filas, com base em regras de roteamento. O AMQP oferece vários tipos de exchanges, cada uma com características diferentes:

- **Direct Exchange:** Roteia mensagens para filas baseando-se em uma chave de roteamento exata. A mensagem é entregue apenas às filas que possuem uma chave de roteamento que corresponde exatamente à chave fornecida.
 - **Vantagens:** Simples e direto, útil quando você precisa de um roteamento específico.
 - **Desvantagens:** Menos flexível para cenários que requerem múltiplos critérios de roteamento.
- **Topic Exchange:** Roteia mensagens com base em padrões de chave de roteamento. Permite um roteamento mais complexo e flexível, pois uma única exchange pode rotear mensagens para múltiplas filas com base em padrões de tópicos.
 - **Vantagens:** Alta flexibilidade e capacidade de roteamento sofisticada. Ideal para cenários onde mensagens precisam ser filtradas por vários critérios.
 - **Desvantagens:** Mais complexo de configurar e gerenciar comparado ao direct exchange.

- **Fanout Exchange:** Roteia mensagens para todas as filas vinculadas, sem considerar a chave de roteamento. É como um se fosse "broadcast", então a mensagem é enviada para todas as filas associadas à exchange.
 - **Vantagens:** Simples e eficiente quando você precisa que todas as filas recebam a mesma mensagem.
 - **Desvantagens:** Não permite filtragem de mensagens e pode resultar em sobrecarga se muitas filas estiverem associadas.

Conclusão

As filas garantem a persistência e a entrega das mensagens, enquanto as exchanges determinam como essas mensagens são roteadas para as filas. A escolha do tipo de exchange afeta diretamente como as mensagens são distribuídas e processadas, então a configuração depende da necessidade da aplicação, pois isso acabará influenciando a flexibilidade e eficiência do sistema de mensagens.