

Configuracoes do Kafka Producer

O Kafka Producer é responsável por enviar mensagens para um cluster Kafka. Abaixo estão as principais configurações e seus relacionamentos.

1. BOOTSTRAP_SERVERS_CONFIG

- Define a lista de brokers Kafka que o produtor pode usar para se conectar ao cluster.
- Exemplo de valor: "localhost:9092"
- Essencial para que o produtor saiba para onde enviar as mensagens.

2. DELIVERY_TIMEOUT_MS_CONFIG

- Tempo máximo (em ms) para que uma mensagem seja entregue antes de falhar.
- Relaciona-se ao REQUEST_TIMEOUT_MS_CONFIG, pois este define o tempo máximo para cada requisição individual ao broker.

3. ACKS_CONFIG

- Define o nível de confirmação do Kafka antes de considerar uma mensagem como entregue:
 - "0" -> Sem confirmação do broker (rápido, mas arriscado).
 - "1" -> Apenas o líder confirma.
 - "all" -> Todas as réplicas devem confirmar (mais seguro).
- Relaciona-se diretamente à garantia de entrega e pode impactar o desempenho.

4. LINGER_MS_CONFIG

- Tempo (em ms) que o produtor pode esperar antes de enviar mensagens, tentando agrupar múltiplas mensagens em um único lote.
- Um valor maior pode melhorar o desempenho, pois reduz o número de requisições ao broker.

5. REQUEST_TIMEOUT_MS_CONFIG

- Tempo máximo que o cliente espera por uma resposta do broker antes de considerar a requisição como falha.
- Relaciona-se ao DELIVERY_TIMEOUT_MS_CONFIG, pois este é um tempo global de entrega.

6. KEY_SERIALIZER_CLASS_CONFIG

- Especifica a serialização da chave das mensagens.

- No exemplo, `StringSerializer.class` indica que as chaves das mensagens são strings.

7. `VALUE_SERIALIZER_CLASS_CONFIG`

- Define a serialização do valor das mensagens.
- No exemplo, `JsonSerializer.class` indica que os valores das mensagens serão convertidos para JSON antes de serem enviados.

8. `ENABLE_IDEMPOTENCE_CONFIG`

- Ativa a idempotência, garantindo que uma mesma mensagem não seja processada mais de uma vez.
- Depende do `ACKS_CONFIG = "all"` e do `MAX_IN_FLIGHT_REQUESTS_PER_CONNECTION <= 5`.

9. `MAX_IN_FLIGHT_REQUESTS_PER_CONNECTION`

- Número máximo de requisições pendentes que o produtor pode ter por conexão.
- Se for maior que 1, mensagens podem ser entregues fora de ordem.

10. `TRANSACTIONAL_ID_CONFIG`

- Identificador da transação quando o produtor usa transações Kafka.
- Essencial para mensagens atômicas, garantindo que um conjunto de mensagens seja processado completamente ou não.

Relacionamentos Importantes:

- `DELIVERY_TIMEOUT_MS_CONFIG` e `REQUEST_TIMEOUT_MS_CONFIG` -> `DELIVERY_TIMEOUT_MS_CONFIG` define o tempo total de entrega, enquanto `REQUEST_TIMEOUT_MS_CONFIG` controla o tempo de resposta de cada requisição.
- `ACKS_CONFIG` e `ENABLE_IDEMPOTENCE_CONFIG` -> Para garantir idempotência, `ACKS_CONFIG` deve ser "all".
- `LINGER_MS_CONFIG` e `MAX_IN_FLIGHT_REQUESTS_PER_CONNECTION` -> Atrasar envios (`LINGER_MS_CONFIG`) pode ajudar a reduzir o número de requisições em paralelo.
- `TRANSACTIONAL_ID_CONFIG` e `ENABLE_IDEMPOTENCE_CONFIG` -> Para usar transações Kafka, a idempotência precisa estar ativada.

Com essas configurações, o Kafka Producer pode ser ajustado para entregar mensagens de maneira eficiente, segura e consistente.

