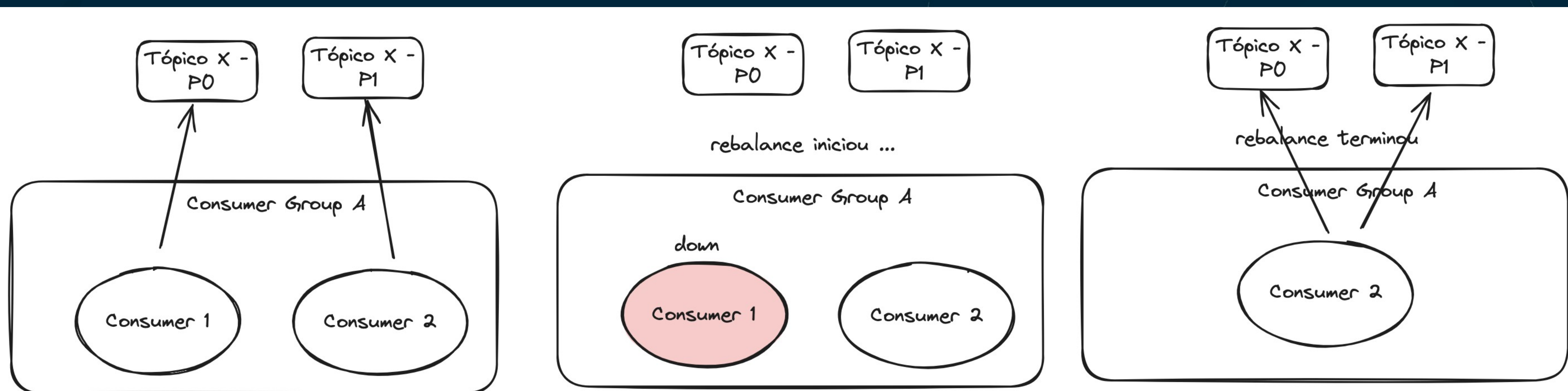


# Rebalance



# O que é Rebalance?

- Quando um consumidor entra/sai de um consumer group ocorre o que chamamos de Rebalance.
- O Rebalance é a tentativa de redistribuir as partições entre os consumidores, garantindo balanceamento.



## Outros triggers para o Rebalance ...

- Consumer ocioso por muito tempo, sem consumir mensagens.
- Novas partições adicionadas ao tópico.

## Quais os problemas do Rebalance ? Porque evitar ...

- Latência aumenta. Dependendo do tamanho do cluster pode levar minutos ou horas.
- Vazão diminui.
- Aumenta uso de recursos (CPU, Memória e Network).
- **Potencial duplicação e/ou perda de dados.**





## Dicas para evitar..

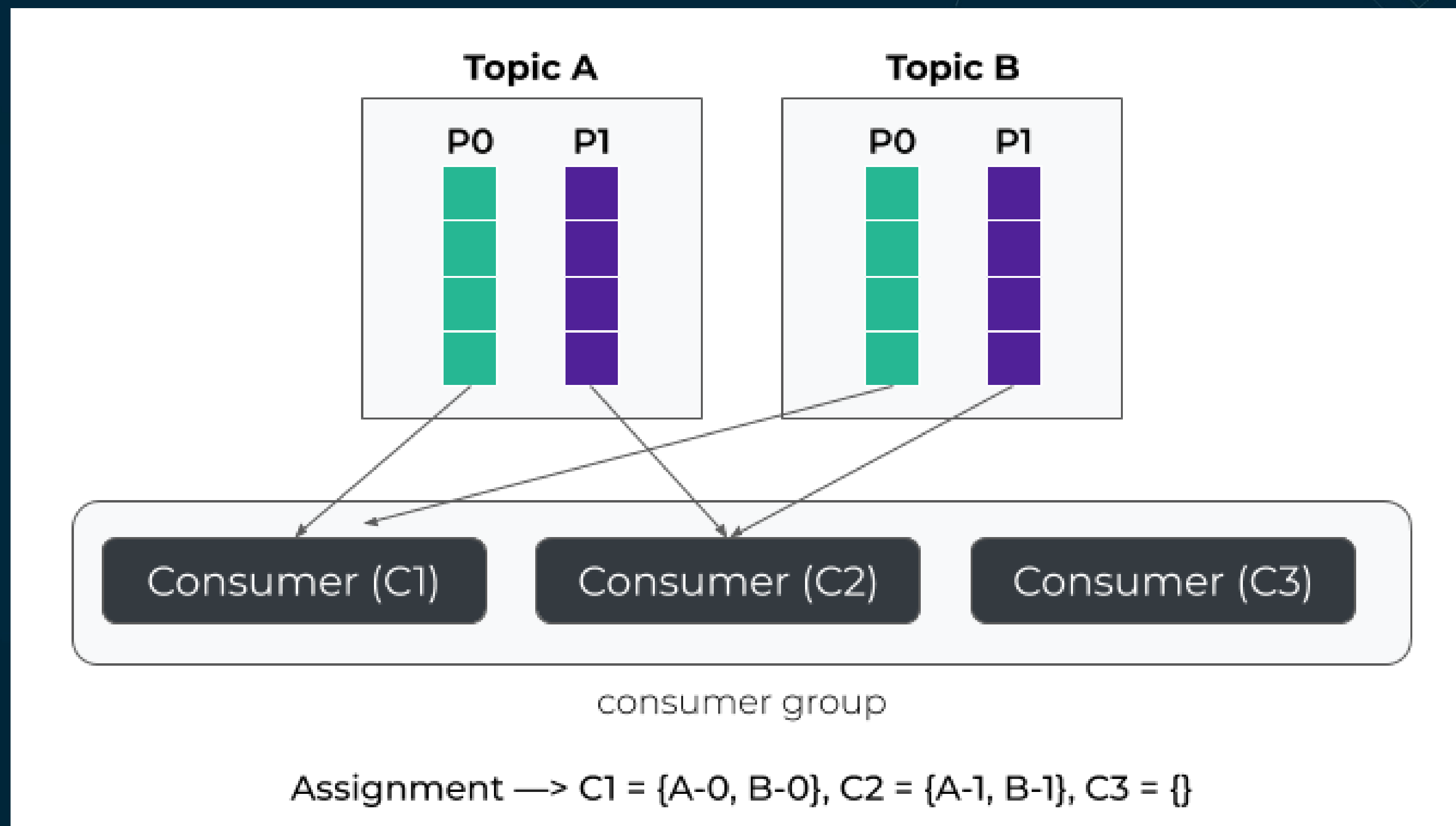
- Evite configurar Auto Scaling nas suas aplicações. Operações de Scale In e Out irão causar rebalance no consumer group. Se for realmente necessário, faça em momentos de pouco uso.
- Evite adicionar partições a tópicos existentes.
- Evite processamentos muito demorados, ao receber mensagens.
- Use estratégias de rebalance e particionamento corretas.

## Estratégias de Particionamento

- Algumas vezes o rebalance é inevitável, mas podemos reduzir o impacto, usando a estratégia de particionamento correta.
- Particionamento ocorre quando uma partição precisa ser assinada a um consumidor.
- Quando um consumidor entra/sai do Consumer Group, usualmente as partições são redistribuídas entre todos, causando o “stop-the-world”.
- As estratégias existentes são: Round Robin, Sticky, Range.

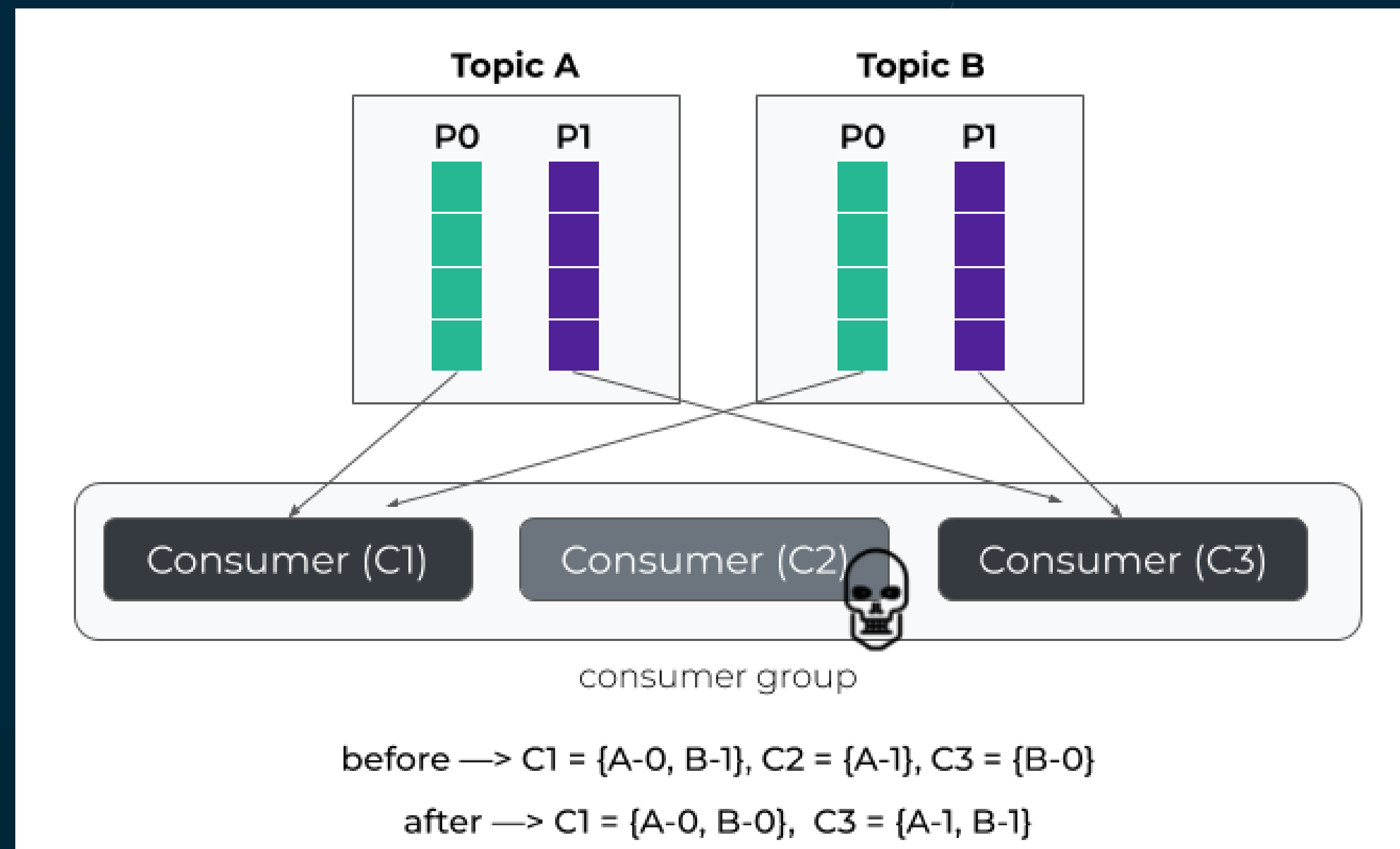
# Estratégias de Particionamento: Range

- Consumidores são colocados em ordem lexicográfica.
- Partições são ordenadas de forma numérica. Então temos: A0,B0,A1,B1 e consumidores: C1,C2,C3.
- Casos de uso: colocation.



## Estratégias de Particionamento: Round Robin

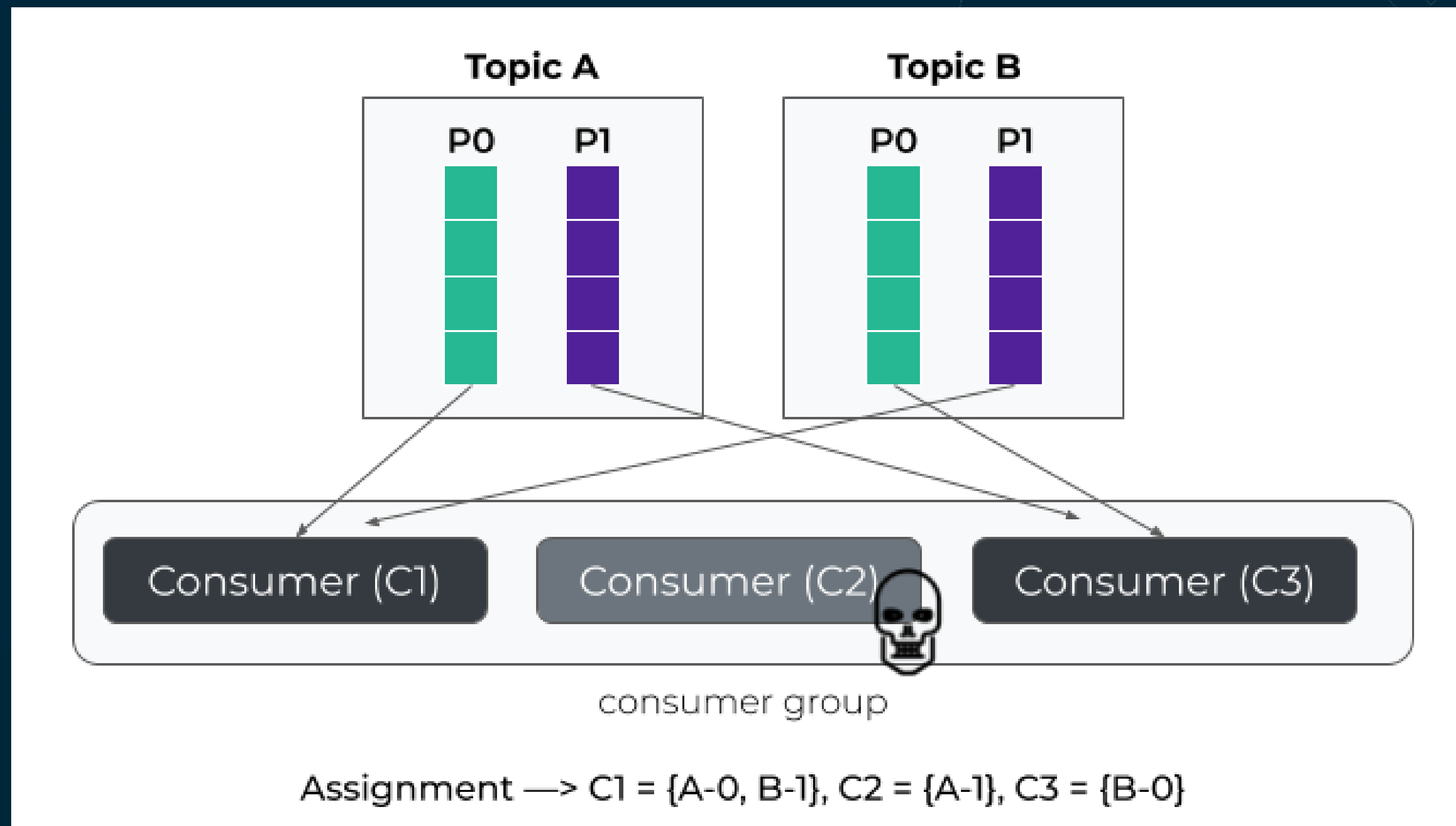
- Neste modelo as partições são distribuídas entre os consumidores usando o Algoritmo Round Robin.
- Aumenta o número de consumidores que serão usados, mas não diminui o número de movimentos.





## Estratégias de Particionamento: Sticky

- Aplica a mesma regra do Round Robin.
- A diferença está quando um dos consumidores sai do grupo, apenas essa partições é redistribuída, diminuindo o número de movimentos.



## Static Group Membership

- Define para cada consumer um set de partições, baseado no `group.instance.id`.
- Este consumer pode sair e entrar do grupo, as mesmas partições serão atribuídas a ele e nenhum outro consumer será afetado. (Zero Rebalance).
- O Rebalance ainda pode ocorrer, se o consumer ficar muito tempo fora (acima do `session timeout`).

## Static Membership Demo