

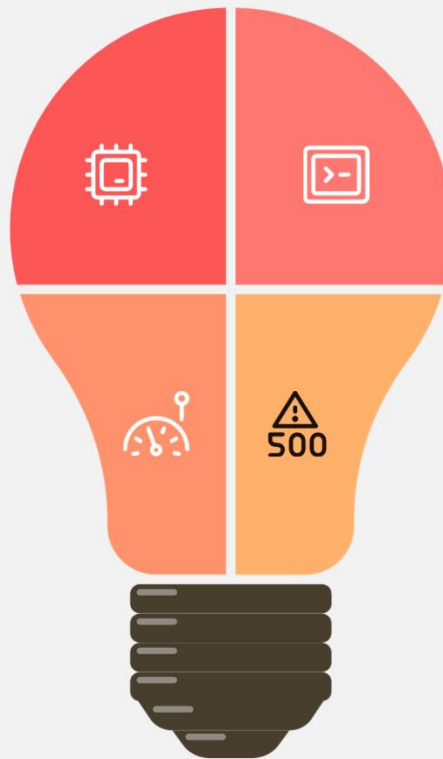
Broadcast Join

Execução Local

Uso no Spark

Ganho de Tempo

Limite de Memória



Broadcast Join

Dataset de exemplo

Tabela GRANDE (fato)

id	valor_fato
1	100
2	200
3	300
4	400
5	500
6	600
7	700
8	800

Distribuída em 4 partições (original)

Partição 1

id	valor_fato
1	100
2	200

Partição 2

id	valor_fato
3	300
4	400

Partição 3

id	valor_fato
5	500
6	600

Partição 4

id	valor_fato
7	700
8	800

Tabela PEQUENA (dimensão)

id	categoria
1	A
2	B
3	C
4	D

✓ Tabela pequena o suficiente para caber na memória de cada executor

O que é Broadcast Join?

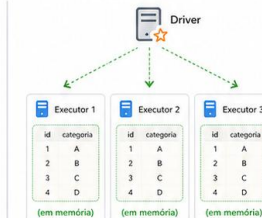
- O Spark envia (broadcast) a tabela pequena para todos os executores.
- Cada executor guarda uma cópia em memória.
- As partições da tabela grande são processadas localmente, usando essa cópia.
- Não há shuffle da tabela grande → muito mais eficiente!

1 Como funciona o Broadcast Join

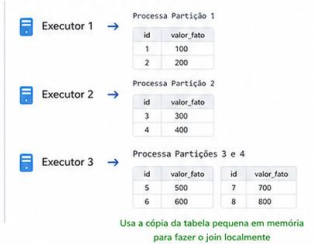
1 Spark identifica a tabela pequena e decide fazer broadcast

id	categoria
1	A
2	B
3	C
4	D

2 Spark faz broadcast para todos os executores



3 Cada executor processa suas partições da tabela grande localmente



2 Resultado: Join sem shuffle da tabela grande

Resultado final do join

id	valor_fato	categoria
1	100	A
2	200	B
3	300	C
4	400	D
5	500	(null)
6	600	(null)
7	700	(null)
8	800	(null)

Por que é eficiente?

- ✓ Apenas a tabela pequena é transmitida (uma vez por executor)
- ✓ A tabela grande não passa por shuffle
- ✓ Processamento local em cada executor
- ✓ Menos uso de rede, disco e CPU

Quando usar?

Quando uma das tabelas do join for pequena o suficiente para caber na memória de cada executor.

Configuração automática:

`spark.sql.autoBroadcastJoinThreshold` (padrão: 10MB)

Resumo rápido

Aspecto	Broadcast Join	SortMerge Join (shuffle)
Shuffle da tabela grande?	Não	Sim
Dados transmitidos	Apenas tabela pequena	Ambas as tabelas
Uso de memória	Memória dos executores	Disco + Memória
Rede	Baixo	Alto
Performance	Muito mais rápida	Mais lenta
Melhor caso	Tabela pequena + tabela grande	Tabelas grandes

Dica de ouro

Broadcast Join é uma das otimizações mais poderosas do Spark. Sempre que possível, deixe o Spark usar broadcast para evitar shuffle desnecessário!

```
df.join(broadcast(dim_df), "id")  
# força o broadcast
```

⚠ Lembre-se: Broadcast Join não é bom para tabelas grandes. Se a tabela pequena não couber na memória dos executores, pode causar estouro de memória (OOM).