



# **Administração de banco de dados**

## **Aula 08 - Planos de execução**

---

**Gustavo Maia**

**[Gustavo.Maia@FaculdadeImpacta.com.br](mailto:Gustavo.Maia@FaculdadeImpacta.com.br)**

# Agenda

---

- **Plano de Execução**

- Conceitos de Planos de Execução
- Elementos Comuns de Planos de Execução
- Trabalhando com Plano de Execução
- Projetando Efetivamente Índices Nonclustered
- Monitorando Performance

# Importância do Plano de Execução

---

- **Questões comuns**

- Por que a minha consulta demora tanto tempo para executar?
- Porque é que esta consulta leva muito mais tempo do que uma consulta muito semelhante?
- Por que é que o SQL Server está ignorando o índice que eu criei?

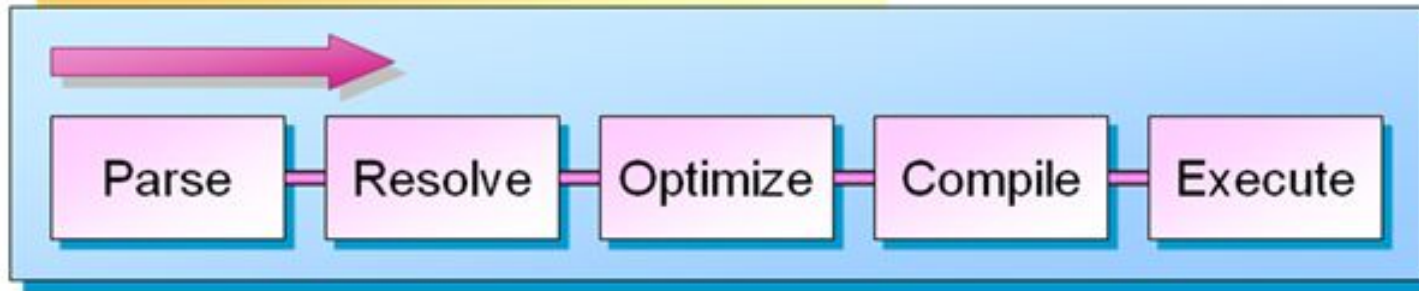
- **Planos de execução podem ajudar**

- Melhor uso de índices e na verificação de planos esperados

# Fases de execução da query

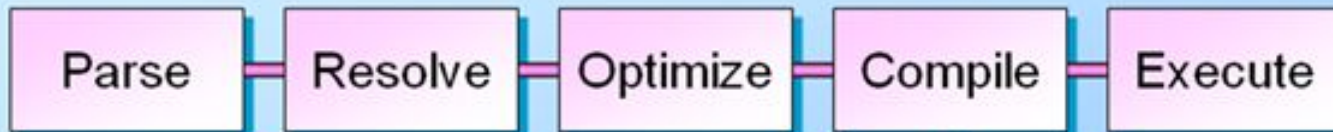
## ◆ How Queries Are Processed

### Uncached Queries (Ad Hoc)



### Cached Queries

#### First Execution



#### Subsequent Execution



# O que é um Plano de Execução

Um plano de execução mostra como a consulta foi executada ou como ele seria executada

- **Planos de execução detalham escolhas que o SQL Server faz:**
  - Tipos de operações
  - Ordem das operações
  - Escolha de índices
  - Estimativas de número de linhas com base em estatísticas disponíveis
- **SQL Server executa a otimização baseada no custo**
  - Um custo é atribuído a cada elemento

# Plano Real x Estimado

- **Existe uma opção para pedir ao SQL Server como seria executar uma consulta**
  - Conhecido como um plano de execução estimado
  - O SQL Server não executar a consulta, ele simplesmente simula o modo como iria executá-la
  - Nem sempre é possível obter um plano estimado
- **Um plano real pode ser diferente de um plano estimado**
  - Quando faltam estatísticas ou estão desatualizadas
  - Os recursos indisponíveis em tempo de execução
- **Plano estimado retorna contagens estimadas baseadas em estatísticas**
  - Plano real retorna contagens reais baseadas nos objetos

# O que é Execução de Contexto

- **Vários usuários podem executar um plano simultaneamente**
- Plano de execução detalha como a consulta seria executada
- Não mantém os dados para uma execução específica
- **O contexto de execução contém os dados associados a uma execução específica do plano**
- Um contexto de execução é necessário para cada execução simultânea
- Os contextos de execução em geral não são idênticos

# Formatos de Planos de Execução

- **Planos baseados em texto (agora obsoletos)**

- SET SHOWPLAN\_TEXT ON
- SET ON SHOWPLAN\_ALL
- SET STATISTICS PROFILE

- **Planos baseados em XML**

- SET SHOWPLAN\_XML ON
- SSMS salva em arquivos .sqlplan, formato portátil para planos de execução

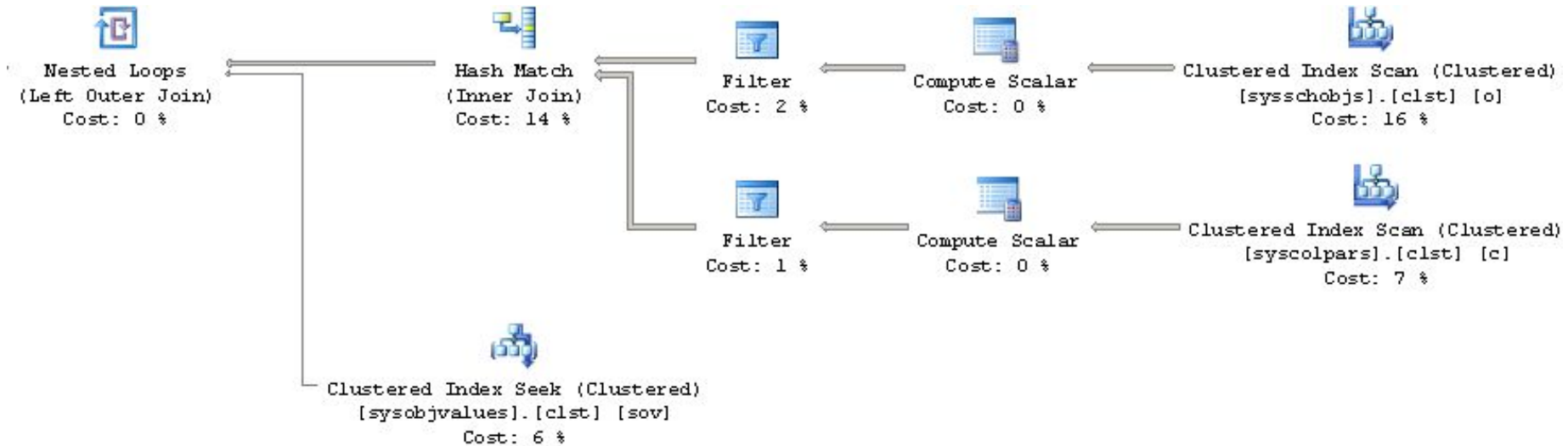
- **Planos Gráficos**

- Processa planos de XML em um formato de fácil leitura
- Não contém todas as informações do plano de XML



# Interpretando um Planos de Execução

```
select *
from sys.objects so
inner join sys.columns sc
on so.object_id = sc.object_id
where so.type = 'U'
```



# Interpretando um Planos de Execução

```
select *
from sys.objects so
inner join sys.columns sc
on so.object_id = sc.object_id
where so.type = 'U'
```



```
Clustered Index Scan (Clustered)
[sys].[sys].[sysschobjs].[clst] [o]
Cost: 16 %
```

## Clustered Index Scan (Clustered)

Scanning a clustered index, entirely or only a range.

Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Row
Storage	RowStore
Estimated I/O Cost	0,043125
Estimated Operator Cost	0,0460243 (16%)
Estimated Subtree Cost	0,0460243
Estimated CPU Cost	0,0028993
Estimated Number of Executions	1
Estimated Number of Rows	191,769
Estimated Row Size	176 B
Ordered	False
Node ID	23

## Predicate

```
[master].[sys].[sysschobjs].[nsclass] as [o].[nsclass]=(0) AND
[master].[sys].[sysschobjs].[pclass] as [o].[pclass]=(1) AND
CONVERT(char(2),CASE WHEN [master].[sys].[sysschobjs].
[type] as [o].[type]='ET' THEN 'U' ELSE [master].[sys].
[sysschobjs].[type] as [o].[type] END,0)='U'
```

## Object

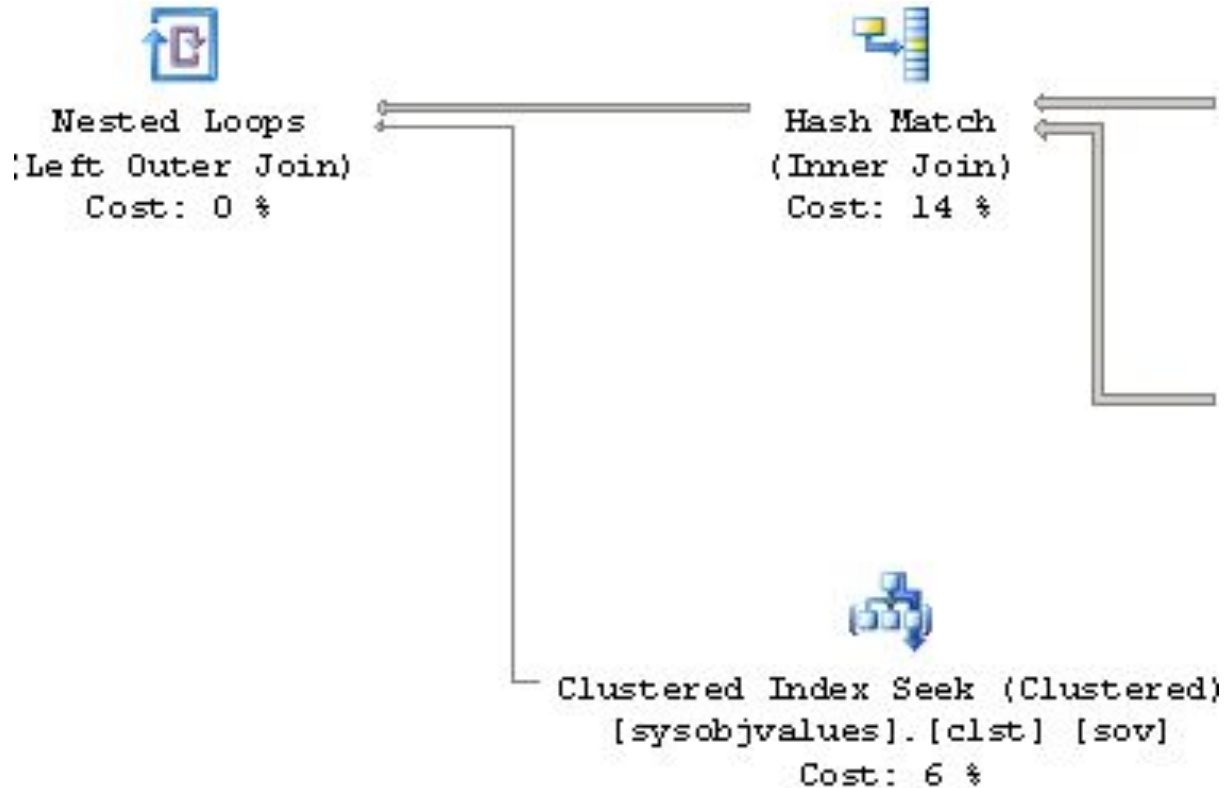
```
[master].[sys].[sysschobjs].[clst] [o]
```

## Output List

```
[master].[sys].[sysschobjs].id; [master].[sys].
[sysschobjs].name; [master].[sys].[sysschobjs].nsid; [master].
[sys].[sysschobjs].status; [master].[sys].[sysschobjs].type;
[master].[sys].[sysschobjs].pid; [master].[sys].
[sysschobjs].created; [master].[sys].[sysschobjs].modified
```

# Interpretando um Planos de Execução

```
select *
from sys.objects so
inner join sys.columns sc
on so.object_id = sc.object_id
where so.type = 'U'
```



# Métodos de acesso

- **Table Scan**

- Leitura de varredura um HEAP



Table Scan

- **Clustered index scan**

- **Index Scan**

- A leitura de varredura numa tabela com um índice clustered / nonclustered



Clustered Index Scan (Clustered)

- **Clustered index seek**

- **index seek**

- Leitura de pesquisa um índice clustered / nonclustered para uma específica posição dentro da tabela



Clustered Index Seek (Clustered)

# Lookups

---

- **RID Lookup**

- Procura em um HEAP usando o row ID



RID Lookup (Heap)

- **Key Lookup**

- Procura em um índice clustered



Key Lookup (Clustered)

# Tipos de Joins

## • Nested Loops

- Comumente usado para operações de junção interna
- Para cada linha de caminho de dados superior, realiza uma pesquisa para o caminho de dados inferior



## • Merge Join

- Comumente usado em associações internas
- Requer que as duas entradas estejam na mesmo ordem de classificação



## • Hash Match

- Joins mais difíceis, onde uma tabela hash é construída calculando um valor de hash para cada linha de entrada



# Aggregations

- **Stream Aggregate**

- Os dados já estão na ordem correta para o processamento da agregação



- **Hash Match Aggregate**

- Tabela de hash é criada para formar agregação com os dados que não estão na ordem necessária



# Filter and Sort

## • Filter

- Operação de baixo custo
- Normalmente usado para predicados da cláusula WHERE ou HAVING
- Passa somente pelas linhas que correspondem aos critérios de filtro necessário



## • Sort

- Usado sempre que uma operação de classificação é necessária
- Muitas vezes usado para cláusulas ORDER BY
- Pode ser usado para outras operações como triagem insumos para a junção mesclar operações ou realizar operações distintas
- Pode ser muito caro





# Modificações de dados

- **INSERT**

- Usado em operações de inserção



- **UPDATE**

- Usado em operações de atualização



- **DELETE**

- Usado em operações de remoção



# Métodos para Captura de Planos

---

- SQL Server Management Studio
  - Plano estimado e real
- Dynamic Management Views (DMVs)
- Activity Monitor
- SQL Server Data Collection

# Reaproveitando execução de Queries

- **Plano de re-uso é geralmente desejável**
  - Questões de parâmetros sniffing são uma exceção a regra
- **Os planos podem se tornar inutilizáveis ou abaixo do ideal**
  - Exatidão da estrutura (alterações de schema, modificação de opções, ...)
  - Otimalidade (dados tenham sido modificado o suficiente para que um novo plano deva ser considerado)
- **Planos despejados pelo custo de compilação do algoritmo**
  - Custo reduzido ao longo do tempo
  - Reutilização restabelece custo
  - Custo de zero indica um candidato para despejo
- **As opções estão disponíveis para forçar o comportamento de compilação**

# DMVs Relacionadas

- **Dynamic Management Views and Functions**
  - Mostram o estado atual do servidor
  - Não persistem os dados internamente
  
- **Many useful views/functions related to execution:**
  - sys.dm\_exec\_connections
  - sys.dm\_exec\_sessions
  - sys.dm\_exec\_query\_stats
  - sys.dm\_exec\_requests
  - sys.dm\_exec\_sql\_text()
  - sys.dm\_exec\_query\_plan()
  - sys.dm\_exec\_cached\_plans
  - sys.dm\_exec\_cached\_plan\_dependent\_objects()

# Atuando sobre um plano de execução

- **Ambiente de testes / coletando mais informações**

--para coletar estatísticas de tempo de execução e disco

set statistics time on

set statistics io on

--para limpar o cache

dbcc freeproccache

dbcc dropcleanbuffers

# Atuando sobre um plano de execução

**Query Hints** – afeta toda a consulta

<Query>

OPTION (...)

**Table Hints** – afeta o método de acesso daquela tabela

<tabela> WITH (...)

**Join Hints** – afeta o algoritmo daquele join

<tabela 1> inner <nome do algoritmo> join <tabela 2>

Link: <https://docs.microsoft.com/en-us/sql/t-sql/queries/hints-transact-sql-query?view=sql-server-2017>

# Atuando sobre um plano de execução

## Query Hints

Ex: Reduzindo o problema de parameter sniffing ( planos de execuções muito diferentes baseado nos parâmetros recebidos )

Declare @codCli int

Select \* from <tabela> where cliente = @codCli

OPTION ( OPTIMIZE FOR UNKNOWN )

...Ou....

OPTION ( OPTIMIZE FOR ( @codCli = 1234 ) )

# Atuando sobre um plano de execução

## Query Hints

Ex: Controlando paralelismo

```
Select * from <tabela> where cliente = @codCli
OPTION ( MAXDOP = 1 )
```

```
sp_configure 'show advanced options', 1
reconfigure
go
sp_configure 'cost threshold for parallelism', 50
sp_configure 'max degree of parallelism', 4
sp_configure 'show advanced options', 0
reconfigure
go
```



# Atuando sobre um plano de execução

## Table Hints

- Ajustando o nível de isolamento

Select \* from <tabela> WITH(NOLOCK)

- Indicando o tipo de LOCK a ser colocado

Update <tabela> WITH(ROWLOCK) set <campo> = 10

- Forçando um índice

Select \* from <tabela> WITH( index = <nome do índice> )

# Atuando sobre um plano de execução

## Join Hints

-Escolhendo o algoritmo de join:

<tabela 1> inner LOOP join <tabela 2>

<tabela 1> inner MERGE join <tabela 2>

<tabela 1> inner HASH join <tabela 2>

Obs:

Isso é válido para os vários tipos de join

Às vezes o plano adiciona passos adicionais para atender à sua especificação



**Obrigado !**

---

**Gustavo Maia**  
**Gustavo.Maia@FaculdadeImpacta.com.br**