

Important Information in a Plan



Erin Stellato

PRINCIPAL CONSULTANT

@erinstellato www.sqlskills.com/blogs/erin



Module Overview



Review additional, essential information in a query plan, including:

- Row estimates/actuals
- Parallelism
- Predicates
- Warnings

Input Parameters

Input values for a query, used on initial compilation, can be found in the plan

This can be extremely important when comparing an actual plan (with different values) to the plan in cache



Demo



Finding Input Parameters in a Plan



Trace Flags

Global and session trace flags are captured by execution plans in SQL Server 2014 SP2 and SQL Server 2016

Use IsCompileTime attribute to determine if flag present at compilation and execution

<https://support.microsoft.com/en-us/kb/3170115>



Demo



Trace Flags in a Query Plan



Cardinality Estimator Version

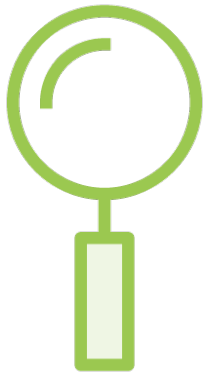
The cardinality estimator (CE) was significantly changed in SQL Server 2014

The new CE can be enabled in a variety of ways

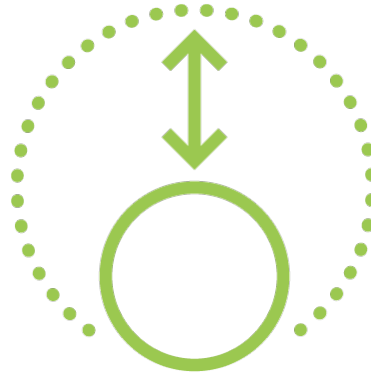
To confirm CE used, check attribute value in the plan



Cardinality Estimate Issues



Check for differences
between estimates
and actual



Can cause significantly
more data to move
through the plan
than needed



Validate statistics, CE
version, table
variables, table value
functions to start

Demo



Examining CE Version and Estimates



Operator Execution Statistics



**Available in SQL Server 2014 SP2
and SQL Server 2016**



**Includes CPU, duration, and I/O
information on a per-operator basis**



Demo



Execution Statistics for Operators



Parallelism in Plans

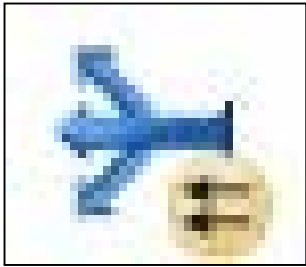
If MAXDOP is not set to 1, the optimizer can generate a serial or parallel plan

Not every plan can be parallelized

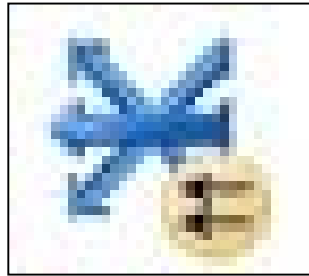
Indicates a higher cost query – not necessarily good or bad



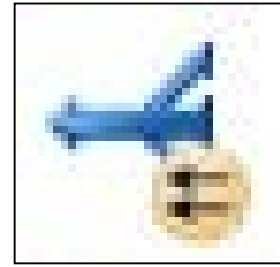
Parallelism and Operators



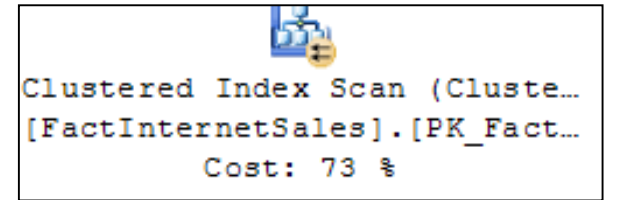
**Distribute
Streams**



**Repartition
Streams**



Gather Streams



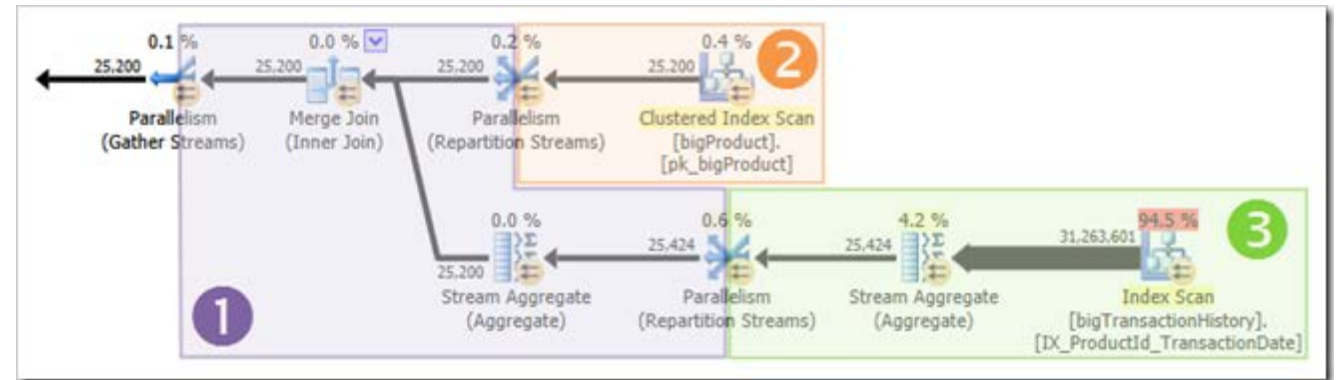
**Operator
running in
parallel**

Parallelism

Parallel plans can have multiple zones of concurrent execution

MAXDOP limits the maximum number of threads *per operator*

MAXDOP does not limit the total threads for the plan



<https://sqlperformance.com/2013/10/sql-plan/parallel-plans-branches-threads>



Demo



Parallelism in Plans



Seek Predicate

Clustered Index Seek (Clustered)	
Scanning a particular range of rows from a clustered index.	
Physical Operation	Clustered Index Seek
Logical Operation	Clustered Index Seek
Actual Number of Rows	290
Estimated I/O Cost	0.003125
Estimated CPU Cost	0.0001581
Number of Executions	290
Estimated Number of Executions	290
Estimated Operator Cost	0.071616 (83%)
Estimated Subtree Cost	0.071616
Estimated Number of Rows	1
Estimated Row Size	187 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	True
Node ID	3
Object	
[AdventureWorks].[Person].[Contact]. [PK_Contact_ContactID] [c]	
Output List	
[AdventureWorks].[Person].[Contact].Title, [AdventureWorks].[Person].[Contact].FirstName, [AdventureWorks].[Person].[Contact].MiddleName, [AdventureWorks].[Person].[Contact].LastName, [AdventureWorks].[Person].[Contact].Suffix	
Seek Predicates	
Seek Keys[1]: Prefix: [AdventureWorks].[Person]. [Contact].ContactID = Scalar Operator([AdventureWorks]. [HumanResources].[Employee].[ContactID] as [e]. [ContactID])	

Object

[AdventureWorks].[Person].[Contact].
[PK_Contact_ContactID] [c]

Output List

[AdventureWorks].[Person].[Contact].Title,
[AdventureWorks].[Person].[Contact].FirstName,
[AdventureWorks].[Person].[Contact].MiddleName,
[AdventureWorks].[Person].[Contact].LastName,
[AdventureWorks].[Person].[Contact].Suffix

Seek Predicates

Seek Keys[1]: Prefix: [AdventureWorks].[Person].
[Contact].ContactID = Scalar Operator([AdventureWorks].
[HumanResources].[Employee].[ContactID] as [e].
[ContactID])



Residual Predicate

Clustered Index Seek (Clustered)	
Scanning a particular range of rows from a clustered index.	
Physical Operation	Clustered Index Seek
Logical Operation	Clustered Index Seek
Actual Number of Rows	1
Estimated I/O Cost	0.003125
Estimated CPU Cost	0.0001581
Number of Executions	290
Estimated Number of Executions	290
Estimated Operator Cost	0.071616 (83%)
Estimated Subtree Cost	0.071616
Estimated Number of Rows	1
Estimated Row Size	200 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	True
Node ID	3

Predicate	
[AdventureWorks].[Person].[Contact].[LastName] as [c].	
[LastName]=N'Ellerbrock'	
Object	
[AdventureWorks].[Person].[Contact].	
[PK_Contact_ContactID] [c]	
Output List	
[AdventureWorks].[Person].[Contact].Title,	
[AdventureWorks].[Person].[Contact].FirstName,	
[AdventureWorks].[Person].[Contact].MiddleName,	
[AdventureWorks].[Person].[Contact].LastName,	
[AdventureWorks].[Person].[Contact].Suffix,	
[AdventureWorks].[Person].[Contact].EmailAddress	
Seek Predicates	
Seek Keys[1]: Prefix: [AdventureWorks].[Person].	
[Contact].ContactID = Scalar Operator([AdventureWorks].	
[HumanResources].[Employee].[ContactID] as [e].	
[ContactID])	

Predicate

```
[AdventureWorks].[Person].[Contact].[LastName] as [c].  
[LastName]=N'Ellerbrock'
```

Object

```
[AdventureWorks].[Person].[Contact].  
[PK_Contact_ContactID] [c]
```

Output List

```
[AdventureWorks].[Person].[Contact].Title,  
[AdventureWorks].[Person].[Contact].FirstName,  
[AdventureWorks].[Person].[Contact].MiddleName,  
[AdventureWorks].[Person].[Contact].LastName,  
[AdventureWorks].[Person].[Contact].Suffix,  
[AdventureWorks].[Person].[Contact].EmailAddress
```

Seek Predicates

```
Seek Keys[1]: Prefix: [AdventureWorks].[Person].  
[Contact].ContactID = Scalar Operator([AdventureWorks].  
[HumanResources].[Employee].[ContactID] as [e].  
[ContactID])
```



Demo



Seek and Residual Predicates





Warnings

Spill to tempdb (hash and sort)

Implicit conversions

No statistics

Missing index

Missing join predicate

Demo



Generating Plan Warnings



What We Covered



Review additional, essential information in a query plan, including:

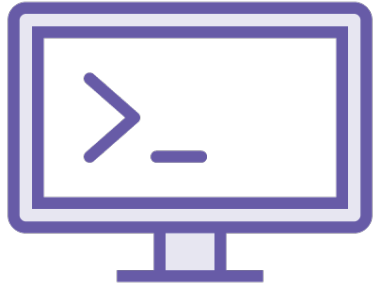
- Row estimates/actuals
- Parallelism
- Predicates
- Warnings



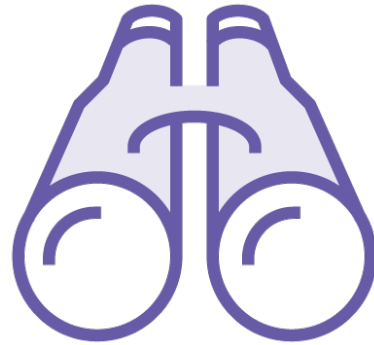
Next steps



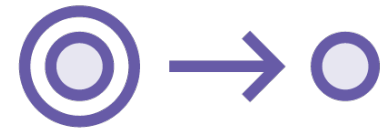
Practice, practice, practice



**Develop scripts to
capture query data**



**Review data and
execution plans**



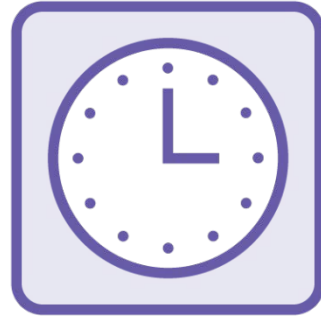
**Make incremental
changes to see how
performance changes**



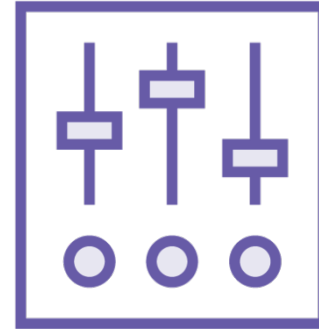
Course Summary



Information
about query
execution can
be captured in
SQL Server



Data can be
historical or
real-time



Use this data to
determine what
to tune/change



Continue to
monitor
performance

