

“How to Analyze MSSQL Query Execution Plans”

Quick Trainer Series



dataresearchlabs.com

How to Analyze MSSQL Query Execution Plans

- **What is a Query Execution Plan?**
- How to Read a Query Execution Plan
- How to Identify Warnings
- How to Identify the Slowest Operator
- How to Identify and Fix “Table Scans” and “Clustered Index Scans”
- How to Identify and Fix “Spools” and “Sorts”
- How to Identify and Fix “Missing or Stale Statistics”
- How to Identify Common Issues Automatically

What is a Query Execution Plan?

Definition of a “Query Execution Plan”...

“An execution plan...is the result of the query optimizer’s attempt to **calculate** the **most efficient** way to **implement** the request represented by the **T-SQL query** you submitted.”

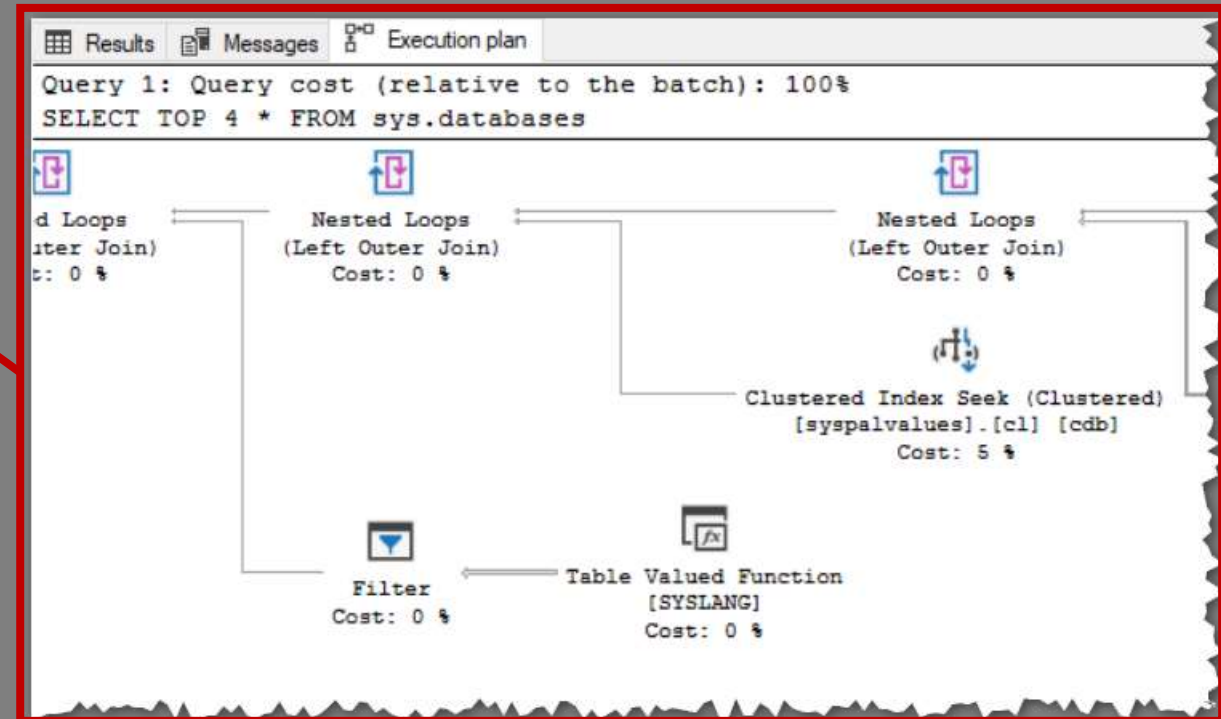
– red-gate.com

What is a Query Execution Plan?

■ Three Visual Formats for Exec Plans:

1. Graphical Exec Plans (this video)

- Easy to Read
- Low Detail

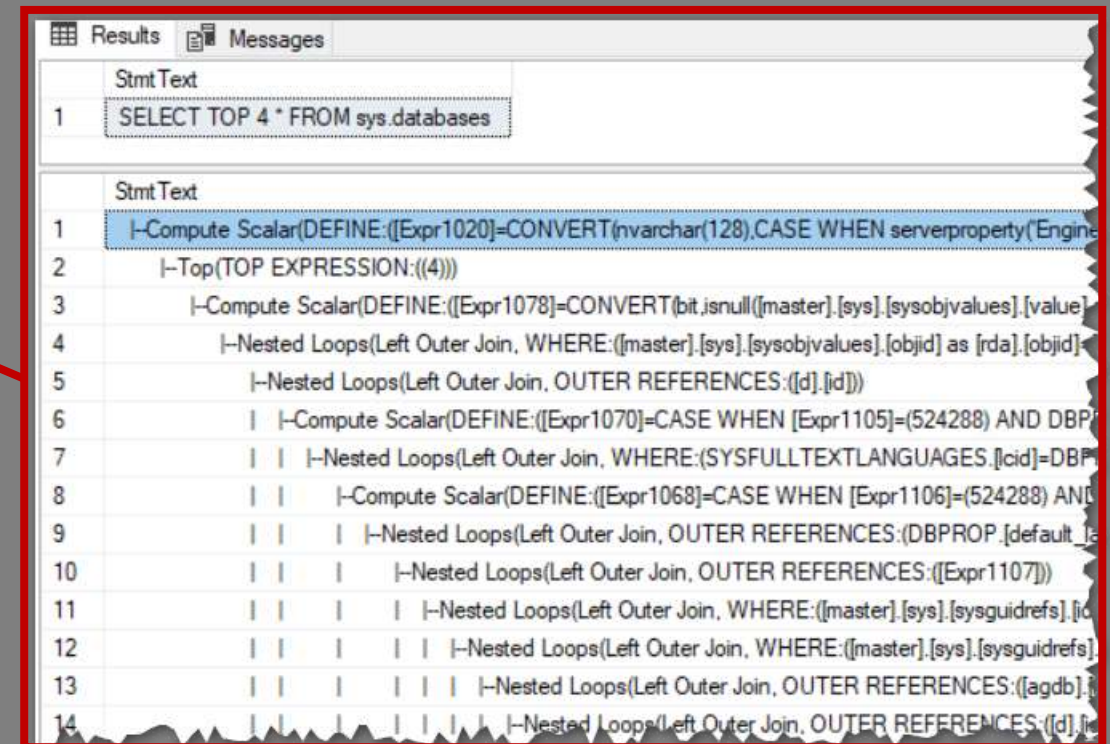


What is a Query Execution Plan?

■ Three Visual Formats for Exec Plans:

1. Graphical Exec Plans (this video)
2. Text Exec Plans (more detail)

- Harder to Read
- More Detail



	Stmt Text
1	SELECT TOP 4 * FROM sys.databases

	Stmt Text
1	-Compute Scalar(DEFINE:([Expr1020]=CONVERT(nvarchar(128),CASE WHEN serverproperty('Engine
2	-Top(TOP EXPRESSION:((4)))
3	-Compute Scalar(DEFINE:([Expr1078]=CONVERT(bit,isnull([master].[sys].[sysobjvalues].[value]
4	-Nested Loops(Left Outer Join, WHERE:([master].[sys].[sysobjvalues].[objid] as [rda].[objid]
5	-Nested Loops(Left Outer Join, OUTER REFERENCES:([d].[id]))
6	-Compute Scalar(DEFINE:([Expr1070]=CASE WHEN [Expr1105]=(524288) AND DBP
7	-Nested Loops(Left Outer Join, WHERE:(SYSFULLTEXTLANGUAGES.[cid]=DBP
8	-Compute Scalar(DEFINE:([Expr1068]=CASE WHEN [Expr1106]=(524288) AND
9	-Nested Loops(Left Outer Join, OUTER REFERENCES:(DBPROP.[default_
10	-Nested Loops(Left Outer Join, OUTER REFERENCES:([Expr1107]))
11	-Nested Loops(Left Outer Join, WHERE:([master].[sys].[sysguidrefs].[id]
12	-Nested Loops(Left Outer Join, WHERE:([master].[sys].[sysguidrefs]
13	-Nested Loops(Left Outer Join, OUTER REFERENCES:([agdb].[
14	-Nested Loops(Left Outer Join, OUTER REFERENCES:([d].[id]

What is a Query Execution Plan?

■ Three Visual Formats for Exec Plans:

1. Graphical Exec Plans (this video)
2. Text Exec Plans (more detail)
3. XML Exec Plans (more detail)
 - Hardest to Read
 - Most Detail

```
Execution plan.xml  SQLQuery3.sql - testods*  SQLQuery1.sql - testods*
1  <?xml version="1.0" encoding="utf-16"?>
2  <ShowPlanXML xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://schemas.microsoft.com/sqlserver/2004/07/showplan" >
3  <BatchSequence>
4  <Batch>
5  <Statements>
6  <StmtSimple StatementCompId="1" StatementEstRows="1" StatementText="SELECT * FROM sys.tables" >
7  <StatementSetOptions ANSI_NULLS="true" ANSI_PADDING="true" ANSI_WARNINGS="on" ARITHABORT="on" >
8  <QueryPlan DegreeOfParallelism="0" NonParallelPlanReason="C" >
9  <Warnings>
10 <PlanAffectingConvert ConvertIssue="Cardinality Estimation" >
11 </Warnings>
12 <MemoryGrantInfo SerialRequiredMemory="0" SerialDesiredMemory="0" >
13 <OptimizerHardwareDependentProperties EstimatedAvailableParallelism="1" >
14 <OptimizerStatsUsage>
15 <StatisticsInfo Database="[master]" Schema="[sys]" Table="[sys].[tables]" >
16 <StatisticsInfo Database="[mssqlsystemresource]" Schema="[sys]" Table="[sys].[sysobjects]" >
17 <StatisticsInfo Database="[master]" Schema="[sys]" Table="[sys].[tables]" >
18 <StatisticsInfo Database="[master]" Schema="[sys]" Table="[sys].[tables]" >
19 <StatisticsInfo Database="[master]" Schema="[sys]" Table="[sys].[tables]" >
20 <StatisticsInfo Database="[master]" Schema="[sys]" Table="[sys].[tables]" >
21 <StatisticsInfo Database="[master]" Schema="[sys]" Table="[sys].[tables]" >
22 <StatisticsInfo Database="[master]" Schema="[sys]" Table="[sys].[tables]" >
23 <StatisticsInfo Database="[master]" Schema="[sys]" Table="[sys].[tables]" >
```

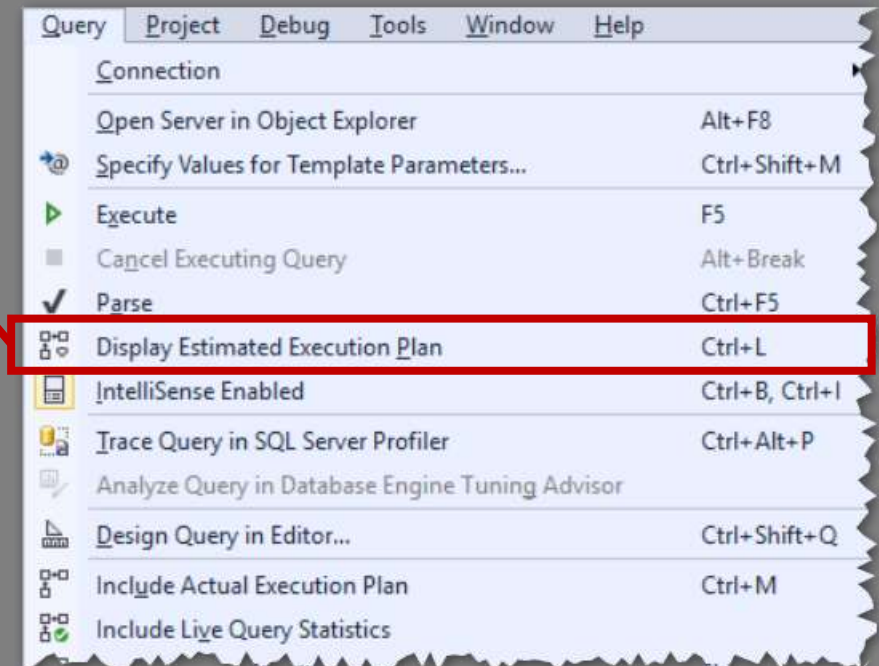
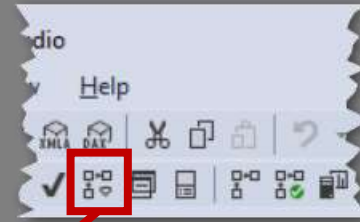
What is a Query Execution Plan?

- Three Visual Formats for Exec Plans:

1. Graphical Exec Plans (this video)
2. Text Exec Plans (more detail)
3. XML Exec Plans (more detail)

- Three Time Points for Exec Plans:

1. **Before:** Show Estimated Exec Plan
 - Less accurate, but virtually instant



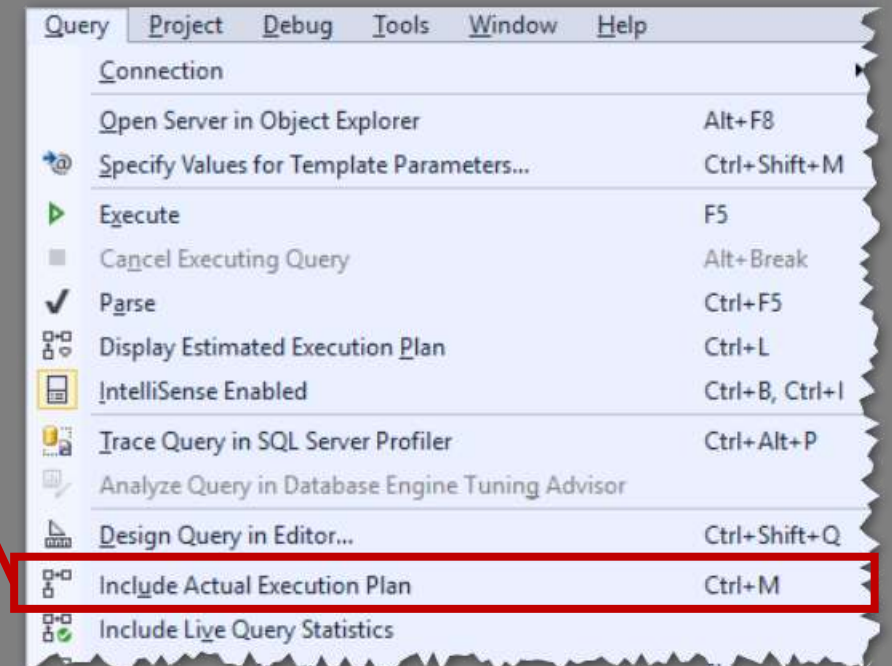
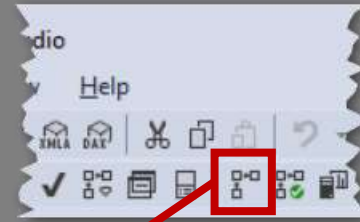
What is a Query Execution Plan?

- Three Visual Formats for Exec Plans:

1. Graphical Exec Plans (this video)
2. Text Exec Plans (more detail)
3. XML Exec Plans (more detail)

- Three Time Points for Exec Plans:

1. Before: Show Estimated Exec Plan
2. **After: Include Actual Exec Plan**
 - Most accurate, but must wait completion



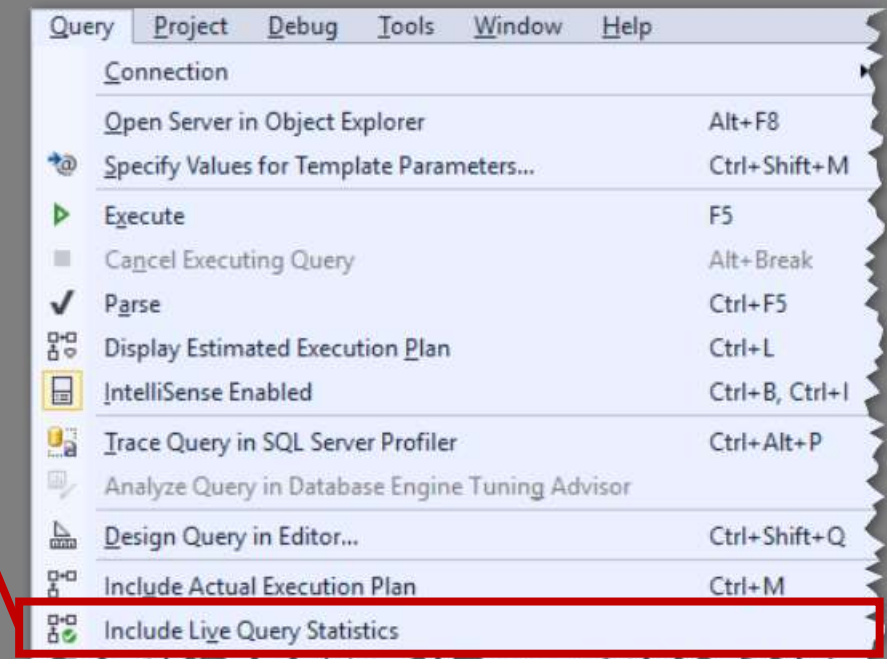
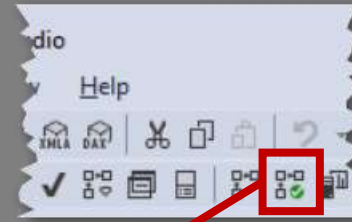
What is a Query Execution Plan?

- Three Visual Formats for Exec Plans:

1. Graphical Exec Plans (this video)
2. Text Exec Plans (more detail)
3. XML Exec Plans (more detail)

- Three Time Points for Exec Plans:

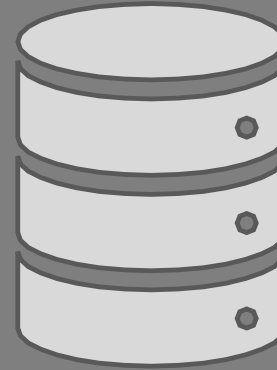
1. Before: Show Estimated Exec Plan
2. After: Include Actual Exec Plan
3. **During: Include Live Query Stats**
 - More accurate, results while run



What is a Query Execution Plan?

- Three Visual Formats for Exec Plans:
 1. Graphical Exec Plans (this video)
 2. Text Exec Plans (more detail)
 3. XML Exec Plans (more detail)
- Three Time Points for Exec Plans:
 1. Before: Show Estimated Exec Plan
 2. After: Include Actual Exec Plan
 3. During: Include Live Query Stats
- Other Types of Exec Plans:
 1. “Cached Plans” in SQL memory
 2. “Logged Plans” as XML in Table

CACHED PLAN

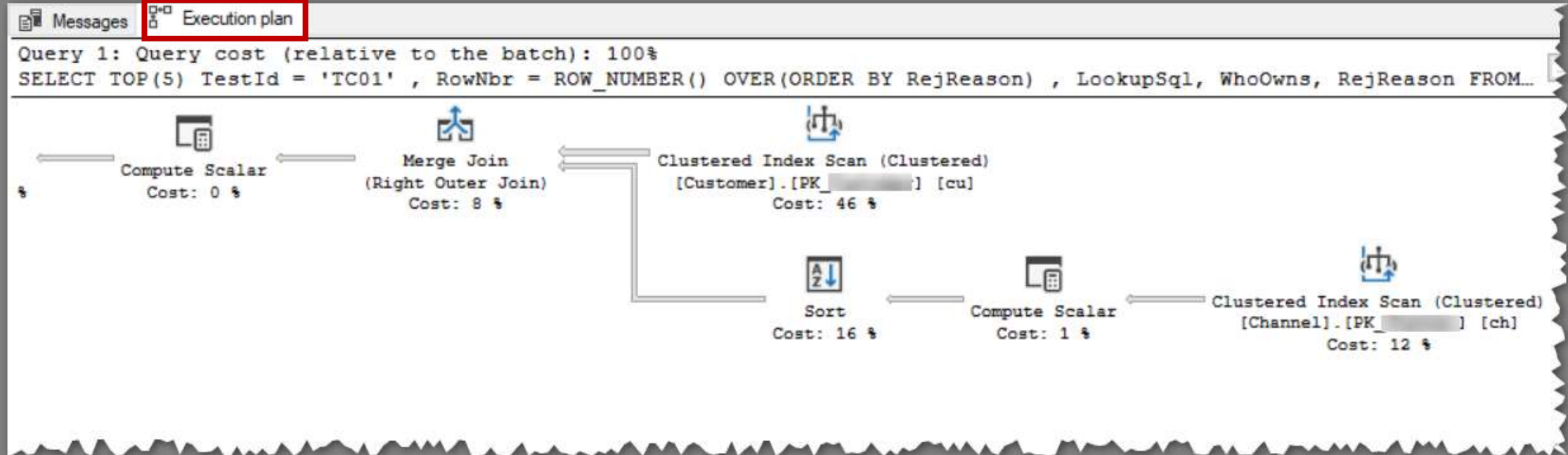


LOGGED PLAN

What is a Query Execution Plan?

Here is a screenshot of a graphical Query Execution Plan

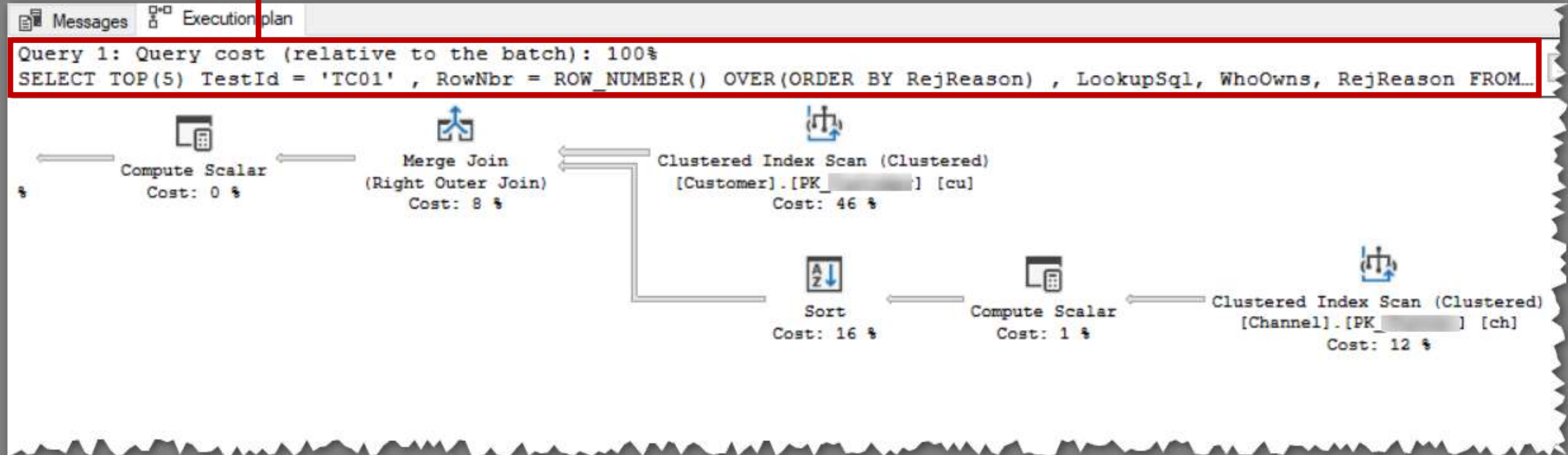
- Notice the tab section



What is a Query Execution Plan?

Here is a screenshot of a graphical Query Execution Plan

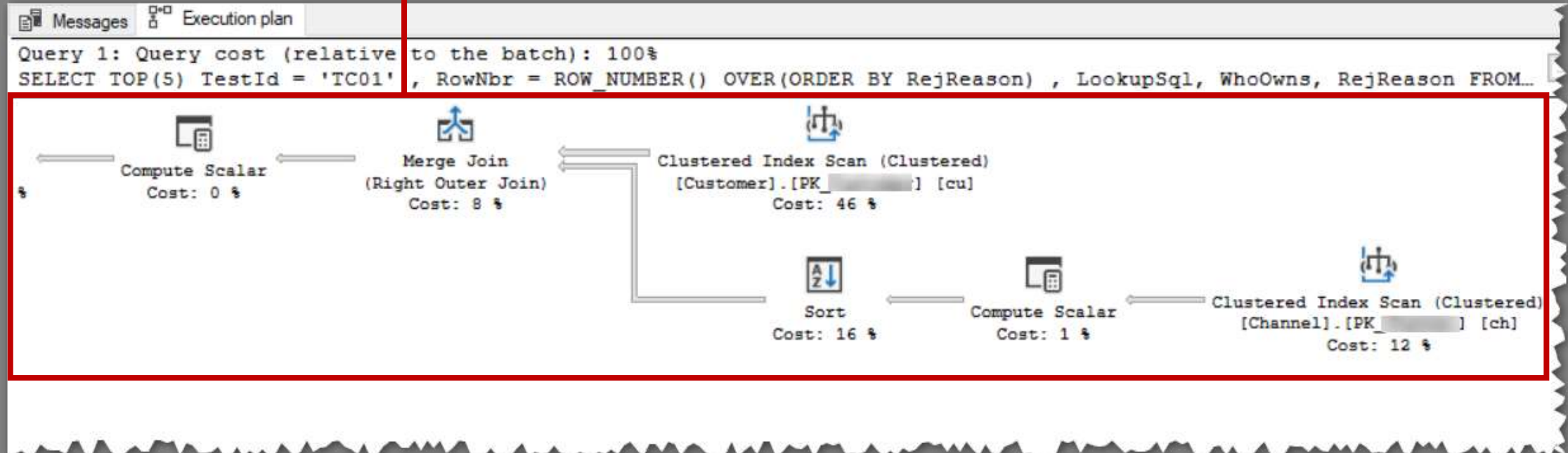
- Notice the tab section
- Notice the SQL text section



What is a Query Execution Plan?

Here is a screenshot of a graphical Query Execution Plan

- Notice the tab section
- Notice the SQL text section
- Notice the Execution Plan Flowchart section



How to Analyze MSSQL Query Execution Plans

- What is a Query Execution Plan?
- **How to Read a Query Execution Plan**
- How to Identify Warnings
- How to Identify the Slowest Operator
- How to Identify and Fix “Table Scans” and “Clustered Index Scans”
- How to Identify and Fix “Spools” and “Sorts”
- How to Identify and Fix “Missing or Stale Statistics”
- How to Identify Common Issues Automatically

How to Read a Query Execution Plan

Assumptions:

- You have permissions

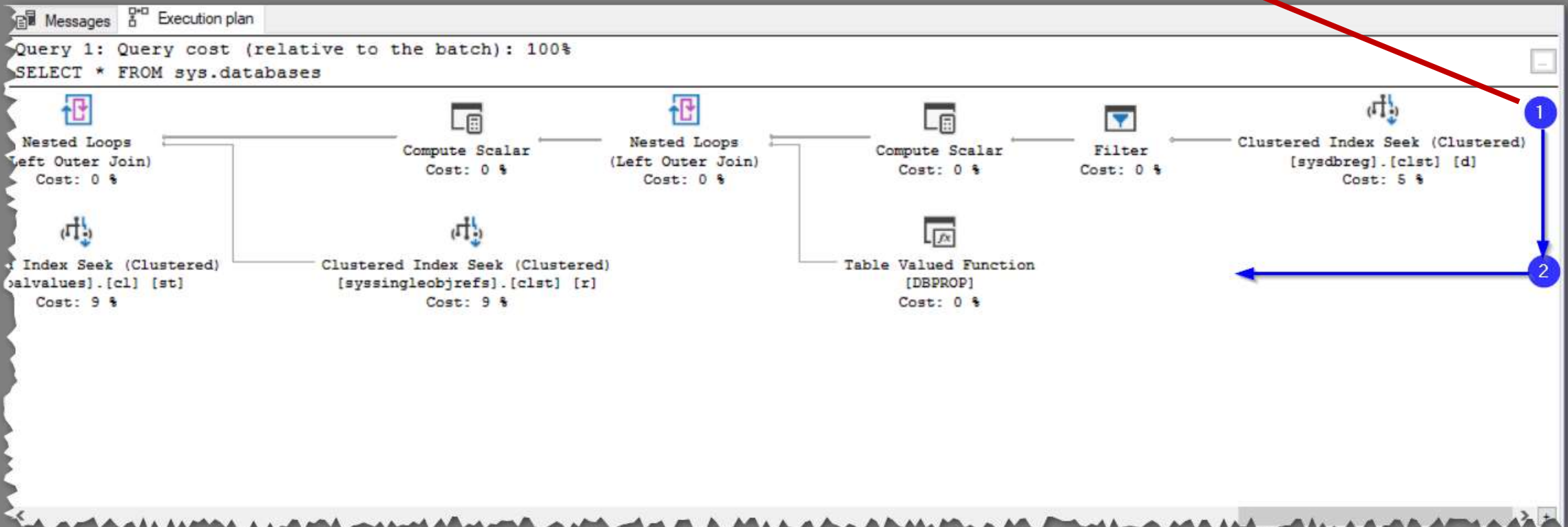


```
1 | GRANT SHOWPLAN TO johndoe
2 |
```

How to Read a Query Execution Plan

Assumptions:

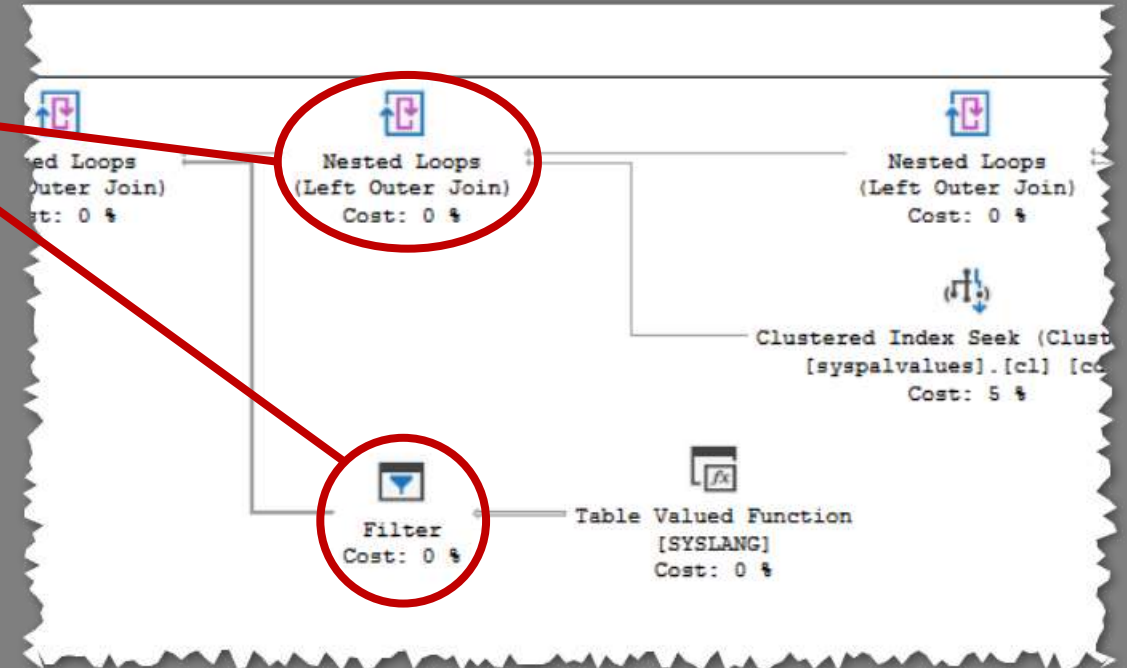
- You have permissions
- You Read the Flow Chart Top-to-Bottom, Right-to-Left



How to Read a Query Execution Plan

Flow Chart Objects:

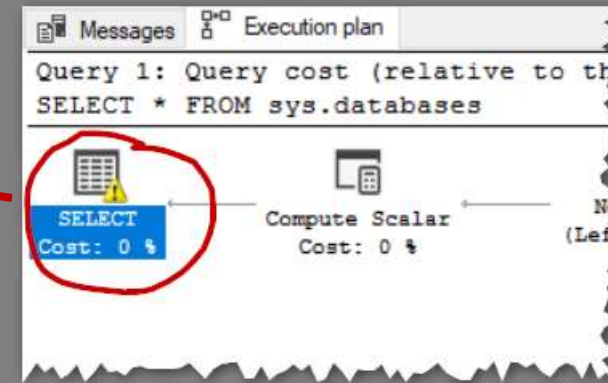
- Icons are “Operators”
 - Each operator is named
 - There are approx. 78 operator types



How to Read a Query Execution Plan

Flow Chart Objects:

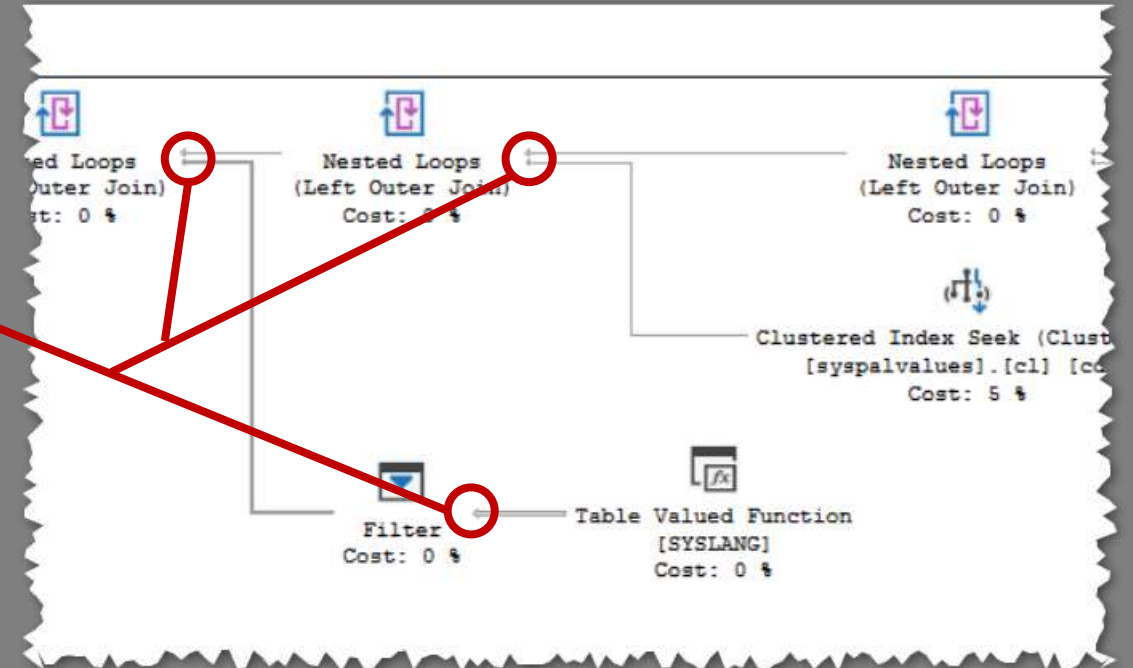
- Icons are “Operators”
- Ignore Select Operator at far left
(just signifies end of the flow)



How to Read a Query Execution Plan

Flow Chart Objects:

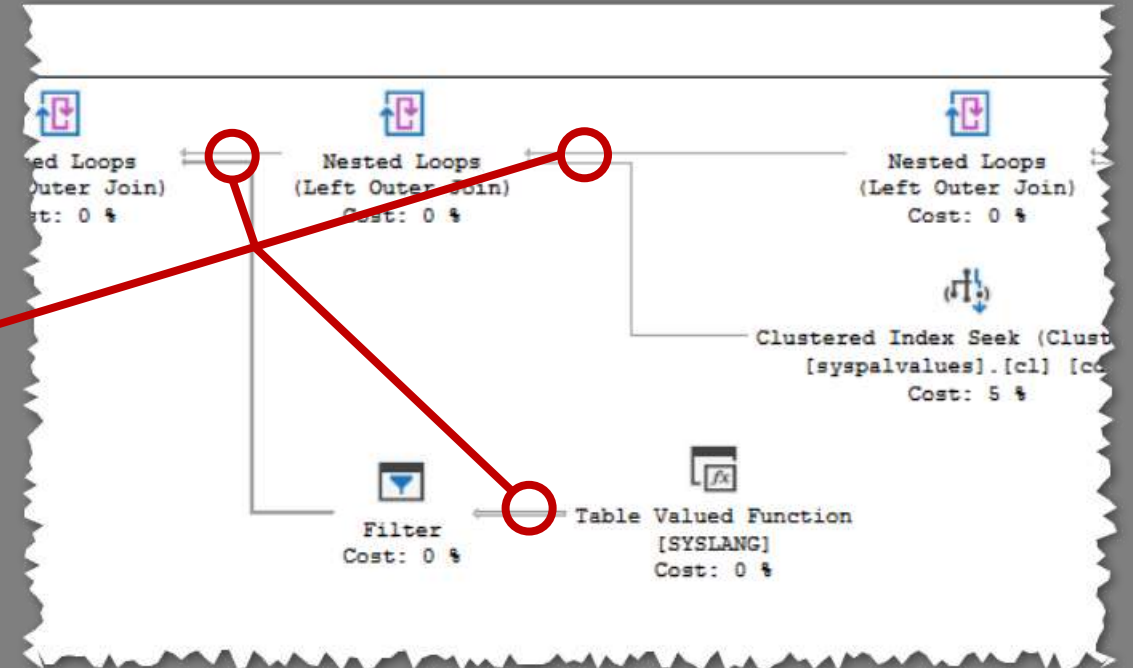
- Icons are “Operators”
- Ignore Select Operator at far left
- Arrows indicate data flow direction



How to Read a Query Execution Plan

Flow Chart Objects:

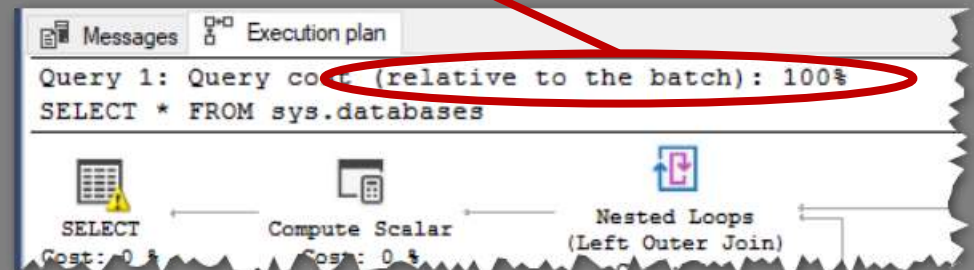
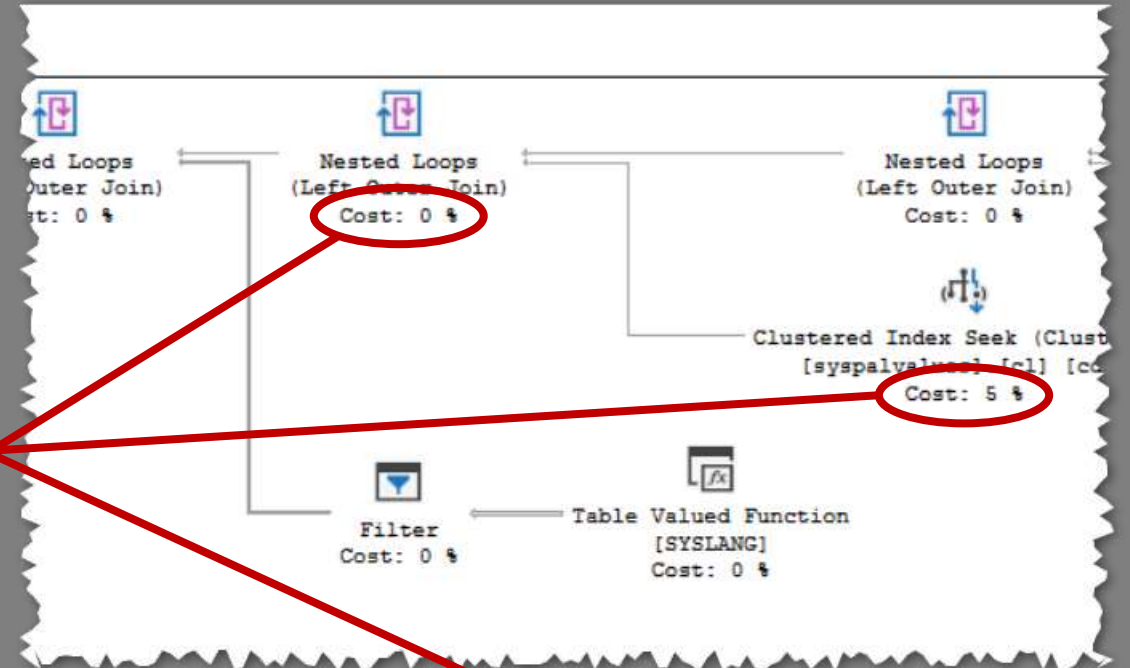
- Icons are “Operators”
- Ignore Select Operator at far left
- Arrows indicate data flow direction
- Thickness of arrows indicates relative amount of data flowing
(hover mouse points to get row count)



How to Read a Query Execution Plan

Flow Chart Objects:

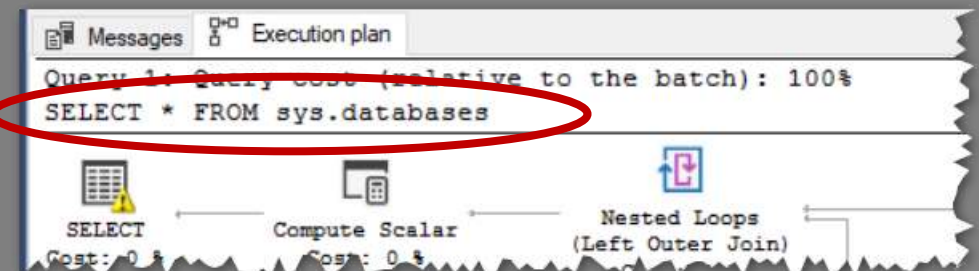
- Icons are “Operators”
- Ignore Select Operator at far left
- Arrows indicate data flow direction
- Thickness of arrows indicates relative amount of data flowing
- Cost percent indicates relative cost to the query for given operator
(will total to 100% in the flow diagram)



How to Read a Query Execution Plan

Flow Chart Objects:

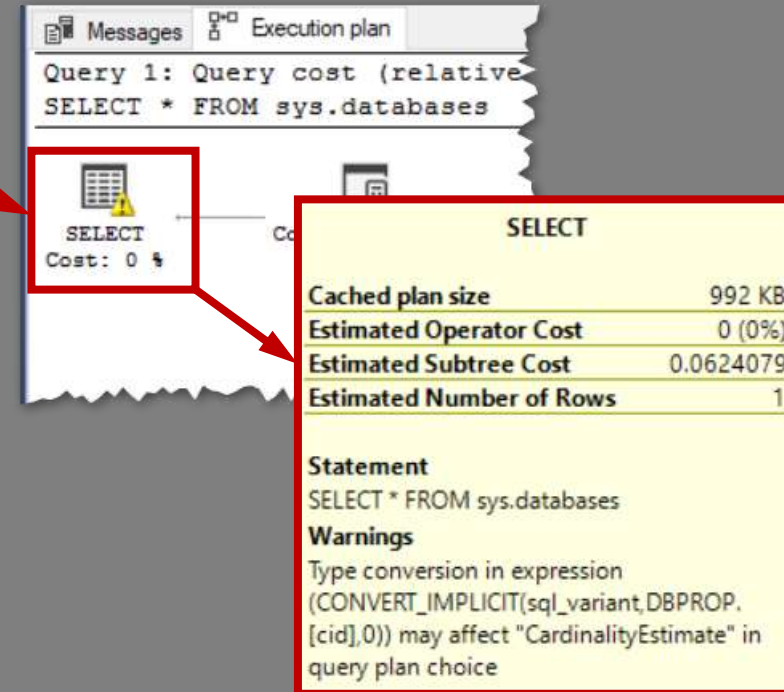
- Icons are “Operators”
- Ignore Select Operator at far left
- Arrows indicate data flow direction
- Thickness of arrows indicates relative amount of data flowing
- Cost percent indicates relative cost to the query for given operator
- As much of the Query SQL as possible is displayed



How to Read a Query Execution Plan

Object ToolTip Details:

- Hover over 1st Operator “SELECT”
 - Up pops ToolTip box with details
 - Shows:
 - Cached plan size (memory used)
 - Estimated Operator Cost (same % in flow chart)
 - Estimated Number of Rows
 - SQL Statement



Messages Execution plan

Query 1: Query cost (relative
SELECT * FROM sys.databases

SELECT
Cost: 0 %

SELECT

Cached plan size	992 KB
Estimated Operator Cost	0 (0%)
Estimated Subtree Cost	0.0624079
Estimated Number of Rows	1

Statement
SELECT * FROM sys.databases

Warnings
Type conversion in expression
(CONVERT_IMPLICIT(sql_variant,DBPROP.
[cid],0)) may affect "CardinalityEstimate" in
query plan choice

How to Read a Query Execution Plan

Object ToolTip Details:

- Hover over 1st Operator “SELECT”
- Hover over 2nd Operator “Compute...”
 - Notice diff details for diff object
 - Shows
 - Same: Estimated Number of Rows
 - Diff: Multiple estimated costs

Query 1: Query cost (relative)
SELECT * FROM sys.databases

SELECT
Cost: 0 %

Compute Scalar
Cost: 0 %

Compute Scalar
Compute new values from existing values in a row.

Physical Operation	Compute Scalar
Logical Operation	Compute Scalar
Estimated Execution Mode	Row
Estimated Operator Cost	0.0000001 (0%)
Estimated I/O Cost	0
Estimated Subtree Cost	0.0624079
Estimated CPU Cost	0.0000001
Estimated Number of Executions	1
Estimated Number of Rows	1
Estimated Row Size	1176 B
Node ID	0

Output List
[master].[sys].[sysdbreg].id, [master].[sys].

How to Read a Query Execution Plan

Object ToolTip Details:

- Hover over 1st Operator “SELECT”
- Hover over 2nd Operator “Compute...”
- You can Right-Click > Properties too
 - More information that ToolTips

The screenshot displays the SQL Server Enterprise Manager interface. At the top, the 'Messages' and 'Execution plan' tabs are visible. The 'Execution plan' tab shows a query: 'Query 1: Query cost (relative)' and the SQL statement 'SELECT * FROM sys.databases'. Below the query, the execution plan is shown with two operators: 'SELECT' and 'Compute Scalar'. The 'Compute Scalar' operator is highlighted with a red box. A red arrow points from the 'Compute Scalar' operator to the 'Properties' window. The 'Properties' window is titled 'Compute Scalar' and shows various properties for the operator. The 'Misc' section is expanded, showing a table of properties.

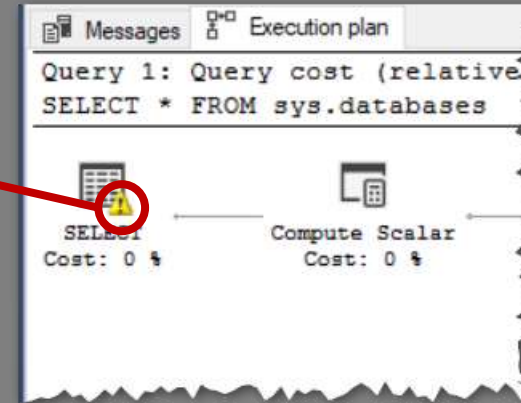
Misc	
Defined Values	[Expr1020] = Scalar Operator
Description	Compute new values from e
Estimated CPU Cost	0.0000001
Estimated Execution M	Row
Estimated I/O Cost	0
Estimated Number of E	1
Estimated Number of R	1
Estimated Operator Co	0.0000001 (0%)
Estimated Rebinds	0
Estimated Rewinds	0
Estimated Row Size	1176 B
Estimated Subtree Cost	0.0624079
Logical Operation	Compute Scalar
Node ID	0
Output List	[master].[sys].[sysdbrg].id

How to Analyze MSSQL Query Execution Plans

- What is a Query Execution Plan?
- How to Read a Query Execution Plan
- **How to Identify Warnings**
- How to Identify the Slowest Operator
- How to Identify and Fix “Table Scans” and “Clustered Index Scans”
- How to Identify and Fix “Spools” and “Sorts”
- How to Identify and Fix “Missing or Stale Statistics”
- How to Identify Common Issues Automatically

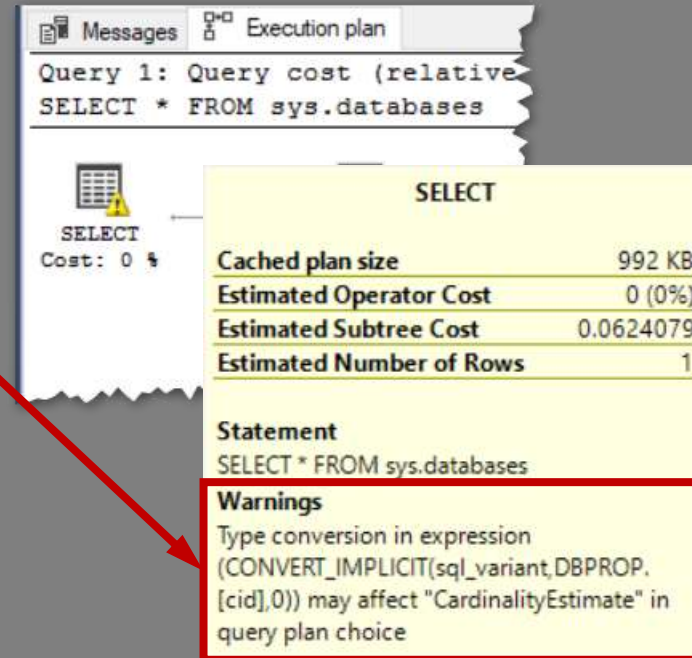
How to Identify Warnings

- Run Estimated or Actual Plan
- Check for Warning Sign @Operator



How to Identify Warnings

- Run Estimated or Actual Plan
- Check for Warning Sign @Operator
 - Check Operator's ToolTips for Details
 - Ex: Implicit Type Conversion force tblscan
 - Ex: TempDB to spill data during execution
 - ...and so on...too many to enumerate



The screenshot shows a SQL Server Enterprise Manager interface. At the top, there are tabs for 'Messages' and 'Execution plan'. Below the tabs, the text 'Query 1: Query cost (relative)' and 'SELECT * FROM sys.databases' is visible. A yellow tooltip is displayed over a 'SELECT' operator in the query plan. The tooltip contains the following information:

SELECT	
Cached plan size	992 KB
Estimated Operator Cost	0 (0%)
Estimated Subtree Cost	0.0624079
Estimated Number of Rows	1

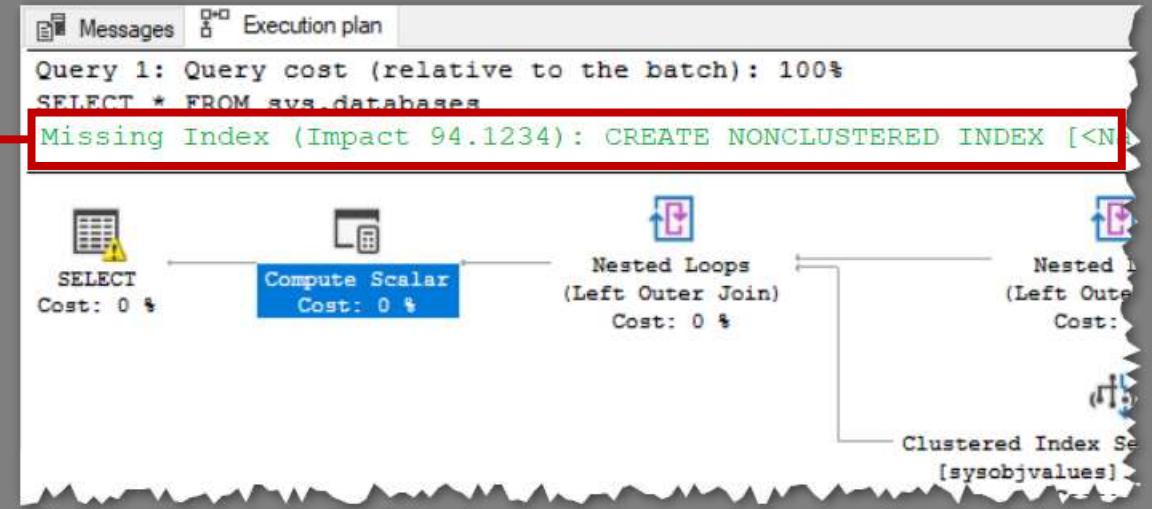
Below the table, the 'Statement' is shown as 'SELECT * FROM sys.databases'. At the bottom of the tooltip, a section titled 'Warnings' is highlighted with a red border. It contains the following text:

Warnings
Type conversion in expression
(`CONVERT_IMPLICIT(sql_variant,DBPROP,[cid],0)`) may affect "CardinalityEstimate" in query plan choice

A red arrow points from the 'Warnings' section of the tooltip to the 'Check Operator's ToolTips for Details' bullet point in the list on the left.

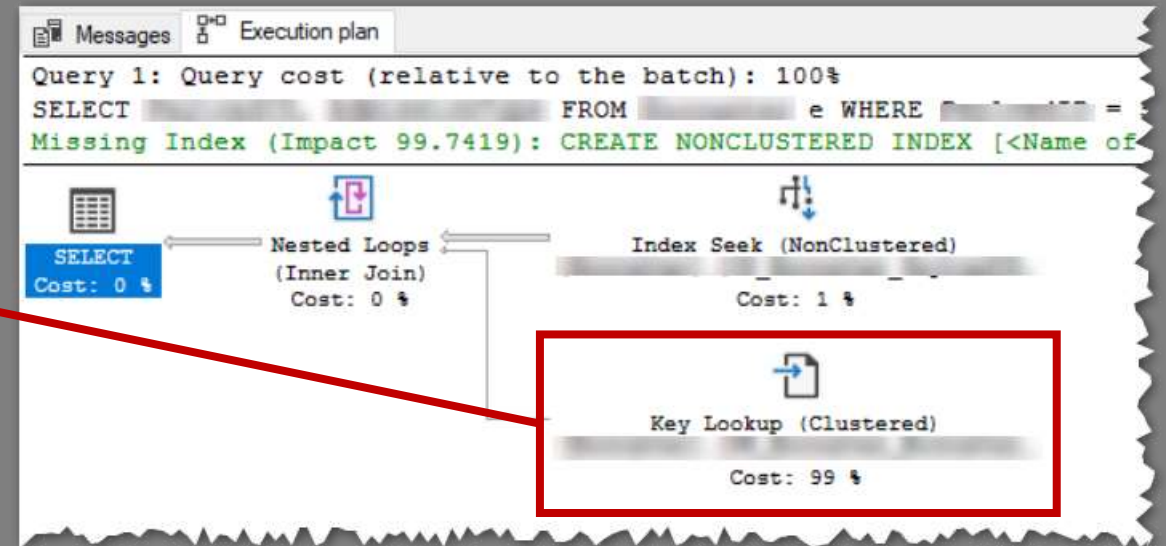
How to Identify Warnings

- Run Estimated or Actual Plan
- Check for Warning Sign @Operator
- Check for Missing Index Warning



How to Identify Warnings

- Run Estimated or Actual Plan
- Check for Warning Sign @Operator
- Check for Missing Index Warning
 - Note that this Warning precludes the need to watch for “Lookups” where an index should have fields added to avoid looking back at table to fetch additional field values

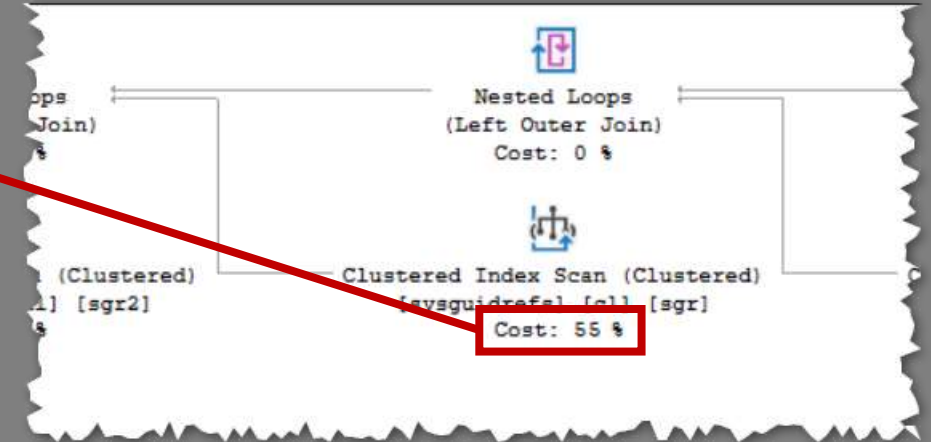


How to Analyze MSSQL Query Execution Plans

- What is a Query Execution Plan?
- How to Read a Query Execution Plan
- How to Identify Warnings
- **How to Identify the Slowest Operator**
- How to Identify and Fix “Table Scans” and “Clustered Index Scans”
- How to Identify and Fix “Spools” and “Sorts”
- How to Identify and Fix “Missing or Stale Statistics”
- How to Identify Common Issues Automatically

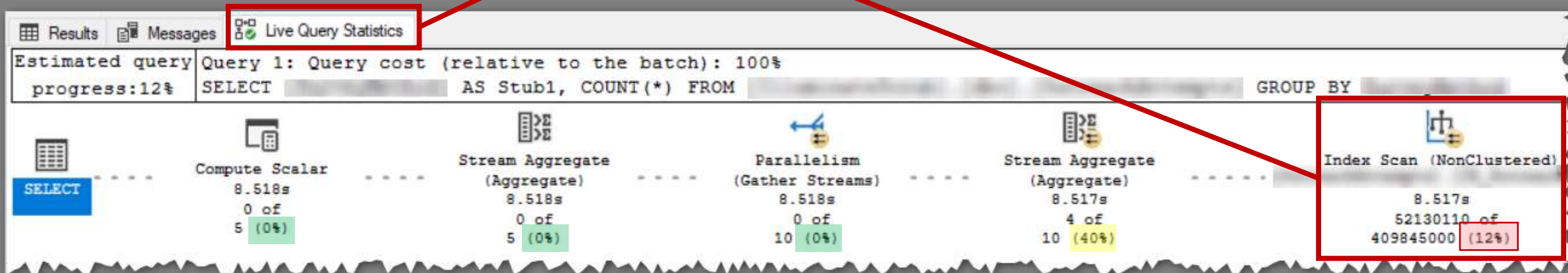
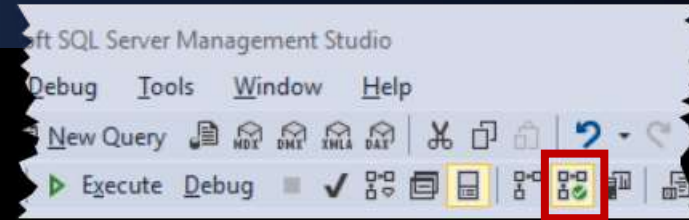
How to Identify the Slowest “Operator”

- Run Estimated or Actual Plan
- Find Biggest “Percent Cost” in Flow
 - Then optimize that Operator’s SQL
 - Repeat until all the biggest costs (%) have been optimized
 - This works well when query execution time is < a minute or so



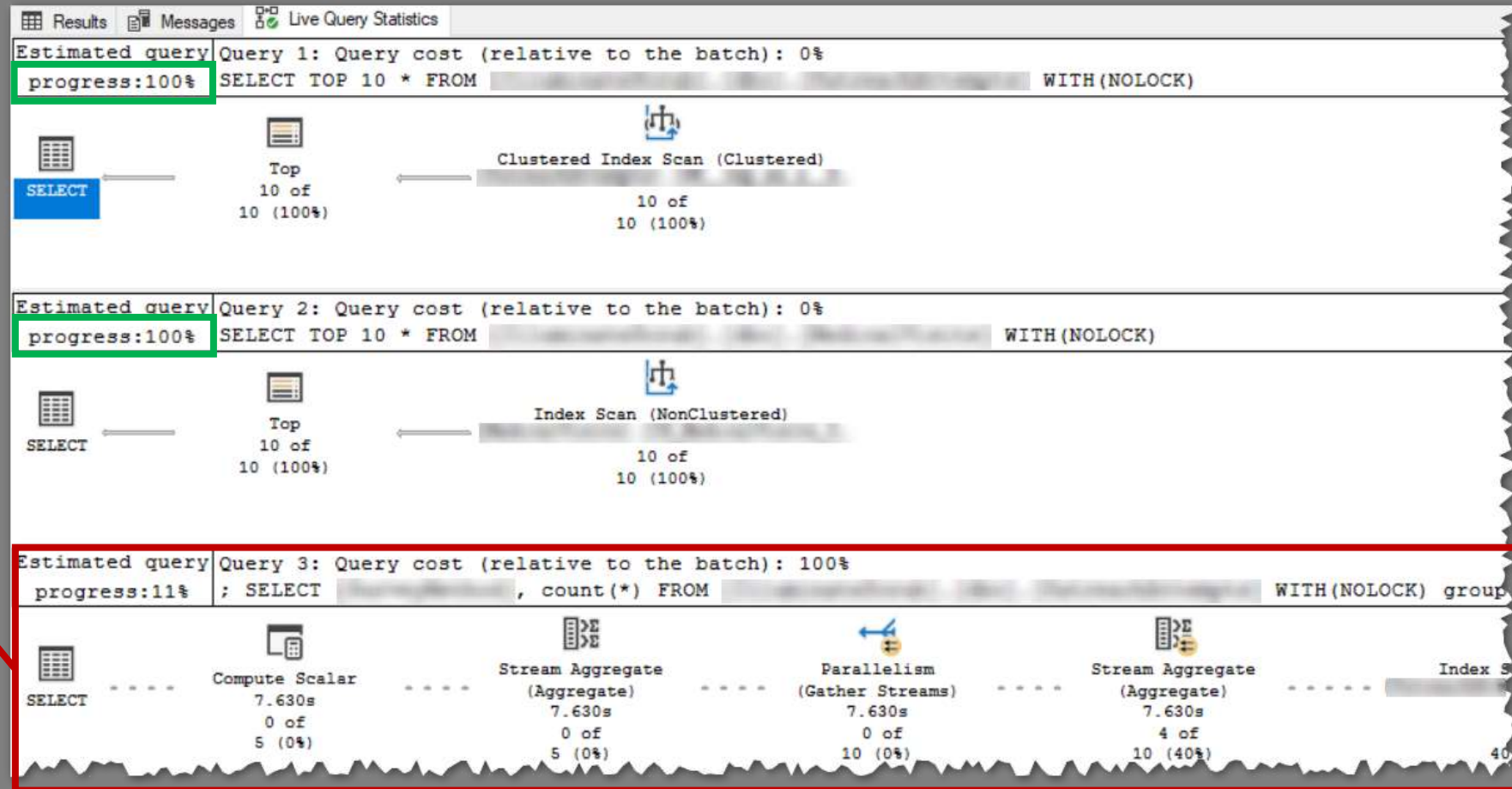
How to Identify the Slowest “Operator”

- Run Estimated or Actual Plan
- Find Biggest Percent Cost
- Tick “Include with Live Query Stats”
 - For Long Running Queries so you don’t have to wait...
 - You can see which operator is running slow in real-time (it’s % cost will move slow)



How to Identify the Slowest “Query in Script”

In large multi-query scripts, I use the same “Live Query Statistics” trick to identify which query is stalling



How to Analyze MSSQL Query Execution Plans

- What is a Query Execution Plan?
- How to Read a Query Execution Plan
- How to Identify Warnings
- How to Identify the Slowest Operator
- **How to Identify and Fix “Table Scans” and “Clustered Index Scans”**
- How to Identify and Fix “Spools” and “Sorts”
- How to Identify and Fix “Missing or Stale Statistics”
- How to Identify Common Issues Automatically

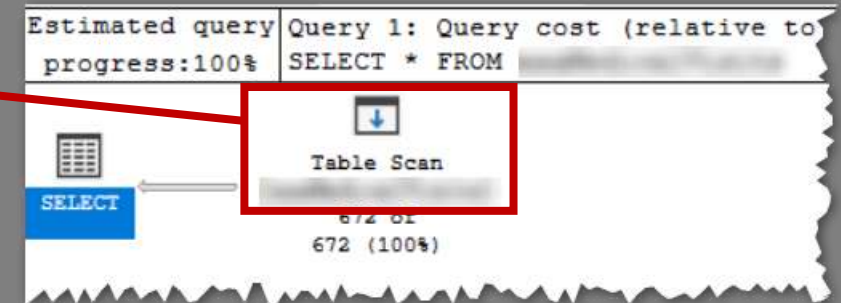
How to Identify and Fix “Scans”

Important Note:

“Much of the time, a ‘Table Scan’ or ‘Clustered Index Scan’ is okay and cannot be further optimized. Do not assume that seeing one is a red flag. You must analyze to be sure .”

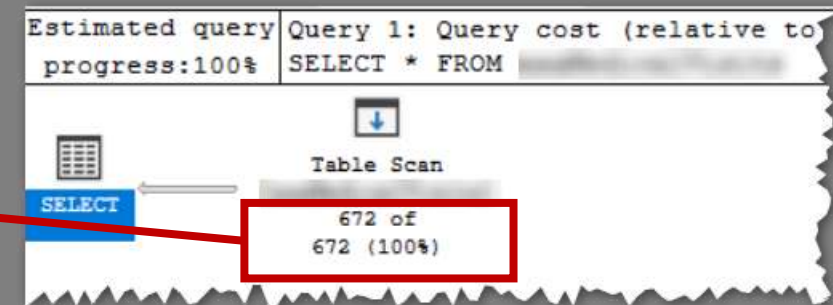
How to Identify and Fix “Table Scans”

- Run Actual Plan
- Find “Table Scan” operator
 - Labeled “Table Scan”
 - Has a table icon with blue down arrow
 - Table name is blurred out here



How to Identify and Fix “Table Scans”

- Run Actual Plan
- Find “Table Scan” operator
- Fix “Table Scan”??
 - It depends...
 - **Ignore** for small tables
 - Having few rows
 - Having few columns and < 100,000 rows
 - **Resolve** for larger tables
 - You’ll know because % cost is high, and time slow



How to Identify and Fix “Table Scans”

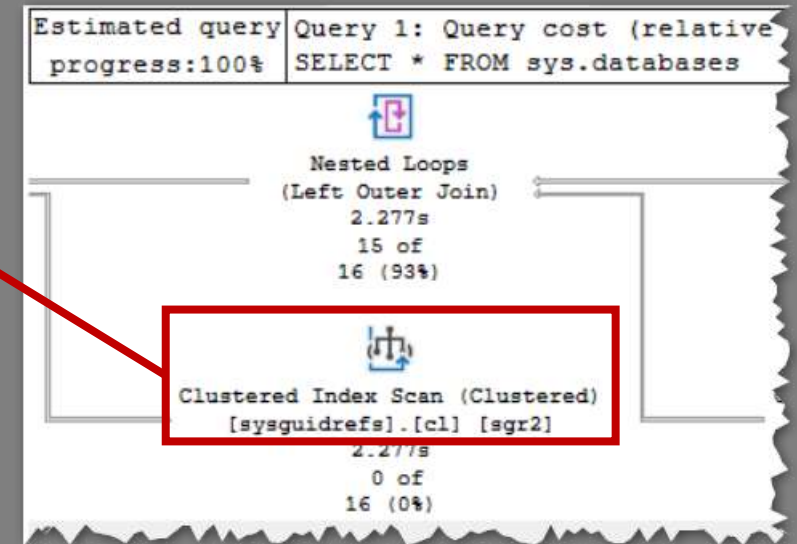
- Run Actual Plan
- Find “Table Scan” operator
- Fix “Table Scan”??
- How to fix “Table Scan”
 1. Create a **Clustered Index** for PKey
 2. Add **Where clause** to return fewer rows

“WHERE” clause

“Clustered Index”

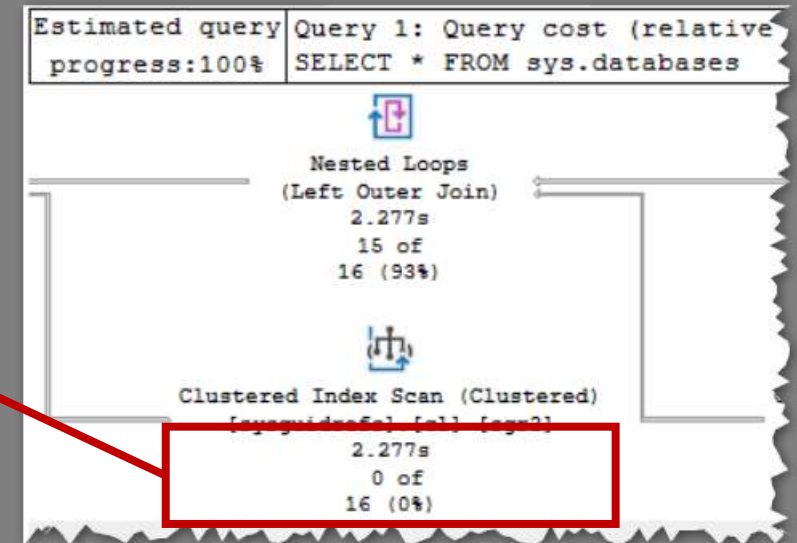
How to Identify and Fix “Clustered Index Scans”

- Run Actual Plan
- Find “Clustered Index Scan” operator
 - Labeled “Clustered Index Scan”



How to Identify and Fix “Clustered Index Scans”

- Run Actual Plan
- Find “Clustered Index Scan” operator
- Fix “Clustered Index Scan”??
 - It depends...
 - **Ignore** for small tables
 - Having few rows
 - Having few columns and < 100,000 rows
 - **Resolve** for larger tables
 - You’ll know because % cost is high, and time slow



How to Identify and Fix “Clustered Index Scans”

- Run Actual Plan
- Find “Clustered Index Scan” operator
- Fix “Clustered Index Scan” ??
- How to fix “Clustered Index Scan”
 1. Create or Fix **Non-Clustered Index** to include extra fields

“Non-Clustered Index”

How to Analyze MSSQL Query Execution Plans

- What is a Query Execution Plan?
- How to Read a Query Execution Plan
- How to Identify Warnings
- How to Identify the Slowest Operator
- How to Identify and Fix “Table Scans” and “Clustered Index Scans”
- **How to Identify and Fix “Spools” and “Sorts”**
- How to Identify and Fix “Missing or Stale Statistics”
- How to Identify Common Issues Automatically

How to Identify and Fix “Spools”

- What are Spool Operators?
 - Look for 3 Types of Spool Operators
 - Act like a “cache” in query processor



Index Spool
(Eager Spool)



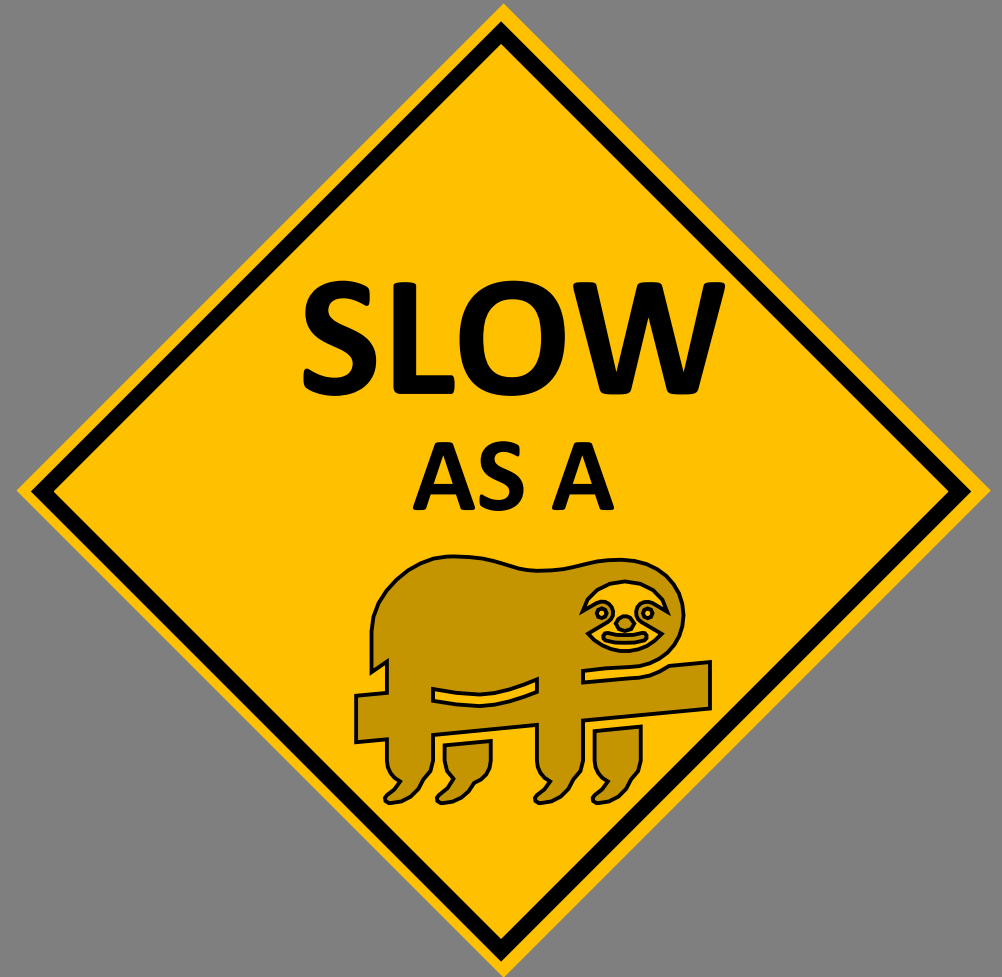
Table Spool
(Eager Spool)



Table Spool
(Lazy Spool)

How to Identify and Fix “Spools”

- What are Spool Operators?
- Why are Spool Operators Bad?
 - Implemented as tempdb tables
 - Impacts performance



How to Identify and Fix “Spools”

- What are Spool Operators?
- Why are Spool Operators Bad?
- How to Fix?
 - Add DISTINCT to query
 - Re-arrange SQL Logic
 - See Adam Machanic’s Presentation



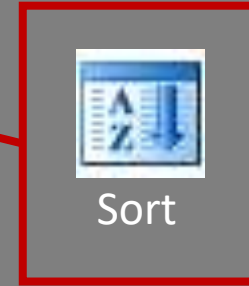
<https://youtu.be/bS0q1nBP3As>

Query Plan Analysis:
5 Culprits That Cause 95% of
Your Performance Headaches

Adam Machanic
@AdamMachanic
Data Education

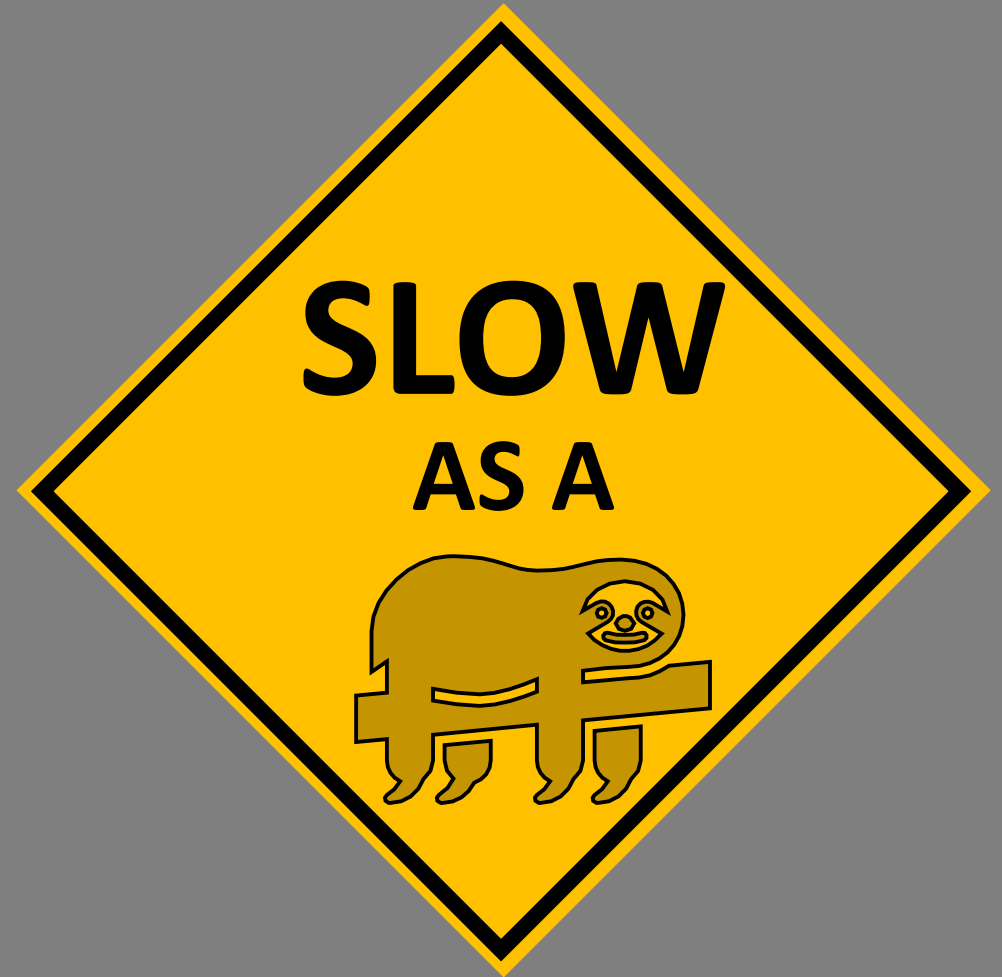
How to Identify and Fix “Sorts”

- What are Sort Operators?
 - ORDER BY
 - Merge Join, Stream Agg, Windowing



How to Identify and Fix “Sorts”

- What are Sort Operators?
- Why are Sort Operators Bad?
 - Slow!
 - Performs worse as size increases



How to Identify and Fix “Spools”

- What are Spool Operators?
- Why are Spool Operators Bad?
- How to Fix?
 - Remove it
 - ...Unless you really do need it

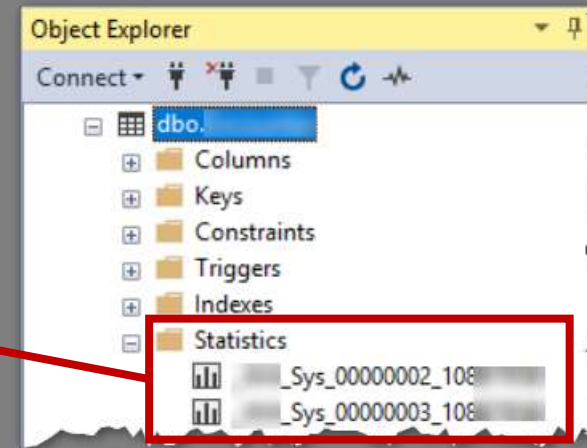


How to Analyze MSSQL Query Execution Plans

- What is a Query Execution Plan?
- How to Read a Query Execution Plan
- How to Identify Warnings
- How to Identify the Slowest Operator
- How to Identify and Fix “Table Scans” and “Clustered Index Scans”
- How to Identify and Fix “Spools” and “Sorts”
- **How to Identify and Fix “Missing or Stale Statistics”**
- How to Identify Common Issues Automatically

How to Identify and Fix “Missing or Stale Statistics”

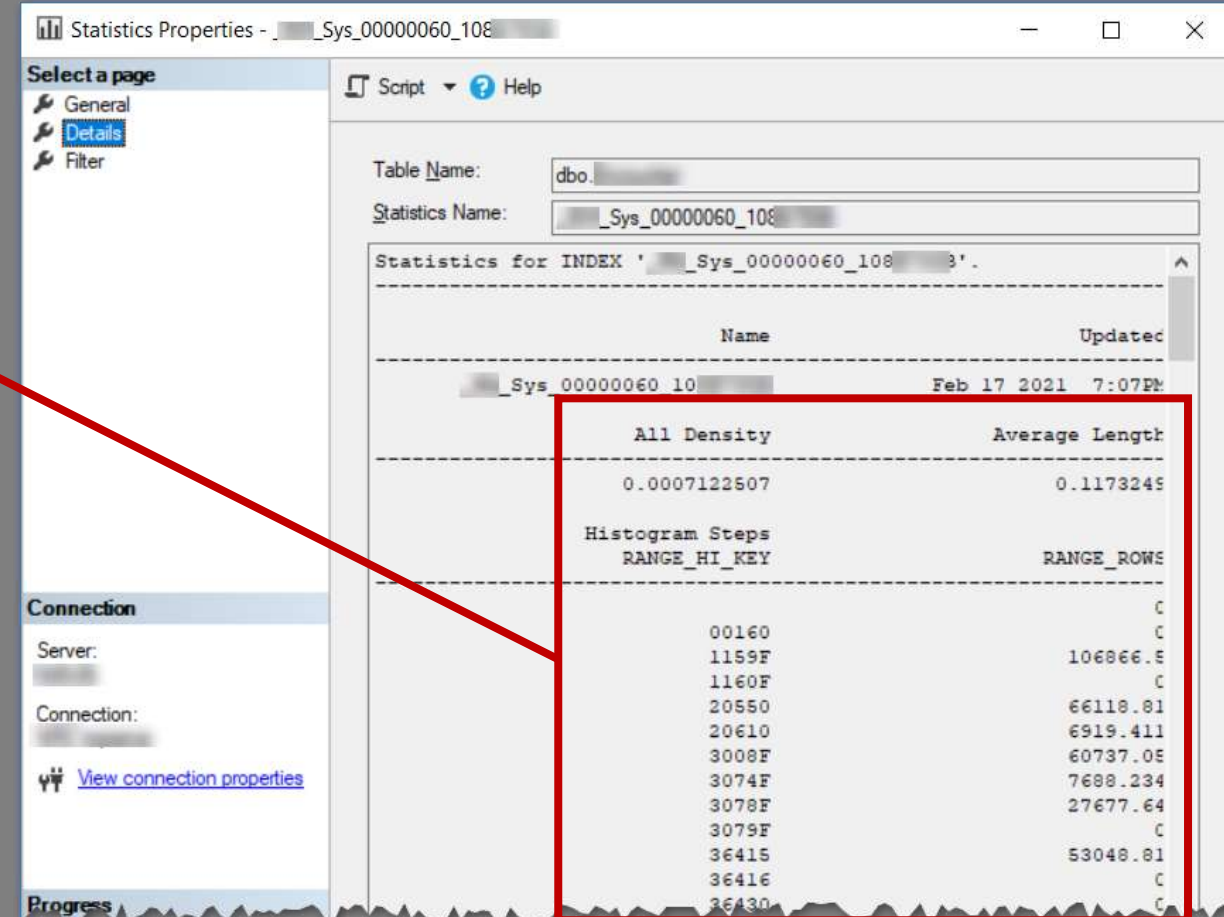
- **What** are “Statistics”
 - In SSMS, each Table Object has a folder of Statistics



How to Identify and Fix “Missing or Stale Statistics”

■ What are “Statistics”

- In SSMS, each Table Object has a folder of Statistics
- Statistics = frequency / distribution of values for a given column
- It is an estimate for the number of rows having a given value
- It is also known as “Cardinality”



Statistics Properties - _Sys_00000060_108

Select a page: General Details Filter

Table Name: dbo. [Table Name]

Statistics Name: _Sys_00000060_108

Statistics for INDEX ' _Sys_00000060_108_3 '.

Name	Updated
_Sys_00000060_108	Feb 17 2021 7:07PM

All Density	Average Length
0.0007122607	0.1173249

Histogram Steps	RANGE_HI_KEY	RANGE_ROWS
		C
	00160	C
	1159F	106866.6
	1160F	C
	20550	66118.81
	20610	6919.411
	3008F	60737.08
	3074F	7688.234
	3078F	27677.64
	3079F	C
	36415	53048.81
	36416	C
	36430	C

How to Identify and Fix “Missing or Stale Statistics”

- What are “Statistics”
- **Why** do “Statistics” Matter?
 - Query Optimizer uses Statistics to estimate rows returned by Predicates
 - Predicate = logic expression in Where
 - **Problem**: If Statistics are missing or inaccurate, then optimizer will often **table scan**! (1,000x slower than idx seek, so WHERE, JOIN, and GROUP BY clauses all slower)

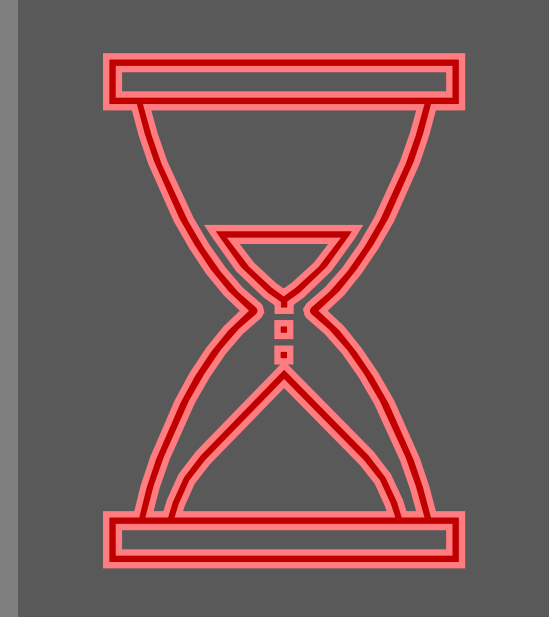


Table Scanning...

How to Identify and Fix “Missing or Stale Statistics”

- What are “Statistics”
- Why do “Statistics” Matter?
- How to Spot Inaccurate “Statistics”
 - Right-click operators to view on the actual Query Execution Plan
 - When “Estimated Number Rows” is far larger than “Actual Number Rows”

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 Execution Mode	Row
Estimated Execution Mode	Row
Storage	RowStore
Actual Number of Rows	1
Actual Number of Batches	0
Estimated I/O Cost	0.003125
Estimated Operator Cost	0.0032831 (5%)
Estimated Subtree Cost	0.0032831
Estimated CPU Cost	0.0001581
Estimated Number of Executions	1
Number of Executions	15
Estimated Number of Rows	283132996
Estimated Number of Rows to be Read	1
Estimated Row Size	4027 B
Actual Rebinds	0
Actual Rewinds	0

How to Identify and Fix “Missing or Stale Statistics”

- What are “Statistics”
- Why do “Statistics” Matter?
- How to Spot Inaccurate “Statistics”
- **Nested Loops Too!**
 - From Adam Machanic: “Have you ever come in the next day and a query always takes 10 seconds all the sudden takes 12 hours?”
 - “It due to stale statistics”
 - “Signature is Nested Loop operators in Joins”

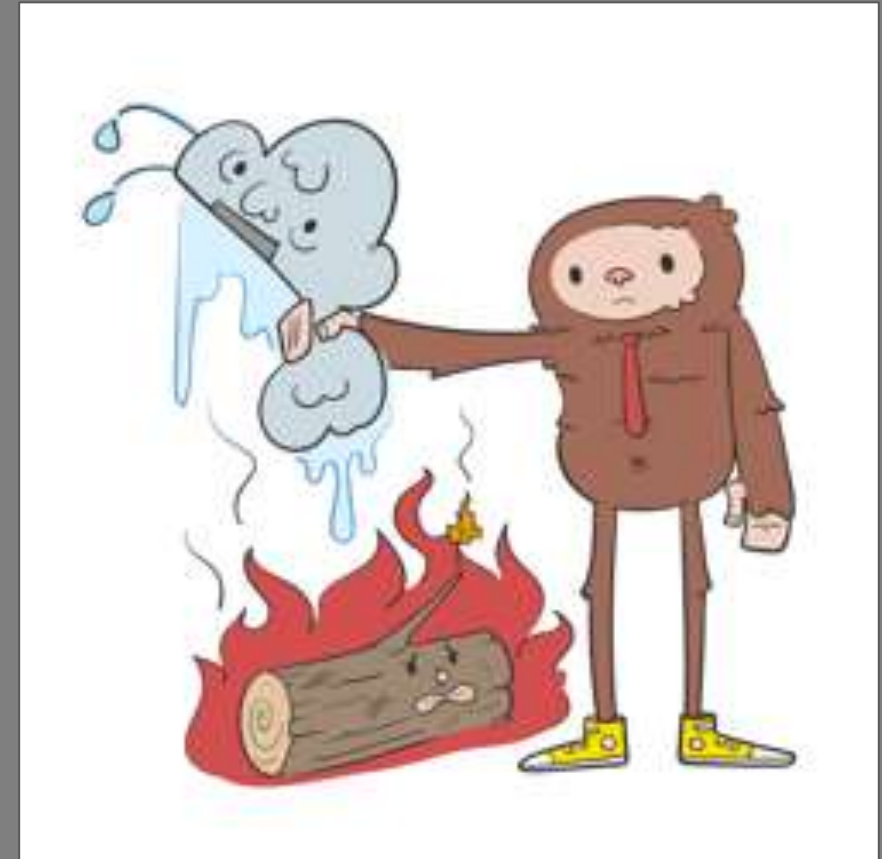


Nested Loops

How to Identify and Fix “Missing or Stale Statistics”

- What are “Statistics”
- Why do “Statistics” Matter?
- How to Spot Inaccurate “Statistics”
- Nested Loops Too!
- How to Fix Inaccurate “Statistics”
 - “Auto Update Statistics” set to True
 - DBA’s schedule maintenance jobs to run “UPDATE STATISTICS”
 - Sometimes = complex SQL JOIN logic

See “sqlperformance.com” link in YouTube description below for more fixes



How to Analyze MSSQL Query Execution Plans

- What is a Query Execution Plan?
- How to Read a Query Execution Plan
- How to Identify Warnings
- How to Identify the Slowest Operator
- How to Identify and Fix “Table Scans” and “Clustered Index Scans”
- How to Identify and Fix “Spools” and “Sorts”
- How to Identify and Fix “Missing or Stale Statistics”
- **How to Identify Common Issues Automatically**

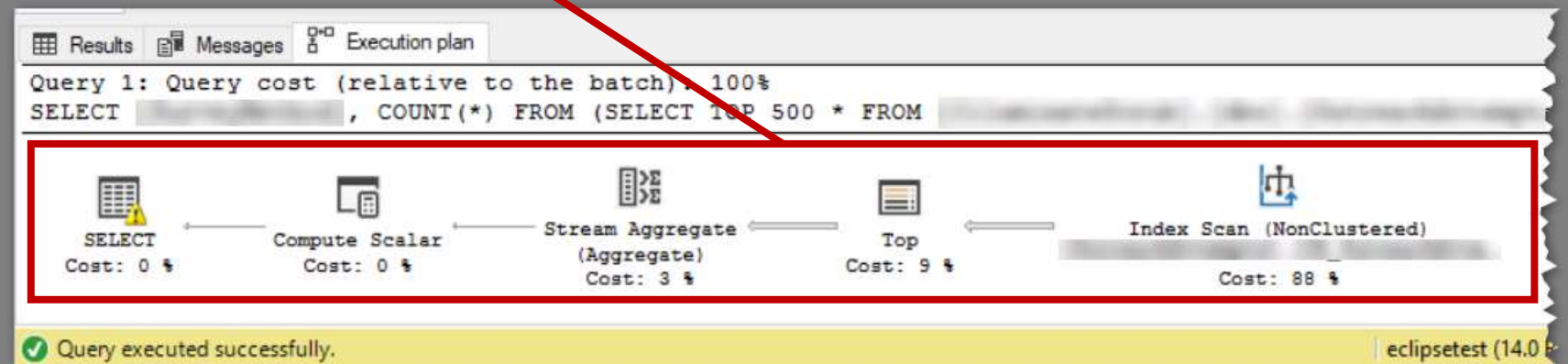
How to Identify Common Issues Automatically

And if all else fails...

“SQL Server has a built-in Analyzer feature that will do the work for you!”

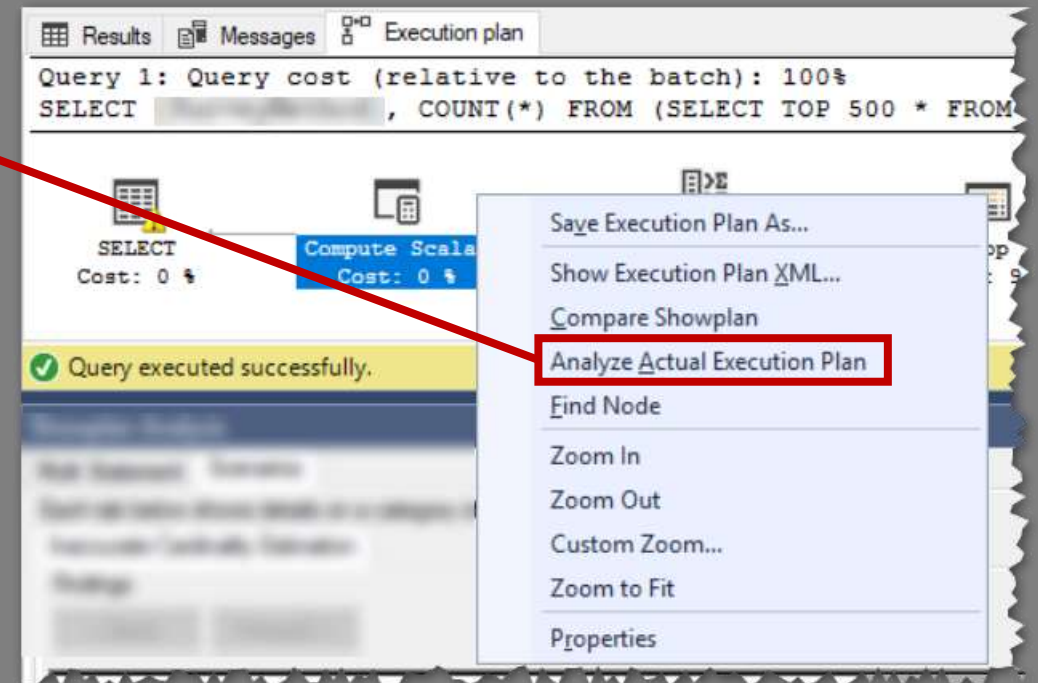
How to Identify Common Issues Automatically

- Run Actual Plan
- Right-Click Whitespace of Exec Plan



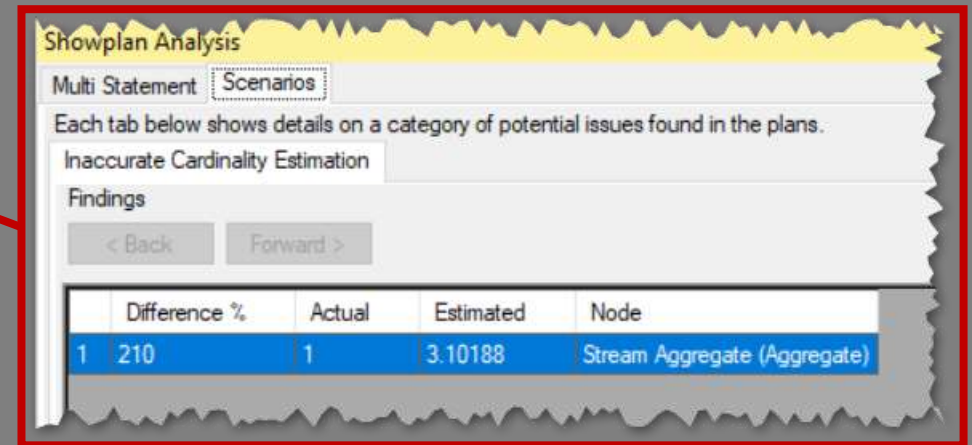
How to Identify Common Issues Automatically

- Run Actual Plan
- Right-Click Whitespace of Exec Plan
- Select “Analyze Actual Exec Plan”



How to Identify Common Issues Automatically

- Run Actual Plan
- Right-Click Whitespace of Exec Plan
- Select “Analyze Actual Exec Plan”
- Pops Up “Showplan Analysis”
 - MSSQL did the Analysis Work
 - Found One Issue in Blue
 - Can find multiple issues



How to Identify Common Issues Automatically

- Run Actual Plan
- Right-Click Whitespace of Exec Plan
- Select “Analyze Actual Exec Plan”
- Pops Up “Showplan Analysis”
- Click “Finding Details” Link
 - ...to Popup “Scenario Explanation”
 - So you have all the details you need to identify and fix issues

