

# Uncovering the hidden gems in Execution plans

Kevan Riley



#269 | EXETER 2014

# Who is this guy?

---

- Kevan Riley
- Independent freelance SQL Server
- 15+ years SQL Server
- Moderator at [ask.sqlservercentral.com](http://ask.sqlservercentral.com)



# What's it all about?

---

- Query performance tuning
- Execution plan analysis
- XML
- Some show in graphical
- Don't always get full picture

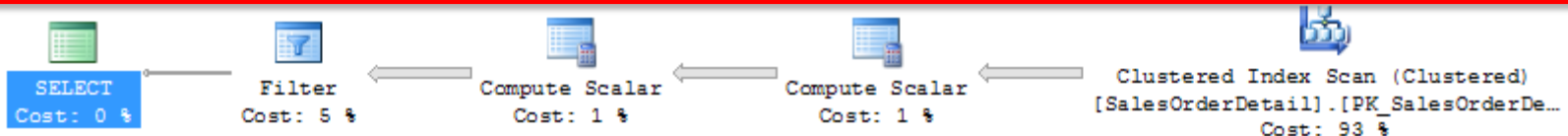
# Missing Indexes

- Query level warning
- 2005 'hidden' in XML
- 2008+ Graphical execution plan

Query 1: Query cost (relative to the batch): 100%

~~select \* from Sales.SalesOrderDetail where CarrierTrackingNumber = 'E101-4649-85'~~

Missing Index (Impact 99.6587): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON



# Missing Indexes

---

SSMS Demo

# Missing Indexes

```
<MissingIndexes>
  <MissingIndexGroup Impact="99.6587">
    <MissingIndex Database="[AdventureWorks2008R2]" Schema="[Sales]" Table="[SalesOrderDetail]">
      <ColumnGroup Usage="EQUALITY">
        <Column Name="[CarrierTrackingNumber]" ColumnId="3" />
      </ColumnGroup>
    </MissingIndex>
  </MissingIndexGroup>
</MissingIndexes>
```

- Every statement in multi-statement batch
- Only one missing index

# Reason for Early Termination

---

- Plan level attribute
- No graphical indication / No tooltip info
- Available in Properties / XML
- 3 values
  - Good enough plan found
  - Timeout
  - Memory limit exceeded

# Reason for Early Termination

---

SSMS Demo



# Reason for Early Termination

---

```
<StmtSimple  
  StatementOptmLevel="FULL" StatementOptmEarlyAbortReason="TimeOut"
```

- Full optimization level
- Timeout – complex query

# No Join Predicate Warning

- Operator level warning



Nested Loops  
(Inner Join)  
Cost: 0 %

## Nested Loops

For each row in the top (outer) input, scan the bottom (inner) input, and output matching rows.

Physical Operation	Nested Loops
Logical Operation	Inner Join
Estimated Execution Mode	Row
Estimated Operator Cost	0.0000295 (0%)
Estimated I/O Cost	0
Estimated Subtree Cost	0.0066179
Estimated CPU Cost	272.823
Estimated Number of Executions	1
Estimated Number of Rows	10
Estimated Row Size	11 B
Node ID	1

## Output List

[AdventureWorks2008R2].[Sales].  
[SalesOrderDetail].SalesOrderID

## Warnings

No Join Predicate

# No Join Predicate Warning

---

SSMS Demo

# No Join Predicate Warning

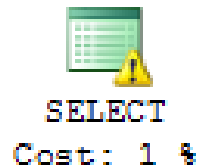
---

```
<Warnings NoJoinPredicate="true" />
```

- Cartesian Product
  - Cross Join
  - 'old' join syntax
- Incomplete predicate
- Complex query (possibly with Timeout)
- Complex predicate

# Cardinality Estimate

- Query level warning
- SQL 2012



SELECT	
Cached plan size	8 KB
Estimated Operator Cost	0.003146 (1%)
Estimated Subtree Cost	0.545484
Estimated Number of Rows	9439.5

## Statement

```
SELECT  
SalesOrderID, OrderDate  
from Sales.SalesOrderHeader  
where convert(datetime,convert  
(varchar,orderdate,112))  
>= convert(datetime,convert (varchar,dateadd  
(mm,datediff(mm,0,'1 jan 2006'),0),112))
```

## Warnings

Type conversion in expression (CONVERT (varchar(30),[AdventureWorks2008R2].[Sales].[SalesOrderHeader].[OrderDate],112)) may affect "CardinalityEstimate" in query plan choice

# Cardinality Estimate

---

SSMS Demo

# Cardinality Estimate

---

```
<Warnings>  
  <PlanAffectingConvert ConvertIssue="Cardinality Estimate"  
    Expression="CONVERT(char(10),[soh].[SalesOrderID],0)" />  
</Warnings>
```

- Potential tuning opportunities
- All conversions

# Sort Warnings

- Operator level warning
- Introduced SQL 2012



Sort	
Sort the input.	
Physical Operation	Sort
Logical Operation	Sort
Actual Execution Mode	Row
Estimated Execution Mode	Row
Actual Number of Rows	31465
Actual Number of Batches	0
Estimated Operator Cost	0.535306 (49%)
Estimated I/O Cost	0.0112613
Estimated CPU Cost	0.524047
Estimated Subtree Cost	1.09904
Estimated Number of Executions	1
Number of Executions	1
Estimated Number of Rows	9439.5
Estimated Row Size	341 B
Actual Rebinds	1
Actual Rewinds	0
Node ID	0
<b>Warnings</b>	
Operator used tempdb to spill data during execution with spill level 1	
<b>Order By</b>	
[AdventureWorks2008R2].[Sales]. [SalesOrderHeader].OrderDate Ascending	



# Sort Warnings

---

SSMS Demo

# Sort Warnings

---

```
<Warnings>  
  <SpillToTempDb SpillLevel="1" />  
</Warnings>
```

- Bad stats, bad estimates
- Certain query patterns
- Parameter sniffing
- Spill Level
  - 1 = single pass
  - n = n multiple passes

# Retrieved from cache

---

- ~~Was plan retrieved from cache?~~
- Was the plan cached?
  - Plan was in cache => True
  - Generated plan now in cache => True
  - Generated plan not in cache => False

# Retrieved from cache

---

```
<Statements>  
  <StmtSimple  
    RetrievedFromCache="true">
```

- Plan level attribute
- SQL 2012
- No graphical indication / No tooltip info
- Available in Properties / XML

# Taking this further

---

- sys.dm\_exec\_cached\_plans
- sys.dm\_exec\_query\_plan
- Proactive tuning
- Jason Strate MVP MCM ([www.jasonstrate.com](http://www.jasonstrate.com))
  - Blog series - Can you dig it?

# References

---

- Grant Fritchey [www.scarydba.com](http://www.scarydba.com)
- Joseph Sack [www.sqlskills.com/blogs/joe](http://www.sqlskills.com/blogs/joe)
- Benjamin Nevarez [www.benjaminnevarez.com](http://www.benjaminnevarez.com)
- Jason Strate [www.jasonstrate.com](http://www.jasonstrate.com)

# Questions?

---

kevan.riley@rileywaterhouse.co.uk

www.rileywaterhouse.co.uk

@kevriley

# SQL Saturday Exeter would like to thank all of our sponsors



Microsoft

