

Advanced T-SQL and Troubleshooting

Module 6: Troubleshooting

Bob Beauchemin
BobB@SQLskills.com



Overview

- When to optimize T-SQL code
- Why queries run slowly
- How to find query information
- Deadlocks



2

© SQLskills, All rights reserved.
<http://www.SQLskills.com>



When to Optimize T-SQL Code

- The users will point out when to optimize
- "SQL Server is running slowly"
 - Is it all queries?
 - It is a specific set of queries?
 - It is specific timeframes?
- Or proactive optimization

What's Running Slowly

- If the whole instance is running slowly
 - Data Collection
 - Bottleneck Analysis
 - sys.dm_os_wait_stats
 - DBCC SQLPERF(WAITSTATS)
 - sys.dm_os_waiting_tasks

Which Queries are Running Slowly

- **Types of queries to tune**
 - Frequently executed queries
 - Small improvement is worthwhile if query runs 10000 times/hour
 - Parameterization helps detecting similar queries
 - May need to combine similar queries
 - sys.dm_exec_query_stats (aggregates)
 - Built-in SSMS reports (by CPU, by IO)
 - Heavy resource consumption queries
 - Month-end reports
 - Analysis queries
 - Watch for high response time in profiler
 - Queries that are slow because in lock waits

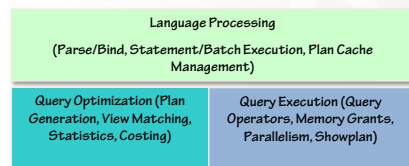
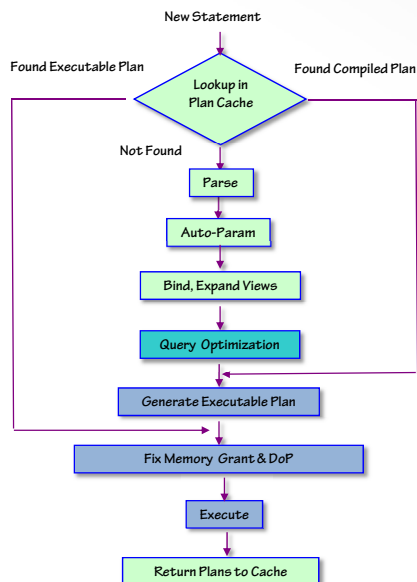
Why Code Runs Slowly

- **Queries run slowly because of**
 - Suboptimal physical structures
 - Missing indexes
 - Lack of covering indexes
 - Lack of index on computed columns
 - Out of date statistics
 - Non-parameterized queries
 - Excessive compilation time
 - Parameter sniffing - too little compilation
 - Overly complex queries
 - cardinality estimates incorrect

Troubleshooting Queries - Tools

- Dynamic Management Views
- SQL Profiler
- Showplan
- Performance Monitor
- Database Tuning Advisor
- SQL Server 2008 Adds
 - Data Collection
 - Extended Events

Statement Execution



Where to Find Query Plan Info

- **SQL Server 2000/2005**
 - Access syscacheobjects
- **SQL Server 2005/2008 – DMVs**
 - sys.dm_exec_cached_plans
 - sys.dm_exec_cached_plan_dependent_objects
 - For list of query plans, execution plans
 - sys.dm_exec_sql_text(sql_handle)
 - For the text of the sql statement executed
 - sys.dm_exec_query_plan(plan_handle)
 - sys.dm_exec_text_query_plan(plan_handle) SP2
 - For the execution plan of the sql statement executed
 - sys.dm_exec_query_stats
 - For a variety of query statistics – like number of executions, plan creation time (first execution into cache), last execution time, etc.
 - sys.dm_exec_plan_attributes(plan_handle)
 - Determine why plan is/is not being reused

Locks – Granularity And Escalation

- **Granularity chosen at compilation time**
 - Row Level/Page Level
 - Table Level
 - Database Level
 - Partition Level using ALTER TABLE (SQL Server 2008)
- **Escalated at execution if resources (to handle all of the smaller locks) are not available**
 - Row to TABLE Level (or partition level)
 - Page to TABLE Level (or partition level)
 - Locks do NOT escalate Row ⇒ Page ⇒ Table
- **Lock escalation is usually desired**
 - Can be globally changed by trace flags
 - 1224 allows escalation under memory pressure
 - 1221 does not allow escalation under memory pressure
 - See KB323630

Controlling Wait Time

- **Check current value**
 - `SELECT @@lock_timeout`
- **Change current value**
 - `SET LOCK_TIMEOUT N`
 - Where N represents the number of MILLISECONDS
 - Great for readers to give up waiting but what if a statement fails within a transaction?
 - You must detect the runtime error (because this is user defined) and determine the fate of the transaction
 - The transaction will be rolled back IF you have `SET XACT_ABORT ON`
- **Skip over locked rows with READPAST**

Troubleshooting Blocking - Tools

- **Dynamic Management Views**
- **System Stored Procedures**
- **SQL Profiler**
- **Performance Monitor Counters**
- **SQL Nexus**
- **Blocked Process Report**
- **Deadlock Events and Trace Flag 1222**
- **SQL Server 2008 adds**
 - Data Collection
 - Extended Events

Deadlocks

- **Two (or more) transactions request mutually desired resources in an undesirable order**
 - Tx1 locks resource A
 - Tx2 locks resource B
 - Tx1 attempts to access resource B
 - Live lock here
 - Tx2 attempts to access resource A
 - Deadlock here
- **Two main patterns**
 - Deadly embrace deadlock (above)
 - Conversion deadlock

Deadlocks and SQL Server

- **All resources have similar deadlock handling**
- **Deadlocks can be caused by**
 - Data locks - locks of rows, pages, tables, and database metadata
 - Worker Threads - a queued tasks waiting for worker thread can cause deadlock (more later)
 - Memory - two tasks waiting for memory
 - Parallel query execution resources
 - MARS interleaving (only if MARS is enabled on client)
 - User resources - thread waiting for call in app
 - Session mutex - blocked by same connection, different session
 - Transaction mutex - tx is per session/not connection

Deadlock Resolution

- **Choose the victim that is first in the chain – moving backwards in time**
 - This is typically the cheaper of the two to kill.
 - Or – choose the victim based on SET DEADLOCK_PRIORITY LOW
 - SQL Server 2005 - set to -10 through 10 or LOW-MEDIUM-HIGH (can also set w/variable)
- **The deadlock victim receives error 1205:**
 - Your transaction (tran id) was deadlocked with another process and has been chosen the deadlock victim. Rerun your transaction.

Avoiding Deadlocks

- **Minimize blocking**
 - More effective queries
 - More effective indexes
 - Add or Remove lock hints
- **Access resources in the same order**
- **MARS connections for client applications**
- **SET DEADLOCK_PRIORITY to lower value**
- **Add retry logic in application if error 1205 returned**
- **Consider versioning isolation levels**

Summary

- **Determine which queries to tune**
 - Everything is slow
 - Frequently used
 - Big resource consumers
- **Be careful of deadlocks**
 - Controlling locks
 - Troubleshooting deadlocks
 - Avoiding deadlocks

Resources

- **Statistics Used by the Query Optimizer in Microsoft SQL Server 2005,** Eric Hanson and Lubor Kollar,
 - <http://www.microsoft.com/technet/prodtechnol/sql/2005/qrystats.mspx>
- **Batch Compilation, Recompilation, and Plan Caching Issues in SQL Server 2005,** Arun Marathe,
 - <http://www.microsoft.com/technet/prodtechnol/sql/2005/recomp.mspx>
- **Troubleshooting stored procedure recompilation**
 - <http://support.microsoft.com/default.aspx?scid=kb;en-us;Q243586>

Resources (Blogs)

- **Tips, Tricks, and Advice from the SQL Server Query Optimizer Team**
 - <http://blogs.msdn.com/programmability>
- **SQL Server Engine Tips**
 - <http://blogs.msdn.com/sqltips>
- **SQL Programmability Team Blog**
 - <http://blogs.msdn.com/sqlqueryprocessing>
- **Craig Freedman's Web Log**
 - <http://blogs.msdn.com/craigfr>



19

© SQLskills, All rights reserved.
<http://www.SQLskills.com>

Questions?

