

- COURSE CODE:** SQSATS
- COURSE TITLE:** SQL Server: Advanced Transact-SQL Tuning via Execution Plans
- CURRENCY:** SQL Server 2005, 2008, 2008 R2
- AUDIENCE:** Experienced T-SQL developers, DBAs and administrators
- PREREQUISITES:** Completion of our course 'SQL Server: Microsoft Transact-SQL for Beginners' or equivalent knowledge.
- DURATION:** 4 days
- SUMMARY:** This presentation details T-SQL tuning and best practices. To these ends, its mainly hands-on approach uses SQL Server's Execution Plans and Database Engine Tuning Advisor (DTA) to optimize index and query selection, including features such as:
- Join optimisation
 - Grouping sets
 - Correlated vs. non-correlated subqueries
 - Common Table Expressions (CTEs)
 - Indexed views
 - Index design and usage
 - Isolation levels
 - Locking and concurrency
 - Elapsed, CPU and I/O times
- OBJECTIVES:** Upon completion of this course, the participant should be able to use SQL Server's Execution Plan and DTA to assist in making efficient T-SQL and index choices for both OLTP and data warehousing environments.
- LAB:** About 65% hands-on, using Execution Plan facilities

1. SQL SERVER OPERATION & ARCHITECTURE – OVERVIEW

- Components
- Memory
- Configuration
- Databases
- Files
- Logging and recovery
- Tables
- Indexes
- T-SQL logical vs. physical processing

2. PLAN CACHING & RECOMPILATION

- Plan caching and Autoparameterization
- Causes of recompilation
- T-SQL hints
- Plan guides
- Plan statistics

3. ALL ABOUT NULLs

- Definition
- Testing for NULLs
- Handling NULLs
- Using COALESCE()
- Using NULLIF
- In arithmetic operations

4. INDEX ORGANISATION & USAGE GUIDELINES

- Non-clustered indexes
- Clustered indexes
- Covering (Include) indexes
- Filtered indexes (SQL Server 2008)
- Indexed views
- Bitmap vs. optimised bitmap filters (SQL Server 2008)
- Index access methods

5. EXECUTION PLANS: USAGE & INTERPRETATION

- Formats (e.g., graphical vs. text)
- Logical vs. physical operators
- Information boxes
- Statistics (e.g., I/O timing)

6. TUNING SELECTS VIA EXECUTION PLANS

- SELECT *
- TOP *n*
- SELECT ... CROSS APPLY
- Column functions (e.g., SUM())
- Scalar functions (e.g., COALESCE)
- ORDER BY
- GROUP BY
- GROUP BY ... HAVING

7. TUNING JOINS VIA EXECUTION PLANS

- Join types (LEFT, RIGHT, FULL, etc.)
- Hash
- Nested loop
- Merge
- Semi-join
- Table expressions

8. TUNING SET OPERATIONS VIA EXECUTION PLANS

- UNION
- UNION vs. CASE
- EXCEPT
- INTERSECT

9. DATETIME QUERIES

- DATETIME storage format
- DATETIME manipulation
- Some advanced uses:
 - ⇒ Nearest birthday based on today's date and individual's birthdate
 - ⇒ How do I identify overlapping sessions so as to track 'chat' sessions, for example?
 - ⇒ How do I use DATEFIRST to find, for example all orders that took place on Thursday?
 - ⇒ Etc.

10. TUNING USING VIEWS

- Views defined
- ORDER BY in a view
- View refreshing
- View modularisation
- Nested views
- Updatable views
- Options (e.g., SCHEMABINDING, CHECK option)
- Common Table Expression (CTE)
- Indexed view
- Inline views, a.k.a. nested tables and derived tables

11. TUNING SUBQUERIES VIA EXECUTION PLANS

- Non-correlated
- Correlated
- EXISTS vs. NOT EXISTS
- IN vs. NOT IN

12. TEMPORARY OBJECTS (E.G., TABLES) OPTIONS

- When to consider
- Local temporary tables
- Tempdb usage
- Scope and visibility considerations
- Transaction considerations
- Distribution statistics considerations
- Caching of temporary objects
- Local vs. global temporary tables
- Table variables vs. temporary tables vs. table expressions
- Minimally logged inserts
- Use of SET CONTEXT-LOG
- TRANSACTION CONTEXT considerations (e.g., statement ROLLBACK)

13. LOCKING & CONCURRENCY

- Reasons for
- Pessimistic vs. optimistic locking
- Transaction concepts via ACID
- Isolation levels
- Lock modes
- Application locks
- Lock duration
- Lock metadata
- Lock compatibility
- Deadlocks
- Row vs. page locks
- Lock escalation
- Row versioning
- Snapshot isolation
- Lock hints

14. BULK OPERATIONS

- BULK INSERT
- BCP (Bulk Copy Program)
- T-SQL OPENROWSET

15. DATABASE ENGINE TUNING ADVISOR (DTA) USAGE

- Exploratory analysis
- Index selection and recommendations
- Analyse proposed changes