

SQL Server Query Tuning

Deepak Rangarajan

deepak.rangarajan@gmail.com

<http://deepakrangarajan.blogspot.com>

Vidhya Sagar

kvs1983@gmail.com

<http://www.sql-articles.com/blogs/>

About Us

- ▣ We are SQL Server MVPs since 2008
- ▣ Hosting www.sql-articles.com website
- ▣ Started Chennai SQL Server User Group (CSSUG)
- ▣ Around 30 members are registered with our UG
- ▣ More details can be found @
 - <http://www.chnsqlug.co.cc>
 - <http://www.twitter.com/chnsqlug>
 - <http://www.facebook.com/group.php?gid=138306012283>

AGENDA

▣ Session Objectives:

- Dive into the query performance and tuning capabilities in SQL Server
- Demonstrate common tuning and troubleshooting techniques using SQL Server
- *Disclaimer: Query Tuning is a deep & broad topic – won't cover everything*

▣ Key Takeaways:

- Get a refresher on certain fundamental concepts
- Common pitfalls to avoid and optimization techniques

Factors Affecting Performance?

- ▣ Application Design
 - 3 tier architecture facilitates caching, connection pooling.
 - Partitioning tables and databases
 - Not mixing logs and database on the same disk
 - Choosing right size and disk for tempdb database
- ▣ Database Design
 - Normalization of database structure, choosing indexes.
- ▣ Microsoft SQL Server Setup.
 - Affinity mask, lightweight pooling, max worker threads and degree of parallelism.
- ▣ Hardware
 - CPU, Disk and Memory

Indexes

- ▣ Scan and Seek *
- ▣ Clustered and non-clustered indexes
- ▣ Seekable Predicates *
 - Single column Indexes
 - Multi Column Indexes
- ▣ Covered Columns
 - A Clustered index covers all columns
 - A non-clustered index covers only key columns
 - A non-clustered index on a table with a clustered index covers the clustered index key

Example : Covered Columns

```
CREATE TABLE T_heap (a int, b int, c int, d int, e int, f int)
CREATE INDEX T_heap_a ON T_heap (a)
CREATE INDEX T_heap_bc ON T_heap (b, c)
CREATE INDEX T_heap_d ON T_heap (d) INCLUDE (e)
CREATE UNIQUE INDEX T_heap_f ON T_heap (f)
```

```
CREATE TABLE T_clu (a int, b int, c int, d int, e int, f int)
CREATE UNIQUE CLUSTERED INDEX T_clu_a ON T_clu (a)
CREATE INDEX T_clu_b ON T_clu (b)
CREATE INDEX T_clu_ac ON T_clu (a, c)
CREATE INDEX T_clu_d ON T_clu (d) INCLUDE (e)
CREATE UNIQUE INDEX T_clu_f ON T_clu (f)
```

Example : Covered Columns Cont'd

Index	Key Columns	Covered Columns
T_heap_a	a	a
T_heap_bc	b, c	b, c
T_heap_d	d	d, e
T_heap_f	f	f
T_clu_a	a	a, b, c, d, e, f
T_clu_b	b, a	a, b
T_clu_ac	a, c	a, c
T_clu_d	d, a	a, d, e
T_clu_f	f	a, f

Indexing – Best Practices

- ❑ Create Useful indexes
- ❑ Choose the clustered index carefully
- ❑ Make Non clustered indexes highly selective
- ❑ Decide on the optimal number of indexes
- ❑ Cover the right columns
- ❑ Choose the right columns to index on
- ❑ Pay attention to column order
- ❑ Index columns used in Joins
- ❑ Create or drop indexes as needed
- ❑ Start using DTA

Plan Caching

- ▣ Caching Mechanisms
 - Adhoc plan caching *
 - Autoparameterization *
 - Forced Parameterization
 - ALTER DATABASE <database_name> SET PARAMETERIZATION FORCED;
 - Use Prepared statements *

Locks and Deadlocks

- ▣ Keep transactions short
- ▣ Avoid external work inside the transaction such as email
- ▣ Take care to avoid deadlocks
- ▣ Use a appropriate isolation level.

Query Hints

- ▣ Use Query hints to influence the kind of plan SQL Server takes
- ▣ Use Query hints with care
- ▣ Review query hints periodically

Best Practices

- ▣ Learn T-SQL
- ▣ Avoid Cursors as far as possible
- ▣ Do you have missing indexes – Take the help of DTA
- ▣ Avoid joining tables on different data types
- ▣ Avoid using functions in where clause
- ▣ Don't use preceding wildcard in LIKE clause
- ▣ Don't use unqualified object names



COMMUNITY
TECHDAYS

Microsoft[®]
Your potential. Our passion.[™]

<http://www.chnsqlug.co.cc>



Microsoft TechNet



Thanks for participating Community Tech Days
Visit us @ www.sql-articles.com or www.chnsqlug.co.cc

COMMUNITY
TECHDAYS

Microsoft[®]
Your potential. Our passion.[™]

<http://www.chnsqlug.co.cc>



Microsoft TechNet