

A Grand Performance

Guerrilla Tactics - Performance Testing
MS SQL Server Applications

Peter Marriott

peter.marriott@catalystcomputing.co.uk

[@peter_marriott](#)

About Me

- Working with RDBMS's since the late 80's
(MS SQLServer, Oracle, Ingres and MySQL)
- Started as software developer
- Spent time as DBA and Systems administrator
- Consultant working with developers to improve application performance

Agenda

- Why Guerrilla Tactics?
- Capturing & Analysing SQL Statements
- Indexing
- Locking & Row versioning
- Questions (at any point)

Why Guerrilla Tactics?

- Ideally
 - Performance is designed in from day one
 - You know the max and median load
 - You have full size hardware for testing
 - You have a full size database
 - You have good monitoring software

Why Guerrilla Tactics?

- No new software on servers
- No software costs
- Not much help/documentation
- Not much time

Capturing & Analysing SQL Statements

- Demo
 - SQLServer Profiler

Capturing & Analysing SQL Statements

- SQL Server Profiler
 - Requires Alter trace privilege
 - Don't remote desktop into the server
 - Do's
 - Pick events carefully
 - Filter as necessary
 - Run on a PC fast enough to take the load
 - Set Application Name in connection string

Capturing & Analysing SQL Statements

- Analysing – SQL Server Profiler output
 - ClearTrace
 - <http://www.scalesql.com/cleartrace/>
 - Roll your own SQL

Indexing

- Demo
 - Database Engine Tuning Advisor

Indexing

- Database Engine Tuning Advisor
 - Requires sysadmin server role
 - Check for similar indexes / statistics
 - Retest!

Locking & Row versioning

- Locking
 - Synchronize multi user access to data
- Row versioning
 - Reader gets previously committed values without blocking

Locking

- Shared – Allow other readers
- Update – Reduce potential deadlocks
- Exclusive – Prevent access to resource
- Intent – Prevent higher level locks occurring

Row versioning

- Read Committed Snapshot
 - Database wide setting
 - Maintain committed data for read duration
- Snapshot isolation
 - Database wide setting but set by session
 - Maintain committed data for transaction duration
 - Errors occur if there are update conflicts

Locking & Row versioning

- Demo

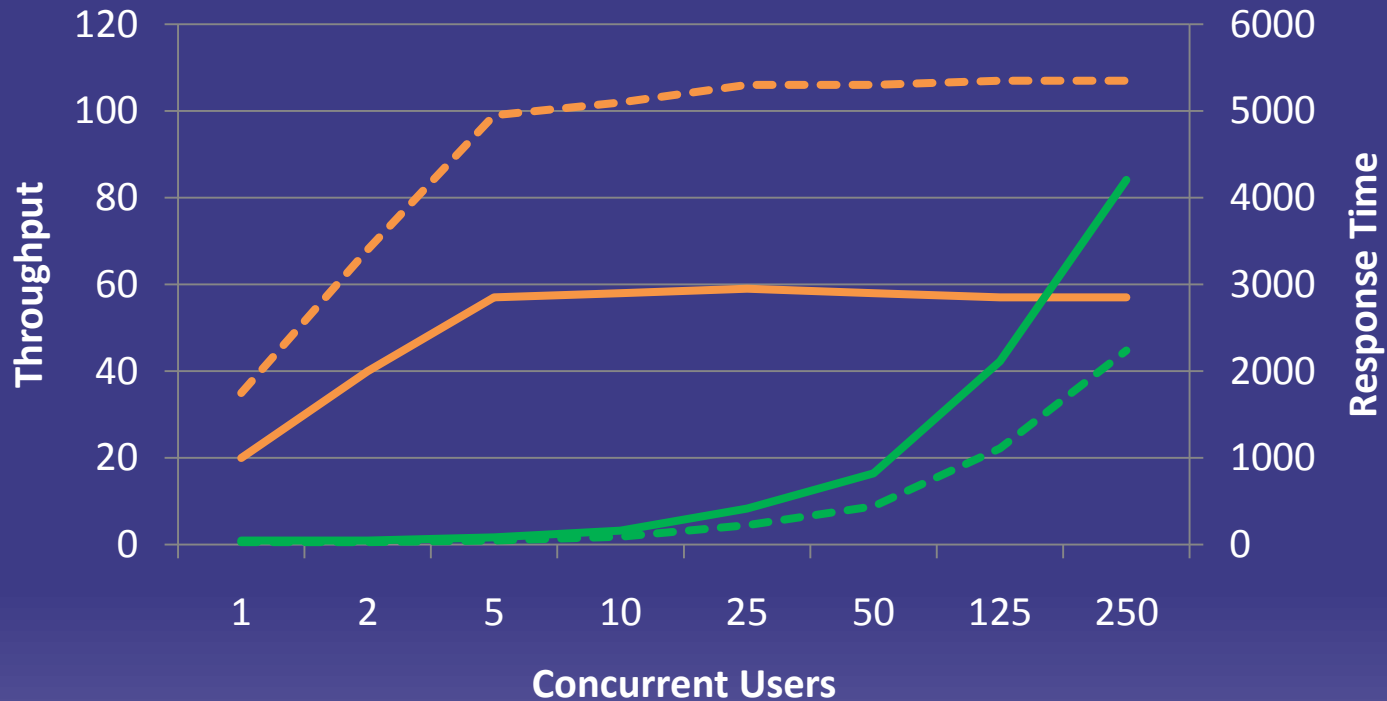
Locking & Row versioning

- Row versioning
 - Database setting
 - Behaviour change
 - Helps with lock waits
 - Overhead – hits tempdb

Locking & Row versioning

- Read Committed Snapshot
 - No code change
- Snapshot isolation
 - Code change
 - More error handling

Row Versioning Results



Throughput

Mean Response (ms)

Throughput Snapshot

Mean Response (ms) Snapshot

Summary

- Good tests - a lot of data + multiple threads
- SQL Profiler
 - Find hot spots - Hardest hit not slowest
- Perfmon – get high level picture
 - Locking, IO, CPU issues?
- Database Engine Tuning Advisor
 - Indexes & Stats

Summary

- Fixes
 - Necessary / Appropriate
 - Indexes – Add & Remove
 - Row Versioning
 - Recode
 - Denormalise
 - Cache
 - Remove Scalar functions / Cursors

Questions

peter.marriott@catalystcomputing.co.uk
[@peter_marriott](https://twitter.com/peter_marriott)

Resources

- Slides and Demos
 - <http://blogs.catalystcomputing.co.uk/2010/09/a-grand-performance>
- ClearTrace - <http://www.scalesql.com/cleartrace/>
- Brent Ozar - <http://www.brentozar.com/>
 - http://sqlbits.com/Agenda/event5/Using_Perfmon_and_Profiler/default.aspx
- Quest Software – (Google and sign up to get)
 - [QuestSoftware-PerfmonPosterSQLServer.pdf](#)
- The Art of Application Performance Testing – O'Reilly
- Microsoft SQL Server 2005 The Storage Engine
- Microsoft SQL Server 2008 Internals