

Optimizing Stored Procedure Performance



Jared Westover

SQL ARCHITECT

@WestoverJared



Module Overview



Creating a baseline

- Dynamic management view
- Pages being read
- Testing multiple values

Performance myths

- Avoiding temp tables
- Select 1 versus select *

Parameter sniffing

- Can have a negative effect
- Overcoming it
- Additional resources

“You can’t hit a target you cannot see, and you cannot see a target you do not have.”

Zig Ziglar



Performance Baseline



Create a performance baseline

- Users complain
- Blocking problems

Testing performance

- Dynamic management view
- Number of page reads

Development phase

- Off to a good start

Demo



Dynamic management view

Gathering performance data

- Reviewing pages read



Performance Fact or Fiction



Busting performance myths

Do temp tables stop plan reuse

- Outdated information
- Add indexes to our temp tables

Check if the value exists

- Select 1 versus select *
- Does this hold up

Breaking up stored procedures

- Perhaps another myth

Demo



Temporary table plan reuse

- Simple test



Demo



Using If exists or if not exists

- Select 1 versus select *



Parameter Sniffing



Cached parameter values

- Expected behavior
- One size fits all

We need a bigger car

- Reusing parameter values

How to fix it

- Recompile the sproc
- Recompile the statement

Identifying & Fixing Performance Issues Caused by Parameter Sniffing

Demo



Identify parameter sniffing

- Execution plan

Dealing with parameter sniffing

- Recompile



What We Covered



Sproc performance markers

- Reviewed execution statistics
- Reduce pages being read
- Temp tables and plan reuse
- Select 1 versus select *

Importance of testing multiple values

Parameter sniffing

- When it can affect performance

Overcoming Parameter sniffing

- Recompile sproc
- Recompile statement



Next Module: Building Stored Procedures in the Real World

