

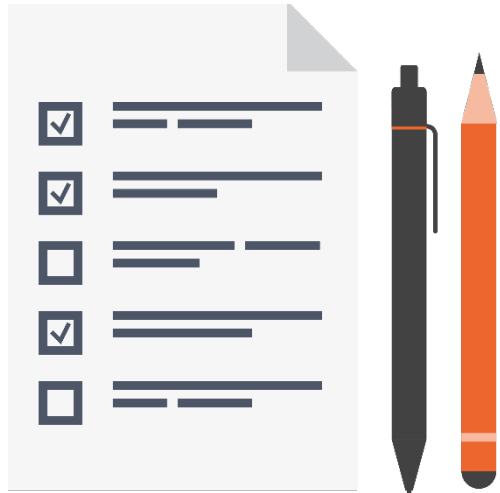
Troubleshooting Techniques



Kimberly L. Tripp

@KimberlyLTripp | www.SQLskills.com

What This Module Covers



Demystifying plan caching for poor performance

Demystifying plan caching for inconsistent plans

Optimizing statement and procedure performance

Where to go next

Course summary

Demystifying Plan Caching for Poor Performance

Check a recompiled plan against the
current plan

EXEC procedure PARAMs -- this will use the plan
from cache (that has the same session settings, etc.)

EXEC procedure PARAMs WITH RECOMPILE -- this
will create a plan just for this execution

Demystifying Plan Caching for Poor Performance

If you get a different plan, then you know there's a parameter sniffing problem

Temporarily use **OPTION (RECOMPILE)** to solve the problem

Look at the long term ramifications of this and see if you can do something else:

OPTION (OPTIMIZE FOR ...) or **OPTION (OPTIMIZE FOR UNKNOWN)** might be less expensive

Or, consider rewriting the code for the “hybrid solution” described in *SQL Server: Optimizing Stored Procedures Performance*–Part 1

Demystifying Plan Caching for Poor Performance

If you don't get a different plan, remember it's not conclusive – WITH RECOMPILE is not comprehensive

Does the statement work when executed from SSMS and then perform poorly in the procedure?

If yes, are there any features that **REQUIRE** recompilation?

Does the procedure work efficiently in SSMS but not in your application?

It's probably the different application / environment settings (e.g. SET Options)

Demystifying Plan Caching for Inconsistent Plans

First consideration for inconsistent plans
and/or poor performance **across
applications**

It's probably the different application / environment
settings (e.g. SET Options)

Check plans using the code from my
ProcedurePlanState procedure...

Now you know – it's definitely session settings!

How do you verify inconsistencies around
SET options?

Resolving Problems with SET Options

Check all of your connected users and their application session settings

At the user level:

```
SELECT *  
FROM [sys].[dm_exec_sessions]  
WHERE [session_id] = @@SPID
```

Across sessions:

```
SELECT *  
FROM [sys].[dm_exec_sessions]  
WHERE [is_user_session] = 1
```

Resolving Problems with SET Options

Check into how the session settings are being set on the client

Can you change them?

Do they negatively affect the application?

Resolving Problems with SET Options

Check the plan cache and procedure attributes
Check resource governor pools

Optimizing Statement and Procedure Performance

- Between these three courses, you have a complete understanding of the most commonly problematic areas around caching, compilation, and performance:
 - *SQL Server: Optimizing Ad Hoc Statement Performance*
 - *SQL Server: Optimizing Stored Procedure Performance Part 1 and Part 2*
- There are still other areas that can frustrate you but there IS a rhyme and a reason to it and there is a methodology for tracking it down and resolving it
 - Compiled plans aren't always best
 - Recompiling isn't always a bad thing
- It depends!
 - And now you know – *even more thoroughly* 😊 – what it depends on!

Where to Go Next?



Check for new Pluralsight courses from me

You should consider these SQLskills courses on Pluralsight next:

- SQL Server: Why Physical Database Design Matters
- Developing and Deploying SQL Server ISV Applications
- SQL Server: Common Performance Issue Patterns
- SQL Server: Troubleshooting Query Plan Quality Issues

Everyone should watch Paul's course: SQL Server: Myths and Misconceptions

- It gives you all sorts of great advice – across the entire Engine!

Course Summary



Performance doesn't just "happen"

Do not expect the SQL Server defaults to perfectly support every environment and every use case

The effect on performance of using only one strategy for all of your procedures DOES NOT WORK

- Use the right method for the right request (and the right data pattern)
- Caching isn't *always* good... it isn't *always* bad either
- Knowing what works and how to test it is key to good statement execution as well as reducing plan cache pollution and CPU!

Thanks for watching!