(June 23rd, 2014)

If you know someone who you think would benefit from being an Insider, feel free to forward this PDF to them so they can sign up here.

Quick Tips for our Insider friends!

Hey Insiders!

This newsletter is coming to you from Redmond where we've just finished our last ever IE1 and IE2 classes. Don't panic! We've revamped them and changed their names so together the two new classes form a 10-day combined course on performance tuning and optimization. You can read about the changes and their **first public offering in Chicago in October** here.

Registrations are coming in thick and fast for the next of our popular SQLintersection conferences, in Las Vegas this November – see here for details. Also, use the discount code "SQLskills" and you can save $50 off registration!

The most recent book I've read is Bernard Cornwell's *1356*. I've been a huge fan of Cornwell for many years as he writes great historical fiction. This is the latest in his Grail Quest series following a group of English archers in the turbulent 14th century as they fight across France in various battles. The book describes the Battle of Poitiers where the Black Prince routed King Jean of France, along with some intrigue and side-stories along the way. If you're a fan of the genre and you haven't checked out Cornwell before, I recommend starting with The Archer's Tale. Also try his Saxon Chronicles series set in 10th century Anglo-Saxon 'England', starting with The Last Kingdom.

Please let us know if you liked what you read/saw here and/or have any suggestions for future Quick Tips.

Note: you can get all the prior Insider newsletters here.

Paul's Ponderings

One of my pet peeves is knee-jerk performance tuning, and I've written about it several times in this newsletter and I have an ongoing blog post series over on SQLPerformance.com. Just yesterday, two things happened which made me want to write about it again in this newsletter.

Thing number 1: there was a discussion about OLEDB waits during DBCC CHECKDB and the first thing proposed as an answer was that OLEDB is from linked-servers so there must be a linked-server query running that is sucking down performance. As you all know, OLEDB waits are actually *caused* by DBCC CHECKDB (it uses OLE-DB under the covers) and nowadays the major cause of OLEDB waits is actually third-party monitoring tools running DMVs (DMVs

use OLE-DB under the covers too). Having a baseline of what the normal wait statistics are would have allowed the original poster to tell whether the OLEDB waits were relevant or not.

Thing number 2: someone wrote a comment on my blog post about the online index rebuild corruption bug, stating:

> Most of the time rebuilding indexes is completely unnecessary. Just disable the index rebuild jobs on your systems to be safe from this corruption bug. Performance gains from index rebuilds are almost always from updating the statistics.
>
> Next time you think you have a perf problem due to index fragmentation, update your stats instead to see if that fixes the issue.

Wrong, wrong, wrong.

The knee-jerk thing here is about updating statistics causing performance gains. This is a long-held misconception about how query plans and plan caching work in SQL Server.

Yes, it is true that after updating statistics, performance often improves, but it's often not the updated statistics that caused the performance gain, it's the fact that the query plan recompiled. Or, it could be a combination of a new plan with updated statistics but more often it's actually just the plan and the statistics update has nothing to do with it.

If there's a poor plan in cache, then allowing the plan to recompile may cause a better plan to be cached, resulting in a performance gain. Kimberly covers statement execution and plan caching in her Pluralsight course *SQL Server: Optimizing Ad Hoc Statement Performance* (and there's a demo from that course below) and she's recording a new course on stored procedure performance as you're reading this newsletter that should be out in a month or so.

She also covers this exact scenario in a blog post (*What caused that plan to go horribly wrong – should you update statistics?*) that I recommend you read.

It's a similar situation to the old misconception that full backups clear the log. Yes, often you'll find that the log will clear when a full backup completes, but it's not the full backup that allowed the log to clear. It's the fact that there were concurrent log backups while the full backup was running and then deferred log clearing happened when the full backup completed (discussed and proved here).

Call to action: When you're troubleshooting, avoid the temptation to blindly update statistics without making sure that statistics really are the cause of poor performance. Often they're not, and you'll pick up a bad habit of always updating statistics to try to fix performance in the future. A better choice for troubleshooting (if it's a stored procedure) is to run sp_recompile *procedurename*. If the problem seems to go away (at least temporarily), it's likely to be a parameter sniffing problem (also called parameter sensitivity problem). Further work will need

to be done to keep this problem at bay but an sp_recompile is a lot less work than an update statistics and it directly addresses the immediate problem.

And for the OLEDB waits issue, having a baseline of wait statistics would really help, because then you'd see whether OLEDB waits were part of the normal wait statistics distribution for your system. Baselines are one of the cornerstones of efficient troubleshooting, as without them you don't know what's normal and what's not

I'm curious to hear your thoughts on knee-jerk performance troubleshooting and especially the scenarios discussed above, so please feel free to drop me a line, always treated confidentially, of course.

Video Demo

In this week's Insider demo video, I'm giving you a demo video from Kimberly's Pluralsight course *SQL Server: Optimizing Ad Hoc Statement Performance*. The demo shows how to use the query_hash and query_plan_hash values during plan cache investigations of whether ad hoc statements are safe for caching or not.

The video is just over ten minutes long and you can get it in WMV format here.

You can get the demo code here.

Enjoy!

SQLskills Offerings

Please know that all of our classes will run and their dates will not change. Additionally, most of our public training courses will be held in the first half of this year. We will add a few more classes in the second half of the year in Australia. Please plan accordingly. And, if you're in Australia and want direct notice when our classes are added, please email training@SQLskills.com.

Finally, to help your boss understand the importance of focused, technical training, we've added a few new items to help you justify spending your training dollars with us:

- Letter to your boss explaining why SQLskills training is worthwhile
- Community blog posts about our classes
- Immersion Event FAQ

2014 Immersion Events

Chicago, IL

- October 6-8, 2014: **IE0**: Immersion Event for the Accidental/Junior DBA
- October 9-10, 2014: **IEHW**: Immersion Event on SQL Server Hardware
- October 6-10, 2014: **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1 (formerly IE1)
- October 13-17, 2014: **IEPTO2**: Immersion Event on Performance Tuning and Optimization – Part 2 (formerly IE2)

See here for the main Immersion Event Calendar page that allows you to drill through to each class for more details and registration links. **We will release our 2014 Australia class schedule shortly.**

Fall SQLintersection

This year our Fall SQLintersection conference will be the week of November 10th in Las Vegas. See here for details. Don't forget to use the discount code "SQLskills" (without the quotes and it isn't case-sensitive) and you can save $50 off registration! We hope to see you there!

Summary

We hope you've enjoyed this issue - we really enjoy putting these together.

If there is anything else you're interested in, we'd love to hear from you - drop us a line.

Thanks,
Paul and Kimberly

Paul@SQLskills.com and Kimberly@SQLskills.com