(September 29th, 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're at home for three weeks before we dive back into our Immersion Events running in Chicago starting next week (October 6). These are our last IE classes in the US in 2014; we still have a few seats open if you're interested in attending IEPTO2. And, for those of you who have been to IE1 – we can even offer you a further discounted "alumnus" price. Shoot me an email if you're interested in attending and I can give you a one-time code for this October alumnus special! If you're waiting for 2015, we plan to have our 2015 calendar posted shortly.

Kimberly and I have been putting in some very late nights recently doing a series of remote user group presentations in Australia – it's amazing how bandwidth has improved since the last time I did the infamous Naked Tour back in 2007…

Important: Make sure you update your PASS profile so you're eligible to vote in the Board of Directors election. There was a big kafuffle last week when I and many others noticed we weren't eligible because we hadn't been notified that we had to update our profiles by June 1st. I and a few others complained loudly and strongly and PASS came through with a fix. You've got until October 5th to update your profile – see here for details.

Also important: Last chance to register for our IE0 and IEPTO1 classes in Chicago next week – there are two seats available in each.

Registrations are steadily coming in for the next of our popular SQLintersection conferences, in Las Vegas this November – see here for details. Our show focuses on the troubleshooting and performance problems that we know you're facing today but will also highlight best practices in architecture, design, and SQL Server 2014. For a total of 6 full days – you can immerse yourself into great content with speakers that you know will deliver! Also, use the discount code "SQLskills" and you can save $50 off registration.

I read a couple of books since the last newsletter:

Firstly, P. W. Singer's *Wired for War: The Robotics Revolution and Conflict in the 21st Century*. This is a very interesting book that charts the growth of the use of robots in the US military, from mine detectors to remote sentries to various kinds of drones. Regardless of your views on drone strikes (and no, although I was accused of it when I included a review of The Divide earlier this

year, I'm not trying to push a political agenda in a SQL Server newsletter – sigh), the technical background behind military robot development and the machinations involved are fascinating.

Secondly, James S. A. Corey's *Abaddon's Gate*. This is the third book in Corey's excellent The Expanse space opera series, centered around our solar system with Mars, Earth, and outer planets factions dealing with a long-dormant alien technology that awakes, and of course pandemonium ensues. Great stuff for sci-fi fans.

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

Last time I wrote about ensuring you have successful backups. This time I'd like to talk about changing recovery models and the effect it can have on your backups, brought to my attention this week through a few email questions I got.

As a quick reminder, there are three recovery models:

- Full: all operations fully logged, log backups required to allow log clearing. All recovery options are available when a database is in the full recovery model (and has been since the last backup).
- Bulk-logged: some operations (e.g. index build/rebuild and bulk load, but NOT regular insert/update/deletes) are minimally logged (only allocations are logged, not inserts), log backups required to allow log clearing. Using bulk-logged, you trade off some recovery options (point-in-time restore and tail-of-the-log backups) for the performance gains associated with minimally logged operations.
- Simple: exactly the same logging characteristics as bulk-logged, but no log backups are possible and log clearing is attempted when a checkpoint occurs. This is the most limited in terms of recovery options.

Most people use the full recovery model, to allow log backups and permit all possible restore operations.

Some circumstances call for simple, if you don't need the ability to do point-in-time restore or zero-to-minimal data loss restores using log backups. An example would be a scratch database that's repopulated once per day and any changes can be lost or easily regenerated.

I'll talk about bulk-logged in a minute.

One of the emails I received was about switching to the simple recovery model during ETL operations to avoid log growth. This is the wrong thing to do as switching to the simple recovery model breaks the log backup chain, requiring a full or differential backup before any further log backups can be performed.

Furthermore, it limits your ability to recover during a disaster because you've now only got one full backup from which you can restore: the one you performed most recently. Think about it: your restore options become:

- Full backup after switch to simple, plus any differentials log backups, or
- Most recent full backup before switch to simple, plus the differential after the switch back from simple, plus any log backups

If that most-recent full backup (before or after the switch to simple) is damaged, you cannot restore – period.

Switching to the simple recovery model is not something you automate or do multiple times.

What the developer really meant to do was switch to bulk-logged during the ETL process so that the large load was minimally-logged. Switching back-and-forth between full and bulk-logged doesn't affect the log backup chain in any way. However, using bulk-logged can cause problems for disaster recovery.

Problem 1: a log backup that contains a minimally-logged operation cannot be used during a point-in-time restore. This means the time you specify in the *WITH STOPAT* clause of the restore statement cannot be a time covered by such a log backup. You can use that log backup as part of a restore sequence, and stop at any point in time after it (as long as that point in time is not covered by another log backup containing a minimally-logged operation, of course), but just not during it.

Problem 2: if you need to perform a tail-of-the-log backup to capture all the log generated since the most recent scheduled log backup, and the log to be backed up contains a minimally-logged operation, that backup will fail prior to 2008 R2, and from 2008 R2 on will succeed, but be corrupt and unusable.

So if you're going to use bulk-logged to save on log space during large operations, you need to make sure that a) there's no possibility you're going to want to restore between the last log backup and the next one, and b) there are no changes made to the database that you cannot recreate in case a disaster occurs and you can't take a valid tail-of-the-log backup.

So even switching recovery models between full and bulk-logged may not be as safe as you may have thought.

I think it's very important that you know when your databases are having their recovery models changed so you can make sure that inadvertent problems are not being created. There are several ways you can track when a database's recovery model changes and with Policy Based Management you can track and prevent changes – extremely useful. Check out Anup Warrier's blog post that explains how to do all that.

Call to action: For every database that you're responsible for, make sure that no-one can change the recovery model without you allowing it as it could cause you problems with disaster recovery. Put in place monitoring, or better yet, disallow recovery model changes.

I'm curious to hear your thoughts on recovery model changes, so please feel free to drop me a line, always treated confidentially, of course.

Video Demo

For the video this time I want to share a clip from Kimberly's new Pluralsight course SQL Server: Optimizing Stored Procedure Performance. In the clip Kimberly explains some of the operations that can cause a plan to be invalidated.

The video is just over 5 minutes long and you can get in MOV format here.

Enjoy!

SQLskills Offerings

We've released all the classes for the remainder of 2014, including two in Australia in December. We expect to release the first half of our 2015 schedule by mid-October.

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 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)

Sydney, NSW, Australia

- December 8-12, 2014: **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1 (formerly IE1)

Canberra, ACT, Australia

- December 15-19, 2014: **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1 (formerly IE1)

See here for the main Immersion Event Calendar page that allows you to drill through to each class for more details and registration links.

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