(July 5th, 2016)

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 I've been busy producing more Pluralsight courses so you have something to watch over the summer. ☺

The latest course published is Erin's *SQL Server: Replacing Profiler with Extended Events*, based on her popular user group/conference session. Check it out here.

Our September classes in Bellevue and Fall classes in Chicago are filling up, so be sure to book your Immersion Event in advance to avoid the class you want selling out – all the schedule details are here.

SQLintersection session and workshop details have been posted; we have an exciting line-up scheduled for our show! If your team needs architectural advice and sessions from speakers who not only know their technology but know how to convey it – this is the place to be! Be sure to use the 'SQLskills' discount code to save $50 on registration. Check it out here.

Due to popular demand, we've extended our special offer through the end of July, where any client – new and existing – can get a **single-instance health check for only US$2,500**, which is more than 1/3 off the normal price! Details about why our health checks are so cost-effective are here.

The latest book I've read is Tom Holland's *Rubicon: The Last Years of the Roman Republic*. I've read nearly all of Holland's books and this one is just as good as all the others. As the title suggests, it covers the gradual degradation of the Roman Republic and its transformation into an empire. It goes a lot deeper into the time leading up to Julius Caesar's assassination, and gives a lot of detail of the machinations of the likes of Cicero, Cato, Crassus, and other consuls and wielders of power. Great to read this after Mary Beard's *SPQR: A History of Ancient Rome* and looking forward to diving deeper still with more focused works. Highly recommended! (And yes, I've seen the *Rome* series set around this time - excellent too.)

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

The Curious Case of…

This section of the newsletter explains problems we've found on client systems; they might be something you're experiencing too.

Glenn was working with a new client recently who had implemented a comprehensive high availability strategy based on database mirroring on Standard Edition, but was having problems. Whenever they tried failing over their main production database to its mirror, even though the mirroring state was *SYNCHRONIZED*, it would take several hours for the new principal database to come online. They were running in high safety mode (i.e. synchronous) with a witness, so were worried that an automatic failover would occur and lead to extensive downtime.

It turned out that the storage subsystem on the mirror server was severely overloaded and so the redo queue for the mirror database was enormous, which they hadn't noticed so when a failover occurred, they had to wait for all of the redo to occur before the database could come online.

Once Glenn helped them get the storage subsystem back to normal performance, the problem was solved.

There are a bunch of lessons to be learned from this, and some more explanation required about what was happening under the covers, so I'll discuss in more detail below...

Paul's Ponderings

Everything I'm going to discuss below applies to both synchronous database mirroring and to using a synchronous availability group replica, but for the sake of simplicity, I'm just going to use mirroring terms.

The crux of the problem described above is that when the mirroring state is *SYNCHRONIZED*, that doesn't mean that a failover will bring the mirror database online quickly. A state of *SYNCHRONIZED* means that there is no *send queue* on the principal server, or to put it another way, all committed transactions on the principal are also durable on the mirror, so when a failover occurs, no data is lost. (Let's not get bogged down by discussing the exception to that rule when using delayed-durable transactions in SQL Server 2014 or later…)

However, many people think that *SYNCHRONIZED* also means that the mirror is completely up-to-date with replaying the log records in the mirror database, so will be immediately ready for use when a failover occurs. This is not the case. (There's an additional misconception that when a transaction commits on the principal, and synchronous mirroring is in use, that the transaction also commits on the mirror – no – all that has to happen is that all the log blocks from the principal, up to and including the one with the *LOP_COMMIT_XACT* log record of the committing transaction, are sent to the mirror and written to its log drive.)

The mirror server is constantly replaying log records in the mirror database and it is quite possible that it can fall behind the principal, meaning that there will be a queue of log records that have not yet been replayed – called the *redo queue*. If the storage subsystem of the mirror is slow for some reason, this can slow down the replaying of log records and lead to a lengthening of the redo queue.

When a failover occurs, and there is a redo queue, it must be processed before the mirror database can come online as the new principal database. You can think of it as crash recovery running – with the redo of committed transactions and the undo of uncommitted transactions.

The client case described above was exacerbated because they were using Standard Edition, where both redo and undo must occur during crash recovery (or a failover) before the database can come online. In Enterprise Edition, *fast recovery* applies, where only the redo portion of crash recovery has to happen before the database comes online (see this blog post for a lot more detail).

Call to action: If you're using synchronous database mirroring or a synchronous availability group replica, it's crucial that you monitor the redo queue (discussed in this blog post), so you can be alerted if a failover is going to require significant downtime.

Video Demo

In this video, Jonathan looks at a new column that was added to *sys.dm_db_session_space_usage* in SQL Server 2012 SP3, SQL Server 2014 SP1 and SQL Server 2016 RTM to track deferred page deallocations for temporary objects with 128 or more extents allocated to them.

The video is 3.5 minutes long and you can get it:

- In WMV format here.
- In MOV format here.

The demo code is available here.

Enjoy!

SQLskills Offerings

Our 2016 classes are all open for registration (listed below), including three new classes in Chicago in November added because of popularity. We hope to see you at a class or a conference this year!

To help your boss understand the importance of focused, technical training, we've also added a few 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

Upcoming Immersion Events

Bellevue, WA

- **IEPTO1:** Immersion Event on Performance Tuning and Optimization – Part 1
  - September 12-16
- **IEBI:** Immersion Event on Business Intelligence
  - September 19-22 **Just added!!**
- **IEPTO2:** Immersion Event on Performance Tuning and Optimization – Part 2
  - September 19-23
- **IESSIS2:** Immersion Event on Advanced SQL Server Integration Services
  - September 19-22 **New course!!**

Dublin, Ireland (*returning to Europe in 2018* ***NOT 2017***)

- **IEPTO1:** Immersion Event on Performance Tuning and Optimization – Part 1
  - October 3-7

Chicago, IL

- **IE0**: Immersion Event for Junior/Accidental DBAs
  - November 7-9 **Just added!!**
- **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1
  - November 7-11 **Just added!!**
- **IEPDS**: Immersion Event on Practical Data Science
  - November 7-11 **Just added!!**

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

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