(May 22nd, 2017)

If you know someone who 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,

As this hits your inboxes we're in Orlando kicking off the first few sessions of our Spring SQLintersection conference after a weekend of workshops! One day we'll schedule ourselves some extra time to go to the Kennedy Space Center… For now, we'll just have to enjoy the warm evenings and the fantastic company here in FL. We're excited for an information-packed week and our favorite evening event: SQLafterDark on Tuesday night.

IMPORTANT: Please be aware of a virulent computer threat that's going around right now, WannaCrypt/WannaCry, which encrypts your hard drive and then demands payment in bitcoins. There's more info about it and how to make sure you're protected in the *Glenn's Tech Insights* section.

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

SQLskills News

There's an exciting new addition to my waits library – I've teamed up with SentryOne to add infographics to all the waits, showing the prevalence of wait types out in the world. Check out more information here.

We've just released our October line-up of classes in Chicago, including IE0, IEPTO1, IEPTO2, IEPDS, IESSIS1, plus THREE new courses. We're debuting a **new two-day class on Azure SQL Database and Azure VMs**, a **new three-day class on Upgrading SQL Server**, and a **new two-day class on Clustering and Availability Groups**. See here for the complete 2017 SQLskills Immersion Event class schedule.

We've started a new initiative where we're all blogging about introductory topics, to help the burgeoning numbers of non-DBAs or junior DBAs who are responsible for SQL Server instances. We're calling it SQL101, and hopefully it'll even help refresh or reinforce topics for the more seasoned DBAs in the community. The blog posts will be automatically collected here.

Even though we're not teaching any Immersion Events in Europe this year, Kimberly and I will both be **presenting at SQLSaturday #620 in Dublin in June**. This is coming up quickly so we hope you can join us for our two-day SQLskills Performance Track prior to SQLSaturday in Dublin – check out the details here.

Finally, even if you can't join us in person, we're still taking requests for remote user group sessions for this year. We have over 70 scheduled so far; if you'd like one of us to present for your user group, check out my blog post here.

Book Review

The latest book I've read is Hugh Howey's *Shift*. This is the second in Howey's Silo trilogy. This is a really interesting post-apocalyptic tale about survivors living in 140-story deep, hermetically-sealed silos in the ground over hundreds of years. Hard to describe the story without giving away some of the plot twists, but in this book we find out the back story of how it all started and why. I'd definitely start with the first book, *Wool*, before reading this one. Highly recommended!

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.

While in class last week, I chatted with a student about a performance problem they experienced. The symptom was that every afternoon their main workload would start to slow down and they were also seeing an increasing amount of *PAGEIOLATCH_SH* waits.

After some investigation, we found that a large amount of their buffer pool was being taken up with pages from a database snapshot.

They had been using transactional replication to keep a reporting server up-to-date, but had recently switched to using a database snapshot on the production instance that was being recreated every morning and then used to run reports.

As more and more reports were run, more and more of the pages from the database snapshot were being read into memory and this was forcing more pages from the production database to be dropped out of memory, causing the increase in reads and slower performance.

So they had a choice of whether to go back to using transactional replication or to increase the size of their buffer pool (which they hadn't made at the time of writing this).

Bottom line: database snapshots are cool, but there are some hidden side effects of using them so it's important to do thorough testing and production-monitoring to ensure their benefits. I'll go into a bit more of this in the Ponderings section below.

Paul's Ponderings

In the Curious Case section above, I mention that database snapshots have some hidden side effects so I'd like to detail them here.

Firstly, there's the potential for a lot of extra writes occurring. A database snapshot creates and preserves a 'point-in-time, unchanging' snapshot of a database. To do this, after a database snapshot has been created, for all pages that were in the source database at that time, the first time a page is about to be changed, the pre-change image must be written into the database snapshot.

The write is performed synchronously by the thread that's about to change the page, and one write must be performed for all database snapshots that need that pre-change image of the page. Although this process only happens once per page, if a lot of the source database changes, that's a lot of extra write activity happening, and can cause a slowdown in the main workload.

Secondly, there's the extra disk space that is required to store these pre-change page images and this is directly tied to the amount of the source database that changes while a database snapshot exists. Although disk space may not be a problem in your environment, if the free space is fragmented in the NTFS volume hosting a database snapshot file, it's possible that the database snapshot may hit an architectural limit in the amount of fragments a sparse file can have and fail.

This particular issue can be particularly prevalent when running DBCC CHECKDB, as it creates database snapshot files in the same location as the source database data files. This can be avoided by placing the database snapshot files on a volume with contiguous free space. See KB article 2002606 for more details and various fixes.

Lastly, there's the case I discussed in the section above. When a page is read from a query performed in the context of a database snapshot, it enters the buffer pool as a page 'owned' by the database ID of the database snapshot. This is the case even if the page was unchanged since the database snapshot was created and was actually read from the source database itself.

If the same page is then read from a query performed in the context of the source database, it enters the buffer pool as a page owned by the database ID of the source database. This means there will be two copies of the same page in memory – each owned by a different database ID.

The buffer pool architecture does not allow a page to have shared ownership by multiple databases (I guess it could be done, but it would really complicate buffer pool code around operations like changing pages and maintaining the Least Recently Used metrics).

This means that an extensive query workload on a database snapshot could cause a large amount of buffer pool memory to be taken up by secondary copies of pages from the source database, which could lead to other databases not having enough buffer pool space for their workloads and a drop in performance. And this is exactly what happened in the Curious Case I discussed.

I produced an Insider demo video about this back in July 2011 – check it out here.

Call to action: database snapshots are very useful, but make sure you understand some of the side effects from using them so your workload performance isn't detrimentally affected.

Glenn's Tech Insights

This section of the newsletter highlights recent news and views from the hardware and Windows worlds that we think will be interesting to SQL Server community members.

WannaCrypt/WannaCry Ransomware Guidance

There has been quite a bit of media coverage about the WannaCrypt/WannaCry ransomware over the past several days. Microsoft has a new page with information about this particular issue and steps that can be taken to protect your systems. I have also collected some more detailed background information about this and about SQL Server security patching in general.

Just to be clear, there is no known threat to SQL Server from this method, but there was an out-of-band security update for SQL Server 2012, 2014, and 2016 that was released on November 8, 2016. Here are the most current cumulative updates for SQL Server 2012, 2014, and 2016 (which will include that security update).

- SQL Server 2012 SQL Server 2012 SP3 CU9 11.0.6598.0 May 15, 2017
- SQL Server 2014 SQL Server 2014 SP2 CU5 12.0.5546.0 April 17, 2017
- SQL Server 2016 SQL Server 2016 SP1 CU3 13.0.4435.0 May 15, 2017

WannaCrypt/WannaCry Information

Here are some links to useful resources about this outbreak. Making sure your servers and client machines are current with their Microsoft Update hotfixes and possibly disabling SMB v1 are the best defenses.

- Alert (TA17-132A) Indicators Associated With WannaCry Ransomware
- Microsoft Security Bulletin MS17-010 – Critical
- MS17-010: Description of the security update for Windows SMB Server: March 14, 2017
- Windows Update Catalog Download Links

SMB v1 Information

Another mitigation measure for this vulnerability is to disable Server Message Block (SMB) v1 (which has been deprecated since Windows Server 2012). Depending on what version of Windows Server you are running, you may be able to do this using various methods.

- The Deprecation of SMB1 – You should be planning to get rid of this old SMB dialect
- Stop using SMB1
- How to enable and disable SMBv1, SMBv2, and SMBv3 in Windows and Windows Server

SQL Server Security Update Information

Microsoft now recommends proactively installing SQL Server Cumulative Updates as they become available. The most recent, specific security update (MS16-136) for SQL Server 2012, 2014, and 2016 was released on November 8, 2016. If you are up to date with your SQL Server Service Packs and Cumulative Updates, you will already have that SQL Server security update. Just to be clear, there is no indication that SQL Server is vulnerable to WannaCry. It is merely a best practice to stay current with SQL Server security and other updates.

- Announcing updates to the SQL Server Incremental Servicing Model (ISM)
- Where to find information about the latest SQL Server builds
- MS16-136: Security update for SQL Server: November 8, 2016

Finally, there are a number of other good reasons to make an effort to keep your SQL Server instances up to date with the latest Service Pack and Cumulative Update. I highlight some of the more important hotfixes for every cumulative update in the blog posts linked below:

- Performance and Stability Related Fixes in Post-SQL Server 2012 SP3 Builds
- Performance and Stability Related Fixes in Post-SQL Server 2014 SP1 Builds
- Performance and Stability Related Fixes in Post-SQL Server 2014 SP2 Builds
- Performance and Stability Related Fixes in Post-SQL Server 2016 SP1 Builds

SQL Server Core Factor Table

During the SQL Server 2012 release time-frame (when SQL Server core-based licensing was first introduced), Microsoft initially published a SQL Server Core Factor Table document that detailed a 0.75 core factor for selected AMD server processors with six of more physical cores. This made it slightly less expensive to purchase SQL Server 2012 processor core licenses for these AMD Opteron server processors.

This document was updated for SQL Server 2014. Even with this discount, it was not really cost-effective to use AMD processors for SQL Server usage, because of their extremely poor single-threaded performance. You could easily get more total capacity, better single-threaded performance, and lower SQL Server licensing costs with an appropriate Intel Xeon processor.

For the SQL Server 2016 release, there was no update for the SQL Server Core Factor Table. In fact, Microsoft has a useful new document, entitled *Introduction to Per Core Licensing and Basic Definitions* where they explicitly state that the Core Factor Table is not applicable to SQL Server starting with SQL Server 2016.

Intel Marketing Tricks

Intel has released a document that claims *"1.59X Higher performance for database workloads"* with the new Intel Xeon Processor Scalable family for SAP HANA database workloads compared to the previous-generation Intel Xeon E7 v4 processor family.

This sounds rather notable, but looking into the small amount of detail provided by Intel makes it less impressive. Here are the testing details provided by Intel:

"Up to 1.59x higher online transaction processing (OLTP) performance vs. Intel Xeon processor E7 v4 family performance estimate based on SAP HANA internal S-OLTP workload (internal testing). Baseline config: one-node, 4S Intel Xeon processor E7-8890 v4 with 1,024 GB total memory on SUSE Linux Enterprise Server* (SLES*) 12 SP1 vs. estimates based on SAP internal testing on one-node, 4S Intel Xeon Processor Scalable family system."*

It is likely that the SAP HANA S-OLTP workload result is primarily a measure of the ***total CPU capacity*** of the system. As such, it is going to be directly influenced by the total number of processor cores in the system. The Intel Xeon E7-8890 v4 processor has 24 physical cores, while the unnamed, unreleased Xeon Processor Platinum Scalable family processor probably has more physical cores (perhaps up to 32).

The difference in physical core counts for the entire system is likely responsible for the bulk of the benchmark score improvement. With the small amount of detail Intel gives us, we really can't evaluate the possible single-threaded performance improvement for the next generation Intel Xeon processors.

#TBT

(Turn Back Time…) This section of the newsletter highlights some older resources we've referred to recently that you may find useful, plus select blog posts we've published since the previous newsletter.

I was reading a cool post by Bob Ward last week where he used Extended Events to analyze the phases of a restore. Profiler's been deprecated since SQL Server 2012, and no new events have been added since then, but there's still a lot of reluctance to use them.

So Extended Events are the theme for #TBT this time. Here are some Extended Events resources for you:

- Jon's two Pluralsight courses: *SQL Server: Introduction to Extended Events* and *SQL Server: Advanced Extended Events*
- Erin's Pluralsight course: *SQL Server: Replacing Profiler with Extended Events*
- Jon's An-XEvent-a-Day blog post series
- Jon's Extended Events blog category
- Jon's article on sqlperformance.com comparing the overhead of trace vs. XEvents

- Erin's Extended Events blog category
- Erin's Stairway Series on SQL Server Central
- My Extended Events blog category which shows how to capture SQL Server call stacks

Here are a few of the blog posts we've published since the last newsletter:

- Paul: *New class: Immersion Event on Clustering and Availability Groups*
- Paul: *SQLskills SQL101: Why is restore slower than backup*
- Paul: *Waits library now has infographics from SentryOne monitored instances*
- Glenn: *Guidance for WannaCrypt/WannaCry Attacks*
- Erin: *Thoughts on public speaking / presenting / teaching*
- Jonathan: *Using Storage Spaces Direct S2D on VMware Workstation*
- Tim: *SQLskills SQL101: Database Maintenance for Azure SQL Database*

I hope you find these useful and interesting!

Video Demo

In this video Tim continues his series on how to perform maintenance task on Azure SQL Databases. In the first video he demonstrated the use of a linked server and this time Tim shows how to configure database maintenance plan to connect to Azure and how you can use PowerShell to run Ola Hallengren's IndexOptimize procedure.

Stay tuned for Tim's next video, showing you how to use Azure Automation and Elastic Jobs to schedule your jobs against an Azure SQL Database.

The video is just under 4 minutes long and you can get it:

- In WMV format here
- In MOV format here

And the demo code is here.

Enjoy!

Upcoming Immersion Events

All 2017 classes are available for registration!

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

Bellevue, WA, July/August 2017

- **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1
  - July 31-August 4
- **IEPTO2**: Immersion Event on Performance Tuning and Optimization – Part 2
  - August 7-11

Chicago, IL, October 2017

- **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1
  - October 2-6
- **IE0**: Immersion Event for the Junior/Accidental DBA
  - October 2-4
- **IECAG**: Immersion Event on Clustering and Availability Groups
  - October 5-6 **** New class! ****
- **IESSIS1**: Immersion Event on Learning SQL Server Integration Services
  - October 2-6
- **IEPTO2**: Immersion Event on Performance Tuning and Optimization – Part 2
  - October 9-13
- **IEPDS**: Immersion Event on Practical Data Science
  - October 9-13
- **IEAzure**: Immersion Event on Azure SQL Database and Azure VMs
  - October 9-10 **** New class! ****
- **IEUpgrade**: Immersion Event on Upgrading SQL Server
  - October 11-13 **** New class! ****

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