(July 23rd, 2018)

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.

Note: As an Insider, you can read all prior Insider newsletters here.

Quick Tips for our Insider friends!

Hey Insiders,

This newsletter is coming to you from Cenderawasih Bay, West Papua, Indonesia where we're on the second stage of a family diving vacation, introducing our girls to muck diving, whale sharks, and the general wonderfulness that is diving in Indonesia.

Note: it's time for our annual newsletter break over the summer, so the next newsletter will be on Monday, August 27th – have a great summer!

We have **three brand-new, live, online classes** that are filling up steadily – see below!

Additionally, we still have a few seats remaining in our London classes in September; they're our last, in-person Immersion Events to be delivered this calendar year!

SQLskills News

Glenn's latest Pluralsight course has been published: *SQL Server 2017: Diagnosing Configuration Issues with DMVs*. Check it out here!

Live, online classes: there are three new classes available for registration:

- August 28-30: ***IEPUM2017****: Immersion Event on Planning and Implementing an Upgrade/Migration to SQL Server 2017* (taught by Glenn)
- October 9-11: ***IETLB****: Immersion Event on Transactions, Locking, Blocking, Isolation, and Versioning* (taught by Kimberly)
- October 23-25: ***IEQUERY****: Immersion Event on Fixing Slow Queries, Inefficient Code, and Caching/Statistics Problems* (taught by Erin, Jonathan, and Kimberly)

These classes will be delivered live via WebEx 10am-3pm PST, Tuesday-Thursday (roughly 12-13 hours of content including open Q&As; similar to two, full, workshop days without leaving the comfort of your home/office!) and you also get access to the class recordings. By dedicating only 3 half-days of your time you still have time to get some work done during the day and with **lifetime** access to the recordings, you get amazing ROI!

Each of these classes are priced at US$699 and we're offering a combo package of all three for US$1,749, saving US$350. Click on the class links above for all the details.

In-person London classes: We're bringing four of our Immersion Events to London in September: IEPTO1 and IEPTO2, plus our new classes: **IEAzure** on Azure, Azure VMs, and azure Managed Instance and **IECAG** on clustering and availability groups **(register for one of the new classes and get the other one half-price!)** See here for all the details.

Finally, even if you can't join us in person, I've put out a call for second-half 2018 remote user group sessions and we've done 59 this year already! If you'd like one of us to present for your user group, check out my blog post here.

Book Review

I've read a bunch of books since the last newsletter and here are two that stand out to me.

The first book is Herman Hesse's *Siddhartha*. What a wonderful book! It follows the life of a wealthy Brahmin, Siddhartha, who follows various paths in his life – from privilege to ascetic samana to lust to wealthy merchant to self-disgust to ascetic ferryman to despairing father of a rebellious, disrespectful son to calm and final enlightenment. Along the way he realizes what he's lacking at each stage, and has much self-reflection. I particularly liked how his childhood friend comes back into the story occasionally and we see how the friend, on a lifelong quest for enlightenment by following the teachings of another, didn't fulfill his quest, but Siddhartha, following his own path did. It's altogether an excellent book, with much to think about and take away for one's own journey through life.

The second book is Jack Kerouac's *The Dharma Bums*. Great book! It's based on his experiences in the '50s when he was interested in Buddhism and follows the main character as he hitch-hikes and rides freight trains around, goes mountaineering, spends the summer in a fire lookout in the NW Cascades (in an area I know well) and is part of many whacky parties. He picks up a lot of his Buddhist ideas from his buddy Japhy Ryder, who's based on his real-life poet friend Gary Snyder. I haven't yet read his most well-known book *On The Road*, but I will do on this trip I think.

Both books are highly recommended!

The Curious Case of…

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

Last week we were helping a new client who was experiencing large numbers of *SOS_SCHEDULER_YIELD* waits and really poor workload throughput on a critical system after making some configuration changes.

We asked what the changes were and they told us that they’d been experiencing a lot of *CXPACKET* waits and had followed advice they’d read on the internet to set the server *cost threshold for parallelism* (CTFP) to 50 for that instance, to prevent small queries from running in parallel.

They hadn’t done any testing and had made the change directly in production. As soon as they did that, workload throughput deteriorated drastically.

We lowered the configured CTFP value to 20 as a stop-gap measure and that immediately increased performance, and after testing we determined that a CTFP of 30 was the best value to use for them. This eliminated the *SOS_SCHEDULER_YIELD* waits and brought the workload throughput to a higher level than they started at before their first change.

The problem here was that they were blindly following ‘set it to this’ advice, which is usually the wrong thing to do. The advice should have been ‘experiment with values to see what the sweet spot is for your environment and workload’.

By setting the CTFP to 50, they were preventing some of their critical queries from running in parallel, so producing single-threaded plans doing large scans leading to *SOS_SCHEDULER_YIELD* waits, vastly increased signal wait times across the board, and overall poor performance.

Bottom line: Be very wary of anyone proposing ‘set it to this’ advice. Nearly always you should experiment to find the best value for you, as all scenarios, environments, and workloads vary.

Paul's Ponderings

*I’m on vacation this week so this is a re-work of an editorial from 2012 that’s still hugely relevant today – finding and removing *true* duplicate indexes. Enjoy!*

Our two flagship 5-day classes are about performance tuning. Basically we discuss the common question: what’s taking up all of the resources on my SQL Server and preventing my workload from running as fast as I want? As a part of the first class we discuss indexing in depth, and Kimberly explains a commonly overlooked cause of poor performance: having duplicate nonclustered indexes.

SQL Server will let you create up to 999 nonclustered indexes per table, and if you really want to, you can create them all to be exactly the same. *Exactly* the same, down to the *INCLUDE*d columns and key order. Why? The reason is backwards compatibility, so that SQL Server doesn’t break poorly-written applications that use hard-coded index hints. In actuality, there’s absolutely NO good reason to allow duplicate indexes and often people end up with duplicates that don’t even look similar (because of how SQL Server internally changes the indexes based on the clustering key). The end result is that you might have duplicates and not even know it!

These indexes are not just wasted disk space however…

Every *INSERT*, *UPDATE*, and *DELETE* operation (which I'll just call 'DML' from now on) will affect every nonclustered index (except possibly filtered indexes), as each nonclustered index has to have a row for each row in the table. This means that each DML operation generates transaction log records for the change to each nonclustered index, and the index pages for each index have to be in the buffer pool to have the change made on them.

So each duplicate index is taking up space in the data files, taking up space in your precious buffer pool, taking up space in the transaction log, needing locks, needing latches, taking up processing time, and so on.

And then what about index fragmentation? Each of these duplicate nonclustered indexes will be getting fragmented (unless you've created them all to have the same index keys that match your insert pattern), so there will be a bunch of page splits happening, with all the extra log records those generate. Those fragmented indexes will likely be picked up by automated index maintenance, causing further resources to be used and transaction log to be generated. And then they may be picked up by automated statistics maintenance, and they will have to be checked by your consistency checking jobs, and so on.

You get the idea. Duplicate nonclustered indexes are really bad. And, all you have to do is remove them and you'll reduce overhead and increase performance.

That's the catch though – you have to know internals and how SQL Server stores data in the leaf and non-leaf levels of an index. Luckily Kimberly worked out how to find them and she wrote two posts:

- How can you tell if an index is REALLY a duplicate?
- Removing duplicate indexes

The second post has all the code to use to figure out which indexes are duplicates on 2008 onwards. Additionally one of our class attendees also ported the code to SQL Server 2000 – see here.

(Edit 2018: I left the final link in there, updated to Randolph's new site, as we still meet people in our classes who have SQL Server 2000 instances kicking around because of vendor application limitations.)

Call to action: Read the first post and then download the code from the second post and check out some of your databases to see whether you have any duplicate indexes. You'll be happy if you find them as every one of them is a leech sucking resources out of your SQL Server. One DBA found 77 duplicates in one of their databases just last week!

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.

Intel Motherboard Partners add support for Coffee Lake Refresh to Z370

Many major motherboard manufacturers have released a BIOS update for their current Intel Z370 motherboards that improves system performance and includes an updated version of the Intel Management Engine. The larger news is that this BIOS update enables support for the upcoming Intel "Coffee Lake Refresh" desktop processors, which are supposed to include an eight-core, Core i9-9xxx SKU.

This will be the first time that Intel has released an eight-core, mainstream desktop processor. It is also welcome news that these newer processors will work in existing enthusiast Z370 motherboards, which is a nice change from recent Intel practice. This means that you won't be forced to get a new motherboard if you already have a Z370 motherboard in order to use a Coffee Lake Refresh processor (as long as you have the updated BIOS).

Table Variable Deferred Compilation in Azure SQL Database

Microsoft's Joe Sack has a blog post up that discusses a new feature that is in public preview for Azure SQL Database. When you have this feature enabled (by using compatibility level 150), you should see much better query performance from queries that use table variables to store intermediate results that are later used to join to other objects used by the query. Here is how Joe describes the feature:

> *"Table variable deferred compilation improves plan quality and overall performance for queries referencing table variables. During optimization and initial compilation, this feature will propagate cardinality estimates that are based on actual table variable row counts. This accurate row count information will be used for optimizing downstream plan operations."*

This should be a very useful feature enhancement, especially if your database has many existing queries and stored procedures that use table variables in this way (which is quite common). This would be a much easier fix than using the typical old methods of alleviating table variable performance issues. Hopefully we will see this feature also surface in a future version of the boxed SQL Server product.

Importance of Processor Choice for SQL Server

ServeTheHome (STH) has a pretty exhaustive test of the Intel Xeon Gold 5117 processor, which has 14 cores / 28 threads, along with a 2.0GHz base clock speed and a 2.8 GHz Turbo clock speed, and a 19.5MB L3 cache. This particular SKU is one of six different SKUs in the Xeon Scalable Processor (Skylake-SP) family that has 14 physical cores.

Unfortunately, **none** of those six choices are good choices for SQL Server usage because of their relatively low base clock speed and small L3 cache sizes. The best one of the bunch would be the Intel Xeon Gold 6132 processor, which has 14 cores / 28 threads, along with a 2.6GHz base clock speed and a 3.7GHz Turbo clock speed, and a 19.5MB L3 cache.

A much better choice would be the Intel Xeon Gold 6146 processor, which has 12 cores / 24 threads, along with a 3.2GHz base clock speed and a 4.2GHz Turbo clock speed, and a 24.75MB L3 cache. This 12-core processor would give you much better single-threaded performance and more total CPU capacity than any of the 14-core SKUs. Selecting two of these processors for a two-socket server (rather than a 14-core processor) would also save you nearly $30,000.00 in SQL Server license costs per server.

Remember, the license cost for SQL Server 2017 is based on core counts, not performance. Microsoft does not care about the performance per core, the license cost is the same regardless. With some careful analysis, you can take advantage of this to get better performance for less money.

As it turns out, STH was not impressed with the Intel Xeon Gold 5117 processor.

#TBT

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

I've been dealing with a couple of corruption issues over the last week, so that's the theme for this TBT:

- My course: *SQL Server: Detecting and Correcting Database Corruption*
- My course: *SQL Server: Advanced Corruption Recovery Techniques*
- My blog post: *SQLskills SQL101: Why DBCC CHECKDB can miss memory corruption*
- My blog post: *SQLskills SQL101: Dealing with SQL Server corruption*
- My blog post: *Disaster recovery 101: fixing a broken boot page*
- My blog post: *Corruption recovery using DBCC WRITEPAGE*
- My *Corruption* blog post category

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

- Paul: *New Pluralsight course: SQL Server 2017: Diagnosing Configuration Issues with DMVs*
- Jonathan: *Tracking down high CLR_MANUAL_EVENT waits*
- Glenn: *T-SQL Tuesday #104: Code I Have Written That I Would Hate to Live Without*
- Glenn: *SQL Server 2016 SP2 CU2 Available*

- Erin: *Monitoring Space Used by Query Store*

I hope you find these useful and interesting!

Video Demo

In this demo, Jonathan takes a look at the enhancements to ShowplanXML in SQL Server 2016 Service Pack 2, and how the version of SQL Server Management Studio being used to collect ShowplanXML can limit the information that is available due to changes in the schema definition used.

The video is about 4.5 minutes long and you can get it in WMV format here.

The demo code is here.

Enjoy!

Upcoming SQLskills Events

We have events coming up in 2018 – from our ***new, live, online*** courses to our own live, in-person Immersion Events to our own conference: SQLintersection; all of our Fall events are open for registration.

Each and every event has a different focus as well as different benefits – from deep-technical training in our Immersion Events to wide-ranging topics at SQLintersection where you can learn more effectively how to keep moving forward in both your environment and your career! And, of course, one benefit you'll always get from in-person events is networking; we hope to meet/see you at an event soon!

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
- So why do you want to come to our training? And the winners are…
- Community blog posts about our classes
- Immersion Event FAQ

LIVE, ONLINE Immersion Events:

- **IEPUM2017**: Immersion Event on Planning and Implementing an Upgrade/Migration to SQL Server 2017
  - August 28-30 **(** NEW **)**
- **IETLB**: Immersion Event on Transactions, Locking, Blocking, Isolation, and Versioning

    - October 9-11 **(** NEW **)**
- **IEQUERY**: Immersion Event on Fixing Slow Queries, Inefficient Code, and Caching/Statistics Problems
    - October 23-25 **(** NEW **)**

LIVE, IN-PERSON Immersion Events:

London, UK, September 2018

- **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1
    - September 10-14
- **IEAzure**: Immersion Event on Azure SQL Database and Azure VMs
    - September 10-11 **(** Buy IEAzure, get IECAG half-price **)**
- **IECAG**: Immersion Event on Clustering and Availability Groups
    - September 12-13 **(** Buy IECAG, get IEAzure half-price **)**
- **IEPTO2**: Immersion Event on Performance Tuning and Optimization – Part 2
    - September 17-21

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