(March 19th, 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.

Quick Tips for our Insider friends!

Hey Insiders,

We're firing on all cylinders, getting ready for SQLintersection starting on March 24, Kimberly's live, online IEVLT class that starts tomorrow (Tuesday, March 20), and our Spring Immersion Events (live, in-person classes) in Chicago (starting, Monday, April 23).

Speaking of our events – you still have a small amount of time left to join us somewhere! For Kimberly's online course that starts tomorrow; today is the last chance to register and attend her highly-rated *IEVLT: Immersion Event on Very Large Tables: Optimizing Performance and Availability through Partitioning* course. **And, it's the last delivery slated for this year!** See here for all the details, including the incredible feedback from the January class.

Also, as for our Immersion Events; our Fall schedule is already packed with events and engagements. As a result, there will be **no October/November in-person classes in Chicago this year** so in the U.S. the only in-person classes will be in Chicago in April/May and Bellevue, WA in June. This means that each of **our in-person classes will only be offered once in the U.S. in 2018** (with an extra IEPTO1 in June in Bellevue), and these classes are all filling up quickly – please don't wait to the last minute to register otherwise you'll be disappointed.

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

SQLskills News

Our Spring SQLintersection conference is next week, and we have a phenomenal line up of workshops, sessions, and speakers. Check out this blog post for all the details, and use the discount code 'sqlskills' when you register to save $50.

In-person US classes: In 2018, we're offering our usual Immersion Events on Performance Tuning (IEPTO1 and IEPTO2) and for the Accidental DBA (IE0), as well as PowerShell, Azure, Clustering and Availability Groups, BI strategies, and Practical Machine Learning. See here for our 2018 Immersion Event class schedule. As noted above, there will be no Fall in-person classes in the U.S. in 2018.

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 and Azure VMs) and **IECAG** (on clustering and availability groups). See here for details.

Finally, even if you can’t join us in person, I’ve put out a call for 2018 remote user group sessions. In 2017, we did more than 100 of these around the world and we have set up more than 50 for 2018 already! If you’d like one of us to present for your user group, check out my blog post here.

Book Review

Two books I read while on vacation are each the start of classic sci-fi series; I can’t believe I’ve read neither of them until now.

The first book is Isaac Asimov’s *Foundation*. This is Asimov's classic story of the Foundation; ostensibly created to preserve mankind's knowledge once the Empire collapses, but actually the seed of the next Galactic Empire and the brainchild of Hari Seldon who can see the future through mathematically extrapolating from the present. Well written and gripping - highly recommended for sci-fi fans!

The second book is Frank Herbert’s *Dune*. The first book in the Dune series sees House Atreides take over the desert planet Arrakis (aka Dune) from House Harkonnen and then lose it amidst treachery and murder. Paul Atreides then becomes the leader of the native Fremen people and also becomes the legend they've been waiting for, plus all the drama with the giant sand worms and the priceless spice.

Both are excellent and highly recommended for sci-fi fans!

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.

Glenn recently had a client who had migrated from SQL Server 2012 to SQL Server 2016 on a relatively ancient two-socket server (an early 2011-vintage, HP Proliant DL380 G7 with two Intel Xeon X5687 processors and PCIe 2.0 x8 expansion slots).

They were experiencing fairly high write latency (65-70ms) to their tempdb data files, which is a common problem that we see with many SQL Server instances, regardless of what version of SQL Server is involved. In this case, they had tempdb located on two 200GB HP 6Gbps SAS MLC SSDs in a RAID 1 array.

Glenn recommended that they consider purchasing two 375GB Intel Optane SSD DC P4800X PCIe NVMe AICs that would be used in a Windows Storage Spaces RAID 1 array, to host their tempdb data and log files.

Before deploying the Intel Optane cards to production they ran a pretty extensive set of Microsoft DiskSpd tests against an identical server, comparing the performance of their old SAS MLC SSDs to the new Intel Optane storage cards. They compared sequential throughput, IOPS, and latency with many different test types, and they saw anywhere from a 10x to a 30x improvement compared to their legacy SAS SSDs.

Once they moved to Production, their tempdb data file write latency went from 65-70ms down to 0.2ms, and they saw a very noticeable improvement in their overall workload performance. Keep in mind that this was in a pretty old server that only has PCIe 2.0 x8 support.

These cards are not that expensive (about $1,600 each), and they are a great solution for heavy tempdb usage.

Bottom line: "Killing a problem with iron" is always a valid option to consider. And, sometimes throwing hardware at the problem is the most cost-effective way to vastly change the performance of a workload. Architecture is always a better long-term solution but can be costly in terms of development, implementation, and migration; you should always consider the reasonable (and sometimes quick) hardware solutions to give you more immediate gains.

Paul's Ponderings

Last week I was involved in an email thread where people were wondering about some 'weird' behavior from SQL Server. The problem was occurring on SQL Server 2016 and used the default isolation level of read committed. The scenario was the following:

- Create a table with a few columns
- Batch 1: In one SSMS window, do the following (which takes 10 s to run):
  - Start a transaction
  - Insert 1,000 rows into the table, with a 0.01s gap between each insert
  - Commit the transaction
- Batch 2: In a second SSMS window:
  - Select * from the table

The 'weird' behavior is that when the "Batch 2" select completes, after having been blocked by the "Batch 1" transaction, it doesn't return all 1,000 rows (even though "Batch 1" has completed). Furthermore, depending on ***when*** during the 10-seconds that "Batch 1" executes, the "Batch 2" select is initiated, "Batch 2" returns different numbers of rows. This behavior had also been reported on earlier versions of SQL Server as well. It's easy to reproduce on SQL Server 2016/2017 and can be reproduced in all earlier versions with a single configuration change (more details in a moment).

Conversely, if the table has a clustered index created, 1,000 rows are returned every time, on all versions of SQL Server.

So why is this weird? The problem is that many expect all 1,000 rows to be returned every time AND that the structure of the table or the version of SQL Server should not make any difference.

Unfortunately, that assumption is not correct when using read committed. The read committed isolation level guarantees that only committed data will be read; it does NOT guarantee how much of the committed data will be returned and is prone to inconsistent analysis problems. If you want to avoid inconsistent analysis, you'd need to increase your isolation level (if using locking) or change to versioning for read committed (or even snapshot isolation).

I do agree that the assumption is reasonable.

The definition of the requirement is straightforward however. SQL Server guarantees that only committed data will be read, not how much of it will be read, and so the amount of committed data returned in this case is variable. Furthermore, depending on the table structure, SQL Server version, and configuration option (*which I've mentioned twice now, but haven't explained as I don't want to spoil my story…*), the number of rows returned will vary wildly.

So, what's going on? Why the discrepancy in the number of rows?

It's because the table is a heap.

The 'weird' behavior manifests itself when the heap has an extent allocated to it immediately, from which the first data pages are allocated. When the allocation-order scanner for the select starts, it looks at the PFS bytes for the 8 pages in the extent to see which ones are allocated. These will be scanned. Depending on when the select starts within the 10 seconds that "batch 1" executes, there will be more rows or fewer rows read by the scanner (and returned in the "batch 2" results) because more or fewer of the PFS byte 0x40 bits will have been set indicating the page in the extent is allocated.

For a table to have an extent allocated to it immediately, mixed pages have to be disabled, which is the default for SQL Server 2016 onward, which is why the 'weird' behavior manifests with a small number of rows on SQL Server 2016. However, if you've enabled trace flag 1118, you'll see this behavior in all versions. This is why we're seeing some people report the 'weird' behavior on versions earlier than SQL Server 2016.

When mixed extents are NOT disabled, i.e. in earlier versions than 2016 when 1118 is not enabled, the first 8 pages allocated are mixed pages, and so the allocation order scanner has to pick them up individually and sees them all (without going into details of the synchronization around the single-page slot array on the first IAM page in the IAM chain/allocation unit for the table), and so as long as no more than 8 data pages are used by the insert, all the rows on them will be returned by the select in this example.

So there you have it, the behavior is "by design" and hasn't changed at all, but is understandably confusing unless you know exactly how the Engine is performing the select under the covers.

And if you create a clustered index, the 'weird' behavior doesn't occur for this example in any version. This is because the select gets hung up on the S(hare) row lock for the first row in the clustered index, and by the time it gets it, the insert has completed and the index leaf scan picks up all 1,000 rows.

Don't think that using a clustered index stops 'weird' behavior using read committed; it doesn't. However, it can reduce some forms of inconsistencies. In this old blog post, I demonstrate a scenario where a scan of clustered index returns four rows when only three rows exist, using read committed. And, Kimberly also describes and shows "inconsistent analysis" in these blog posts:

- Locking, isolation, and read consistency
- Inconsistent analysis in read committed using locking

Call to action: The read committed isolation guarantees that only committed data is read; however, there are many forms of potential concerns (known as "inconsistent analysis"). It really pays to understand what these are and what the various isolation levels that SQL Server provides do and do not allow. In addition to the blog posts above, check out the MSDN lesson titled: *Understanding the Available Transaction Isolation Levels* here.

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.

New Details about 2nd Generation AMD Ryzen CPUs

Some pretty detailed specifications have leaked out about AMD's upcoming Ryzen 2000 series "Pinnacle Ridge" desktop processors that are scheduled to launch on April 19, 2018. These are part of the 12nm Zen + architecture, and there will be four specific SKUs at launch.

These will include the AMD Ryzen 7 2700X, AMD Ryzen 7 2700, AMD Ryzen 5 2600X, and AMD Ryzen 2600. The two Ryzen 7 models will have 8 physical cores plus SMT (which is AMD's version of hyper-threading), while the two Ryzen 5 models will have six physical cores plus SMT.

These new CPUs will work in existing AMD Socket AM4 300-series motherboards with a BIOS update, plus there will be new AMD Socket AM4 400-series motherboards with additional features. The main new feature for enthusiasts is called "Precision Boost Overdrive" which lets the motherboard more aggressively ramp up the clock speed on more cores simultaneously.

These new processors will let you build a competitively powerful gaming system or desktop workstation for less money than a roughly equivalent Intel-based system. The continued improvements from AMD across all of the market segments will keep the pressure on Intel, which is good for us as consumers.

Comparing the Costs of Cloud Hosting to Self-Hosting

Patrick Kennedy from Serve the Home (STH) has a pretty interesting post detailing his experience moving the infrastructure behind Serve the Home from Amazon Web Services (AWS) to a local web hosting data center.

Some of the more interesting points include how reliable modern rack-mounted server hardware has become, with very few component failures, and how reliable enterprise SSDs have become (with annual failure rates of less than 0.2%), which has dramatically reduced the number of visits to the remote data center.

It is also notable that STH has gotten significantly better VM and storage performance from self-hosting compared to their AWS experience. Storage performance is still the biggest weakness of cloud hosting.

For many small to medium organizations, it may still make more economic sense to self-host in a typical web hosting data center, as long as **they buy or lease their own hardware and storage** rather than renting it from the hosting providers. In my experience, many web hosting companies do not have the best hardware and storage choices for SQL Server, and they also overcharge for the monthly rent of their hardware and storage.

The main takeaway here is that you should actually run the numbers and see what makes sense for your organization.

#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.

The theme for the TBT this time is database mirroring. Although database mirroring has been deprecated, it's still in wide use and it's still a very easy way to perform a rolling upgrade or move data around. Here are some useful links:

- Glenn's Pluralsight course: *SQL Server: Understanding, Configuring and Troubleshooting Database Mirroring*
- Whitepapers:
  - *Database Mirroring and Log Shipping Part I – Prescriptive Guidance*
  - *Prior Clustering and Mirroring Deployments Part I – Prescriptive Guidance*
  - *Proven SQL Server Architectures for High Availability and Disaster Recovery* (written by me)
  - *SQL Server Replication: Providing High-Availability Using Database Mirroring* (written by me)
  - *Database Mirroring and Log Shipping Working Together*
  - *Database Mirroring: Alerting on Database Mirroring Events*
  - *Database Mirroring: Best Practices and Performance Considerations*

- KB article *Things to consider when setting up database mirroring in SQL Server*
- My blog posts:
    - *Importance of network latency when using database mirroring*
    - *Importance of monitoring a database mirroring session*
    - *Database mirroring blog category*

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

- Glenn: *SQL Server Diagnostic Information Queries for March 2018*
- Glenn: *New CPU Microcode Updates for Intel Processors*
- Erin: *Query Store Requests*
- Erin: *Query Store Fix in SQL Server 2017*

I hope you find these useful and interesting!

Video Demo

This is the last in the 3-part series on statistics enhancements in SQL Server 2016/2017. Ever wondered what statistics the optimizer considers when generating an execution plan? You've been able to find that information using trace flags, but did you know that it's included in the execution plan starting in SQL Server 2017? Erin explores what statistics data shows up in the plan in this week's Insider video.

The video is just over 8.5 minutes long and you can get it:

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

And the demo code is here.

Enjoy!

Upcoming SQLskills Events

We have lots of events coming up in 2018 – from our online IEVLT course AND SQLintersection in March to our own LIVE, in-person Immersion Events in both the U.S. and London; all events are open for registration. Every event has a different focus and different benefits – from deep-technical training in our online courses and in-person IEs to wide-ranging topics at SQLintersection where you can learn more effectively how to keep moving forward in both your database and your career! And, of course, one benefit all our in-person events provide is networking.

Note: there will be **no October/November in-person classes in Chicago this year** so in the U.S. the only in-person classes will be in Chicago in April/May and Bellevue, WA in June. This means that each of **our in-person classes will only be offered once in the U.S. in 2018** (with an extra IEPTO1 in June in Bellevue), and these classes are all filling up quickly – please don't wait to the last minute to register otherwise you'll be disappointed.

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

Online, March 2018

- **IEVLT**: Immersion Event on Very Large Tables: Optimizing Performance and Availability through Partitioning **(Final planned delivery in 2018)**
  - March 20-22

Orlando, FL, March 25-28, 2018

- SQLintersection co-located with the DEVintersection conferences (register with the 'sqlskills' discount to get save $50 on registration). See here for details.

Chicago, IL, April/May 2018

- **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1
  - April 23-27
- **IE0**: Immersion Event for the Junior/Accidental DBA
  - April 23-25
- **IEUpgrade**: Immersion Event on Upgrading SQL Server
  - April 23-25
- **IECAG**: Immersion Event on Clustering and Availability Groups
  - April 26-27
- **IEAzure**: Immersion Event on Azure SQL Database and Azure VMs
  - April 26-27
- **IEPTO2**: Immersion Event on Performance Tuning and Optimization – Part 2
  - April 30-May 4
- **IEBIStrat**: Immersion Event on Developing a BI and Analytics Strategy **(** NEW **)**
  - April 30-May 2
- **IEPS**: Immersion Event on PowerShell for SQL Server DBAs
  - April 30-May 2

- **IESSIS1**: Immersion Event on Learning SQL Server Integration Services
    - May 7-11
- **IEPML**: Immersion Event on Practical Machine Learning
    - May 7-11 **(** NEW **)**

Bellevue, WA, June 2018

- **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1
    - June 18-22

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
- **IECAG**: Immersion Event on Clustering and Availability Groups
    - September 12-13
- **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