(July 5th, 2022)

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!

This newsletter is coming to you from Camp Savage, where I've been working with my tractor and building new machinery. I was very humbled to have been awarded as a Data Platform MVP again today, making this my 15th year as an MVP.

Note that the summer sale ends this Friday, and I'll be taking the usual summer break from the newsletter as we're heading to Scotland next week! Have a great summer!

SQLskills News

It's here – the Summer School Special Savings SALE!

Starting just in time for summer, we are offering fantastic prices to keep you learning and give you the summer for free! Get current and sharp on SQL Server 2019 so that you're ready to learn about SQL Server 2022 when it comes out. These SUPER LOW (**AT LEAST 80% OFF**) prices are for all of our recorded courses and bundles. Our "Summer School Special" includes extra time for you to really dive in as you'll get THE SUMMER FOR FREE! Recordings are usually accessible for 1 year from the date of purchase. For purchases made during the sale, your access won't expire until **Sept 30, 2023**!

The sale lasts through the end of this **Friday, July 8**. Each course is available individually but there are also even better budget-friendly bundles.

These prices are so low… there are no additional discounts available. If your team wants to bulk purchase courses or bundles, we can offer an easier way to pay (one bill and then each attendee sets up their accounts after payment is received). Orders for individuals are available for credit card payment only. Send Training an email to ask about a bulk purchase order.

Here's what we have to offer (all prices in USD):

- IE0: Immersion Event for the Accidental and Junior DBA
  - Recorded on SQL Server 2019 over 8 half days for over **32 HOURS** of content
  - LIVE (normal price $3,495) and 1 year access recordings for $1,695
  - **Summer School Savings SALE price: only $699 (80% OFF)**

- IEAzure: Immersion Event on Azure SQL Database, Azure VMs, and Azure Managed Instance
    - Recorded on SQL Server 2019 over 6 half days for over **22 HOURS** of content
    - Regularly priced at $3,295 LIVE and 1 year access recordings for $1,695
    - **Summer School Savings SALE price: only $659 (80% OFF)**
- IEPTO1: Immersion Event on Performance Tuning and Optimization – Part 1
    - Recorded on SQL Server 2019 over 10 half days for over **43 HOURS** of content
    - Regularly priced at $3,995 LIVE and 1 year access recordings for $1,995
    - **Summer School Savings SALE price: only $799 (80% OFF)**
- IEPTO2: Immersion Event on Performance Tuning and Optimization – Part 2
    - Recorded on SQL Server 2019 over 10 half days for over **40 HOURS** of content
    - Regularly priced at $3,995 LIVE and 1 year access recordings for $1,995
    - **Summer School Savings SALE price: only $799 (80% OFF)**
- IECAG: Immersion Event on Clustering and Availability Groups
    - Recorded on SQL Server 2019 over 4 half days for over **18 HOURS** of content
    - Regularly priced at $1,695 LIVE and 1 year access recordings for $845
    - **Summer School Savings SALE price: only $339 (80% OFF)**
- IEQS: Immersion Event on Solving Common Performance Problems with Query Store
    - Recorded on SQL Server 2019 over 3 half days for almost **10 HOURS** of content
    - Regularly priced at $995 LIVE and 1 year access recordings for $495
    - **Summer School Savings SALE price: only $199 (80% OFF)**
- IEVLT: Immersion Event on Very Large Tables: Optimizing Performance and Availability through Partitioning
    - Recorded on SQL Server 2019 over 4 half days for almost **14 HOURS** of content
    - Regularly priced at $995 LIVE and 1 year access recordings for $495
    - **Summer School Savings SALE price: only $199 (80% OFF)**
- IEQuery: Immersion Event on Fixing Slow Queries, Inefficient Code, and Caching/Statistics Problems
    - Recorded on SQL Server 2017 over 3 half days for over **11 HOURS** of content
    - Regularly priced at $695 LIVE and 1 year access recordings for $395
    - **Summer School Savings SALE price: only $139 (80% OFF)**
- IETLB: Immersion Event on Transactions, Locking, Blocking, Isolation, and Versioning
    - Recorded on SQL Server 2017 over 3 half days for over **13 HOURS** of content
    - Regularly priced at $695 LIVE and 1 year access recordings for $395
    - **Summer School Savings SALE price: only $139 (80% OFF)**
- IECS: Immersion Event on Columnstore Indexes
    - Recorded on SQL Server 2017 over 3 half days for almost **11 HOURS** of content
    - Regularly priced at $695 LIVE and 1 year access recordings for $395
    - **Summer School Savings SALE price: only $139 (80% OFF)**

And we have four bundles:

SQL Server Jumpstart Bundle

Courses: IE0 + IEQuery + IETLB:

Summer School Savings price: only $749

***** 85% SAVINGS off of full price PLUS 3 months FREE *****

SQL Server Jumpstart+ Bundle

Courses: IE0 + IEQuery + IETLB + IEQS + IEVLT + IECS

Summer School Savings price: only $1,210.50

***** 85% SAVINGS off of full price PLUS 3 months FREE *****

Performance Tuning Bundle

Courses: IEPTO1 + IEPTO2: **Summer School Savings price: only $1,198.50**

***** 85% SAVINGS off of full price PLUS 3 months FREE *****

Complete Short Course Bundle

IEQuery + IETLB + IEQS + IEVLT + IECS: **Summer School Savings price: only $686.25**

***** 85% SAVINGS off of full price PLUS 3 months FREE *****

BlackBelt Bundle

IEAzure + IEPTO1 + IEPTO2 + IECAG + IEQS + IEVLT + IECS: **Summer School Savings price: only $2,125.00**

***** 87% SAVINGS off of full price PLUS 3 months FREE *****

PASS Data Community Summit Workshops

Both Kimberly and I are presenting pre-con workshops in Seattle at PASS in November, and we'll both be on stage for each other's workshop to help with questions.

- Monday, 11/14: Paul: Performance Troubleshooting with Waits and Latches
- Tuesday, 11/15: Kimberly: Indexing for Performance

Check out the details here.

SQLskills Insider Sessions

We wrote about the motivation behind our Insider Sessions and you can register for just one, or the entire series here: https://www.SQLskills.com/iSessions. We've started them back up in 2022 but we've cut back on the frequency. **But, the best news is that we've posted all of the prior sessions online for streaming. For free!** Just go to the completed sessions and you'll find all of the resources, and the videos! Enjoy.

Our thought: block some time – EVERY WEEK! **Dedicate 2 hours a week to Improving Your SQL skills with SQLskills**! Join us when it makes sense (hopefully always! ☺) or, read blog posts, review topics of interest, go spelunking in the documentation on a topic that's always interested you. Stay fresh, stay current – stay ahead of the competition!

Block some time NOW, register online, and when a session is planned to run, we'll send you a meeting link with joining instructions.

Book Review

The latest book I've read is Winston S. Churchill's *The World Crisis Volume I: 1911-1914*. As as I've said here before, I love Churchill's writing style so I won't repeat myself on that front. This is the first in his five-volume history of WWI and it's fascinating because he was there and he was integral to the prosecution of the naval war against Germany during the opening year of the conflict, as First Lord of the Admiralty. It goes into much more depth than other books on the intricacies of the Admiralty and the thinking behind the various naval maneuvers and stratagems. Highly recommended for history buffs!

The Curious Case of…

This section of the newsletter explains recent problems we've helped with on client systems or been asked about over email or #sqlhelp; they might be something you're experiencing too.

On Twitter this morning someone was asking about shrinking a giant log file. You can never shrink the log smaller than the first two VLFs, so if they're multi-GB each, you're kind of stuck. There is one hack you can use though, which is to make use of the bug when reverting to a database snapshot where it deletes the log and creates a new 0.5MB one. You can read about that here… and beware that doing so breaks your log backup chain.

Ponderings…

(From me this time. I was contacted by two random people from different companies last week with the same question! I wrote a long editorial explaining the problem back in 2015, so here it is again – enjoy!)

One of the email questions I received last week was on something I've seen a few times before and I thought it would make a good editorial for this week. In a nutshell, the question is: *When*

our log shipping secondary is applying log backups, sometimes it takes a lot longer than usual. Any ideas why this might be the case?

Log shipping has been around forever, and it's still a hugely applicable and useful feature for very simply maintaining one or more secondary copies of a database. You can also use a secondary copy for reporting, where the restore of the log backup uses the WITH STANDBY option, leaving the database in an accessible, but read-only state (when the logs aren't being applied).

This works as follows:

1. Make sure all users are disconnected from the database
2. Write all the log records from the backup into the database's log file
3. Perform the REDO part of recovery (ensuring that all operations from committed transactions are present in the database)
4. Perform the UNDO part of recovery (ensuring that all operations from uncommitted transactions are not present in the database)

Step 4 writes all the log records generated by the UNDO operations into a special file called the undo file. This means that the database is in read-only mode and is transactionally-consistent so that users can access it. The reason the log records are written into the undo file is so that the transaction log of the database is not altered in any way, allowing subsequent log backups to be restored. If this weren't the case, the UNDO log records would advance the database's LSN, meaning that subsequent log backup restore operations would fail.

When the restore process begins on the secondary database, if an undo file exists, there is another step that is performed before steps 2-4 above. This additional step needs to take all the log records in the undo file and undo the effects of them – essentially putting the database back into the state as of the end of step 3. This database state is the same as if the previous log backup had been restored using *WITH NORECOVERY* instead of *WITH STANDBY*.

The occasional long-running restore problem happens when a log backup is restored that contains a long-running transaction that does not commit before the end of the log backup. This means that it must be completely undone as part of restoring the log backup (step 4), resulting in a very large undo file. This in itself can make restoring a log backup take a lot longer than usual.

When the next log backup is restored, the additional step that undoes all the log records in the undo file has a very large undo file to process and takes much, much longer than usual. And if the log backup being restored *also* has an uncommitted, long-running transaction then it's the perfect storm as the step 4 will also take a long time.

The situation where I've seen this is when the primary database is undergoing index maintenance and a log backup finishes near the end of a very long-running index rebuild operation of a large

clustered index. The initial restore of that log backup on the secondary database takes much longer than usual to complete because of step 4 in the restore process. The next log backup on the primary also completes just before an index rebuild completes. When it is restored on the secondary, the whole of the large undo file has to be undone again, then the log restore occurs, and then another large undo file is generated to undo the second uncommitted index rebuild.

This is a possibility you have to be aware of if the secondary database must be available 24x7 for reporting, with only minimal downtime when each log backup is restored. In that case I would carefully augment the index maintenance operations on the primary with log backups to ensure that only complete, committed index rebuilds are present in the log backups being restored on the secondary database.

An alternative would be to move from log shipping to database mirroring or availability groups, where the log records are continually being sent from the principal to the mirror database (or primary to secondary replica databases, in availability group terms) and there are no extra steps involving undoing log operations multiple times.

With database mirroring, the drawback of this is that reporting would have to use database snapshots, so there's a complexity trade-off involved. With availability groups, the reporting would have to use a readable secondary, which can lead to index fragmentation on the primary replica (which I talked about in the newsletter last October), but that can be surmounted.

Call to Action: Not really a call to action this time, but this is yet another example where understanding how SQL Server performs common operations can make it much easier to diagnose performance problems.

#TBT

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

The #TBT this time is around stored procedures, so here are some resources for you:

- Kimberly's post: *SQLskills SQL101: Stored Procedures*
- Kimberly's course: *SQL Server: Optimizing Stored Procedure Performance*
- Kimberly's course: *SQL Server: Optimizing Stored Procedure Performance – Part 2*
- Kimberly's *Optimizing Procedural Code* blog post category

And blog posts since the last newsletter:

- Paul: *An Introduction to SQL Server Logical Joins* (on the SentryOne blog)

I hope you find these useful and interesting!

Video Demo

For this newsletter I've pulled a demo from Erin's Pluralsight course *SQL Server: Understanding and Using DBCC Commands*. In the demo, Erin shows how to use the *DBCC CHECKCONSTRAINTS* command that allows you to valid one or more constraints. Any time a repair is run against a database, if a table that was repaired has any (or is involved in any) constraints, this command must be run to check the constraint.

The video is about 3.5 minutes long and you can get it in MOV format here.

The demo code is available here.

Enjoy!

Upcoming SQLskills Events

We're not running any public classes this year, instead we're waiting for SQL Server 2022 to be released and we'll run classes in 2023.

With our new streaming system, you can now choose to attend a live, online event or purchase a recording to watch at your leisure, either individually or as part of a bundle. And all attendees of live events get lifetime access to the class recordings too!

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

You can get all the details on our training options page or just go directly to our new shop.

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