(July 30th, 2019)

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 Redmond, where Kimberly and I just celebrated our 12th wedding anniversary yesterday – woohoo! I hope you're all enjoying your summer (or winter if you're in the southern hemisphere) – the weather here is just glorious so I decided to skip the July 16th newsletter too.

SQLskills News

Live, ONLINE classes: we have three classes coming up in the remainder of the year:

- IETLB: Transactions, Locking, Blocking, Isolation, and Versioning – October 1-3
- IEVLT: Very Large Tables: Performance/Manageability through Partitioning – October 29-31
- IECS: Columnstore Indexes – November 12-14

The classes are US$595 each, with a US$495 early-bird price through the end of August, and there's an all-class-combo deal for US$1,350, which makes each class US$450. You can get all the details and registration information through the class schedule page here.

Live, IN-PERSON classes: we have two more in-person classes this year, in Chicago in October:

- IEPTO1: Performance Tuning and Optimization – Part 1 – October 7-11
- SOLD OUT: IEAzure: Azure SQL Database, Azure VMs, Azure Managed Instance – October 7-10

These will be our only, in-person Immersion Events running this Fall and you can get all the details and registration information through the class schedule page here.

Finally, even if you can't join us in person, I've put out a call for **2019 remote user group sessions** and we've got 43 scheduled this year already! If you'd like one of us to present for your user group, check out my blog post here.

Book Review

The most recent book I've read is Edward Wilson-Lee's *The Catalogue of Shipwrecked Books: Christopher Columbus, His Son, and the Quest to Build the World's Greatest Library*. This is really a biography of Hernando Colón, the natural son of Christopher Columbus, who labored throughout his life to preserve the good name (at the time) and accomplishments of his father. He set out in the early 1500s to build the biggest library in the world, holding everything that has ever been printed, but focusing more on pamphlets and pulp writing and not just on traditional books on learned topics. He ended up with a 15-20,000 volume collection at the time of his death, with about 4,000 books remaining to this day in the collection at the Seville Cathedral (dwarfing my own 2,500 collection). Apart from just collecting a lot of books, Hernando pioneered methods for indexing the books, so people could actually find relevant titles in the library. The title of the book comes from a portion of the index that described almost 1,700 of his books that were lost when a cargo ship sailing to Spain sank - a huge blow! Very interesting and 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.

Do you know why the transaction log can't use instant file initialization?

Read the explanation in my blog post here.

Ponderings…

(This time from Tim… Enjoy!)

Testing should be a key function of every technologist's job. You should be testing any change that is planned for a production system, as by testing the impact of the upcoming change, you can better determine the overall effect prior to it actually affecting production. The goal is to discover any negative or adverse effect the change will have so that it doesn't cause a surprise degradation in production performance and/or functionality. Unfortunately, this is often not what happens in the real world.

We've had clients impacted severely from doing any of the following without adequate testing (and this is by no means an exhaustive list!):

- Upgrading SQL Server version
- Moving to new hardware
- Applying a service pack or cumulative update
- Deploying new code changes to their vendor application
- Deploying in-house code updates to production

- Changing compatibility level

Testing is very important to ensure stability in systems and provides two key validation points: make sure the change you are planning works as designed and make sure that there aren't negative impacts overall. Too many times we hear things like "That change shouldn't have anything to do with the system being slow/down/broken". In cases where it is easy to back out the change, that is typically a good place to start and check if that change was the impact. Other times, the change was too invasive or would cause an even bigger impact to rollback so all that can be done is to triage the current state and try to mitigate the impact.

Often testing isn't done because people don't really know *how* to test. In order to test code changes, you need another environment (often development and sometimes a dedicated QA environment) other than production to test the changes in. You also need performance baseline and a representative workload to replay against that environment. Many organizations also have a run-book of basic functionality that they go through to make sure all the screens and processes function as expected.

How much testing is enough? That's up to each organization to determine, but there's also the extent of the change to take into consideration. For instance, if you are upgrading the version of SQL Server, then really extensive testing will be needed, including replaying workload captures from varying times during the day, running business critical ETL jobs, and more. An application upgrade from the vendor should also be considered a major change and involve replaying workloads, lines of business stepping through the application looking for functionality changes, and more.

What about something simpler such as making a change to a stored procedure or creating/dropping indexes? This kind change might seem simple but can also have positive or negative impacts on production so I would strongly encourage testing these changes too if you can. Restore your database to development, make your changes, replay the workload, and compare the changes to the performance baseline to ensure you are getting the positive result you are hoping for in production. Testing definitely adds time to projects and code releases, however it also helps to ensure that you have a more stable and reliable production environment and can end up saving time in the long run.

Microsoft provides many resources for customers to perform adequate testing. Developer editions of SQL Server are free, Distributed Replay and Database Experimentation Assistant (which leverages Distributed Replay) exist for workload capture and replays, and for compatibility testing of versions there is the Database Migration Assistant.

For proper testing you need to duplicate the compute and storage environments, which is one of the main blockers for technologist to have proper development environments. Fortunately,

virtualization makes it easier for organizations to spin up environments for testing, and then destroying the environment to reclaim the compute and storage. We've also seen a big uptick in organizations leveraging Azure for the compute and storage resources for development.

Call to action: Regardless of your industry, testing should be a fundamental part of your change control process. The tools exist to help simplify the process as well as making it more repeatable. Take the time, build the test environments, and define a proper testing policy for any production code changes. Reputational loss due to outages and performance issues are difficult to overcome.

Glenn's Tech Insights

Recent news and views from the hardware and Windows worlds that we think will be interesting to SQL Server community members.

In Glenn's regular roundup of tech news this time he talks about the following:

- In this post:
  - Management Studio 18.2 release
  - SQL Server 2014 cumulative updates
- In this post:
  - July release of Azure Data Studio
  - Security updates for SQL Server
- In this post:
  - Leaked internal Intel memo about AMD
  - SQL Server 2019 CTP 3.1

Check it all out!

#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

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

- Glenn: *Building an AMD Ryzen 7 3700X Desktop Machine*
- Glenn: *Diagnostic Information Queries for SQL Managed Instance*
- Glenn: *Is Intel Doomed in the Server CPU Space?*
- Glenn: *Presenting at PASS Summit 2019*
- Glenn: *SQL Server Diagnostic Information Queries for July 2019*
- Erin: *Troubleshooting a Change in Query Performance*
- Erin: *Important Query Store Fixes – Summer 2019*
- Erin: *Query Store Best Practices*
- Tim: *Common SQL Server Mishaps*
- Jonathan: *Unintended Side Effects – Sleeping Sessions Holding Locks*

I hope you find these useful and interesting!

Video Demo

The SQL Server team introduced the Query Tuning Assistant with Management Studio 18.0, a tool which uses Query Store to capture query metrics before and after a change to compatibility mode. This tool is something you can use for the critical task that Tim talked about: testing upgrades and changes. In this week's video we'll walk through how to use QTA – a tool that anyone running a lower compatibility should leverage!

The video is about 12 minutes long and you can get it here.

The demo code zip is here.

Enjoy!

Upcoming SQLskills Events

Our set of 2019 live, in-person classes for Chicago in October are filling up fast and we've announced our Fall live, online events too.

You have multiple learning opportunities as 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, IN-PERSON Immersion Events:

Chicago, IL, October 2019

- **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1
  - October 7-11
- **IEAzure**: Immersion Event on Azure SQL Database, Azure VMs, and Azure Managed Instance
  - October 7-10 **SOLD OUT!!**

LIVE, ONLINE Immersion Events:

- **IETLB**: Immersion Event on Transactions, Locking, Blocking, Isolation, and Versioning
  - October 1-3
- **IEVLT**: Immersion Event on Very Large Tables: Performance/Manageability through Partitioning
  - October 29-31
- **IECS**: Immersion Event on Columnstore Indexes
  - November 12-14

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