(February 27th, 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,

This newsletter is coming to you from Redmond, where I've managed to survive two weeks without further injuring myself in any lack-of-common-sense incidents ☺

We're looking forward to seeing a bunch of you this year in our classes in Chicago in April/May, our classes in Bellevue in July/August, and our Spring SQLintersection conference in Orlando in May. See here for the class schedule and here for SQLintersection details. Note that IEPTO1 and IE0 in Chicago are already sold out and all other classes are 1/2 to 2/3 full. I'm particularly pleased that we have a **three-day class on using PowerShell** to administer SQL Server, taught by industry-expert, MCM, and MVP, Ben Miller.

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**.

And even if you can't join us in person, we're still taking requests for remote sessions for this year. We have 50 scheduled so far; if you'd like one of us to present for your user group, check out my blog post here.

I haven't finished a book since the last newsletter, so I want to tell you about an interesting magazine I've just subscribed to: *Lapham's Quarterly*. It's an interesting concept. Each issue focuses on a single topic, and has curated essays and articles from throughout history on that topic, along with images and paintings, and in a super-high quality format. I'm sure some of my friends here would enjoy it. You may think it's a little pricey, at $40 for a year-long subscription of 4 issues, but at 200 pages per issue, that's no more expensive than a regular book.

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

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.

I was working with a former Immersion Event student last week who had a problem after a mirroring failover. They also had transactional replication configured on the mirroring principal and after the failover, the Log Reader Agent job (on the new principal) was failing.

It turns out that someone had enabled trace flag 1448. This allows the Log Reader Agent on the principal to send committed transactions to the Distributor even if those transactions have not yet been hardened on the mirror. This happened in their environment. After the failover, when the Log Reader Agent job started running on the new principal, it found that there were transactions that the Distributor had that didn't exist on the new principal, and so it failed.

This can be a tricky situation from which to recover as the Subscribers can end up having more data than the Publisher! I discuss all of this in great depth in the whitepaper I wrote *SQL Server Replication: Providing High Availability with Database Mirroring*, which is still perfectly applicable today. Incidentally, the behavior of this trace flag also applies to Availability Groups, but with the difference that the Log Reader Agent will always wait for a synchronous replica before continuing – see this MSDN page for more details.

Having said that, the real problem here was that nobody knew why that trace flag had been enabled or the effects of having it on. We often see clients with trace flags enabled where the ramifications are not understood properly. Whether you read about them on the internet or a consultant or colleague recommends them, be sure to document, test, and regularly check on the status of these trace flags. Some should only be used temporarily and in the long term may be problematic.

Bottom line: make sure you understand what all the trace flags you're using actually do. There are two good resources for looking up trace flags:

- A Topical Collection of SQL Server Trace Flags
- SQL Server Trace Flags

Paul's Ponderings

One thing that many people don't take into consideration when comparing performance between servers is the raw single-threaded performance of the underlying processors.

As an example, we had a client recently that had a process that was taking longer on a production system that on a development system and they couldn't figure out why.

We ran an Extended Events session to see what statements were running as part of the process and how long each was taking, and used PerfMon to track the resources that were being used. The PerfMon data ruled out a resource issue as neither machine was limited by memory and neither machine had CPU throttled. Also the disk latency was comparable for both systems, and they shared the same storage.

On comparing the statements from the Extended Events output, we found that there was a set of statements that consistently ran slower in production than in development – bulk inserts – with the difference sometimes being as high as two minutes slower in production!

We then ran the single slowest bulk insert against development and production, with an Extended Events session that just captured the waits for that statement. In both cases the waits added up to less than 5 seconds.

We then checked the processors on both system and asked Glenn to compare them. Glenn explained that there was pretty big difference in single-threaded performance (and of course the bulk insert is single threaded) between the two processor models. The development system had newer, faster processors (Haswell-EP), with faster base clock speed, and using faster DDR4 memory instead of DDR3 memory.

Based on metrics that Glenn uses for processor comparisons (with higher score per core being better), the development system had Xeon E5-2650 v3 processors with a score of 104.78 per core, while the production system had Xeon E5-2640 v2 processors with a score of 79.96 per core.

So assuming all other things being equal (which they basically were in this case), single-threaded processor performance often controls how long a query is going to take to execute, as was the case here. They'd bought a newer system for development and were trying to compare performance between the two systems, wasting a lot of time trying to figure out what was wrong in production.

Call to action: It's really important to bear in mind the different processors when trying to compare performance between servers, especially if everything else seems exactly the same and you're frustrated by difference performance. This also means you have to choose new processors carefully too. Glenn's written a lot on this on his blog, and in this newsletter, most recently here.

Video Demo

With the availability of In-Memory OLTP in Standard Edition of SQL Server 2016 SP1, more of you might be thinking about what data and processes you could put in memory. As always, data to support your thoughts and ideas is critical before implementing anything in production. In this Insider Video Erin goes through a simple testing methodology that you can use or work from to test whether code and data from your environment is a candidate for In-Memory OLTP.

The video is about 14.5 minutes long and you can get it:

- In WMV format here
- In MOV format here

And you can get the demo code here.

Enjoy!

SQLskills Offerings

The classes for both Spring and Summer 2017 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

Upcoming Immersion Events

Chicago, IL, April/May 2017

- **IE0**: Immersion Event for Junior/Accidental DBAs
  - April 24-26 ****SOLD OUT****
- **IEPTO1**: Immersion Event on Performance Tuning and Optimization – Part 1
  - April 24-28 ****SOLD OUT****
- **IESSIS1**: Immersion Event on Learning SQL Server Integration Services
  - April 24-28
- **IEBI**: Immersion Event on Business Intelligence
  - May 1-5
- **IEPTO2**: Immersion Event on Performance Tuning and Optimization – Part 2
  - May 1-5
- **IEPS**: Immersion Event on Powershell
  - May 8-10 ****NEW** class**
- **IEPDS**: Immersion Event on Practical Data Science
  - May 8-12
- **IEHADR**: Immersion Event on High Availability and Disaster Recovery
  - May 8-12

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

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