IBM i: Time-sensitive transactions and disk performance

From all the hardware resources in a computer system – CPU, GPU, memory, disk, PCIe IO bus, disk has always been the slowest component. Even if considering the latest flash disk or NVMe, they are still the slowest, although by a smaller factor. Back in the days of hard disk, in my experience, it was generally the most common cause of performance problems and would be the first place I would look in my investigation. Even the modern SAN flash disk can still be the performance bottleneck, particularly if not deployed correctly, as I have shared in a previous article.

When I evaluate an IBM i server performance, my personal rule-of-thumb is that a disk response time of 5 millisecond or less is considered good to great and if it stays consistent at this level over a range of disk I/O workloads (IOPS and MB/s), then even better. For SSD, I use 2.5 ms as a good value. Let's analyse a sample of good disk response time vs disk MB/s in Figure 2.

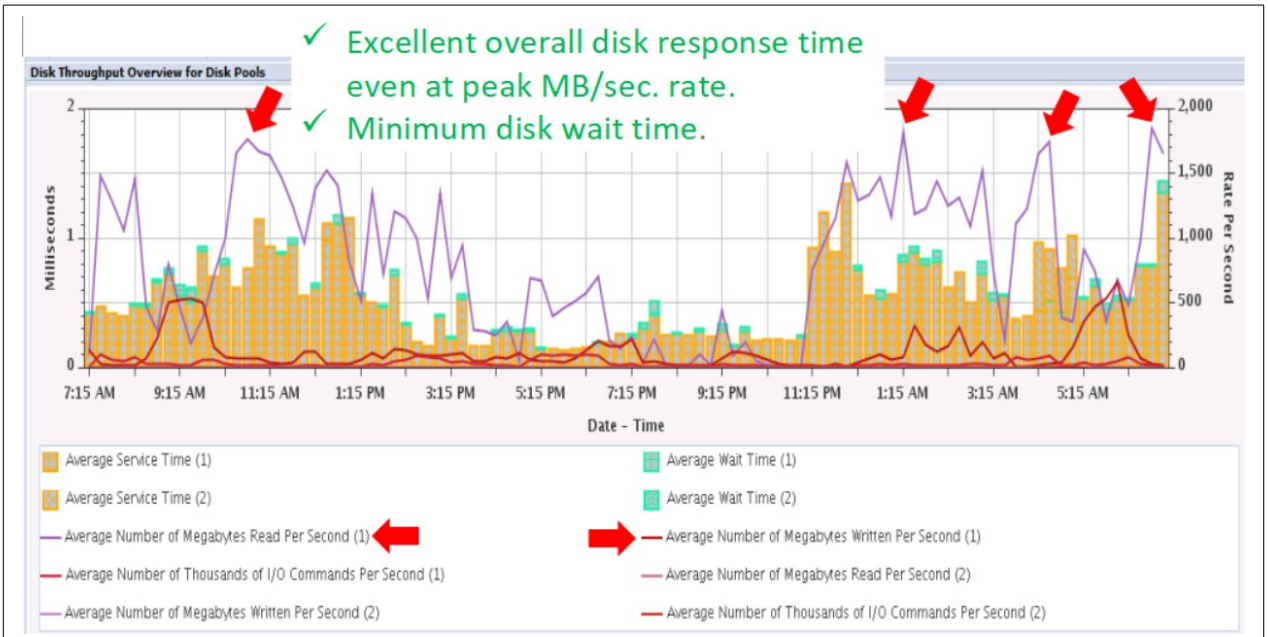

Figure 1: Disk workload showing good response times

These results are from a SAN flash disk and exhibits very good, but inconsistent response times (we discuss the inconsistency below). During this 24-hour window, the response times (orange bar) never goes beyond 1.5 ms, which is a very desirable performance. Another good result is the overall Disk Wait Time (green bar), which is also very low.

As for the disk I/O workload, my personal performance evaluation guideline is that disk data rate workload of 500 MB/s and beyond is considered high and beyond 1,000 MB/s is very high (purple and red lines). You can see that peak disk data rate reaches about 1,800 MB/s on several occasions (shown by the 4 red arrows) but the response time never increases above 1.5 ms indicating the system is not under stress and is handling the load.

In the middle part of the chart, you see low data rate with very low response time – that is a period of relatively low disk workload.

In my past performance assessment experience, it was typical that disk response times did degrade during periods of high I/O workload, which was also the case in this sample chart. However the good point for this sample is that the degraded response times still stayed in the good range. Typically you need to configure a large number of disk controller cards and physical disk units, incurring considerable cost, to maintain a consistent response time during periods of high I/O. My rule of thumb is that disk response times can fluctuate with fluctuating disk I/O workload (IOPS and MB/s), but shouldn't exceed 5 ms (or 2.5 ms for SSD).

For evaluation of hard disk response time vs IOPS chart, I consider 10,000 IOPS as high I/O workload and 20,000 IOPS very high. For SSD, I use 50,000 IOPS as high.

This guideline above served me well in most past cases but there were also a few exceptions. I would like to discuss one such case here.

The exception has to do specifically with time-sensitive workload which is a kind of client/server transaction in which a client system/device needs response from its server within a very short time period - a few seconds in most cases. A transaction from a cash ATM machine is one good example. Many such financial transactions for example ,money transfers, all kinds of payment, account inquiries, etc are time-sensitive as it would be unwise for the financial services provider to allow a client system/device to wait too long for the server response during these transactions. If somehow the server cannot respond in a suitable time, it's sensible to terminate the transaction, roll back any data changes made and let the user retry or not.

Here is a case study involving such kind of workload.

A banking customer upgraded their POWER6 server to POWER8 one and ran this new server for a few months. Internal hard disk units were used in the new server, not SSD. I was assigned to conduct an overall system performance health assessment of the new server for the customer, no problem solving here. On the first day when my IBM colleague and I met the CIO and chief IT operation and at the end of our discussion of my mission, I asked them whether they had any thing they would have liked me to do in addition to my stated mission and the chief responded that he had one problem for me to see if I could be of any help to the bank. Let's see how their nagging problem was described...

Many times in each month, bank's clients complained to the call centre about failed ATM transactions, some said they had to retry the transactions a few times to be successful. On any normal day, there were just a few or no complaints, however the number increased on payday (could reach up to 100 complaints), and this was a cause for serious concern to the bank. I asked if they noticed any discernible pattern to the problem and the chief responded with a yes, this problem happened only around 8:00 PM and lasted for about half an hour. It never happened on any other time slots of the day! After a brief pause that I needed to use my brain cells, my next question was whether the chief observed what kind of workload was running in IBM i server at that particular time slot and the answer was that 8:00 PM was the start of their nightly batch processing. Now, it was getting more and more interesting each second that went by.

I then asked the chief to tell me what operations were run between 8:00 and 8:30 PM and the answer was just one operation: pre-batch data backup, which took roughly half an hour for the data consisting of many GBs in total size. What a coincidence with the period of the problem! I made such an utterance and my colleague and the chief said they could not have agreed more. I guessed the chief already believed that this was the cause of this problem but had no idea how to resolve it.

Two more facts were that the backup was made to a number of IBM i save files, not to tape, and they ran some 15 concurrent backup jobs aiming to finish these backups as quickly as possible. This means the operation involved a lot of concurrent reads and writes to the disks – a significant disk I/O workload for sure.

I thanked the chief for all these crucial information and told him I might be able to help after gathering and analysing performance data. A few hours later, I looked at several PDI charts and obtained the following facts:

- CPU was initially high at about 90% for the first 5-10 minutes but much lower for the rest of the data backup period. But there was no corresponding CPU Queuing nor Machine Level Gate Serialisation wait time. <= No problem here.
- Memory faulting of the period was in a range of 300 faults per second. There was just a very small amount of corresponding Disk Page Fault wait time. <= No problem here.
- Disk IOPS was on the low side in the range of 2,000 IOPS. <= No problem here.
- The only dominant wait time of this period was Disk Other Time. My additional research revealed that this could have had to do with disk wait time from data backup operation. <= Not a clear clue for the problem here.

Note: I forgot to produce a Wait by Generic Job or a Task chart because this was the early days and the PDI tool was still new and I had to learn more about it.

I then looked at this disk response time vs MB/s workload chart and had a nasty surprise, see Figure 2:

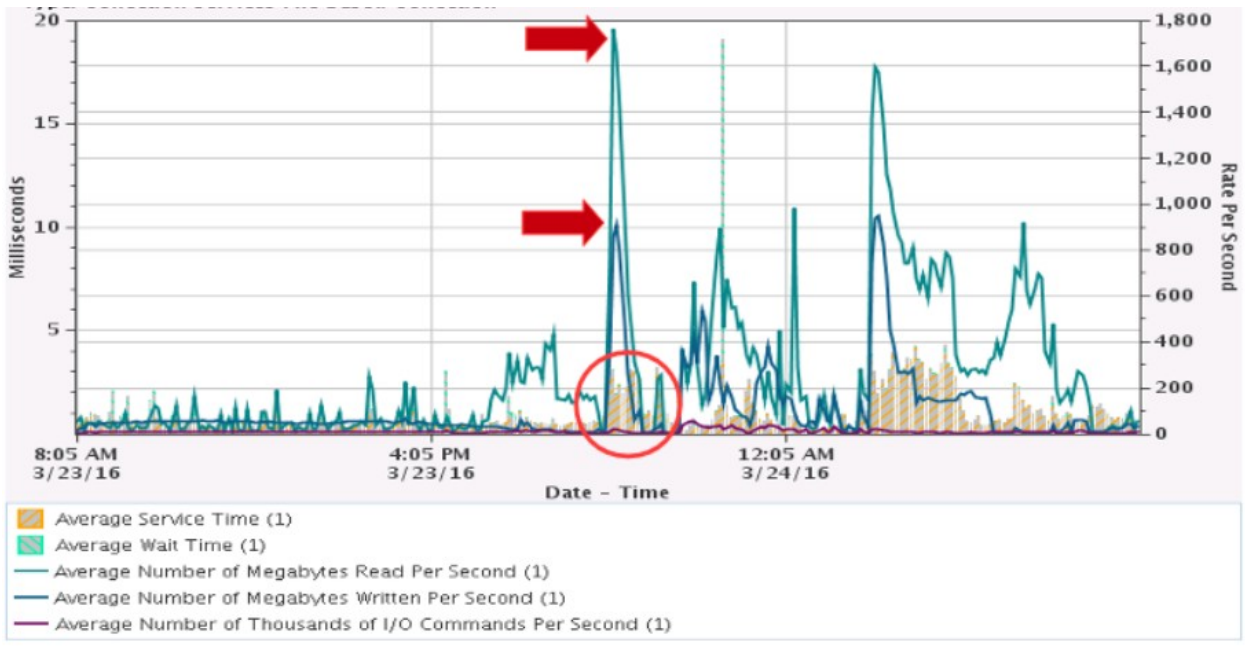

Figure 2: Disk workload and response times

Right at the middle of this 24-hour chart is 8:00 PM and you can see both high read and write data rate (two varying-shade blue lines), one peaks at 900 MB/s while the other at almost 1,800 (red arrows). But disk response time (orange bar in red circle) for this period is only about 3 ms or less. You should also be able to see that the response time then degrades considerably when compared to the preceding 12 hours (where it is mostly under 1 ms). This is quite understandable with the burst of data rate at 8:00

PM due to the data backup operation to save files. So, I saw it at the time that there was NO performance problem at all and therefore my nasty surprise was that there was still no indication of a viable explanation for the time-out issue.

At the time, I was not familiar with time-sensitive transactions but, luckily, my colleague explained to me about it and then we checked with the chief. He asked someone who confirmed back to us that it was the case that there was a time-out specified for ATM transactions. Two brains are always better than one and it helps to fully investigate the application!

What went on in my head at that time was a hypothesis that the degraded disk response time, although considered good in a general sense, could have been a possible contributing cause for the relevant server jobs on the IBM i to take a fraction more of a second beyond the time-out to respond to ATM machine, since the server jobs took more time to access data on disk. If we could reduce the degradation, we might be able to solve the problem. But how?

The answer was based on the fact that 15 concurrent jobs ran the backup tasks which was clear from the chart above that they caused high data rate workload stress to disk HW. So, I asked the chief to try reducing the concurrent jobs down to half (8 jobs) and observing the complaints, especially on payday. The chief later told us payday complaints reduced dramatically but not totally disappeared. So, I suggested reducing the concurrent backup jobs to 6 and this time it eliminated the complaints totally and everyone was happy. It also meant that the pre-batch backup time increased to about 40 minutes, but this was considered a small price to pay for resolving the problem.

Now, if you notice the rightmost half of the chart above (after midnight), you will see that disk response time degrades even further and for a longer period (post-batch backup, perhaps), but this does not cause the time-out problem most likely because of the simple fact that very few people, if any, use ATM machines during early hours of the morning just after midnight. Saved by luck!

After this case, I encountered a few more cases of transaction time-out problems from other financial services companies and they all had the same cause in degraded disk response time that was explicitly bad (thus was an easy case to analyse) or was not in a general sense like the case above (which took more analysis to conclude). But the cause of degradation was different in that disk hardware configuration was changed for more usable disk space without proper understanding about its performance consequence. But remember that PDI charts can always help us see what are needed to be seen.

This is a good case to remind you that performance data from PDI charts may not always show what is the prominent cause of a performance-related problem. In these cases, it is prudent to scan a basic set of PDI charts (on CPU utilisation, waits, disk response time vs I/O workload, memory faulting, and a few more) to identify and pay attention to what you think is the best clue from the data and build a sensible hypothesis from it. It is then a good idea to perform an experiment to test the idea to see if you are on the right track or need to refine your hypothesis.

The lesson learned here is also that in certain situations in which disk response time degrades from its normal range, time-sensitive transactions can be vulnerable to this and misbehave. Keep in mind that there is a random nature to this problem in that not all, but some to many, of such transactions will misbehave during that vulnerable period but a general pattern to the problem may be discernible and you pay attention to identify it. IBM i PDI charts supply a good source of clues for you to look at the time when disk response time degrades and check if they correspond to the times when time-out problems occur or not. If so, gather and analyse more PDI charts, formulate the most viable theory, and take actions to test it to resolve the degradation.

From now on, whenever you try to solve transaction time-out problems in an application environment with time-sensitive transactions, it would serve you well to apply the experience I share with you here and in all my preceding articles.

Satid Singkorapoom has 31-year experience as an IBM i technical specialist since it was called AS/400 + OS/400. His areas of expertise are general IBM i system administration and performance of server HW, OS, and Database. He also has an acquired taste in troubleshooting problems for his IBM i customers in ASEAN geography.