What I Learned This Month: z10 to zEC12 Migration

Scott Chapman

American Electric Power

This has been a very busy month. We did our first z10 to zEC12 migration and we're busy getting ready to migrate our disk subsystem from a DS8300 to a DS8870. This month's topic is the processor migration, which is a story of fewer, faster processors vs. more, slower processors.

I was more concerned about the z10 to zEC12 migration than I have been about a processor upgrade in quite a while. This was because we are doubling the number of CPUs while cutting the effective processor speed in half. To be precise, we're replacing our two z10 505 machines for two zEC12 410 machines. The z10s each have 5 general-purpose CPUs; the zEC12s have 10 CPUs. But the effective processor speed of the 5xx series z10 is just about exactly twice that of the zEC12 4xx series.[1]

Capacity planning for any processor configuration change like this should start by using IBM's free zPCR tool to estimate the relative capacity of the different processor and LPAR configurations. In this case, zPCR indicated that the effective per-engine speed of our new zEC12 CPUs would be 51% of our existing z10 engines. Overall capacity of the new zEC12s was expected to be effectively the same as our existing z10s.

Even though the total capacity was not significantly changing, the fact that the CPUs are running half as fast means that workloads will take twice as much CPU time as before. But during times of CPU contention, most workloads should see less waiting for CPU because there are twice as many CPUs to dispatch on.[2] Workloads that aren't experiencing much CPU delay would be negatively impacted by the longer CPU execution time. This includes work running at off-peak times and work running with a high dispatching priority.

However, the total elapsed time for a unit of work includes more than just the time spent executing on the CPU and the time spent waiting to get on a CPU: almost all work also has to do some amount of I/O, whether to a disk or the network. So for most workloads that aren't suffering any CPU delay, the total impact of reducing the CPU speed by half is not that the work will take twice as long.

There is another practical issue that shouldn't be overlooked. Because the CPU time measurements will approximately double, there is a significant risk that the CPU time limit specified for individual batch jobs (either explicitly coded in the JCL or defaulted by the system) will be insufficient. For example, if a job normally consumed 20 minutes of CPU time on the existing processors, it would be expected to take 40 CPU minutes on the slower processors. If the programmer coded a CPU time limit of 30 minutes on the batch job, then the first execution on the new processor would likely fail with an S322 (CPU time limit exceeded).

Despite the risks of moving to more, slower CPUs, the zEC12-410 configuration really did seem to be the optimal choice. Going up to a faster 5xx series zEC12 would be inconvenient because the capacity of a z10-505 falls right between a zEC12 503 and 504. Going to a pair of 503s would quite likely result in too little capacity being available and going to 504s would trigger additional software license payments for capacity that we didn't really need. Additionally, my "gut feeling" was that our workloads would do better overall with more processors. That feeling was based on my memory of how our workloads behaved over 10 years ago when we made almost exactly the opposite migration: from two machines with 10 CPUs each to two machines with five or 6 CPUs. I forget the exact details, but I do recall thinking that the new machines didn't seem as efficient as the old ones. Of course many things have changed since then.

---

[1] For those that are not familiar with mainframe capacity settings, this is purely an artificial construct. The actual zEC12 processors are faster than the z10 processors, but for software pricing reasons, customers can elect to downgrade the CPUs to effectively a slower capacity setting. In this case were taking a 5.5Ghz zEC12 processor and effectively cutting it all the way down to just under 1Ghz!

[2] Nonetheless, queuing theory suggests that having fewer faster servers is better. But today's z/OS environment is much more complicated than the simple textbook queuing models. That's not to say that such models don't have some applicability to z/OS, but one needs to understand their limitations.

The available modeling tools were not as optimistic as my initial feeling, which made me seriously question whether the 410 solution really was best. But from a software cost perspective, there could really be no other solution. And one should remember that models are simplifications of reality that only provide approximate answers. So I documented the risks the best I could for management, made sure that I configured the machines such that we could switch[3] to a faster 5xx series if need be, updated the default time limits for batch jobs, and we moved forward.

At this point, the first machine has only been in place for a week, so it's perhaps a bit early to draw too many conclusions. However, the initial results have been very positive. The zPCR prediction was that CPU times should increase 1.95x on the new processors. Most comparisons that I've done put the actual increase at around 1.7x. As expected, different workloads react differently to the configuration change and there are a few things that show a CPU increase of over 2x, but the vast majority of the work has done better than the 1.95x. So overall, the zEC12-410 is over-performing vs. what was predicted by zPCR. I'm not going to complain about that, although I would like to understand it better. My initial re-review of my zPCR model seems to indicate that I did give zPCR the proper input data. I need to do some more research to see if I missed something, or if maybe our workloads are just particularly well-suited to this configuration.

As expected, online response time has increased slightly because that workload wasn't significantly CPU delayed. The increase has been less than I expected, at least in part because the CPU execution time increase was less than expected. The less-than-expected increase in CPU time combined with the pre-migration work to adjust the assigned CPU time limits means that we haven't had any problems with S322 abends. The coupling facility response times are somewhat confusing, with most values improving as expected, but a few strangely degrading. That may because there is still a slower CF in the sysplex that many structures are duplexed to. It will be interesting to see if that clears up when we migrate the other machine.

In short, it seems that moving to more, slower processors is working well for us. I'd like to congratulate myself on choosing a good configuration, but the fact that it's performing significantly better than expected means that I clearly didn't understand and plan everything perfectly. Or maybe I did everything right and zPCR was simply overly pessimistic. Either way, getting your capacity planning wrong in the direction of having too much capacity is usually better than finding you ordered too little capacity!

As always, if you have questions or comments, you can reach me via email at sachapman@aep.com.

---

[3] Using On/Off Capacity on Demand. If you're thinking about doing this, note that the tricky part is that you can't use OOCoD to reduce the CPU count below what was enabled when the box shipped. So you have to order a configuration with just a few CPUs (such as a 403) so you can move to either many slower CPUs (410) or a few faster (503). If it had shipped as a 410, I wouldn't be able to use OOCoD to move to a 504 even though that's more capacity because the engine count would have decreased from 10 to 4. There are a few other concerns too, but that's the most significant one.