What I Learned This Month:
That which you ignore can be a lost opportunity

Scott Chapman
American Electric Power

There’s probably a more colloquial and colorful phrase that I could use as the title for this column, but I’ll leave that as an exercise for the reader. I presume most shops are like mine in that the performance guy doesn’t always get notified when some new subsystem, software, or workload shows up. And even when he or she does know, he or she often takes the easy way out which is to let it fall into the default service class until it’s proven itself. In this case “proven itself” for started tasks means that it either has a really good excuse for why it needs to be in something like SYSSTC instead of STCMD and/or it’s proven that it’s not a significant CPU consumer over some reasonable period of time. So the long and the short of it is that we have a number of started tasks that fall into STCMD.

But every now and then you find a subtle effect which you weren’t expecting. There are a fair number of things that are above STCMD in importance, so it’s not unusual for those address spaces to experience some CPU delay during peak times. This is usually just fine, but occasionally you find an exception. In this particular case we were looking at some important CICS transactions that had elapsed times of several seconds during a particular peak interval. During our investigation, we noticed CICS transactions that were spending some significant time on the S8 TCB, something we hadn’t noticed before. The S8 TCB includes SSL socket wait time. In poking around further I found some IBM documentation that states: “Without the CICS-ICSF Attachment Facility, the application that requests a long-running ICSF service is placed into an OS Wait.” That doesn’t sound good.

In looking at some more transactions outside of our problem period, but still during periods of high activity, I found more elongated S8 times. That led me to looking at the ICSF task, which I found, not surprisingly, in STCMD. In looking at the history for address space, it was pretty clear that this one had “proven” itself: it had consumed only a few hundred CPU seconds on LPARs that had been up for about 8 months. Clearly, it’s not given to consuming large amounts of CPU time. And if CICS transactions are calling for its services, clearly it’s important. Therefore the decision to move it to SYSSTC was pretty easy. And in spot-checking some transactions after doing that, the strangely high S8 times were gone.

I still don’t believe that was the root cause of our problems that night: ICSF had been configured like that “forever”, nothing had changed (that we know about) in the CICS
transactions, and the overall utilization pattern was normal. But I do think the change was a previously missed optimization opportunity, and it never hurts to optimize something if you have the opportunity to do so before it becomes a problem! In this particular case, watching STCMD for things that maybe shouldn’t be there might be as important as watching for things showing up in the NEWWORK service class.

So that’s what I think I learned this month. If you think I didn’t learn my lesson correctly, please email me at sachapman@aep.com and let me know!