

What I Learned This Month: Don't skip CMG sessions

Scott Chapman

American Electric Power

I was fortunate to be able to go to CMG'11, and let me tell you, if you couldn't make it you missed some great presentations. As usual, most time slots had more than one session that I wanted to attend, and since I don't have Hermione's Time-Turner¹, hard choices had to be made. However, there was one time slot on Monday where none of the options really called to me, but I decided to go to the Apdex Update presentation by Peter Sevcik instead of taking an hour to catch up on email. This turned out to be a great example of why I never skip a CMG timeslot; because sometimes things that I don't think are going to be very interesting or applicable to me turn out to be so.

Apdex is an easy-to-compute metric that normalizes the user experience of all applications to a value between 0 and 1, with higher numbers being better and 0.85 or higher being considered "good". The idea is that by scoring all applications on a consistent scale you can easily tell at a glance which applications are being problematic and compare their relative health. If you're going to try to help somebody today, work on the application that has an Apdex score of .79, not the one with an Apdex of 0.88.

Apdex is readily computed from application response time. The critical point is that you need to choose a response time, designated T, such that users achieving a response time of T or better are fully satisfied with the application performance. Based on the results of various human behavior studies, Apdex then says that response time F is 4 times T and is the point at which the users are frustrated with their performance. Between T and F is the toleration zone: users achieving response times greater than T and less than F are tolerant of the application's performance, but not satisfied. The Apdex score is then calculated as:

$$Apdex_T = \frac{\left(Satisfied\ Count + \frac{Tolerating\ Count}{2} \right)}{Total\ sample\ count}$$

That's Apdex in a nutshell, but if you want the full story, www.apdex.org will tell you everything you need to know about it.

I've been aware of Apdex for years, I believe since the 2004 or 2005 CMG conference. However, I've never really used it, even though I can see where it could be a valuable metric. The problem I've always bumped up against is how to define the value of T. Everything hinges on T and determining T seems fraught with problems. My assumptions about what people are satisfied with are

¹ If you're not a Harry Potter fan, the Time-Turner allowed Hermione to go back in time so she could take multiple overlapping classes. Eventually the stress of all the extra work forced her to relinquish the charm. Reportedly the entire known stock of Time-Turners has since been destroyed.

likely biased and hence wrong. If I ask the users, I'm guessing I will get a different bias, one that skews the result in the opposite direction. I may think for a given application that anything under 2 seconds is perfectly fine. If I ask the users, they may say they're only happy when it's less than 0.5 seconds. Likely the answer is somewhere in between.

However, while I was sitting in Peter Sevcik's session in December, another idea came up which I think makes a lot of sense: figure out F instead of T. I have a feeling that determining F should be a lot easier. Strangely, users don't call us when they're satisfied, but they sure do when they're frustrated! If we look at response times during those frustrated times, we can probably get a pretty good feel for where F is. And F may be more important than T anyway. The optimal performance goal is the one that lets you meet your business needs while not frustrating your users. If the users aren't fully happy with the application performance, but can get the work done in the time allotted, then it doesn't make a whole lot of business sense to spend time, money and effort to improve performance. Tolerable performance and work getting done is probably the sweet spot in terms of optimal spending on performance. However, having a frustrated user base, even if the work is getting done, is not a good thing as that frustration will likely cause further problems. Unhappy workers are not the most efficient workers.

A recent case in point was the investigation of an application where we were meeting our SLAs, but some of the users were frustrated with performance during time reporting days. Taking a look at their response times while they were frustrated led to the conclusion that my expectation for T was likely too high for that application. Instead of 2 seconds, a T of 1 to 1.5 was likely more appropriate given that frustrations seemed to set in when they were seeing consistent response times around the 4 to 6 second range.

The next step to confirm that a T of 1 to 1.5 is a reasonable value is to actually generate the Apdex scores for periods where the users reported being tolerant of the response time vs. periods when they reported frustration.

If I do that, will I have a valuable metric that might allow us to more readily detect user frustration? Maybe, but I'm still not entirely convinced. I guess I won't know until I try it and now that I have what appears to be a good value for T, I should at least try it. This is another reason why I need a Time-Turner.

So that's what I've learned this month; never skip a CMG session, because even ones you think might be relatively uninteresting to you may very well inspire you to look at something a different way. And if you're interested in calculating a user satisfaction score, Apdex is likely a good way of doing it, but if you're having trouble coming up with the toleration point (T), try thinking about the frustration point (F) instead.

As always, if you think I've got it all wrong, please email me at sachapman@aep.com and let me know!