

Standards: Lean, Mean, or Pointless?

By Catherine Michel

Who Cares?

Collective groan. In trying to keep pace with technology evolution and agile development, when one hears “standards” one automatically thinks: pointless bureaucracy. And frameworks? Useless job aids.

But it needn't be that way. It can't be that way. In our increasingly connected world (a la apps, M2M, over-the-top services, and so forth) standards and frameworks are ever more critical. They are easily the fastest way in which an independent capability becomes interoperable with the supporting infrastructure. Think Apple and its App Store, or utility companies and smart metering, grid and home.

Without a published standard in which developers of such functions could connect to the backbone of, and devices within, these networks, capabilities would be limited to the internal R&D of the networks or device providers. Or, costly investment would be wasted working through the kinks of technical incompatibilities.

Instead, open standards and frameworks enable an exponential community of new ideas to become a reality overnight—and turned into revenue the next day.

On the contrary, the challenge, then, is not the presence of standards and frameworks but the glacier-like pace in which they are typically established and evolve, and deciding who is actually responsible for them.

A common approach is for industry standards bodies to form to become the common reference point. But that is when the trouble can start, particularly when competing technologies have already taken hold.



Over time contributors lose sight of the greater goal and get lost in the minutiae (though I'm not saying all minutiae is unimportant). Time drags on, and the standards become irrelevant rather than enabling.

So what approach works? Does a company strike out on its own and hope to match the power of Apple? Do like-minded companies forego the standards bodies and establish a collective agreement directly? Do the standards bodies trim the fat

and stay agile like the rest of the industry?

The key is getting to and adopting the right standards quickly, then keeping them relevant with the pace of innovation.

The Standards Succubus

I once spent a year of my life helping a large North American service provider become Capability Maturity Model (CMM) level 3 and 4 compliant. I was young, and maybe even a bit delusional, so at the time I did not appreciate just how much of my youth was being squandered. With a few consulting colleagues in tow, I enthusiastically went about mulling over literally thousands of pages of system documentation and devising a massive volume of common processes, templates and technology standards for this particular organization's software development life cycle. Yay me.

Of course it seems absurd to me now that a

New Multimedia Research Center

Search OSS/BSS topics across various media types

Company Brochures, Analyst Reports, Whitepapers, Videos and more

[Click here](#)

Pipeline Research Center - Start Your Search Today!

Not for distribution or reproduction.

technology-based organization would have either the time or the millions in budget required for what was a huge undertaking. But since it was the late '90s, when IT outsourcing was becoming a billion-dollar business, standardizing your approach to software development, and in the process achieving CMM compliance, was de rigueur.

Believe me, I am not trying to dismantle with one pithy retrospective the reputation or suggested value of a model that has served the industry for decades.

Rather, I am left wondering: what was it all for? Surely there were faster, more cost-effective ways of identifying, agreeing on and deploying key standards to get the organization working towards a common goal.

Therein lay the problem, I think: somewhere along the line the standards became the goal. You see, the original goal was a more commercially motivated one: cut costs by removing unnecessary steps in the development chain and improve time to market by establishing a common set of technical standards usable by multiple resources. But with all that time and budget and people allocated, the standards effort took on a life of its own, and no one, including myself, had the foresight or pragmatism to rein it back in.

Either way, did our efforts pay off? I don't really know. After my colleagues and I achieved the CMM certification we never went back and assessed our original goals.

From Greek Tragedy to Erotic Thriller

So why on earth, as an executive committee member of the TM Forum, am I coming off as anti-standards?

I don't mean to. In fact what I said earlier is what I genuinely believe: standards are critical for the industry to interoperate. As a founder and CTO of a software company, I know this all too well.

So I'm not trying to avoid them. What I'm trying to avoid is the myth of Sisyphus in the creation and management of standards—that is, the futile and endless pursuit of standards in the absence of a tangible commercial reason for them.

When I talk to other organizations about standards or manage them inside my own company I always try to apply a few simple core principles to creating, managing and using standards:

1. Have a good reason. Make sure there is a good, commercial reason for creating or adopting a standard. Don't pursue standards for standards' sake.

2. Focus, focus, focus. Focus on the core inconsistencies (e.g., baseline definition of M2M

The world of standards needn't be expensive, fruitless and frustrating.

devices, key definitions for product-service-resource) or the interoperability issues (e.g., radio network protocols, key handover points between Product Management and Software Development) you are trying to solve. Don't try to get everything standardized before you act. Technology does not like to wait for documentation, nor should it.

3. Keep it simple, keep it brief. If you end up writing more in the standards than in lines of code, chances are you've lost sight of points 1 and 2 above. Process standards should be seen as light guidelines, not how-to-breathe manuals, while technology standards should be binary, matter-of-fact positions that are not subject to interpretation. Standards should not stifle or waylay innovation.

4. Move quickly and with common sense. Paralysis by analysis is still rife in the industry. Several weeks, and sometimes months, are spent defining or deciding which standards to roll out and how. This is another sign of missing points 1 and 2. As my father always said, "Stop your navel-gazing!" Now, as fond of my navel as I may be, I would much rather make a quick decision based on sound logic and move forward at the risk of having to revisit it later than to make no decision at all while endlessly weighing all my different options. Rework is often less costly than lack of progress, and the two should not be confused.

5. Borrow heavily or collaborate. Avoid the not-invented-here syndrome. Industry bodies have done a lot of the heavy lifting and industry consensus when it comes to basic standards, so use them. In the absence of an existing standard, where some cross-company consensus is needed, quickly find a like-minded third-party company and work together.

Before I was a CMM guru I was a Software Test Manager. I know, I must have done something terrible in a previous life, right? But whenever I think of standards I'm reminded of what a young test analyst said to me at three in the morning when I was relentlessly instructing him on how to fill out a proper SIR (system investigation request) report: "Dammit, Catherine, we are not saving lives here!"

As he went on to berate the inanity of the process, and as I started to question the meaning of life, I

could not deny that the guy had a point. What was it all for? I just wish I had remembered that point a couple years later once I'd moved on to CMM.

But I remember it now, and often.

The world of standards needn't be expensive, fruitless and frustrating. In fact if done pragmatically and—I really must stress this again—quickly, standards do become the enabling backbone of something truly amazing.

“Dammit, Catherine, we are not saving lives here!”