

Can Continuous Delivery Help Government IT? (Industry Perspective)

December 18, 2014

Before starting the next big project, government IT organizations should get up to speed on continuous delivery (CD), a software development methodology whose best practices have evolved and matured within the software world for decades. Don't be put off by the name. Continuous delivery and continuous deployment are not one and the same. Your organization might not need to deploy new software continuously, but you can still reap the many benefits from practicing CD in your development pipeline.

Something Old, Something New

The Agile Manifesto, published in 2001 to promote software development improvement, states as its first principle: "Our highest priority is to satisfy the customer through early and continuous delivery of valuable software."

Although agile development was often pitted against more formal process improvement methodologies that government organizations favored such as CMMI (capability maturity model integration) and ITIL (information technology infrastructure library), most organizations end up merging the best of both: CMMI for regulatory compliance with measures such as ISO 9001 and agile for speed and enterprise readiness.

Back to Basics

One of the absolute musts in achieving CD is following the well established discipline of configuration management (CM). CM has evolved to provide a detailed view of the state of a system at any point in the past, as well as controlling the constant flow of changes. The discipline has been around in some form since the dawn of the Industrial Age, and has been enshrined in various industry and government standards, such as ANSI/EIA-649, the National Consensus Standard for configuration management. Achieving CD drives organizations to apply CM beyond its origins of mere source code management to digital assets at every stage of the software delivery pipeline, from requirements to delivery to tracking deployment and updates to deployed systems.

Audit Readiness

With high-visibility failures such as the Social Security Administration's recent IT debacle, government IT projects face elevated levels of scrutiny. When things go wrong, the resource demands of an internal or external audit make delivering on schedule and within budget even harder.

CD not only helps with the routine challenges of keeping budgets and schedules with higher-quality final products, but it also prepares a project for external scrutiny. It provides IT management with the information it needs — updated on a continual basis — to understand the current state of the software and where it's going next. CD solutions involve a variety of dashboard-style information systems that help IT management focus effort and resources on blocking issues. Done right, this information supplies IT management with valuable insight that helps manage the delivery process efficiently.

Those same practices result in a high degree of transparency that makes it easy to answer the kinds of questions auditors love to ask, such as “Who changed what?”, “When was it changed?”, and “Did that fix make it into last night’s test?” Whether the person asking is a representative of an external audit firm or an IT manager keeping on top of things, the answer cannot be, “I’ll get back to you on that,” which wastes more time and effort. CD pipelines enable any authorized user to answer those questions at any time.

Code Review and Extended Maintenance

Government IT organizations are often challenged to support software systems for an extended period, sometimes spanning decades. Long durations lead to difficulty maintaining knowledge capital because staff members move on to other things.

CD drives best practices such as code review largely for the immediate benefit of getting higher quality code out the door faster. Modern code review tools capture new discussions about old code changes. Importantly, those discussions are captured in context, right alongside the code changes themselves. That record helps current staff understand the ancient systems they need to maintain.

CD and the Development Environment Architect

CD influences go broad and deep into IT systems involved in software development and delivery. You need to do more than simply mark off a checkbox indicating that you have a certain list of tools installed. Applying CD requires taking a big picture view when it comes to software development infrastructure, recognizing that the capabilities and limitations of an entire organization are determined, to a large degree, by the capabilities and limitations of its tools.

With that in mind, organizations do well to ensure that they have a development environment architect on staff. That role is charged with continuously making sure the development projects have the right tools for the job and are using them well. These architects are on the lookout for industry trends and emerging technologies, making sure their own organization doesn’t fall behind the times in terms of tools and processes. And more importantly, that it doesn’t end up being ridiculed as yet another boondoggle.

Continuous delivery will keep growing in popularity in the private sector, and government agencies should take notice. Although the practice will never take over the more traditional development methodologies completely, it will make a positive difference once its best parts are adopted.



C. Thomas Tyler is a senior consultant at Perforce Software.

<http://www.govtech.com/opinion/Continuous-Delivery-for-Government-IT.html>