

COMMERCIAL IOT DEVELOPMENT: AGILE OR WATERFALL? OR BOTH?

Getting agile and waterfall cultures to work well together takes an extra level of commitment to communication and appreciation for each team's responsibilities.

IOT INDUSTRY INSIGHT

MESH SYSTEMS™

OCTOBER 2016



This blog post was authored by TJ Butler, Chief Software Architect at Mesh Systems. It was originally posted in October 2016 on The Internet of Things Agenda Tech Target blog.



TRANSFORMING EVERY DAY DEVICES WITH THE INTERNET OF THINGS



SMART MERCHANDISE

Generate excitement and brand connections with your customers. Design products to adapt instantly to users' preferences at time and place of use. Lights, signage, interactive displays, proximity coupons and shelving.



SMART FUNCTIONS

Exponentially expand opportunities for new functionality, utilization and reliability by connecting devices, data, people and processes. Monitor and control key assets and manage health of the entire system.

COMMERCIAL IOT DEVELOPMENT: Agile or Waterfall? Or both?

It doesn't take long working in the world of IoT to feel the pulse of the clock. From idea inception to implementation, there is a sense of urgency to move fast. Each tick is a reminder that the competition is working diligently to provision a patent first, get to market first or be the first to scale.

Fast pace development is nothing new to software development teams big or small. Quick and nimble development — often called “agile” — is at the heart of the startup culture that has driven the IoT boom, and it's resulted in an expectation that these products can be delivered fast — especially if you work with a small team. On the other hand, the typical process that ensures enterprises deliver a controlled, predictable and consistent product — often called “waterfall” — can get in the way of fast innovation. As a result, many enterprises work with small teams or create internal innovation teams to spearhead

research and proof of concept efforts. Enterprises leverage smaller teams because they aren't tightly coupled to the extended business processes that support everyday business like regulations, quality assurance, billing and customer service. The smaller, focused team aims to fail fast, iterate, improve and repeat, quickly honing



in on delivering a minimum viable product. In the case of IoT especially, it is extremely helpful, I'd even say critical, to get out in the field and learn what you couldn't experience in a lab before getting too far down the design road. IoT, unlike software, includes a hardware element that

may not be nearly as flexible due to manufacturing physical product as part of the core solution. You may not have the luxury to make an iterative improvement on a sensor once it's been manufactured and is sitting in a warehouse.

Certainly, this is the nature of innovation in general, but IoT introduces a software component and culture which is often new to traditional product manufacturers. In many cases, IoT is digitally transforming products that have had no technical dependencies other than parts procurement or resource planning. The transformative combination of hardware and software creates new value beyond the widget itself, but it also creates dependencies which disrupt long-standing workflows in manufacturing and product lifecycles. It's often at odds with the enterprise processes supporting the product itself outside the innovation garage.



Embrace Agile Failure in Pilot and Respect Waterfall Quality for Commercial Rollout

The transition from pilot to production is where attention to detail becomes top priority. No matter how flexible or progressive an enterprise may be, some level of process and control will become necessary. Arguably as important as innovation, enterprises have an underlying core responsibility to brand reputation. Failure, which is a core tenant of the agile process, is counter to brand reputation.

That's not to say enterprises can't be innovative, it's simply a salient reality that everyone involved in commercializing a product must appreciate. Enterprises passionately want to avoid failure when the product hits customer's hands. This is why waterfall still plays an important role. It's most apparent when an IoT product moves beyond version 1.0.; this is when agile starts to take a back seat to waterfall

and the integration into all of the supporting business systems begins. All of those integrations collectively make up the end customer experience and the profitability of the business. Therefore, the pace slows, scrutiny increases and the infamous red tape is put up. It's at this point where the agile teams feel the jarring affect because they're still under pressure to commercialize

fast in an environment that expects hard delivery dates.

The fact is, it's hard enough to manage resource planning for manufacturing. Adding dependencies on a software layer or several software layers, especially for teams unfamiliar with software development styles like agile can be absolutely jarring. It's natural for software teams to update the product as soon as an improvement is available, but it would be unthinkable to pull a completed product sitting in a

warehouse to change the physical shape of a widget. Enterprises leverage rigid, gated processes because that's what ensures they get it right the first time and deliver a quality product, on time at scale. Orchestrating and planning across the entire spectrum is just plain hard when things are changing on the fly ... but changing on the fly is what makes software great. Getting agile and waterfall cultures to work well together takes an extra level of commitment to communication and appreciation for each team's responsibilities.

Ironically, one of the most popular agile software development methodologies called "Kanban" came from the large-scale manufacturing world. It's certainly a testament to the fact that common ground exists. The key is to acknowledge there will be a transition as the solution matures and work to compromise wherever possible, ultimately leveraging the strengths of both approaches. My recommendation is to stop thinking agile versus waterfall and embrace agile and waterfall.



Blending the Manufacturing Processes of *THINGS* with Software

Even enterprise software development teams familiar with agile may have a hard time with IoT projects because IoT isn't comprised of a single app. IoT solutions are truly distributed applications spanning several layers of hardware, software and transport borders. These layers have different lifecycles but have tight dependencies on each other. It can be difficult to make changes to one layer without having to consider the full stack. Here are, for example, are some high-level layers you can expect in an IoT solution, all relying on firmware and software to some extent:

- Device and sensors
- Local area transport
- Gateways/concentrators
- Wide area transport
- Data ingestion and back-end integration services
- Dashboard/reporting interfaces

Ideally, there are clear delineations between dev, QA and

production, but early on it may not be practical or cost effective to have completely segregated environments across all of these layers. Some low-power, low-cost devices may sacrifice over-the-air update functionality. This means they may be permanently tied to a specific build pointing to a specific environment limiting iterative improvements that are core to the agile process. This can have a significant impact on feature validation and testing timelines for early iterations. Transport layers such as cellular or LPWAN may have to be shared, early on, precluding testing that may risk stability. Some of the back-end services or integrations may be too complex or time consuming to set up in parallel across each party involved. While these aren't best practices, they are the practical reality that enables IoT projects to get off the ground. That's not to say the product is reliable or, more importantly, safe. The checks and balances expected in manufacturing automation, financial transactions and medical device require due process control, but I'd argue getting off the ground is just as important early on.



TJ BUTLER

TJ Butler is the Chief Software Architect at Mesh Systems where he is responsible for the design and development of IoT systems. Mesh Systems provides IoT software and services for product manufacturers.

TJ has helped start and grow several companies during his 20+ years of experience. That entrepreneurial spirit ensures his engineering approach is grounded with a solid business perspective.

TJ's career began with industrial automation building manufacturing intelligence systems, which has been a fantastic foundation for IoT. Just prior to Mesh Systems, TJ led the engineering team for Symantec's Risk Automation Suite after joining through an acquisition of Gideon Technologies where he was the Senior Architect. Butler is named as an inventor on two patents and holds a Computer Science degree from Purdue University.

Summary

From idea inception to implementation, there is a sense of urgency to move fast. Each tick of the clock is a reminder that the competition is working diligently to provision a patent first, get to market first or be the first to scale.

IoT is digitally transforming products that have had no technical dependencies other than parts procurement or resource planning. The transformative combination of hardware and software creates new value beyond the widget itself, but it also creates dependencies which disrupt long-standing workflows in manufacturing and product lifecycles.

Getting agile and waterfall cultures to work well together takes an extra level of commitment to communication and appreciation for each team's responsibilities. Ironically, one of the most popular agile software development methodologies called "Kanban" came from the large-scale manufacturing world. It's certainly a testament to the fact that common ground exists.

Stop thinking agile versus waterfall and embrace agile AND waterfall.

Article originally posted October 2016 on The Internet of Things Agenda Tech Target blog.

<http://internetofthingsagenda.techtarget.com/blog/IoT-Agenda/Commercial-IoT-development-Agile-or-waterfall-Or-both>

MESHSYSTEMS™

MESH SYSTEMS™

ABOUT MESH SYSTEMS



Mesh Systems rapidly delivers Smart IoT systems for companies looking to outsmart their competition.

We are an IoT software, services and solution provider that delivers turnkey systems that include hardware, software and networking frameworks for Smart products.

Using the latest IoT services and MESHVista® as the cornerstone, we help companies get new Smart products to market faster, with more scale and reliability, than ever imagined.

GET IN TOUCH



12400 N. Meridian, Suite 175
Carmel, IN 46032 USA



www.mesh-systems.com



sales@mesh-systems.com



1.855.301.2550



@Mesh_Systems