Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

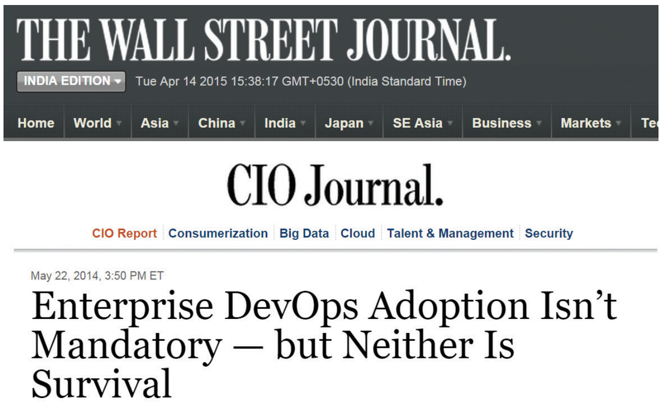
DevOps: What is the Right Speed?

People matter, results count.

Introduction

DevOps has caught fire in the IT world in the last few years. Not surprising as delivering faster has become a major imperative especially with the increasingly digital world and the convergence of internet, cloud, mobile, social and analytics. Speed has become the new currency for IT.

Most enterprises have some degree of DevOps implementation. Leading adopters, per the annual survey “State of DevOps”, are reaping the benefits. They are more agile doing up to 30x more frequent deployments than their peers. They are more reliable, up to 2x success change rate. And they are more successful, 2x more likely to exceed profitability, market share and productivity goals. Clearly, enterprises are seeing the value. Many are viewing DevOps as a necessity as one a Wall Street Journal article headline declares:



THE WALL STREET JOURNAL.

Tue Apr 14 2015 15:38:17 GMT+0530 (India Standard Time)

CIO Journal.

May 22, 2014, 3:50 PM ET

Enterprise DevOps Adoption Isn’t Mandatory — but Neither Is Survival

The enterprise question on DevOps is not if or why, but rather how best to implement DevOps in their respective organizations. Should they aspire to a Netflix, Amazon, Google, Etsy level of lighting speed, doing hundreds or even thousands of deployments a day?

What DevOps practices and capabilities should they implement? Where do they start….environment automation or testing? How does it all fit together - tools, organizationally, life cycle stage?

Our belief is there is no one right way to implement DevOps. Our view is that DevOps might even be adopted in varying ways throughout an enterprise.The correct speed in adopting DevOps depends on a variety of factors. The most relevant include:

1. the desired business goals and outcomes,
2. the type of workload,
3. the delivery methodology,
4. the desired release frequency, and
5. key speed and quality blockers. So in an enterprise, we can see digital applications achieving higher DevOps speed compared to stable, legacy apps.

The rest of this document expands on our hypotheses and provides guidance on determining the appropriate speed and capabilities to achieve the desired speed. But before we drill down, let’s get grounded on DevOps.

DevOps Level Set

DevOps is a philosophy and practice of collaboration and integration to continuously deliver fast and enable business goals. It is about a high performance teaming and execution among developers, QA, IT operations, project management, release and change management and of course the business. It has people (culture), process (methodology) and technology (automation) components. The key principles in these areas are:

Culture

- Execute as one team with shared goals and responsibilities
- Focus on delivering customer value and business goals
- Build high trust, collaborative environment
- Value communication, knowledge sharing

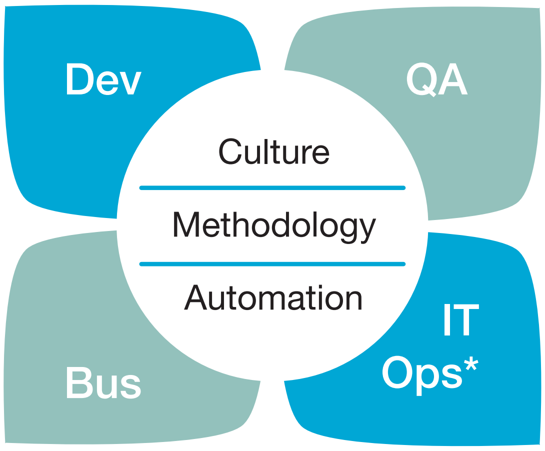
Methodology

- Drive towards smaller, more frequent releases
- Think components, micro-services
- Maximize flow, manage constraints
- Enable fast feedback and foster experimentation

Automation

- Standardize
- Make it modular for re-use
- Automate everything!
- Systematize feedback

Any successful implementation of DevOps embodies the principles above. The degree of embodiment is what drives the speed of DevOps. The next section further explores the various speeds possible.



Different Speeds of DevOps

The speed of DevOps can vary greatly from company to company, workload to workload depending the capabilities implemented. The diagram below describes the potential tiers of speed: **FAST, FASTER, and FASTEST**.

In **FAST**, the focus is on implementing a base level of DevOps. Strong virtual teaming, build and environment deployment automation, and foundational application and environment monitoring are the key capabilities needed to achieve this speed. Most legacy applications using waterfall methodology would find FAST with monthly or bi-monthly releases as appropriate.

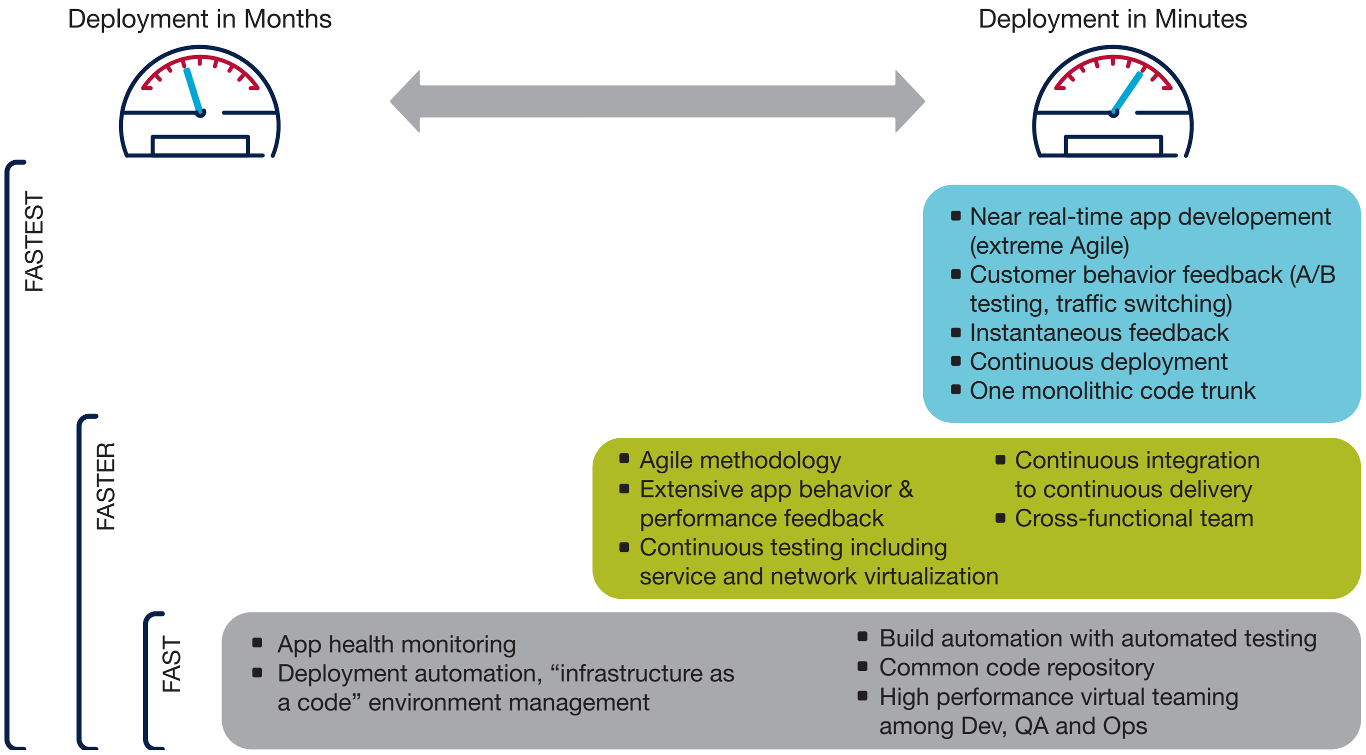
On the other hand, for digital and customer-facing assets, **FASTER** is a more appropriate speed. With these workloads, Agile methodology is used and is typically constrained by downstream testing and release management activities which operate on waterfall mode. To unclog this bottleneck and enable a FASTER speed, continuous "everything" needs to be adopted, from continuous integration, continuous testing, to continuous delivery. Cross-functional team structure replaces virtual teaming. And more comprehensive application behavior and performance feedback are available. At this speed, deployments can be done as fast a days or weeks.

Finally, if an application is the company's business and a key competitive weapon, **FASTEST** is the desired speed. This means operating like a Netflix, Amazon, or Google, being able to perform thousands of deployments a day. Continuous practice gets extended all the way to continuous deployment. The application, although utilizing a loosely coupled architecture, is one monolithic trunk? The highest degree of self-service is enabled for developers. This entails almost full automation in build, testing, deployments, and environment management.

Determining the Right Speed

Targeting the appropriate speed of DevOps needs to happen by application or workload area. To determine the right speed, some key dimensions need to be considered:

- **Desired business goals and outcomes:** Any DevOps endeavor should first and foremost consider the business problem and intent. DevOps is not just about automation. It is more about achieving a business goal. For example, achieving more frequent releases requires different capabilities when compared to increasing quality of releases. With more frequent releases, enabling continuous integration and delivery is essential. For improving quality, what's more important is having common artifact repository with strong version control and a very high coverage for testing automation. As a result, the end speed of more frequent releases may be weeks; while for quality it may be months.
- **Workload type:** The type of application can also determine appropriate speed. As described in the previous section, at least a FASTER speed is appropriate for digital workloads because of the pressure to continuously innovate and



- Near real-time app developement (extreme Agile)
- Customer behavior feedback (A/B testing, traffic switching)
- Instantaneous feedback
- Continuous deployment
- One monolithic code trunk

- Agile methodology
- Extensive app behavior & performance feedback
- Continuous testing including service and network virtualization
- Continuous integration to continuous delivery
- Cross-functional team

- App health monitoring
- Deployment automation, "infrastructure as a code" environment management
- Build automation with automated testing
- Common code repository
- High performance virtual teaming among Dev, QA and Ops

deliver quicker. In contrast, legacy applications or systems of records are fairly stable so FAST is sufficient.

- **Delivery methodology:** It is difficult to achieve FASTER or FASTEST when a workload is delivered using a waterfall approach. Having the right methodology like Agile is essential to achieve higher DevOps speed.
- **Delivery blockers:** The pain points and speed impediments drive what capabilities are needed. And the capabilities implemented, in turn, determine the speed of DevOps that can be achieved.
- **Desired frequency of releases:** The level of DevOps is also dictated by desired release pace. If an application is integrated with external partner systems, the release pace can only be as fast as to how long frequent partner systems can take. If the partner systems can only absorb bi-monthly updates, then FAST will suffice.

To illustrate how this may work, let's take the example of a CRM application area for a global commercial bank. They use a banking specific CRM package that is constantly updated by the package vendor. The package has been tailored to specific geographic needs. To be more market and customer responsive, the bank would like to increase the frequency of releases from quarterly to monthly. At the same time, quality is even more critical especially with the various code branches and package versions per geographic deployment. The key impediments for speed and quality are concentrated in the operations and infrastructure areas. Builds, deployments, and restart/refreshes take days. Limited visibility to the health and capacity of the application and environments result in building too much capacity or having too little resulting in delays. Provisioning new environments take weeks. In addition, code and vendor updates reside in multiple repositories with no consistency in tracking which build packages are in which environments. Based on situation described, the key DevOps capabilities needed to maintain quality and do monthly releases include: common code repository, continuous build, automated provisioning and release management, and baseline visibility to application and environment health. Embracing and implementing these capabilities lead to a FAST speed.

Enterprises will need to determine their own adoption speed for implementing DevOps depending on the workloads. Varying degrees of DevOps adoption depending on the workload should be the norm. Maturing in DevOps does not mean achieving the top speed; instead, it is about taking a thoughtful approach to implement the right speed given an workload's context and intent.

About Capgemini

With almost 145,000 people in over 40 countries, Capgemini is one of the world's foremost providers of consulting, technology and outsourcing services. The Group reported 2014 global revenues of EUR 10.573 billion. Together with its clients, Capgemini creates and delivers business and technology solutions that fit their needs and drive the results they want. A deeply multicultural organization, Capgemini has developed its own way of working, the Collaborative Business ExperienceTM, and draws on Rightshore®, its worldwide delivery model.

Learn more about us at

www.capgemini.com

For more details contact:

Rene Claudio
Cloud Platforms Leader,
Enterprise Cloud Services
+1 (425) 443-5062
rene.claudio@capgemini.com

Steve Horning
Strategy & Operations Lead,
Enterprise Cloud Services
+1 (415) 316-9254
steve.horning@capgemini.com

People matter, results count.