



Agility in the Cloud: How Dev/Test Teams Can Increase Velocity While Reducing Defects

By Seth Payne, Sr. Product Manager

Contents

Introduction	3
Impediments to Agility: Infrastructure Acquisition, Provisioning, & Deployment	5
Removing the Impediments: Leveraging the Cloud	7
Cloud Differences Impact Adoption Choices	12
Environments vs. Infrastructure	13
Conclusion	14
About Skytap	15

Introduction

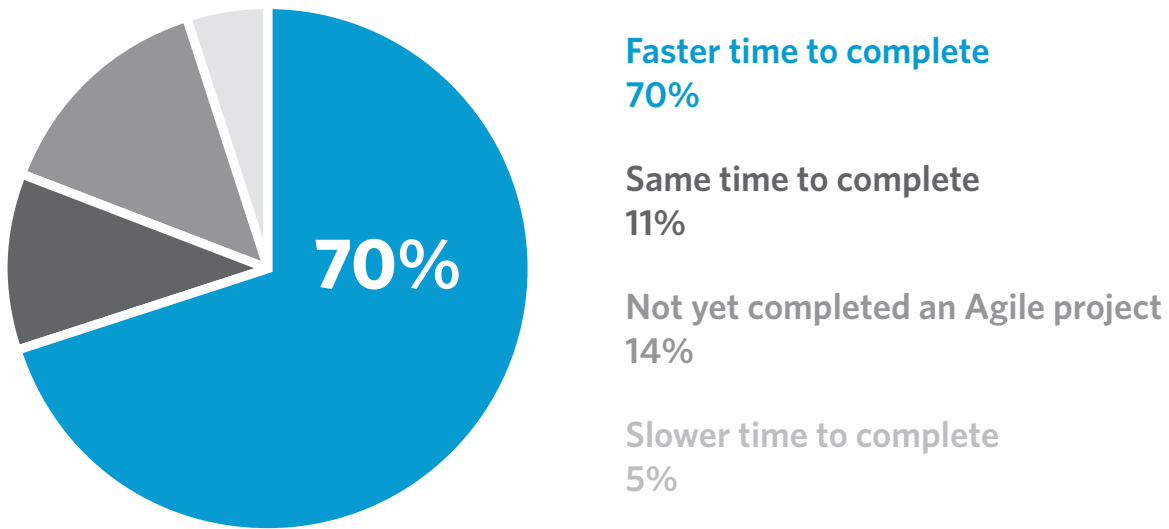
Over the past decade, Agile development methods and continuous integration have seen broad adoption across the enterprise, large ISVs, and even amongst system integrators. The reasons for this adoption are not difficult to understand. Adopting an Agile mindset can help development and test teams improve software quality, shorten release cycles, and enable business owners and stakeholders to quickly deliver higher value to their customers.

When looking at enterprise-class agile development (EAD) in particular, Gartner notes, “Through 2015, 40% of organizations will actively adopt EAD to gain business differentiation for projects and programs needing strong collaboration and cross-process cooperation.”¹ Why is Agile being so widely adopted? Seventy percent of those recently polled by VersionOne indicated that the adoption of Agile methods has led to a “faster time to complete [software development projects].”²

¹ Gartner, Hype Cycle for Application Development, 2013, Nathan Wilson, Gordon van Huizen and Brian Prentice, July 25, 2013.

² VersionOne, 7th Annual State of Agile Development Survey, 2013.

Faster Time to Completion



Yet, even with the adoption of Agile methods, impediments to the delivery of quality software remain. One of these impediments is the acquisition and provisioning of environments to support continuous integration and other build-and-test processes. The emergence of cloud computing—and in particular, the ease with which virtualized infrastructure may be managed and deployed—has provided development and test teams with a powerful tool to aid in their goal of delivering high-value software to business owners and other stakeholders quickly and consistently.

This paper explores how software development and test teams can leverage complex virtualized computing environments, delivered via the cloud, to improve existing workflows and decrease the time-to-value of software development projects.

Impediments to Agility: Infrastructure Acquisition, Provisioning, & Deployment

Hardware virtualization has radically altered how IT and software development professionals acquire, provision, and deploy the infrastructure necessary to support critical enterprise applications as well as software development efforts. Today, over fifty percent of all x86 server infrastructure are virtualized and the number of virtualized servers increases every day.

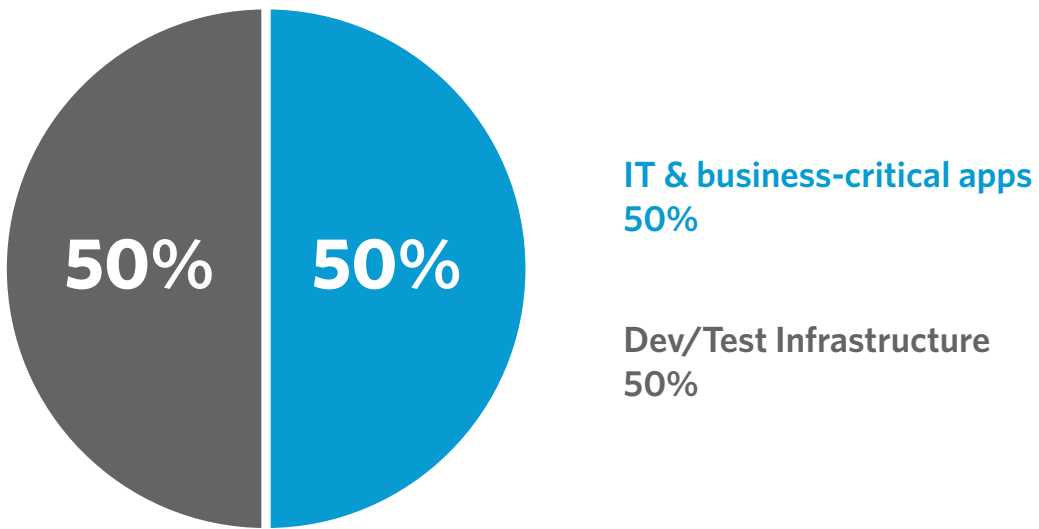
Balancing stability, flexibility, and productivity

The flexibility provided by server virtualization allows IT managers to provision environments for dev/test teams quickly and with minimal configuration. But the primary responsibility of IT administrators is to support and ensure critical applications remain stable and healthy. Conversely, development and test teams are not concerned about long-term server or application stability. Rather, they are focused on build-and-test processes that often require rapid changes to server and network infrastructure. IT managers may struggle to balance the need for stable long-running enterprise applications with development and test infrastructure that may be in constant flux. As a result, the acquisition of needed environment resources can impede development as teams wait for needed servers to be provisioned and deployed. This need for infrastructure has led to fifty percent of on-premise resources being devoted to dev/test. However, as dev/test managers are reticent to give up resources once acquired, up to ninety percent of this infrastructure remains idle.

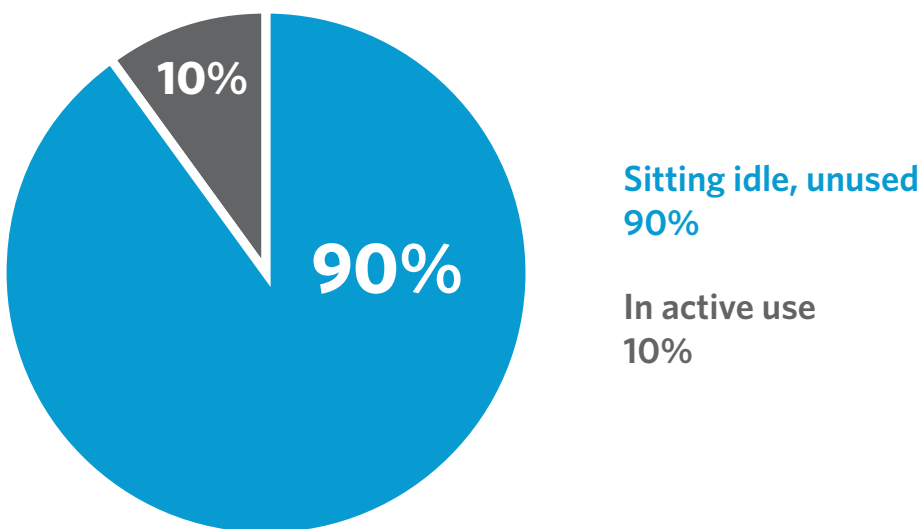
³ Hernandez, Pedro. "Survey: 51% of X86 Servers Now Virtualized." Quinstreet Enterprise, <http://www.serverwatch.com/server-trends/survey-51-of-x86-servers-now-virtualized.html>.

⁴ IBM Extends Development and Test to the Ibm Cloud." IBM, <http://www-03.ibm.com/press/us/en/pressrelease/29685.wss>.

IT Infrastructure in the Enterprise



Dev/Test Resource Utilization



Development and test teams rarely, if ever, request a single server or even several servers. Given the nature of modern multi-tiered applications that require complex heterogeneous compute, storage, and network environments, teams often require many machines—running in parallel—to perform build-and-test functions as part of their continuous integration workflows. Often, development and test teams are left waiting to acquire this needed infrastructure, which negatively impacts the team’s ability to maintain high levels of Agile velocity in any given sprint or throughout multiple sprints.

Removing the Impediments: Leveraging the Cloud

The success of any Agile development project is the fluid interaction among people, Agile processes, and resources. Many teams are successful at determining the appropriate people and process interactions, but they struggle to manage computing environments that support their Agile methods.

Software development and test teams can leverage cloud computing to remove impediments around resource acquisition and management, increase Agile velocity, and reduce time-to-value for business owners and stakeholders.

There are five primary characteristics of cloud computing:

1. On-demand self-service provisioning
2. Elasticity and scalability
3. Broad network access
4. Resource pooling
5. Metered or measured service

On-demand self-service provisioning, elasticity and scalability, as well as the tight integration of cloud-based resources with CI processes are likely the most important to development and test teams seeking to take advantage of cloud computing as a critical part of their Agile development efforts.

On-demand self-service provisioning

By giving development and test teams the ability to provision needed infrastructure on-demand, organizations solve two major problems. The first is that of wasted resources sitting idle and waiting to be used. If dev/test managers can gain access to infrastructure resources on-demand, they will have no need to hold those resources idle until needed. The second problem is that even when available, existing environment resources may not meet the specific needs of a given software development project. Allowing development and test teams to acquire the specific resources they need, when needed, empowers these teams to maximize efficiency and Agile velocity. Resources may be adjusted almost immediately to meet the needs of a specific build or test.

Additionally, on-demand self-service enables dev/test teams to run complex environments in parallel. This enables multiple team members to leverage and access isolated environments simultaneously—eliminating wait time as well as resource contention within the team. Providing teams with the environments they need, when they need them, can significantly improve velocity and overall team efficiency.

Elasticity, Scalability, and Flexibility

Measured services help maximize resource utilization by allowing teams to utilize the resources they need only for as long as they are required. This eliminates the impulse to overprovision existing resources in anticipation of future requirements, as well as freeing up existing idle resources for organizational use.

Application complexity and design may also drive cloud adoption by development and test teams. Modern applications are multi-tiered and may require a broad mixture of technologies. Managing such a heterogeneous environment is tedious and time-consuming. Any time spent by developers and testers configuring or deploying systems is time not spent on developing and testing code. Cloud technologies can help alleviate if not altogether eliminate this problem.

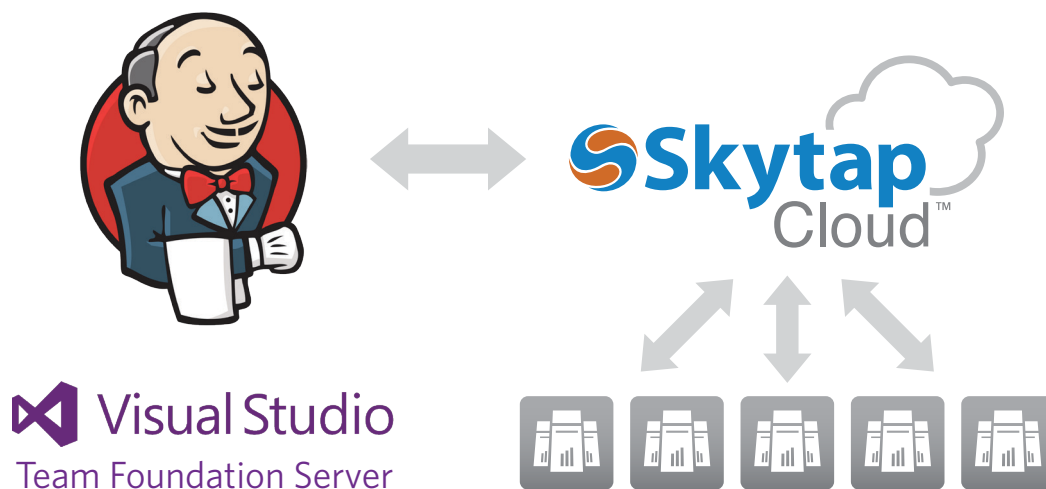
Complex, multi-tiered application environments can be built in the cloud and, through the use of snapshot or template functions, be captured as a golden copy that can be deployed over and over again quickly with a few button clicks or the use of simple scripts. The ability to replicate identical computing environments rapidly on-demand eliminates the time-consuming task of building, re-building, and tearing down complex application environments.

Similarly, snapshot and template technologies enable developers and testers to save the state of an application environment at a given point in time. Such saved states can be invaluable in tracking down and resolving bugs and defects. Snapshot and template technologies allow team members to quickly share saved environments and demonstrate bugs or issues that may come to light during the continuous integration process. An onshore tester, for example, can easily save the state of an application environment so that an offshore developer can observe and work within the environment in which the bug or issue occurred. This capability frees developers and testers from the often tedious and time-consuming task of configuring whole environments in order to reproduce problematic software behavior.

API and CLI Support in the Cloud

The vast majority of cloud providers offer Application Programming Interface (API) or Command-line Interface (CLI) support to control the provisioning, deployment, and management of cloud-based computing environments. API and CLI support allows infrastructure build up, tear down, and configuration to become part of test-driven development workflows. This functionality is especially powerful within teams who leverage continuous integration systems such as Jenkins or Microsoft's Team Foundation Server where infrastructure management can be made part of build, test, and continuous integration workflows.

For example, when using cloud technologies, a team may define a continuous integration workflow that kicks off the automatic provisioning of a multi-tiered, multi-Virtual Machine, heterogeneous application environment before deploying new code and performing integration or other tests. When the test is complete, the environment can be torn down as part of a single, uninterrupted workflow. Cloud computing then removes infrastructure constraints or delays from continuous integration workflows. This allows developers and testers to focus on their core responsibilities while maximizing and ensuring the efficient use of compute resources. Teams often see significant cost savings through the efficient use of on-demand compute resources provided by a public cloud.



Post-Sprint Activities Made Easier

In addition to improving test-driven development and continuous integration workflows, cloud technologies can also aid in end-of-sprint and post-sprint activities. A core principle in Agile is that usable, shippable software, however limited in functionality, should be the end-goal of any given sprint. In order to validate software functionality and ensure user stories have been completed to the satisfaction of business owners and stakeholders, iteration (or sprint) demonstrations, as well as user-acceptance testing are all essential.

Ideally, such demos would be performed in-person with all team members and stakeholders in the same room. However, in today's business world, with disparate team members spread across the globe—this is rarely feasible.

Virtual desktop, and similar technologies, enable scrum masters and product owners to easily share and demonstrate the results of any given sprint iteration. Business owners and stakeholders can quickly and easily observe completed user stories and provide essential feedback for the next sprint. Ultimately, teams can leverage these same technologies for user-acceptance testing. Teams and stakeholders can quickly and easily track the status of any development project throughout sprint iterations by leveraging the cloud.

Cloud Differences Impact Adoption Choices

At the most basic level, public cloud service providers must deliver three essential resources: storage, network, and compute services. At a higher level, providers offer similar on-demand, scalable compute infrastructure. However, there are differences in underlying infrastructure that may impact adoption decisions.

Hardware

First, virtualization requires the abstraction of physical hardware in order for platform and software to function. Hypervisors offer varying levels of support for different types of hardware. If development and test teams have specific hardware needs, it is essential that they select a cloud infrastructure that supports the virtual hardware required.

Functionality

Second, hypervisors support a wide range of snapshot, template, and restore functionality. For example, some hypervisors support in-memory snapshots of entire environments while others may support only saving states of individual virtual machines. Depending on application complexity, environment-level snapshots and templates may or may not be of particular importance.

Geographic

Third, cloud service provider data center locations where underlying infrastructure is housed may be of importance to a given team given geography and latency concerns.

Platform

Finally, cloud service provider infrastructures tend to optimize specific functionality. One provider may provide a simple collaboration platform while others offer granular control of disk input/output (I/O). Because of these differences, it is not uncommon for development and test teams to adopt diverse platforms to meet a range of technical and business requirements.

Environments vs. Infrastructure

It is important to also recognize that raw infrastructure (compute, networking and storage) is only part of the puzzle. Complete environments for developers and test engineers are an absolute necessity to the success of any application under development or test. In many software development projects, development teams often share resources or wait for environments to become available for their task. These bottlenecks slow the release of applications to production and ultimately cause increased delays, poor test coverage and more defects released into production than what is ultimately desired. Many enterprises today are looking for IaaS or on-premise resources to provide the adequate and underlying infrastructure necessary for application development. However, adding additional infrastructure is not the entire solution. There are three main contributing issues and they are as follows:

1. Developers are not in the business of wiring stacks together, configuring hardware and network connections and maintaining those environments. Development teams need turnkey environments that meet their specifications and are ready on-demand and easily disposable when not in use.
2. Self-service access to raw infrastructure does not deliver what developers need. Developers want to write code and release applications to market. Their desire to secure public cloud resources in particular is out of the necessity to complete a task. This is often called Shadow IT and it is only a by-product of developers not getting the resources they need, due to corporate IT's inability to facilitate those resources in a timely manner.
3. Regardless of public or private cloud, developers need access to full stacks that are ready to run. In the enterprise, application stacks are often very complex. Development teams need the ability to get access to those complete stacks quickly, to replicate those stacks based on a defined golden template, and to share and collaborate with those environments quickly.

Conclusion

This paper highlights the advantages of leveraging cloud computing to enhance and extend Agile development and test methods. Whether a team is just beginning an Agile software project or has been steeped in Agile methods for years, the benefits of cloud computing are clear.

No longer are development and test teams required to procure, provision, and manage infrastructure. Being free from this significant burden allows these teams to focus on their primary responsibilities and functions, thus increasing team efficiency and reducing time-to-value for business owners and stakeholders.

In addition to removing the infrastructure management burden, cloud computing enables development and test teams to easily collaborate, perform necessary tests, and perform both iteration demonstrations and user-acceptance testing by quickly sharing access to development and test environments. Add to this full cloud integration with popular continuous integration tools and it is easy to see how the combination of cloud computing and Agile development methodologies provides development and test teams with the tools necessary to create better software built to deliver significant value to organizations and their customers.

About Skytap

Skytap provides SaaS-based dev/test environments to the enterprise. Our solution removes the inefficiencies and constraints that companies have within their software development lifecycle. As a result, our customers release better software faster.





Often, development teams are unable to test frequently enough due to a scarcity of test environments and long wait times accessing them. These delays have a material impact on the ability to release quality software on time. Additionally, enterprise development teams are trending towards mixed labor models, and collaboration challenges further delay projects.

Skytap also provides IT with full visibility and control in order to enable dev/test teams to be effective while avoiding infrastructure sprawl and runaway costs.


Skytap Cloud is an enterprise service purpose-built specifically for development and testing of complex applications. Customers can import existing virtualized applications or build new applications in the cloud. Skytap Cloud can be easily accessed through any modern web browser, REST-based API, Command Line Interface (CLI), or ALM tool (Jenkins, Visual Studio TFS, etc.).

Skytap Cloud customers have a choice of infrastructure. Customers can run complex computing environments on Skytap's native ESX-based infrastructure stack on the Amazon Web Service, Amazon Elastic Compute Cloud (Amazon EC2) infrastructure.

Skytap™ | STACK


 Self Service Web Console	 REST-based API	 Command Line Interface	 ALM Tools
--	--	---	---

DEV/TEST COLLABORATION




Save & Reproduce Defects *Share Environments*

IT ADMIN, VISIBILITY & CONTROL





Access Controls *Configuration Management* *Utilization Rules*

SKYTAP CLOUD - LAB MANAGEMENT SaaS



Application Environments *Self-Service Access* *Network Isolation* *Build, Deploy, Test*

 GLOBAL VMWARE-BASED INFRASTRUCTURE	 GLOBAL AWS-BASED INFRASTRUCTURE
Compute Network Storage	Compute Network Storage

Skytap, Inc.

719 2nd Ave, Suite 300
Seattle, WA 98104 USA

Toll Free: +1-888-SKY-TAP8 (1-888-759-8278)

Web: www.skytap.com