cloudshare

Technical Whitepaper:

Accelerating Business Application Development

Using cloud dev/test environments to deliver innovation at the speed of DevOps

Includes:

- Why your current application infrastructure is holding you back
- How to break through the build process logjam
- Using 'Portable IT Environments' to keep the team moving fast
- Overcoming barriers to change

Copyright © 2013 CloudShare. All Rights Reserved.

Let's face it. Line of business (LOB) applications have much more complex requirements than the web-born applications of today. They require more infrastructure and setup to test and build on top of. Too often, these requirements slow down the build process, yet organizations with heavier applications still wish to deliver functionality to their users faster, and with greater quality.

Organizations that build applications to integrate large enterprise systems with custom built back ends are dealing with a range of complexities between system configurations, custom integrations, and siloed user requirements -- not to mention the developers' own code base. While the code portion of these applications may be developed in an agile way, it's the system configurations and integrations that bog down the process and prevent development teams from moving fast and innovating.

> ***..."through 2015, only 10 percent of I&O (Infrastructure & Operations) organizations will be able to deliver the speed of change required by the business processes they enable.*** *-- Gartner press release March 28, 2013. http://www.gartner.com/newsroom/id/2394815*

'DevOps' is a world where ideas move from code to production quickly and easily, and where development works with operations to deliver solutions. Is there a way to make business application development as agile as DevOps?

Needs, Plans, Projects – But No Progress?

When an organization wants to try out a killer feature in their existing enterprise applications -- ERP, CRM, ECM, and others -- they face a wall of tasks and change requests.

A business need leads to a chain reaction of process-heavy events that translate to a full-on, managed project. It usually starts with an idea to improve existing systems with a new feature or update. Next that idea gets aligned with a business objective and goal. Then a business analyst or team member gets assigned to research the impact of the change. They quantify cost, timelines,

Copyright © 2013 CloudShare. All Rights Reserved.

and expected result of executing on the idea.

When it comes time to test out the idea, the fear factor kicks in:

- How can we test the idea without impacting current configurations, code and integrations?
- What if something goes wrong?
- How can we be sure that we tested it fully in all (or even a few important) scenarios?
- What if the test breaks something else?
- How much time will it take to replicate our current environment to test?
- Will this project take too much time, and delay other projects?
- How do we demo the solution when it's nearly complete?
- How can we control changes after the demo?

These fears can often halt progress entirely. Even when an idea gets executive buy-in and a project starts, it can take a very long time, maybe even years, for the final solution. It will involve many stakeholders, and is subject to bombardment from competing objectives.

This process can kill innovation and prevent enterprises from adapting their applications as needs change. Over time, this limits the organization's ability to stay current and excel in their operations.

Infrastructure Is at the Root of the Problem

The IT environment is at the root of the conflict between enterprise application development and an organization's ability to move fast and innovate.

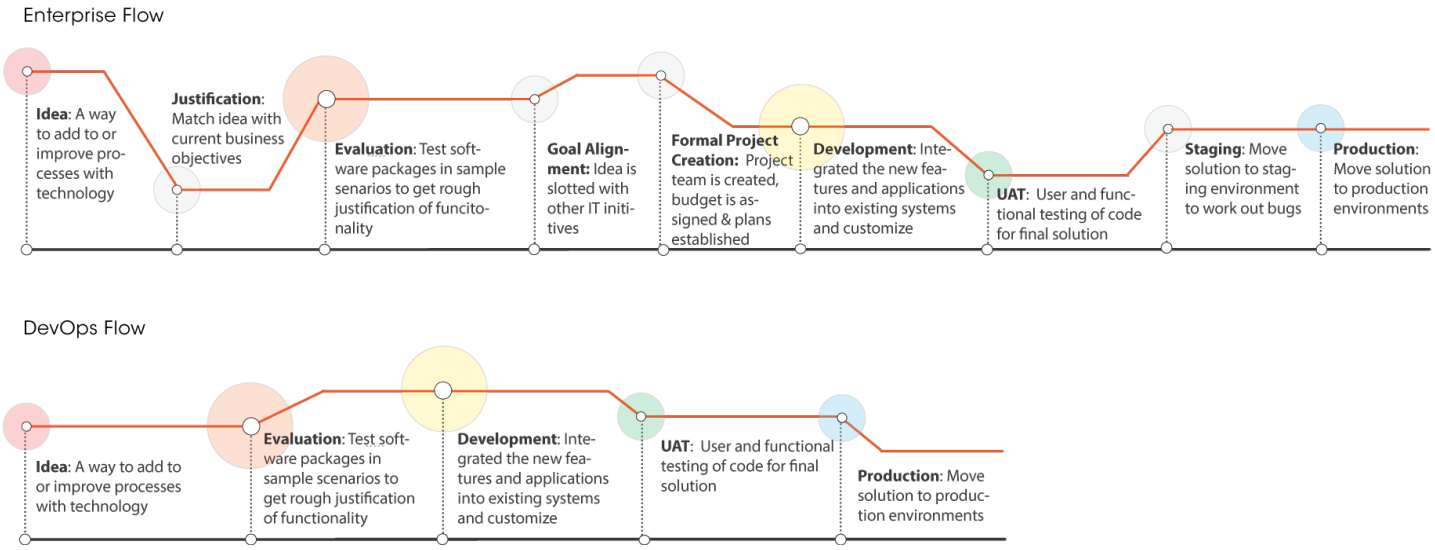
Development teams want to be agile and move closer to the world of DevOps. These teams seek to make greater impact sooner, with smaller changes and enhancements to the product over shorter sprints.

The limiting factor is not the developers' environments or their code. It's the complex enterprise applications themselves. These heavy server-side applications are built with countless special configurations. There can be custom code, custom connectors, custom services, specific and optimized server configurations or multi-server farms for load balancing. Even a team with

Copyright © 2013 CloudShare. All Rights Reserved.

the most agile coding process will suffer when they try to change complex configurations. These environments are big, complex, and connected – they are not inherently agile.

Decisions in DevOps happen fast. Compare the flows:



What It Could Be

Believe it or not, it is possible to develop business applications in a similar fashion to the world of DevOps, where ideas become features in as little as a week. New concepts are tested without considering for risks, empowering developers and analysts to directly impact the future of product.

DevOps is not a technology. It is a culture growing in tandem with a powerful set of tools. Above all, it values experimentation and embraces tools that allow for (safe) fast failure.

This culture dominates SaaS application shops, but the world has also changed inside large enterprises and SMBs. For those who haven't, it's not for lack of interest. Many development teams are frustrated by how long it takes to release their code and test new things quickly.

Copyright © 2013 CloudShare. All Rights Reserved.

Organizations are finding that the culture has changed, but the applications and legacy systems have not.

> ***"The market has gone through rapid consolidation, and much of the focus has shifted to service virtualization and enabling early testing, rather than how to test the services themselves. Driven by the emergence of DevOps practices, we are now seeing service virtualization become more closely tied to other elements of test lab management by the leading vendors. As the core capabilities become more commoditized, we expect to see solution-oriented offerings emerge."*** *-- Gartner's IT Market Clock for Application Development, September 2013*

Bottom line: Infrastructure + System Configuration is preventing many organizations from being opportunistic at all.

Introducing the Portable IT Environment

The simplest solution for development and IT managers is to capture this complexity and work with all the elements as one unit that can be saved, stored, and shared. By capturing the memory state, network configuration, virtual machines, and their installed software as a solitary object, IT and development give themselves the freedom to lift, shift and manipulate these objects.

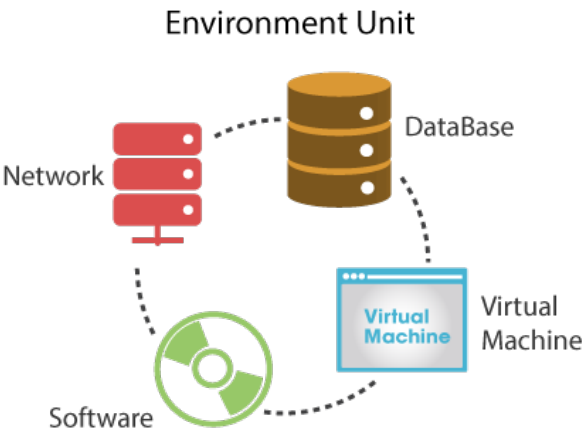
When you bring all these elements into a single unit, you can start talking about ways to version, replicate, and share – with all the complexity intact.

This does however imply some technological abilities of the organization. First, is the ability to virtualize infrastructure -- existing and new. Second is the snapshotting technology that captures the entire environment, not just its disk. The snapshot must include current memory state, network state, and all installed

Copyright © 2013 CloudShare. All Rights Reserved.

software and configurations. The end result of this is the ability to save an exact copy of the environment, run it, and work with the entire project, from front end to back, exactly as it was left.

When an organization can secure the environment and lift it to the cloud, then the organization is making progress. Adding the ability to automate complex orchestration moves the group even further. They can copy even the smallest changes and share them with stakeholders. They have taken the most inhibiting factor to innovation, and made it trivial and simple.



Environments consist of all machines, their software, their configurations, the networking between them in their current state.

Applying a DevOps Approach to Legacy App Development

The solution is to lower the entry barrier to enterprise application development and testing. First, there must be technology for capturing environment complexity. It must then be built for sharing so that it ships easily across development and operations teams. The lower the bar to testing, the more experiments are possible. As tests are completed faster, more questions are asked, and a virtuous cycle transforms the enterprise from inside out.

In practice, this allows the business analyst to go from idea to a fully configured test in minutes. Quality assurance engineers can then run more variations of tests more frequently with no compounded effort. And Developers can code creatively, wildly even, knowing that failed tests will be safe in their test environment. This brings a wider range of tests -- slight variations to major shifts in the way a company operates –and more testing to every feature.

The key to this is capturing the bulky interconnected complexity of an enterprise application -- the above-mentioned IT environments – in a simple, portable unit. And moving the units to the cloud. This unit can then be edited, shared, saved and versioned just like a document.

Copyright © 2013 CloudShare. All Rights Reserved.

There is then no limit to the number of versions or variations of this IT configuration. There is no need to involve large groups to run a test, so there is no need for formal projects. Instead, the execution begins as soon as the idea can be coded.

DevOps Agility Requirements: Beyond the Cloud

The generic benefits of the cloud are well known: rapid scaling, a global collaboration and sharing, and access from any device, anywhere. Enterprise teams are large. They work all over the world, and their systems are resource intensive and complex. A development and testing cloud is a must; on premise systems just cannot connect teams with the resources they need.

But developing in the cloud requires a platform designed specifically with this use case in mind. It should provide a simple orchestration system for building and using environments, APIs to program custom flows, as well as a cloud for scaling and distributing the resources.

The infrastructure in this cloud must also be programmable. No two enterprises work the same way, and workflows often change from project to project, so the platform must be flexible enough for midstream changes. The ideal platform has a granular REST API, a command-line interface and code samples for quick fixes.

All of this automation should rest on stable orchestration technology that automates vLan creation and environment walling. Teams should not have to provision virtual machines, write install scripts, or configure networks and software. And these environments, whether single or multiple machines, need to be self-contained and completely walled off from any other environments on the platform.

This type of service allows organizations to provide resources immediately, regardless of the bulk or complexity of demands. At any point, they can create 1, 10, or 100 copies of the environment, each fully isolated from the others. These versions can contain identical configurations with the software and processes in the exact state specified by the snapshot. For example, if the environment contained the ERP system midway through a workflow and application process, both the process and workflow will pick up exactly where they left off.
With all of this activity in a public cloud, administrators must have control and

Copyright © 2013 CloudShare. All Rights Reserved.

robust usage reporting. Management must be able to create various roles and access levels for users so that cost and cloud resources are under control. For the same reason, management also requires the ability to analyze usage and associate spending with the appropriate department, otherwise called "charge back".

With this approach in place the infrastructure is as agile as the development and testing. Analysts, developers and QA are empowered and IT has oversight. The organization stays nimble and current, building applications with a DevOps philosophy and approach.

Overcoming Barriers

Technology is enabling this shift in many organizations, but there are some common barriers to change. When moving infrastructure to the cloud and shifting to agile development these obstacles include:

1. Platform & Infrastructure Ownership
2. Security
3. Availability

Organizational structure can create some confusion over project and platform ownership. While DevOps culture grew out of a world of all-for-one teams, traditional enterprises often work in matrixed, siloed organizations. Some include development as a sub-unit of IT. Some give development teams their own infrastructure. And some teams are totally separate, but have to communicate to find infrastructure. Moving development infrastructure to the Cloud can help coordinate IT and Development. Developers can provision environments without getting into the IT queue. There is no impact on existing IT workloads, and IT can comfortably hand off resources when they set the best-practice templates, have full reports, and assign roles themselves.

Cloud security will always be a concern. While some believe that moving content to a public cloud can bring more risk, using a dev/test cloud can in fact add controls. First, any security applied on premise can be replicated in the cloud. IT can apply any protective firewall, VPN, or password authentication that the team needs. Then there are the cloud's additional controls, allowing IT to set and manage templates, network configurations and settings. Also, because it is a development and test cloud, sensitive data is usually not placed in these

Copyright © 2013 CloudShare. All Rights Reserved.

environments. This prevents mistakes that could put the whole organization at risk.

Availability is one of the great benefits of the cloud. As web services and public clouds have become more popular, downtime incidents have increased in visibility and impact. Consequently, some organizations want control over their services, uptime and responses. The truth is that the vast majority of individuals working in any enterprise need internet connectivity for basic services -- email, storage, and just about anything else connected to the web.

Availability can become difficult when corporate governance limits types of browsers and plugins users can install. For this reason teams, need to choose a dev/test cloud that is available over the public web, port 80. It should also be accessible through default corporate options such as RDP and VNC. The service also needs to provide accelerated connections to environments so that bandwidth is not restricted. And finally, they should offer the ability to connect cloud environments via a point-to-point VPN tunnel to local networks. This keeps compliance and governance intact without changing anything, plus offers the added benefit of cloud resources on the local intranet and vice versa.

Organizations that choose the right provider and plan effectively can only gain from moving dev/test to the cloud. The right platforms add control, reduce costs, and make resources available across the enterprise.

Tangible Benefits

Cloud's generic benefits are well known – capex reduction and zero infrastructure maintenance. Dev/test clouds take it much further than that. By capturing the complexity of modern enterprise, the dev/test cloud eliminates configuration challenges at every stage of a project. By allowing for replication and sharing across teams, it breaks down organizational and geographic barriers to productivity. In short, if your organizations stands any chance of being more current with technology, a dev/test cloud must be part of the plan.

From idea to production, and all points in between, a dev/test cloud saves you time, money, and risk of stagnation:

- Faster LOB application development, updates, patches
- Lower project-related costs

Copyright © 2013 CloudShare. All Rights Reserved.

- More testing, with no added effort
- Protection for production and private clouds from volatile dev/test work and contamination
- IT focus on optimizing production infrastructure

Organizations are already seeing the benefits of cloud-based development and testing. They have completely offloaded large infrastructure concerns and increased the number of releases to their users to be daily or weekly, instead of monthly or yearly.

The only choice to make now is if the critical business applications your organization runs are worth keeping up-to-date, or advancing. And if it's time to let your developers deliver more capabilities, more quickly to your internal users.

There is no need to left behind any longer. Adopt cloud-based development and testing approaches for your organization and your line of business application development, and you will close the gap between idea to implementation.

To learn more about specially designed dev/test clouds, contact CloudShare at info@cloudshare.com

About CloudShare

CloudShare is the leading provider of virtual environments in the cloud. CloudShare's self-service SaaS platform and suite of solutions enable business users to build, freeze and share complex computing environments for development, testing, sales & marketing, and training. CloudShare's on-demand IT environments empower users to kick-start new projects, explore new technologies, develop complex applications and collaborate across teams without battling for limited IT infrastructure resources. Over 175,000 users worldwide and more than half of the Fortune 100 use CloudShare. Headquartered in San Mateo, Calif., CloudShare is backed by Sequoia Capital, Charles River Ventures and other top investors. For more information, please visit www.cloudshare.com.

Copyright © 2013 CloudShare. All Rights Reserved.