



## Jumpstarting DevOps with Continuous Testing

By tightly connecting testing with development and operations, and automating the design, development, quality assurance and deployment of new applications and systems, IT organizations can more effectively collaborate and deliver on the dual mandate of increasing operational agility and speeding software's time to market.

### Executive Summary

IT organizations are adopting DevOps practices to enable more frequent software releases to production, both to accelerate time to market and improve the overall customer experience. But adopting DevOps practices requires a cultural shift in terms of the people, processes and technologies involved in development, testing and operations. Many organizations encounter significant challenges when embracing DevOps, particularly when implementing it in legacy areas. Such obstacles have proved far more difficult to overcome compared with those encountered by digital-native companies.

A popular belief is that DevOps requires throwing away everything the organization has built and starting from scratch. However, that is untrue;

the critical component of DevOps in large organizations is the need to improve communication and collaboration between IT team disciplines. Enabling such behavior can be cumbersome, especially when regulatory and audit compliance requires existing organizational structures to remain as-is.

In traditional IT organizations, developers, testers and operations engineers typically play different roles, with varying responsibilities, job descriptions and cultures, and all working in separate functional areas as distinct entities. The question is whether QA can unify these various groups under the DevOps flag and spur adoption. This white paper will attempt to answer that question and elaborate on the approach.

## The Role of Testing in DevOps

The word DevOps<sup>1</sup> was coined by combining the development and operations functions. DevOps is not a technology, a process or a standard; rather, it is an IT culture or movement that emphasizes ways in which development, testing and operations can collaborate more effectively (see Figure 1). DevOps is more about trust, people and teamwork than about process, and creating software as an ongoing service, not a static product.

While there are several interpretations of the DevOps definition, all share the same working understanding. Gartner, for instance, defines DevOps as "... a change in IT culture, focusing on rapid IT service delivery through the adoption

of agile, lean practices in the context of a system-oriented approach. DevOps emphasizes people (and culture), and seeks to improve collaboration between operations and development teams. DevOps implementations utilize technology – especially automation tools that can leverage an increasingly programmable and dynamic infrastructure from a lifecycle perspective."<sup>2</sup>

DevOps has emerged for the following reasons:

- The existence of traditionally siloed team structures that did not scale to meet the varied needs of modern enterprises.
- The philosophical disconnect between development, testing and operations teams, which resulted in poor communication, collaboration and integration.
- The accelerating progression of digital technologies, which has evolved faster than the underlying processes used to deploy, extend and manage them.

From our perspective, four key activities define DevOps (see Figure 2). Further, the role of testing in DevOps can be best understood by comparing how testing is approached in Agile vs. DevOps, as illustrated in Figure 3 (next page).

## DevOps: An Intersection of Functions

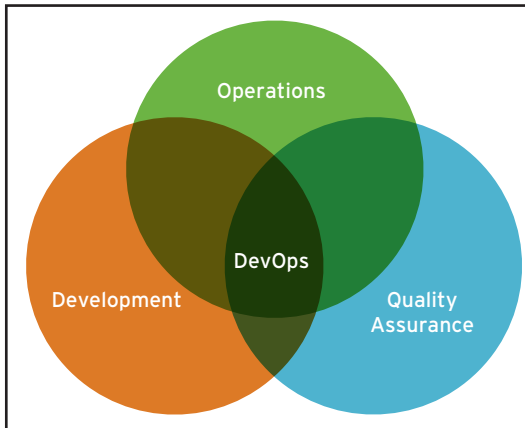


Figure 1

## DevOps at a Glance

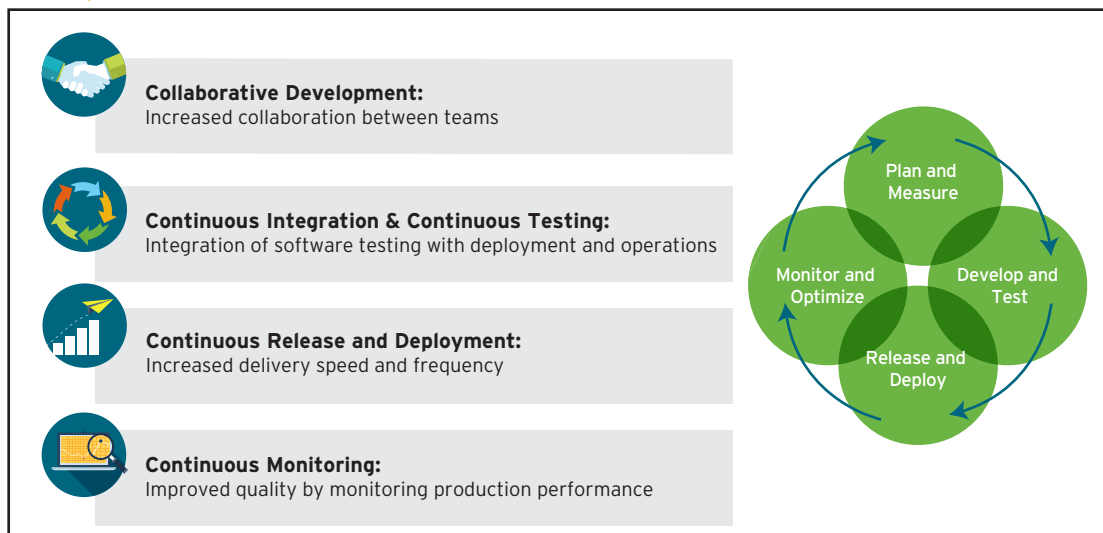


Figure 2

## DevOps vs. Agile

Testing in Agile	Testing in DevOps
Test as early and as often as possible	Test continuously
Automate testing as much as possible	Automate almost everything
Continuous integration and testing is a step forward	Continuous integration and testing is mandatory
Potentially shippable code at the end of a sprint	Potentially shippable code following every integration

Figure 3

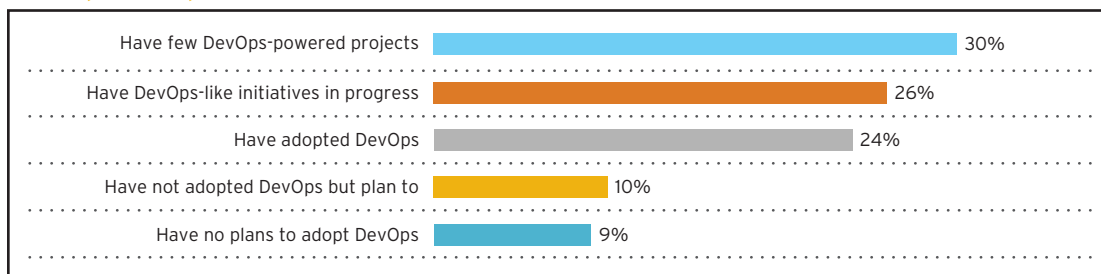
## DevOps: An Industry Perspective

According to a recent industry report published by online educational resources purveyor DevOps.com,<sup>3</sup> over 70% of IT organizations have embraced DevOps in some form. The rush to DevOps is on, and for some, the journey has just begun (see Figure 4).

As illustrated in Figure 5, the three main focal points of DevOps practices are automated build and testing, configuration management, and continuous integration/continuous deployment. As a result, a new software development lifecycle (SDLC) is emerging, according to Forrester Research (see Figure 6, next page). According to a Forrester<sup>4</sup> survey, the characteristics of the new SDLC include:

- **Faster release cycles:** More than 24% of companies have daily and weekly software releases.
- **Changing requirements and lack of coordination between groups:** Roughly 47% believe these disconnects were primary points of failure during a release review.
- **The emergence of DevOps:** Approximately 66% of CIOs have this on their IT agenda.

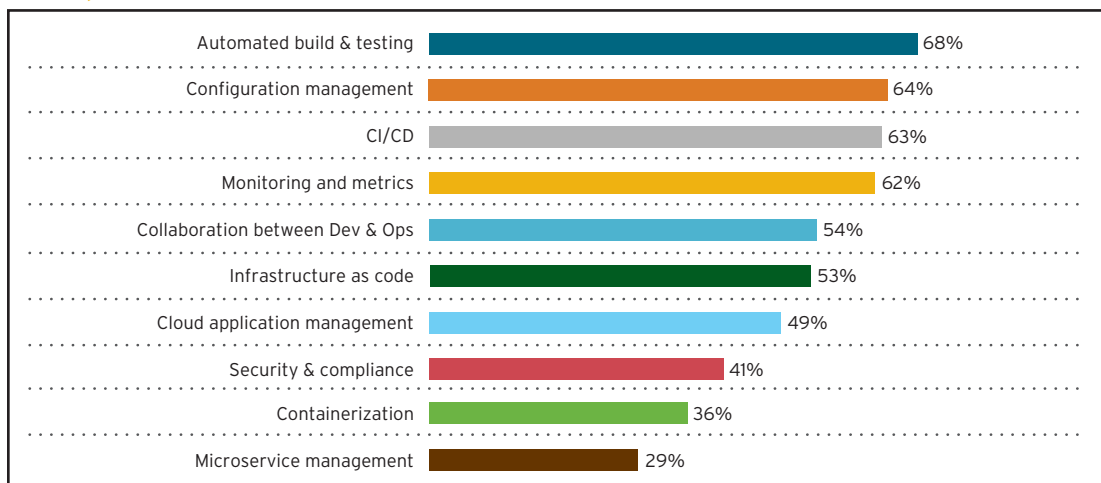
## DevOps Adoption 2015



Source: DevOps.com and ElasticBox, <http://devops.com/the-real-value-of-devops-download/>. Response base: 300 participants, including DevOps specialists, IT engineers and executives across a wide range of organizations.

Figure 4

## DevOps at a Glance



Source: DevOps.com and ElasticBox, <http://devops.com/the-real-value-of-devops-download/>. Response base: 300 participants, including DevOps specialists, IT engineers and executives across a wide range of organizations.

Figure 5

## A New Software Development Lifecycle Is Emerging

Old SDLC	New SDLC
Directive	Adaptive
Task-oriented	Goal-directed
Specialized roles	Empowered teams
Resistant to change	Optimally responsive
Outsourced	Automated
Project optimization	Portfolio optimization

Source: Forrester Research  
Figure 6

### Adoption Challenges

Organizations born in the cloud era (aka, digital natives) adopt DevOps from day one; as such, DevOps practices match more organically with their DNA. DevOps is also gaining momentum with legacy organizations but at a slower pace due to a variety of challenges.

The reasons are manifold. For starters, applications that large companies depend on typically contain extremely complex business logic; many have been in use for more than a decade and have been updated and extended with new mission-critical functionality. Additionally, there are development, operations and testing team processes and practices, as well as the organization's expectations that revolve around them. The Agile movement, for instance, has radically changed how applications are built and deployed by more closely aligning IT with the business, while operations and QA remain in silos.

Typical challenges in IT organizations seeking to embrace DevOps include:

- **Traditional Ops:** This group spends the majority of its time putting out production fires, dealing with recurring failures and making urgent fixes.
- **Test environment overhead:** Each environment (testing, development, pre-production/staging and production) is unique. Operations must have a laser focus on production environment issues and, as such, does not have time to work on improving infrastructure.
- **Traditional testing:** Several thousand inter-dependent regression test cases run through the application user interface for several days, causing longer test cycles and manual handoffs, resulting in much slower feedback. A loosely defined test data management strategy and downstream/upstream dependen-

cies without service virtualization techniques cause testing delays.

- **Traditional development:** Late code integration results in integration issues and late defect fixes, leading to longer cycles. Schedule delays often occur due to the absence of automated build verification and deployment procedures.

A huge technical debt is accumulating in these IT disciplines,<sup>5</sup> and collaboration is often lacking between IT teams. The solution is to build DevOps practices incrementally, one step at a time.

### Continuous Testing: A Step in the Right Direction

In our view, continuous testing (CT) is the first step in the right direction when embarking on a DevOps journey. CT is a metaphor for a continuous feedback mechanism that drives software delivery through the SDLC tunnel. Automated feedback at each checkpoint is an auto-trigger for the next process in the delivery chain if the feedback is to move forward, or green. If the feedback is to not move forward, the process immediately is stopped, and corrective measures are taken.

Traditional IT organizations can shorten the path to implement CT by reusing and realigning existing test automation capabilities (as revealed in Figure 7, next page).

Establishing a CT ecosystem involves the following steps at a high level:

- **Shifting automation test scripts to an enterprise version control tool** to establish a single source of truth. Testing organizations have commonly stored automation scripts in test management tools or a shared folder structure. However, an automation code base, just like an application code base, should reside in a version control repository.
- **Integrating the automation suite with a build deployment tool** to enable centralized execution and reporting.
- **Classifying the automation suite in multiple layers of tests** to enable faster feedback at each checkpoint. Typical tests include:
  - **Health check:** An automated check verifies services are up after deployment. Typically, health checks run for just a few minutes.
  - **Smoke tests:** This most critical set of automated tests ensures that key system features are operational and no blocking

defects occur. Smoke tests are typically executed in less than 15 minutes, and as the CT process matures, the response time should be further optimized.

- **Intelligent regression:** If execution time for overall regression is significantly high, CT setup becomes less effective due to longer feedback cycles. In such a scenario, a subset of regression carved out at run-time based on criticality and impacted areas can only be executed to provide feedback within a reasonable timeframe. Full regression execution can be shifted to overnight or during the weekend depending on its

alignment with recurring build frequencies.

- **Full regression:** This is the final feedback from the CT ecosystem. The goal is to minimize feedback time by running a parallel execution of automated tests through multiple threads or machines.

All of the above tests run as part of the build deployment process. If any fail, the deployment process is halted, and everyone involved in delivering software - developers, testers and operations staff - is notified immediately, triggering corrective action. Shorter feedback loops enable the team to fail fast and recover quickly.

## Transitioning to DevOps

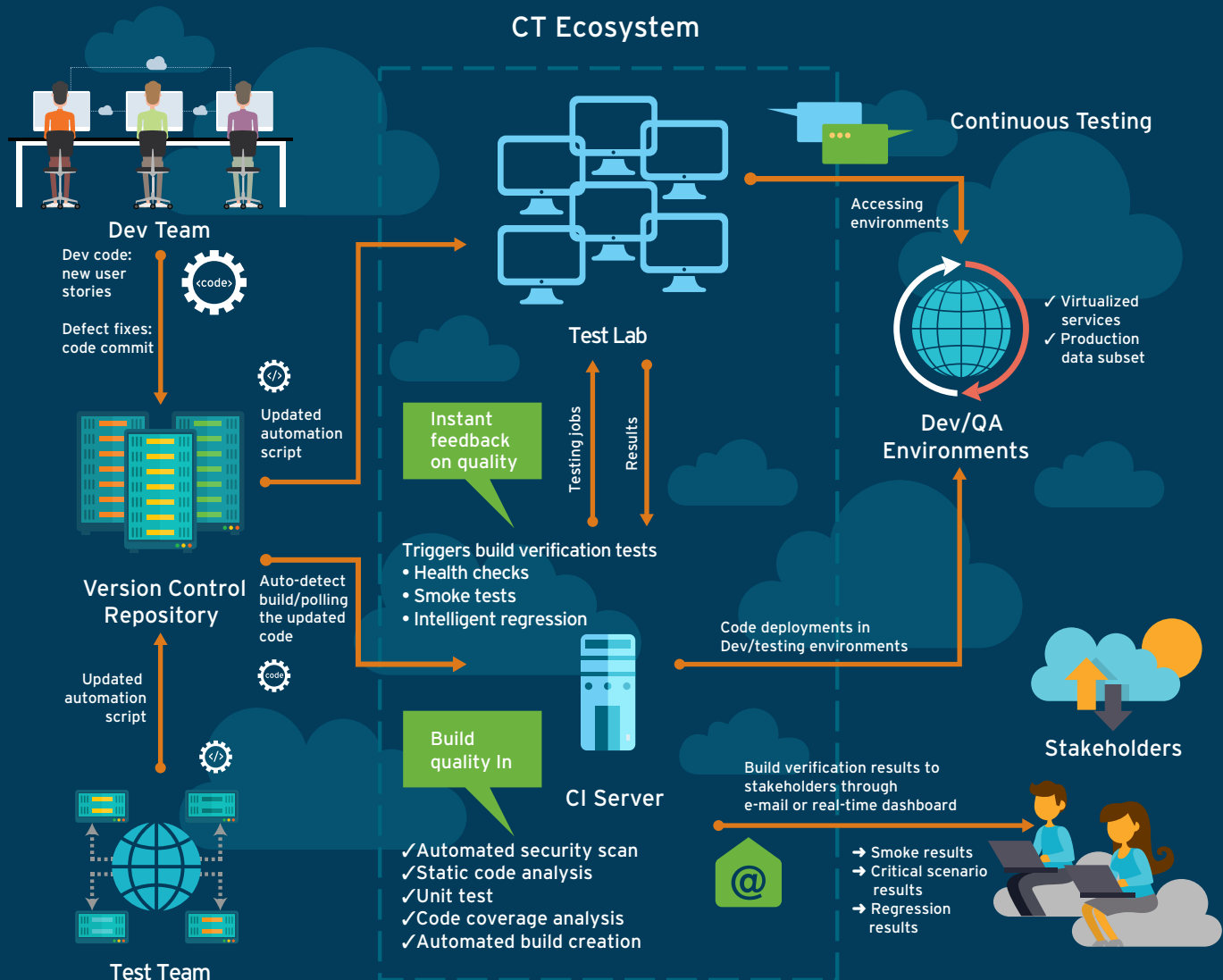


Figure 7



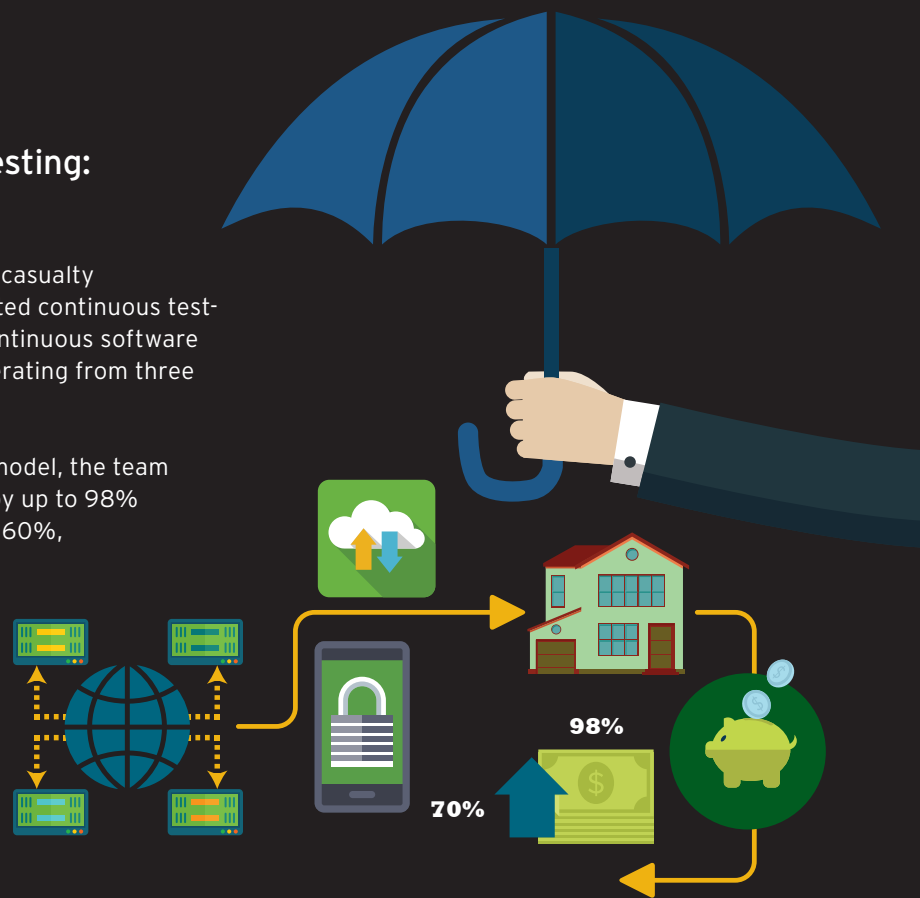
## Quick Take

### The Virtues of Continuous Testing: Improved Availability, Faster Deployment

A leading North American property and casualty insurance carrier leveraged our automated continuous testing solution to sustain the fast-paced continuous software delivery driven by distributed teams operating from three different geographies.

Through successful adoption of the CT model, the team improved test environment availability by up to 98% from 70%, reduced deployment time by 60%, and saved 800 hours per release, spanning 10 weeks.

A key success factor was the team's ability to deliver new features more frequently and get immediate feedback in minutes through continuous frictionless automated testing.



### Intelligent Test Automation Is Key

Test automation is a clear winner over manual processes in DevOps organizations. But automation tests commonly face the following challenges:

- Tests built on commercial tools slow down over time and face scalability challenges, often resulting from license constraints.
- Often, automation is built using different tools for UI, APIs, mobile coverage, etc.
- The business's influence over regression definition often pushes testing teams to build long-running tests.

Introducing intelligence into existing automation tests and migrating tests to more open source tools such as Selenium can further enhance the effectiveness of a DevOps setup.<sup>6</sup>

As Figure 8 shows (see next page), establishing intelligent regression models, such as a Monday-to-Friday weekend model, is critical to providing early testing feedback in the CT process.

Subsets of the total regression suite (smoke tests, health checks and intelligent regression sets) should be carved out to accelerate test runs during weekdays, using principles such as:

- A dynamic regression scope for each build based on the type of code change; automatically generated release notes will feed into the regression job.
- An intelligent test automation framework capable of turning test scenarios on and off based on application changes.
- Test cases traceable to the code files.
- A centralized repository of code-to-test case traceability.
- A risk-based approach with near zero risk.

### Integration of Nonfunctional Testing with CT

The core philosophy of CT processes is to test every change made to the application under test as early as possible. However, if nonfunctional tests such as performance and security are not included in the overall CT process, only a portion of the puzzle is solved. Issues identified later can override all CT efficiencies gained from a functionality standpoint and still jeopardize the release schedule.

As CT practices evolve and mature, the ability to integrate nonfunctional testing into the overall

## Intelligent Regression Model Testing Schedule: An Illustrative View

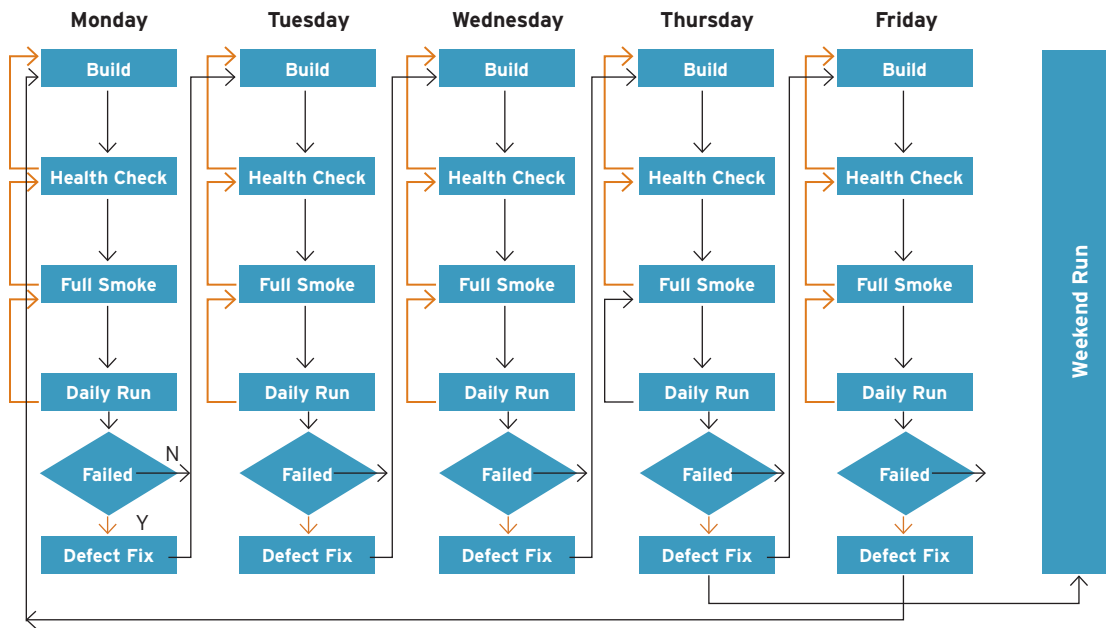


Figure 8

CT process remains an outlier but is seen as a core best practice. However, there are known challenges for integrating nonfunctional tests into the CT process, especially in load testing, such as:

- Non-availability of dedicated servers to generate the desired user load.
- Capacity constraints impacting the ability to scale the CT environment and sustain the size of load tests.

- On-demand utilization of monitoring tools to identify the bottlenecks.

In order to effectively utilize available resources and establish the ability to run the load tests on-demand without wasting time and resources, organizations should consider building a cloud infrastructure. Tools such as Apache JMeter are quickly emerging as a tool of choice for performance testing in DevOps organizations.

## Quick Take

### The Virtues of Continuous Testing: Accelerated Error Detection

Our Quality Excellence and Assurance business unit helped a leading North American property and casualty insurance carrier identify defects 200% faster in its claims center by establishing the culture of failing fast through a continuous testing model empowered by an intelligent regression model.

Through intelligent regression, updated application code was implemented by the automated tests in a few minutes post-deployment, compared with days in the past.



## Looking Forward

The notion that traditional IT organizations cannot do DevOps without throwing away what they have built over a period time is a myth. Testing teams can play a key role in overcoming the challenges of transitioning to DevOps by orchestrating a CT infrastructure and creating an early feedback loop with development. By successfully implementing CT as a first step, the dots of DevOps can be connected with development and operations to create automated build deployments, code coverage analysis, production monitoring and on-demand environment provisioning – enabling a full cycle of CT and CD to be established.

In summary, unlike any other tech movement, DevOps is not only driven by a passionate and enthusiastic tech crowd, but it has also

resonated with business groups, many of which are first-line customers of IT's latest digital products and services.

Gene Kim, co-author of *The Phoenix Project*, describes DevOps as important to businesses today because it solves the most important business problems of our generation – enabling organizations to make the transition from good to great.<sup>7</sup> Decision-makers often wonder about how much return on investment a DevOps establishment can provide. According to Kim, the opportunity cost of wasted IT spending is some \$2.6 trillion – which is enough motivation for every company to embrace DevOps because after all, every company is an IT company.

## Footnotes

- <sup>1</sup> DevOps is an approach to software development that is focused on streamlined communication, collaboration, integration, automation (of testing as well as coding) and measurement of cooperation between software developers and other IT functions. The term was popularized through a series of “DevOps Days” starting in 2009 in Belgium. Since then, DevOps Days conferences have been held in many countries worldwide. Source: <https://en.wikipedia.org/wiki/DevOps>. For more on DevOps, read our white papers “How DevOps Drives Real Business Growth,” <http://www.cognizant.ch/InsightsWhitepapers/How-DevOps-Drives-Real-Time-Business-Growth.pdf>, and “DevOps Best Practices Combine Coding with Collaboration,” [http://www.cognizant.ch/InsightsWhitepapers/DevOps-Best-Practices-Combine-Coding-with-Collaboration\\_nn.pdf](http://www.cognizant.ch/InsightsWhitepapers/DevOps-Best-Practices-Combine-Coding-with-Collaboration_nn.pdf).
- <sup>2</sup> “DevOps,” Gartner, <http://www.gartner.com/it-glossary/devops/>.
- <sup>3</sup> “The Real Value of DevOps: 2015 Survey Report,” DevOps, <http://devops.com/the-real-value-of-devops-download/>
- <sup>4</sup> Kurt Bittner with Phil Murphy, John R. Rymer, Jeffrey S. Hammond, Tom Grant, Ph.D., Steven Kesler, “Continuous Delivery Is Reshaping the Future Of ALM,” Forrester Research, July 2013, <https://www.forrester.com/Continuous+Delivery+Is+Reshaping+The+Future+Of+ALM/fulltext/-/E-res99321>.
- <sup>5</sup> Technical debt is a widely used term that describes the material liability or exposure inherent in an enterprise’s current IT. It was originally an estimate related to the code or software development needed to complete an application, or elevate it to acceptable quality, performance and sustainability. Many organizations now view it as the estimate of the financial costs and effort needed to remediate the systems, infrastructure, data, tools, skills, IT processes and governance to bring the enterprise’s IT to a point of industry parity.
- <sup>6</sup> Selenium is an open source suite of tools (<http://www.seleniumhq.org/>) to automate Web browsers across many platforms. Selenium runs in many browsers and operating systems and can be controlled by many programming languages and testing frameworks.
- <sup>7</sup> Gene Kim, Keving Behr, George Spafford, *The Phoenix Project: A Novel about IT, DevOps and Helping Your Business Win*, IT Revolution Press, January 2013, <http://itrevolution.com/books/phoenix-project-devops-book/>.



## About the Authors

*Sujay Honnamane is an Associate Director within Cognizant's Quality Engineering and Assurance Business Unit. He has over 14 years of experience in the information technology field, of which the last 10 years have been with Cognizant. Sujay has helped numerous Cognizant clients implement successful transformation programs and establish world-class testing centers of excellence. As a thought leader in global delivery and establishing managed test centers, Sujay brings expertise in the fields of test consulting, test automation and DevOps strategy. He has an engineering degree in electronics and communication from Bangalore University, India. Sujay can be reached at [Sujay.Honnamane@cognizant.com](mailto:Sujay.Honnamane@cognizant.com).*

*Rameshkumar Bar is a Senior Automation Architect with Cognizant's Technology Center of Excellence within its Quality Engineering and Assurance Business Unit. He has 13 years of experience in the information technology field, of which the last five years have been with Cognizant. Ramesh has extensive experience in developing Web applications on the .NET platform, building test automation frameworks using commercial and open source tools. His areas of specialization include test automation, continuous integration and test data management. Ramesh received an engineering degree in information technology from Saurashtra University, India. He can be reached at [Rameshkumar.Bar@cognizant.com](mailto:Rameshkumar.Bar@cognizant.com).*

---

## About Cognizant

Cognizant (NASDAQ: CTSI) is a leading provider of information technology, consulting, and business process outsourcing services, dedicated to helping the world's leading companies build stronger businesses. Headquartered in Teaneck, New Jersey (U.S.), Cognizant combines a passion for client satisfaction, technology innovation, deep industry and business process expertise, and a global, collaborative workforce that embodies the future of work. With over 100 development and delivery centers worldwide and approximately 219,300 employees as of September 30, 2015, Cognizant is a member of the NASDAQ-100, the S&P 500, the Forbes Global 2000, and the Fortune 500 and is ranked among the top performing and fastest growing companies in the world. Visit us online at [www.cognizant.com](http://www.cognizant.com) or follow us on Twitter: Cognizant.



**Cognizant**

### World Headquarters

500 Frank W. Burr Blvd.  
Teaneck, NJ 07666 USA  
Phone: +1 201 801 0233  
Fax: +1 201 801 0243  
Toll Free: +1 888 937 3277  
Email: [inquiry@cognizant.com](mailto:inquiry@cognizant.com)

### European Headquarters

1 Kingdom Street  
Paddington Central  
London W2 6BD  
Phone: +44 (0) 20 7297 7600  
Fax: +44 (0) 20 7121 0102  
Email: [infouk@cognizant.com](mailto:infouk@cognizant.com)

### India Operations Headquarters

#5/535, Old Mahabalipuram Road  
Okkiyam Pettai, Thorapakkam  
Chennai, 600 096 India  
Phone: +91 (0) 44 4209 6000  
Fax: +91 (0) 44 4209 6060  
Email: [inquiryindia@cognizant.com](mailto:inquiryindia@cognizant.com)