DevOps with Quality

Achieving the desired quality at every stage of the DevOps lifecycle.

DevOps: increasing Speed and Quality

DevOps is being embraced by the IT industry as the answer to fulfill the continuous demand from the Business to deliver new and improved IT solutions to end-users and customers at ever increasing speed.

The core idea behind DevOps is to remove the boundaries between the traditional silos of Business, IT Development, QA and IT Operations and to extend the agile and lean-principles from software development to the software deployment phase. This would ultimately lead to an IT process with a continuous stream of IT updates to the end-user. The proliferation of DevOps is driven by time to market with business metrics, such as Return on Revenue and Return on Investment (ROR/ROI) for new development, and throughput efficiencies for legacy systems. In our view, to begin to realize the benefits promised by the DevOps philosophy, automation of the quality activities is not only required but it is the core enabler of increasing throughout and velocity. Without maximizing automation, the speed at which a team can deploy features is limited by the speed at which the quality activities can be successfully completed. In other words, testing activities, done traditionally will become the constraining factor or the alternative is testing will not be done adequately, therefore putting your organizations reputation at risk.The 2015 World Quality Report reveals some significant data points to support this view:

- 61% of organizations rate, time to market as a very important part of their corporate strategy
- Corporate image is the number 1 executive concern when it comes to quality, demanding protection from software glitches that hit the news
- Customer Experience or Customer Value is a key objective, and this is determined by speed, fit for purpose solutions and ease of use.

However, the same World Quality Report survey also shows that achieving the appropriate level of quality validation within these ever shorter cycles is a serious challenge for most organizations:

- 49% of organizations complain that still largely manual testing phases are a bottleneck in speeding up the development cycle times
- 33% of organizations have difficulty to determine the right coverage of quality validation checks in Agile and DevOps processes
- 31% of organizations claim that test environments and test data sets are not flexible enough to quickly test micro-services.
- Proliferation of 3rd party services integrated into applications increase the risk that changes in these services cause unexpected failures

Not overcoming these challenges means that organizations either have to accept higher risks of IT product release failures – with potentially serious consequences for business operations and corporate image, or have to accept lower speed in adapting new technologies – with potential risk of losing to the competition.

In this point of view document we will offer some concrete solutions that will enable the achievement of the benefits of the DevOps philosophy while minimizing the risk to brand image and user experience.

The characteristics of DevOps

Business Agility and business speed to market is more than ever linked to agility of IT provisioning:

- Monolithic legacy systems and lengthy release cycles often impede an organizations ability to quickly adapt to new business processes, because it is still common for releases to be done quarterly or even only twice a year
- Newer and more nimble solutions provide faster ROI through operational improvements
- Agile methods deliver usable product to market in smaller, faster increments and reduce risk by identifying issues earlier in the lifecycle
- Lean processes and advanced automation tooling empower organizations to adapt to changes at a previously unachievable rate
- Future advancements in testing will include the incorporation of artificial intelligence and predictive analytics that will allow even further improvement in automation of delivery.

By adopting DevOps an organization can dramatically improve the value delivered by its business. DevOps is based on two principles

- Collaboration and industrialization leveraging highly automated approaches to deploy solutions that evolve as fast as the business needs it.
- A team centric ethos that tears down traditional silos to tightly integrate business, development and operations to drive agility and service delivery excellence across the entire lifecycle.

Implementation of DevOps requires a changed focus on three aspects

- **Cultural:** Business/Development/QA and operations must work in a fully integrated fashion in the software delivery process; Change and adaptability are the norm.
- **Process:** an agile based approach with continuous development, continuous build, continuous integration, continuous testing, continuous deployment and continuous monitoring
- **Tools:** implementation of a flexible tool environment that allows maximum automation In order to achieve the desired pace of delivery while protecting the brand and user experience.

DevOps is a revolution for all disciplines: Business, Development, Operations and Quality Assurance. All disciplines come together and will integrate extensively to make DevOps work. Each has a role to play.

We believe that four key aspects must be made part of your DevOps process, teams and infrastructure in order to achieve DevOps with the appropriate and fit-for-purpose level of quality:

1. Implementing an integrated DevOps Quality Approach
2. Assigning the role of DevOps Quality Engineer to DevOps teams
3. Providing a coherent yet easy adaptable DevOps Quality Automation Framework
4. Providing a one-click DevOps Test Environment and Test Data solution to teams
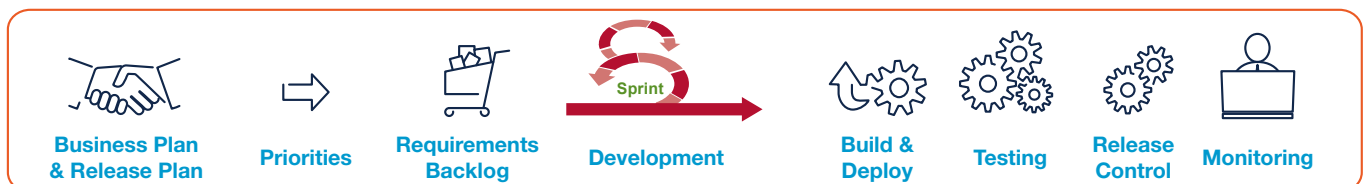
Implement a DevOps Quality Approach

We believe an integrated DevOps Quality Approach is based on seven key principles

1. **Continuous Testing and Continuous Quality Monitoring** are completely connected to the continuous development, continuous build and continuous deployment process.

2. **Continuous Development based on TDD (Test Driven Development) and BDD (Behaviour Driven Development)**. These methods enable relatively quickly a common shared understanding of the expected operations and functionality of the application among all team members (users, developers, testers and operations). By extending these descriptions with priorities and potential risk identifications, they can be used as input to automatically generate test cases.

3. **Quality is everyone's responsibility**. At each step in the DevOps cycle team members execute quality checks: At continuous build the focus is to validate whether this build could break operations; at continuous development the focus is to validate functionality and usability. At continuous integration the focus is to validate performance, security and usability.

4. **Clear understanding of priorities and potential risks to customer value and business performance**. The quality ambition of DevOps deployments must be targeted at the level "good-enough". Pre-requisite is an agreed and commonly shared vision on risks that are acceptable and risks which cannot be taken with a deployment. The agreed priorities and identified risk areas are the common starting point for all team members. A priority/risk charter per application which is automatically updated based on

Figure 1. DevOps Quality Approach

Business Plan & Release Plan → Priorities → Requirements Backlog → Development (Sprint) → Build & Deploy → Testing → Release Control → Monitoring

a. **Zero Touch Continuous Testing:** A fully automated process for testing is in place, covering all steps of the test process: from test strategy, test planning, test case generation, test environment set up, test data provisioning, to test execution and covering functional as well as non-functional testing aspects (especially performance and security). Quality Analysis tools are used to switch on and switch off validation checks with a push on the button. Newly defined/created quality validation checks are automatically absorbed in the full available set of quality checks. The test and quality checks are focused at: essential functional operations, performance, security and ease of use.

b. **Continuous Quality Monitoring;** The actual quality level of applications is continuously monitored based on instantaneous visibility of production incidents, user reports.

customer and end-user feedback is used to enable this. Predictive analysis from combined logging of historical project and deployment data-facts with actual quality monitoring scores is used to automatically identify the risk and focus areas of test and the selection of the appropriate test cases.

5. **Quality Indicators per application**. Per application there is a defined set of measurable quality indicators with pre-defined targets. The standard quality indicators cover four aspects: fit for purpose, ease of use, performance and security. This standard set can be expanded depending on the specific objectives of an application. The operational scores of the quality indicators are monitored continuously based on production incident reports and end-user feedbacks. Meeting the quality KPI's is a shared responsibility for all team members.

Figure 2. DevOps Quality Metrics Model

Test Execution Productivity	Sprint / Iteration	Engagement Level Metrics	Effort Variance	
Velocity			Schedule Variance	
			Commitment Reliability	
Defect leakage	Release	Business KPI	Number of Releases / Year	
Severity 1 and 2 defects			Reduction in Defects and Support Calls	
Features delivered per release			Release Predictability Measure	

6. **Flexible quality coverage decision**; For each DevOps deployment cycle the appropriatequality validation checks and test cases are defined and/or selected based on objective rules and with help of predictive analysis tools.

7. **Implement metrics to manage QA efficiency:** A small set of metrics is used to manage the efficiency of Quality activities. These metrics cover four aspects: Sprint/ Iteration performance (measured through Test Velocity and Test Execution Productivity); Release Quality (measured through the metrics: Defect leakage, Number of high Severity Defects, Number of features delivered) Engagement Level (Effort variance, schedule variance and commitment Reliability) and Business Level, which is measured by Number of releases per year, Reduction in Defects and Production calls, Release Predictability). These metrics are continuously reported in a real-time dashboard.

Figure 3. Measure of Success

Success Levers		Basic	Intermediate	Advanced
Lifecycle Automation	Functional, non functional, Data , UAT		50%	70+%
Continuous Test	Build, integrate and deliver		70%	100%
Lean and Agile adoption	Kanban, Scrum,Safe,XP,SaFe		40%	70+%
Test virtualization	Services and Environment automation		30%	40+%
Test optimization	Risk based testing and OA leverage		40%	50%
Standardization	Model based work benches		40%	60%
Continuous monitoring	Availability & sentiment analysis		30%	60+%
QE leverage	TDD, BDD, Unit automation		70%	100%

The key success factors to achieve DevOps with Quality are:

- Lifecycle test automation
- Continuous Test
- Lean and Agile process adoption
- Test Virtualization
- Test Optimization and Standardization
- Continuous Monitoring
- Acknowledge the role of Quality Engineer

Assign the role of DevOps Quality Engineer to DevOps teams

DevOps with Quality requires the presence of multiple skills within the teams:

Business and domain skills: DevOps means anything that eliminates bottle-necks between concept to deploy. Without understanding the unique needs of domains, it is impossible to design an appropriate quality strategy that drives validation activities upstream.

Architecture skills: Think modular, think open, think abstraction, think service. Build object oriented, service oriented loosely coupled architectures with micro-services. Ensure sufficient middleware skills to test infrastructure early.

Infrastructure, cloud and virtualization skills: Test environments are typically the largest bottlenecks in testing. Understanding automated environment provisioning, cloud architectures and environment virtualization, can improve delivery pipeline by a significant 25% or more.

Native programming languages skills: Coding skills in native programming is necessary to move away from user interface (UI) based test automation and to work more closely and seamlessly with developers.

Developing non UI based automation framework skills, using proven data driven, keyword driven and hybrid automation frameworks will be the way forward.

Enterprise level hybrid agile testing skills: Analysis shows that agile development is being driven by digital programs and mobile initiatives, with the Scaled Agile framework (SAFe) methodology taking the biggest market share (31%). Typically organizations are using a combination of DSDM, SCRUM, XP noting that SAFe is enterprise driven.

DevOps with Quality can only be achieved by assigning the central role of DevOps Quality Engineer in each DevOps team. The DevOps Quality Engineer is a team member who contributes to development and operations – but with focus on quality validation. (Some teams use the term SDET (Software Development Engineer in Testing) for this role. The DevOps Quality Engineer is never the sole person responsible for quality – since quality is a shared responsibility of everybody in the team. The DevOps Quality Engineer plays a vital enablement role for the whole team to achieve the required quality standards with each DevOps cycle. The most important responsibilities of the DevOps QA engineer are:

1. Provide QA checkpoints/guidelines for all processes and all team members
2. Provide a common backbone QA Automation platform
3. Provide flexible QA Environments with service virtualization and cloud based environments
4. Run end-to-end QA validation checks per application
5. Validate correct Quality Coverage across the DevOps cycle
6. Validate and Monitor the overall Quality level of the applications

Empowerment of teams is an important aspect of DevOps. Efficiency in the empowerment of the QaA activities requires a base set of re-usable guidelines, frameworks and best practices. To manage and provide this a DevOps QMO (Quality Management Office) could be established. This QMO is not intended to act as a new "silo" with execution responsibility for quality validation activities, as its core function is to provide the QA validation standards and core tool platform and frameworks across teams.

Figure 4. DevOps Quality Engineer Role
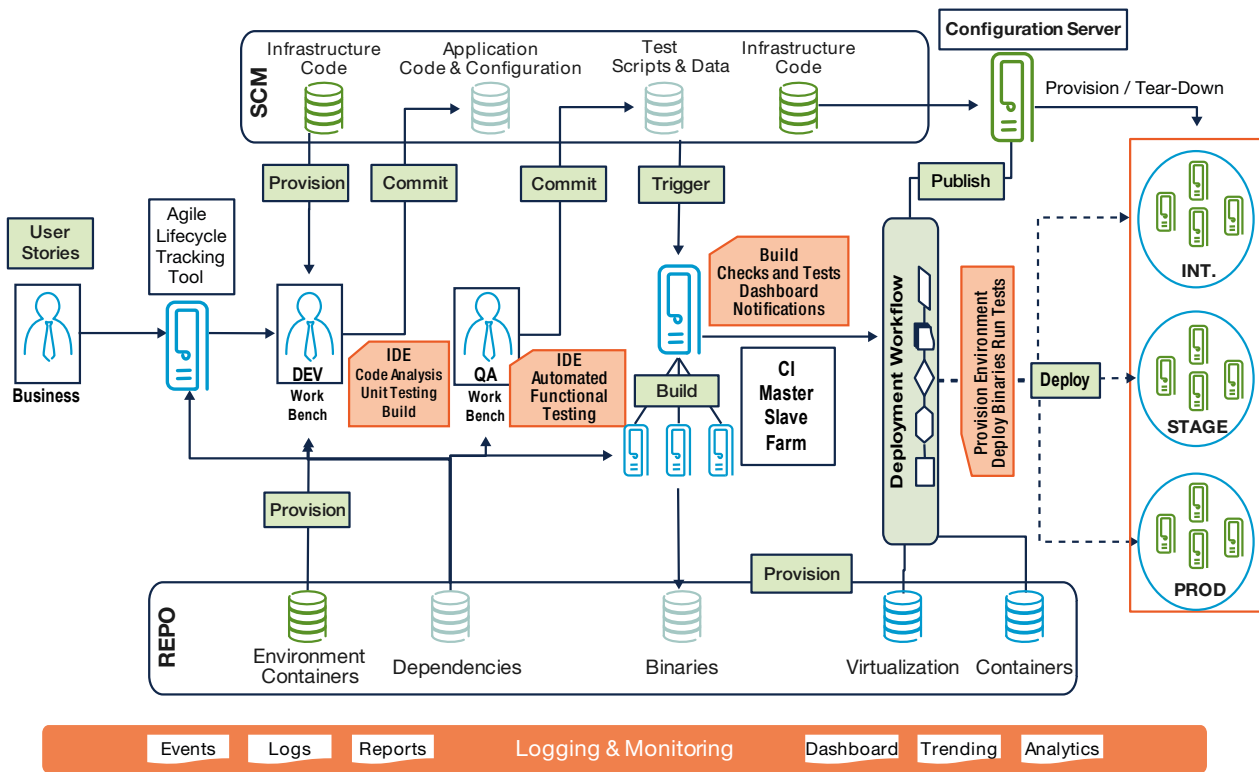
Set up flexible DevOps Quality Automation Framework

The enhanced DevOps process requires maximum automation of all parts of the process. From User Story gathering to Continuous Development, Continuous Build, Continuous Deployment and Continuous Monitoring. The more automation that can be achieved, the more the benefits promised by DevOps can be realized.

A multitude of tools is already available to support these activities, and many more will be emerging in the near future.

A proper set up of the automated quality environment is best served by a backbone of corporate agreed tool framework, with enough freedom to individual teams to plug-in special tools.

In order to enable continuous improvement of team performance and implement corporate governance across this landscape of tools, there is a need for one corporate level standardized dashboard of quality metrics. This corporate dashboard allows for automatic and continuous reporting of source data from all the different tool repositories.

The commonly used tools for managing and realizing quality in DevOps environments today are

Components	Indicative Tools
Test Driven development	JUnit, xUnit,
Behavior driven development	Cucumber, JBehave
Exploratory Testing	HP Sprinter
Configuration Management	GIT, PVCS, VSS,
Continuous Integration	Jenkins, TFS
Build automation	Make, GNU make, Apache
Test lifecycle and defect management	HP QC, Rally, Jira, VersionOne
Service automation	Parasoft, HP Service Test, CA Lisa
Functional automation	HP QTP, HP UFT, Selenium
Performance test	HP Loadrunner
Continuous Monitoring	Application, Server, Database Monitors
Environment virtualization& Cloud leverage	Puppet, Chef, VM ware , HP Codar
Service Virtualization	CA Lisa
Test Data Management	IBM Optim, Parasoft File-Aid

Use a virtualized Environment for DevOps Quality Validation

A flexible and instantaneous DevOps Quality validation Environment is required to enable right coverage of validation checks in a repeatable, and cost –effective manner. Such an environment is based on three principles:

- Virtualization of Services, Network and Devices
- Cloud Based Environment Provisioning
- Test Data Management and Generation.

Service Virtualization enables the setup of an environment in which individual applications functions and/or services can be tested as if they were embedded in a real life environment with all connections to other systems present. Service virtualization can be leveraged to record / create all the required behavior expected by dependent systems. Through a standard API set up each Application Under Test (AUT) communicates with the virtual services, instead of the real systems and this enables teams to perform early integration testing. Service Virtualization reduces the complexity of extensive test infrastructures and test data sets. It also reduces delays that are normally caused if teams need to wait for other teams to complete their services under development. Service Virtualization allows a working functional test environment to be made available with much more ease and at lower cost, and integration defects can be unveiled at earlier stages. The increased need for a test environment using service virtualization is pushed further due to movement to a software architecture based on microservices (small decoupled building blocks that interact with each other through language-agnostic API's).

Service Virtualization solutions are offered by key technology partners such as Hewlett Packard Enterprise, CA, IBM and Parasoft.

Network Virtualization enables the virtualization of mobile and broadband networks. This allows near real-life testing of application behavior and application performance in a controlled test environment.

Network virtualization is offered by multiple technology partners: Hewlett Packard Enterprise (HPE), Citrix, IBM, OpenSolaris.

Device Virtualization enables the virtualization of devices such as smart phones, tablets and other devices. It allows quality validation with a large number and versions of devices immediately and at lower costs.

Device Virtualization is offered by multiple technology partners: PerfectoMobile, VMWare, Xamarin, MobileLabs and Red bend.

Cloud-based Environment Provisoning. With cloud based environment provisioning it is possible to switch on or switch off personalized test environments with test automation tools on an as-needed basis. The test environments are delivered remotely through secure networks, reducing the need for onsite hardware installation, administration and maintenance.

So teams pay for the specific testing and access to dedicated hardware platforms and industry-leading software licenses – but only when they need them. Cloud-based Environment Provisioning is a critical enabler for DevOps Quality Validation.

- It provides instantaneous flexibility to switch-on or switch-off environments
- It provides fully equipped validation environments including core test automation tools
- It reduces complexity, cost and overhead in managing and maintaining test infrastructure

Test Data Management (TDM), ensures the provisioning and management of the test data for all DevOps test sets. Test Data Management offers the ability to (automatically) generate test data, fast test data sub-setting, test data depersonalization/obfuscation/masking, test data archiving and test data refresh. In short the test data management activity creates and manages high quality and consistent test data across all test environments to perform testing activities.

A well managed and automated Test Data Management process is a key enabler to

- Ensure the timely availability of test data
- Test data consistency across all test environments
- Sufficient volume of test data for real-life tests
- Predictive test results for test automation
- Re-usability of test data sets
- Adherence to data security and privacy regulations
- Test data creation/modeling as per business rules

Proposed Roadmap to achieve DevOps with Quality

For organizations that are implementing the DevOps philosophies, we recommend using our DevOps QBP approach to establish the baseline maturity, roadmap and continuous process for improvement in DevOps maturity.

The roadmap to DevOps with Quality will have to cover all the critical elements

- Integrated QA approach covering Continuous Testing to Continuous Quality Monitoring
- QA awareness and expertise among all disciplines
- Quality Engineering role in DevOps teams
- QA automation
- QA environment and test data provisioning

It is imperative to create business cases and DevOps driven roadmaps with right lean metrics aligned to the DevOps business objectives. This means shifting from the traditional "cost of quality business cases" to "cost to achieve desired test velocity business cases".

To find out how Capgemini and Sogeti's Testing Services can help your organization achieve its Testing and QA business goals, please contact your local Capgemini or Sogeti testing representative or our Global Testing Services Sales Team:

Mark Buenen
VP Global Service Line Testing
mbuenen@capgemini.com

Renu Rajani
VP Global Service Line Testing
renu.rajani@capgemini.com

Deepika Mamnani
Principal | Financial Services GBU
Deepika.Mamnani@capgemini.com

Steve Hughes
Senior Vice President | Testing Practice Manager, Sogeti USA
Steve.hughes@us.sogeti.com

About Capgemini and Sogeti

With more than 180,000 people in over 40 countries, Capgemini is one of the world's foremost providers of consulting, technology and outsourcing services. The Group reported 2015 global revenues of EUR 11.9 billion. Together with its clients, Capgemini creates and delivers business, technology and digital solutions that fit their needs, enabling them to achieve innovation and competitiveness. A deeply multicultural organization, Capgemini has developed its own way of working, the Collaborative Business Experience™, and draws on Rightshore®, its worldwide delivery model.

Sogeti is a leading provider of technology and software testing, specializing in Application, Infrastructure and Engineering Services. Sogeti offers cutting-edge solutions around Testing, Business Intelligence & Analytics, Mobile, Cloud and Cyber Security. Sogeti brings together more than 23,000 professionals in 15 countries and has a strong local presence in over 100 locations in Europe, USA and India. Sogeti is a wholly-owned subsidiary of Cap Gemini S.A., listed on the Paris Stock Exchange.

For more information, please visit:
www.capgemini.com/testing or www.sogeti.com/testing

MCOS_GI_AH_20160321