W18
Track
10/3/2012 3:00:00 PM

"Testing in the DevOps World of Continuous Delivery"

Presented by:
Manoj Narayanan
Cognizant Technology Solutions
Manoj Narayanan
Cognizant Technology Solutions

Manoj Narayanan is the director of testing services at Cognizant Technology Solutions, a leading provider of information technology and consulting services. Manoj is the QA practice leader for Cognizant’s retail and consumer goods, travel and hospitality, and manufacturing verticals in North America. He provides thought leadership and implementation assistance for his clients as they transition the QA organization toward the next maturity level. During his fifteen years in the IT services industry, Manoj has successfully played multiple roles including performance services evangelist, program manager, and transition advisor. Prior to Cognizant, Manoj was a management consultant with A T Kearney where he focused on business process re-engineering and risk optimization.
Testing in the **DevOps** world of Continuous Delivery

by
Manoj Narayanan
Senior Director –Testing Services, Cognizant

---

**What we will discuss today**

- Is there a need to change traditional testing approaches?

- What is DevOps - Is it different from Agile?

- How does it help the organization?

- Impact to testing in a DevOps scenario
  - People
  - Process
  - Governance
  - Technology
Traditional testing approach is geared towards sequential delivery

Development ➔ Testing ➔ Deployment
leads to leads to

Resulting in a siloed organization structure ... ... with communication challenges

Also leads to focus on discrete skill sets - minimizing cross functional behavior

Need to know Java, .Net etc

Limited opportunity for mutual assistance and sharing of ideas

Master of QA, but ignorant of all else
Result: Unintended Consequences!

What the end user wanted
What the developers built
What was tested
Finally Deployed!

Sequential delivery approach also results in longer delivery times & higher risk

Long wait time
Higher risk due to discrete functionality addition

Is this a viable option???
What does business want today?

Reducing Risk

- More than 60%

Responding Fast

- 24% Reducing risk by preventing poor quality from impacting satisfaction
- 22% Quickly responding to customer requests and requirements
- 17% Reducing costs by improving development and testing activities

Beat Competitors

- 14% On time budget
- 13% Development and QA
- 7% Competition
- 5% Issues

More than 60%

Source: QAI - Edista Testing

How does one achieve this?

Moving from a discrete delivery approach to...

.... a Continuous Delivery approach

From “Build it Right” to “Build the Right It”
Agile – Path to Continuous Delivery

From Discrete and Independent Testing to...

Waterfall Methodology

...Continuous and Integrated Testing

Scrum

Agile is a well known option. But it doesn't quite yet provide the final solution

“Last Mile Problem” still exists

Development

Testing

leads to

Deployment

- Long time between testing and deployment
- Testers still do not have SysAdmin skill sets
- Additional delivery pace adds to the confusion
DevOps aims to be the solution to this integration challenge

One Team: Development + Testing + Operations

One Team: Capable of playing all the roles

“Ensure the fastest path to deployment ready code”

Is DevOps different from Agile?

“Our highest priority is to satisfy the customer through Early and continuous delivery of valuable software”

- Agile Manifesto

1. Courtesy: Jez Humble
Who is doing it now?

- Web 2.0 firms with heavy reliance on eCommerce have been the front runners

- Fast changes are mandatory - the better the integration, the better the response time. **Hence DevOps.**

Impact on testing

- Change to a DevOps environment has to be gradual as it has multiple impact points

![People](People), ![Process](Process), ![Governance](Governance), ![Technology](Technology)
**Impact on testing**

- **People**
  - Focus on Knowledge Management
    - Information access to increase re-usability
    - Testers need to learn development languages – rise of user friendly tools like Python and Cucumber
    - Testers need to learn deployment process and tools
  - Train Developers & SysAdmin on test processes, design techniques, tools
    - Increasing reliance on developers for shift – left testing

**Impact on testing**

- **Process**
  - Continuous Integration becomes mandatory
    - Test Driven Development
    - Single source code repository
    - Automate the build process
    - Fast Build - What is my “zero release time”?
    - Everyone should know what is going on - transparency
Automation becomes critical

Heavy reliance on innovative automation embedded early into the life cycle

Need to move away from traditional approach focusing on GUI based testing.

...to focusing on increased defect capture through Unit and Service Layer Testing

Impact on testing

“Smart Testing” dissolving boundaries of traditional system & integration testing

QE Approach

QA Approach

User Acceptance Testing

Regression Testing (automation)

Beta Testing

Functional Testing (automation)

Service Layer Testing

Unit Testing
Leverage optimal mix of automation across lifecycle

- Re-usable, automated test cases used by developers
- Leveraging xUnit Frameworks

- Leverage re-usable frameworks & tools like Fitnesse & Cucumber
- Web Services testing
- Leverage Service Virtualization

- >90% automated regression tests
- 100% automated smoke testing
- Continuous validation of test flow relevancy

& Release Management Automation facilitating Continuous Integration

Need for “industrialization” – focus on getting application functionally and operationally ready

Additional types of testing needed
- Operational Readiness Testing
- Network & Geographical Testing
- Failover & Disaster Recovery Testing

Testing in the SysAdmin world!
Impact on testing

- Teams work best in pods of “jack of all arts”
  - Critical to have pods talking to each other
  - Central co-ordination needed – a rotating position?
  - Incremental changes driven by business – continuous interface and demos
- Focus on core competence increases!
  - Greater need for specialists - they will focus on challenging sub-tasks that pods cannot handle
  - Identification of repeatable tasks – automated and executed by pods
- Might not work in large organizations

Impact on testing

- Need for user-friendly tools that can be used by developers, testers and SysAdmin
- Greater focus on automation and re-usability
- Cost consciousness or the desire to be technically esoteric!
And most importantly, do not forget to communicate

• Ensure Executive & Stakeholder buy-in on the change

• Establish a communication channel for regular updates and feedback

Summary

• Traditional testing approaches are limited by their long delivery timeline and inefficient cross pollination of work

• Agile works better, but faces challenges in the “last mile”

• DevOps provides an integrated approach to delivery

• Focus on Continuous Integration and Continuous Delivery

• Need for extensive automation & re-usability

• Teams work in pods performing all roles – however, specialists are now more in demand

• Transition to DevOps is facilitated by the rise of a new breed of user and business friendly tools

• Effective Communication & Change Management – very critical for an organization embarking on this journey