DevOps: Are You Pushing Bugs to Your Clients Faster?

Wayne Ariola
Chief Strategy Officer - Parasoft
Continuous Testing provides a real-time, objective assessment of the business risks associated with an application under development. Applied uniformly, Continuous Testing allows both business and technical managers to make better trade-off decisions between release scope, time, and quality.

Generally speaking, Continuous Testing is NOT simply more test automation. Rather, it is the reassessment of software quality practices—driven by an organization’s cost of quality and balanced for speed and agility. Ultimately, Continuous Testing can provide a quantitative assessment of risk and produce actionable tasks that will help mitigate these risks before progressing to the next stage of the SDLC.

Read this 44-page eBook to learn how Continuous Testing can help your organization answer the following questions at the time of the critical “go / no-go” decision for a software release candidate:

- Are we done testing?
- Does the release candidate achieve expected quality standards?
- What are the quantifiable risks associated with the release candidate?
- How confident are we that we won’t end up in the news for software failures?

http://alm.parasoft.com/continuousetestingbook
OF 780 Respondents
What best describes your role in your organization?

- Architect: 15%
- Business Analyst/Product Owner: 6%
- Developer/Engineer: 32%
- Tester: 26%
- Senior Management: 17%
- IT Ops: 4%
Does your organization measure or monitor compliance to NON-functional requirements?

- **No**: 56%
- **Yes**: 44%
‘Yes,’ My Organization Monitors Compliance to NON-Functional Requirements (NFRs)

- Agile-ish: 34%
- Agile: 43%
- Iterative: 49%
- Hybrid: 53%
- Waterfall: 59%
Testing is a Bottleneck

Software Quality Processes

- Basic Tasks
- Skilled Tasks
- Process Oriented
- End-to-End Tasks
- Data Analytics
- Fully Automated Provisioning
- Business Views
- Organic Systems

Automation is Incremental

Efficiency
Test as a time-boxed event is the prevailing method

“Are we done testing?”
Impact of Test in the Evolving SDLC

“Does the release candidate have an acceptable level of risk?”
Software Drives Innovation

Software has shifted from process enabler to **business differentiator**

**SDLC Speed** will be the difference between a first mover and a follower

**Switching costs** associated with software are dramatically lower

The **Cost of Quality** associated with software has shifted dramatically
Parasoft Proprietary and Confidential

Software Failures = Headlines

<table>
<thead>
<tr>
<th>Financial</th>
<th>Airlines/Aero</th>
<th>Government</th>
<th>Media</th>
<th>Internet/Tel</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHASE</td>
<td>United</td>
<td>CIA</td>
<td>Netflix</td>
<td>Google</td>
</tr>
<tr>
<td>RBS</td>
<td>American Airlines</td>
<td>Homeland Security</td>
<td>SEGA</td>
<td>Amazon Web Services</td>
</tr>
<tr>
<td>London Stock Exchange</td>
<td>Aer Lingus</td>
<td></td>
<td>SONY</td>
<td>Facebook</td>
</tr>
<tr>
<td>Knight</td>
<td>Boeing</td>
<td></td>
<td>ABC</td>
<td></td>
</tr>
<tr>
<td>DBS</td>
<td>Barclays</td>
<td></td>
<td>eBay</td>
<td></td>
</tr>
<tr>
<td>Citibank</td>
<td>Toyota</td>
<td></td>
<td>Verizon</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NISSAN</td>
<td></td>
<td>AT&amp;T</td>
<td></td>
</tr>
</tbody>
</table>

Automotive       | Technology    | Retail             |

Parasoft Proprietary and Confidential
Software failures make headline news—eroding customer confidence, shareholder value and brand equity.

The day of the announcement companies lost an average of shareholder value:

\[-$2.3 \text{ Bn} \]
\[\text{−3.75\%} \]
Escalating Cost of Failure

Impact on Market Capitalization

y = -0.0032x - 0.0278

-3.12%  -3.37%  -3.75%  -4.06%
Software Failures = Headlines

Market does not forget— news about failure increase and average of 167%
Building a stronger American - Sometimes

48.87  -2.32 (-4.53%)  NASDAQ - As of 03:56pm EDT

-5.70%
-1.9B

iPad glitch grounds American Airlines flights - Apr. 29, 2015
money.cnn.com/2015/04/29/.../american-airlines.../index.html  CNNMoney
9 hours ago - American Airlines pilots use iPads in the cockpit -- and an apparent software glitch is grounding several planes in the United States.
Airlines face risk of worse disruptions from computer glitches

"Airlines are flying computers," said industry analyst Henry Harteveldt.

Increased reliance on technology has enabled (airlines) to become a much more successful and efficient business, and that also creates an exposure.

3.61% - $750M
Let’s Reverse the Game…Spot the Failure

The Bank of New York Mellon Corporation (BK) ★ Watchlist

40.41  +0.45(+1.13%) NYSE - As of 09:49am EDT

-7.4B
-15%
The Cost of Software Failure - Sony

Cumulative Loss = $18B

We must Re-invent Test

- The penalty for faulty software is increasing
  - Brand erosion
  - All industries at risk for total transformation

- Technology is faster and more distributed
  - DevOps
  - Containers
  - Microservices
Next Generation Testing

- Ops Data
- Integration
- Requirements

Continuous Regression Suite

- Exploratory
- Process Intelligence
- Automated Acceptance Testing
Getting from “Automated” to “Continuous”

There is a big leap moving from “Automated” to “Continuous”

- Business expectations clearly defined
  - Business risks identified
  - Per application, team, release
- Defects automatically prioritized versus **business drivers**
  - No defect left behind
  - Mitigate business risks
- **Testing in ‘complete’ test environments continuously**
  - Leveraging simulation – Service Virtualization
  - Protecting the user experience “top-down”
- Feedback loop for defect prevention
  - Pattern definition
  - Defect prevention practices enhanced
Service Virtualization delivers a simulated test environment to help development and testers with earlier, faster and more complete testing
So, What are you Virtualizing?

- Mainframe
- SOA / Web services
- Java
- .NET languages
- XML
- WSDL
- WADL
- UDDI
- WSIL
- SOAP
- PoX (Plain XML)
- REST
- JSON
- BPEL
- Web Applications

- Mobile Interfaces
- AJAX
- JSP
- JavaScript
- HTML
- CSS
- WS-*
- Standards
- MTOM(XOP) / MIME / DIME
- OAuth
- TCP/IP
- webMethods Broker
- webMethods IS
- HL7
- FTP

- HTTP 1.0
- HTTP 1.1
- HTTPS
- JMS
- IBM WebSphere MQ
- Sonic MQ
- RMI
- EJB
- SMTP
- Tibco Rendezvous
- .NET WCF
- ISO 8583
- EDI
- SAP
- custom
Access to Dev / Test Systems

Simple

Complex

Configuration

Test Access

- Cloud App
- Mainframe
- External Database
- ERP
- Web Server
- Message Queue
- External Application
- Internal Database
- ESB
- Internal Application
- 3rd Party Service
- Internal Service
- Cloud App
- Mainframe
- External Database
- ERP
- Web Server
- Message Queue
- External Application
- Internal Database
- ESB
- Internal Application
- 3rd Party Service
- Internal Service
- **Access**
  - Dependent applications difficult
  - Scheduling conflicts
  - High access fees
  - Geo-political boundaries
  - 3rd party or partner applications

- **Configuration**
  - Complex to configure
  - No control
  - Limited variability
  - Consumes test time
Service Virtualization *dramatically reduces the cost* of delivering a dev / test environment.

- Only manage the functionality you need
- Only manage data you need
- Low-Risk “disposable instance” that can be accessed at anytime
- Dramatically reduce test environment setup and management
The Next Generation Software Quality ‘System’

Development

- Development (E1)
- Test (E2)
- Integration (E3)
- Pre-Production (E4 – Eₙ)
- Production

Operations

- Continuous Verification of Requirements
- Continuous Assessment of Risk

Release Candidate

- Environment Simulation (SV)
- Development Tests
- Auto Test Construction – Synthetic
- Test Optimization
- Test Environment Management - Provisioning and Elastic Cloud Scale

APM – Real Time Data

Parasoft Proprietary and Confidential
Transforming the Identification of Risk

From

- Causal Observations
- Defect Documentation
- Only Structured Data
- Filters
- Ad Hoc
- Tool Dependent

To

- Probabilistic
- Most Likely Causes
- Structured and Unstructured
- Business Policies
- Automated
- Tool Independent
Parasoft Virtualize is used throughout the SDLC to speed access to and reduce the costs of managing development and test environments.

- **Mobile Application** development and extension
- **Agile/Parallel development** limited by system dependencies
- **Capacity Constrained** staged environments
- **Limited access** to mainframes, ERPs, or 3rd party systems
- **Test data management** for complex transactions
The Test Environment Challenge

Test environment access is outside the control of development and test leaving gaps in the process

IT Operations

- Staged Assets
- Dependent Applications
- 3rd Party Assets
- Virtual Environments

Scheduling

Configuration

Access Limits

Parallel development delays... Need simple, realistic access to dependent components...

Too much time waiting for access... Need reliable test data

Need a realistic test environment easy to maintain
The Test Environment Challenge

Test environment access is outside the control of development and test leaving gaps in the process.

**IT Operations**

- Staged Assets
- Dependent Applications
- 3rd Party Assets
- Virtual Environments
- Scheduling
- Configuration
- Access Limits

*Create, Manage, Provision*
Value of Service Virtualization

Service Virtualization drives better software quality while reducing CapEx and OpEx

- **Increase Access to “Trustworthy” Test Environments**
  - Provide a complete test environment
  - Provide access 24/7
  - Flexibility provides a platform for extending test scenarios, coverage

- **Reduce Operational Expenditures**
  - Significantly reduce test environment configuration time
  - Simplify test environment access
  - Remove the bottleneck of test environment setup

- **Reduce Capital Expenditures**
  - Software assets can be run anywhere
  - Significantly reduce bandwidth demand
  - Reduce dependency on staged environments
Continuous Testing provides a real-time, objective assessment of the business risks associated with an application under development. Applied uniformly, Continuous Testing allows both business and technical managers to make better trade-off decisions between release scope, time, and quality.

Generally speaking, Continuous Testing is NOT simply more test automation. Rather, it is the reassessment of software quality practices—driven by an organization’s cost of quality and balanced for speed and agility. Ultimately, Continuous Testing can provide a quantitative assessment of risk and produce actionable tasks that will help mitigate these risks before progressing to the next stage of the SDLC.

Read this 44-page eBook to learn how Continuous Testing can help your organization answer the following questions at the time of the critical “go / no-go” decision for a software release candidate:

- Are we done testing?
- Does the release candidate achieve expected quality standards?
- What are the quantifiable risks associated with the release candidate?
- How confident are we that we won’t end up in the news for software failures?

Download the eBook

First Name *

Last Name *

Company *

Job Title *

Work Email *

Phone *

Country *

Mailing State/Province *
### Virtual Lab Management

<table>
<thead>
<tr>
<th><strong>Strengths</strong></th>
<th><strong>Weaknesses</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>“Low” Risk</td>
<td>Must Have Access</td>
</tr>
<tr>
<td>Reduce CAPEX</td>
<td>Permissions to Control</td>
</tr>
<tr>
<td>Reduce Server Sprawl</td>
<td>Configuration Permutations</td>
</tr>
<tr>
<td>Manage Environment Complexity</td>
<td>- Test Data</td>
</tr>
<tr>
<td>Uniformity for Global Teams</td>
<td>- Performance</td>
</tr>
<tr>
<td>Operational Efficiency</td>
<td>Hardware Capacity Constraints</td>
</tr>
</tbody>
</table>
## The Value/ROI Drivers – Human

<table>
<thead>
<tr>
<th></th>
<th>Virtual Lab</th>
<th>Service Virtualization</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Reduce OPEX</strong></td>
<td>Eliminate scheduling delays associated with staged test labs</td>
<td>Eliminates configuration time by delivering more granular ‘service’ versus system</td>
</tr>
<tr>
<td></td>
<td>Eliminate ‘some’ configuration delays</td>
<td>Eliminates scheduling delays</td>
</tr>
<tr>
<td><strong>Understand</strong></td>
<td>Gives a good view of inter-relationship of applications within geo/political control</td>
<td>Gives a complete view of all system dependencies</td>
</tr>
<tr>
<td><strong>Environment</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Complexity</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Operational</strong></td>
<td>Reduces demand on test lab resources</td>
<td>Eliminates delays associated with</td>
</tr>
<tr>
<td><strong>Efficiency</strong></td>
<td>Reduces delays of scheduling, setup and re-configuration</td>
<td>System configuration</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performance configuration</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Setup and tear down</td>
</tr>
</tbody>
</table>
## The Value/ROI Drivers – Physical

<table>
<thead>
<tr>
<th>Virtual Lab</th>
<th>Service Virtualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce CAPEX and server sprawl</td>
<td>Versus physical test lab will significantly reduce CAPEX</td>
</tr>
<tr>
<td>Eliminate physical test labs (Careful!)</td>
<td>Can eliminate physical test lab BUT be careful what you want to virtualize due to compliance and data privacy concerns</td>
</tr>
<tr>
<td></td>
<td>Can reduce incremental CAPEX due to physical test capacity constraints</td>
</tr>
<tr>
<td></td>
<td>Can reduce demand on “virtualized” infrastructure resources</td>
</tr>
<tr>
<td></td>
<td>Eliminates the need for incremental increases in physical test labs or virtual lab capacity</td>
</tr>
<tr>
<td>The Value/ROI Drivers – Project/Process</td>
<td></td>
</tr>
<tr>
<td>----------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>Virtual Lab</strong></td>
<td></td>
</tr>
<tr>
<td>▪ Time to Market (Careful!)</td>
<td></td>
</tr>
<tr>
<td>▪ Increases time to market by providing access to “production-like” environments within geo/political control</td>
<td></td>
</tr>
<tr>
<td>▪ Access anytime or anywhere</td>
<td></td>
</tr>
<tr>
<td>▪ Good access to completed and built applications within control</td>
<td></td>
</tr>
<tr>
<td>▪ Manage and understand Environment</td>
<td></td>
</tr>
<tr>
<td>▪ Good view of applications managed within resource pool</td>
<td></td>
</tr>
<tr>
<td>▪ Defect Reproduction</td>
<td></td>
</tr>
<tr>
<td>▪ Captures great information for applications under control</td>
<td></td>
</tr>
<tr>
<td>▪ Better Testing</td>
<td></td>
</tr>
<tr>
<td>▪ Increased code coverage</td>
<td></td>
</tr>
<tr>
<td>▪ Increased test efficiency</td>
<td></td>
</tr>
<tr>
<td>▪ Increased test coverage</td>
<td></td>
</tr>
<tr>
<td>▪ Increased # and quality of tests</td>
<td></td>
</tr>
<tr>
<td><strong>Service Virtualization</strong></td>
<td></td>
</tr>
<tr>
<td>▪ Increases time to market by providing more complete access</td>
<td></td>
</tr>
<tr>
<td>▪ Provides a net new environment earlier in the process</td>
<td></td>
</tr>
<tr>
<td>▪ Access to both complete and incomplete components</td>
<td></td>
</tr>
<tr>
<td>▪ Much broader view of total system interdependencies</td>
<td></td>
</tr>
<tr>
<td>▪ Understanding of 3rd party impacts</td>
<td></td>
</tr>
<tr>
<td>▪ Extends information by including components outside of control</td>
<td></td>
</tr>
<tr>
<td>▪ Increase test efficiency</td>
<td></td>
</tr>
<tr>
<td>▪ Increased test coverage</td>
<td></td>
</tr>
<tr>
<td>▪ Increased # and quality of tests</td>
<td></td>
</tr>
</tbody>
</table>