Best Practices in Testing

Leveraging Open Source Technologies in Test Automation
Siva Arunachalam
Managing Director – Application Services

Siva Arunachalam heads the Application Development, Maintenance and Testing services Practice at Emtec, Inc. Headquartered in Radnor, PA, Emtec, Inc. provides technology-empowered business solutions to its customers in US and Canada.

Siva has extensive experience managing and executing transformational IT programs, systems integration and Testing Centers of Excellence. Siva has held leadership positions at companies such as Accenture, Cognizant and Syntel over his 28 year career.

Siva holds degrees in Electrical Engineering and Management, and, lives in Farmington Hills, Michigan.
Agenda

Demands on QA

Commercial v/s OSS Tools Debate

Best practices in building Test Automation Framework

Guts of a Framework

Demo

Q & A
Quality Assurance and Testing: Demands and Challenges

- Companies spend over $50B on QA and the demand is growing
- QA organizations are constantly under pressure to test more within shorter timeframes and a reduced budget.

**QA Organization Demands**

- Scope Increasing
  - Cross-browser compatibility – web, mobile-web
  - Regulatory compliance
  - Threat Vulnerability
- Less Time to Test
  - Rapid application development
  - Shorter product release cycles
  - Distributed development
- Increased Consequence of Defects

**QA Organization Challenges**

- Typical QA Effort Planned – 25%-35%
- Whereas QA Effort Expended – 15%-20%
- Typical Automation Coverage – 55%
- Typical Automation Effectiveness – 65%
- 45% of Automation projects stalled
- 55% of Automation projects don’t achieve ROI
- Effective utilization of QA resources – 70%

*QA Organizations need a comprehensive strategy to address these challenges*
WHAT DOES IT ALL MEAN?

The trend in software Quality Assurance is to do more with less, with higher stakes.

Agile development methodologies are driving shorter development cycles, with an emphasis on testing.

Automated testing is the logical solution to meet the challenge.

QA success requires Economic solutions tailored to organization imperatives with a constant focus to improvise and adapt.
Proven ROI exists for Automated Regression Testing

- Automation effort includes the development and maintenance, in addition to execution effort
- Manual test execution effort increases as new features are being added every sprint
- The graph (below) shows ROI calculations over 25 biweekly product development sprints

- Intangible benefits
  - Regression test suite can be used post Production Maintenance releases with full regression capabilities
  - Dev team can use the automated regression suite during regular sprints as part of CI
  - Increased test coverage across all future releases by having automated regression suite

- Initial Investment
- Realizing the benefit
Commercial vs. OSS Toolset?

**Commercial Tools**

- **Pros:**
  - Ease of use
  - Fully integrated suites
  - Integrations exist to SDLC Tools
  - Documentation, Training availability
  - Larger available experienced resources

- **Challenges:**
  - Licenses and infrastructure costs
  - Longer term ROI even with wider adoption
  - Difficulty to switch
  - Limitations in customizing / adapting

**Open Source Tools**

- **Pros:**
  - Lower TCO
  - Tool Customization / Adaptation
  - Community
  - Co-existence

- **Challenges:**
  - Learning Curve
  - Needs integration
  - Talent availability

Increasing number of organizations are adopting a hybrid strategy using best of breed tools to take advantage of the benefits, and, are building frameworks to provide an anchor to implement such strategies.
Best Practices in Building a Test Automation Framework

• Plan for reusability and automated verification
  – Use case based testing – focus on commonly used scenarios
  – Support reusability of scripts – assemble scenarios using atomic scripts
  – Provide Object-based automation (minimize impact of UI layout changes)
  – Incorporate automated verification of results

• Design for extensibility
  – Multi-Browser compatibility testing
  – Data driven and parameterized (same script can run for multiple data conditions without need to modify scripts)
  – Cater for diverse data sources for results verification
  – Support execution against multiple environments (dev, test, stage) through configuration

• Test Result Analytics
  – Compare execution results across multiple runs
  – Reporting by functional components and business scenarios
  – Provide trend analysis for individual features

• Plan for unattended execution
  – Enable Configurable execution for unattended parallel operation

• API testing
  – Somewhat neglected area in test automation
  – Key element in web development where support needs to be provided on various platforms.
  – APIs are mainly used for data transfer and various key operations
  – Ability to verify each parameters and its variation in standalone manner

• Enable extended business scenario testing
  – Ability to use English statements (leveraging Behaviour Driven Design) to create test scenarios, empowering your business users to create dynamic scenarios

• Tool Agnostic
  – Plug and Play
  – Ability to work with different tools with minimal impact on test coverage
Popular Open Source Tools

End User Defined Testing
- Cucumber
- Concolic

UI Record and Play
- Selenium
- TestLink

Test Execution Framework
- Gallio
- NUnit
- Redmine

Project & Defect Management
- Bugzilla
- Testopia

Continuous Integration
- Jenkins
- CruiseControl
Open Source Tools and References

- Cucumber - [http://cukes.info/](http://cukes.info/)
- Gallio - [https://code.google.com/p/mb-unit/](https://code.google.com/p/mb-unit/)
- Jenkins - [https://wiki.jenkins-ci.org/display/JENKINS/Meet+Jenkins](https://wiki.jenkins-ci.org/display/JENKINS/Meet+Jenkins)
- Bugzilla - [http://www.bugzilla.org/about/](http://www.bugzilla.org/about/)
Putting it together – an example Framework

- One-Click Automation
- Plug & Play Architecture
- Software Development Kits
- Minimal Maintenance
- Unified Reporting
- Catch Regressions as they happen
- Multi Protocol Support
- Data Driven Tests
- Configuration Flexibility

Putting it together – an example Framework

Emtec, Inc. Proprietary & Confidential. All rights reserved 2013.
Demo

Charts
Charts allow user to get quick graphical view of selected test runs

Multi Protocol Support
Facilitates API test harness with support for various protocols (SOAP)

Unified Reporting
Empowers unified reporting with customizable report generation

Plug & Play Architecture
Employs plugin architecture for leveraging various open-source

Bugzilla

JIRA & bugzilla
AuFait Architecture

Application Under Test (AUT)

UI

Business Layer

DB

WEB / WINFORMS / HEADLESS

SOAP / REST

DATASOURCES

Test Scenarios

Scenario 1
Business Process Workflow(s)

Scenario 2
Business Process Workflow(s)

Scenario N
Business Process Workflow(s)

AuFait Dashboard

AuFait Core

Application API

API Framework
Application Manager Skeleton
Object Cache

Configuration Framework
Base Classes

Utility Framework

Data Driven Capabilities
Utility Classes
Session Manager

Execution & Reporting Engine

Reporting Framework
Result Trend Analysis
Multi process Execution

AuFait Team

AUT Automation Team

Business Adaptor

<Client>Manager
ResourceManager
UserController

Object Cache

Opportunity
User
Project
Account

Emtec
Business & Technology Empowered
Thank you for your time

Please visit us online at www.emtecing.com

For more information contact:

Chris Barton
chris.barton@emtecing.com
312-216-4956

Or email us at:
qa.practice@emtecing.com