Frank Cohen’s Presentation To
International SOA Conference, Rome, Italy
June 25, 2009

Retooling For The Next Generation SOA

Enterprise managers see the success of social networking Web sites like Facebook and mySpace, the usability of Ajax applications like Google Maps, and the flexibility of mashups that use REST interfaces like Amazon and Flickr combinations, and tell their software developers, I want that too! Implementing all these features requires a new base of technology, including tools, platforms, and technology for implementing Service Oriented Architect (SOA), Business Process Management (BPM), Complex Event Processing (CEP), and Service Virtualization (SV) technology.

Frank Cohen, author of FastSOA, the way to build performance and scalability into SOA and CEO at PushToTest, presents his recently completed research on the issues, workarounds, patterns, and anti-patterns that are common when using software development tools and platforms to architect and build SOA, BPM, and CEP applications. See http://www.pushtotest.com for this slide deck and additional details.
About PushToTest

- Mission: Make Internet Services Run Faster, More Reliably
- Founded by Frank Cohen in 2001
- Functional Test, Load and Performance Test, Service Monitors
- Privately Held
- http://www.pushtotest.com

Next let’s talk about

SOA Vision and Promises

- The gap between vision and delivery
SOA Promises

- Reduced Costs from Component Reuse (Registry)
- Better Able to Achieve SLAs (Governance)
- Architecture for Modern Application Development (Web 2, Enterprise 2)

Today’s Tools Deliver

- Confusing Mix of Features, Functions, and Bugs
- Tools Demand Mix of Backgrounds, Skills, Architectural Experiences
- Tools Do Not Play Well Together
- Huge Variations In Performance (Due to Developer Decisions)
Service Virtualization

- Too much code to deploy and manage
  - Service Virtualization White Paper
  - [http://soakit.pushtotest.com](http://soakit.pushtotest.com)

![Diagram of service virtualization with percentages]

**Service Composition**

- Maps Service Interdependency
- Reduces Maintenance Costs
- Increases Uptime
Is SCA The Magic Bullet?

- Software Component Architecture (SCA)
- Type “SCA Architecture” into Google, Gives Articles from 2006

Next let’s talk about

The Problem

- RIA and SOA Is Everywhere
- Complexity and Lack of Standards Driving Test Costs Up
- Solutions for Test Development
RIA, SOA Is Everywhere

AIR is only one RIA/SOA success story of Dozens More
Hey Frank: Show Appcelerator Example

6.7 Billion People
1.6 Billion Internet Users

0 RIA, SOA Standards

How Do I Test?
Iterative, Continuous, Agile

Design → Code → Unit Test

ALM Cycle = 8 Weeks

Test & Launch

Design → Code → Unit Test → Function Test → Load Test → Monitor

ALM Cycle = 8 Weeks
Integration, Regression, Load
Need Rapid Agile Testing

- Design
- Code
- Unit Test
- Monitor

ALM Cycle = 8 Weeks

But, You Only Have 1 Week To Test

Test & Trash Methodology

- 30% Application Changes
- 30% Tests Re-recorded

First test = $20K, Second Test = $20K, etc.
HP Built a $2 Billion Business On This
Don't Forget The Accidental Participant...

SOA Application Performance

- Test Development/Authoring
  - Protocols, Data, Events
  - Record/Playback
  - Scripting
- Deployment and Orchestration
- Repurposing
- Root Cause Analysis and Mitigation
- Automation, Continuous Build
Applies Everywhere

But Mostly...

Application Testing Models

- Agile Test-First Methodology

Environment

- Browser/Playback, Unit Test, Object-Oriented, or Model-Driven Development

Methodology

- Service-Oriented Architecture, All Tests and Implement SOA-based and ready for use

Architecture Model

- Build Tools and Monitors of Web 2.0, SOA, RIA, Ajax, Flash, Flex
Open Source Test Automation

Do More Testing and Monitoring With Less IT Budget

The Open Source Alternative To Commercial Test Tools

- One Tool for Functional Testing, Load Testing, Monitoring
- Test Across Rich Internet Applications (RIA, Ajax, Flex)
- Proven Test Methodology Optimizes Your Test Tool Use
- Integrates into Your Development/QA Lifecycle
- Your Full-Service Partner For Success

Start right now: How To Test Rich Internet Applications | Test SOAP-based Web Services

1. Integration Testing
2. Functional Testing
3. Performance Monitoring
**Test Script (Selenium)**

Use Case
1. Enter Search Text
2. Click Item 6
3. AssertTextExists

Selenium Script Runner
- HTMLUnit

**TestMaker Transformer: Java Unit Test**

```java
public class CheckEmail {
    public void setUp() {} ...
    public void runTest {
        selenium.setBaseUrl("http://myurl");
        selenium.open("/login.srf");
        selenium.type("search", "ctest");
        selenium.click("//Item[6]");
        selenium.assertTextExists("mytext");
    }
}
```

**Deployment**

Your cloud or mine? Or both!
Clouds Enable New Levels

- Run At User Levels Never Before Possible
- Run From Multiple Locations Around The World
- Pay As You Go

Next let’s talk about

The Accidental Tester

- “I didn’t even know we had Ajax in our data center”
Have You Seen Google Maps?
I WANT THAT!

Hmmmm... I see someone put him back on the network.
Ajax Architecture

Missing Ingredients

- Missing Event Model Standards
  - 70% of my Selenese commands are to synchronize the test to the application

- Missing Data Model Standards
  - String, XML, JSON, Binary, Frankonium

- Missing Protocol Standards
  - Usually HTTP, but now messaging broker technology adds new protocols for component-to-component messaging

Frank: Don’t forget the Selenium Example
Not Limited to Ajax

15% SOAP Standards (WS-*)
   Managed By W3C.org and WS-I.org

85% REST Standards
   Managed By NoOne.org

Amazon.com Web Services by 2004

Observe

Observe Business Flows
Protocols
Data

Web Server

Server-Side Application

JavaScript Program
Open Source Test Automation

Test

Correlate

Performance Hotspots
Broken Functions
Open Source Test Automation

Repurposing

Developer
- Functional Tests, XML, Object Performance Optimization
- Run On Development Machine
- Recorders & Wizards
- Turn Unit Test Into Functional Test Automatically

QA Technician
- Load Tests, Release Tests, Regression Tests
- Support/Maintenance Contract
- Test Runtime
- Turn Existing Functional Test Into Load Test Automatically

IT Manager
- Deployment Confirmation Monitor / Dashboard
- Proof of SLA
- Support/Maintenance Contract
- Test Runtime
- Turn Existing Functional Test Into Service Monitor Automatically

Repurposing

<table>
<thead>
<tr>
<th>Role</th>
<th>Task Description</th>
<th>Investment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developers</td>
<td>Unit Test Coverage, As Many As Possible, Jython=9000</td>
<td>1000</td>
</tr>
<tr>
<td>QA Functional Test</td>
<td>Test Each Step In A Business Process</td>
<td>500</td>
</tr>
<tr>
<td>QA Load, Performance Test</td>
<td>Test Heavyweight Tasks</td>
<td>100</td>
</tr>
<tr>
<td>Ops Production Monitoring</td>
<td>Just Enough To Know The App is Up</td>
<td>10</td>
</tr>
</tbody>
</table>
Next let’s talk about

How To Repurpose

- The PushToTest Repurpose Methodology

Business Flow of a Test
Check-In Unit Test Using DPL

Unit Test

Data Production Library (DPL)  Protocol Handler

CSV  RDBMS  Object

Developer Data
QUnit Load Data
Pre-Production Data
Production Data

Unit Test Using Protocol

Unit Test

Data Production Library (DPL)  Protocol Handler

HTTP  SCAP  Telnet

Extensible for other protocols
Open Source Test Automation

Unit Tests To Test Flows

- Data Production Library (DPL) and Protocol Handlers Enable Repurposing

Check-in → Buy Chips → Buy Dinner

From Unit Tests To Functional Tests

Functional Test

1. Start
2. Setup
3. Run Test
4. Run Test
5. Teardown
6. End
From Unit Tests To Functional Tests

Functional Test:
- Start
- Setup
- Run Test
  - User Interface
- Run Test
  - API
- Run Test
  - Database
- Teardown
- End

Load Test:
- Start
- Thread 1
  - Setup
  - Run Test
    - User Interface
  - Run Test
    - API
  - Run Test
    - Database
  - Teardown
- Thread 2
  - Setup
  - Run Test
    - User Interface
  - Run Test
    - API
  - Run Test
    - Database
  - Teardown
- End

Business Service Monitor:
- Start
- Setup
- Run Test
  - (User Interface)
- Run Test
  - (API)
- Run Test
  - (Database)
- Teardown
- End
Next let’s talk about

Results Oriented Approach

- Surface Performance Issues Quickly
- Automated Functional Tests
- SLA Compliance Monitoring

Surface Issues Quickly

- Root Cause Analysis & Mitigation
- Product Search Latency Becomes The Problem Over Time
- Due To Network Saturation When Running More Than 500 Simulated Users
Automated Functional Tests

- Avoid Manual Testing
- Functional Test Operates a Test Flow
  - Multiple Steps In Flow
- Build, Test, Deploy, Monitor in One Script

SLA Compliance Monitoring

- Reuse Tests To Monitor Compliance with SLAs
- Provide Immediate Dashboard To Surface Production Performance Issues
Next let’s talk about

What Holds Us Back

- Take The Next Step: Open Source Test Methodology
Alex: load test is running fine
Inga: Coolio!
Alex: Uh oh
Alex: Success: 3210, Failure: 15
Alex: Success: 3210, Failure: 204
Alex: Success: 3210, Failure: 483
Inga: What’s up?
Alex: The whole application failed
Inga: Yep, avail threads = 0
Alex: Back to the drawing board

Fear Not
We get paid to break things.
When you break it you understand it.
When you understand it you can fix it.
Where To Go From Here

- Attend a PushToTest Training (soapUI, Selenium, TestMaker)
  - http://soapuitraining.pushtotest.com

- Attend An Open Source Test Workshop
  - http://workshop.pushtotest.com

- Ask Peter Schumacher, 512 468-6679 (USA, Texas),
peter@pushtotest.com, For A Proposal on Licenses, Training,
Support, Consulting