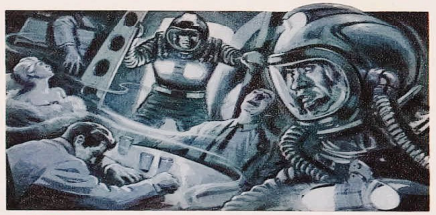
Lessons Learned in Test Automation

Peter Chau, David Cranstone

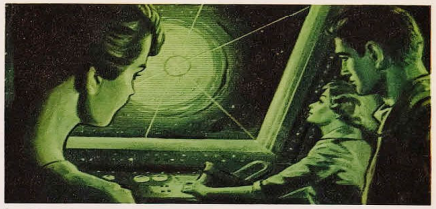
SQDG - January 19th 2010

SEE: LIFE ABOARD THE MIGHTY ROCKET-CITY AS IT HURTLES THROUGH SPACE!



SEE: THE HORRIBLE SECRET OF THE ROCKET SHIP LOST 9 MILLION YEARS!

SEE: THE MYSTERIOUS ATTACK FROM THE BLACK STAR!



SEE: THE UNBELIEVABLE SECRET AT THE END OF THE UNIVERSE!

STARRING DENNIS STEPHENS · FRANCIS SMOLEN · AN AMERICAN INTERNATIONAL PICTURE

64/287

Introduction

- Our experiences with functional test automation
- High level presentation, not a tool demo
- Focus on open source test automation

Agenda

- Why Automate?
- Four Phases of Test Automation
- Lessons Learned
- Questions

Why Automate?

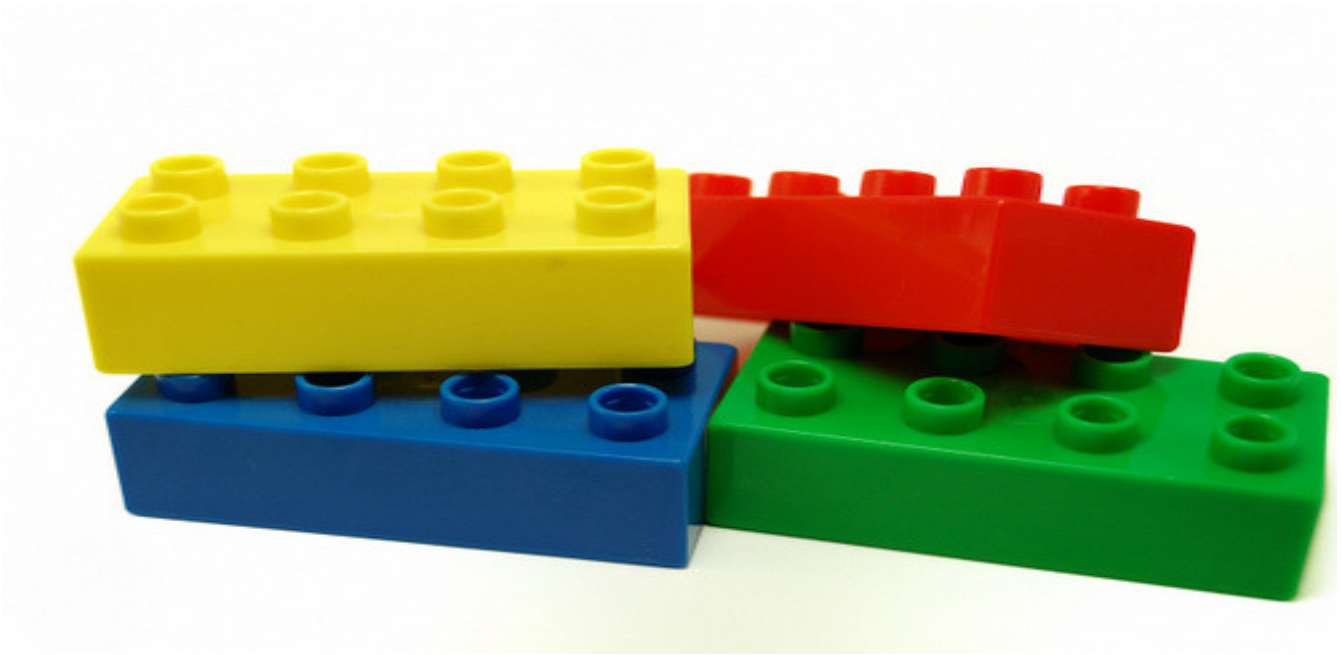
[Credit: carrie227 – flickr.com]

What is Automation?

- Not just about writing/recording scripts
- Creating tools to assist testing
- Using tools (JMeter, SoapUI, Fiddler)
- Distinctive from unit testing

Benefits of Test Automation

- More consistent testing
- Faster than manual testing
- Quick Feedback
- Add scripts to build
- Deliver set of tests to sustainment
- Required for load testing



Phases of Test Automation

Credit: carrie227 – flickr.com

Phases of Automation

1. Big Bang
2. Recovery
3. Ad-hoc
4. Consolidation

Big Bang

- Quick Test Pro to automate legacy system
- 3rd party consultants to write scripts
- Sent QA staff on training course

Big Bang – Results

- Poor return on investment
  - Expensive to license, write scripts
  - Produce few tangible benefits
- Scripts difficult to maintain and execute
- No single tool that will solve all problems
  - Web, Windows GUI, Web Service, Terminal
- Set back the cause of test automation

- Approx. Level of Automation - 5%

Recovery Phase

- Little automation performed
- Ad-hoc scripts
  - e.g. Shell test scripts
- Custom utilities
  - Network, Payroll Checker
- Excel
- Diff Tools

- Approx. Level of Automation - 5%

Ad-hoc Phase - Strategy

- Investigation of tool sets
- Adoption of open source tools
- Multiple tools selected depending on technology
- Recruitment of technical QAs
- Patchy usage of automation
- Led by automation enthusiasts

Ad-hoc Phase - Results

- Gained experience in tool usage
- Built experience in automation strategies
- Started to see benefits of test automation
- Multiple tools and languages being used:
  - Selenium/Ruby
  - White/C#
  - Groovy/Java
  - SOAPUI
- Approx. Level of Automation - 10%

Consolidation - Strategy

- Set of tools that cover web page, WPF and web service testing
- Common underlying scripting/programming language
- Open Source

Consolidation – Selected Tools

Scripting

- Selenium/C# for web testing
- White/C# for WPF testing
- NBehave/C# for web service testing

Non-Scripting

- JMeter for load testing
- Fiddler for back end testing
- SOAPUI for web service testing

Consolidation – Advantages

- Using C# scripting language
  - NUnit to drive tests
  - Code shared between projects
  - QA resources more flexible
  - Better developer support
  - Training
  - Recruitment
  - Simplify handoff to sustainment group
- Approx. Level of Automation - 40%

Lessons Learned

Credit: carrie227 – flickr.com

Selling Automation

- Management, QA and other disciplines
- Show results
  - Can be started quickly
  - Start by automating something regarded as high value
  - Showcases, automation fair

Skill Set

- Mix of QA and development skill sets
- Technical skill set
  - Filtering for the right experience
- Training strategies
  - Does not have to cost money
    - Lunch and learns, courses, collaboration
  - Focused continual training

Team Structure

1. Test Automation as a service
2. Embed specialist within a team
   - Create / execute and maintain tests
   - Assist with creating tests and tools
   - Who they report to
3. Automation as a team responsibility

Information Technology

SHAW

PAROS

ACCOUNTABLE BALANCE CUSTOMER FOCUSED INTEGRITY LOYALTY POSITIVE, CAN DO ATTITUDE TEAM PLAYER

Tools

- Selection
  - In house experience
  - Recommendations
  - Trial a tool
- Open source vs. commercial vs. in house
- Lots of innovation in the UI automation space

Mocks

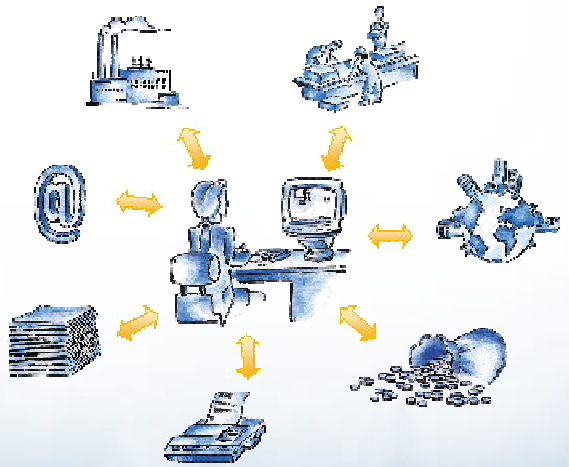
- Simulate components for more efficient testing
- Mocking
  - 3rd party dependencies
  - Hardware
  - Integration components

Frameworks

- Example: keyword driven to accelerate automation
- Minimal upfront, evolve over time
  - Comparison: BA having every single requirement upfront
- “Guideline” instead of framework

Automation Management

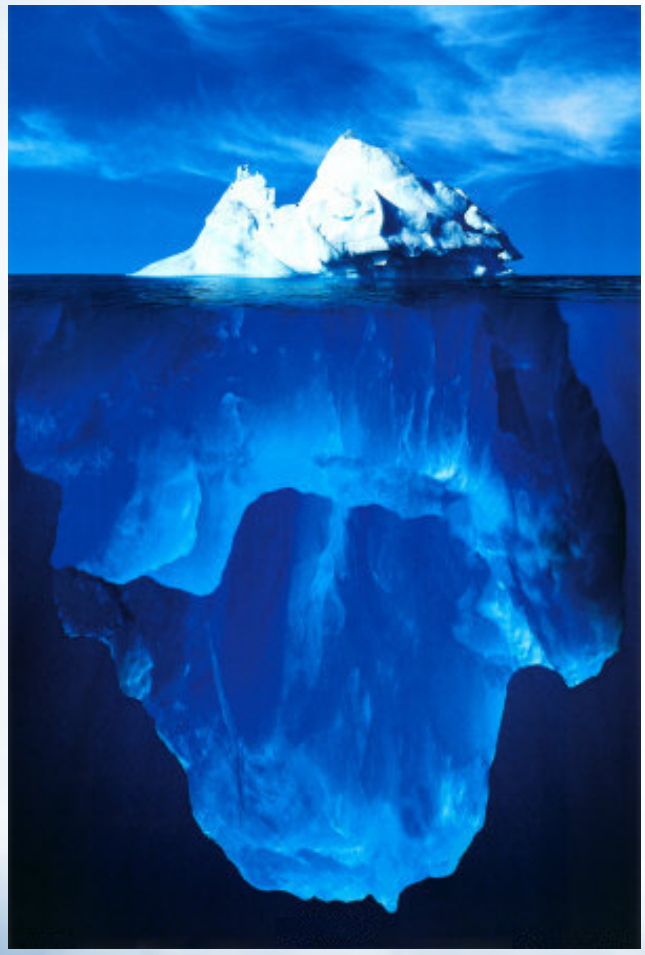
- Large amounts of test scripts, data, results
- Source and version control
- Continuous integration / test scheduling
- Test case, data and result integration tool



Finding Bugs With Automation

- Change detector
- Not meant to replace manual testing
  - Allows testers to test smarter
  - Safety net
  - Level of Confidence *
- Exploratory testing
- Better understanding of system

Hidden Benefits



- Increase collaboration between development team, production support
- Blurring the lines
- Employee engagement

Things to Keep in Mind

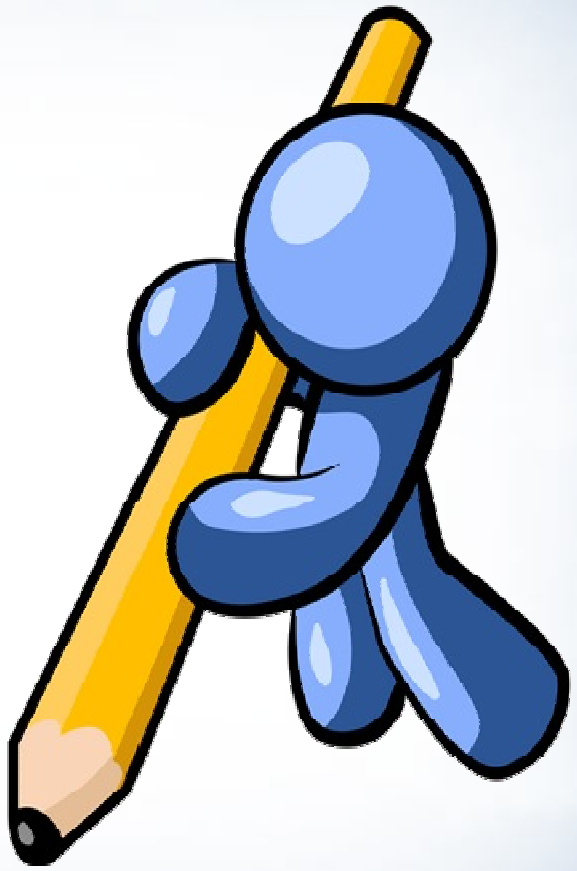
- Maintenance is a very big component to successful automation
- Automation consume a lot of data
- Tests can run in more than one environment
- It's a development project

Building an Automation Culture

- No automation to a critical component in the QA/QC process
- Sustainable by the community
  - Support and buy in
  - Maintain and developed by the team
  - Collective ownership

How to Get Started

- Automation is not easy
- Start small
- Structure is not important at the beginning
- Just do it!
  - Don't wait for a tool to be approved, or a framework to be created.




Questions…
Credit: Max F Williams – flickr.com