A practical guide to a cross-platform Java GUI test
EclipseCon Europe 2013

by Reginald Stadlbauer
About me

- Name: Reginald Stadlbauer
- Company: froglogic GmbH, vendor of Squish GUI Tester and Squish Coco
- Position: co-founder and CEO
- Worked as Software Engineer at Trolltech and the KDE project
About froglogic

- HQ: Hamburg
- Founded: 2003
- US presence since 2008
- Product focus on Squish
  - Squish GUI Tester (Cross-Platform GUI Test Automation)
  - Squish Coco (C, C++ and C# Code Coverage)
- More than 3,000 customers world-wide
Overview

- Types of Testing
- Why Automate?
- Practical demo “A Java GUI test in 10 minutes”
Types of Testing

- Unit Testing
- Performance Testing
- ...
- Functional GUI Testing
  - Black/Gray Box Testing
  - Assume user's point of view
  - Automate to spot regressions
  - Combinable with profiling, coverage and other analysis and monitoring tools
Why Automate?

- Faster
  - Get results quicker
  - Run more tests in the same time
- Trivial to replay in different configurations
- Reliable, reproducible and repeatable
- Relieve testers from monotonous tasks
But...

- Automating GUI tests is not trivial
- Following best practices is vital for the success of automated GUI tests
Automating a Java GUI Test

- Defining manual test case
- Recording the test scenario
- Refactoring and restructuring recording
- Implementing test case based on definition
  - **Scripted implementation**
  - Keyword driven implementation
- Cross-platform test run
- Integrate into automation framework (CI / ALM)
1. Defining manual test case

- AUT: 2-Plan Project Management (Eclipse/RCP GUI)
- Test Case: Assign and rename user role

- Open 2-Plan
- Create new Project “Test Project”
- Expand “Teams->Management Team” in Project tree
- Open person “Hoffmann Simon”
- Assign role “Project Manager”
- **Verify role “Project Manager” shows up in Roles tree**
- Expand “Roles” in Project tree
- Open role “Project Manager”
- Change name to “Project Lead”
- Open person “Hoffmann Simon”
- **Verify role changed to “Project Lead” in Roles tree**
- Delete project
- Exit 2-Plan
2. Record Test & Refactor

- Record test
- Inspect generated script and object names
- Structure and refactor
- Redo test using function calls
3. Check Cross-Platform Compatibility

- Run test on a different platform and verify results
4. Setup Automation

- Check into Revision Control
- Integrate into automation, such as CI, ALM or a simple scheduled batch job
- Make results available to all
About Squish GUI Tester

- Cross-Platform / Cross-GUI-Technology Test Automation
  - Windows, Linux, Mac OS X, Unix, RTOSes, Mobile
  - Qt, Java (Swing/AWT, SWT/RCP, JavaFx), Web, MFC, WinForms, WPF, iOS, Cocoa, Carbon, Android, Tk, Flex, …
- Object-based GUI object identification
- Record & replay and powerful scripting (JavaScript, Python, Ruby, Tcl, Perl)
- Eclipse-based IDE
- Batch-testing via command-line tools
- Remote/distributed testing architecture
- Integrations: Microsoft ALM, HP QC/ALM, Rational RQM, Seapine TCM, SpiraTest, MKS, XStudio, Jenkins, Hudson, JUnit, Maven, …

Free and supported trial at http://www.froglogic.com/evaluate