

Automated JAVA GUI Testing

Challenges and Experiences

Java Forum Stuttgart

2008

About me

- **Reginald Stadlbauer <reggie@froglogic.com>**
- **Co-founder and CEO of froglogic GmbH, Hamburg, Germany**
- **Former Senior Software Architect at Trolltech ASA**

Automatically Testing Graphical JAVA User Interfaces

froglogic



- **Founded 2003**
- **Privately held**
- **No Investments**
- **Profitable with steady growth**
- **Small company, large clients**
- **Specialized in Automated Testing (Squish)**



Agenda

- **Overview**
- **Motivation**
- **Challenges**
- **Solutions**
- **Open Discussion**

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 tools**

Agenda

- Overview
- **Motivation**
- Challenges
- Solutions
- Practical Demonstration
- Q & A

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**

What *Not* to Expect from Automation

- **Don't expect**
- **To get rid of all your testers**
- **A tool *writing* tests automatically for you**
- **100% automation**

How to Automate?

- **Choose a framework**
- **Make it easy to write tests**
- **Go slow**
- **Commit time and resources**
- **Track progress**
- **Process results, track defects**
- **Award tests**

Agenda

- **Overview**
- **Motivation**
- **Challenges**
- **Solutions**
- **Practical Demonstration**
- **Q & A**

Challenges of Automated GUI Tests

- **Variety of toolkits and controls**
 - **Need toolkit specialization**
- **Evolution of application and toolkit**
 - **Need abstraction layer**
- **Event driven, asynchronous program flow**
 - **Need synchronization and exception handling**
- **Dynamic layout**
 - **Need robust object identification/interaction**

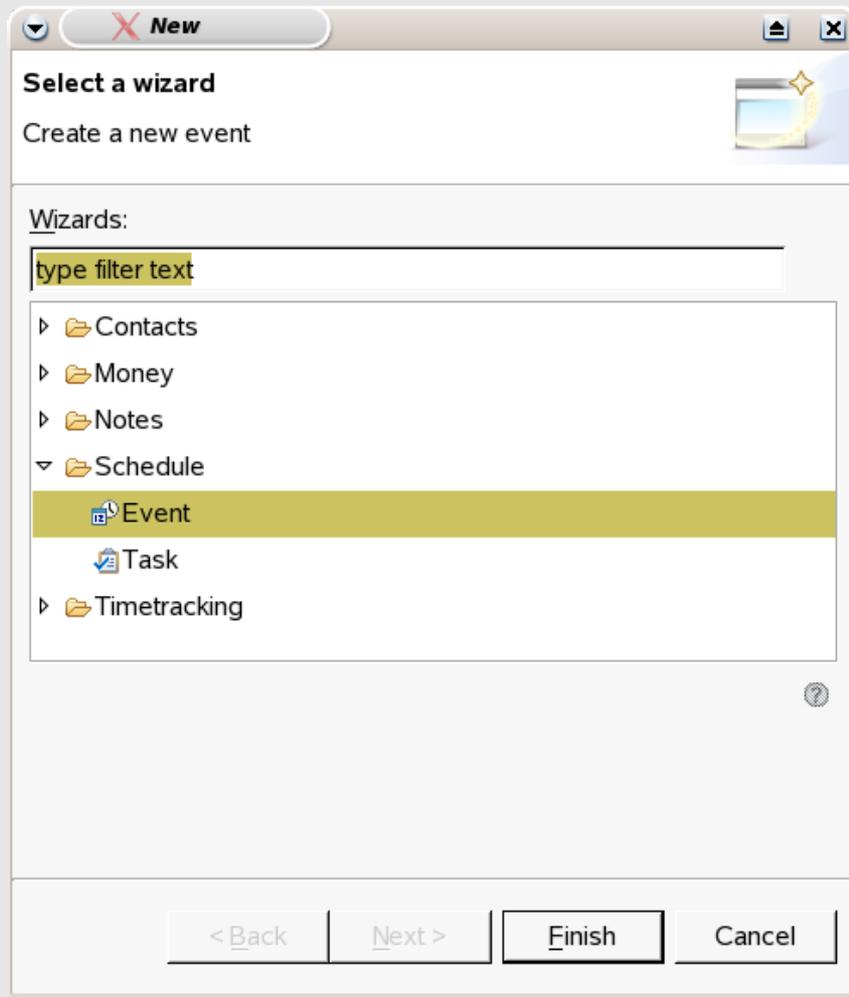
Agenda

- Overview
- Motivation
- Challenges
- **Solutions**
- Practical Demonstration
- Q & A

Solutions - Robustness

- **Built-in knowledge of JAVA GUI toolkits**
- **Object hierarchy, properties, etc.**
- **Identification of controls vs. low-level coordinates**
- **Proper synchronization points**
- **Extensible for custom widgets**

High-Level Widget Recognition



VERY BAD:

```
mouseClick(132, 367)
```

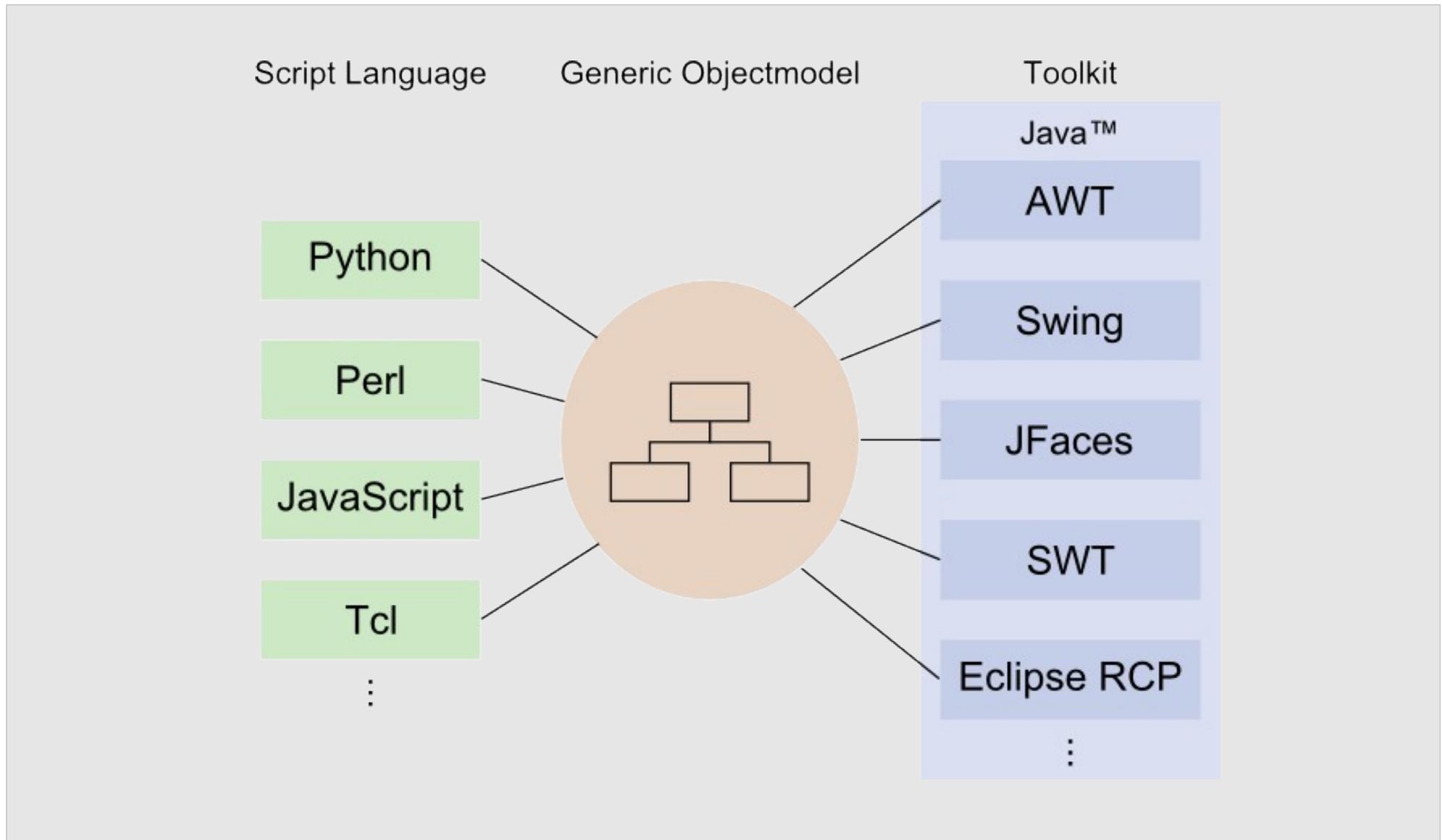
BAD:

```
mouseClick('Tree', 30, 266)
```

GOOD:

```
clickItem('Tree', 'Event')
```

Toolkit Abstraction

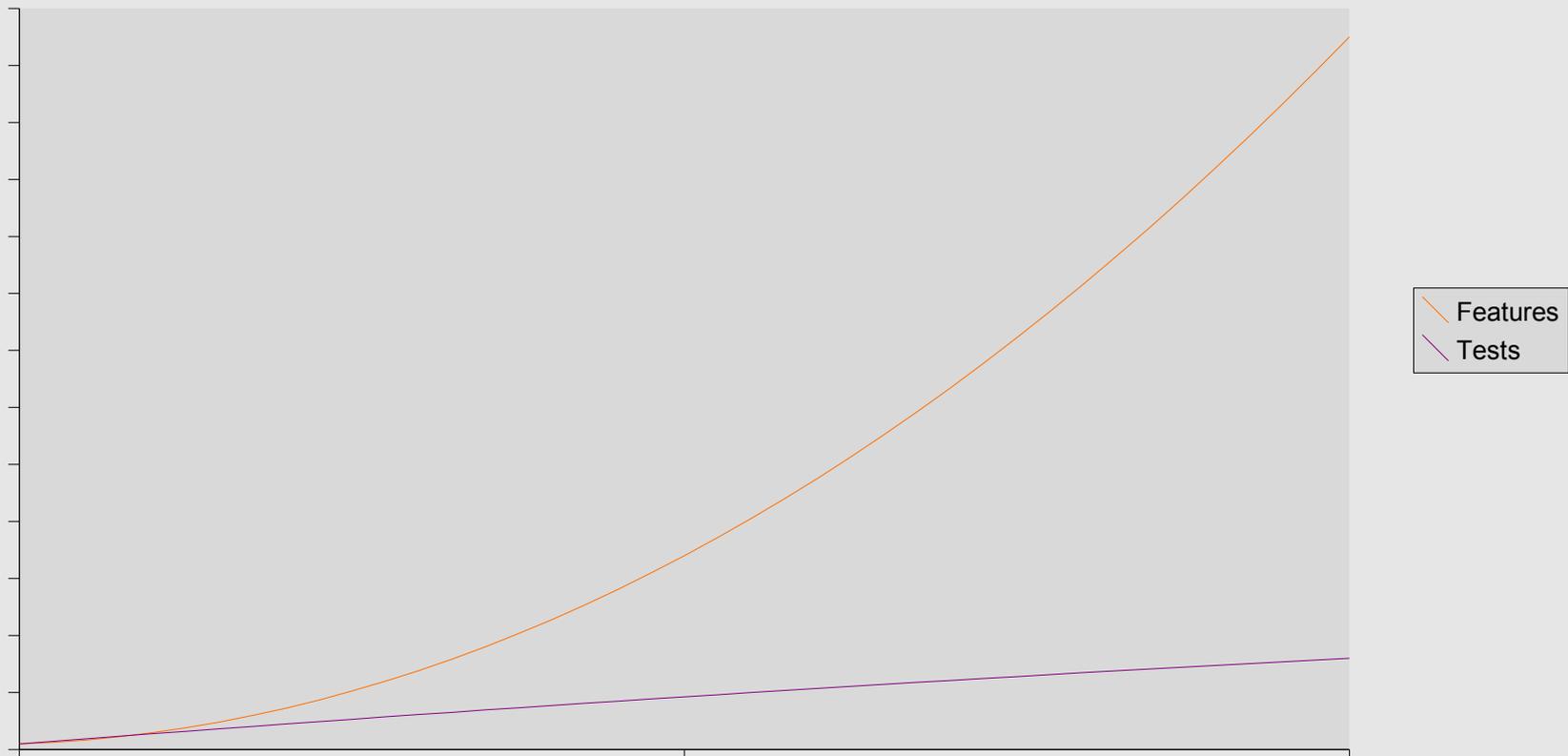


Solutions – Test Design

- **Do not just record**
- **Re-factor generated script code**
- **Create small, reusable test functions**
- **Introduce abstractions layers**
- **Apply same methods as for software development**

Benefits

Benefit of accumulation over time



- **Dedicated support for JAVA toolkits**
 - **Swing/AWT/JFaces and SWT/Eclipse Rich Client Platform (RCP)**
 - **API and extensions for supporting custom widgets**
- **Cross-platform (Windows, Linux/Unix, Mac OS X)**
- **Powerful scripting support**
- **Open for integration into existing environments**
- **Additional support for other GUI technologies**
 - **Web/AJAX/DHTML, C++ Qt, Mac OS X Carbon/Cocoa, etc.**

Squish[®] Features

- **Choice of test scripting languages (JavaScript, Perl, Python, Tcl, TSL)**
- **Record or manual code (usually a combination of both)**
- **Record high-level actions instead of generic low-level, coordinate based events**
- **Test debugging and verifications**
- **Modularization**
- **Squish IDE and command line tools**

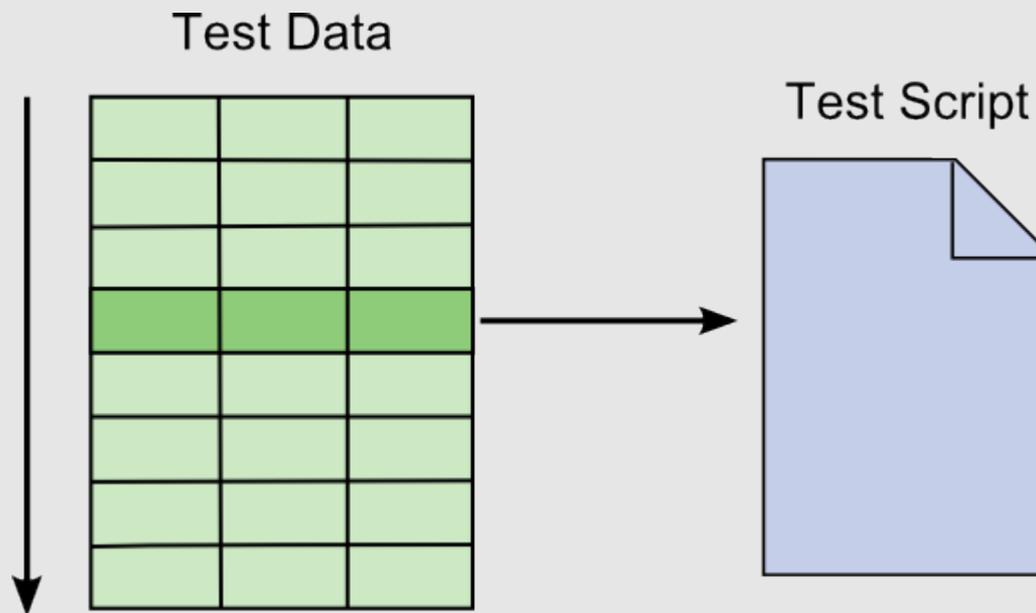
Agenda

- Overview
- Motivation
- Challenges
- Solutions
- Practical Demonstration
- Q & A

Practical Demonstration

- **Recording a test (high-level actions)**
- **Inserting verifications**
- **Ensuring synchronization**
- **Re-factor common actions into functions**
- **Separate test logic and data (data-driven testing)**
- **Run in different configuration**
- **Unattended execution (automation)**

Data-Driven Testing



Data-Driven Testing

- **Benefits:**
- **Clean separation between test data and test logic**
- **Easily extensible tests**
- **Frees domain expert from programming tasks**
- **Helps with future portability**

Unattended Testing

- **Using command line tools**
- **Control from one machine running tests on different systems**
- **Use Unix cron jobs, Windows Services, Eclipse TPTP, HP Quality Center, CruiseControl, Ant, etc.**
- **Post-process results (XML, HTML, XLS, etc.)**

Agenda

- **Overview**
- **Motivation**
- **Challenges**
- **Solutions**
- **Practical Demonstration**
- **Q & A**

More about Squish at www.froglogic.com

Get an evaluation at www.froglogic.com/evaluate

or visit our booth!

Thank you for your attention!