

# **GUI Test Automation from A – Z**

**Reginald Stadlbauer  
froglogic GmbH**

**Java Forum Stuttgart 2010**

# Introduction



Reginald Stadlbauer  
CEO / co-founder  
froglogic GmbH

Specialized in GUI Testing and Automation  
Makers of Squish GUI Testing Tool

# Agenda

- Introduction to GUI Testing
- Preparations (Specs)
- Manual vs. Automated GUI Testing
- Choosing the right GUI Test Tool
- Implementation of GUI Tests
- Automation of GUI Test Runs
- Managing the Flood of Results
- Demo, Wrap-up and Q & A

# Intro. in GUI Testing

- Automating what a user would do
- Interacting with the application
- Verifying the application's state
  - Functional GUI Testing
- Automation can be combined with
  - Coverage tests
  - Load / Stress tests
  - ...

# Preparations (Specs)

- If requirements docs exist
  - Define test specs to verify reqs. met
- Otherwise
  - Specify use cases
- Can be done informally or formal (test management)
- Start on low-hanging fruits / most used functionality

# Manual vs. Automated (1)

- Manual GUI Testing
  - Initial barrier is low
  - Boring and monotonous
  - Error prone
  - Execution time limited (humans are slow)
  - Human resources are expensive and don't “scale”

# Manual vs. Automated (2)

- Automated GUI Testing
  - Initial barrier is higher
  - Faster (more tests in less time)
  - Easy to run in different configurations
  - Reliable and reproducible
  - Interesting & fun

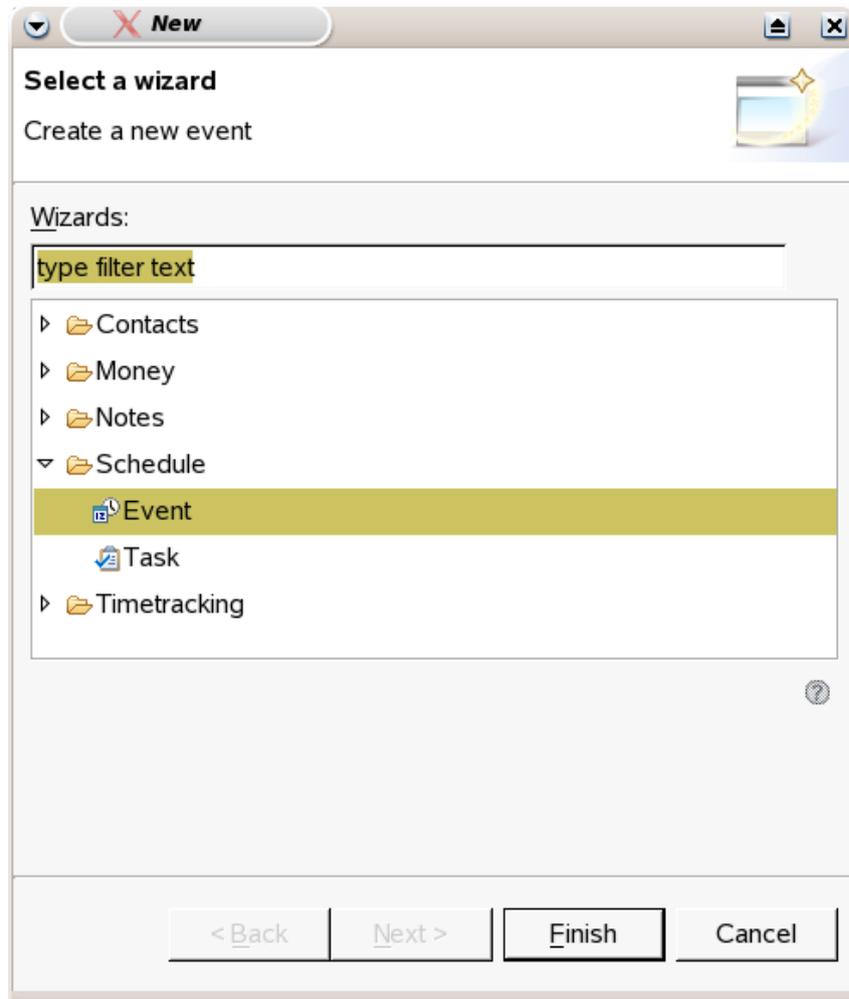
# Manual vs. Automated (3)

- Usually not a either/or decision
- Start with test specs
- Execute test cases manually
- Start automating test cases
  - It is an iterative process

# Choose GUI Test Tool (1)

- What GUI technology is used?
  - Dedicated toolkit support
  - Access to non-GUI layer
  - Scripting support
  - Platform support
  - Extensibility
- Support service & updates
  - Do I get help when needed?
  - Updates for new platform versions

# Choose GUI Test Tool (2)



**VERY BAD:**  
`mouseClick(132, 367)`

**BAD:**  
`mouseClick('Tree', 30, 266)`

**GOOD:**  
`clickItem('Tree', 'Event')`

# Implementing GUI Tests

- Need dedicated resources
- Not just a side task
- Spend time on modules/framework
- Actively maintain and extend existing tests
- Start automating “low hanging fruits”

# Automation of Runs

- Start automating early on
- Start simple and refine over time
- Managing of tests, runs and results
  - (Re)use test management system
  - Make adding tests simple
  - Make results easily accessible & visible
  - Push results on everybody
  - Make it a team effort
  - At least daily runs or even CI

# Managing results

- Collect as much details (in DB)
- Present summary to get quick overview
- Allow drilling into details
- Traceability (line numbers, screenshots)
- Record history to judge progress
- Assign responsibilities to get things fixed
  - Use a test management solution!

# Demo, Wrap-up, QA

Thanks for your attention!

Please visit us at our booth!

Get your free and supported evaluation

<http://www.froglogic.com/evaluate>