

Muss die Automatisierung von Java GUI Tests fehlschlagen?

Reginald Stadlbauer
froglogic GmbH

Über Mich

Name:	<i>Reginald Stadlbauer</i>
Firma:	<i>froglogic GmbH</i>
Position:	<i>Gründer & Geschäftsführer</i>

Über froglogic

Wo:	<i>Hamburg</i>
Gegründet:	<i>2003</i>
Mitarbeiter:	<i>20</i>
Produkt:	<i>Squish (Cross-Platform GUI Test-Tool)</i>
Kunden:	<i>> 1.300</i>

REUTERS 

SIEMENS

 XILINX®



 intel Leap ahead™

 hp invent



 AMD
Smarter Choice



 EADS

 AIRBUS
AN EADS JOINT COMPANY WITH SAS SYSTEMS

 BOEING



 SCANIA

Motivation

Warum wollen wir GUIs automatisch testen?

Falle 1:

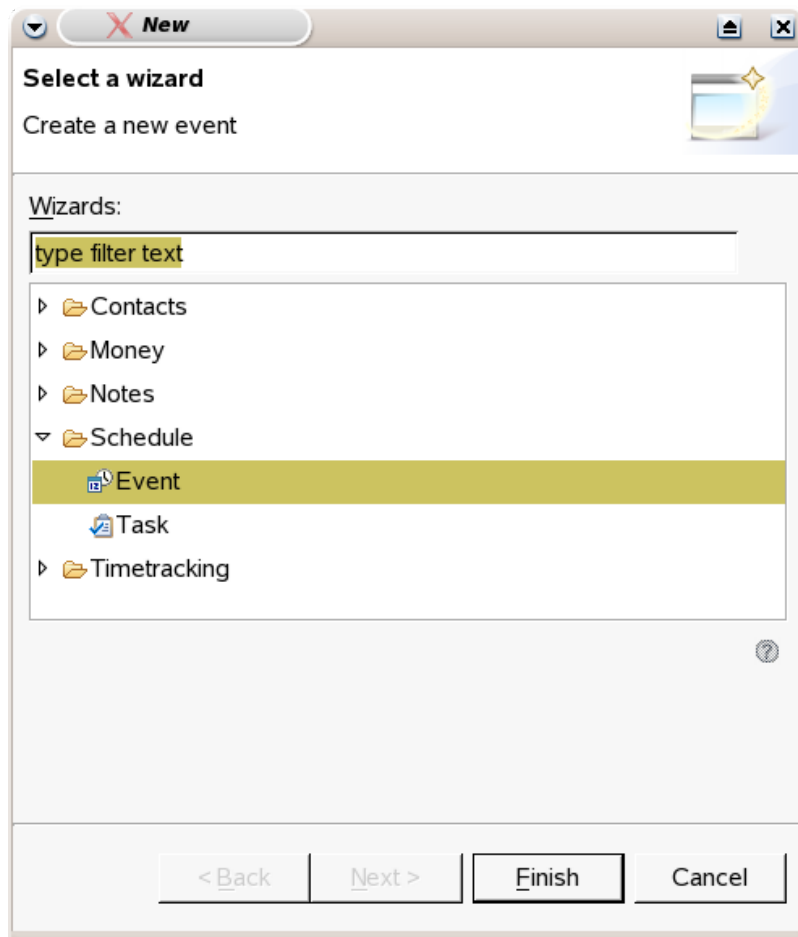
Reines Capture & Replay



Falle 2:

Verlass auf Koordinaten

Falle 2 (Fortsetzung)...



Sehr gefährlich:

`MouseClicked(132, 367)`

Gefährlich:

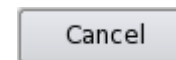
`MouseClicked('Tree', 30, 266)`

Gut:

`ClickItem('Tree', 'Event')`

Falle 3:

Verlass auf Bildschirmvergleiche



Falle 4:

Timing

Falle 5:

Objektidentifizierung

Falle 6:

Instabiler Ausgangszustand

Falle 7:

Beschränkte Makrosprache

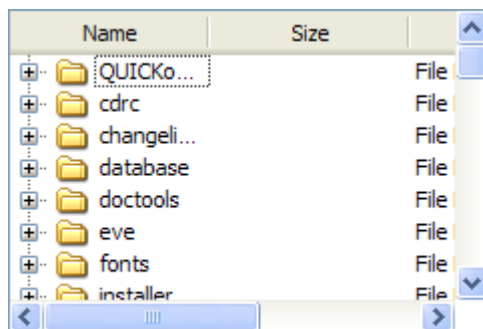
Start	AnwendungA
Eingabe	BenutzerName
Eingabe	Passwort
Click	OK

Falle 8:

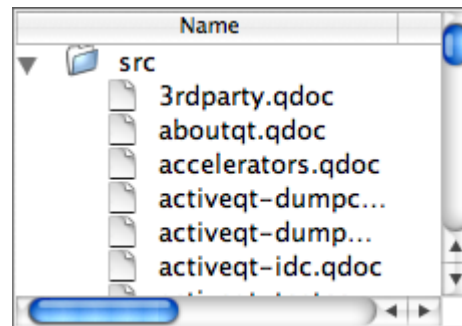
Oberflächlichkeit

Oberflächlichkeit (Fortsetzung): GUI Styles

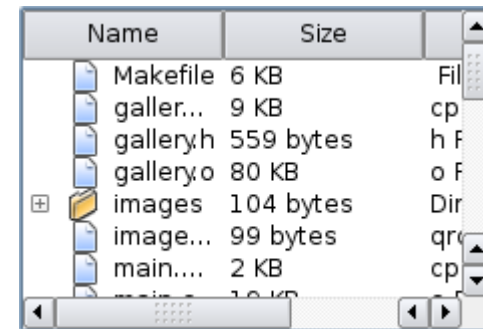
Drei Arten einer Baumdarstellung:



Windows



Mac OS X



Linux

Falle 9:

Nebenjob

Falle 10:

Pseudo-Automatisierung

Richtlinie 1:

Tool-Evaluierung

Richtlinie 2:

GUI Testing $\approx$ Software Entwicklung

Abstraktionen schaffen

Richtlinie 3:

Verantwortlichkeiten festlegen

Wer schreibt und wartet Tests?

Was passiert wenn Fails auftreten?

Richtlinie 4:

Frühe, volle Automatisierung

Test Resultate sichtbar für alle

.... Start Simple!

Richtlinie 5:

Konzentration auf “Low hanging fruits”

Nicht alles ist automatisierbar!

Zum Schluß

Live Demo

Danke...

www.froglogic.com