



Abbot

A Better Robot

Von:

Ronny Schulze

E-Mail: rschulze@informatik.tu-cottbus.de

Übersicht

- ” Was ist Abbot?
- ” Umgang mit Abbot
- ” Erstellen von Tests
- ” Skripte
- ” Anmerkungen
- ” Quellen

Was ist Abbot?

- ” Abbot Framework ist Java-Bibliothek für GUI-Unit-Tests und GUI-Funktionstests
- ” Stellt Methoden zur Erzeugung von User-Level-Aktionen (ähnlich wie *java.awt.robot*) und zur Überprüfung des Zustandes von GUI-Komponenten zur Verfügung
- ” für GUI-Units AWT und Swing

Umgang mit Abbot

- ” Open Source Software
- ” Common Public License (CPL)
- ” Aktuelle Version: 0.12.2 (040318) vom 18.3.2004
- ” Release 0.1 vom 22.3.2002
- ” Abbot-Code-Entwicklung auf SourceForge

Umgang mit Abbot

- ” Download von **abbot.sourceforge.net** als zip- oder tgz-Archiv
- ” Archiv enthält:
 - Jar-Archiv abbot.jar
 - Batch-Datei zum Ausführen auf Windows-Systemen
 - Quelltexte (src.jar)
 - Dokumentation (JavaDoc)
 - Beispiele (Tutorials auf Homepage)

Umgang mit Abbot

” Installation und Start

- Archiv entpacken
- Start mit abbot.bat (unter Windows)
- oder per Kommandozeile (vorher CLASSPATH festlegen):

```
java -jar lib/abbot.jar
```

bzw.

```
java -cp lib/abbot.jar abbot.editor.ScriptEditor
```

(bei einigen 1.3-JDKs unter Unix)

- Start des Skripteditors (programmiert in Swing)

Erstellen von Tests

” 2 Möglichkeiten

- direkt im Code eingebunden (Test-First-Entwicklung) innerhalb eines JUnit-Testfalles
- über das Erstellen von Skripten mittels des Skripteditors

Skripte

- ” günstige Art, eine Testsuite zu organisieren
- ” als XML gespeichert
- ” hauptsächlich zusammengesetzt aus
 - Aktionen
 - Prüfungen (Assertions)
 - Referenzen auf GUI-Komponenten

Skripte - Aktionen

” Aktionen sind:

- ♦ Dinge, welche normalerweise der Benutzer durchführt, wie
 - Buttons anklicken
 - Menüeinträge auswählen
 - Text eingeben
 - ...
- ♦ erfordern ein oder mehrere Argumente, von denen eins die Zielkomponente ist

Skripte - Referenzen

- ” Referenzen auf GUI-Komponenten (component references):
 - ♦ Repräsentieren Instanzen von GUI-Komponenten innerhalb des zu testenden Codes
 - ♦ jede mit eigener ID
 - ♦ Bsp:
 - OK-Button in Dialog mit „OK Button“ als ID
 - String „OK Button“ kann als Argument für Aktionen anstatt der aktuellen Button-Instanz verwendet werden

Skripte - Assertions

- ” Assertions (Prüfungen):
 - ◊ erlauben das Prüfen des Zustandes der GUI
 - ◊ man kann den Wert von Komponenteneigenschaften überprüfen
 - ◊ oder Ausdrücke auswerten
 - ◊ wenn Prüfung fehlschlägt, stoppt die Ausführung des Skriptes und ein Fehler wird angezeigt
 - ◊ können auch als „wait“ benutzt werden
- ” für mehr Details siehe auch *abbot.script* Dokumentation

Skripte

” allg. Skriptrahmen:

1. **Starte ...**
 - *main*-Methode (nicht zwingend, aber *public static* Methode muss es sein)
 - Name der Zielklasse
 - Klassenpfad
2. **Warte auf ...**
 - *Wait for Component/Frame Showing*
3. **Beenden**

Skripte

- ” Was kann man in ein Skript einfügen?
- Sequenzen (Benutzeraktionen können mit Hilfe eines Recorders aufgenommen werden)
 - Prüfungen (Assertions)
 - Methodenaufrufe
 - Kommentare
 - andere Skripte
 - Samples und Annotations

Anmerkungen

- ” begrenzte Aktionsvielfalt für AWT, dafür sehr viel für Swing-Applikationen
- ” auch Test von Applets möglich
- ” für mehr Details siehe sehr ausführliche *JavaDoc API* mit Beschreibung aller Pakete und Methoden des Abbot-Frameworks und des Skripteditors



Quellen



” abbot.sourceforge.net