

# Release-QA bei OpenOffice.org

Automatisiertes und manuelles Testen

OpenOffice.org Community Treffen  
05./06.02.2005  
Linux Hotel Essen

- Automatisierte Tests
  - Voraussetzungen - Tools beschaffen und installieren
  - Tests durchführen
  - Ergebnisse bewerten
- Manuelle Tests
  - Webportal TCM
  - Tests online / offline durchführen
  - Testfälle online / offline bearbeiten und ergänzen
- Was benötigen wir noch?

# Warum Testen? - Verantwortlichkeiten

- Prüfen der Codequalität
  - Entwickler, Projektleiter, QA
- Stabilität Buildprozess
  - Entwickler, Release Engineering
- Prüfen Einhaltung der API-Spezifikationen
  - Entwickler, Projektleiter, QA (qadevOOo)
- Prüfen des Applikationsverhaltens gegenüber Spezifikation
  - QA (qatesttool, manuelle Tests), Entwickler
- Prüfen und Freigabe der deutschsprachigen Version
  - **WIR** (qatesttool, manuelle Tests)

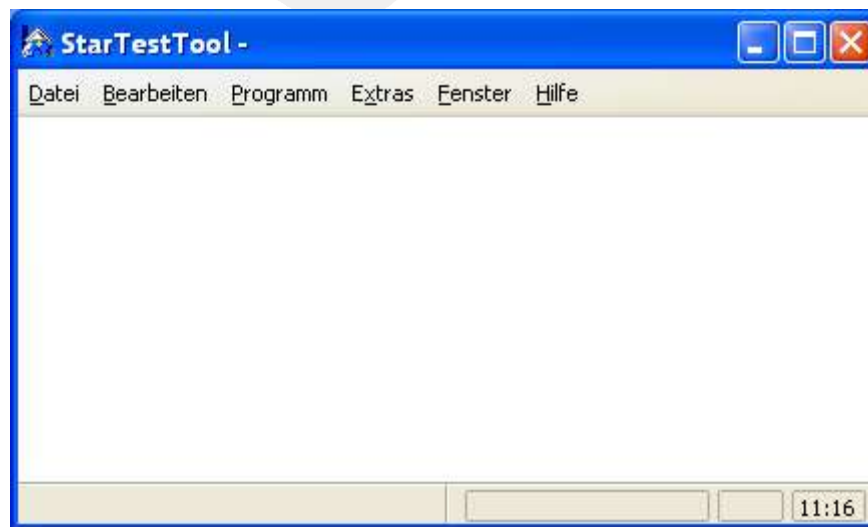
# Testtool – wofür und woher?

- Automatische GUI-Tests gegenüber Spezifikation anhand vorbereiteter Scripts
  - Vorteil: umfangreiche Tests ohne manuellen Eingriff
  - Nachteil: „Grafiktreibersyndrom“ (es funktioniert genau das, was getestet wird, alles andere ...)
- Testtool ist in den OOo-Paketen enthalten:
  - Linux: openofficeorg-testtool-1.9.74-1.i586.rpm
  - Windows: Angepasste Installation, weitere Komponenten, Testtool
- Download als eigenständiges Paket
  - <http://qa.openoffice.org/qatesttool/index.html>
  - Testtool kann innerhalb einer Versionslinie verwendet werden (1.1 / 2.0)

# TestTool – Scripte beschaffen

- Scripte liegen im OOo CVS ( qa/qatesttool )
  - `cvs co qa/qatesttool`  
für die jeweils aktuelle Version
  - `cvs co -r <buildtag> qa/qatesttool`  
<buildtag> sollte dem Ooo-Release entsprechen,  
z.B. ooo114 für OOo 1.1.4  
ggf. auf [dev@qa](mailto:dev@qa) oder [dev@de](mailto:dev@de) nachfragen
- Für Beta / Finalversionen evtl. als Download
  - <http://qa.openoffice.org/qatesttool/index.html>  
(TestToolEnvironment)

- Vorbereitung
  - OOo vor dem ersten Testlauf einmal starten, um Lizenz und Registrierungs Dialoge zu bestätigen
  - OOo **schließen** (Testtool startet OOo selbst)
- Aufruf
  - Windows: `<OOoinstalldir>\programm\testtool.exe`
  - Linux: `<OOoinstalldir>/programm/testtool.bin`



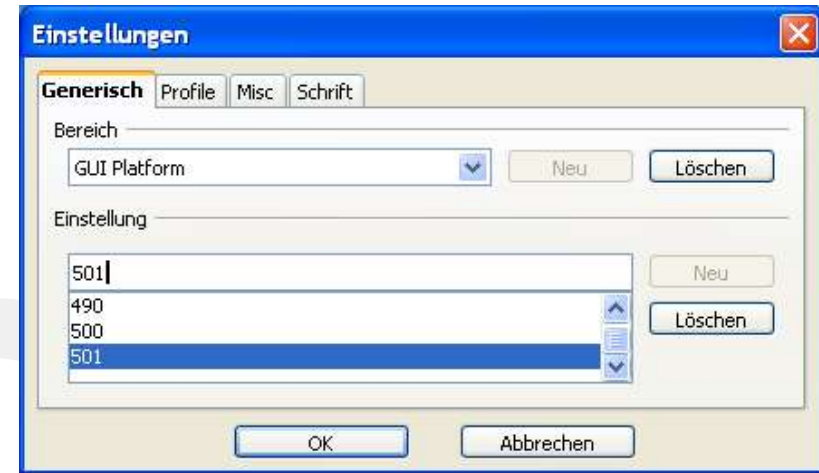
- Menü Extras - Einstellungen

- Generisch

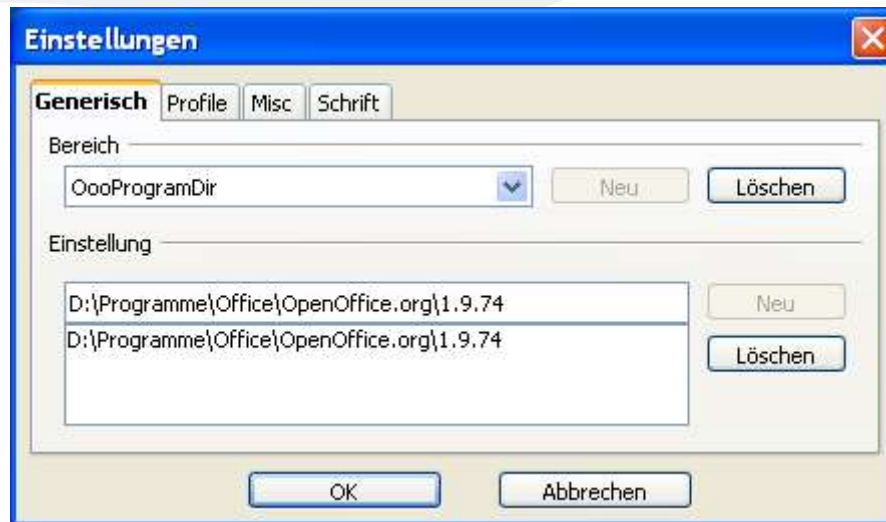
- Bereich GUI Plattform
- Einstellung hinzufügen
- Linux: 03; WindowsXP: 501

- Profile (am besten neues Profil anlegen)

- Log Basis Verzeichnis beliebiges Verzeichnis
- Basisverzeichnis Verzeichnis, in dem die Scripte abgelegt sind (gatestool des Checkouts)
- HID Verzeichnis <basisverzeichnis>/global/hid



- Für Snapshots (Windows) muss der Pfad zu OOo konfiguriert werden (unter Linux und für Beta / Releaseversionen wird er automatisch erkannt)
  - Generisch
    - Neuen Bereich OooProgramDir anlegen
    - OOo Installationspfad als Einstellung eingeben



# Testtool – Script starten

- Datei – Öffnen
- Script auswählen
- Script Starten mit <F5>
- wichtige Scripte:
  - framework/first/first.bas
  - framework/first/topten.bas
  - writer/update/w\_updt.bas
  - calc/update/c\_updt.bas
  - graphics/update/i\_updt.bas
  - graphics/update/d\_updt.bas
- Mindestens diese Scripte sollten für die Freigabe eines Micro-Releases erfolgreich durchlaufen



The screenshot shows a window titled "StarTestTool - E:\cvs\qa\qatesttool\framework\first\fir...". The window contains a VBA script with the following content:

```
!+
! /+*****
!+
!* owner : joerg.skottke@sun.com
!*
!* short description : first test, to check global fun
!*
!\*****

sub main

    use "global\update\inc\g_option.inc"
    use "framework\first\inc\first.inc"
    use "framework\first\inc\graph_in.inc"

    call hStatusIn ("framework", "first.bas", "The fir

    Dim StartZeit
    StartZeit = Now()
    printlog Chr(13) + " - All types of documents"
    call tAllNew

    if gUseSysDlg = FALSE then
        printlog Chr(13) + " - Import of all graphicf
        gApplication = "DRAW"
        call tInFilterCountCompare
        call tQuickInsertGraphNotAsLink
        printlog Chr(13) + " - Java applet test"
```

# Testtool – was passiert?

- Testtool startet das Script und prüft die Umgebung
- Testtool sucht nach installiertem OOo (in der Regel wird das „neueste“ OOo gefunden, wenn mehrere installiert sind)
- OOo wird mit -enableautomation gestartet (Achtung: WinXP SP2 – bei Nachfrage der Firewall, OOo als Server zulassen!)
- OOo wird vom Testtool ferngesteuert und führt die festgelegten Aktionen aus
- Ergebnisse werden detailliert in ein Logfile geschrieben (im Testtool angezeigt, oder als Textfile einsehbar)
- Zusammenfassung bzgl. Fehler und Warnungen

# Testtool – Ergebnisse bewerten

- Im Idealfall keine Fehler und Warnungen ausgegeben- alles OK (auf [dev@de](mailto:dev@de) melden)
- Warnungen (gelb)
  - Sind in der Regel bekannte Fehler oder unerwartete Einstellungen, die die Funktionalität nicht beeinträchtigen
  - Im Log lokalisieren, versuchen manuell nachzuprüfen, ggf. auf den Listen diskutieren
- Fehler (rot)
  - Aktion konnte nicht ausgeführt werden oder das Ergebnis weicht vom erwarteten Ergebnis ab
  - Im Log lokalisieren, versuchen manuell nachzuprüfen
  - Issue aufgeben, ggf. auf [dev@qa](mailto:dev@qa) oder [dev@de](mailto:dev@de) diskutieren

- Info im Netz:
  - <http://qa.openoffice.org/qatesttool/index.html>
  - <http://de.openoffice.org/dev/qa/>  
(muss aktualisiert werden)
- Handbücher (Einführung und Kochbuch):
  - <http://qa.openoffice.org/qatesttool/index.html>
  - Im checkout `qatesttool/documentation`
- Bei Problemen / Fragen:
  - [dev@qa.openoffice.org](mailto:dev@qa.openoffice.org)
  - [dev@de.openoffice.org](mailto:dev@de.openoffice.org)
- Issuezilla:
  - Component: qa; Subcomponent: testscripts

Pause

# Manuell Testen – warum?

- Jeder Anwender arbeitet verschieden – selten entsprechend der Spezifikation
- TestTool geht von definierten Grundeinstellungen aus
- TestToolscripts sind zwar sprachunabhängig, prüfen aber keine Besonderheiten der Lokalisierungen
- TestTool prüft nicht den Installationsvorgang

- Manuelles Abarbeiten von Testfällen und bewerten der Ergebnisse
- Testfälle sind vorgegeben und beschreiben Aktion und erwartetes Verhalten
- Testfälle sind Empfehlungen und keine Arbeitsanweisung, die 100% genau abgearbeitet werden muss (und auch nicht kann), geben aber trotzdem einen Rahmen für koordiniertes Testen
- Testfälle für 1.1 (Smoketests - qa):  
<http://qa.openoffice.org/testcase/index.html>
- Testfälle für 2.0 (TCM – I10n):  
[http://www.sunvirtuallab.com:8001/tcm2/tcm\\_login.cgi](http://www.sunvirtuallab.com:8001/tcm2/tcm_login.cgi)



# Vielen Dank für ihre Aufmerksamkeit

© Copyright 2005, André Schnabel

## **Public Documentation License Notice**

The contents of this Documentation are subject to the Public Documentation License Version 1.0 (the "License"); you may only use this Documentation if you comply with the terms of this License. A copy of the License can be found at <http://www.openoffice.org/licenses/PDL.html>.

The Original Documentation is "Release-QA bei OpenOffice.org; Automatisiertes und manuelles Testen".

The Initial Writer of the Original Documentation is André Schnabel Copyright (C) 2005. All Rights Reserved.

Initial Writer contact(s): [AndreSchnabel@OpenOffice.org](mailto:AndreSchnabel@OpenOffice.org)

Contributors are: