

Test Automation - Why, When & How

*Andreas Steen
Technical Sales Manager
IBM Rational Sweden*

Rational. software

Agenda

- **What**
- **Why**
- **When**
- **Where**
- **How**



What is Test Automation anyway?

- **Software testing is to assure that a system performs as expected**
- **Automated testing is using tools/software to assist in testing software**



Why bother?

- **Save Time**
 - But, creating test automation takes time..?
 - **Save Money**
 - But, the tools cost money..?
 - **Increase Quality**
 - But, it's the same tests that are executed..?
 - **It's FUN!**
 - No but. Automating is more fun than manual testing!
- **Pick the reasons that suits you.**

When should I do it?

- **Early in the applications life**
 - Design the application for test automation
 - Build up your automation as you build up the application
 - Bigger savings
- **During development, not during testing**
 - You will not have time when you need to deliver defects
 - Deadlines are always tight
 - “Automation fridays” ...

Where should I start?

- **Choose the right application**
 - Long life span
 - Reasonable stable GUI / Architecture
- **Choose the right parts in the application**
 - Use Record / Playback to explore
- **Choose the right tests to automate (quick wins)**
 - Use-Case based scenarios
 - Data driven function tests
 - Where do you spend most of your testing time?

How do I do it?

- **Record & Playback**
- **Maintainable Test Automation**
- **Script Types**
 - Function Tests
 - Use-Case / Scenario Tests
 - Utility Scripts
- **Re-Use**



Record & Playback

- **Start small**
 - Pick ONE task and automate it
 - **Get a “feel” for the application**
 - **Validate that automation will work**
 - **Find sweet-spots to automate**
- **Easy to do but don't try to maintain**



Maintainable Test Automation

- **Develop automation just like any other software**
- **Use good software engineering practices and techniques**
 - Organized, modular code
 - Exceptions and error handling
 - Version control
 - Logging
 - Collaboration with development
 - Separate data from logic

Script Types – Function Tests

- **Easy navigation - Complex verification**
- **What to look for**
 - Grids
 - Forms
 - Search functions
- **Why automate?**
 - Testing searches is boring and error prone
 - Easy to create automation scripts

Function Tests – How to do it

- **Record**

- Navigation
- Data inputs
- Verification Points

- **Enhance**

- Create loops
- Use external test data

- **Execute**

- This is where you find most defects

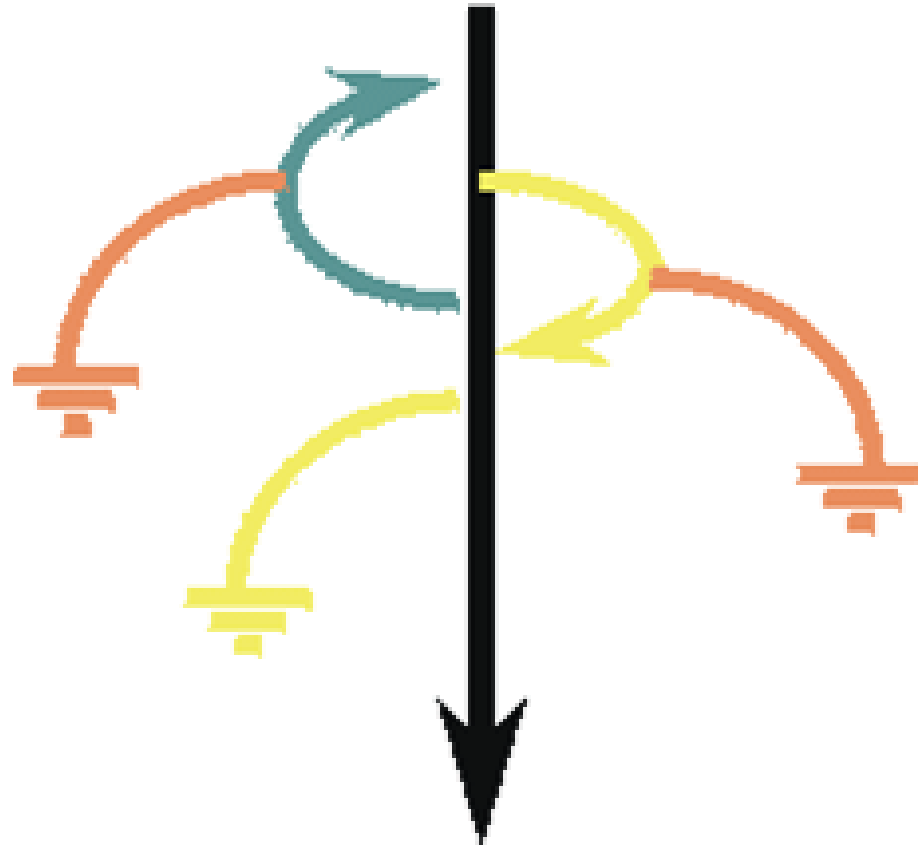


Script Types – Use-Case / Scenario Tests

- **Complex navigation - Simple or no verification**
- **What to look for**
 - Different types of users
 - Many screens
 - “Big” applications
- **Why automate?**
 - Detects overlaps / changes in the GUI
 - Smoke / Ghost tests
 - Find integration problems

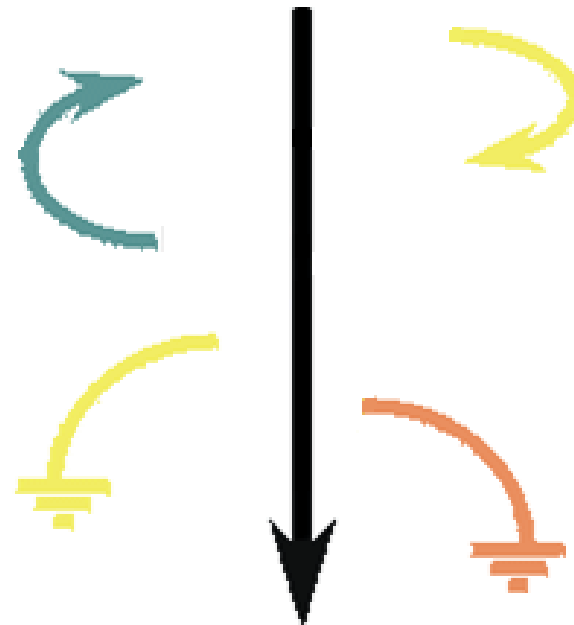
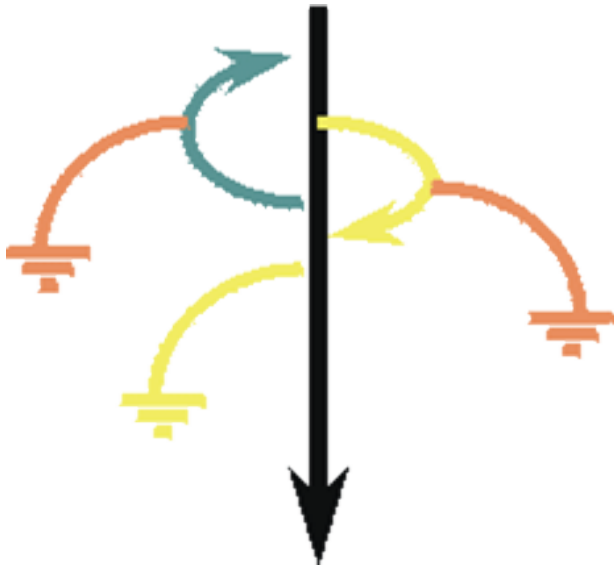
Use-Case Test – How to do it

- **Select a relevant Use-Case**

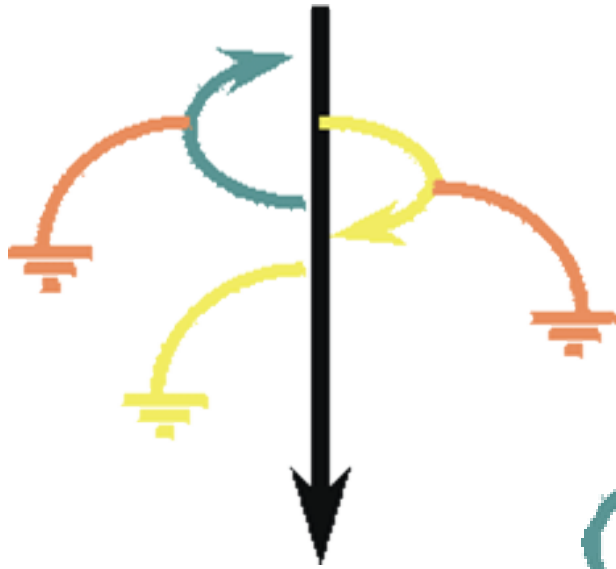


Use-Case Test – How to do it

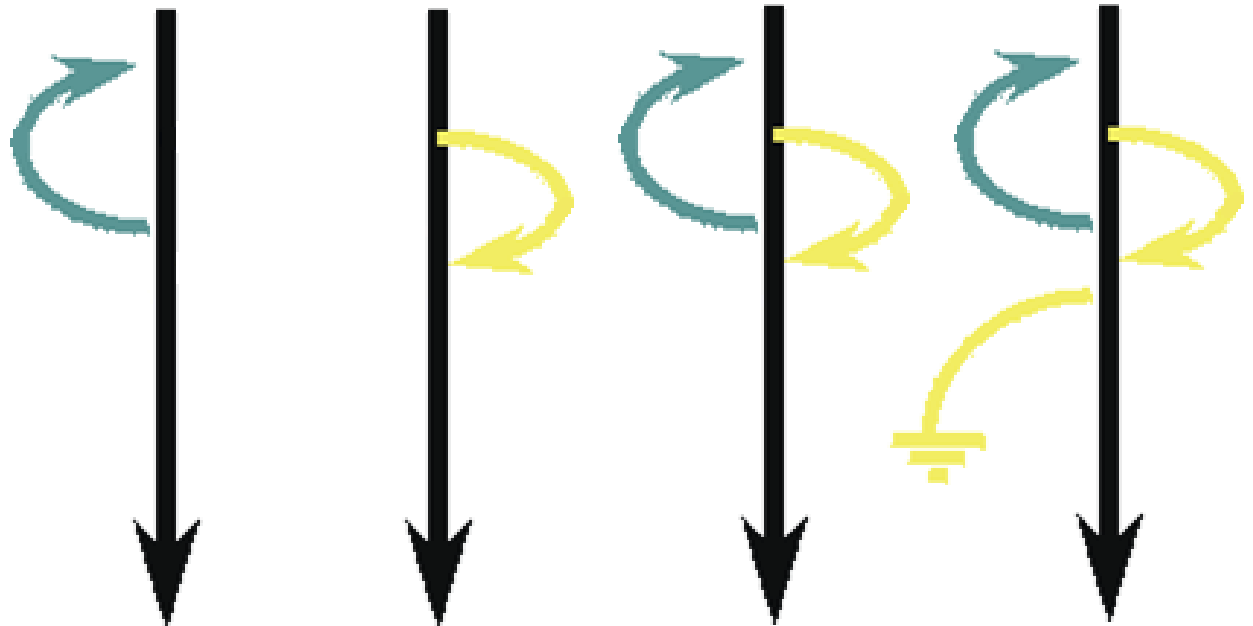
- Select a relevant Use-Case
- Break up into scripts



Use-Case Test – How to do it

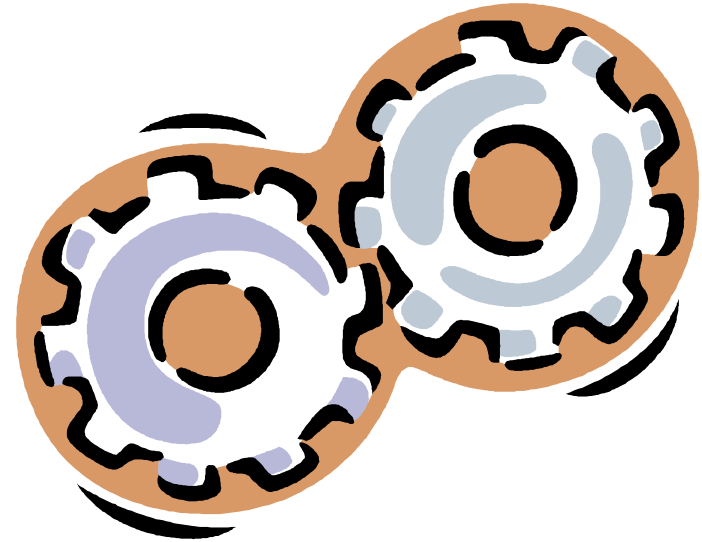


- Select a relevant Use-Case
- Break up into scripts
- Join scripts to run tests



Utility Scripts

- **Install & Verify**
 - Application to test
 - Other infrastructure
- **Populate data**
- **Clean up**
 - Reset databases
- **Recovery**
 - Application and Script out of sync
 - Application crashes



Re-Use

- **No Copy & Paste!**
 - Use global scripts
- **What do you need to re-use?**
 - Start / Restart / Shut down application
 - Get external data
 - Custom verification points
 - Calculations

Summary #1

- **Test Automation is a SOFTWARE PROJECT**
 - Architect, plan and develop your code for the long term
 - Writing good automation is challenging but worth it
- **Automation is a long-term investment with extended ROI**
 - It WILL take longer to create an automated test case than a manual test case
- **Automation doesn't find many defects**
 - It's a regression testing tool: it ensures today's build works like yesterday's and can allow late changes to the product

Summary #2

- **Applications Under Test WILL change**
 - Your automation needs to be architected for maintenance
- **Automation needs buy-in from development**
 - Use only automation approved controls
 - Automation blocking defects need to be fixed, even if customers won't see them
 - Demand information about application changes that break automation

Summary #3

- **Ask for help**
 - Colleagues
 - Partners
 - IBM
- **Read a Book**
- **Realize that it's actually EASY!**
 - YOU decide how complex you want it to be
- **Have fun!**

**IBM Rational Software
Development Conference 2009**
May 31 - June 4, 2009 | Orlando, Florida



Andreas Steen | andreas.steen@se.ibm.com

www.ibm.com/software/se/rational