

Test Automation: Beyond Regression Testing

Douglas Hoffman
Quality Program Manager, Hewlett-Packard

SSQA 11/14/06

© 2006 Software Quality Methods, LLC.

Regression Testing

1. IEEE 610.12: Selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still complies with its specified requirements.
2. Rerunning of tests to verify that a defect has been fixed.
3. Rerunning of tests to verify that a previously fixed defect remains fixed.

It's doing the same test repeatedly to look for a specific defect or a type of error.

© 2006 Software Quality Methods, LLC.

2 November 14, 2006

To Minimize the Chance of Encountering An Error

- Script an activity
- Run the script to be sure it works
- Make sure any encountered errors are fixed
- Repeat the activity as closely as possible
- Automate the activity to minimize variation

© 2006 Software Quality Methods, LLC.

3 November 14, 2006

Test Activities Sequence

1. Identify risk to be tested
2. Design test for that risk
3. Document the test
4. Run the test
5. Monitor SUT behaviors and compare/capture outcomes
6. Evaluate test results
7. Report test verdicts

© 2006 Software Quality Methods, LLC.

4 November 14, 2006

Automated Testing Usually Adds

- 3a. Run the test manually
- 3b. Fix any errors discovered in the SUT
- 3c. Implement the test case/script
- 3d. Debug the test case/script
- 6a. Maintain the test case/script when it (and not the SUT) causes the failure

This minimizes any chance for finding an error

© 2006 Software Quality Methods, LLC.

5 November 14, 2006

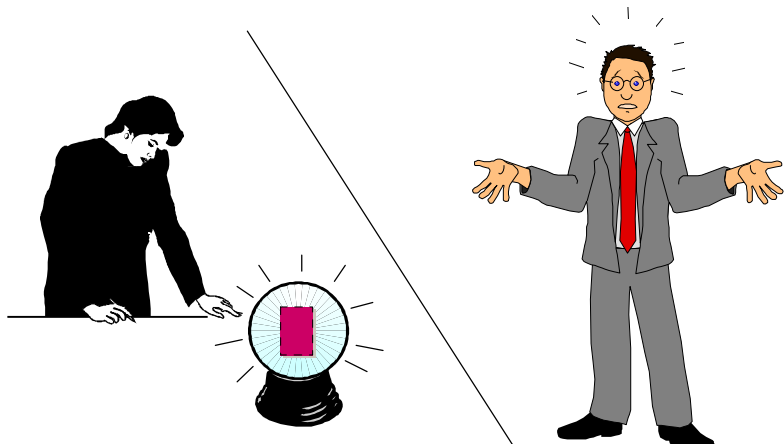
What Automation Can Do For Us

- Select the tests to be run
- Define the test activities
- Select and/or set the conditions
- Run the test exercise(s) (feed inputs to the SUT)
- Monitor and possibly log activities and outcomes (during and/or after the run), including internal program states, data buffers, files, etc.
- Capture specified outcomes
- Compare/evaluate outcomes to draw a verdict

© 2006 Software Quality Methods, LLC.

6 November 14, 2006

The Tough Part is the Test Oracle



How do you know pass from fail?

© 2006 Software Quality Methods, LLC.

7 November 14, 2006

Some Types of Non-Regression Automation

- Data driven
- Model based (lots of types of models – Cem Kaner identified 11)
- Random walks (Stochastic & non-stochastic)
- Function equivalence (random input)
- A/B comparison (potentially massive inputs)
- Statistical models
- Heuristic oracles
- Parallel checking for anomalies (e.g., data base corruption)
- Duration testing, life testing, load generation

© 2006 Software Quality Methods, LLC.

8 November 14, 2006

What Types of Errors Are We Seeking?

- Buffer overruns, security issues through massive variations on input corruption
- Special [unpredictable] cases (error of 23)
- Memory leaks, stack overflows (accumulation)
- Memory corruption, stack corruption
- Resource consumption/exhaustion
- Timing errors

© 2006 Software Quality Methods, LLC.

9 November 14, 2006

What Automation Can't Do^{*}

- Notice things that we haven't specifically told the test (or test mechanisms) to look at
- Decide on new courses of action (not written into the test) based on detection of potentially interesting occurrences
- Analyze boundary conditions, partitions, models, etc., to determine test conditions to cover [exception: tools will tell us how to test that the code does what the code does]

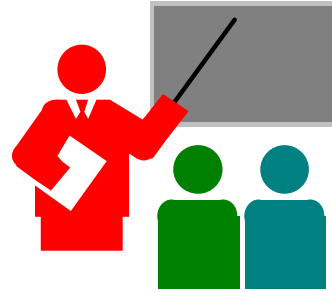
** Yet!*

© 2006 Software Quality Methods, LLC.

10 November 14, 2006

Conclusions

- Regression automation may not be worth much investment
- Other automation may find new types of errors
- The tough part is knowing pass from fail
- Who knows where it will go from here



© 2006 Software Quality Methods, LLC.

11 November 14, 2006