
Exploratory Test Automation

SSQA
April 10, 2012

**Douglas Hoffman, BACS, MBA, MSEE,
ASQ-CSQE, ASQ-CMQ/OE, ASQ Fellow**

Software Quality Methods, LLC. (SQM)

www.SoftwareQualityMethods.com

doug.hoffman@acm.org

About Doug Hoffman

I am a management consultant in testing/QA strategy and tactics. I help transform organizations. Software quality assurance, and especially software testing, had a reputation of being where failed programmers or programmer “wannabe’s” congregate. I believe that’s changing, and it’s through courses like this that we can change the perception. I gravitated into quality assurance from engineering. I’ve been a production engineer, developer, support engineer, tester, writer, instructor, and I’ve managed manufacturing quality assurance, software quality assurance, technical support, software development, and documentation. Along the way I have learned a great deal about software testing and automation. I enjoy sharing what I’ve learned with interested people.

Current employment

- President of Software Quality Methods, LLC. (SQM)
- Management consultant in strategic and tactical planning for software quality

Education

- B.A. in Computer Science
- MS in Electrical Engineering, (Digital Design and Information Science)
- MBA

Professional

- President, Association for Software Testing
 - Past Chair, Silicon Valley Section, American Society for Quality (ASQ)
 - Founding Member and Past Chair, Santa Clara Valley Software Quality Association (SSQA)
 - Certified in Software Quality Engineering (ASQ-CSQE, 1995)
 - Certified Quality Manager (ASQ-CMQ/OE, 2003)
 - Participate in the Los Altos Workshop on Software Testing and many others
-

Automation In This Context

Get the computer to do one or more of:

- Test design
- Setup or capture of pre-conditions
- Input selection/generation
- Automated test case
- Test execution control
- Actual results capture
- Expected results generation
- Results comparison

Most Automated Tests Today

- Are based on the functions of a test tool
- Automate a manual tester's actions
- Are a specific set of test activities (script)
- Work only at the UI or API level
- Do program checking at specified points in the script
- Are used to repeat and speed up manual testing

The Regression Test Approach

- Identify potential problems and tests necessary to expose them
- Identify and prioritize important risks
- Standardize documentation and procedures
- Design and create tests that are repeatable
- Rerun the tests often

Regression testing targets specific potential errors and repeatedly (and only) checks for those errors

Questions We Should Ask About Testing and Automation

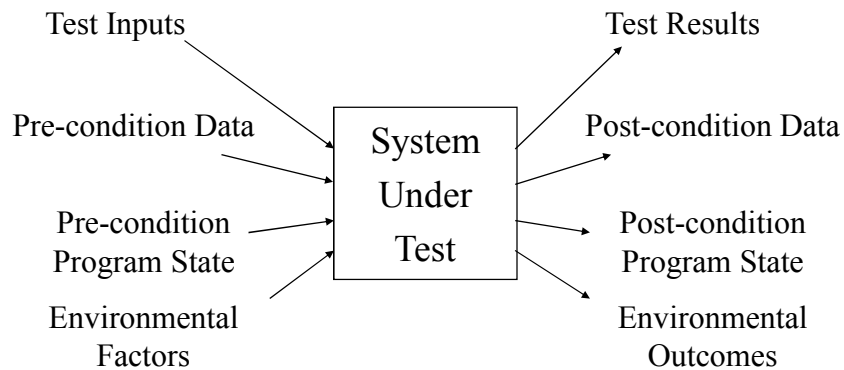
- Should we limit our thinking to what a tool does?
- Should we focus automation on things we can do manually and then script?
- Should we limit ourselves to UIs and APIs?
- Are we checking everything that's important?
- Do speedy manual tests find more or different bugs than manually running tests?
- Can inefficient or approximate tests be valuable?
- Must tests do the same things every time?

Exploratory Test Automation

- Does something new every time (except...)
- Does things a manual tester cannot do
- Gets “under the covers”
- May use massive numbers of iterations
- May have multiple parallel oracles

Exploratory test automation can find bugs we never imagined and couldn't find any other way

Expanded Test Execution Model



Implications of the Expanded Model

- We don't control all inputs
- We don't verify everything
- Multiple domains are involved
- The test exercise may be the easy part
- We can't verify everything
- We don't know all the factors

Oracles and Test Automation

Automated testing depends on our ability to programmatically detect whether the SUT behaves in expected or unexpected ways

Our ability to automate testing is fundamentally constrained by our ability to create and use oracles.

Key For Exploratory Automation: The Oracle

- The principle or mechanism for telling whether the SUT behavior appears OK or if further investigation is required
- Answers the question “is this the expected behavior?”
- Basic part of every test execution

Our ability to automate testing is fundamentally constrained by our ability to create and use oracles.

High Volume Random Tests

- Principle idea
 - High-volume testing using varied inputs
 - Results checking based on individual results or population’s statistical characteristics
- Fundamental goal is to have a huge number of iterations
 - The individual tests may not be not all that powerful or compelling
 - Input is varied for each step
 - Individual results may not be checked for correctness (e.g., heuristics or population statistics)
 - The power of the approach lies in the large number of tests

Low Volume Exploratory Tests

- Principle idea
 - One-at-a-time testing using varied inputs
 - Use automation to make exploration easy
- Fundamental goal is to enable exploration
 - Variations on a theme (modification of existing tests)
 - Quick-and-dirty generation of tests/data/comparisons
 - Background checking
 - Memory leak detection
 - File modification
 - Etc.
 - Command line, API, or UI based variations

Douglas Hoffman

Copyright © 2004-12, SQM, LLC.

13

Examples of Exploratory Automation

- Random events (Cem Kaner's "Telenova" example)
- "Dumb monkeys" (Noel Nyman)
- "Sandboxed" random regression tests (Kaner/Hoffman)
- Single and multi-threaded database locking (Hoffman)
- Database load/unload dropouts (Hoffman)
- Statistical packet profiles for data link testing
- Periodic database unload/check
- Database forward/backward link consistency
- Device front panel state machine long walks
- Database unbalanced splitting
- Random machine instruction generation
- 1/3 or 3x timing difference heuristic in test harness

Douglas Hoffman

Copyright © 2004-12, SQM, LLC.

14

Advantages of Exploratory Automation

- Does something new every time
- Uses massive numbers of iterations
- Oracles may check internal information
- May have multiple parallel oracles
- Supplements regression tests
- Can uncover obscure bugs
- Can uncover bugs impossible to find manually

Disadvantages of Exploratory Automation

- May not be repeatable
- Difficulty capturing program and system information for diagnosis
- May have multiple parallel oracles
- Coordination of autonomous oracles with the test
- Does not provide rigorous coverage
- Can uncover bugs that can't be fixed

Exploratory Test Automation In This Context

Designing tests that get the computer to do:

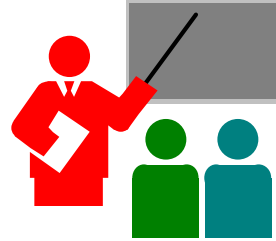
- Setup or capture pre-conditions
- Input selection/generation
- Test execution control
- Actual results capture
- Expected results generation
- Results comparison

Types of Test Oracles

- None
- Perfect
- Consistency
- Self-Verifying
- Model based
- Hand-crafted
- Heuristic
- Statistical
- Computational
- Human

Summary

- Exploratory Test Automation creates tests that get the computer to check for unknown errors
- Opposite of regression tests
- Varying what the test does
- Test oracles are fundamental in testing, especially in automated testing



Douglas Hoffman

Copyright © 2004-11, SQM, LLC.

19

Good Characteristics for Automated Tests

- Start with a known state
- Build variation into the tests
- Plan for the capture of data on error
- Check for errors during the test run
- Capture information when an error is noticed
- Minimize error masking and cascading

Douglas Hoffman

Copyright © 2004-12, SQM, LLC.

20

Automated vs. Manual Tests

An automated test is not equivalent to the most similar manual test:

- The mechanical comparison is typically more precise (and will be tripped by irrelevant discrepancies)
- The skilled human comparison will sample a wider range of dimensions, noting oddities that one wouldn't program the computer to detect
- The automated input is consistent, while humans cannot closely replicate their test activities

Automation Narrows Our Scope

An automated test is more limited than a manual test:

- The test exercise must be automated
- We can only check machine available results
- We can only check result values that we can predict or verify
- Cannot back up when something unexpected happens