

# Using Test Oracles in Automation

---

Software Test Automation Conference  
Spring 2003

Douglas Hoffman  
Software Quality Methods, LLC.  
24646 Heather Heights Place  
Saratoga, California 95070-9710  
Phone 408-741-4830  
Fax 408-867-4550  
doug.hoffman@acm.org  
www.SoftwareQualityMethods.com

Copyright © 2001 - 2003, Software Quality Methods, LLC. No part of these graphic overhead slides may be reproduced, or used in any form by any electronic or mechanical duplication, or stored in a computer system, without written permission of the author.

---

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

1

## Topics

---

- Some Models of Software Testing
- Automating Software Tests
- Strategies for capturing and comparing results

---

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

2

## Running A Software Test

---

- Test setup
  - SUT program state
  - Data values
  - System environment
- Run test exercise
- Capture/compare actual with expected results

---

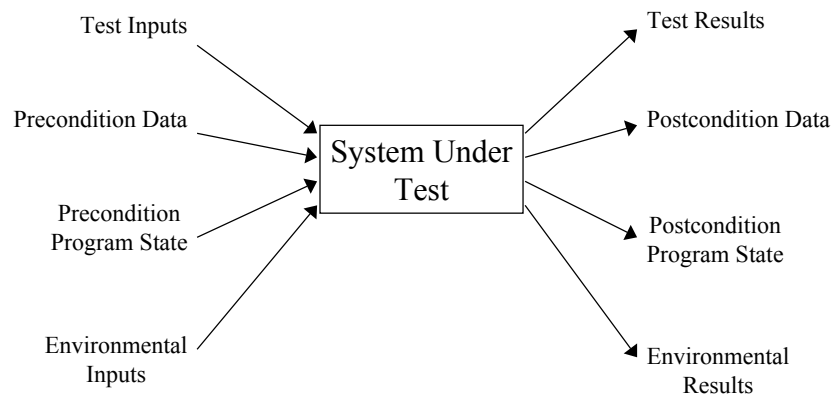
Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

3

## Hoffman's Expanded Testing Model

---



---

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

4

## Fully Automated Software Tests

---

- Able to run two or more specified test cases
- Able to run a subset of the automated test cases
- No intervention needed after launching tests
- Automatically set-up and/or record relevant test environment
- Run test cases
- Capture relevant results
- Compare actual with expected results
- Report analysis of pass/fail

---

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

5

## Some Hard Parts of Test Automation

---

- Some methods of input stimulation (running a test)
- Capturing the results (noticing what happens)
- Differentiating expected from unexpected results (knowing pass from fail)
- Dealing with errors (capturing and recovery)

---

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

6

## What a Test Oracle Is

---

**Two slightly different views on the meaning of the term:**

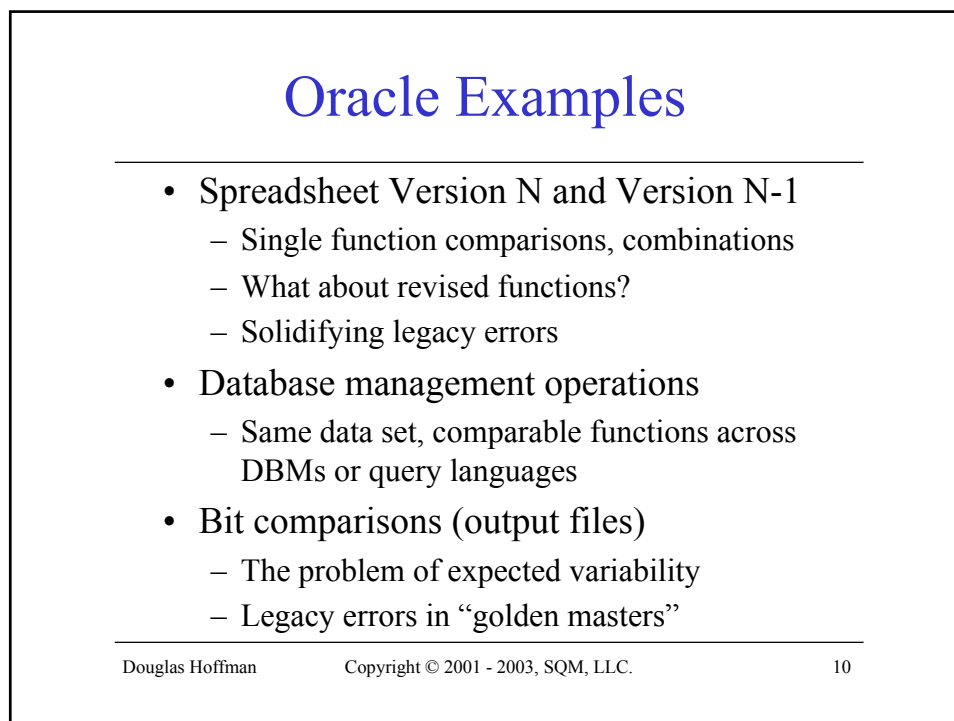
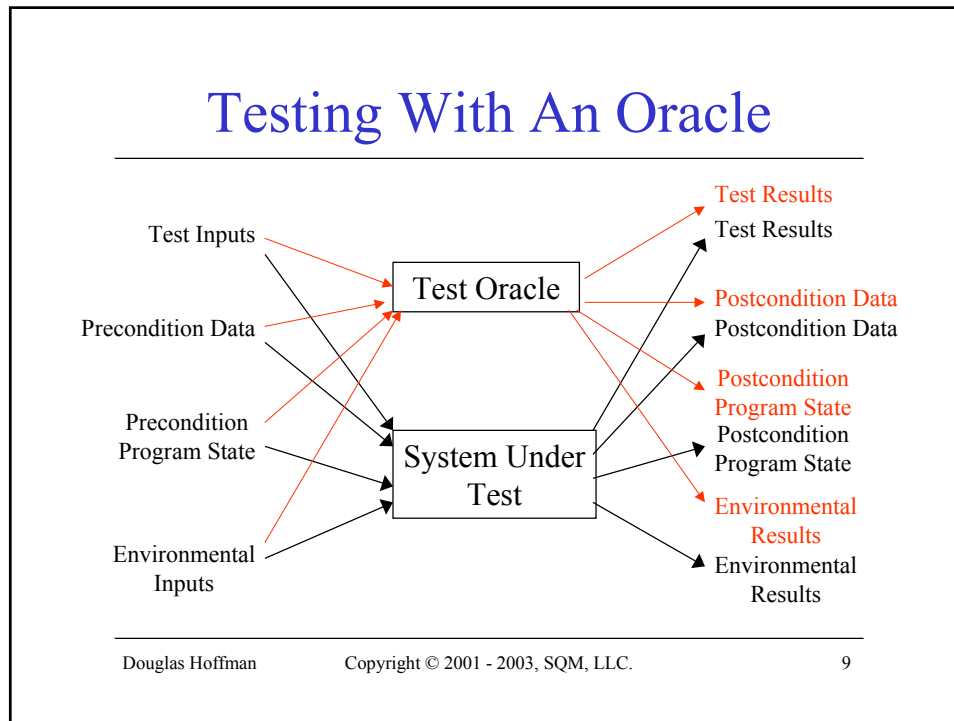
- ***Reference Function:*** You ask it what the “correct” answer is.  
*(It tells you the expected result.)*
  
- ***Reference and Evaluation Function:*** You ask it if the program passed the test.  
*(It tells you if the actual result is OK.)*

## A Test Oracle

---

**With an oracle, you can compare the test’s actual result to a reference value (predicted or expected result) to decide whether the program passed or failed the test.**

- ***Deterministic oracle*** (mismatch means program fails)  
*(This is the commonly analyzed case.)*
  
- ***Probabilistic oracle*** (mismatch means program probably fails.) *(I analyze these in more detail.)*



## Some Deterministic Reference Functions

---

- Saved result from a previous test run
- Parallel function
  - previous version
  - competitor's product
  - reference standard function
  - alternate platform
  - custom model
- Inverse function
  - mathematical inverse
  - operational inverse (e.g. split a merged table)
- Useful invariant rules (e.g.  $\sin^2(x) + \cos^2(x) = 1$ )
- Expected result encoded into data

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

11

## Some Probabilistic Reference Functions

---

- Compare [apparently] complete attributes
  - compare calculated results of two parallel math functions (but ignore duration, available memory, pointers, display)
- Almost-deterministic approach (statistical evaluation)
  - test for outliers, means, predicted distribution
- Compare incidental but informative attributes
  - duration, order
- Check [apparently] incomplete attributes
  - ZIP Code entries are 5 or 9 digits
- Check probabilistic attributes
  - a parent is usually older than their children

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

12

## Oracle Characteristics

- Completeness of information
- Accuracy of information
- Usability of the oracle or of its results
- Maintainability of the oracle
- Complexity
- Temporal relationships
- Costs

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

13

## Oracle Strategies for Verification

	No Oracle	True Oracle	Consistency	Self Referential (SVD)	Heuristic
Definition	- Doesn't check correctness of results	- Independent generation of all expected results	- Verifies current run results with a previous run (Regression Test)	- Embeds answer within data in the messages	- Verifies some characteristics of values
Advantages	- Can run any amount of data (limited only by the time the SUT takes)	- All encountered errors are detected	- Fastest method using an oracle - Verification is straightforward - Can generate and verify large amounts of data	- Allows extensive post-test analysis - Verification is based on message contents - Can generate and verify large amounts of complex data	- Faster and easier than True Oracle - Often much less expensive to create and use
Disadvantages	- Only spectacular failures are noticed.	- Expensive to implement - Complex and often time-consuming when run	- Original run may include undetected errors	- Must define answers and generate messages to contain them	- Can miss errors - Can miss systematic errors

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

14

## 'No Oracle' Strategy

---

- Easy to implement
- Tests run fast
- Only spectacular errors are noticed
- False sense of accomplishment

---

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

15

## True Oracle Strategy

---

- Independent implementation
- Coverage over domains
  - Input ranges
  - Result ranges
- Provides “Correct” results
- Usually expensive
- Never complete

---

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

16

## Consistency Strategy

---

- A / B compare
- Check for changes
- Regression checks
  - Validated
  - Unvalidated
- Alternate versions or platforms
- Foreign implementations

## Self-Referential Strategy

---

- Embed results in the data
- Cyclic algorithms
- Shared keys with algorithms

## Heuristic Strategy

---

- Rules of thumb
  - similar results that don't always work
  - low expected number of false errors, misses
- Levels of abstraction
  - General characteristics
  - Statistical properties
- Simplify
  - use subsets
  - break down into ranges
  - step back (20,000 or 100,000 feet)
- Other relationships not explicit in SUT
  - date/transaction number
  - one home address
  - employee start date

---

Douglas Hoffman

Copyright © 2001 - 2003, SQM, LLC.

19

## Choosing Which Strategy

---

- Decide how the oracle fits in
- Identify the oracle strategy or combinations
- Prioritize testing risks

---

Douglas Hoffman

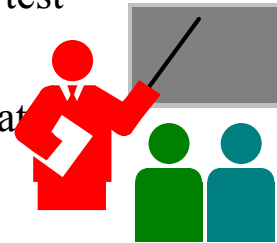
Copyright © 2001 - 2003, SQM, LLC.

20

## Summary

---

- Test oracles are critical factors in making good automated tests
- A variety of different types of test oracles are possible
- There are many ways to evaluate results



## References

---

- Nyman, Noel; "Self Verifying Data - Validating Test Results Without An Oracle" STAR East, 1999.
- Hoffman, Douglas; "A Taxonomy of Test Oracles" Quality Week 1998.
- Hoffman, Douglas; "Heuristic Test Oracles" Software Testing and Quality Engineering Magazine, Volume 1, Issue 2, March/April 1999.
- Hoffman, Douglas; "Mutating Automated Tests" STAR East, 2000.