A Taxonomy for Test Oracles

Quality Week ‘98

Douglas Hoffman
Software Quality Methods, LLC.
24646 Heather Heights Place
Saratoga, California 95070-9710
Phone 408-741-4830
Fax 408-867-4550

Copyright © 1998, 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.
Test Oracles
Generating Expected Results

- Manual verification of results (human oracle)
- Separate program implementing the same algorithm
- Software system simulator to produce parallel results
- Hardware simulator to emulate operations
- Earlier version of the software
- Same version of software on a different hardware platform
- Check of specific values for selected known responses
- Verification of consistency of generated values and end points
- Sampling of values against independently generated expected results
Test Automation

is not just machines running tests!
Test Automation
includes interpreting results!
Human Oracles

- Norm for manual testing
- Sometimes slower than computers
- Can’t observe system internals
- Loses concentration
- Easily “trained” to overlook errors
I-P-O Testing Model
(Black Box)
Expanded Testing Model
(Black Box)

- Test Inputs
- Precondition Data
- Precondition Program State
- Environmental Inputs
- System Under Test
- Test Results
- Postcondition Data
- Postcondition Program State
- Environmental Results
Testing Model With Oracle
Oracles Modeled in Testing

• Differ based on SUT
• May be more than one for SUT
• Inputs may effect more than one oracle
• Oracle only produces some results
Oracle Characteristics

- Completeness of information from oracle
- Accuracy of information from oracle
- Independence of oracle from SUT
  - Algorithms
  - Sub-programs and libraries
  - System platform
  - Operating environment
Oracle Characteristics
(continued)

• Speed of predictions
• Time of execution of oracle
• Usability of results
• Correspondence (currency) of oracle through changes in the SUT
Scale of Oracle Characteristics

None

Complete

Independent

Dependent

Correspondence of Results to SUT
Oracle Measures

- Oracle may become as complex as SUT
- More complex oracles make more errors
- Close correspondence reduces maintainability
- Close correspondence makes common mode faults likely
Types of Oracles

- True
- Stochastic
- Heuristic
- Sampling
- Consistent
True Oracle

Pro:

- Independent of SUT
- Faithful results
- May exhaustively test
- Good for any test case

Con:

- Costly
- Complex
- Slow
- Hard to maintain
**Stochastic Oracle**

**Pro:**
- Random sampling
- Fewer values
- Uniform SUT coverage
- Can select desired coverage level
- Less Costly
- May be simple

**Con:**
- Can miss systematic faults
- Uniform SUT coverage
- May be slow to verify
- Cannot focus

Douglas Hoffman

Copyright © 1998, SQM, LLC.
Heuristic Oracle

**Pro:**
- Check selected points
- Interpolate between using heuristic
- Fast for regular algorithms
- Can ‘check’ large volumes of data

**Con:**
- Can miss systematic faults
- Can miss incorrect algorithms
- Need points and heuristic
- Inflexible
Sampling Oracle

Pro:
• Can select easy values
• Quick
• Inexpensive
• Can focus

Con:
• Can miss systematic faults
• Can miss incorrect algorithms
• Likely to miss specific faults
• Easily biased
Consistent Oracle

**Pro:**
- Good for monitoring changes
- Good for checking for side-effects
- Good for regression testing
- Quick for some environments
- Can check huge volumes of raw data

**Con:**
- Legacy errors not found
- May be slow to verify
- May be difficult to identify fault
- Requires maintenance
Running An Oracle

- Type of results
- Time of running
- Method of verification
  - Manual
  - Automated
    - With test case
    - With automated test environment
Conclusions

• Different types of oracles possible
• Some kind of oracle needed
• Oracle not constrained like SUT
• Solutions differ with SUT
• Oracles are part of automation