

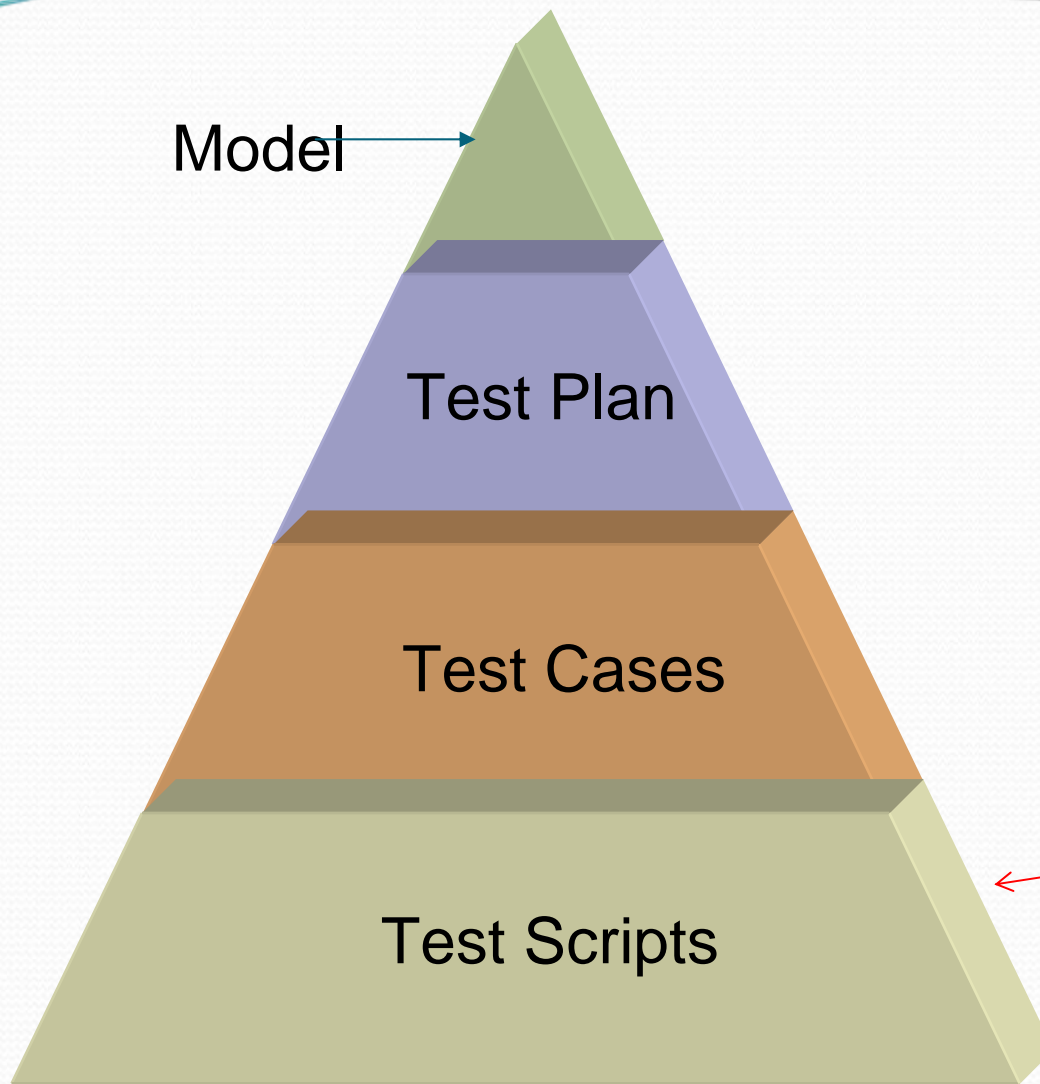


COMP 3711

(OOA and OOD)

**Software Testing 4b
Automated Scripting
&
Scripting Techniques**

Software Test Automation – Fewster/Graham (chapter 3)



What is
automated
test script?

Scripts as Programs

- No less difficult than programming software applications
 - Will be bugs in test scripts
 - Need to test the test
- Trick is to minimize the complexity and minimize the programming
- Avoid building *all* scripts down to the keystroke level
 - *Keystroke level* script not generally needed for manual testing

Fundamental Problem

- Actions to control a program must be specified in great detail
- Programs under development change frequently
- It would be much nicer if scripts did not have to be programmed in detail
- Specify *test case* (*what you want to test*) and have it tested

Scripts and Test Cases

- Traditionally one to one
- Factor out commonality
- Eventually be able to build new test without coding new scripts
- Build small modular scripts that can be assembled into tests
- Control loops into the script

Script Documentation

- A little bit goes a long way
 - No substitute for well designed, well organized code
 - Automatically generated code may require more documentation
1. Purpose
 2. Inputs and outputs
 3. Anything tricky or unusual in the implementation

Attribute	Good set of test scripts	Poor set of test scripts
Number of scripts	Fewer (less than 1 script per test case)	More (at least one script for each test case)
Size of scripts	Small – with annotation, no more than 2 pages	Large – many pages
Function	Each script has a clear single purpose	Individual scripts perform a number of functions, typically the whole test case
Documentation	Specific documentation for users and maintainers, clear, succinct and up-to-date	No documentation or out of-of-date; general points, no detail, not informative

Attribute	Good set of test scripts	Poor set of test scripts
Reuse	Many scripts reused by different test cases	No reuse; each script implements a single test case
Structured	Easy to see and understand the structure and therefore to make changes; follow good programming practices, well organized control constructs	An amorphous mass, difficult to make changes with confidences; spaghetti code
Maintenance	Easy to maintain; changes to software only require minor changes to a few scripts	Minor software changes still need major changes to scripts; scripts difficult to change correctly

Object Recognition

- Uniquely identifying objects in a user interface is a significant problem
- Especially when they change in some way
- Theoretically the object name should work
- Not always available or unchanged

Object Properties

- Not all properties of all objects are available to the test tool
- Especially objects from third parties
- Foundation principle of verification points is the comparison of current object properties against base line values

Scripting Techniques

1. Linear
2. Structured
3. Shared
4. Data-driven
5. Keyword-driven

Not mutually exclusive

1. Linear Scripts

- Generally the result of recording
- Contains all keystrokes to run application, usually a complete test case
- May include comparisons/verifications
- Should always play back after recording as a sanity check

Advantages of Linear Scripts

1. No upfront work or planning required
 - though not advised
2. Can get started quickly
3. An audit trail of what was done
4. Don't need programmer
 - As long as the script does not have to be changed
5. Good for demonstrating tool

When to use Linear Scripts

- demos and training
- repetitive action
- populating a database
- set up and clear-up
- conversion testing

Disadvantages of Linear Scripts

1. Labour intensive
2. Must start over each time
3. Inputs and comparisons hard wired into script
4. No sharing or reuse
5. Vulnerable to software changes
6. Expensive to change
7. Very fragile when running
 1. played versus recorded environment'

2. Structured Scripts

- Like structured programming
- Uses control instructions
- If statements for decisions
- Loops for iteration
- Requires programming skill

Advantages

1. Script is more robust
2. Can deal with similar situations using loops
3. Can be made modular
4. Can check for specific things to keep the test from failing

```
If message = "Replace existing file?"  
LeftMouseClicked 'Yes'  
EndIf
```

Disadvantages

- Now a more complex program
- Data still hard-wired into script

3. Shared Scripts

- Used by more than 1 test case
- Build a script that performs a task that is repeated in different tasks
- Can be specific to one application or more general
- Work best for small systems / utilities

Advantages

1. Similar tests take less time to implement
2. Maintenance costs lower
3. Obvious repetition eliminated
4. Can afford to put more intelligence into writing each script
 - Because they are used more than once
 - EG: wait
5. Keep up with user interface changes

Disadvantages

1. More scripts to catalog
2. Often specific to one part of the software
3. Not much use if people don't use them
 - Need to be organized into library
 - Also need to be carefully documented
 - Programmers look for reusable function for 2 min...

4. Data Driven Scripts

- Test inputs stored in separate file
- Control information is still embedded in script
- Can run more than 1 test with a script by changing the data file
- Or the same test instructions can be repeated using a different data record from the file
- Script decisions can be based on presence or absence of fields in data records

Advantages

1. Implement many more test cases with very little effort
 - Add new records to data file
 - No new script code required
2. Format of data file can be arranged to suit the testers
3. Can put expected results in same data file

Disadvantages

1. Initial setup takes a great deal of effort
2. Specialized programming support required
3. Must be well managed

5. Keyword Scripts

- Move some of the intelligence from the script and put it into the data file
- Keywords, which represent small *sub scripts* are used in the main control script
- *Sub scripts* stored separately
 - Should be fairly generic
- Supporting scripts:
 - “know” about the software under test
 - But know nothing about the test cases

Trick is to get the right
Key Word

Keyword Scripts: Example

Supporting scripts:

CalculatorOpen
CalculatorSum
CalculatorSubtract
CalculatorSaveAs

Test Files

Test 1 CalculatorOpen
CalculatorSum 4 6
CalculatorSubtract 12 6
CalculatorSaveAs test 1

Test 2 CalculatorOpen
CalculatorSum 4 6
CalculatorSubtract 12 6
CalculatorSaveAs test 1

Control script

For each test

 OpenFile testid

 for each record

 call 'keyword'

Keyword Scripts

- Additional layer of Separation between:
 - Construction of actual tests
 - Their implementation using a test tool
- Test consist of high level commands (key words)
- *Descriptive* rather than *prescriptive*
- Spreadsheet or data file actually describes the test
- High level commands implemented as function calls in the test tool scripting language

Advantages

1. Changes in target system interface should only require changes in one or two keywords
2. Tests are easier to specify and manage because they are essentially written in a higher level language
3. Change in test tool requires rewrite of action word scripts but the tests themselves do not have to be changed

More Advantages

- 4. Number of scripts is a function of size of software not the number of tests
- 5. Tests can be implemented in a tool independent manner
- 6. Implementation can be tailored to suit the tester.
- 7. Separation between test developer and programmer

Disadvantages

1. Requires very significant early commitment
2. Requires very significant architectural skill to design the keyword library
3. Requires significant programming skill to implement the library

Design of Keyword System

- Essentially a multilayer function library
- At minimum 2 layers
 - Function corresponding to each key word
 - Functions for "atomic" or low level actions in the application
- One of the difficult decisions is selecting the keywords

Control Script

- Main control script
- Reads the spreadsheet lines one at a time and takes the appropriate action
 - Usually means calling a function
- Built around a loop with a giant conditional expression

Using Spread Sheets to Store Tests

- Spreadsheets are particularly good for function calls and arguments
- Lines can contain comments or other stuff that is ignored by the navigation script
- Keyword is usually the text in the first field

Risks

- Incomplete or incorrect use of the method
- Underestimating effort needed to interpret test results
- Neglecting test maintenance
- Political problems
 - Difficulty justifying expenditures
- Technical problems
 - Object recognition