

***PureTesting***  
*Testing Thought Leadership*

**Creating Real-World Test  
Cases using Extension to  
Noun and Verb technique**

**Wednesday, September 26**  
**CONQUEST 2007**



# Noun and Verb Technique

## \*\*\*Nouns and Verbs, Adjectives and Adverbs

- Some software systems exist to manipulate specific kinds of data
  - The kinds of data sets are the nouns
  - The manipulations are the verbs
  - Adjectives and adverbs affect the manipulation
- Identify these items
- Write test in sentences
- Even trace coverage
- For example, an ATM system should
  - Accept deposits
  - Process withdrawals
  - Answer inquiries
  - Kinds of deposits, withdrawals, inquiries?
  - Quickly? Securely?

This test design technique comes from Elisabeth Hendrickson.

\*\*\*Rex Black: Effective and Efficient Testing Course

## Extending Noun and Verb Technique

- Challenges in using the technique
  - Often Requirements are not sufficiently detailed
  - Extracting enough Noun and Verbs is difficult
  - Writing complex scenario based tests from this technique is difficult
- The technique has been extended at PureTesting and used successfully
- Next few slides will present the extension with the help of an example...

## Extension

- Extract the nouns and verbs
- Use 5W1H to extract adjectives, adverbs and more information
- Use pre and post conditions to extract more nouns and verbs
- Enumerate properties of every noun
- Break down these properties to base elements
- Use boundary values, equivalence classes on the base elements
- Write tests for individual properties
- Combine operations (verbs) to write scenario based test cases of arbitrary complexity



## Applying the Extended Technique

## Example: File Open

- Requirement fragment:
  - The <given> application allows users to open supported files.
- With the given requirement fragment
  - Very few meaningful test cases can be written using any test design technique
  - Using out extension to noun & verb technique we plan to write extensive test cases

## Extracting the Nouns and Verbs

- The <given> application allows users to open supported files
- Extracting Nouns
- Extracting Verbs
- Extracting Adjectives and Adverbs

## File open...

- Noun
  - Look for properties of this noun
  - Also ask What, Why, When, Where, Who, Which, How, How much/many
- Verb
  - Look for properties of this verb
  - Also ask What, Why, When, Where, Who, Which, How, How much/many

## File: Properties

- Enumerate all the properties of FILE
- File has
  - Name
  - Size
  - Location
  - Extension
  - Metadata
    - Creation date
    - Accessing date
    - Modification date
  - State

## File: Properties

- Type
- Access Control
- Password protection
- Attributes
- Compression Support
- Encryption Support
- Creating application version

## Breaking Down the Properties: Name

- Name - is a string
- String - is a SEQUENCE of Characters
  - Sequence = Length of string
  - Characters are elements of some valid character set
- (THUS) Name
  - Has a length
  - Has a set of allowed characters
  - And some characters are not allowed (/ or \ in file name on windows, spaces on Linux/Unix platforms)

## Applying Boundary Values

- Name/String
  - Has a length (maximum and minimum)
  - Has a set of allowed characters (valid equivalence class)
  - And some characters are not allowed (invalid equivalence class)

## Properties: Location

- Location
  - Is a string
    - Has a length (maximum and minimum)
    - Has a set of allowed characters (valid equivalence class)
    - And some characters are not allowed (invalid equivalence class)
  - Also denotes physical location
    - On a locally connected drive (HDD, CD, Floppy...)
    - On the LAN
      - Using UNC path
      - Other operating systems (e.g. Linux connected using samba)
    - On a web page as a link

## Properties: State

- State
  - in use/not in use
  - Saved/unsaved
- In use/open
  - Unmodified
  - Modified
  - New
- New/Modified
  - Without data
  - With data

## Properties: Type

- Type:
  - Supported/not supported by the application
  - Binary or Text
  - Valid/invalid (corrupt)

## Properties: Extension

- With extension
- Without extension
  
- Correct extension
- Incorrect extension

## Properties: Attributes

- Read only
- Writable
- Archive
- Hidden
  
- ...

## Dealing with the Verb(s)

- Applying 5W1H
- Open What?
  - File with various properties
  - Supported and unsupported files
  - ...
- Open Where?
  - Supported Operating Systems
    - Application under test
    - Browser plug-ins, if supported

## Dealing with the Verb(s)...

- Open how?
  - Open in various ways
    - double click
    - drag and drop
    - command line
    - application menu
    - open with
    - ...
- Open how many?
  - Open one (minimum number of) file
  - Many files
  - Max. number of files that can be opened simultaneously...
- How fast
  - How much time does it take to open the file?

## Some Simple Positive Test Cases

- Open a (supported) file
  - Minimum size
  - Maximum size
  - Minimum name length
  - Maximum name length
  - Located on a network drive/web
  - With and without appropriate extension
  - Created using older/newer version of creating application
  - Password protected
  - Read-only
  - ...

## Some Simple Negative Tests

- Open file(s) of
  - Unsupported type
  - Changed extensions
  - Corrupt file
  - Invalid/non-existent name
  - ...

## Scenario Based Tests

- Open a new file. Open another existing but closed file.
- Open an already open file.
- Open an existing file. Modify it. Open the same file again.
- Open an existing file. Modify it. Open another file.
- Open a local file. Open another file on the web.
- Open multiple files together. Check performance
- Open a very large file on the network. Check performance.
- Repeat the Open operation a number of times (repetitive tests for resource leaks...)

## Expanding the requirement

- What (after file open)?
- File is opened
  - Where
    - in the application window
      - What are properties of application window?
        - Location on the screen
        - Size of the window
        - State of GUI elements such as menus/buttons (enabling/disabling etc.)
      - What are the properties of OPEN FILE?
        - Cursor location
        - Zoom level
        - ...

## Writing more Tests

- What is the position of application once file is opened
- What is the cursor position after opening the file
- At what zoom level does the file open?
- ...

## Summary

- Using the extended N&V technique testers can
  - Write effective set of test cases
  - Reduce the impact of bad/missing requirements
  - Raise important questions about software behavior
  - Bring more uniformity in the coverage even when different testers write tests cases

## Exercise

- Use Noun and Verb technique to create test cases for the following requirement excerpt

A word processing software has “Find” feature. User can invoke the find dialog to enter the string to be found. Find is case insensitive and circular. Find button is used repetitively to find the next occurrence.

Note...

- Focus on test cases for Find feature.
- After finding the test cases using noun and verb technique elaborate the requirements by speaking out the actions.
- Create more test cases using pre and post conditions.



## Q & A