The Three Holy Grails of Test Development

Hans Buwalda
The 3 Holy Grails of Test Development

Hans Buwalda
CMG - TestFrame Research Centre
hans.buwalda@cmg.nl
www.testframe.com
The Quest
The Challenges for a Test Process

- Testing should be fun
- Testing should be effective
- Testing should be efficient
- Testing should be under control
Common problems with testing

- Time consuming
- Costly
- Tends to be neglected
- Experienced as boring to do (in particular the execution)
- Hard to start in time
Even more common problems with testing

- Difficult to manage:
  - What is the progress
  - What is the quality

- The proper resources (users, specialists) are not (made) available when needed

- Lack of test design competency

- etcetera
Dimensions of Testing

- Organisation and management
- Test development
- Test automation
Re-usable test products

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>transfer</td>
<td>Houston</td>
<td>Klein</td>
<td>210</td>
</tr>
<tr>
<td>check balance</td>
<td>Klein</td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>transfer</td>
<td>Savy</td>
<td>Klein</td>
<td>150</td>
</tr>
<tr>
<td>check balance</td>
<td>Klein</td>
<td>360</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

test development

navigation scheme

test cluster

test execution

... check balance
... enter customer
The product life cycle

- test plan
  - test design
  - navigation script
    - tool
      - target system
  - report
    - actual results
      - comparison with expectations
      - management information
    - QA/Auditors
    - management
    - system development
    - end users

- test clusters
  - test conditions
  - test lines

- input data
- expected outcomes
- documentation

separation
Example Test Condition

- **nr**: description
- **3.51**: it is checked that the exit date is after the entry date

```
test condition
3.51
```

```
enter employment
name: Bill Goodfellow
entry date: 1999-10-02
exit date: 1999-10-01

check error message
The exit date must be after the entry date.
```
Independence of life cycles

system development

process oriented dependencies

test development

test automation
The Holy Grail

- Part of the legend of king arthur
- Most of all a symbol
- To find the holy grail one has to be the perfect knight
- To search for it is as important as finding it
The Holy Grail as a model

- General principles, good to “strive” for, but difficult to fully achieve

- The three principles suggested in this presentation are for test development

- Coming close to the grails will help quite a bit

- Being far away is a good recipe for trouble

- The first “victim” is the manageability
The three “holy grails” of Test Development

- effective “clustering” of tests
- choosing the right approach per cluster
- the proper level of actions
The First Grail clustering
well known, the V Model (simplified)

(requirements) → logical design → physical design → programming

 acceptance tests → system tests → subsystem tests → module tests

(inflexible) use of the V Model can involve risks

(see Brian Maricks article: www.testing.com/writings/new-models.pdf)
The “cluster model” as an alternative

• fit to project situation as an explicit step

• V Model is “member of the class”

• further decisions are per cluster:
  • how -> which technique, automate or not
  • when -> develop, execute
  • who -> stake holders, tester, auditor, ...
  • why -> business risk, complexity, made by John, ...
What is not likely to be good clustering

• combinations of low and high level tests
• when you have to change all of them for a new system release
• if all clusters look alike
• clusters are dependent on each other
• you can’t start developing them now
Typical consequences / symptoms

- complaining people
- unnecessary high test maintenance (high impact of system changes on the test)
- difficulties in running any test
- “unpleasant” test process
- no (sense of) control
Clustering recommendations

- Logical to all concerned
- Independent from other clusters
- Well differentiated and clear in scope
- Fitting the priorities and planning of the project
- Balanced in size and amount
Some angles for clustering

- business and failure risks
- planning and control
- test execution
- complexity of the test
- stake holders and others involved
- part of the system
- detail level
- quality aspect
- technique used for the test (double role)
- functional and technical requirements
Example of a (customer friendly) business risks analysis at a financial institution

if this goes wrong then...

- ... there are financial (or worse) consequences for our customer
  - all customers: must have
  - one customer: should have

- ... there are non-financial consequences for our customer
  - all customers: should have
  - one customer: could have

- ... there are financial (or worse) consequences for ourselves
  - no workaround: should have
  - workaround: nice to have

- ... there are non-financial consequences for ourselves
  - no workaround: should have
  - workaround: nice to have
The clustering process (example)
The Second Grail

approach per cluster
What is not a good approach (..)

- Complaining people
- The wrong level of techniques, like using a boundary value analysis in a high level business test
- Lots of hard labour without interesting results ("monks work")
Common Design Techniques Summary

- Equivalence Partitioning
- Boundary Value Analysis
- Output Forcing
- State Models
- Error Guessing

Reference: [Beizer, 1990] [Kit, 1996]
Testing techniques

- Functional breakdown
- Quality attributes
- Architectural breakdown

Clusters

- Joint specification development

Specifications

- Decision tables
- Graphs
- Data flow analysis
- Entity life cycle techniques

Test Conditions

- Soap Opera Testing
- Boundary value analysis
- Equivalence classes
- Joint testware development

Experts

- Joint condition development

Tests
Soap Operas

Ashley hears about Jack's deposit when he thought he had to go. Victoria lectures her father about what's wrong with him and Nikki but Victor advises her that it's none of her business. Olivia learns Dru has no regrets about leaving and takes great satisfaction in having Lily as her companion. Dru then asks Olivia why she is raking Malcolm over the coals. Stopping by Gina's, Nikki spots Brad and sits with him, admitting she doesn't want to be alone tonight. Victor stops by Mack's party at the Crimson Lights. Ashley takes a home pregnancy test. Worried about Billy, Raul makes call and J.T. claims he doesn't know where Billy is. Raul rushes over and finds Billy out cold in the snow Raul worries when he can't find a pulse. . . .
Properties of Soap Operas

- About “real life”
- But condensed
- And more extreme
Soap Operas for testing

• Define a scope of the test to develop
• Identify with the business environment
• Which elements would make things difficult
• Draft scenario’s (typical some dozen lines)
• Write them down in clusters
Examples of story lines when used for testing

Pension Fund

William starts as a metal worker for Industrial Entropy Incorporated in 1955. During his career he becomes ill, works part time, marries, divorces, marries again, gets 3 children, one of which dies, then his wife dies and he marries again and gets 2 more children….

World Wide Transaction System for an international Bank

A fish trade company in Japan makes a payment to a vendor on Iceland. It should have been a payment in Icelandic Kronur, but it was done in Yen instead. The error is discovered after 9 days and the payment is revised and corrected, however, the interest calculation (value dating)…
### Example of test lines

<table>
<thead>
<tr>
<th>from</th>
<th>to</th>
<th>amount</th>
<th>valuta</th>
<th>trans nr</th>
</tr>
</thead>
<tbody>
<tr>
<td>enter payment</td>
<td>123421344</td>
<td>4124244123</td>
<td>120000</td>
<td>yen      &amp;keep tx1</td>
</tr>
<tr>
<td>check value dating</td>
<td>&amp;tx1</td>
<td>$0.47</td>
<td></td>
<td></td>
</tr>
<tr>
<td>wait days</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>order to reverse</td>
<td>&amp;tx1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>from</th>
<th>to</th>
<th>amount</th>
<th>valuta</th>
<th>trans nr</th>
</tr>
</thead>
<tbody>
<tr>
<td>enter payment</td>
<td>123421344</td>
<td>4124244123</td>
<td>1200000000</td>
<td>IKr</td>
</tr>
<tr>
<td>check value dating</td>
<td>&amp;tx2</td>
<td>$7,701.56</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

...
Soap Operas (in testing) are not necessarily:

- “Extreme”
- Far fetched
- Long and elaborate
- Pieces of art and creativity
“Killer Soaps”

• More specifically aimed at finding hidden problems
• Run when everything else has passed
• One option: put a killer soap at the end of a normal cluster
• Ask the “specialists” for input
Test Design Templates

- Spreadsheet based technique for designing tests
- Introduced in 1998 as an extension to testframe by edward kit
- Friendly technique and yet very strong, for example in identifying needed situations and combinations
- For a further description please have a look at ed’s article in software development magazine:
  http://www.sdmagazine.com/breakrm/features/s992f2.shtml
Example of Test Design Templates

(applied to SDT’s ReviewPro package for supporting inspections and reviews)
Usage of Test Design Templates as portal to Soap Operas

<table>
<thead>
<tr>
<th>Template ID:</th>
<th>MB des 1</th>
<th>MB des 2</th>
<th>MB des 3</th>
<th>MB des 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>customer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>last name</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>first name</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>balance</td>
<td>positive</td>
<td>too low</td>
<td>positive</td>
<td>positive</td>
</tr>
<tr>
<td>number</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>confirmation letter</td>
<td>yes</td>
<td>yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>automatic numbering</td>
<td></td>
<td></td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>tested in scenario:</td>
<td>MB01</td>
<td></td>
<td>MB01, MB02</td>
<td>MB02</td>
</tr>
</tbody>
</table>
A life cycle for test development

- clustering
  - test conditions
  - test design templates
  - match (cross ref)
  - test scenario's
  - review execution maintenance...

overall

cluster level
What can joint sessions give you

- Test Strategy
- Acceptance Criteria
- Cluster Grouping
- Test conditions
- Evaluation of Results
- Starting up development of scenarios
Joint Testware Development (JTD)

- Moderator / chairman
- Users
- Business specialists
- Developers
- Testers
Set-up of a joint session for a telecom provider

• 1st session
  • Introduction by moderator and project manager
  • explanation about the JTD procedure
  • explanation of the functional area by a specialised user

• 2nd session
  • start of production of test conditions

• 3rd session
  • start of production of test scenarios

• 4th session
  • evaluation test scenarios
The Third Grail

level of actions
Composition of Action Words

• Specification of an action, a check or a documentary statement

• Communication between Navigation and Test Cases

• Consistent

• Standard

• By-product of the test analysis
What is probably not a good action level

- “test all”
- low level “navigation” in a high level test:

```plaintext
<table>
<thead>
<tr>
<th>type</th>
<th>&quot;Hans&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>press key</td>
<td>&lt;tab&gt;</td>
</tr>
<tr>
<td>type</td>
<td>&quot;Buwalda&quot;</td>
</tr>
<tr>
<td>click button</td>
<td>OK</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>
```

- unclear “insiders only” language

```plaintext
<table>
<thead>
<tr>
<th>set code</th>
<th>Fc122x</th>
<th>XX33</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Typical consequences / symptoms

• tests become quite unreadable (especially for non experts)

• unpleasant work to make the tests

• hard to understand the results

• high risk of mistakes

• and, of course, heavy maintenance dependency
Designing Action Words

- What actions does the test tool perform with a specific Action Word
- Scope of the test determines the Action Word level
- Manage the set of Action Words
- Document information about the Action Words
Low-level

- Low-level functions take care of the communication with the application
- Low-level functions are being called from high level action word functions
- Use of low-level functions enhances maintainability
- Typical examples:
  - push button
  - select list box item
Intermediate level

- Optional, use for complex navigation's

- Navigation at level of:
  - Window
  - Record
  - Message
  - ...

- Called as subroutine by high level action words
High level

• Defined by testers,
  □ (not by navigators and not in advance)

• Specification of the test, navigation:
  □ Adds default values
  □ Moves across windows
  □ Takes care of unexpected situations
  □ ...

• Uses low level and intermediate level
The life cycle

- high level business oriented tests
- production acceptance tests

- functional tests
- technical tests

- low level functional tests
- technical tests

system development
 specifications
 design
 programming

Test Execution

navigation

high level actions

intermediate level actions

low level actions
Acknowledgements

- Edward Kit
- Dorothy Graham
- Brian Marick
- Dirk van Dael
- Erik Jansen
- Dennis Janssen
References

The Three Holy Grails of Test Development
Hans Buwalda

Hans Buwalda is project director at CMG, a leading European information technology services group. He is responsible for new developments around the TestFrame approach for testing and test automation of which he is the main architect. In 1996 he presented the main ideas for the first time to an international audience in a speech called 'Testing with Action Words, abandoning record and playback'. Since then the method is being used in an increasing number of countries and Hans has become a frequent speaker at industry conference, tutorials and workshops.