

“A Web Testing Experience”

Jarle Våga
TietoEnator Consulting AS
Oslo, Norway



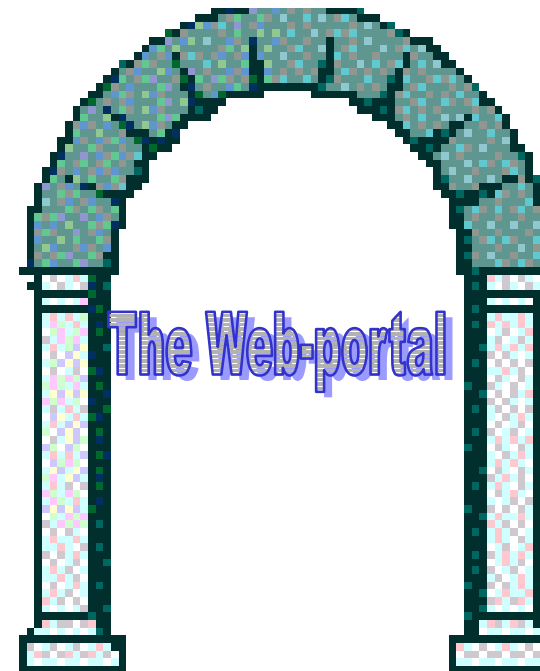
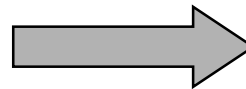
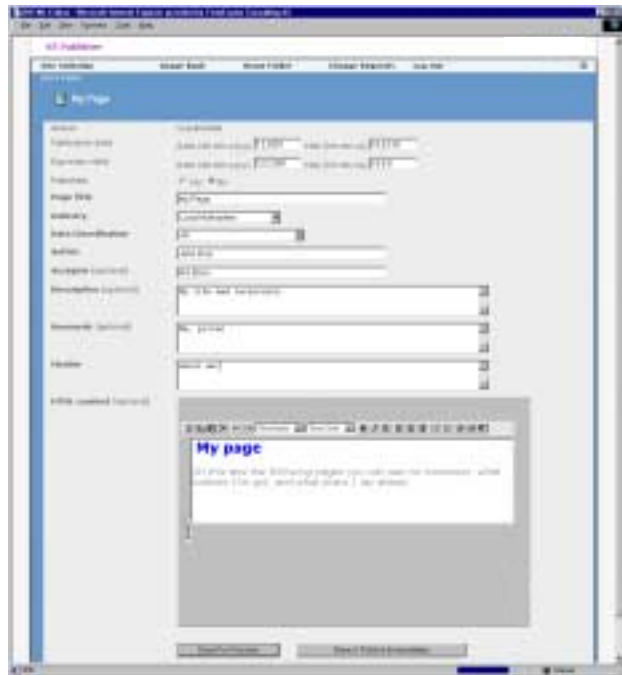
Presentation Outline

- **Intro**
 - About the system
 - Methods
 - Preparation
- **Test Planning**
- **Test execution**
- **“The web testing experience...”**
- **Exploratory pair testing**

“It’s on the Web. Go find the bugs before the client does!”

“One company - One message”

Basic functionality



- “Content Creator” creates pages
- “Topic Manager” have to approve them before they are published in the portal

Planning ahead

- **Documentation**
 - Difficult to obtain system documentation
 - Not prioritised by developer
- **Changes on the production site**
 - Constant changes during planning period
- **Browser difficulties**
 - Not possible to downgrade Microsoft software, or to install two versions of Internet Explorer on the same PC



What to do...

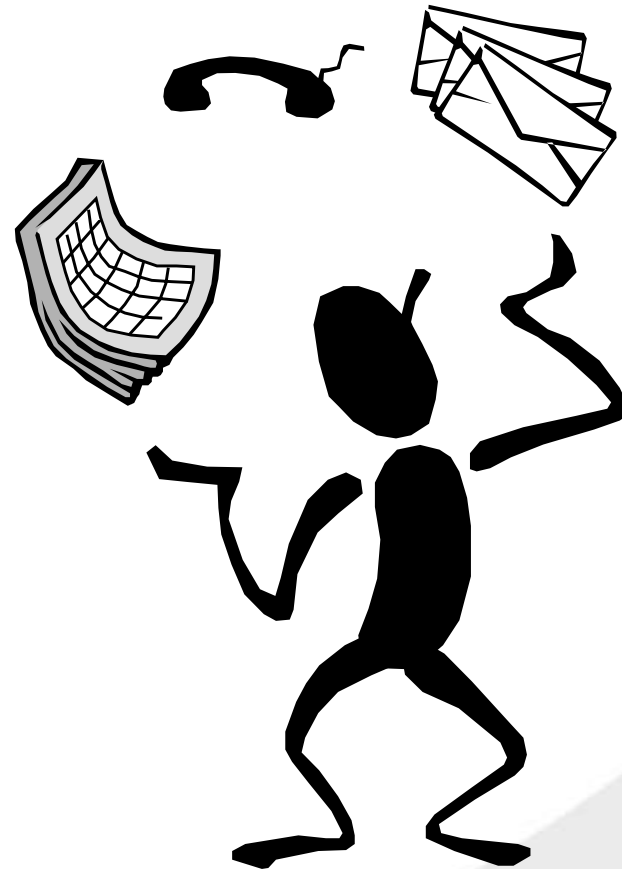
- **Features to be tested:**
 - Functional test
 - Yellow pages (search facilities)
 - Browser test (ie 4.0 + / netscape 4.0 +)
 - Win NT/ Win 98/ Win 2000

- **...would this be enough? And how should it be done?**



Methods

- **Test scripts**
- **Pair testing**
- **Exploratory testing**
- **Coordinating:**
 - 20 pc's
 - 5 browsers
 - 3 Operating systems
 - 12 people
 - 2 user groups simultaneously



Methods – Exploratory Pair testing

- **Two people working at the same computer**
- **One works the PC, the other makes creative suggestions and write deviation-reports**
- **No plans or scripts up front, based on the testers creativity**
- **Enhances confidence among testers, as discussing results while working is easier**



The test plan

- **Features to be tested**
 - Important to be precise, this helps focusing the pair testing
- **Features not to be tested**
 - Just as important, used to limit the test
- **Approach**
 - Preparations and requirements
 - No specific test script (but rather “guidelines”)
 - Use of role-play and test matrixes
- **Item pass/ fail criteria**
 - ...in order to judge the outcome of the test.

Executing the test

- **Major difficulties with the set-up of the test environment**
 - Browsers not installed
 - Win2000 pc's not available
- **Functionality test according to guidelines**
 - Creation of pages in the portal using role-play with pairs (Content Creator/Topic Manager)
 - Using a matrix, where roles, OS and browsers were integrated.
- **Browser and op.sys. test**
 - matrix-script with 5 browsers and 3 op.sys.
- **Yellow pages (search functionality)**
 - re-use of pages published
- **Exploratory test**
 - what happens if...

Executing - The Matrix

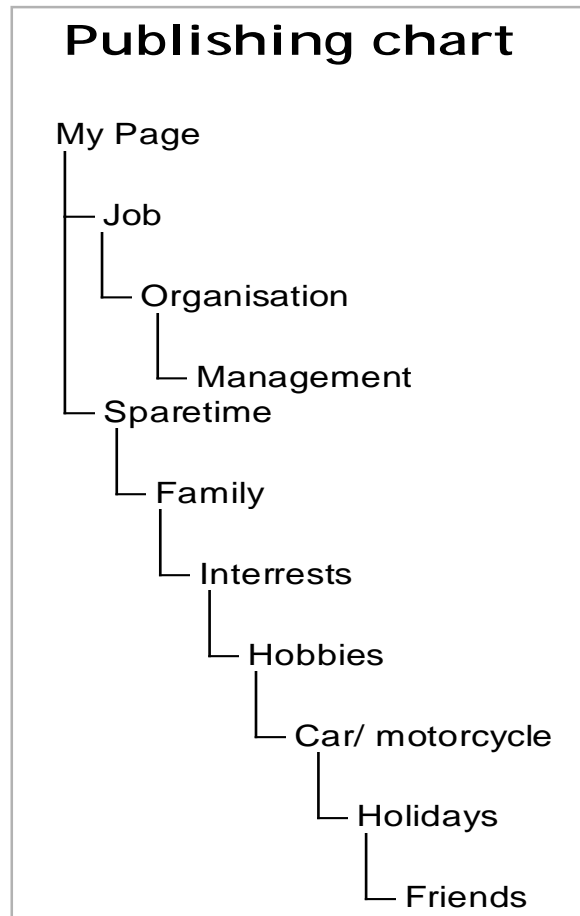
Group #		Operating System		
Browser		W-NT	W98	W2000
	Netscape 4	6	1	
	Netscape 6	3	4/6	5
	Iexplore 4	1		
	Iexplore 5	4/5	2/3	
	Iexplore 5.5	2	5	6
Time	Day 1	10:00-12:00	13:00-15:00	15:00-16:30
	Day 2	09:00-17:00	09:00-17:00	13:00-16:00

The upper matrix indicate how the groups worked on different combinations of browsers and OS'. The lower how the groups were divided into different user groups at the start of the test, for each change of operating system they were alternated.

I.e. Group 3 started as Content Creator working on NT with Netscape 6, and moved to W98 with Explorer 5 after lunch, working as Topic Manager. Different "test guidelines" was developed for each combination, which gave us $OS * Browser = 3 * 5 = 15$ different guidelines.

User group	Groups					
	1	2	3	4	5	6
Content Creator	•		•		•	
Topic Manager		•		•		•

Executing - Publishing guideline



The Publishing chart shows the different pages to be published (and approved by Topic Manager), and the structure and level in which they should be published.

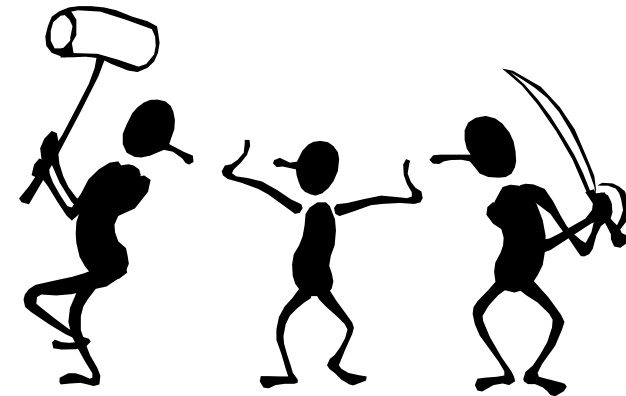
These pages were reused during test of search functionality

Executing – Exploratory pair test

- **Plan – increase coverage by using the testers creativity**
- **...which is what happened**
- **It can be difficult to keep testers “on track”**
- **...but easy to motivate, and many bugs found.**

Acceptance Test Summary

- **150-200 deviations (incl. Some duplicates)**
 - Mainly due to lack of limitations in the system, you could do “anything”.
 - Developer accepted only ca. 60
- **Corrupted pages**
- **Web portal made inaccessible**
 - Menu disappeared when using signs (“#*\’/@ etc”) in page title
- **Customer: Severe safety risk**
- **Developer: Tester have misused the system**



Summary

- **The portal was not recommended for a pilot after the test.**
- **Using pair- and exploratory methods proved successful**



...the web testing experience...

- **Murphy's law**
- **No definite specifications of the system caused a problem**
- **But: An entire test-project was done very rapidly**
 - 1-2 test coordinators.
 - 5-10 days of preparation (Get to know the system, and set up a test plan)
 - 2-3 days of testing
 - ca. 1-2 days to write the Test Summary Report
 - Allow time for bugs to be fixed and re-test
- **Small time frame and immediate feed-back to developers made it a low-cost alternative**

...Better ways than the "right way" ...

- **Why traditional methodology "stinks"...**
- **The v-model might be "your worst enemy"**
- **Retesting: script vs. Minefield**
- **Scripting, Minefield, module allocation, checklists...**
- **88% of bugs found during creation of the scripts**

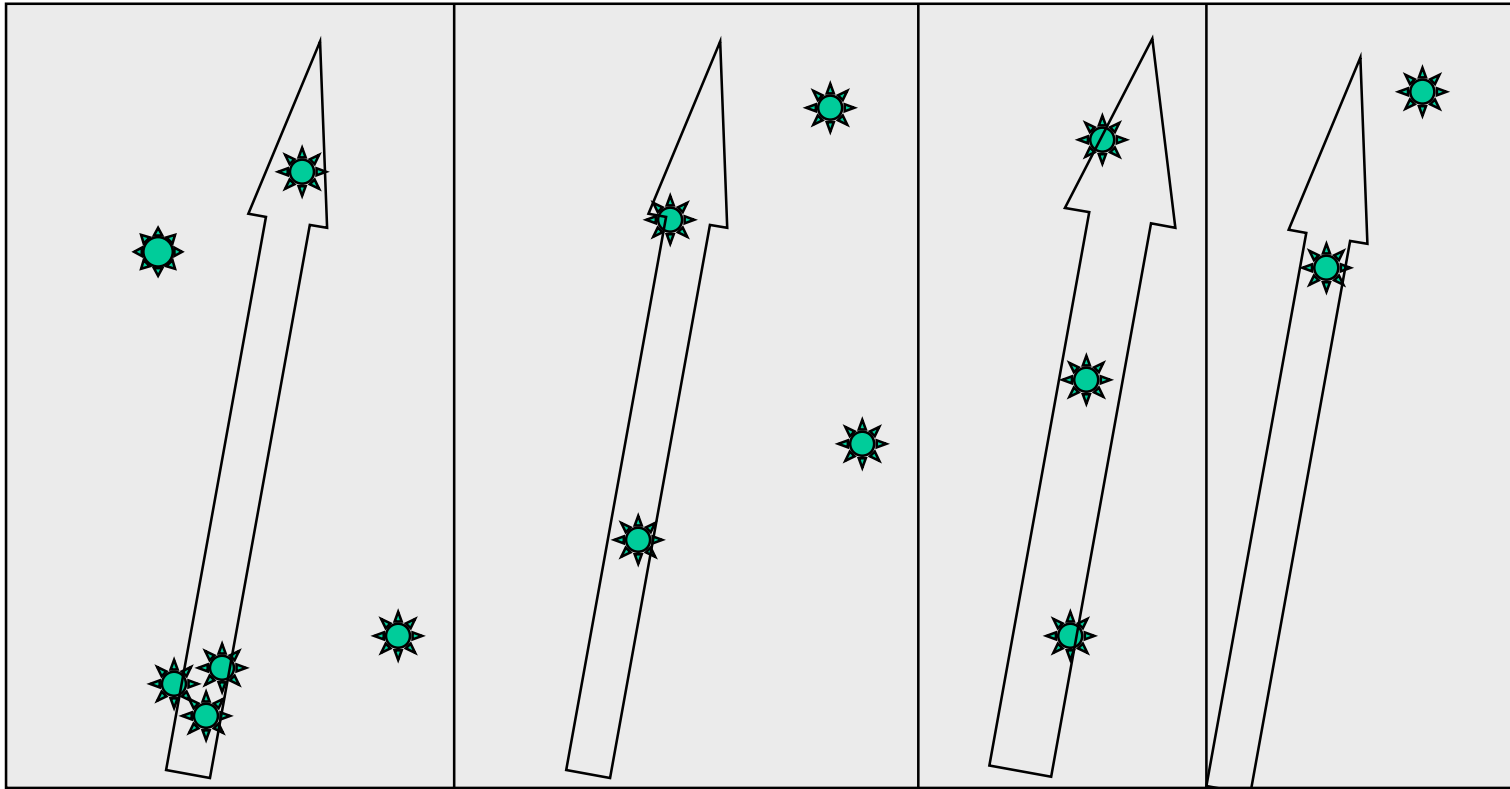
Exploratory Testing

- **Simultaneous learning, planning and testing**
- **Learning while testing, testing while learning**
- **Documenting while testing**
- **”Good enough” is good enough**
- **Customer focused, risk-focused**
- **Responsive to changing circumstances**
- **Keep mission clearly in mind**
- **Skilled work, not random**

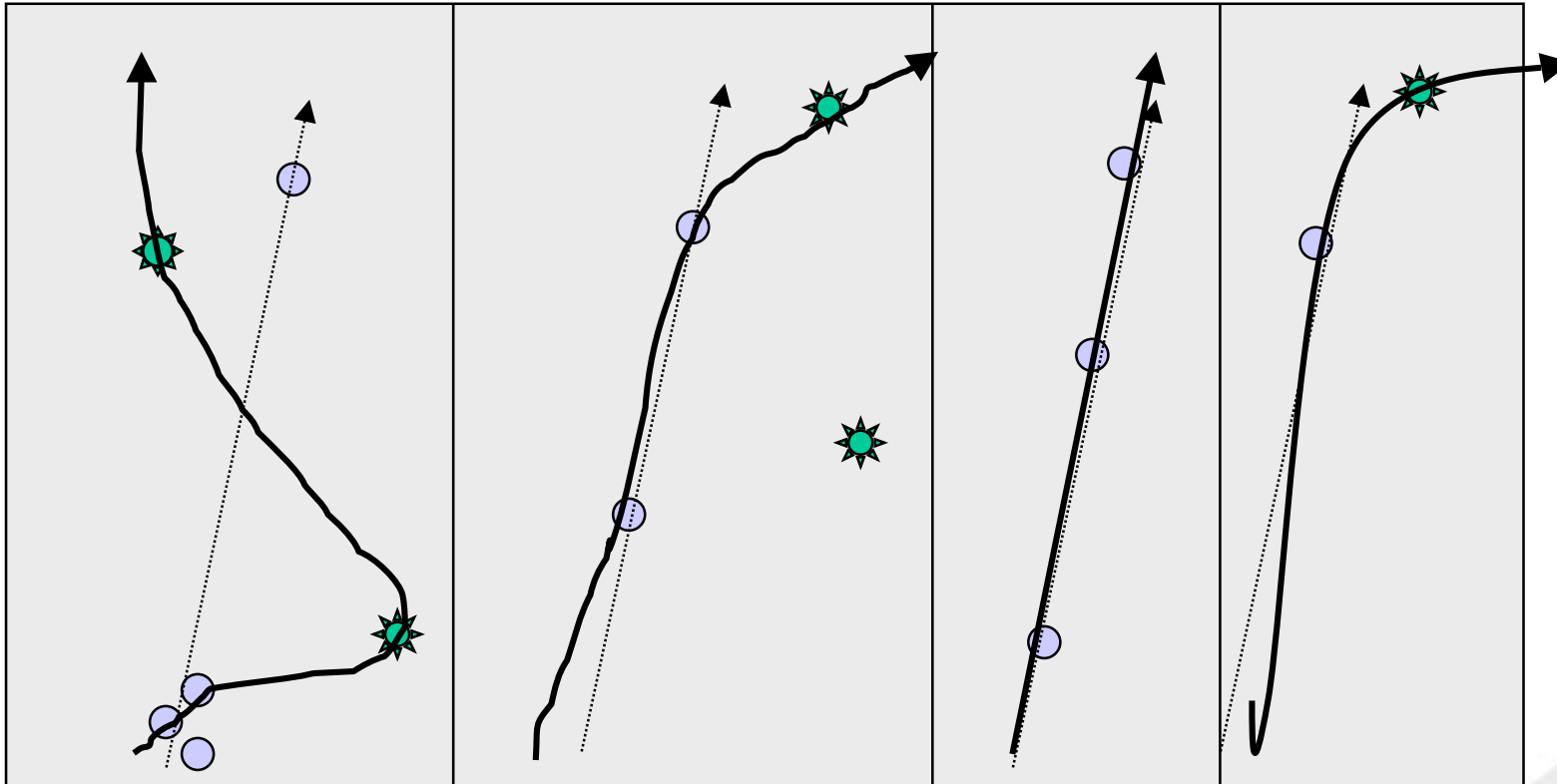
Exploratory Pair Testing

- **Similarities with XP**
- **Two testers on one machine**
- **60-90 minutes sessions (ref. session based test management)**
- **Operate from a detailed project outline, or create outline for each session**
- **High productivity, high creativity**
- **Increased learning and reporting**
- **Testing is an "idea generation activity"**

Methodology - Script



Methodology - minefield



...the web testing experience (2)... Pair testing

- **Pair testing**

- Speeded up reporting errors
- Excellent to promote creativity
- Encouraged discussion
- Demanded a “strong voice” to keep everyone focused!
- Enables effective testing of short-lived, intense projects...
- but has also proved very useful on many test projects, also larger (i.e. at Microsoft, ref. J.Bach)

...the web testing experience (3)... Exploratory test

- **Exploratory testing**

- Requires skilled testers
- Requires some pre-set restraints, and thorough planning
- Requires a system for briefing and reporting/ documenting (ref. J.Bach - “Session based test management”)
- Requires guidance from test manager
- ..but when the requirements are fulfilled, it proved to be very successful.
- Easy to motivate
- MANY deviations found
- ...and it gave testers a greater feeling of participation

References

- <http://www.satisfice.com/>
- <http://www.kaner.com/>
- <http://www.stickyminds.com/>