

Managing Exploratory Testing



Parts of this presentation have been excerpted from:
Rapid Software Testing, by James Bach, james@satisfice.com
Black Box Software Testing, by Cem Kaner, kaner@kaner.com

Ståle Amland, Amland Consulting,
Hulda Garborgsv. 2, 4020 STAVANGER, NORWAY
Mob:+47 905 28 930, Phone: +47 51580587, Fax: +4751585524
e-mail: stale@amland.no WEB: www.amland.no

Software Testing:

“There is artistry. There is craft. There is science. It is all three – as with all technologies.

If you understand the science and have mastered the craft, then it is meaningful to investigate, espouse, and perfect the art.”

Boris Beizer

Software Testing:

“Testing is the process of executing a program with the intent of finding errors.”

“Testing is an extremely creative and intellectually challenging task.”

*Glenford Myers in
“The Art of Software Testing”*

Introductory Note

- Parts of this presentation have been excerpted from:
 - Rapid Software Testing, by James Bach, james@satisfice.com
 - Black Box Software Testing, by Cem Kaner, kaner@kaner.com

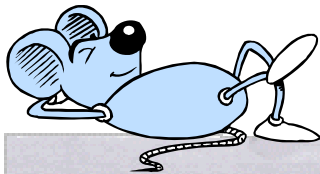
- Here is what Cem Kaner says about this class:

I'm pleased to see that Stale Amland is teaching a course on exploratory testing. Stale's a sharp guy, and has written some very useful papers. From speaking with him and seeing his work, I think he'll develop an excellent course on exploratory testing.

-- Cem Kaner



Presentation Outline



What is Exploratory Testing?

How to do Exploratory Testing?

How to Manage Exploratory Testing?

Different testing approaches

- Skeptical approaches
- Analytical Approaches
- Information-driven approaches
- Time-honored but less effective approaches
- Experiential and intuitive approaches:
 - “Let’s think blue-sky, speculate and follow our intuition.”
 - “We have good hunches about where the bugs are lurking.”
 - “Let’s jump in and explore the system’s behavior hands-on, so we can decide how to test it.”
 - “Let’s find the important bugs fast, and worry about the test paperwork later.”

What is Exploratory Testing?

"Exploratory testing involves simultaneously learning, planning, running tests, and reporting / troubleshooting results."

Dr. Cem Kaner (2001)

"Exploratory testing is an interactive process of concurrent product exploration, test design and test execution."

"To the extent that the next test we do is influenced by the result of the last test we did, we are doing exploratory testing."

James Bach, Satisfice (2001)

ET vs. Scripted Testing



Jarle Våga (2002)

Sample Product Test Cycle

- 1. Receive the product.
 - Formal builds
 - Informal builds
 - Save old builds.
- 2. Clean your system.
 - Completely uninstall earlier builds.
- 3. Verify testability.
 - Smoke testing
 - Suspend test cycle if the product is untestable.
- 4. Determine what is new or changed.
 - Change log
- 5. Determine what has been fixed.
 - Bug tracking system
- 6. Test fixes.
 - Many fixes fail!
 - Also test nearby functionality.
- 7. Test new or changed areas.
 - Exploratory testing.
- 8. Perform regression testing.
 - Not performed for an incremental cycle.
 - Automated vs. manual
 - Important tests first!
- 9. Report results.
 - Coverage
 - Observations
 - Bug status (new, existing, reopened, closed)
 - Assessment of quality
 - Assessment of testability

From Rapid Software Testing, copyright © 1996-2002 James Bach

Exploratory Testing is Useful When...

- There is little or no specifications and / or requirements
- You have little or no domain knowledge
- You don't have time to specify, script and test

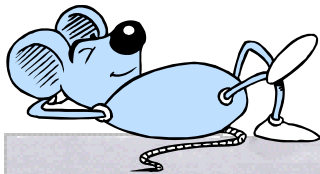
Uncertainty and Time Pressure!

When to use Exploratory Testing?

- A common goal of exploration is to *probe* for weak *areas* of the program.
- Test team's resource consumption per week:
 - 25% of the group's time developing new tests
 - 50% executing old tests (including bug regression)
 - 25% on exploratory testing



Presentation Outline



What is Exploratory Testing?

How to do Exploratory Testing?

How to Manage Exploratory Testing?

Doing Exploratory Testing


- Keep your mission clearly in mind.
- Keep notes that help you report what you did, why you did it, and support your assessment of product quality.
- Keep track of questions and issues raised in your exploration.
- To supercharge your testing, pair up with another tester and test the same thing on the same computer at the same time.



From Rapid Software Testing, copyright © 1996-2002 James Bach

Exploratory Testing Tasks

	Explore	Design Tests	Execute Tests
Product <i>(coverage)</i>	Discover the elements of the product.	Decide which elements to test.	Observe product behavior.
Quality <i>(evaluation)</i>	Discover how the product should work.	Speculate about possible quality problems.	Evaluate behavior against expectations.
Techniques	Discover test design techniques that can be used.	Select & apply test design techniques.	Configure & operate the product.



Testing notes	Tests	Problems Found
----------------------	--------------	-----------------------

From Rapid Software Testing, copyright © 1996-2002 James Bach

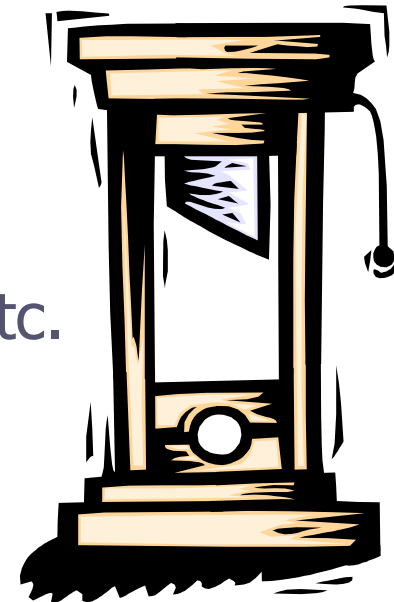
Taking Notes

- Test Coverage Outline/Matrix
- Evaluation Notes
- Risk/Strategy List
- Test Execution Log
- Issues, Questions & Anomalies
 - It would be easier to test if you changed/added...
 - How does ... work?
 - Is this important to test? How should I test it?
 - I saw something strange...

From Rapid Software Testing, copyright © 1996-2002 James Bach

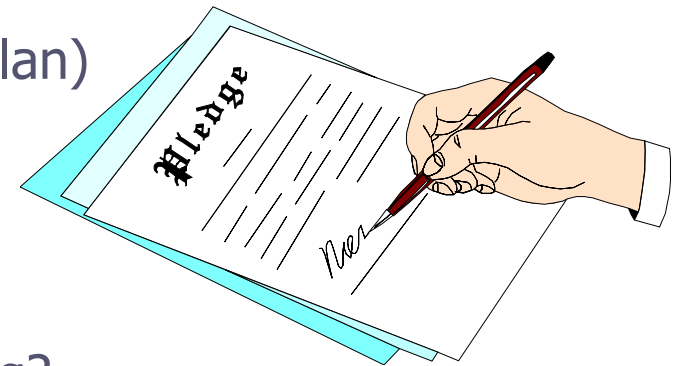
ET Execution - Sessions

- Session
 - Basic testing work unit
 - Uninterrupted
 - e-mail, meetings, telephone calls etc.
 - Reviewable
 - A report should be produced
 - Chartered
 - Mission associated with this session; What are we testing? What are we looking for?



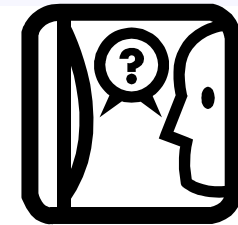
ET Documentation

- Planning:
 - Charter – overview of what to test (plan)
 - Might be a flip chart on the wall
 - Mission – What are we looking for?
- Execution
 - Notes – what happened during testing?
 - What did I do? Why did I do it?
Used to assess product quality after test.
 - Data files – input data used for testing
 - Bug reports – enough details to recreate the test / bug
 - Track questions and Issues



Approaches / Styles of ET

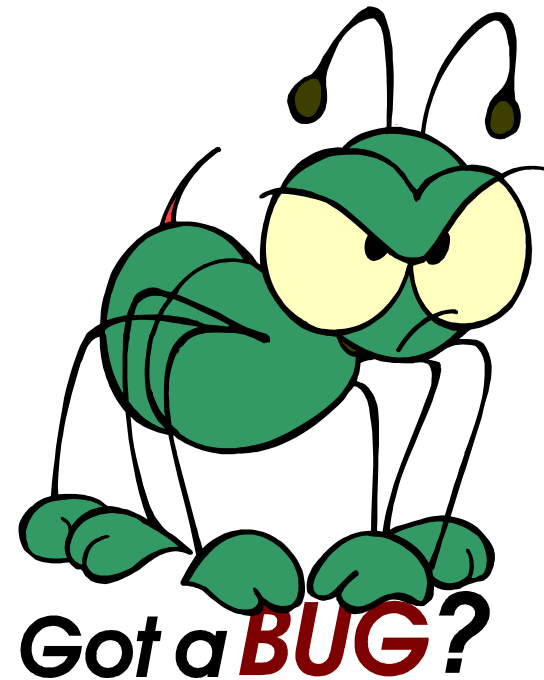
- At the heart of all ET styles:
 - *Questions and Questioning Skills*
- Characterize the styles with respect to each other:
 - Do they focus on:
 - Method of questioning?
 - Method of describing or analysing the product?
 - The details of the product?
 - The patterns of use of the product?
 - The environment in which the product is run?
 - To what extent would this style benefit from group interaction?



From Black Box Software Testing, copyright © 1996 – 2002 Cem Kaner

Styles of Exploration

- Hunches
- Models
- Examples
- Invariances
- Interference
- Error Handling
- Troubleshooting
- Group Insight
- Specifications



From Black Box Software Testing, copyright © 1996 – 2002 Cem Kaner

Howto...

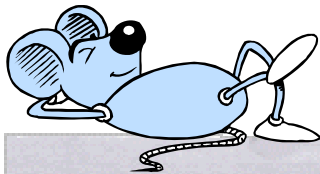
Summary

The object of your mission is to explore the Missouri river, & such principal streams of it, as, by its course and communication with the waters of the Pacific ocean...may offer the most direct & practicable water communication across this continent for the purposes of commerce.

- Thomas Jefferson's letter to Meriwether Lewis, June 1803



Presentation Outline



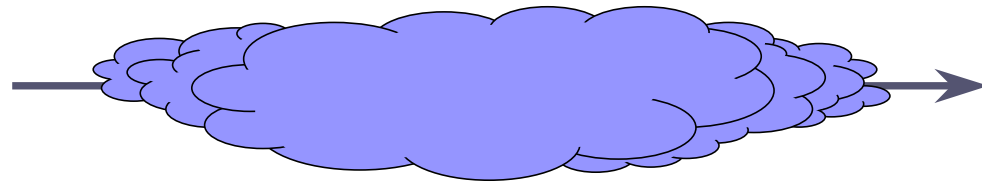
What is Exploratory Testing?

How to do Exploratory Testing?

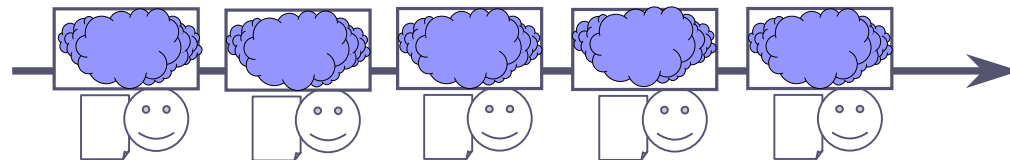
How to Manage Exploratory Testing?

The Test Session

- Charter
- Time Box
- Reviewable Result
- Debriefing



VS.



From Rapid Software Testing, copyright © 1996-2002 James Bach

Charter: A clear mission for the session

- A charter may suggest what should be tested, how it should be tested, and what problems to look for.
- A charter is not meant to be a detailed plan.
- General charters may be necessary at first:
 - “Analyze the Insert Picture function”
- Specific charters provide better focus, but take more effort to design:
 - “Test clip art insertion. Focus on stress and flow techniques, and make sure to insert into a variety of documents. We’re concerned about resource leaks or anything else that might degrade performance over time.”

From Rapid Software Testing, copyright © 1996-2002 James Bach

Time Box: Focused test effort of fixed duration

Short: 60 minutes (+-15)

Normal: 90 minutes (+-15)

Long: 120 minutes (+-15)

- Brief enough for accurate reporting.
- Brief enough to allow flexible scheduling.
- Brief enough to allow course correction.
- Long enough to get solid testing done.
- Long enough for efficient debriefings.
- Beware of overly precise timing.

From Rapid Software Testing, copyright © 1996-2002 James Bach

Debriefing: Measurement begins with observation

- The manager reviews session sheet to assure that he understands it and that it follows the protocol.
- The tester answers any questions.
- Session metrics are checked.
- Charter may be adjusted.
- Session may be extended.
- New sessions may be chartered.
- Coaching happens.

From Rapid Software Testing, copyright © 1996-2002 James Bach

Reviewable Result: The session sheet

- Charter
 - #AREAS
- Start Time
- Tester Name(s)
- Breakdown
 - #DURATION
 - #TEST DESIGN AND EXECUTION
 - #BUG INVESTIGATION AND REPORTING
 - #SESSION SETUP
 - #CHARTER/OPPORTUNITY
- Data Files
- Test Notes
- Bugs
 - #BUG
- Issues
 - #ISSUE

```
CHARTER
-----
Analyze MapMaker's View menu functionality and
report on areas of potential risk.

#AREAS
OS | Windows 2000
Menu | View
Strategy | Function Testing
Strategy | Functional Analysis

START
-----
5/30/00 03:20 pm

TESTER
-----
Jonathan Bach

TASK BREAKDOWN
-----

#DURATION
short

#TEST DESIGN AND EXECUTION
65

#BUG INVESTIGATION AND REPORTING
25

#SESSION SETUP
20
```

From Rapid Software Testing, copyright © 1996-2002 James Bach



Summary Exploratory Test Management

- Exploratory Testing requires skills and experience
- Be prepared – as good as you can!
- Take notes
- Test in Sessions
- Track Sessions



Learn more...

- Exploratory Testing
James Bach at Grove Consulting Event
 - Seminar and Workshop
 - London, UK, July 24, 25 and 26
 - Booking:
 - www.grove.co.uk
 - courses@grove.co.uk



Learn more...

- Exploratory Testing:
 - Rapid Software Testing, by James Bach, james@satisfice.com
 - Black Box Software Testing, by Cem Kaner, kaner@kaner.com
 - Amland Consulting, www.amland.no (training)
- People Issues, test techniques, inspections:
 - Grove Consultants, www.grove.co.uk
 - + many, many more



References/Bibliography

- Argus, C. & Johnson, B., Ad Hoc Software Testing; Exploring the Controversy of Unstructured Testing, http://www.testingcraft.com/ad_hoc_testing.pdf
- Bach, James, 1999a. Risk-Based Testing. How to conduct heuristic risk analysis, Software Testing & Quality Engineering Magazine, November/December 1999, vol. 1, issue 6, <http://www.stqemagazine.com>. See also www.satisfice.com
- Bach, James , 1999b. What is Exploratory Testing? Satisfice.com
- Bach, Jonathan, 2000, Session Based Test Management, STQE Magazine, 11/00, www.satisfice.com
- Beizer, Boris, Software Testing Techniques, Van Nostrand Reinhold. NY. 1990.
- Collard, Ross, 2002, to be published, rcollar@attglobal.net
- Kaner, C., Falk, Nguyen, 1999, Testing Computer Software, John Wiley & Sons
- Kaner, C., 2000. Rethinking Software Metrics, Evaluating measurement schemes, Software Testing & Quality Engineering Magazine, March/April 2000, vol. 2, issue 2.
- Kaner, C., 2001a, Black-Box Software Testing, training at Satisfice July 2001.
- Kaner, C., Bach, J., Pettichord, B., 2001b, Lessons Learned in Software Testing, John Wiley & Sons; <http://www.testinglearnings.com/>, ISBN: 0471081124
- Kaner, C., Bach, J. 2001c, Exploratory Testing in Pairs, presentation at STAREast, Orlando, FL, www.kaner.com
- Våga, J., Amland, S., 2002. High Speed Web Testing, contribution to the book " Software Quality and Testing in Internet Times", Editor: Dirk Meyerhoff et al, Springer, 200

Contact Details



Ståle Amland,
Amland Consulting
Hulda Garborgsv. 2,
N-4020 STAVANGER
Norway

Mob: +47 905 28 930

Phone: +47 51 58 05 87

Fax: +47 51 58 55 24

e-mail: stale@amland.no

WEB: www.amland.no