

Getting the Most Out of Exploratory Testing

James Bach, Satisfice, Inc.

<http://www.satisfice.com>

James@satisfice.com

Acknowledgements

- Several of the slides in this presentation were developed by or in collaboration with **Cem Kaner**, who coined the term “exploratory testing” in 1984.
- Many of the ideas in this presentation were reviewed and extended by my colleagues at the 7th Los Altos Workshop on Software Testing. I appreciate the assistance of the LAWST 7 attendees: Cem Kaner, Brian Lawrence, III, Jack Falk, Drew Pritsker, Jim Bampos, Bob Johnson, Doug Hoffman, Chris Agruss, Dave Gelperin, Melora Svoboda, Jeff Payne, James Tierney, Hung Nguyen, Harry Robinson, Elisabeth Hendrickson, Noel Nyman, Bret Pettichord, & Rodney Wilson.

Contrasting Approaches

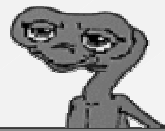
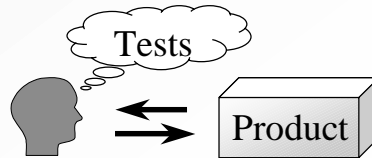
In *scripted* testing, tests are first designed and recorded. Then they may be executed at some later time or by a different tester.



(yes, this is a simplified view)

vs.

In *exploratory* testing, tests are designed and executed at the same time, and they often are not recorded.



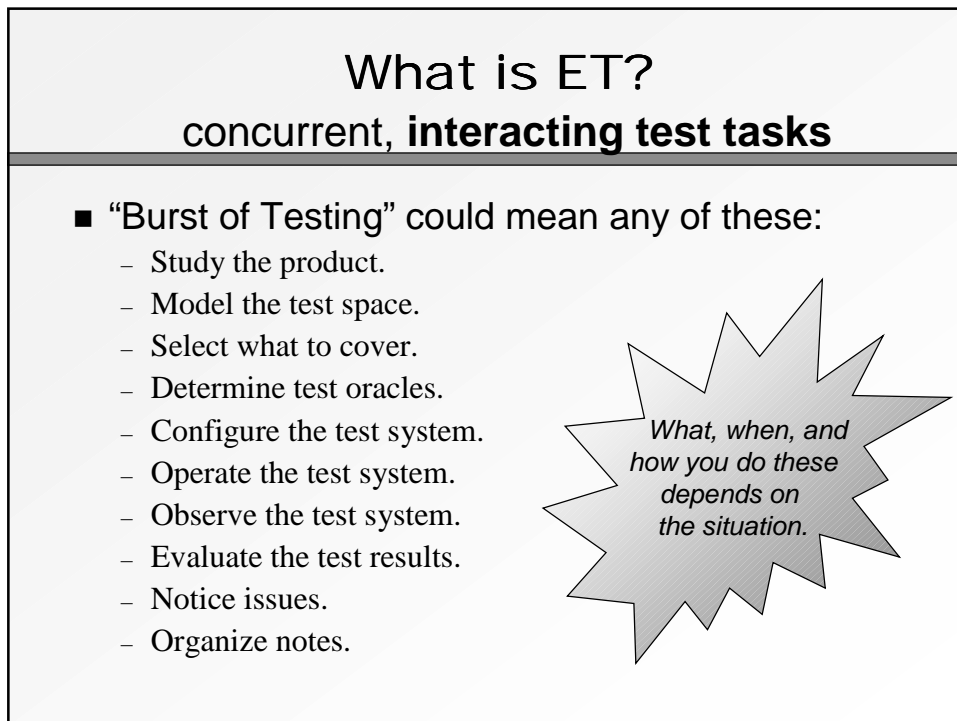
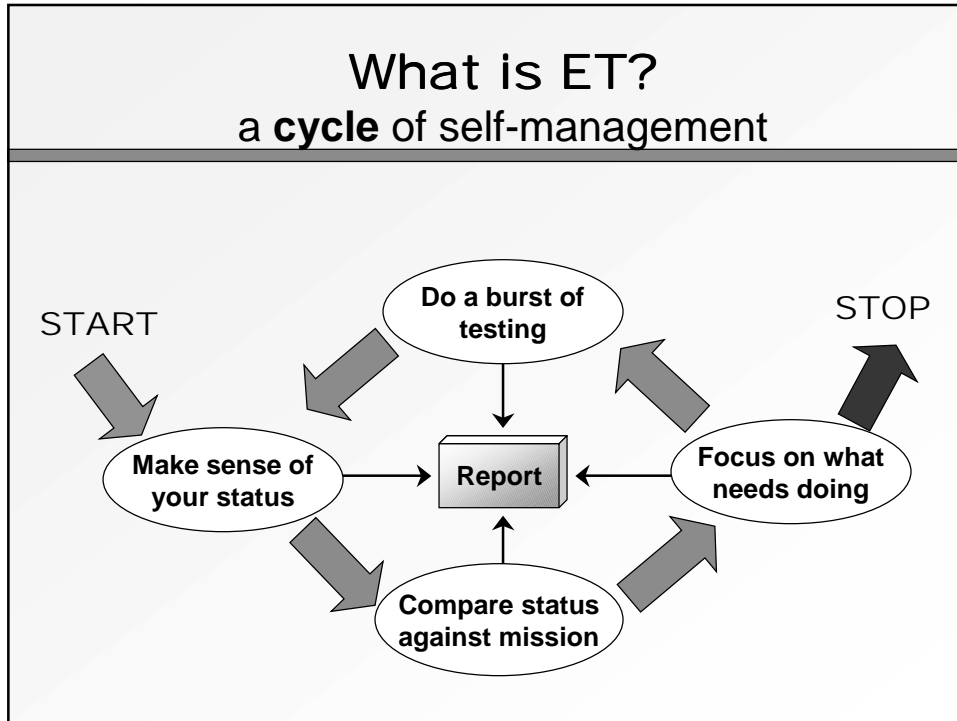
What is ET Good For?

Exploratory testing is useful when...

- it is not obvious what the next test should be.

OR

- we want to go beyond the obvious tests.



What is ET? the core practice of a **skilled tester**



■ Excellent exploratory testers...

- challenge constraints and negotiate their mission.
- alert their clients to project issues that prevent good testing.
- spontaneously coordinate and collaborate, as needed.
- train their minds to be cautious, curious, and critical.
- know how to design questions and experiments.
- know the difference between observation and inference.
- have developed resources and tools to enhance performance.
- adapt their test strategy to fit the situation.
- tolerate substantial ambiguity, indecision, and time pressure.
- take notes and report results in a useful and compelling way.
- have earned the trust placed in them.

Getting the Most Out of ET

- Augment ET with scripted tests and automation.
 - Usually it's best to use a diversified strategy.
- Exploit inconsistency.
 - Let yourself be distracted by anomalies and new ideas.
 - Avoid repeating the exact same test twice.
- Exploit the human factor.
 - Encourage variability among testers.
 - Exploit subject matter expertise.
 - Use your confusion as a resource.
 - Work in pairs or groups, whenever possible.
 - Get to know your developers.
 - Test in short bursts.

Getting the Most Out of ET

- Learn the logic of testing
 - conjecture and refutation
 - abductive inference
 - correlation and causality
 - design of experiments
 - forward, backward, and lateral thinking
 - biases, heuristics, and human error
 - risk, benefit, and the meaning of “good enough”
- Practice critical reading and interviewing
 - Analyzing natural language specifications
 - Analyzing and cross-examining a developer’s explanation.

Getting the Most Out of ET Exploratory questions

- How well does your car work?
- How do you know how well your car works?
- What evidence do you have about how your car works?
- Is that evidence reliable and up to date?
- What does it mean for your car to “work”?
- What facts would cause you to believe that your car *doesn’t* work?
- In what ways could your car *not* work, yet seem to you that it does?
- In what ways could your car work, yet seem to you that it *doesn’t*?
- What might cause your car not to work well (or at all)?
- What would cause you to suspect that your car will *soon* stop working?
- Do other drivers operate your car? How does it work for them?
- How important is it for your car to work?
- Are you qualified to answer these questions? Is anyone else qualified?

Getting the Most Out of ET

- **Adopt a clear and consistent testing vocabulary**
 - bug, specification, risk, test strategy, test case, test plan
- **Learn to model a product rapidly**
 - flowcharting; data flows; state model
 - matrices and outlines
 - function/data square
 - study the technology
- **Use a “grid search” strategy to control coverage**
 - Model the product in some way, then specify broad test areas in terms of that model (not specific test cases). Keep track of what areas you have and have not tested in.

Getting the Most Out of ET

- **Learn to take reviewable notes**
 - Take concise notes so that they don't interrupt your work.
 - Record at least what your strategy was, what you tested, what problems you found, and what issues you have.
- **Practice responding to scrutiny**
 - Why did you test that?
 - What was your strategy?
 - How do you know your strategy was worthwhile?
- **Learn to spot obstacles to good testing**
 - Not enough information, yet.
 - The product isn't testable enough.
 - Don't have enough of the right test data.

Getting the Most Out of ET

- **Develop and use testing heuristics**
 - Try the heuristic test strategy model, on my web site.
 - Practice critiquing test techniques and rules of thumb.
 - Notice the test ideas you use that lead to interesting results.
- **If you're in a highly structured environment, consider using session-based test management to measure and control exploratory testing.**
 - Packages ET into chunks that can be tracked and measured.
 - Protects the intuitive process; gives bosses what they want.
 - See *How to Measure Ad Hoc Testing* on Friday.
 - See *Session-based Test Management* article in STQE.

Books to Help You Be an ET Savant

- *Tools for Critical Thinking*, David Levy
- *Rethinking Systems Analysis and Design*, Gerald M. Weinberg
- *Exploring Requirements*, Don Gause & Gerald M. Weinberg
- *Lateral Thinking*, Edward De Bono
- *Abductive Inference*, John R. Josephson
- *Proofs and Refutations*, Imre Lakatos
- *Conjectures and Refutations*, Karl Popper
- *Explanation Patterns*, Roger S. Schank
- *The Sciences of the Artificial*, Herbert Simon
- *Sensemaking in Organizations*, Karl E. Weick
- *The Social Life of Information*, John Seely Brown & Paul Duguid