Where Do You Want To Go Today?
No More Exploratory Testing

http://www.developsense.com

Where Do You Want To Go Today?
No More Exploratory Testing

Wha..?

The Motive for This Talk

The meaning of “testing” was stolen decades ago.
It’s time to steal it back.

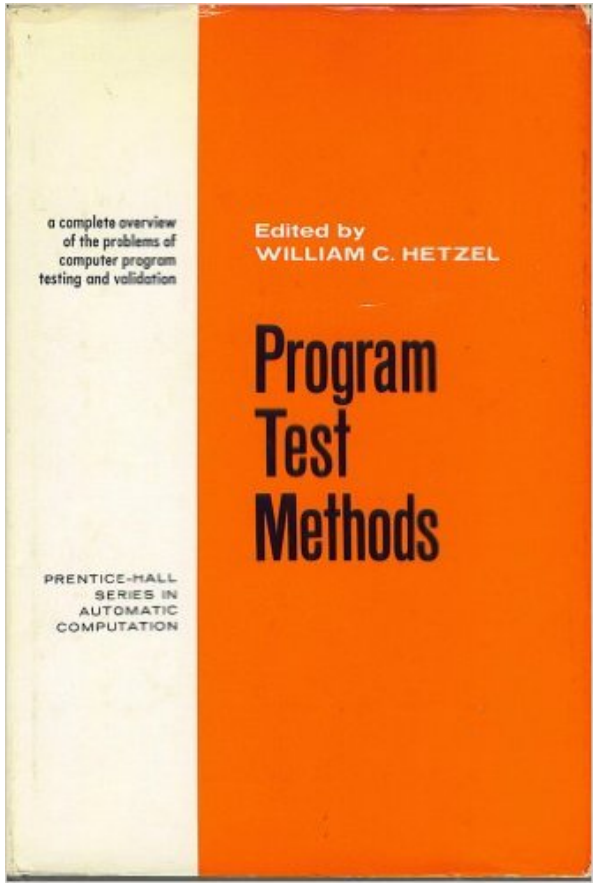
This Talk is About the Arc of Ideas

- How easy it is for design flaws to hang around in the program
- How long it takes to change the world, despite electronic media (media accelerate bad ideas as quickly as good—use a chart for this)
- Bad idea" is relative to some person, at some time
- Figuring out what we mean is intertwingled with figuring out what we're doing.

Disclaimer: This talk is to a great degree a personal memoir, and should not be considered authoritative in any historical sense.

In The Beginning, There Was Testing.

1972: Program Test Methods



Testing became confused with the *artifacts* of testing.

Lo, here be testing!

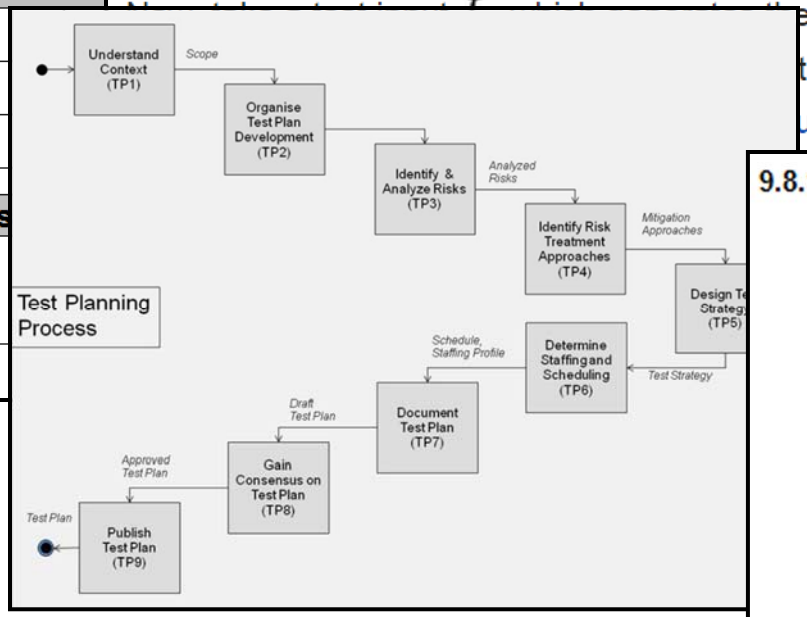
BesterTester 2.xls [Compatibility Mode] - Microsoft Excel

File Home Insert Page Layout Formulas Data Review View Developer Acrobat

D6 Test Objectives

	A	B	C	D	E	F
6	**Sr. No.**	**Test Case ID**	**Feature/ Functionality ID**	**Test Objectives**	**Test Steps**	**Expected Beha**
8	**MyAccount Logged In Page**					
9	1	ST1.1	MyAccount Page	To test for appearance of "Cart Credit Report" Link in MyAccount Logged In page for Users for whom privilege flag	User logs in to MyCarts.com site and navigates to MyAccount Home page	"Cart Credit Report" Link shou MyAccount Logged In page onl whom privilege flag is enabled
10	2	ST				
11						
12	**Accounts T**					
13						
14						
15						
16						
17	**Trans**					
18						

Treating the requirement as axioms, testability can be treated via asserting existence of a function F_S (software $F_S : I \rightarrow O$. Therefore, the ideal software generates the tuple (I_k, O_k) which is the input-output set $\sum$, sta output O_t, that is the test tuple $\tau = (I_t, O_t)$. Now, the questio the test input. Therefore, it is of imperative importance to figure out unction for the specification set $\sum$.



9.8.1 To verify Power Accuracy

9.8.1.1 Connect the components according to the *General Setup* document

9.8.1.2 Power on and connect test jig (instead of the catheter)

9.8.1.3 Power on the Zapper Box

9.8.1.4 Power on the Control Box

9.8.1.5 Set default settings of **Temperature** and **Power** for the Zapper

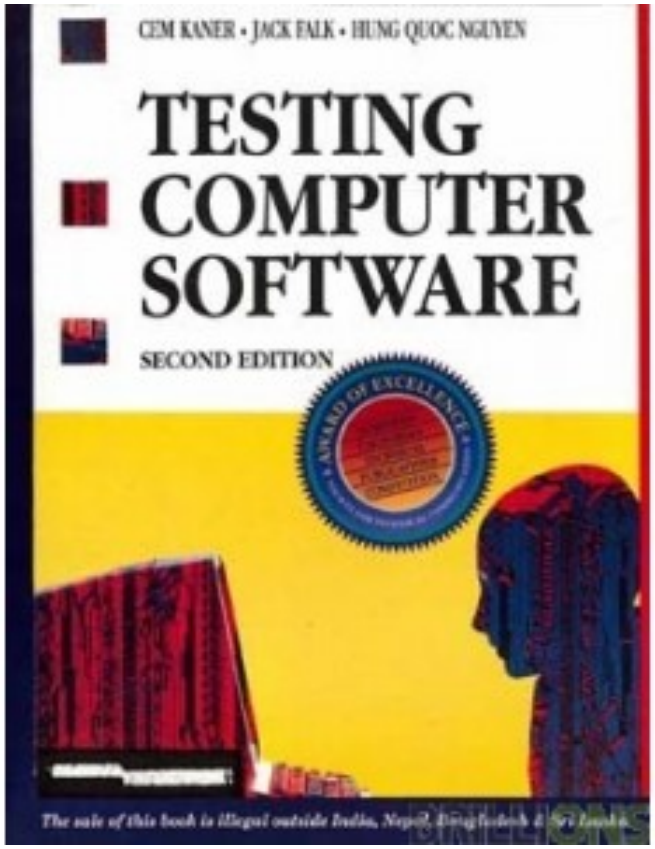
9.8.1.6 Set test jig load to nominal value

9.8.1.7 Select nominal duration and nominal power setting

9.8.1.8 Press the **Start** button

9.8.1.9 Verify Zapper reports the power setting value ±10% on display

Pre ET 1.0: Awakening



Cem Kaner coined the term “exploratory testing” (ET) in 1983 in preparation for Testing Computer Software, published in 1988… and no one really noticed.

ET 1.0: Rebellion

- It was not clear how many people were DOING very informal testing.
- It *was* clear that talk of testing focused on “the test case”, as it still does.
- The word “exploratory” appeared in *TCS*, but few seemed to notice it.
- Those who did treated ET as a distinct activity, or a technique.
- This was a mistake*, serving to marginalize ET!

* in hindsight

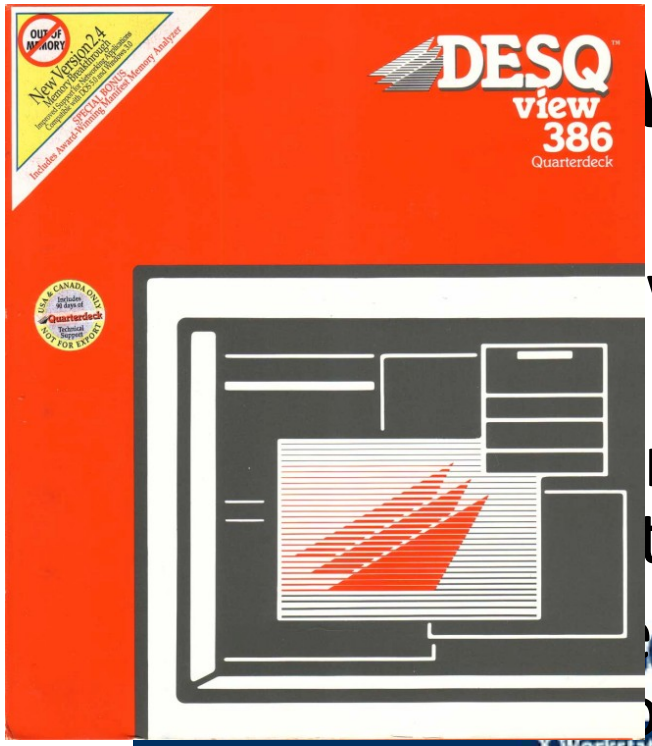
When Did You Get On The Bus?

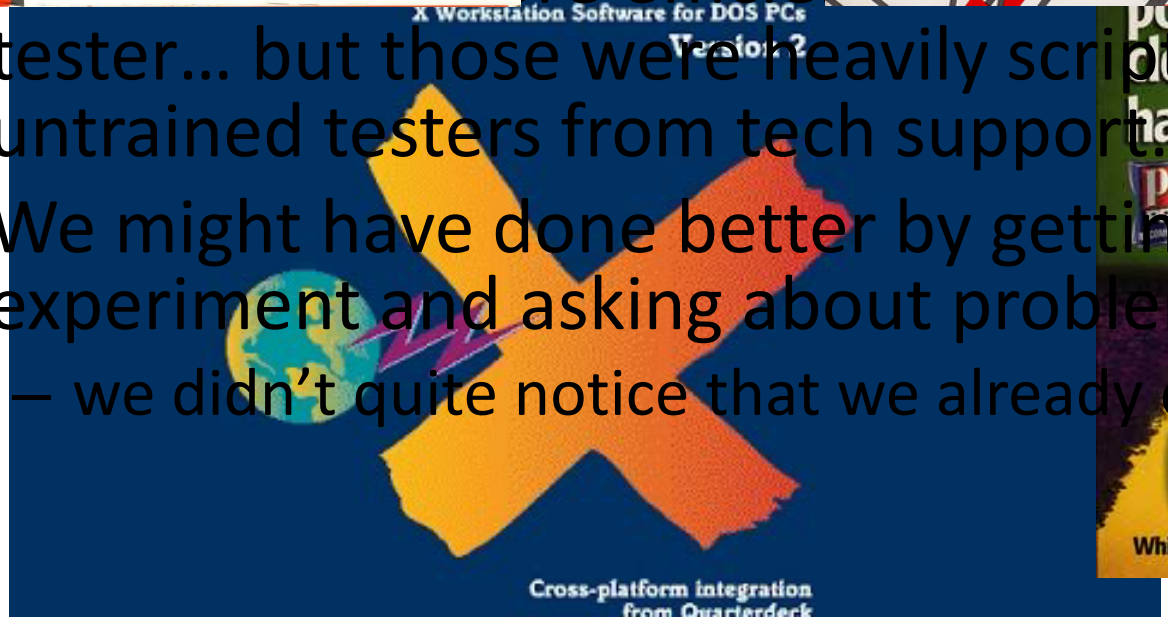
Withou

we practi ght

rmal scrip d,
tty terribl
r claimed
re efficie ated
tester… but those were heavily scripted AND untrained testers from tech support.

- We might have done better by getting them to experiment and asking about problems
  – we didn't quite notice that we already did that.





What IS Exploratory Testing?

- Simultaneous test design, test execution, and learning.

—James Bach, 1995

ET 1.5: Explication

- Things really started to get rolling in 1999
- Appearance of the General Functionality and Stability Test Procedure (formalized ET)
- Some interesting work from Kaner, Bach, Marick, Agruss & Johnson
- Failed experiments with undoing confusion between “ad hoc” and “exploratory” testing
  - really a distinction between skilled and unskilled

The Ad Hoc Business

- "Ad hoc" doesn't mean
  - sloppy
  - slapdash
  - unstructured
- "Ad hoc" does mean
  - literally "to this"
  - implicitly "to this *purpose*"
- So good testing *is* ad hoc, but since many people seem reluctant to use words precisely…

there's a another marketing problem.

SBTM (2000): ET Sessions as Things

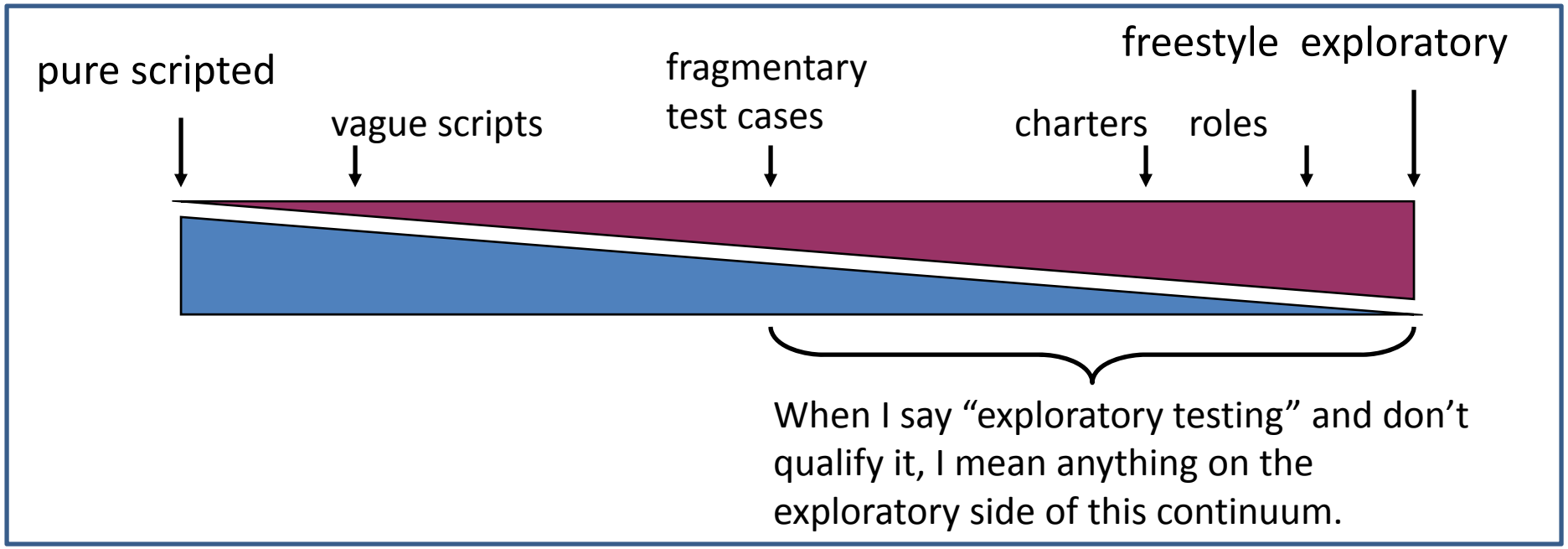
- For some managers, an obstacle to accepting ET is that although engineering is complex, cognitive, exploratory work, they prefer to think in terms of discrete units of production.
- Workaround: put the complex cognitive work inside a discrete unit of production (the session), as Jon and James Bach did with "High Accountability Exploratory Testing" (later, "Session-Based Test Management")
- Backfire: ET was further entrenched as a discrete activity; a *thing*.

Surprise! Applying Session-Based Testing to Medical Software, Medical Device and Diagnostic Industry, May 1, 2003
http://www.mddionline.com/article/applying-session-based-testing-medical-software

The ET Continuum

- Changed ET from “technique” to “approach” that applies to any technique
- Like a style of cooking, a style of dancing, an adjective, a mindset

James Bach: The Scripted/Exploratory Continuum from 2003



Testing Without a Map, 2005

- With Brian Marick's very helpful editing and James Bach's HICCUPPS heuristics, I was able to show that ET *is* a structured process

- Proponents were starting to get noticed. Meanwhile, others were talking about "exploratory testing" and "error guessing" in the same breath, as techniques.
- Detractors usually compared "*scripted*" testing to *unskilled* testing. They left out the skilled exploration that led to scripts!

Exploratory Testing Research Summit, 2006
Participants









James Bach, Jonathan Bach, Scott Barber, Michael Bolton, Elisabeth Hendrickson, Cem Kaner, Mike Kelly, Jonathan Kohl, James Lyndsay, Rob Sabourin, and an impressively large elephant in the room.

The Exploratory Testing Research Summit

- We all accepted the definition “simultaneous test design, test execution, and learning (with an emphasis on learning)”.
- Everyone agreed on what the definition *was*, but not on what it *meant*. (We had reached *shallow agreement.*)
- We could have taken this a sign that ET pervaded all of testing, but we didn’t quite take it that way.
- What the adjective “exploratory” added, and how it contrasted with “scripted,” was the dimension of agency. In other words: *self-directedness*.

What is Scripting?

- A script, in the general sense, is something that constrains our actions in some way.
- A *formal* script is one that must be followed in a specific way to enact a specific procedure.
- An explicit script is a formal script that has been expressed, written, or told.
- *The essence of scripted testing is that the tester is not in control, but rather is being controlled by some other agent or process.*
- This one simple, vital idea took us years to apprehend!

Self-Direction

- We now recognize that by “exploratory testing”, we had been trying to refer to *rich, competent testing that is self-directed*. In other words, in all respects other than agency, skilled exploratory testing is not distinguishable from skilled scripted testing.
- Only agency matters, not documentation, nor deliberation, nor elapsed time, nor tools, nor conscious intent.
- We’re still dealing with the implications of that realization, too.

ET 2.0: Freedom and Responsibility (2007)

- In the year following ExTRS, Cem Kaner produced this definition, to which many agreed

Exploratory software testing
- is a style of software testing
- that emphasizes the personal freedom and responsibility
- of the individual tester
- to continually optimize the value of her work
- by treating test-related learning, test design, and execution
- as mutually supportive activities that run in parallel
- throughout the project.

See Kaner, “Exploratory Testing After 23 Years”, www.kaner.com/pdfs/ETat23.pdf

Sapient Testing (2007)

- "***A sapient process is any process that relies on skilled humans.***"
- Sapient and non-sapient doesn't work because "non-sapience" *sounds like* "stupid" and therefore it sounded like we were condemning checking by calling it non-sapient.
  - but we didn't have a word for checking
- Fortunately, machines and programs didn't take offense—and they're the non-sapient things.

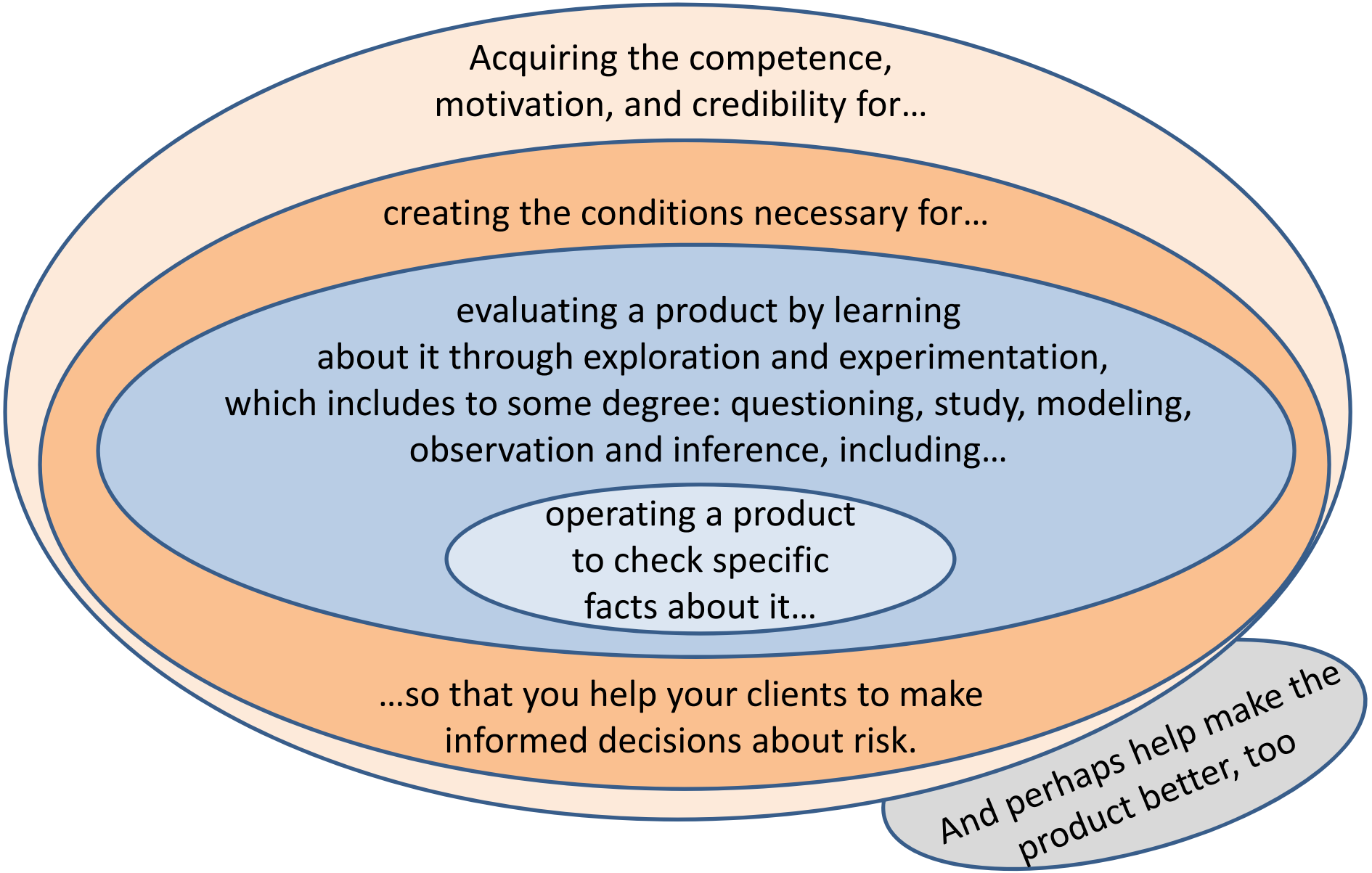
Testing vs. Checking (2009)

- The “vs.” part got a bad reaction (even from colleagues), and still does from some
  - interestingly, even though no one complains about “trees vs. leaves”
  - and no one complains about “compiling vs. programming”, either
- We still have to deal with the problem that people see “checking” as an insult, rather than as a *tactic* of testing.

Testing and Checking Refined (2013)

- ***Testing*** is the process of evaluating a product by learning about it through exploration and experimentation, which includes to some degree: questioning, study, modeling, observation, inference, etc.
- ***Checking*** is the process of making evaluations by applying algorithmic decision rules to specific observations of a product.

Testing is…


Acquiring the competence,
motivation, and credibility for…
creating the conditions necessary for…
evaluating a product by learning
about it through exploration and experimentation,
which includes to some degree: questioning, study, modeling,
observation and inference, including…
operating a product
to check specific
facts about it…
…so that you help your clients to make
informed decisions about risk.
And perhaps help make the
product better, too

Do you notice how fine distinctions of language and thought can take *years* to work out?

On the Road to ET 3.0

- McLuhan's ideas about media and tools
- Karl Weick, *Sensemaking in Organizations*
- Venkatesh Rao, *Tempo*
- James C. Scott, *Seeing Like a State*
- Harry Collins, *Tacit and Explicit Knowledge*
- The concept of "responsible tester" (defined as a tester who takes full, personal, responsibility for the quality of his work).
- The recovery of "testing" within the Rapid Software Testing namespace

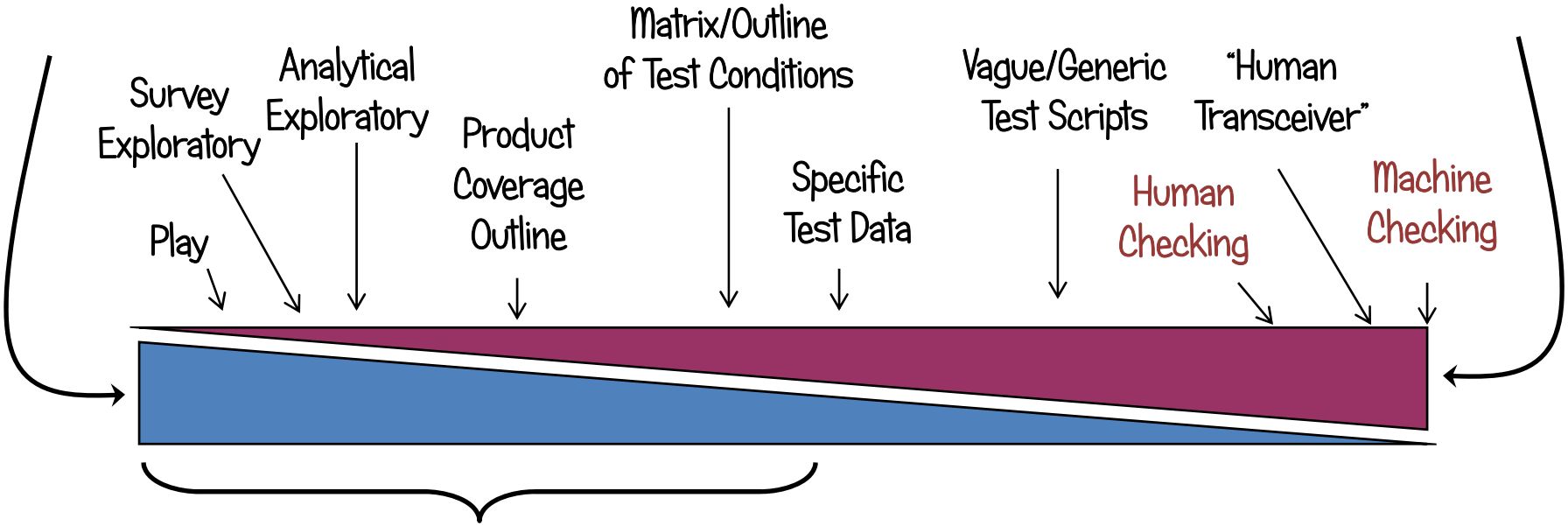
The *Formality* Continuum, 2014

INFORMAL

Not done in any specific way, nor to verify specific facts.

FORMAL

Done in a specific way, or to verify specific facts.



When we say "exploratory testing" and don't qualify it, we mean anything on the exploratory side of this continuum.

Bug fix: formality may intensify over time, thus left-to-right makes more sense.

One Big Problem in Testing

Formality Bloat

- Much of the time, your testing doesn't need to be very formal*
- Even when your testing *does* need to be formal, you'll need to do substantial amounts of informal testing in order figure out how to do *excellent* formal testing.
  - Who says? The FDA. See http://www.satisfice.com/blog/archives/602
- Even in a highly regulated environment, you do *formal* testing primarily for the auditors. You do informal testing to make sure you don't lose money, blow things up, or kill people.

* *Formal testing means "testing that must be done to verify a specific fact, or that must be done in a specific way."*

ET 3.0 (2015)

- With ET 3.0, we demote scripting to a technique—a *tactic* of testing, not its centre—and we promote of exploratory testing to, simply, *testing*.
- Scripting, not exploration, is the guest in the house of testing.
- We are deprecating “exploratory testing” within the Rapid Software Testing namespace.
- We can still talk about “exploratory testing”, but we also look forward to people recognizing that testing is *fundamentally* an exploratory activity: we’re continuously investigating, discovering, and learning—mapping out territory anew, or in more detail.

Objections

- "Isn't it all just testing?"

  Yes. And we would prefer to know what we're talking about when we say "testing", so we can look at specific skills, dynamics, patterns, tools, and tactics—and get better at them.

- "Why bother with all this philosophy-talk? Can't we just get back to work?"

  Go ahead. **But what kind of work do you want to do?** *Our* plan is to continue to develop a detailed and deep sense of the skills of testing and a rich vocabulary for talking about them and effective methods of teaching them. Please feel free to join us.

"A verbal cloak of ignorance is a garment that often hinders progress."

—Topley & Wilson, 1929

We're not home yet!

- Testing is still ruled by test cases
- Testers are (in the main) very poor at telling all three key parts of the testing story
  - the story of the product
  - the story of the testing
  - the story of the quality of the testing
- Testers (in the main) have not yet developed sophisticated use of tools, nor are tools embraced for exploratory purposes.
  - see Kaner and Hoffman, and early stuff by Marick

Conclusion (for now!)

- Are you doing testing? *Then you are already doing exploratory testing*.
- Are you doing scripted testing? If you're doing it responsibly, you are doing *exploratory testing with scripting* (and perhaps with checking).
- If you're *only* doing "scripted testing," then you are just doing *unmotivated checking,* and we would say that you are not really testing. You are trying to behave like a machine, not a responsible tester.

Some of the names…

- Jerry Weinberg, Cem Kaner, Brian Marick, James Bach, Jonathan Bach, Scott Barber, Elisabeth Hendrickson, Jonathan Kohl, James Lyndsay, Rob Sabourin, David Gilbert, Rikard Edgren, Henrik Emilsson, Petter Mattsson, Doug Hoffman, Erik Peterson, Anne-Marie Charrett, Aaron Hodder, Oliver Erlewein, Eric Jacobson,…
- My sincere apologies to those I've forgotten