Testing and Notic ng

Michael Bolton
DevelopSense
michael@developsense.com

James Bach
Satisfice
james@satisfice.com

Updates

- This presentation is ALWAYS under construction
- Updated slides at http://www.developsense.com/past.html

Acknowledgements

- Mark Federman → Terrence Gordon → Marshall McLuhan for the original title (“What Haven’t You Noticed Lately?”)
- Cem Kaner
- Jerry Weinberg
- Adam White
- Pradeep Soundararajan
- …and the authors of the works cited in the references.

A Gospel of Liberation

The Gospel of No Gospels

- No appeals to
  - best practices
  - heavyweight processes
  - management support
- Excellent testing starts with *us*
  - this depends upon the personal freedom, responsibility, and skill of the tester
  - we don’t need anyone’s permission to sharpen our skills
  - let’s invest *ourselves* in our own education

The Question We All Dread

Why didn't you find that bug?!

One Possible Answer

Uh... we didn't notice it.

Yet have you ever noticed *why*?

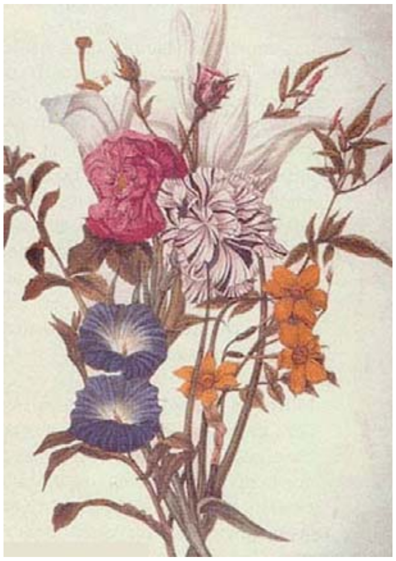
- "*We didn't notice it.*" Intake
- "We did notice it, but we didn't know that it was a problem." Meaning
- "We did find it, and we knew that is was a problem, but we didn't think it was important enough to worry about." Significance
- "We were busy reporting other bugs that we found." Response
- "We didn't explain it well enough."

Let's look at…

NOT

noticing.

What Do You See Here?
(Please answer silently!)

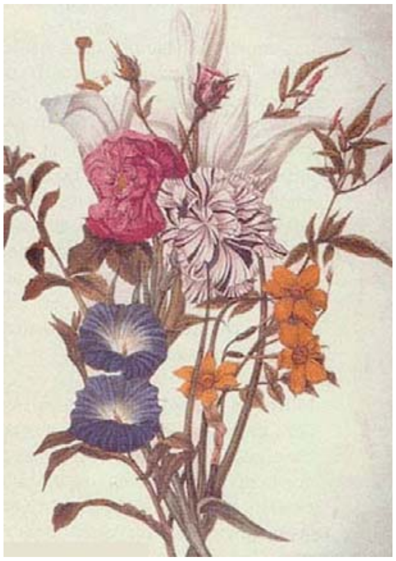


What About When We Change the Question?

Please keep your answers to yourself!

- Notice anything when you look *between* the flowers?
- What happens when we treat the flowers as *ground* and treat the background as *figure*?

Now you can answer out loud!



What is Noticing?

- Recognizing events, objects, or properties
- Starts with direct sensory intake, then moves to meaning, significance, and response
- An internal reaction
- Can be managed by dynamically managing observation and focus
- Often triggered by emotional reactions
- *Very* vulnerable to incompleteness and error
- Can be trained and improved

A Detective Story

What Might We Notice?

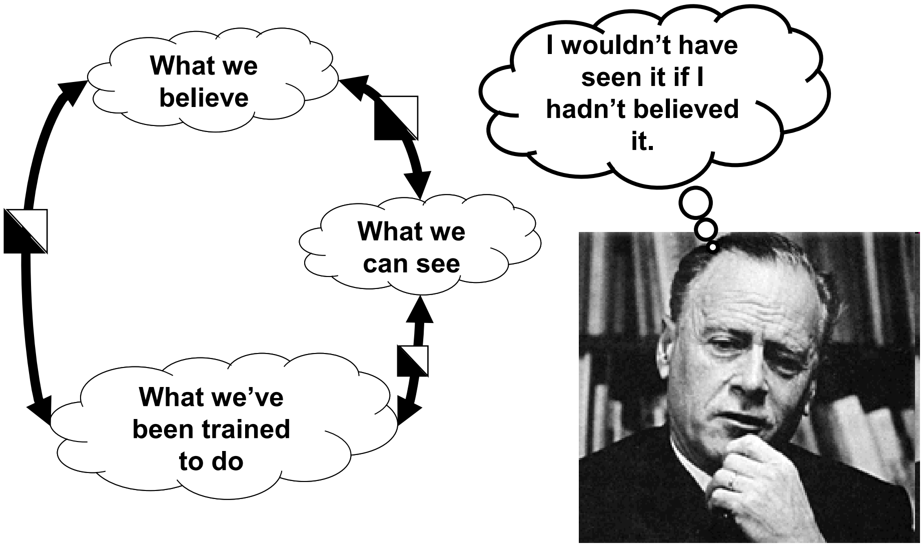
- Problems or other kinds of information about products we test
- Things about ourselves
- Things about others
- Things about our environment
  - especially things that slow down or otherwise inhibit the best testing we can do

Have you ever noticed…?

Our conception limits

our perception

Feedback Loop: Perception & Conception



What we believe

What we can see

What we've been trained to do

I wouldn't have seen it if I hadn't believed it.

Perception and Conception

- What we conceive about things around us is not sufficient to fully understand all the effects that are actually happening in and around us. We are completely unable to perceive all of the dynamics of our environment because our *con*ception limits our *per*ception. Our intense focus on precisely what we have been trained to do controls what we believe, and what we believe controls what we are able to see.

  - Mark Federman

http://whatisthemessage.blogspot.com

Conceptual Priming Ideas

- The skill of factoring
- Your technical knowledge and beliefs
  - beliefs about what is (im)possible
  - beliefs about what is (un)likely
- Guideword heuristics
- Patterns of familiar bugs
- Learning how rapid cognition works
  - …and how it might fail

Have you ever noticed…?

Diversity supports better noticing.

The Law of Requisite Variety

- A controlling system needs to have more states than the system it controls
- "Complicate yourself if you want to understand something complicated."

  - G.F. Smith
- "A tester is someone who knows things can be different."

  - Jerry Weinberg

Practice Factoring!

- *"List all the dimensions of (some common object) that may be relevant to testing it."*
- **"dimensions"** means attributes of the object that may vary
  - from one object to another; or
  - within the same object over time.
- **"relevant to testing"** means that there is probably some value to some client, with respect to some testing mission, of manipulating or observing a particular dimension.
- Pick an object and factor it; compare notes; notice categories of dimensions (and of categories)

There Are Factors To Observation, Too!

- the thing being observed (the *system*)
- the environment (all the things around us)
- our knowledge and models (conceptual)
- our senses (perceptual)
- our experience (experiential)
- our feelings (affective)
- our mission and our client

…*and* there are factors to each one of these factors!

Consistency ("this agrees with that")

an important theme in oracles

History

Image

Comparable Products

Claims

User Expectations

Purpose

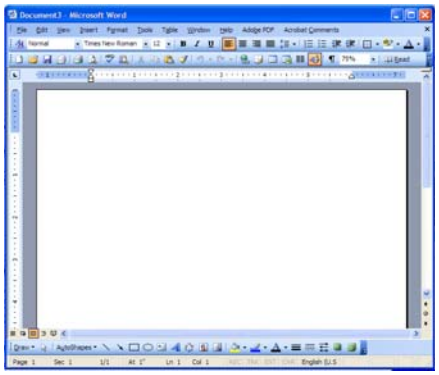
Product

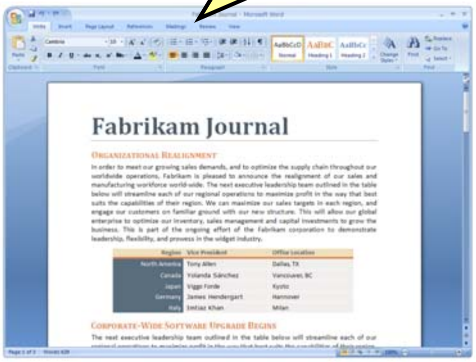
Standards

Consistency heuristics rely on the quality of your models of the product and its context.

History

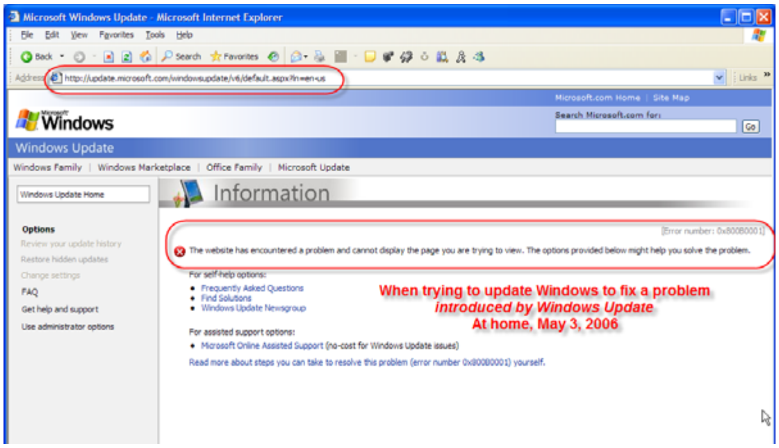
Okay, so how the #&@ do I print now?


Document3 - Microsoft Word
File Edit View Insert Format Tools Table Window Help Adobe PDF Acrobat Comments


Fabrikam Journal
Organizational Realignment
Corporate-Wide Software Upgrade Begins

If a product is inconsistent with previous versions of itself, we suspect that there might be a problem.

Image


Microsoft Windows Update - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://update.microsoft.com/windowsupdate/v6/default.aspx?ln=en-us
Links
Microsoft.com Home | Site Map
Search Microsoft.com for:
Go
Microsoft Windows
Windows Update
Windows Family | Windows Marketplace | Office Family | Microsoft Update
Windows Update Home
Options
Review your update history
Restore hidden updates
Change settings
FAQ
Get help and support
Use administrator options

Information

[Error number: 0x80080001]

The website has encountered a problem and cannot display the page you are trying to view. The options provided below might help you solve the problem.

For self-help options:

- Frequently Asked Questions
- Find Solutions
- Windows Update Newsgroup

For assisted support options:

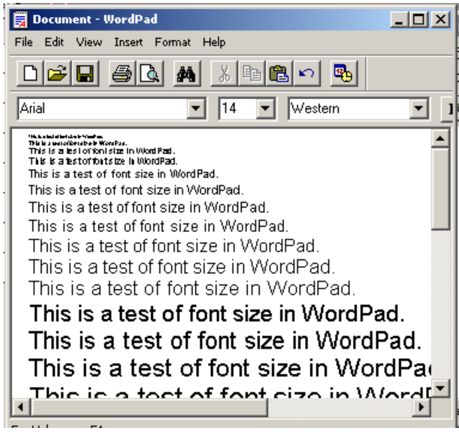
- Microsoft Online Assisted Support (no-cost for Windows Update issues)

Read more about steps you can take to resolve this problem (error number 0x80080001) yourself.

When trying to update Windows to fix a problem
introduced by Windows Update
At home, May 3, 2006

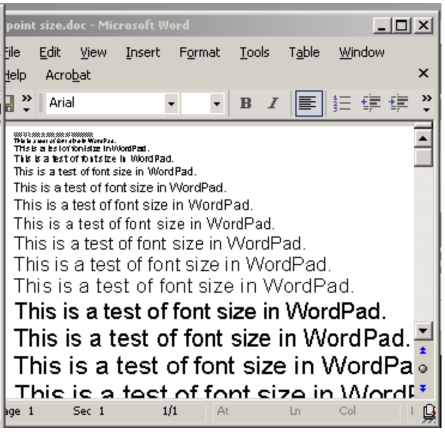
If a product is inconsistent with an image that the company wants to project, we suspect a problem.

Comparable Products



This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPa

WordPad

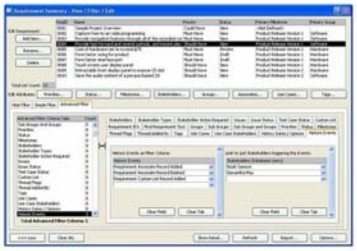

point size.doc - Microsoft Word
File Edit View Insert Format Tools Table Window
Help Acrobat
Arial
B I
age 1 Sec 1 1/1 At Ln Col

This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPad.
This is a test of font size in WordPa

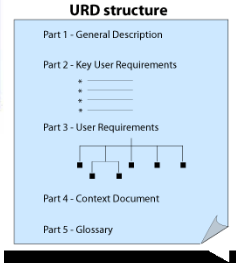
Word

When a product seems inconsistent with a comparable product, we suspect that there might be a problem.

Claims

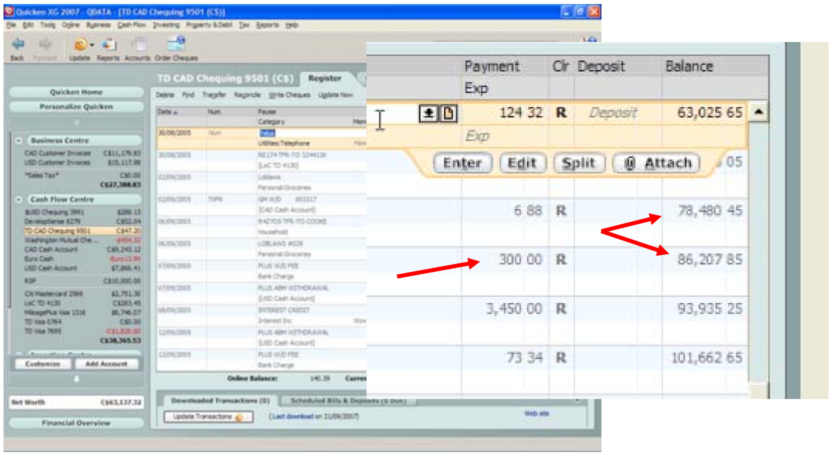


New! Supports Mac OS/X!


URD structure
Part 1 - General Description
Part 2 - Key User Requirements
Part 3 - User Requirements
Part 4 - Context Document
Part 5 - Glossary

When a product is inconsistent with claims that important people make about it, we suspect a problem.

User Expectations

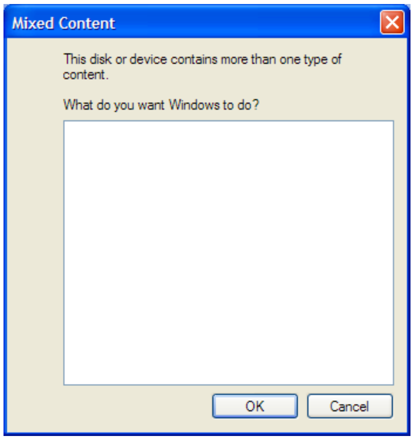


Payment Exp	Clr	Deposit	Balance
124 32 Exp	R	Deposit	63,025 65
6 88	R		78,480 45
300 00	R		86,207 85
3,450 00	R		93,935 25
73 34	R		101,662 65

Quicken XG 2007 - QDATA - [TD CAD Chequing 9501 (C$)]
TD CAD Chequing 9501 (C$)
Register
Enter
Edit
Split
Attach

When a product is inconsistent with expectations that a reasonable user might have, we suspect a problem.

Purpose



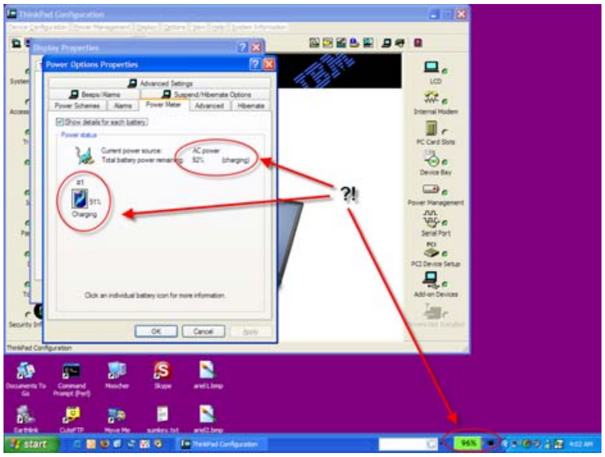
Mixed Content

This disk or device contains more than one type of content.

What do you want Windows to do?

When a product is inconsistent with its designers' explicit or implicit purposes, we suspect a problem.

Product



Current power source: AC power

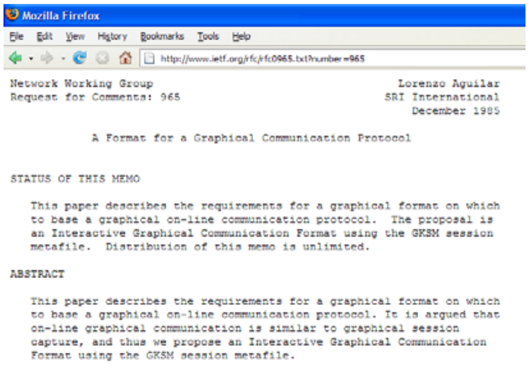
Total battery power remaining: 92% (charging)

#1 91% Charging

96%

When a product is inconsistent internally—as when it contradicts itself—we suspect a problem.

Statutes and Standards



Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.ietf.org/rfc/rfc0965.txt?number=965

```
Network Working Group                                    Lorenzo Aguilar
Request for Comments: 965                              SRI International
                                                           December 1985

             A Format for a Graphical Communication Protocol


STATUS OF THIS MEMO

   This paper describes the requirements for a graphical format on which
   to base a graphical on-line communication protocol.  The proposal is
   an Interactive Graphical Communication Format using the GKSM session
   metafile.  Distribution of this memo is unlimited.

ABSTRACT

   This paper describes the requirements for a graphical format on which
   to base a graphical on-line communication protocol. It is argued that
   on-line graphical communication is similar to graphical session
   capture, and thus we propose an Interactive Graphical Communication
   Format using the GKSM session metafile.
```

When a product is inconsistent with laws or widely accepted or relevant standards, we suspect a problem.

But How Do I Keep Track? HICCUPPS!

- History
- Image
- Comparable Products
- Claims
- User Expectations
- Purpose
- Product
- Statutes

…plus for “Familiar Problems”, add the *inconsistent* F!

Guidewords help to prime our ability to notice.

Thirty-Six Things You COULD Notice

"cidtestdsfdpotcrusspicstmplfdsfscura"

Customers
Information
Developer relations
Team
Equipment & tools
Schedule
Test Items
Deliverables

Project Environment

Structures
Functions
Data
Platforms
Operations
Time

Product Elements

Capability
Reliability
Usability
Security
Scalability
Performance
Installability
Compatibility
Supportability
Testability
Maintainability
Portability
Localizability

Quality Criteria

Function testing
Domain testing
Stress testing
Flow testing
Scenario testing
Claims testing
User testing
Risk testing
Automatic testing

Test Techniques

Have you ever noticed…?

Perception can be practiced and trained.

Sharpening Perception

- Priming our senses
- Learn about magic tricks
- Noticing our emotional triggers
- Training the intuition
- Watching others test or use a product
- Practice!

Have you ever noticed…?

It's sometimes hard

to know how to notice.

Listening for Biases (Just a Sampler)

- Evaluative Bias of Language
  - "*Our* product is full-featured. *Theirs* is bloated."
- Representativeness Bias
  - "It's just a one-line change. Why bother testing?"
- Automation Bias (machines over people)
  - "The green bar tells us we're done."
- Reification Error (counting the uncountable)
  - "We've got 10,487 test cases for our 492 requirements."

cf. *Tools of Critical Thinking: Metathoughts for Psychology*, by David Levy.
Wikipedia also has some fascinating lists of biases.

Assimilation Bias: That's (not) Testing

- Testing is
  - questioning a product in order to evaluate it (Bach)
  - gathering information with the purpose of informing decisions (Weinberg)
- Have you ever noticed that these things are testing activities?
  - code review running tests observing the behaviour of the team assessing tech support records test-driven development designing tests code inspection evaluating test results reporting reviewing requirements

Assimilation Bias (a.k.a. Lumping)

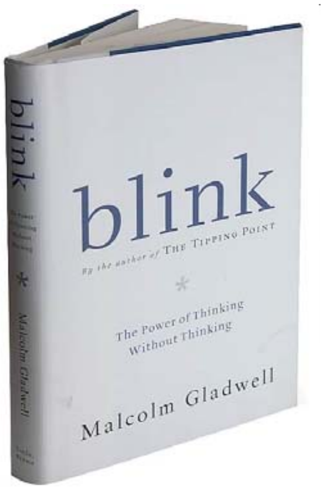
How can we estimate
"the testing phase"
when we're mostly
waiting for other people?

Assimilation Bias (a.k.a. Lumping)

How can we estimate "the testing phase" when it doesn't really exist anyway?

Snap Judgments, Rapid Cognition

- Snap judgments are central to our decision-making process
- Snap judgments are vulnerable to corruption by forces that we're not aware of and that we don't understand
- Sometimes we can improve the quality of our snap judgments by *removing* information, rather than adding it
- Instead of changing the decision-maker, change the context in which the decision is made.


blink
By the author of The Tipping Point
The Power of Thinking Without Thinking
Malcolm Gladwell

Blink Tests

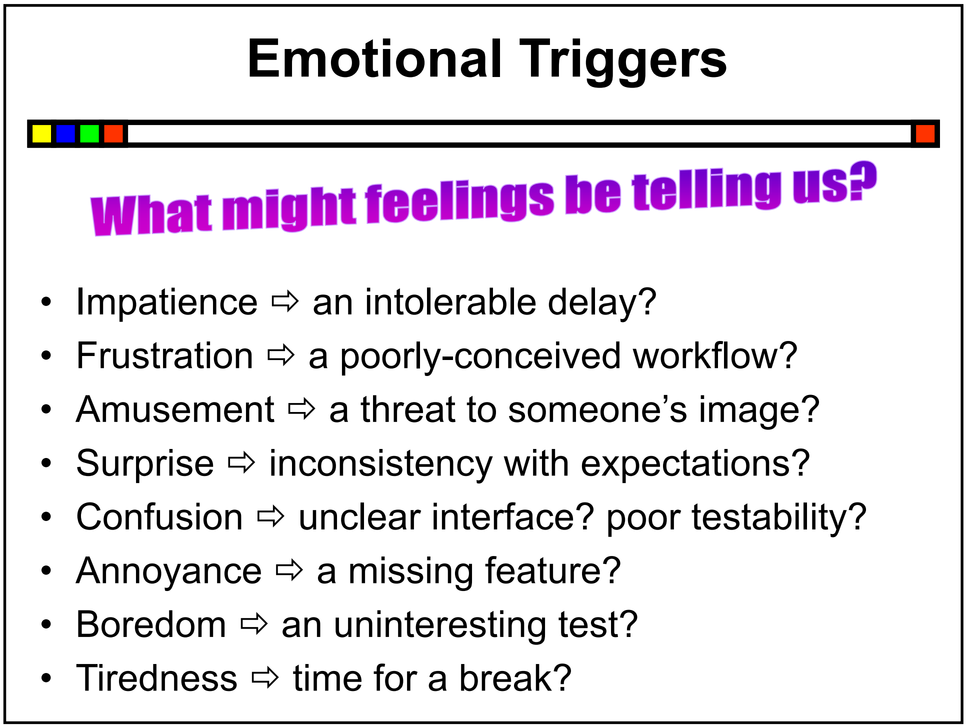
- A blink test is a test that leverages human pattern recognition
- Example: Flipping rapidly between two screens that should be identical
- Example: ProSum
- Example: patterns in the data

Have you ever noticed…?

Noticing can be

triggered by

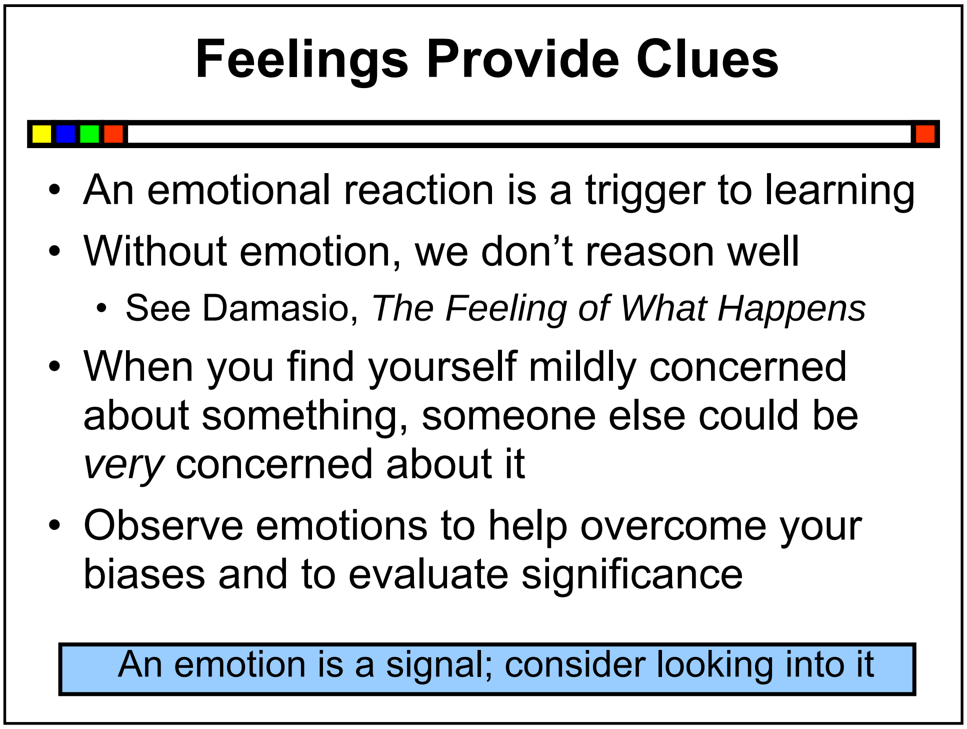
emotions and feelings.



Emotional Triggers

What might feelings be telling us?

- Impatience ⇨ an intolerable delay?
- Frustration ⇨ a poorly-conceived workflow?
- Amusement ⇨ a threat to someone's image?
- Surprise ⇨ inconsistency with expectations?
- Confusion ⇨ unclear interface? poor testability?
- Annoyance ⇨ a missing feature?
- Boredom ⇨ an uninteresting test?
- Tiredness ⇨ time for a break?



Feelings Provide Clues

- An emotional reaction is a trigger to learning
- Without emotion, we don't reason well
  - See Damasio, *The Feeling of What Happens*
- When you find yourself mildly concerned about something, someone else could be *very* concerned about it
- Observe emotions to help overcome your biases and to evaluate significance

An emotion is a signal; consider looking into it

Affective Priming

Preparing Your Emotional Mindset

- building confidence
- developing tolerance for mistakes
- tolerance for confusion
- stress inoculation
- embracing and celebrating the new
- avoiding learned helplessness
- recognizing and dealing with environments that might be unsupportive or hostile

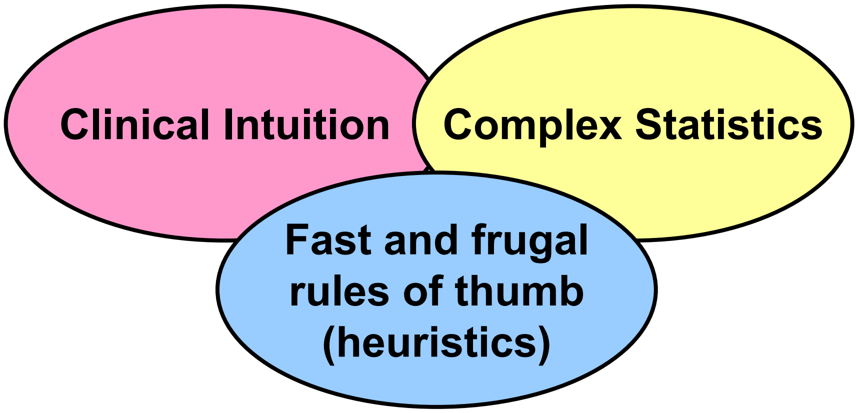
Have you ever noticed…?

Noticing can be

influenced by experience.

Heuristics in Medicine: A Case Study

- There are at least three ways to direct people to treatment



What To Do?

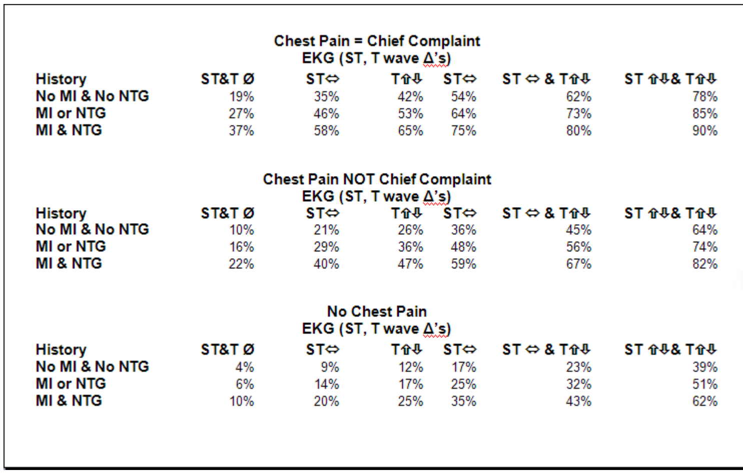
- A man is rushed to hospital with severe chest pain
- Decision:
  - coronary care unit?
  - regular nursing bed (with a heart monitor)?
- Problem:
  - based on long term risk factors (family history, male, advanced age, smoking, diabetes, high cholesterol, hypertension) doctors sent about 90% of patients to coronary care.
  - care unit became crowded, quality of care decreased, cost went up

A story from *Gut Feelings*, by Gerd Gigerenzer.

Research

- Doctors sent most patients to the CCU
- Sent patients who should have been there just as often as those who shouldn't have
- The decision was no better than chance.

Heart Disease Predictive Instrument



Chest Pain = Chief Complaint
EKG (ST, T wave Δ's)

History	ST&T Ø	ST⇔	T⇧⇩	ST⇔	ST ⇔ & T⇧⇩	ST ⇧⇩& T⇧⇩
No MI & No NTG	19%	35%	42%	54%	62%	78%
MI or NTG	27%	46%	53%	64%	73%	85%
MI & NTG	37%	58%	65%	75%	80%	90%

Chest Pain NOT Chief Complaint
EKG (ST, T wave Δ's)

History	ST&T Ø	ST⇔	T⇧⇩	ST⇔	ST ⇔ & T⇧⇩	ST ⇧⇩& T⇧⇩
No MI & No NTG	10%	21%	26%	36%	45%	64%
MI or NTG	16%	29%	36%	48%	56%	74%
MI & NTG	22%	40%	47%	59%	67%	82%

No Chest Pain
EKG (ST, T wave Δ's)

History	ST&T Ø	ST⇔	T⇧⇩	ST⇔	ST ⇔ & T⇧⇩	ST ⇧⇩& T⇧⇩
No MI & No NTG	4%	9%	12%	17%	23%	39%
MI or NTG	6%	14%	17%	25%	32%	51%
MI & NTG	10%	20%	25%	35%	43%	62%

This table, with the aid of a long formula and a pocket calculator, helped doctors to make better CCU assignments.

Method

- The doctors were told to
  - find the right probabilities for each patient
  - type these into a calculator with a long formula
  - press ENTER
  - read off the result
  - compare it to a threshold number
  - route the patient to the CCU or a regular bed

Accuracy went up.

But…

- Even though accuracy was up…
- Even though overcrowding eased…

The doctors hated it.
They didn't understand it.

Testing the Conclusions

- The researchers tested the efficacy of the method and the calculations.
- They took the tables and calculators away.

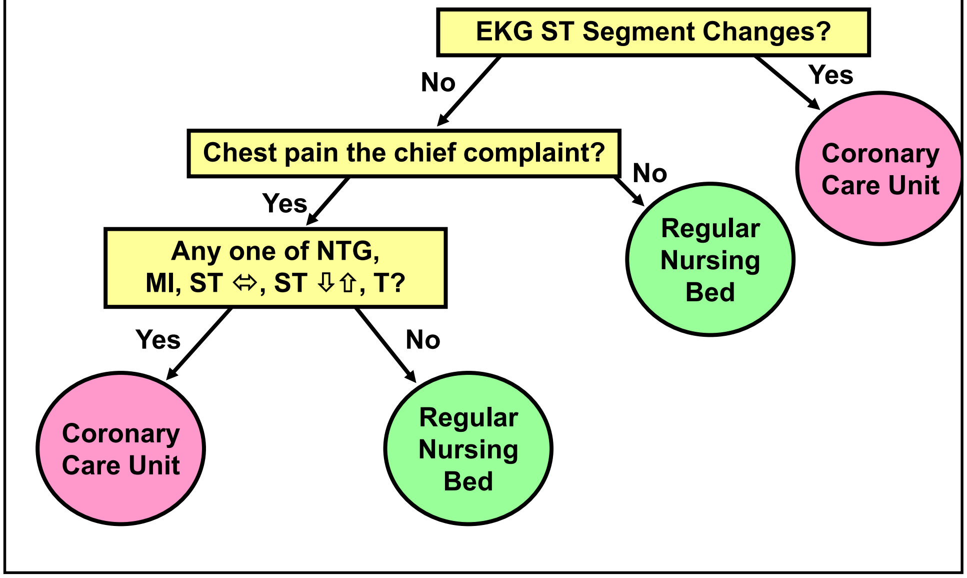
Accuracy remained high.

- After the doctors had been exposed to the chart, their intuitions improved permanently.

Conclusions on the HDPI Approach

- When systems with heavy calculations and many probabilities conflict with intuition, people tend to resist the complex solution
- When there is high uncertainty, simple diagnostic methods tend to be more accurate
- Practice with the complex solution appeared to train doctors' intuitions subconsciously
- This led to the recognition and development of a heuristic…

The Fast and Frugal Tree



EKG ST Segment Changes?

No

Yes

Coronary Care Unit

Chest pain the chief complaint?

No

Regular Nursing Bed

Yes

Any one of NTG, MI, ST ⇔, ST ⇩⇧, T?

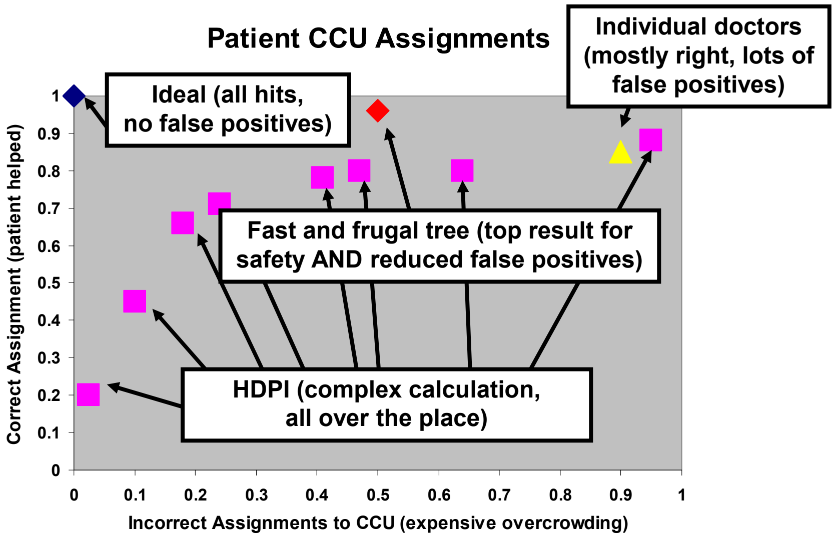
Yes

No

Coronary Care Unit

Regular Nursing Bed

Heuristics Win!



Patient CCU Assignments

Individual doctors (mostly right, lots of false positives)

Ideal (all hits, no false positives)

Fast and frugal tree (top result for safety AND reduced false positives)

HDPI (complex calculation, all over the place)

Practice!

- Testing = interaction with software = learning
- Play with the software
- Be a real user
- Experience patterns of (eventually) familiar problems
  - on the job (scary, career-threatening)
  - experiential training (less scary, less risky)
- Learn stuff seemingly unrelated to testing
  - new skills, hobbies, games, stories...
  - theatre, improv, role-playing, “how to act”

Why *Experiential* Learning?

- “When you reckon up what you know about the world, you will soon see that most of what you know has been purchased, like your socks and shoes, from those whose business it is to put those ideas on the market.”

  - Adam Smith

Environmental Priming

Preparing By Situating Yourself

- Minimizing disruptive distractions
- Maximizing productive distractions
  - taking a break? taking a shower?
- Creating a safe environment for noticing
  - notice the environment, and notice when it's hostile to observation
- Introduce equipment and tools to assist observation

Have you ever noticed…

Some products

are hard to test?

Object Priming

Preparing The Thing To Be Observed

- incorporate log files and debug output
- build scriptable interfaces
- include built-in error detection and correction
  - so we don't *have* to notice
- provide better error and status messages
- design consistent user interfaces and workflows

Testability ~= Usability

One More Big Thing

- Have you noticed how much time you spend on investigating and reporting bugs?

A well-tested program

gives us more time to notice!

How Do We Spend Time?

(assuming all tests below are *good* tests)

Module	Bug reporting/investigation (time spent on tests that find bugs)	Test design and execution (time spent on tests that find no bugs)	Number of tests
A (good)	0 minutes (no bugs found)	90 minutes (45 tests)	45
B (okay)	10 minutes (1 bug, 1 test)	80 minutes (40 tests)	41
C (bad)	80 minutes (8 bugs, 8 tests)	10 minutes (5 tests)	13

Investigating and reporting bugs means….

SLOWER TESTING or…

REDUCED COVERAGE …or both.

- In the first instance, our *coverage* is great—but if we're being assessed on the number of bugs we're finding, we look bad.
- In the second instance, coverage looks good, and we found a bug, too.
- In the third instance, we look good because we're finding and reporting lots of *bugs*—but our *coverage* is suffering severely. A system that rewards us or increases confidence based on the number of bugs we find might mislead us into believing that our product is well tested.

What Happens The Next Day?

(assume 6 minutes per bug fix verification)

Fix verifications	Bug reporting and investigation today	Test design and execution today	New tests today	Total over two days
0 min	0	45	45	90
6 min	10 min (1 new bug)	74 min (37 tests)	38	79
48 min	40 min (4 new bugs)	2 min (1 test)	5	18

Finding bugs today means….

VERIFYING FIXES LATER

…which means….

EVEN SLOWER TESTING or…

EVEN LESS COVERAGE …or both.

- …and note the optimistic assumption that all of our fixed verifications worked, and that we found no new bugs while running them. Has this ever happened for you?

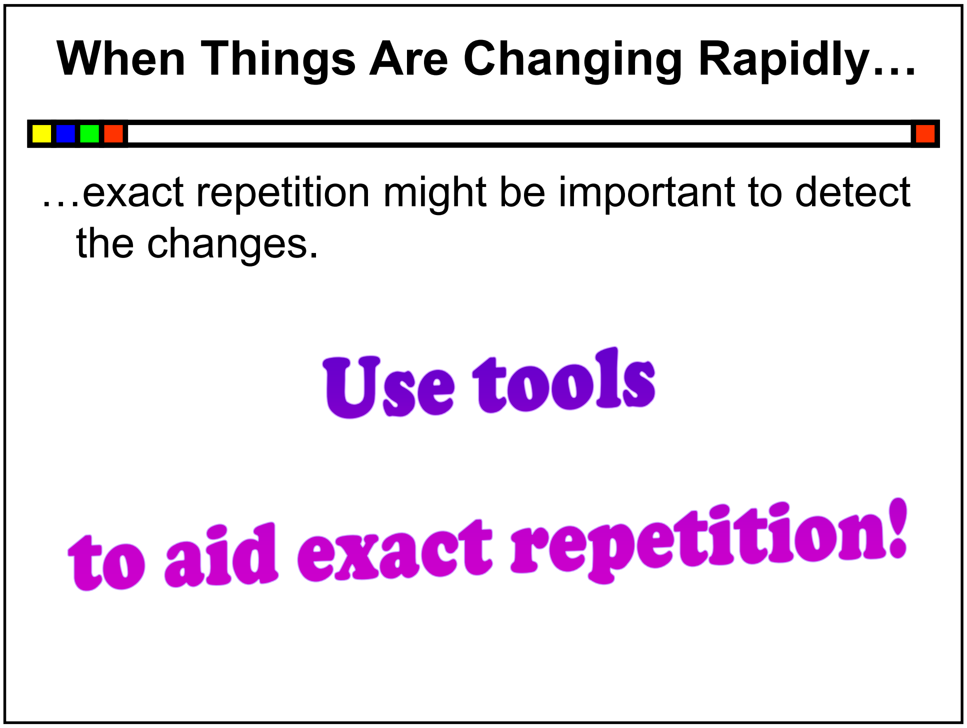
Have you ever noticed…?

We notice different things

at different times.

Repetition

- What does it mean to repeat a test?
- *Exact* repetition means
  - looking at exactly the same things
  - performing exactly the same actions
  - in exactly the same order
  - using exactly the same data
  - making exactly the same observations
- Humans aren’t so good at exact repetition
  - but they can be **great** at noticing new things
- Machines are excellent at exact repetition
  - but they’re lousy at noticing

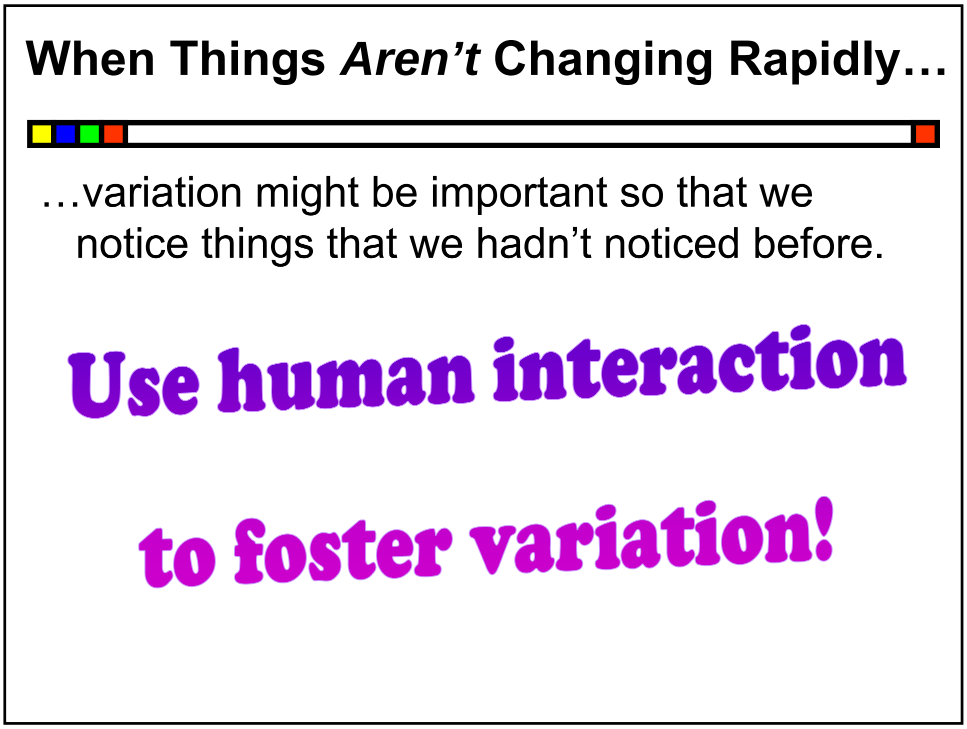


When Things Are Changing Rapidly…

…exact repetition might be important to detect the changes.

Use tools

to aid exact repetition!



When Things *Aren’t* Changing Rapidly…

…variation might be important so that we notice things that we hadn’t noticed before.

Use human interaction

to foster variation!

But…

- …people tend to behave conventionally and systematically (though unconsciously), so…

Use automation to

induce randomness!

Managing Attention

- To test for *anticipated* problems…
- To test a *simple* product, or *part* of a complex product very thoroughly…
- To *pinpoint* an observed problem…
- To *confirm* that a fix has been made…
- To maximize test integrity…
- To stay in grooves…

Focus!

Focusing Heuristics

1. Start the test from a *known* (clean) state.
2. Prefer *simple, deterministic* actions.
3. Vary One Factor At a Time (OFAT)
4. Trace test steps to a *specified model*.
5. Follow *established and consistent* lab procedures.
6. Make *specific* predictions, observations and records.
7. Make it *easy to reproduce* (automated input may help).

Managing Attention

- To find *unexpected* problems*…*
- To find *elusive problems* in sustained field use…
- To *test* whether a fix has broken something else…
- To discover new dimensions of the product or the testing mission…
- To get out of ruts…

De-Focus!

Defocusing Heuristics

1. Start from a variety of different states (not necessarily clean).
2. Prefer complex, challenging actions.
3. Vary Many Factors At a Time (MFAT).
4. Generate tests from a variety of models, or without reference to a conscious model.
5. *Question* your lab procedures and tools.
6. Try to see everything with open expectations.
7. Make the test hard to pass, instead of easy to reproduce (automatic logging and screen recording may help).

Metacognition

- observe what you're observing
- to some degree, just being aware of the pitfalls helps you to defend against problems
- Beware the Meaning Problem
  - you think a signal means one thing, but it means another
  - "That alarm only goes off when there's a fire drill. This must be just another fire drill"
- Beware the Significance problem
  - "If it were a really serious fire, there'd be an announcement. This must not be a serious fire."
- Beware the Fill-in Problem
  - our brains automatically compensate for missing information
- pair up!

Change the Observation

- Choose something specific to observe
- Choose another sense (hearing? touch? smell?)
- Ask
  - “What other things are going on?”
  - “Are there more things like this one?”
  - “What do I believe is the cause of this effect?”
  - “Are there different causes (other than the one that I have inferred) of the same effect?
  - “Are there different effects from the same cause?”
  - “What other meanings or significance we could take from the thing we’ve just observed?”

It’s *Okay* Not To Notice Everything

- We can’t notice everything, even if we wanted to
- That’s why we have development teams
- That’s why we have test teams
- That’s why we have review and testing
- That’s why we test using many different approaches

An Example of Last-Minute Updates

- They think that intelligence is about noticing things that are relevant; to me it is NOT noticing things that are irrelevant.
    - Nassim Nicholas Taleb

Have you ever noticed…?

Sometimes we notice things later.

When all else fails, take a break,

do something else,

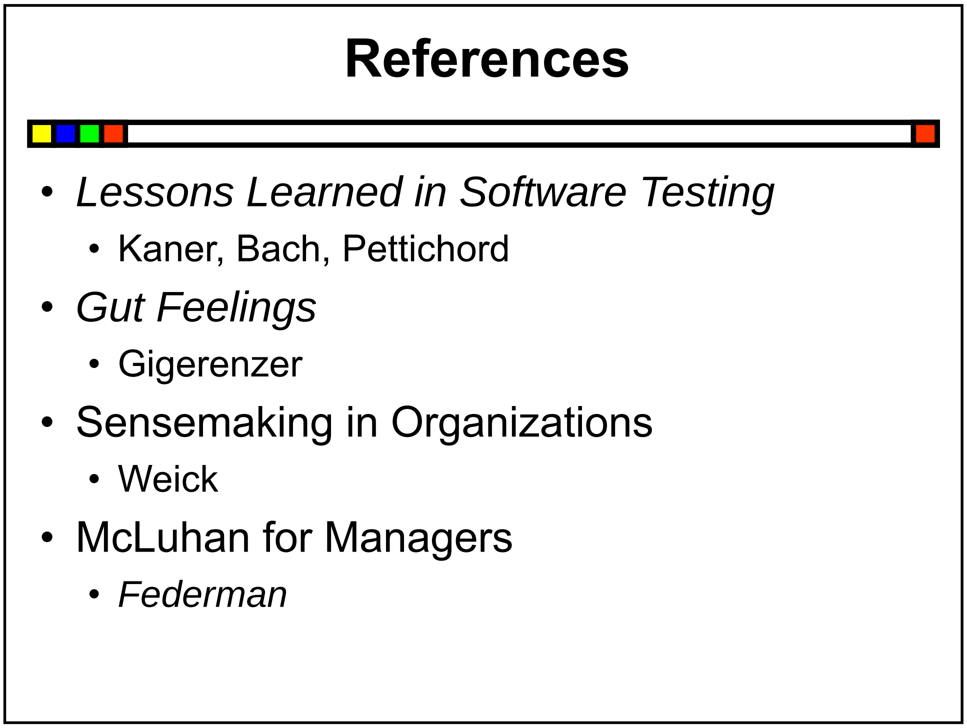
come back to the problem.

Remember This Guy?

- Me, just before the presentation
- Notice any changes?

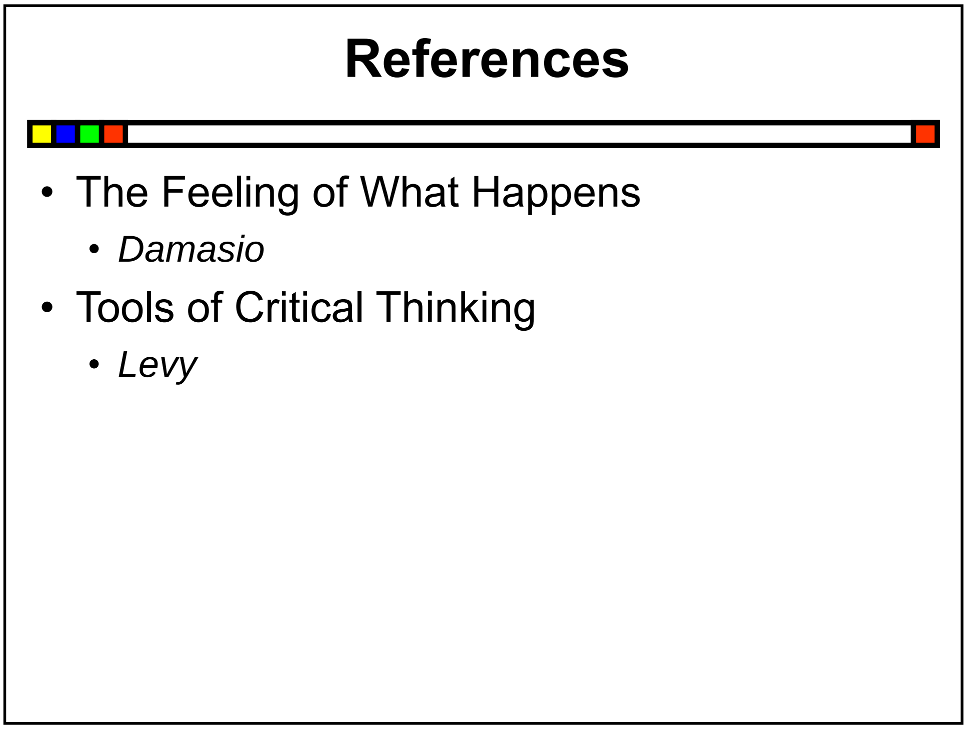
References

- *Blind Spots*
  - Marilyn Von Hecke
- *Introduction to General Systems Thinking*
- *General Principles of System Design*
- *Perfect Software and Other Illusions About Testing*
  - Gerald M. Weinberg
- *Blink*
- *The Tipping Point*
  - Malcolm Gladwell



References

- *Lessons Learned in Software Testing*
  - Kaner, Bach, Pettichord
- *Gut Feelings*
  - Gigerenzer
- Sensemaking in Organizations
  - Weick
- McLuhan for Managers
  - *Federman*



References

- The Feeling of What Happens
  - *Damasio*
- Tools of Critical Thinking
  - *Levy*