Context-Driven Testing

Michael Bolton
Keynote Address, CAST 2011
August 8, 2011

1 Introduction

For those of you who chose to attend CAST to see and hear Cem Kaner speak from this platform, I'd like to offer my sympathies. I wanted to see him too. Perhaps I can reduce your disappointment a little by saying that I've been studying Cem's work and cribbing liberally from it for so long—since 1996—that I might even be learning to anticipate it. When we found that he'd be unable to attend the conference, Cem was kind enough to share some ideas that he intended to address in his keynote. I'm rather pleased to say that had I been asked to develop a talk for this conference a long time ago, I would have touched on a number of points that he would have. So now, having been given the opportunity, I'd like to do that, and I hope I can honour Cem in the doing of it.

2 How To Think About Science

I'll start with a story that I've been rabbiting on about to anyone who'll listen for the last several years. In fact, alas for them, it doesn't really matter whether they were willing to listen or not.

I live in Canada, and apart from our health care system, one of the things that makes life in Canada worthwhile is the Canadian Broadcasting Corporation, our public television and radio service. For the last 30 years or so, CBC Radio has been running a program called Ideas, which is about... ideas. The program has a very open mandate; cosmology one night, community policing the next, trends in education the next, aboriginal mythology the night after that. Typically the shows are one-offs on a particular topic, but every now and then the producers deliver a series. A few years back, David Cayley presented a 24-part series called "How to Think About Science", which included interviews with scholars, scientists, historians of science, sociologists, philosophers, and at least one magician. It's a fantastic series, and it's still available for streaming[1] (or, if you have RealPlayer, downloading). In the first segment, David Cayley interviewed Simon Schaffer, an English historian of science. Schaffer was one of the authors (with Stephen Shapin) of *Leviathan and the Air Pump*[2], a book that examines a raucous and quite nasty dispute that was in full force pretty much exactly 350 years ago today. The argument was between Robert Boyle and his colleagues of the Royal Society on the one side, and Thomas Hobbes and his friends on the other. There were relatively few of Hobbes' friends, by the way; as Schaffer says, Hobbes was the possibly the most irascible man alive in the 17th century.

Boyle—the discoverer of Boyle's Law—had taken on a research project inspired by William Harvey to find out what goes on when the blood circulates. Something happens to blood when it passes by the lungs, and Boyle figured that it had something to do with air. So he and his colleague Robert Hooke set to testing, using a technique that we regard as commonplace and obvious today: if you want to figure out the effect that something has, try getting rid of it. They did that by using the most sophisticated and glamorous scientific instrument of the day, the air pump. The use of instruments extends human capabilities on the one hand, allowing humans to control aspects of the natural world—"to vex

[1] http://www.cbc.ca/ideas/episodes/2009/01/02/how-to-think-about-science-part-1---24-listen/
[2] Shapin, Steven, and Schaffer, Simon: Leviathan and the Air-Pump. Princeton University Press, 1989. ISBN 978-0691024325.

Nature", as Francis Bacon would say[3]. At the same time, instruments often set humans at some distance from the experiment, and thereby reduce our natural variability. Boyle saw this as a good thing, since he had a broader agenda than research on the air—again something that testers today are intimately familiar with: the need to establish credibility and trust, and the need to do that reliably. Boyle wanted to institutionalise knowledge itself, such that one could produce and establish what he called *matters of fact*.

Instrumentation was the first pillar in a set of protocols that he called *experimental philosophy*, which was foundational in establishing science as we know it. The second pillar was that experiments had to be performed in front of witnesses who would provide testimony that the experimenter had indeed done what he had said he'd done. The third pillar was a style of recording and reporting that Shapin and Schaffer in their book called "virtual witnessing"—a complete record of the experimental procedure and how to reproduce it, including plans on how to reconstruct the instrument, if that were necessary. Boyle performed a significant number of experiments in exactly this way, and used the results to back his theories on the behaviour of air and its role in respiration, claiming those theories to be established matters of fact.

And Hobbes wasn't buying any of it. He raised a number of significant and very strenuous objections, nastily and in public.

Hobbes had technical concerns. The air pump leaked. It was notoriously finicky and unreliable. Boyle didn't have the luxury of rubber or plastic seals; he used leather soaked with oil. When the machine worked, Boyle claimed that the experimental results proved his theories. Yet sometimes he used the variability of the instrument to buttress his theories, saying something like "we would have got the answer I'd expect, except the machine doesn't work perfectly". (Does the problem of unreliable tools sound familiar to any testers out there?) Hobbes rarely missed an opportunity when Boyle was trying to suck and blow at the same time. (Sorry about that.)

Hobbes also pointed out that the experiments were taking place in laboratories and drawing rooms in London. How, he asked, could one know if the experimental results would be reproducible elsewhere? On the Continent? At the tops of mountains? You couldn't make universal claims about nature based on an experiment in a single place. (Does the problem of unrepresentative environments sound familiar to any testers out there?)

Apart from the technical problems, Hobbes had other objections more consistent with his reputation as a political philosopher. One philosophical objection was based on the idea that an experiment can't prove anything conclusively. In this, he anticipated David Hume's attack on induction. Hume, in the 18th century, pointed out problems of induction, taking observed cases and extending them to unobserved cases. For example, most of us would agree that the sun will probably rise again tomorrow, just as it did today. Yet we can't really justify that belief based on experience. Pause for a second and think about that. We believe that Nature behaves in this regular fashion, but none of us has been alive long enough to know that that's so over a period longer than a lifetime. Moreover, whatever data we might extract from our own past or from our limited supply of prior reports, we still have no way of knowing, really knowing, that Nature will continue to behave regularly. Although he preceded Hume by a century or so, Hobbes espoused a similar line of reasoning, based on his fascination with Euclid and geometry. Knowledge, true knowledge, comes from universal axioms and theorems that can be logically derived from them. If philosophy is to be based on true knowledge, said Hobbes, it

[3] "…the secrets of nature reveal themselves more readily under the vexations of art than when they go their own way." Bacon, Francis. *The New Organon* (William Wood, translator). Mobile Reference, Amazon Digital Services, 2008.

can't be derived from something as unreliable as a handful experiments performed in front of a handful of supportive witnesses in a few English drawing rooms using leaky equipment. If it's experimental, it's not philosophy. (Some people assume that because something been tested, they can claim they know how it works. Does that sound like a problem for any testers out there?)

Hobbes said that you couldn't come to a conclusion—or even to an experiment—unless you had already reckoned up a set of beliefs. Your conclusions happen inside a specific theoretical framework. So do your experiments and your observations. In fact, your entire research program is based on a particular set of precepts and theories. With this, Hobbes also anticipated Thomas Kuhn in the 20th century: all observations are theory-laden[4]. (In other words, the tests you perform and the assertions that you make about something are based on a particular set of assumptions. Even the observations you make are predicated on what you're looking for and how you're looking for it. Are there any testers who are familiar with that problem?)

Hobbes's most important point, says Schaffer, was that Boyle was not really solving the social problem that he claimed to be solving. Boyle claimed that experiments would lead to consensus and gentlemanly agreement on matters of fact, optimistically asserting that a well-run and well-documented experiment ought to be sufficient to persuade any non-believers. Hobbes disagreed with Boyle's theories and premises, and was therefore unconvinced by his results. He argued that Boyle invited his own colleagues to witness experiments, but not opponents or the public at large. While Hobbes would certainly have claimed that his own disagreement was rational, he also recognized the fact that people would—possibly irrationally—hold on to their beliefs as long as they had an interest or an investment in the idea, resisting or rejecting new information. (Are the problems of confirmation bias and anchoring bias, having to deal with people and beliefs that are dug in—are those familiar to any testers out there?) As Schaffer puts Hobbes' point of view, when people's interests clash, *they*—the people—will clash. In that sense, Hobbes anticipated the idea of cognitive dissonance, which only came to the fore in this century[5]. It takes more than a matter of fact to convince someone. If they don't believe the facts, then they'll start to attack the basis for declaring something a fact.

Okay: so that's a bit of a long story. How does it relate to what we're up to today?

It seems to me that there are a number of interesting observations to be made here. Some of them are paradoxical. For example, Hobbes was advocating for the kind of absolute certainty that one could obtain from the axioms and logical deductions of geometry. Mere humans and mere experimentation, he maintained, couldn't produce that kind of certainty or knowledge that was worthy of being institutionalised. In that sense, although he was advocating for certainty in knowledge on the one hand, he was also advocating a certain kind of epistemic humility for experimental science as we have come to know it. And that advocacy worked. Indeed, in response, Boyle refined his equipment and his protocols. He also, eventually and reluctantly, moderated his claims for the social power of the experimental method.

The debate also highlighted the role of politics in decisions about knowledge, paralleling Jerry Weinberg's critical observation that decisions about quality are always political (and emotional)[6]. Boyle wanted to empower experimenters by setting up a process by which they could come to consensus and conclusions, and by which they could obtain credibility and respect. Because of

---

[4] Kuhn, Thomas, *The Structure of Scientific Revolutions*. University Of Chicago Press; 3rd edition, 1996.

[5] See Tavris, Carol and Aronson, Elliot, *Mistakes Were Made (But Not By Me): Why We Justify Foolish Beliefs, Bad Decisions and Hurtful Acts*. Mariner Books, 2008.

[6] Weinberg, Gerald M., *Quality Software Management, Volume 1: Systems Thinking*. Dorset House Publishing, 1991.

people's tendency to defend their theories and their points of view—and consistent with his political philosophy centered around a strong central authority—Hobbes held that you needed stronger precepts to underlie science, or else there would be disagreement, brawling, and war.

Hobbes and Boyle read different conclusions out of the same experimental results. That's still an issue today. You can see it in politics and in science for that matter: what we call a fact is context-dependent.

The similarities between this little story and the scene in testing today strike me as remarkable. It seems that in many ways, nothing much has changed in the last 350 years or so. There is always disagreement. There are different schools of thought. Let's look at that issue for a moment.

3 What about schools?

3.1 The helpfulness of schools

Some people and people that I respect greatly have questioned, challenged, and even rejected the usefulness of schools.

Here's what I see as the useful part. In the hands of a thoughtful person, the notion of schools allows us to take a more measured tone with people and ideas that we don't agree with. As James Bach has said, the idea of schools give us the opportunity to say "So-and-so's thinking reflects that of a different school of thought from mine", which saves us from saying—or even thinking—"So-and-so is a fool."

Or as Cem Kaner has said to me, reasonable people can disagree reasonably. That makes sense because we reasonably have different values, and we are in service to people who reasonably have different values. Describing ideas and theories and practices in terms of schools reminds us of that.

3.2 The unhelpfulness of schools

Many of my closest colleagues disagree on this point. James and Doug Hoffman will be having a debate on the issue here tomorrow. James will almost certainly hold that you must choose a school in the same way that you must choose a religion. If context-driven testing is like any religious view, it seems to me that it ought to be like strict agnosticism.

In religion (and, so it seems, in testing) fundamentalists of any stripe tend not to have tolerance for non-believers. Herein lies something that looks like paradox. There's no one here from the ISTQB this year. I'm glad of that, because I don't like the goods that they're putting on to the market, and I don't like their marketing, either. At the same time, it seems to me that we ought to welcome them. On the one hand, there are contexts in which the ISTQB and routine-school thinking is desired and even reasonable (albeit largely in organizations that want to manage skepticism and uncertainty away; my chapter in Reducing the Cost of Testing is about that). On the other hand, having ISTQB types here would give us a chance to argue with them. Some people have expressed relief to me that there isn't likely to be the kind of debate that happened between James and Stuart Reid in 2007. I'm sorry about that, because I missed it, and it sounded like a squirmy kind of fun. I envy that. In Finland this spring, I got to say a whole lot of terrible—albeit truthful, in my view—things about the ISTQB, but the ISTQB guy who was there got too upset to argue with me, which was frustrating.

The concept of schools has been regarded as divisive by many people outside of our own. There are parallels to that in religion, too. People really, really dislike being labelled when they're not the ones doing the labelling, especially when there's baggage associated with the label. Up until recently at least, it was fine for black people to call *themselves* "nigger", or gay people to call *themselves* "faggot",

but watch how the mood changes when an outsider uses those labels. Even in this room, just now, I expect that many people squirmed at the sound of those words. Context again.

It's an odd thing to me, but it's true: people who strongly advocate manufacturing models for testing are the same people who don't like being tagged as factory-school thinkers, as some of us are inclined to do. "Can't we just call it all testing?" they ask. That's a reasonable request if—and only if—you see testing through the lens of some kind of assimilation bias, and if you don't think distinctions are useful. Except I believe that distinctions *are* useful, and papering over disagreements is ultimately unhelpful. I'm not happy with pasting a label onto other people, but I don't mind categorizing their beliefs so much. And I don't know of a more efficient way of dealing with the matter, since schools don't declare themselves and the differences are real.

In any case, I think it is definitely reasonable for context-driven people to reject beliefs, practices, organizations that they disagree with, since context-driven doesn't mean "anything goes"; it means that we consider context *first*.[7]

4 What does it mean to be context driven?

The basis of context-driven testing can be found, I think, in a story of an encounter between James Bach and Jerry Weinberg, a story that James tells in *The Gift of Time*[8]. Please forgive me, but I'm going to quote it at some length. (I should also note that Jerry, speaking from the keynote platform at CAST 2008, disagreed with the notion of schools of testing.) Here's James' story.

> "I had been visiting Jerry for a week of conversation and writing work, and we were heading up his driveway on one of our walks, when abruptly he asked "What's the first responsibility of a tester?"
>
> My head was immediately brimming with candidates. "I don't think I can pick just one. How about five?"
>
> "I'll tell you what it is," he said, waving me off. "The first responsibility of a tester is to explore and challenge the constraints of the situation. Kind of like how you just challenged the constraints of my question. A tester is someone who knows that things could be different. (*Isn't that marvellous? —MB*) However they appear, however they really are, whatever we feel about them, it all could be different than it seems, or different tomorrow than it is today. Testers need to be inquisitive; to shake things up a bit, so that people can see things in a different light. Now here's another question for you: What do you do if you don't have enough time to find every bug in a program?"
>
> "That's a trick question. No one ever has the time to find every bug in a program. But that's not our goal. Our goal usually is to find every important bug."
>
> "And if there isn't time to do that?"
>
> "Well, that's why we use our risk analysis to guide us. We put our efforts where they are most needed."
>
> "And if your risk analysis is wrong?"
>
> "That's why we also use a non-risk-oriented test strategy for some of the testing. For instance, we might use a coverage-oriented test technique to scan for unknown risks."

---

[7] http://www.context-driven-testing.com

[8] Charles, Fiona (ed.) *The Gift of Time.* Dorset House, 2008.

"And if you aren't given enough time to do that well? Or maybe you have the time and you do it badly?"

"I suppose we just keep trying, and we also try to learn from our mistakes."

"True, but you might not learn the right things from your mistakes. Do you see what I'm getting at, James?" I shrugged expectantly and he continued. "Beyond any clever strategy you try, there is something further that you need: you need a philosophy of acceptance."

"A philosophy of acceptance? We just accept failure?"

"You accept reality. That's a tester's job. Your strategies will occasionally fail, no matter what you do. While a tester's first responsibility is to refuse to accept the apparent constraints of the situation, a tester must ultimately accept that some ambiguities and constraints may never be identified or resolved. It's an awkward field. If you wanted solid ground to stand on, you chose the wrong vocation."

That absence of solid ground seems to me to be a bass line, the rhythm section, running underneath the principles of the context-driven school[9]: each principle is related in some way to uncertainty.

4.1 Principles

That reminds me of Brian Marick's version of the purpose of testing and his motto for the testing team[10]

"We are a service organization whose job is to reduce damaging uncertainty about the perceived state of the product."

People absolutely do want to be more confident and more certain. Yet if you look around a software development project, you'll typically see more than enough certainty. That certainty is obtained in all kinds of ways, but it's often reinforced by confirmation bias, by anchoring bias, by cognitive dissonance. Hallmarks of that, it seems to me, include

- the impulse to make testing clerical and pro forma;
- the impulse to reduce testing to test cases;
- a related impulse to reduce the number of test cases;
- the impulse to delegate observation to programs as much as possible;
- the impulse to spend time on busy work that prevents or delays us from examining the product and gaining experience with it.

This isn't a new thing; it's been a well-known phenomenon in the social sciences for years.As Kirk and Miller say in their book Reliability and Validity in Qualitative Research[11], "Most of the technology of confirmatory, non-qualitative research in both the social and natural sciences is aimed at *preventing* discovery." (my emphasis)

---

[9] http://www.context-driven-testing.com
[10] http://www.exampler.com/testing-com/writings/purpose-of-testing.htm
[11] Kirk, Jerome, and Miller, Marc L., *Reliability and Validity in Qualitative Research*. Sage Publications, Inc., 1985.

That's not the kind of testing that I want to do. Surely it's our job as testers not to confirm that the product works as expected, but instead to discover how the product actually works. That requires us to expose ourselves and the product to the unexpected. So I disagree with Brian Marick. If we really want to provide something valuable, I'd say, "We are a service organization whose job is to reduce damaging *certainty* about the product." That is, "It's not what we don't know that's the problem, it's what we know that ain't so.[12]" (By the way, The Penguin Dictionary of Modern Humorous Quotations tell us that Josh Billings is the actual author of that aphorism. Everyone knows that quote was uttered by Mark Twain. Or by Will Rogers. But that ain't so. This is the kind of thing that makes an irony junkie like me very, very happy.) Here are the principles of the context-driven school:

- The value of any practice depends on its context.
- The value of any practice OR approach, model, tool, document, heuristic...
- There are good practices in context, but there are no best practices.
- People, working together, are the most important part of any project's context.
- Projects unfold over time in ways that are often not predictable.
- The product is a solution. If the problem isn't solved, the product doesn't work.
- Good software testing is a challenging intellectual process.
- Only through judgment and skill, exercised cooperatively throughout the entire project, are we able to do the right things at the right times to effectively test our products.

4.2 Process maturity

Context-driven testing is based on a different form of process maturity than we're used to. Context-driven testing recognizes and accepts this fact as the point of departure for a testing journey: the world is a complex, variable, and uncertain place. That distinguishes context-driven testing from context-oblivious testing, in which you simply don't recognize complexity, variability, and uncertainty. That approach makes sense for naive, incompetent, or lazy testers. It makes sense in organizations with sufficiently low standards. It makes sense for testers who are working in a context where someone else is entirely responsible for the quality of the work. Finally, it makes sense when testing isn't the mission, as when you, as a tester are being asked to do something that isn't testing—training, demos, sales meeting, that kind of stuff. Then the issue becomes your personal bottom line: do you want to associate your name and your reputation with the activity? You might, but I'd suggest you make it clear that, at that point, you're not testing.

Context-driven testing is also distinct from context-specific testing, in which you do recognize complexity, variability, and uncertainty, but declare that they don't apply to you. That approach makes sense for a context-driven tester whose context never changes. That approach also makes sense if you're in a highly restrictive or regulated kind of context.

Finally, context-driven testing is distinct from context-imperial or context-driving where you, in some way, decide to resist complexity, variability, or uncertainty and actively change the context. That's the approach taken by what we might call the routine or factory school, the Agile school, or the quality school.

---

[12] Metcalf, Fred, *The Penguin Dictionary of Modern Humorous Quotations*. Second Edition, Penguin, 2002.

The acceptance of the world as it is is, to me, central to context-driven testing. In that, it's a hallmark of maturity. Maturity isn't about repetition, nor is it about following some holy process manual written by anonymous sock puppets who have not observed your context or your processes or your business. It's not about doing what Daddy and Mommy tell you to do. It's about developing the skills and the confidence and the independence to deal with a complex, messy, variable, context-sensitive world[13]. That's scary for many people, but I think it's what we have to do.

There's a paradox to be found here, too, perhaps. It it, I think, easy for context-driven testing to seem context-imperial to people from other schools. That's not our intention, but we can be perceived that way. We're foreign elements, after all. In our community, there are lots of big thinkers and loud talkers. People like us who don't have all the answers are intensely annoying to those who do. We also face the arrogance paradox, experienced by people like Socrates in the ancient world, and people like Nassim Taleb (the author of The Black Swan) in the modern world. For some reason, some people refuse to accept the disclaimers that we're not sure about things. There are various psychological reasons for that. Most of our life experience leads us to believe that smart people know what's going on, so when a presumably smart person claims not to know what's going on, it raises cognitive dissonance. "What do you mean, you can't be sure? You're just saying that." Well, we can't be sure because it's our job as testers to be skeptical.

4.3 Context-Driven Testing is ...

4.3.1 Skeptical

Skepticism is not the rejection of belief; it's the rejection of certainty and the preservation of uncertainty. As testers, I argue that we're obliged to maintain our uncertainty, even while everyone around us is losing theirs. Suspending certainty prevents us from automatically rejecting behaviour or interpretations of observation that we might otherwise consider bizarre. Expansive ways of thinking, keeping things open, reduces the risk that we'll miss important problems because of overly narrow definitions, overly narrow focus, insufficient alternative views, or coming to premature conclusions.

We may need to rescue skepticism from sloppy interpretations that some people make of it. Although many skeptics are cynical, skepticism is not cynicism. Suspending judgement is far from cynical. In fact, it's quite humanist:

"By suspending judgment, by confining oneself to phenomena or objects as they appear, and by asserting nothing definite as to how they really are, one can escape the perplexities of life and attain an imperturbable peace of mind[14]." That's Pyrrho, who was so far as we know the founder of skepticism, back in the 4th century BC. This is reminiscent of Montaigne, who said much of the same thing in the 16th century. Scientist might suggest that science has some conclusive answers. The better scientists (Richard Feynman comes to mind) will tell you that even in the face of the strongest evidence we have, nothing in science is a closed topic. Meanwhile, closed arguments, best practices, focus on processes rather than people seem to me to put humanity in the back seat. (I should note that Pyrrho isn't for everybody. Defending uncertainty works great for the Myers-Briggs Ps. It probably won't work so well for the Js. Perhaps some Js can help me work on that.)

---

[13] http://www.developsense.com/blog/2009/10/maturity-models-have-it-backwards/
[14] http://en.wikipedia.org/wiki/Pyrrho

"The proper course of the sage," said Pyrrho[15], "is to ask himself three questions. First, we must ask what things are and how they are constituted. Second, we ask how we are related to these things. Third, we ask what ought to be our attitude towards them." Pyrrho's answer was that things are indistinguishable, unmeasurable, undecidable....He concluded that human senses neither transmit truths nor lie. This reminds me of two testing skills that I think are essential: distinguishing observation from inference and recognizing that all is heuristic.

4.3.2 Heuristic

A heuristic is a fallible method for solving a problem or making a decision. Let's look at what Billy Vaughan Koen has to say about them, in terms of definition, application, and signatures. First, he says that "a heuristic is anything that provides a plausible aid or direction in the solution of a problem but is in the final analysis unjustified, incapable of justification, and potentially fallible."[16]

For application, he says, "The engineering method is the use of heuristics to discover the best change in a poorly understood situation with the available resources."

If that's the engineering method, then it seems to me that the testing method is to add to our understanding of a poorly understood situation, and to ask whether the change implemented by the change is indeed the best one.

What are the hallmarks of heuristics? A heuristic

- Does not guarantee a solution
- May contradict other heuristics
- Reduces the search time for solving a problem, and
- Its acceptance depends upon the immediate context instead of an absolute standard.

Koen says that "All is heuristic." (That includes the statement that all is heuristic.) This is consistent with skepticism; as someone in Wikipedia says, "A philosophical skeptic does not claim that truth is impossible (which would be a truth claim)[17]". Paradox again. Instead, we test our beliefs against the world and our observations of phenomena. That's empiricism.

4.3.3 Empiricist

Here's what Wikipedia says about empiricism. "Empiricism is a closely related, but not identical, position to philosophical skepticism. Empiricists see empiricism as a pragmatic compromise between philosophical skepticism and nomothetic science (that is, law-based and generalized science); philosophical skepticism is in turn sometimes referred to as "radical empiricism."[18]

This is closely related to Feynman's description of theories in science: it doesn't matter how elegant or how beautiful the theory is; if experiment doesn't bear it out, it's wrong[19]. And it reminds me of something that Jerry Weinberg identified in 1961: "Because we are humans, we will tend to believe what we want to believe, not what the evidence justifies. When we have been working on a program for a long time, and if someone is pressing us for completion, we put aside our good intentions and let

---

[15] Ibid.
[16] Koen, Billy Vaughan, *Discussion of the Method: Conducting the Engineer's Approach to Problem Solving*. Oxford University Press, USA, 2003.
[17] http://en.wikipedia.org/wiki/Skepticism
[18] Ibid.
[19] See http://www.youtube.com/watch?v=b240PGCMwV0&feature=related

our judgment be swayed. So often, then, the results must provide the impartial judgment that we cannot bring ourselves to pronounce. One of the lessons to be learned from such experiences is that the sheer number of tests performed is of little significance in itself. Too often, the series of tests simply proves how good the computer is at doing the same things with different numbers. As in many instances, we are probably misled here by our experiences with people, whose inherent reliability on repetitive work is at best variable. With a computer program, however, the greater problem is to prove adaptability, something which is not trivial in human functions either. Consequently we must be sure that each test does some work not done by previous tests. To do this, we must struggle to develop a suspicious nature as well as a lively imagination."[20]

4.3.4 Adaptive

That's an idea that I'd like to underscore, and that I think that we need to get out to the wider world. A focus on repeatability leads to weak testing, since it reduces coverage and inhibits discovery. Testing should be focused on adaptability, not as few tests as possible, but as many as possible, as challenging as possible. As go our products and our tests, so should go our approaches to testing itself. If there are different contexts, we have to be able to respond to them. If quality is value to some person, we have to adapt to different people and different sets of values.

4.3.5 Diversified

When we multiply out the number of possible quality criteria by the number of possible users (or even models of users), times the number of operating systems and browsers and hardware platforms, we get an intractibly large number. Yet all the books suggest that we should run as few tests as possible. I'd argue the opposite: let's ask how we can run more tests, more quickly, more powerful, while increasing our capacity to observe multiple quality criteria. Tools will help the performance of checks, but the task is to increase coverage, not reduce it; to perform many powerful tests, on behalf of many people, and to obtain the broadest coverage possible. That, to me, isn't well-served by running *fewer* tests.

4.3.6 Humanist

Above all, our testing must focus on human values. If quality is value to some person, then we need to understand people and their values. We're here to serve people. We must learn to observe people; our customers, our managers, our programmers, our project community. We must learn how to respond compassionately and empathetically to people who are under pressure and getting bad news from us. We must also learn the skills of tester self-defense.

5 Where do we go from here?

Well, it's a keynote, and one of the things I'm supposed to do in a keynote talk is to suggest some directions. So where do we go from here?

Simon Schaffer, in that CBC interview I talked about, describes a change that's gone on in science since he was a student. In the old days, the "pattern science", the kind of science that people aspired to was theoretical physics. That's changed. The social sciences and the systems sciences are ascendant. This is, in part, because fundamental physics is stuck. String theory, one of the hottest topics in science, is stuck, because no one has yet been able to develop testable experiments. The "pattern

[20] Leeds, Herbert, and Weinberg, Gerald M., *Computer Programming Fundamentals*. McGraw Hill (2nd edition) 1970.

sciences" these days, says Schaffer, are the field sciences: field botany; agronomy. That raises different questions for science and its observers, according to Schaffer, questions other than "who discovered what first?". Instead the questions become more like "Why does this plant grow here and not there? What are the factors that contribute to something thriving in one context and dying in another?" Those are the kinds of questions that are essential whenever we're dealing with social systems—like how we develop, use, and test software. What makes this practice work in one context rather than another? How might this software serve one set of people well and another set badly? It's not possible to understand only the media aspects of social media without also understanding the social bits.

In 2005 or so, Cem Kaner came to the Toronto area and gave a talk to the Toronto Association for System and Software Quality. His talk was called, "Software Testing as a Social Science." I urge you to look that up the various versions of it that he has posted on his Web site[21]. Let me summarize it.

Social sciences

- study humans, especially humans in society
- ask what will the impact of things will be on people
- work with qualitative & quantitative research methods
- have high tolerance for ambiguity, partial answers, situationally specific results
- treat values and ethics as relevant issues
- accept diversity of values and interpretations as normal
- accept observer bias as an accepted fact of life and in well-designed research manage it explicitly instead of ignoring it

5.1 Anthropology

For Christmas last year, my mother gave me a book[22] by a Canadian guy named Wade Davis. He's ethnobotanist—that is, he studies the ways various cultures use plants—and an anthropologist. There's a recurring theme in Wade Davis's writings and talks. It goes something like this. If you think we're facing a crisis in biological diversity at the moment, we're facing a far more profound crisis in linguistic and cultural diversity. At the beginning of this century, there were something like 7,000 spoken languages on Earth. Now there are closer to 6000, and every two weeks or so the last speaker of yet another spoken language dies. As that language vanishes, a way of interpreting and explaining the world does too, since that's what languages are for. Each language is at least the hallmark of a distinct culture.

Cultures don't exist simply because some people prefer eating beef and others prefer beans, or because some people like Maori war dances better than classical ballet, prefer Irish traditional music to hip-hop. Languages and cultures reflect the need to adapt to different climates, sources of food, histories, conflicts, and aspirations. They're different ways of looking at the world that have been designed or have evolved to fit their context. And we have a lot to learn from them. Both Wade Davis in his book The Wayfarers[23] and Ed Hutchings in his book Cognition in the Wild[24] look at the Polynesian

---

[21] http://www.kaner.com/pdfs/KanerSocialScienceSTEP.pdf
[22] Davis, Wade, *Book of Peoples of the World: A Guide to Cultures*. National Geographic; 2nd edition, 2008.
[23] Davis, Wade, *The Wayfinders: Why Ancient Wisdom Matters in the Modern World*. House of Anansi Press, 2009.
[24] Hutchings, Edwin, *Cognition in the Wild*. The MIT Press, 1996.

seafarers, who without maps, compasses, global positioning systems, or modern ships, managed not only to settle the islands of the South Pacific, but also established and maintained a thriving exchange of goods and culture How did they do that? Not with our brand of science and technology, but by their capacity to observe the world around them, read the waves, follow birds, learn from each other, and to develop tacit knowledge and skills. Some of these skills are profoundly different from ours, yet they did the job splendidly for the Polynesians for hundreds of years.

Yet as cultures endure, they're not static either. They're always changing, frequently adopting or adapting ideas from other cultures, dealing with environmental pressures, responding to the world.

What does that have to do with us? As Davis says, the cultures that we might think of as primitive are not failed attempts to be modern. He says, "When asked the fundamental question, 'What does it mean to be human, mankind responds in seven thousand different voices.'" Well... alas, maybe six thousand.

There are cultures in software development and in testing. Whether we appreciate them or not, we're probably stuck with certain kinds of approaches and practices, at least for a while, because they reflect the environments and cultures in which they exist. The context driven approach, I would argue, not only allows but mandates that we must be cautious about condemning certain practices without understanding how and why they became what they are.

So we need to study anthropologists, and how they do their work. We need to study how to observe people in their settings.

5.2 Psychology

Psychology, the study of how our minds work, is also the study of they don't work so well, and how we fool ourselves. How we understand and deal with other people. How people can be motivated and persuaded, especially to deal with unpleasant truths that we'd rather not deal with. It also looks at things like empathy (or the lack of it), and establishing self-defence.

5.3 Philosophy

Why philosophy? Philosophy is the study of what we know and how we know it. There are those who claim that our work is "too philosophical"—which, as James Bach has pointed out, is a philosophical argument. But then let's go to Immanuel Kant who said that theory without observation, but observation without theory is blind[25]. That is, context is interwoven in what we observe and how we observe it. In order to give warrant to the product story, and the testing story, and our evaluation of the testing story, we need to examine our ontologies—our systems for defining and describing the world—and our epistemology—how we know what we know.

5.4 Design of experiments and measurement

We all bemoan bogus metrics. Yet, it seems to me, few of us have made a serious study of excellent measurement in the social sciences. (I can guess why people in other communities don't do that much. One reason is that they're busy learning the next programming language or the next automation tool. Or, worse, writing it.) We need to study measurement, and we need to impart what we've learned to each other and to our clients. Let's start with learning the difference between measurements and metrics: measurement is the art and science of making reliable observations[26] (that's Jerry Weinberg,

---

[25] "Thoughts without content are empty, intuitions without concepts are blind." Kant, Immanuel, *Critique of Pure Reason.* 1781,

[26] Weinberg, Gerald M., Quality Software Management, Volume 2 First Order Measurement. Dorset House, 1993.

there); quantitative measurement is the application of numbers, based on a rule or model, to attributes of objects or events with the intention of describing them[27]; a metric is a function—an activity combined with a mathematical formula or mapping—that connects the number to the attribute or event. All kinds of factors can confound variables and the relationships between them. God help us, we probably need a measurement spinoff from the Education SIG.

5.5 Journalism

If testing is finding out the story about the product, then I think we have to look to journalism to understand how journalists investigate, and how journalists construct stories. I'm not talking about anyone employed by Rupert Murdoch here.

5.6 Positive Deviance

Positive deviance is a relatively little-known approach to solving problems[28], and I believe our community should study it and apply it. The positive deviance approach presumes that in every group of people, with the same training, backgrounds, resources, and constraints, some people, at whatever level, deviate positively from the norm. The positive deviance approach sets up ways for those people and what they do to be observed, recognized, and celebrated—and put to work. It's the antithesis to process imperialism.

5.7 Safety Language

Safety language—I first learned about it from Jon Bach—is a precise, circumspect style of speaking and writing, intended to make clear the difference between observation and inference. It's informed by a determination to suspend conclusions, certainty, and judgment. In that, it's also form of tester self-defence. Judgment is always uncertain, and decisions about quality are based on politics and emotions. It's funny, in a way, because avoiding making a commitment sounds wishy-washy to some ears. Yet in my experience, speaking in safety language requires more discipline and commitment than the way we're used to speaking.

We don't really know what's going on inside the computer from one moment to the next. If that's the case, it's prudent for us to say "It seems..."; "it appears..."; "this time..."; "to me..."; "it looks like..." Instead of saying, "This is...", safety language prefers "It might be...", or "it could be...", or "one possibility..."; or "my current theory is..."

That carries into something that I call the "A vs. THE heuristic: instead of saying "The terrible bug in the system is..." it's wiser and safer to say "a terrible bug in the system is...". Instead of saying, "The root cause is...", we'd say "a root cause is..." or "this phenomenon contributed to the problem". After all, if you're looking for real root causes, the bug would never have happened if the programmer had never been born.

Safety language allows us to answer challenging questions gracefully. ""Do you use the principle of concomitant variation?" "Perhaps I do, but not by that name." Safety language simultaneously reminds us and others that what we know is heuristic: "I don't know how to determine the minimum number of test cases." rather than "It's impossible..."

---

[27] Kaner, Cem, and Bond, Walter P. "Software engineering metrics: What do they measure and how do we know?" 10th International Software Metrics Symposium (Metrics 2004), Chicago, IL, September 14-16, 2004. http://www.kaner.com/pdfs/metrics2004.pdf

[28] See http://www.positivedeviance.org; also, Pascale Richard; Sternin, Jerry; and Sternin, Monique, *The Power of Positive Deviance: How Unlikely Innovators Solve the World's Toughest Problems*. Harvard Business Press, 2010.

In our Rapid Testing class[29], James and I teach people that "No user would ever do that" really means, "No user that I've thought of, and that I like, would do that on purpose." Wouldn't things be clearer if programmers used safety language too?

Safety language is also a factor in making discussion and debate safe, too. One of the key phrases in safety language is, "I don't know." Follow that quickly with "Let's find out," and I predict that at least progress will ensue.

My mother once told me something along the same lines; if you want to conclude an argument or a dispute, one way to do it is to say, "You may be right." Some years later, I realized that was only the introductory-level approach to dealing with disagreement. The real trick is to be able to say, "You may be right," and believe it as you're saying it.

5.8 Rapid Learning

There's a stupidly large amount of stuff on the table already, so let me pile on just one more item. To handle all that stuff, we need to learn, rapidly, how to learn rapidly[30].

6 Conclusions

The large number of skills and practices; the open-ended nature of investigation; the fact that technologies are developing all the time, and the world to which those technologies relate is constantly changing too; all these things demand an expansion of our craft into other disciplines, and they welcome people from other disciplines into our craft. I hope that you'll join us. Thanks.

[29] http://www.satisfice.com/rst.pdf

[30] See Bach, James, *Secrets of a Buccaneer-Scholar: How Self-Education and the Pursuit of Passion Can Lead to a Lifetime of Success*. Scribner, 2009