System Design Frontier Journal
Featured Column

Why Do We Resist Testing?

Orit Hazzan and Uri Leron
Department of Education in Technology and Science
Technion - Israel Institute of Technology
{ oritha, uril } @tx.technion.ac.il

> *Work with human nature, not against it.*
> Kent Beck (2000, p. 116) on testing.

1. Introduction

The difficulties with software testing are well-known and well-documented; and they are unsettling: Every-body agrees that proper software testing is essential, yet there is not enough of it, and what there is, is not good enough. On both the individual and the team levels, a prevalent phenomenon of testing resistance can be observed. For example, Van Vliet (2000) states that "the testing activity often does not get the attention it deserves. By the time the software has been written, we are often pressed for time, which does not encourage thorough testing. […] Postponing test activities for too long is one of the most sever mistakes often made in software development projects. This postponement makes testing a rather costly affair. (p. 386-387). Hamlet and Maybee (2001) speculate that "maybe the reason testing is not always thought of as fun is that there's a flip side: the program may not work." (p. 397). Why do we resist testing, and why is testing resistance so hard to overcome?

In this column we try to offer an explanation by putting testing resistance in a wider context than just software development, starting with Karl Popper's view of the scientific method, moving to the so-called *confirmation bias* from cognitive psychology, and ending with *human nature* as re-defined in evolutionary psychology. By widening the scope we hope to gain better understanding, but will it lead to any applicable conclusions? In the last section we suggest that this understanding lends support to some of the solutions offered in the industry, especially test-driven development. Improvising on the Beck quotation at the top, we will propose that human nature is not all of a kind, and that we can enlist one part to help overcome a weakness in another.

2. Testing and the scientific method

One way to understand testing resistance is to think of your code as hypothesis, and of testing as trying to refute your hypothesis. To be specific: You have some expectations (or specifications) for how your code is supposed to behave, and the code you write expresses the hypothesis that it will indeed behave as intended. Testing, then, is an attempt to find disconfirming evidence for this hypothesis. This brings to mind Karl Popper's (1992/1959) view of *the scientific method*. On this view, the scientific method consists not of confirming the scientist's hypotheses, but rather of trying to find disconfirming evidence, and of looking for refinements or for alternative hypotheses that will better withstand the refutation attempts. In other words, scientists are busy *debugging* their hypotheses. This analogy is further borne out by the almost identical claims by Dijkstra (1972) ("*Program testing can be used to show the presence of bugs, but never to show their absence!"*) and by Popper (*a scientific hypothesis can only be empirically falsified but never fully verified* [1]).

Can this analogy help us understand testing resistance? We think it can. Starting at the 1960's, Wason and others conducted experiments that showed that people are not naturally Popperian, that is, they tend to look for confirming evidence for their hypotheses, and they avoid looking for alternative hypotheses and for disconfirming evidence.

> [Peter Wason] saw a key component of scientific thinking as being the testing of hypotheses. […] Wason focused on whether people adopt a strategy of trying to confirm or disconfirm their hypotheses. Using Popper's (1959) theory that scientists should try and falsify rather than confirm their hypotheses, […] Wason concluded that people try and confirm their hypotheses, whereas normatively speaking, they should try and disconfirm their hypotheses. One implication of this research is that confirmation bias is not just restricted to scientists but is a general human tendency. (Dunbar and Fugelsand, 2005, p. 707).

In the last 30 years or so, an extensive research program on judgment and decision making – the *heuristics and biases program* – was carried out by the cognitive psychologists Kahneman and Tversky (1982), for which Kahneman received the 2002 Nobel Prize in economics (2002). In this program they catalogued and explained a long list of *biases*, that is, "cognitive illusions" (in analogy with the well-known optical illusions), which are situations in which most people react in ways that do not conform to the standards of logic, mathematics or statistics. Some researchers (and most of the popular media) took these findings as showing that people are "irrational", but this conclusion was hotly debated by others who claimed that these are simply not the right norms for judging rationality. To avoid going into this argument (fascinating though it is; see e.g. Stein, 1996) it has become customary to refer to this kind of responses as *non-normative*. One of the items on this catalogue, and the one most relevant to our discussion, is the *confirmation bias*.

> […] philosophers, historians, and experimental psychologists have devoted a considerable amount of research to "confirmation bias." This occurs when scientists consider only one hypothesis (typically the favored hypothesis) and ignore alternative hypotheses or other potentially relevant hypotheses. This important phenomenon can distort the design of experiments, formulation of theories, and interpretation of data. Beginning with the work of Wason (1968) […], researchers have repeatedly shown that when participants are asked to design an experiment to test a hypothesis, they predominantly design experiments they think will yield results consistent with the hypothesis. […]

---

[1] This is our own synopsis. Compare e.g., "Agreement [of hypothesis and testing] is taken as corroboration of the hypothesis, though not as final proof; clear disagreement is considered as refutation or falsification". (Popper, 2002/1957, pp. 132-133).

> Confirmation bias is very difficult to overcome. Even when participants are asked to consider alternate hypotheses, they often fail to conduct experiments that could potentially disconfirm their hypothesis. (ibid, p. 709)

Thus it can be seen that we, with our testing resistance, are in good company. If testing software is like looking for disconfirming evidence for one's hypo-theses, then we can see that we share this resistance with the entire scientific community and in fact with most human beings. In this sense, Beck (2000, see opening statement) is indeed correct in claiming that testing goes "against human nature".

3. Is testing against human nature?

To say, as we did in the last section, that people resist testing doesn't mean that it can't be done. It only means that it doesn't come naturally, and that we have to invest some conscious effort to overcome this resistance. We are thus led to consider the relationship between thinking and behavior that come to us naturally, or intuitively, and that which requires conscious effort and analytical tools.

Just as we placed testing resistance itself in the context of scientific thinking, we will now try to understand overcoming the resistance in the context of a recent influential theory of intuitive thinking from cognitive psychology, called *dual-process theory*. According to this theory, our cognition operates in parallel in two quite different modes, called *System 1* (*S1*) and *System 2* (*S2*), roughly corresponding to our common sense notions of intuitive and analytical thinking. These modes operate in different ways, are activated by different parts of the brain, and have different evolutionary origins (S2 being evolutionarily more recent and, in fact, largely reflecting *cultural* evolution). S1 processes are characterized as being fast, automatic, effortless, "cheap" in terms of working memory resources, unconscious, and inflexible (hard to change or overcome). In contrast, S2 processes are slow, conscious, effortful, fully engage the working memory resources, and relatively flexible. In addition, S2 serves as monitor and critic of the fast automatic responses of S1, with the "authority" to override them when necessary. In many situations, S1 and S2 work in concert, but there are situations in which S1 produces quick automatic non-normative responses, while S2 may or may not intervene in its role as monitor and critic. For an accessible survey of the heuristics and biases research, including dual-process theory, see e.g., Kahneman, 2002. See Leron & Hazzan (in press) for similar phenomena in advanced mathematics thinking.

In terms of dual-process theory, we might say that testing resistance resides in S1, but that it's S2's job to overcome it. We know from the research literature that S2 does not continuously monitor S1 and that, in fact, it often remains dormant and fails in this role. In Kahneman's words:

> [This] illustrates how lightly the output of System 1 is monitored by System 2: people are not accustomed to thinking hard, and are often content to trust a plausible judgment that quickly comes to mind. (2002, pp. 451-452)

From this perspective, we can interpret Massimo Arnoldi (in Beck, 2000, p. 116):

> Unfortunately at least for me (and not only) testing goes against human nature. If you realize the pig in you, you will see that you program without tests. Then after a while, when your rational part wins, you stop and you start writing tests.

This quote can actually be interpreted as being about dual system: "If you realize the pig in you [S1] … you program without testing. … when your rational part wins [S2] … you start writing tests."

We now move to consider the question at the head of this section. We take from the young discipline of evolutionary psychology (EP) the scientific view of *human nature* as a collection of universal, reliably-developing, cognitive and behavioral abilities – such as walking on two feet, face recognition, and the use of language – that are spontaneously acquired and effortlessly used by all people under normal development, independently of culture, race, geography or education (Cosmides and Tooby, 1992, 1997; Pinker, 1997, 2002; Ridley, 2003). In contrast, walking on two hands or using a programming language are not part of human nature – not because they are impossible to learn, but because they are only learned by a few individuals with special motivation and training.

We also take from EP the evolutionary origins of human nature, hence the frequent mismatch between the ancient ecology of our hunter-gatherer ancestors to which it is adapted and the demands of modern civilization. To the extent that we do manage to learn many modern skills (such as writing or driving, programming, or mathematics), according to EP, this is because of our mind's ability to "co-opt" ancient cognitive mechanisms for new purposes (Bjorklund and Pellegrini, 2002; Geary, 2002). But this is easier for some skills than for others, and the ease of learning in such cases is determined by the availability and accessibility of the co-opted cognitive mechanisms.

We have previously placed "confirmation bias" – hence testing resistance – within S1, which would indeed make it part of human nature, since S1 is universal and "reliably developing".[2] In this sense, working against testing resistance is indeed, as in Kent's words, working against human nature. However, the existence of the "cognitive monitor" in our S2 is also part of human nature (though the specific skills of S2 are mostly not). As in other complex cognitive tasks, we can set this monitor to help us overcome the (S1) weakness in our nature. Hence our own elaboration of Beck's admonition: *Employ one part of human nature to overcome a weakness in another*.

4. Why Test Driven Development works?

We have argued that testing resistance is a special case of confirmation bias – the propensity to look for confirming evidence for one's hypothesis, and avoid looking for disconfirming evidence. The remarkable insight behind Test Driven Development (TDD) (though its developers may not have thought of it in those terms) is that by incorporating testing into the development, testing becomes part and parcel of "the hypothesis", and is no longer conceived as a search for disconfirming evidence. TDD thus helps developers avoid confrontation with this deep-seated confirmation bias, indeed it helps working "with human nature, not against it".

---

[2] We are referring here only to the universal parts of S1, excluding specific skills that migrate to S1 of specific individuals due to specialized expertise and training.

We have also seen that testing is a function of our S2 cognitive monitor, but that the monitor is not easily kept alert at all times. We propose that here too TDD may come to the rescue, since writing test prior to the code helps keep S2 on the alert.

5. References

Beck, K. (2000). *Extreme Programming Explained*, Addison-Wesley.

Bjorklund, D.F., and Pellegrini, A.D. (2002) *The Origins of Human Nature: Evolutionary Developmental Psychology*, American Psychological Association Press.

Cosmides, L. and Tooby, J. (1992). Cognitive Adaptations for Social Exchange, In Barkow, J., Cosmides, L., and Tooby, J. (Eds.), *The Adapted Mind: Evolutionary Psychology and the generation of Culture*, Oxford University Press, 163-228.

Cosmides, L. and Tooby, J. (1997) *Evolutionary Psychology: A Primer*, retrieved 24 October 2004, from http://www.psych.ucsb.edu/research/cep/primer.html.

Dijkstra, E.W. (1972). Notes on structured programming, in O.J. Dahl et al. (ed.), *Structured Programming*, Academic Press, London, pp. 1-82.

Dunbar, K. and Fugelsand, J. (2005). Scientific thinking and reasoning. In Keith J. Holyoak & Robbert G. Morrison (eds.), *The Cambridge handbook of thinking and reasoning*, Cambridge University Press, Ch. 29, pp, 705-725.

Geary, D. (2002). Principles of Evolutionary Educational Psychology, Learning and individual differences, 12, 317-345.

Hamlet, D. and Maybee, J. (2001). *The Engineering of Software*, Addison Wesley.

Kahneman, D. (Nobel Prize Lecture, December 8) (2002). Maps of bounded rationality: A perspective on intuitive judgment and choice, in *Les Prix Nobel*, T. Frangsmyr (ed.), pp. 416-499. Also accessible at http://www.nobel.se/economics/laureates/2002/kahnemann-lecture.pdf

Kahneman, D., Slovic, T. & Tversky, A (1982). *Judgment Under Uncertainty: Heuristics and Biases*, Cambridge University Press.

Leron, U. and Hazzan, O. (in press). The Rationality Debate: Application of Cognitive Psychology to Mathematics Education to appear in *Educational Studies in Mathematics*.

Pinker, S. (1997). *How the Mind works*, Norton.

Pinker, S. (2002). *The Blank Slate: The Modern Denial of Human Nature*, Viking.

Popper, K. R. (2002/1957) *The Poverty of Historicism* . Rooutledge.

Popper, K. R. (1992/1959). *Logic of Scientific Discovery*, New York: Harper and Row.

Ridley, M. (2003). *Nature via Nurture: Genes, experience, and what makes us human*, Harper Collins.

Stein, E. (1996). *Without Good reason: The Rationality Debate in Philosophy and Cognitive Science*, Oxford

Van Vliet, H. (2000). *Software Engineering – Principles and Practices*, Wiley.