

Zen, Innovation, Being Open, HackerSpaces [2009-11-05]

Interview for TheAge article: Robert Pirsig's "Zen and the Art of the Motorcycle" 35th anniversary¹

Along with some other local technologists ([Rich Evans](#)², [Con Zymaris](#)³ and [Veechi Curtis](#)⁴), I was interviewed (via email) by [Patricia Maunder](#)⁵ for an article in today's edition of The Age newspaper ("livewire" section of the Green Guide, page 22). It discusses and celebrates [Robert Pirsig's](#)⁶ seminal book "[Zen and the Art of the Motorcycle](#)"⁷.

*The on-line version of the article ... "Down to the nuts and bolts"*⁸

Being a keen technologist and motorcycle rider, I was pretty excited to be interviewed about the book and contemporary technology and put some serious effort into my replies. Of course, I expected that my extensive response would be heavily edited (for good reason) and so, I asked Patricia if she'd mind if I blogged my complete replies (once the article was published).

In case you are wondering what I'm holding in the article's photograph ... it is a [Bike Interceptor](#)⁹, created by a very talented local designer, [Marc Alexander](#)¹⁰. One of the best things about being part of the [Melbourne HackerSpace](#)¹¹ has been meeting many people like Marc, who have incredible technical skills and a generous desire to share their experience. There is a lot more innovation going on by individuals and small companies in Melbourne than any of us realize.

Fortunately, Patricia did an excellent job of extracting the pithy comments (from all interviewees) and created an article that captured what I'd hope to convey, but in a much more concise way. I was pleasantly surprised that the important themes of openness and hacking as a means for self-education, self-empowerment and innovation came through so strongly.

Q: Briefly describe what you do, and what aspects of technology interest you professionally and personally.

I'd say that my vocation is a distributed systems engineer, which means that I design and develop networked computer software and (to a much lesser extent) hardware, as well as lead technical teams in that domain.

As a kid, I was fascinated with building small computers connected to real world devices. Throughout my career, I've been able to pursue my interests, sometimes with [extraordinarily powerful computers](#)¹², and at other times with increasingly [small, networked devices](#)¹³. Fortunately, I've been around knowledgeable people from whom I've learnt and shared that passion.

These days, my personal and professional interests are aligned. I'm both working on Internet based systems, like [WatchMyThing](#)¹⁴, as well as participating in local technology groups, such as [Melbourne HackerSpaces](#)¹⁵. The combination of these efforts are an attempt to measure our environment, acting as

1.

<http://geekscape.vox.com/library/post/interview-for-theage-article-robert-pirsigs-zen-and-the-art-of-the-motorcycle-35th-annivers>

2. <http://rich-evans.net/>

3. <http://white.cyber.com.au/conz/blog>

4. <http://twitter.com/veechicurtis>

5. <http://users.tpg.com.au/maunderp>

6. http://wikipedia.org/wiki/Robert_Pirsig

7. http://en.wikipedia.org/wiki/Zen_and_the_Art_of_Motorcycle_Maintenance:_An_Inquiry_into_Values

8. <http://www.theage.com.au/technology/technology-news/down-to-the-nuts-and-bolts-20091104-hwfg.html>

9. <http://www.bikeinterceptor.com/>

10. <http://twitter.com/mtronic>

11. <http://groups.google.com/group/connected-community-hackerspace>

12. <http://www.dsd.gov.au/sigint/supercomputers.html>

13. <http://arduino.cc/>

14. <http://watchmything.com/>

a catalyst to positively changing our behavior, and improving the world.

It's also challenging and fun to be involved in groups like [Lunar Numbat](http://lunarnumbat.org)¹⁶ (<http://lunarnumbat.org>)

Q: Are you the kind of person who reads instruction manuals, or tosses them in the bin ? Does it depend on the nature of the product you've just acquired ?

The best consumer products are designed so that you can get going quickly, without having to trawl through detailed instructions. These days, most products containing electronics have sophisticated features that do require you to eventually read the manual to get the most from your equipment.

When it comes to engineering using third-party technology, you risk misusing or damaging something, if you are too impatient to read the manual before leaping in.

I prefer instruction manuals to be on-line where you can always find them when needed. Much better than hoarding, then eventually losing large collections of booklets and CDs.

Q: Is there some technology, old or new, that you like to be really hands-on with, such as maintaining a vintage car or building computers ? Why ?

Modern desktop and laptop computers, even mobile phones are too complex for any one person to completely understand from the base hardware, all the way through to the complete stack of software. It takes a large team of people and many organizations to design, build and maintain such things.

However, there are [smaller computers](#)¹⁷ and [networked devices](#)¹⁸ that are both comprehensible by a single person and can be used to perform interesting tasks. A system-on-a-chip computer can be bought for a couple of dollars, connected to some inexpensive sensors and actuators, and some control software written during a [casual afternoon on the weekend](#)¹⁹.

Best of all, these types of projects are being shared amongst a large and growing community of software and hardware hackers. The [good kind of hacker](#)²⁰, of course.

You can experiment with home automation, 3D printing, wearable computing, augmented reality and vehicle embedded systems, including putting some [telemetry onto your motorbike](#)²¹ !

This takes me right back to where I started as a kid ... noodling around with technology. Except these days, we have much better access to information and knowledgeable people via the Internet, much more open-source software and unbelievably capable and cheap hardware.

Q: The fact that you ride a motorcycle will be great for the story ! Has this been a long-term interest? Do you maintain and repair your bike ? What does riding mean in your life, and is there a connection with /Zen and the Art of Motorcycle Maintenance ?

I first started riding motorbikes over 25 years ago, starting with [trail bikes](#)²², then moving onto [road bikes](#)²³. Early on, I was shown how to maintain, service and repair my motorbikes by a friend. This is a pretty important skill to have, when you are stuck out in the bush, with some sort of problem. In my case, it was often from drowning the bike, when attempting a deep river crossing.

These days, due to other priorities for my time and the sheer technical sophistication of modern sports bikes (along with the need for specialist diagnostic equipment and manufacturers who don't share maintenance information) ... I don't service my motorbike, any more. It's still important to understand your vehicle, so that you can let the service people know what the problems are, and to be certain that they have repaired it properly.

15. <http://groups.google.com/group/connected-community-hackerspace>

16. <http://lunarnumbat.org/>

17. <http://arduino.cc/>

18. <http://openwrt.org/>

19. <http://groups.google.com/group/aiko-platform>

20. http://en.wikipedia.org/wiki/Hacker_%28hobbyist%29

21. <http://www.bikeinterceptor.com/>

22. http://en.wikipedia.org/wiki/Honda_XR250R

23. http://en.wikipedia.org/wiki/Fireblade#2004_CBR1000RR

However, performing basic checks (tyres, brakes, chain, cables, fluids) and maintenance is something that I still do ... as should every bike rider, for their own safety and ease of mind.

Not long after I started riding motorbikes, I read "Zen and the Art of Motorcycle Maintenance". Soon after, I got my dream job, co-incidentally working for a [company based in Minnesota](#)²⁴, which is where the story in the book begins. The company I worked for was legendary for their focus on [engineering quality and performance](#)²⁵.

The best way to experience motorcycling riding, is to head out into the country, leave the [main highway behind](#)²⁶ (straight, safe and boring) and ride the secondary roads (curvy, unpredictable and interesting). Unlike driving around in a car, you're not in an enclosed container, but rather you are part of the environment. Staying at old hotels, like the [Dargo Pub](#)²⁷, is a real experience.

The book still rings true. The first few pages of "Zen" describe the experience of motorcycle touring perfectly, including the not-so-comfortable parts and interactions with other people. Pirsig provides a fresh perspective on the differences in how people relate to solving mechanical problems and understanding modern technology. Always observing and "listening" to your motorbike, any piece of equipment, and most importantly your own body and mind ... is exceptionally good advice.

Q: What ideas in this book have had an impact on you?

On the inside cover, Pirsig says "The real cycle you're working on is a cycle called 'yourself'". So, I didn't have to look very hard to discover the basic message of the book is about self-improvement. Of course, different people will interpret his dialogues and metaphors to suit themselves.

The most important concept for me was the notion that there are various ways of perceiving the world, and that different people may have incompatible approaches for processing what they see and how they tackle problem solving. However, there is great value in understanding those differences and when possible reconciling them.

Another common thread concerns having a good understanding of how a given technology should work when functioning correctly, especially those technologies that you rely upon. Then, whenever you notice a deviation from normal operation, e.g. something sounds different, then performing a systematic diagnosis and pro-actively repairing the fault is the best approach.

Q: In Zen, Robert Pirsig argued that having an understanding of how tech-based things work, and being able to maintain and repair them, is important for one's quality of life in the modern world. Do you agree? How relevant is this view today, given that intuitive interfaces are critical to many products' success, and instant obsolescence often makes maintenance and repair irrelevant?

I completely agree that being able to understand, repair and best of all customize (hack) technology to suit new purposes, provides a better quality of life. This empowers people to innovate and helps them avoid feelings of powerlessness in this technological age. This doesn't just apply to high-technology such as electronics, but also to any activity that involves the use of tools or craft skills.

Of course, no one person can have all the experience required in every available technology. Therefore, people with different skills should band together and based on common interest ... collaborate to create or repair things that they could not manage by themselves. This can occur as a community, using the Internet to effectively self-organize.

I believe that the need for communities and individuals to educate themselves (throughout their lives) will continue to become more essential as time passes. Many of the problems facing the world today, need to be solved independently of any industry vested interest or maintenance of the government status quo. Additionally, Australia needs to innovate to remain competitive. This can only occur with an educated population, where education and self-improvement doesn't stop at the end of school, college or university.

Intuitive interfaces, such as those on modern electronic devices (rarely done right), typically provide an

24. http://en.wikipedia.org/wiki/Cray_Research

25. http://en.wikipedia.org/wiki/Seymour_Cray

26. http://en.wikipedia.org/wiki/Blue_Highways

27. <http://dargohotel.com/>

abstraction away from the underlying internal mechanisms. By necessity, good end-user interfaces simplify the operation of the device. An unfortunate side effect is that general users of technology have no idea about how an appliance really works, which means that they may not know how to use it most effectively or repair it when it breaks.

Hiding the internal operation by an intuitive interface is compounded by the manufacturer's natural tendencies to protect their product from competitors and to increase sales by ensuring that broken appliances have to be replaced, not repaired.

Now that pollution, insufficient recycling or re-use and limited resources are now our biggest problems, wasteful approaches can no longer be accepted, if we wish to maintain a reasonable quality of life.

Everyone should insist on open-standards for all infrastructure, and whenever possible, [open-source design](#)²⁸ and good documentation for all appliances, equipment and tools ... so that we can repair and re-use them as much as possible. This also enables the next generation of kids to be able to learn about how things work, pull them apart and put them back together again in new and interesting ways.

You can't do this with black-boxes.

We should have a [right-to-repair](#)²⁹ anything that we purchase.

Q: Can you give examples from your professional and/or personal life where a failure to grasp the technology behind a product or process has had a negative impact on an individual or organization?

Computer back-ups !

Everyone "knows" that mechanical devices fail eventually and that this applies to computer hardware, especially spinning hard disks. However, many people, including some in the computer industry (who should know better) fail to protect their precious information properly.

I think that people don't have a solid enough understanding of the certain likelihood of failure (over a long period of time) based on the many ways in which a computer can break both at the hardware and software levels, as well as taking into account good old human error.

This is compounded by a lack of understanding of the correct approaches to properly backing up and recovering a computer system. Although, computer vendors are gradually addressing this problem, e.g. Apple's Time Machine.

A new problem for everyone to deal with is that more of their precious digital information is being stored on the Internet, in web services like Google, Flickr, Twitter, Facebook ... with little understanding of the data retention and recovery policies. This is going to catch out a lot of individuals and organizations in the future.

Q: And examples of the reverse, where there have been benefits from putting in some effort to understand, if not manage, back-end technology?

Over recent years, around the world, groups of people have formed what are called "[HackerSpaces](#)³⁰". These are collectives that share a physical space, equipment and knowledge from many different domains. Their goals are usually to learn how to use technology to create new and interesting things.

Typically, these groups bring together people with an interest in electronics, software, mechanics and performance art to either re-work existing equipment to do things it wasn't originally intended to do ... or to build things using recent developments, such as inexpensive [3D printers](#)³¹.

The goals of a HackerSpace are usually to "have fun, learn and collaborate" and within that context a lot of innovation occurs, often on a very low-budget.

Q: Pirsig wrote that art and technology have been divorced from each other in the modern world,

28. <http://en.wikipedia.org/wiki/Open-source>

29. <http://www.righttorepair.org/>

30. <http://hackerspaces.org/>

31. <http://en.wikipedia.org/wiki/Makerbot>

that people have become divided between 'in the moment' Romantic thinkers and rational, practical Classicists. Would you say you are firmly in one camp, or do you endeavor to straddle the two – for example, I expect you are primarily a rational, practical person, but do you also have faith in seemingly irrational concepts such as intuition?

My personal inclination and experience have always leaned strongly towards the Classical camp. However, I appreciate that to achieve a completely satisfying outcome, one must also consider the aesthetics of the final result.

Creating a high quality software design requires a systematic approach to ensure that the end result is fit for the intended purpose. However, you'll also hear some software engineers talk about beautiful or elegant solutions, which strongly suggests that aesthetics also has a part to play. Often this means that the design is efficient, has clarity, is flexible and is cohesive. It must be strange to see software developers looking at a page of ugly, incomprehensible symbols and hear them say "that's beautiful" !

Xerox Parc³² had an "Artist-in-Residence Program", which paired artists with researchers to produce new innovations combining art and science. I think this is a great idea that more organizations should emulate. I'd suggest taking that a step further and pairing an engineer, an artist and a young member of the team, so that skills can be passed onto the next generation ... and that the senior staff are kept on their toes :)

I believe that intuition is the ability to make quick decisions based on past personal experience, or communal shared experience (aka folklore). In situations where you don't have sufficient time to figure things out from first principles ... "going with your gut" is a very rational means of coping.

Q: Will the future be populated by people who self-install open-source software, or those who press the 'on' button of equipment with pre-installed corporate software? Why will that future be good or bad, philosophically speaking?

Both. And, often people who do have the ability to manage things at a deeper level will simply want to "press the 'on' button".

As long as everyone has the *choice* to interact with their equipment as either an end-user (just at the interface) or at a deeper level (inside the box), as their circumstances require ... then, that is a good future. Having your choices artificially restricted by a third-party who has conflicting interests compared to yours ... is a bad place to be.

Typically, appliances should just work and perform simple, every day operations, without having to refer to the manual or the source code. However, should an appliance not operate in a way that you need, then you should be able to improve it or engage someone who can do that for you. You should be able to interconnect your appliances, without having to break down barriers artificially introduced by different vendors to lock you out of your own possessions.

Prior to electronic or software control, all appliances could be disassembled by an individual and their operation understood by inspection. As more appliances use software to control their operation, it is becoming impossible to understand the operation of these devices without also being able to read the software source code. This is why open-source software is so important (to everyone, except the vendor who wants to keep it a secret).

In a world where the majority of computers and appliances are controlled by closed-source software, we will have lost control of the things that have become essential to our current life-styles and businesses. And, the next generations of Australians will have greatly reduced options for innovation and making much needed changes in society.

Q: Is there anything else about the book, your own experience of technology, or the general topic of getting behind technology's interfaces that you'd like to comment on ?

The most important thing to remember is that it costs very little, close to zero these days, to share knowledge and information with the whole world. We should be encouraging our kids to create, innovate and solve problems ... without shackling them with increasingly draconian laws that only protect old-school special interests.

Instead, we should encourage kids to take things apart, understand how things work and share the experience. Along with combining different perspectives and approaches, such as artistic sensibilities

32. http://wikipedia.org/wiki/Xerox_Parc

and solid engineering.

Copyright Notice

This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 2.5 License](http://creativecommons.org/licenses/by-nc-sa/2.5/)¹



Fields

Name	Value
Type	Various ideas

1. <http://creativecommons.org/licenses/by-nc-sa/2.5/>

2. <http://creativecommons.org/licenses/by-nc-sa/2.5/>