

# Hacking is a Mindset, Not a Skillset: Why civic hacking is key for contemporary creativity.

 [blogs.lse.ac.uk/impactofsocialsciences/2014/01/16/hacking-is-a-mindset-not-a-skillset/](http://blogs.lse.ac.uk/impactofsocialsciences/2014/01/16/hacking-is-a-mindset-not-a-skillset/)

1/16/2014

*From MIT's famed pranks to Silicon Valley's approach to design, core values drive the clever, ethical, enjoyable, excellence-seeking behaviour of a civic-oriented hackers mindset. [Tanya Snook](#) makes the case for everyday hacking and provides five principles that you can use to rethink situations, re-evaluate problems, and hack everything you do.*

When I say “hacker” what images come to mind? Some pimply-faced kid in a dark basement, breaking into a high security website to post a picture of a LOL cat? Or a hoodied twenty something male typing furiously with his Guy Fawkes mask beside him, liberating corporate, government or military documents in the name of Anonymous? Using AutoCad software of course, as the movies would have us believe.

But how many of you have referred to yourselves as a hacker? You probably should. My goal is to either convince you that you've been hacking all along or that you really should be hacking. But why hacking? Much like gamification is the application of game design principles to non-game uses, the principles of hacking can be applied to non-hacking uses. And I don't mean by sitting at a computer every day; I mean by hacking in the true sense of the word.

You see, originally, hacking had nothing to do with computer programming: In fact, “hack” was originally a term used to describe pranks performed by MIT students: their pranks are projects or products that are completed to some end, but that also afford the participants some enjoyment by the mere fact of participating. The [MIT hackers](#) describe what we call “hacking” as “cracking”. When the MIT hackers hack, their goal is to devise “a clever, benign, and ‘ethical’ prank or practical joke, which is both challenging for the perpetrators and amusing to the MIT community.”

The website (there's always a website) dedicated to cataloguing the pranks is actually called the Hack Gallery ([hacks.mit.edu](http://hacks.mit.edu)). The list of hacks includes lining a hall with invading green army men, installing a lunar module on top of the Great Dome, setting up a giant statue of Athena on a main lawn, and my personal fave, hanging mannequins on high wires and trapezes from a ceiling along with long flowing ribbons, to turn a lobby into a circus.

Along these lines, I found a fantastic quote in the Wikipedia definition of “hacker”: “Hacking entails some form of excellence, for example exploring the limits of what is possible, thereby doing something exciting and meaningful. Activities of playful cleverness can be said to have “hack value.” Hack value. What have you done today that can be said to have “Hack Value”?

Now within the mainstream, the term hacking has been appropriated to describe people who hold these same values and do computer programming. But the principles of creativity and cleverness are still there, although the fun may be more intrinsic than obvious to us the outside observer.

It's what Silicon Valley is built on. A recent [Harvard Business Review blog post](#) described Silicon Valley as having a culture that believes “things are hack-able— that the way we've designed various systems is not pre-ordained or immutable. We can tinker, re-design, and play with them.” It went on to say that participants in Silicon Valley companies “don't ask for permission to do what they do... They are less interested in technologies per se than in playing with established ways of doing things and conventional ways of thinking, creating, learning, and being.”



They've adopted a hacking mindset. They translate this clever, ethical, enjoyable, excellence-seeking behaviour to their everyday lives. See? Hacking is a mindset, not a skillset. When you seek, in your everyday life, to deliberately find opportunities to be clever, ethical, to enjoy what you are doing, to seek excellence, then you're hacking. Now the key here is that this behaviour is deliberate. Not a happy accident. If you aren't acting this way deliberately, then we need to change your thinking and behaviour a little bit in order to make this your default MO. In fact, your mission for tomorrow should be: "Do something that has hack value."

I recently wrote about the science of behaviour change in a [post on New Year's resolutions](#). Behavioural science has indicated that in order to effect behaviour change, you need to break things down into small tasks. Successfully completing tiny tasks is a necessary step in making new behaviours stick. With that in mind (and I've probably oversimplified things), I've boiled hacking down to 5 principles that you can keep in mind, and incorporate into your day to day thinking.

The Principles of The Hacking Mindset (according to spydergrll)

Yeah, these should probably always have the "according to spydergrll" qualifier since I'm sure that there would be many people (hackers and crackers alike) with their own opinions, happy to jump on the Interwebs and tell me I am wrong:

### **1. Challenge accepted! (Barriers are welcomed)**

This is part of the "clever" component to hacking: accepting barriers as sources of motivation in their own right, motivating you to find a solution to break through them; the feeling of accomplishment at the end being its own reward.

Take for example the origins of crowdfunding, especially through sites like Quirky, Kickstarter, Indiegogo. A lot of the projects funded through these sites wouldn't necessarily get backing by venture capitalists or even banks in order to become a reality. Going directly to consumers to get what essentially amounts to pre-purchases of products and services is a fantastic way to hack the economic system and traditional business models.

In 2002, one of the founders of Indiegogo co-produced a play. It was popular with audiences but not exactly self-funding so they decided to look at other ways to raise investment capital. Indiegogo was born and launched at

Sundance in 2008, initially geared to raising money to fund films. It's evolved over the past 5 years to support all sorts of projects: even local communications company MediaStyle is using it to crowdsource funding for a new collaborative event space in their building.

Kickstarter is a bit different because people don't "invest" in Kickstarter projects to make money. They "back" projects in exchange for a tangible good or experience. One example is local tech company teknision who used it to fund and launch their Chameleon app for Android.

Now let's think about you and a barrier you might face at work. At some point in your career (or maybe your whole career) you've likely been told that there are no or very little funds for training. And let's assume that you don't have the wallet to fund your own training. So how do you creatively make up your own training? Thankfully there are plenty of other people who believe in the freedom of information which means you can access free webinars, online articles, and library books. You can find or build a network of people who have the skills you need: joining meetups, social networking, etc. And as you learn, you can find speaking opportunities to test your ideas and your learning.

That's one way. But you could also jump right into projects that will stretch your skills and knowledge: At work, advocate for projects you think are important and if you don't get approval, do them anyway! Partner with people in your network to grow your skills, get experience in a new field. Can't do what you want at work/ in your job? Find a way to collaborate with other communities of interest in your organization. Or find a pro bono project that will let you do what you love. I've done some pro bono in the past because I have a skillset that wasn't being tapped into at my job. So, through a friend I got referred to work with a local charity to donate my time and keep those skills up to date. When you find opportunities like this to hone or maintain your skills, you can put it on your resume. And then, hacking your training could even lead you to hacking your career.

Phew. This hacking stuff is a lot of work. Ok, let's step back and start a little smaller.

You get home from work, (stay with me), you get home from work, look up the recipe you want to make tonight and notice you have all the wrong ingredients. But if you throw away the recipe and modify your expectations to make something with what you had, you'll realize that you had all the right ingredients all along. Just the wrong recipe.

Those are the small tasks I was talking about; tiny wins that can help you make hacking your default behaviour. A barrier might only exist due to a perception issue: do you have the wrong tools or do you have the right tools but are looking at the problem from a particular perspective which is limiting your ability to think beyond it? Which brings me to my next principle...

## **2. Blow away the box. Look for unexpected ways to make something better.**

You've probably heard the adage: if you have a hammer, everything looks like a nail. Need proof? Watch a kid walk around the house with a toy hammer. You know it's true. The problem as adults is that we become too practical and start to limit our thinking: only nails look like nails. Sometimes, finding the fun or creative way to approach to something, means that we need to step back to the place where everything looked like a nail.

Take social media, for example. If you've ever had to explain to someone that Twitter is not millions of people posting their lunch, and that there is a lot to gain from investing a little bit of time each day in reading blogs or answering questions on Quora, then you know exactly what I am talking about. Those people think nails are nails: networking happens at formal corporate events, training takes place in boardrooms and classrooms, and it is necessary to respect the hierarchical chain of command. Riiight.

But you know that everything is a nail. You know that you will find out much more in 5 minutes on Twitter first thing in the morning than by spending an hour reading one newspaper or magazine. You have figured out how to get value, how to find what you need out of a noisy, messy system where others only see chaos. Which is why you follow hundreds if not thousands of thought leaders on Twitter. You've chatted with them in the comments section of their blogs. You've rubbed elbows with them (or \*gasp\* senior executives) at informal, non-work-related networking

events (aka meetups).

Maybe you've offered yourself up as a reverse mentor to share your knowledge with your senior executives. Don't have a reverse mentoring program at work? The basic idea is that working level, even entry level, employees mentor the executives in a subject area that execs are less exposed to. It could be anything: social media, user experience, web development, new classification trends. In turn, it gives staff the opportunity for some face time with those they might not see very often. Win-win. If you have a reverse mentoring program, get in on it. If you don't, start one. Chances are, you'll probably skip a couple of rungs on your way up the ladder, yourself.

See, you're already hacking.

Now, going beyond thinking outside the box, and actually blowing away the box can be difficult because it requires you to know who you are. That means not defining yourself by how others define you. Take for instance Ada Lovelace. She was the first reknown woman mathematician and computing specialist, back in the 1800s, when men were men and women were at home.

You're not just a "fill-in-the-blank-with-your-title-here," you're a mentor, a teacher, a coach, an advisor... Hacking your approach and blowing away the box means not allowing your thinking to be limited by the rules and restrictions that others put up around you. It means being confident enough to take a risk, and trusting your gut especially when it seems to have more conviction than your mind.

For example, maybe your library looks like a library but deep down it's actually a seed bank or a planetarium or even an open source community centre. Let's start with the seed bank. Basalt, Colorado has a public library that has started a seed bank. Not just to catalogue and store the seeds, but to enable citizens to check out seeds and plant them at home! In exchange, the borrowers take the best produce from their crops, harvest the seeds and return them to the library for someone else to use. The seed bank is expected to evolve since some crops will succeed and others will fail, the strongest always coming back into the bank. Over time, it could evolve to serve as an effective, proven index of the vegetation that is best suited to the climate and soil in that region.

If all libraries opened up seed banks, they could catalogue and store all of the best crops for each region of the country, an extremely valuable inventory in times of drought or other natural disaster. Not to mention, since librarians are masters of classification and information management, you just know those seeds would be classified and organized like nobody's business. It's a match made in info-science heaven.

For the most part, blowing away the box may mean changing your perception, taking risks and testing out assumptions out on a limb. But if you have built a strong, supportive network and if you have crowdsourced your ideas, then chances are you won't be alone when it's time to challenge traditional approaches. And that's another great thing about hacking: hackers have each other's back.

### **3. Bring your friends. Unique perspectives create more robust solutions.**

A colleague of mine recently tweeted that "we cannot solve a complex problem with a solution from a single discipline of study #justsaying" — Ralph Mercer (@ralphmercer) December 4, 2012. Which is one of the reasons why I love hackfests. Now, I don't consider myself a coder. I think of my abilities in HTML, JavaScript and CSS as coding-light. So when someone suggested I attend a hackfest, I thought that was probably one of the most intimidating ideas ever.

And then I worked as a business analyst in R&D, collaborating with technical architects and developers, whiteboarding solutions and brainstorming development approaches. And I realized that going to a hackfest would be no different. As I wrote late last year about LearnHack YOW, hosted by OpenData Ottawa, it takes all kinds to come up with a solution. You need experienced developers, analysts, user experience types and especially users. Non-hackers are as important to a good hackfest as hackers are... how else will you know if your solution will be useful and usable outside of the hackfest bubble?

Now think about your local library. What if it was an open source loving, hackfest hosting community centre? Maybe it has a ton of computers that were once used for looking up physical copies of books, but with e-readers and digital media taking over, they lie dormant most days. So what if they were offered up as web terminals for hackfests. The library could line up extra sets of tables and chairs where hackers and non-hackers can come together to talk about their open data needs and brainstorm app ideas using the City's data. Your data. Maybe they could even host hackathons for Little Geeks, encouraging them to think about data and how it can improve not only their lives but that of those around them. Maybe our libraries rekindle their role as the heart of the community; a gathering place for people and the birthplace for new ideas.

Even the MIT hackers understood the importance of being a part of something bigger than yourself. It's why people join social groups, volunteer, even play massively multiplayer online role playing games like World of Warcraft or Halo. It's even why people play Farmville and tend to each other's crops.

Outside of the development space, there are plenty of examples of collaborative solution building, or as we now refer to it: crowdsourcing. The first time I came across it was back in the mid-2000s with the Dell Social Innovation Challenge. This was a contest open to post-secondary students across the entire globe, in which they submitted ideas for social change. They posted them to an interactive site, and the general public could vote them up or down. The prize was a scholarship, and the potential to work on their idea.

More recently, there was the public consultation on the Icelandic constitution. If you didn't check it out, it was posted on a wiki, so anyone could log in and make changes. All of the changes were recorded in the history, so it was possible to sift through the evolution of the document over time (and revert to a previous version if it got vandalized).

And you? If you need to work through a question or an idea, if you need to whiteboard an issue, who do you turn to? Have you built up a network of resources that you can tap when you need help? Maybe you have, and you call on subject matter experts for help with work, with technical knowledge. But what about calling on someone who has absolutely nothing to do with the project/ department? No experience or subject matter expertise at all? Would you ask your mom for help on a work issue? Probably not.

Maybe your question seems technical but actually has an interpersonal angle you just aren't noticing. Maybe your personality conflict has an underlying technical issue. In the same way that a hackfest needs to be multidisciplinary and reach across all sorts of stakeholder groups to be the most effective, maybe you should reconsider your network and include people who can bring in completely new perspectives, and are more than happy to share their knowledge to help you evolve your thinking. Which brings me to point #4...

#### **4. Give it away now. Information and knowledge should be shared openly, freely.**

Sharing information empowers others to change the world. Well, at least their part of the world. Now I may have a bit more of an extreme view of these things than the average person, but I believe that information deserves to be free; research, knowledge, history, these are all beneficial to all of us. And hoarding them or locking them down in the name of intellectual property or profit is counter-intuitive to innovation.

Now you might not be able to decide what to do with the information or data that you work with if you're employed by someone else, so how can you infuse this giving nature into your own life? By giving away your knowledge: post it, assign a creative commons license, do pro bono work, make it open source, give away your expertise.

If you've never used a creative commons license, it's like a copyright on intellectual property that allows others to build on your content as long as they reference you. There are several tiers of licenses that can apply to any IP such as images, content, and media. There are even sites like ccMixter where people will post music for re-use (you will often hear CBC mention creative commons music credits). You might have even used Wikimedia Commons (with full attribution, of course), which is a site where people will share their media for re-use.

But you can also license your own content on your own site. For example, when I post my presentations, I do so with

a Creative Commons attribution. I post my full scripts and slides, and open them up so that anyone else can re-use or riff on them at their leisure, as long as they credit me for the original content.

You can also look at community activities that foster information sharing, some easier than others: I mentioned pro bono but you might also consider a for-donation project, where you donate the proceeds. One that you might have heard of (or even been part of) is the 100 Strangers Project by Kim Usan. She took photos of 100 strangers over the course of a year, blogged their stories and then did an art show and sold a coffee table book. All the proceeds went to a charity that was near and dear to her heart.

Closer to home, and more manageable on a day-to-day basis, you can contribute to open information, open source and open data projects. You could also be contributing to community-knowledge wikis like wikipedia or those for makers or even Lego lovers. Or contributing information or classification to all sorts of collaborative projects. Don't know where to start? Check out Code4Lib, which is a collaboration of coders for libraries and libraries for coders working to evolve libraries and library tech. Now, that said, you don't need to be technical to contribute (you should know by now that it doesn't take a developer to hack) and the best part is that they share their projects openly for re-use.

There are also some really fantastic and surprising open source projects. Like [Stellarium](#), an open source planetarium app. It's maintained by the development community and can run off a simple desktop, allowing users to pan and zoom skies all over the world. If you're an astronomy buff or are multilingual, they need your help developing new catalogues and translating their content. Oh, and the app is free and openly available for use. Imagine dedicating a space in our school libraries to showcase the skies, with computers that allow visitors to test drive the heavens for themselves. Did I mention that it's free? And it's actively maintained and it contains more than 600,000 star catalogs. Bet we don't have anything like that on our library shelves right now. If it's good enough for planetarium projection systems, it must be good enough for our schools.

Need more ideas on sharing your knowledge? You could openly blogging on your areas of expertise, organize a community learning event or unconference, teach a class. You could even donate some time to your kid's school to teach them something in your area of expertise because we need to...

## **5. Pay it forward. Teach the next generation to think like a hacker.**

Hubby and I are raising a hacker. The running joke in our house is that The Dude will get arrested for hacking into some website when he's 17 and he'll tell the police that his mom taught him how to hack. But that's not the type of hacking we're teaching him. We're teaching him to hack his thinking, to think like a hacker to solve problems.

The whole idea of teaching him to hack actually started last Christmas. I found a set of instructions to build a Star Wars Lego DeathStar tree ornament. Hubby and the Dude took a look at the instructions and decided that they didn't have the right pieces to build it. We have 10,000 pieces of Lego. And this ornament used about 100. So, I got a bit frustrated when I heard this. They were really limiting their thinking.

So I asked the Dude if he knew what hacking was, and told him that we were going to hack the Lego. I made him think up all sorts of combinations to substitute the pieces we didn't have. I made him figure out how we could rework the instructions to suit the pieces we could find in his Lego bin. He was so proud when we finished building it. He even started to refer to himself as a hacker. These days, he knows exactly what I mean when he brings me a problem and I tell him to "hack it". He knows that means to look at the problem again and to reimagine the solution in a new way.

Kids have the most unique perspectives; their thinking is not limited as is ours, they are very perceptive. Some of their ideas are crazy but given a real problem, they will often come up with very real solutions. The Dude begged me to bring him to a hackfest a couple of months ago and it was one of the most fun activities we've done together.

I explained all of the data sets to him in simple terms and had him design his own app concept on paper. He didn't

draw it, but wrote down the functions and features that it would have. It might have been easier for him to just draw a picture but by making him think about the functionality, he had to spend a little more time thinking from the user's perspective and understanding what they would want out of it. And it got me thinking about hackfests in schools. It would be so easy to bring a simple hackfest into the classroom and have kids make up their own apps. Any time it's possible to demonstrate the real-life applicability of their studies, kids seem to get more engaged in their learning.

But teaching a hacker, fostering hacking in the next generation doesn't have to be so formal. It can start small with game or toy hacking, in our case, Lego hacking. We'll often check out Brickipedia (yes, a Lego wiki!) for ideas and instructions for sets we don't own. We make him Google ideas, science experiments and definitions when he has questions. We try to turn his questions into opportunities for problem solving and hacking.

And this. If some of you read today's post, you'll know what this is. It's the walker of a 4 year old who recently had surgery on both legs. He hated his walker and didn't want to use it for therapy because he associated it with pain. A friend of his mom's commented that the shape reminded her of an At-At walker. So a friend of the family, who happens to be a cartoonist, hacked it and turned it into an At-At walker. Needless to say, the kid loves his walker now and especially walking around the house pew-pewing his family. A little hacking goes a long way.

It's important to teach kids to hack and actually call it that; to explain to them the importance of reinventing problems, working around constraints, and modding whatever they've been given. At home, at school, when our kids says something is impossible, make it a priority to prove them wrong. These are skills that can serve them their entire lives. And it starts with us modeling the behaviour, and hacking ourselves.

There you go. The Principles of The Hacking Mindset (according to spydergrll)

1. Challenge accepted! (Barriers are welcomed)
2. Blow away the box. Look for unexpected ways to make something better.
3. Bring your friends. Unique perspectives create more robust solutions.
4. Give it away now. Information and knowledge should be shared openly, freely.
5. Pay it forward. Teach the next generation to think like a hacker.

Based on the original definition of hacking, these are 5 principles that you can use to rethink situations, re-evaluate problems, and hack everything you do. So, if hacking is the application of hacking principles to everyday life, then I'll ask again, how many of you would be willing to call yourselves hackers after you leave here today?

This post is an [adapted version](#) of a talk given at the Ottawa Chapter of the Canadian Library Association. "Hacking is a mindset, not a skillset" by [Tanya Snook \(@spydergrll\)](#) is licensed under a [Creative Commons Attribution 3.0 Unported License](#).

*Note: This article gives the views of the author, and not the position of the Impact of Social Science blog, nor of the London School of Economics. Please review our [Comments Policy](#) if you have any concerns on posting a comment below.*

## About the Author

**Tanya Snook (@Spydergrll)** is an avid fan of technology and its impact on culture. By day, she's a business analyst, by night she writes for her blog, <http://Spydergrll.com>.

- Copyright © The Author (or The Authors) - Unless otherwise stated, this work is licensed under a Creative Commons Attribution Unported 3.0 License.