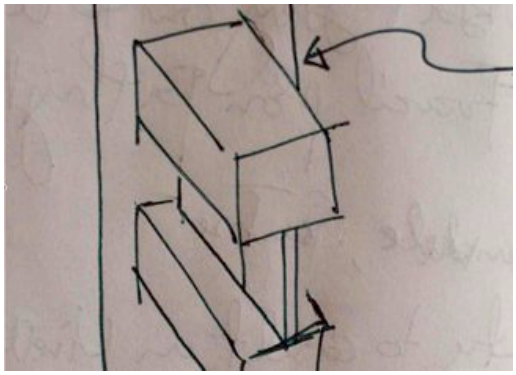
Sketching with Mike!

@ frogdesign



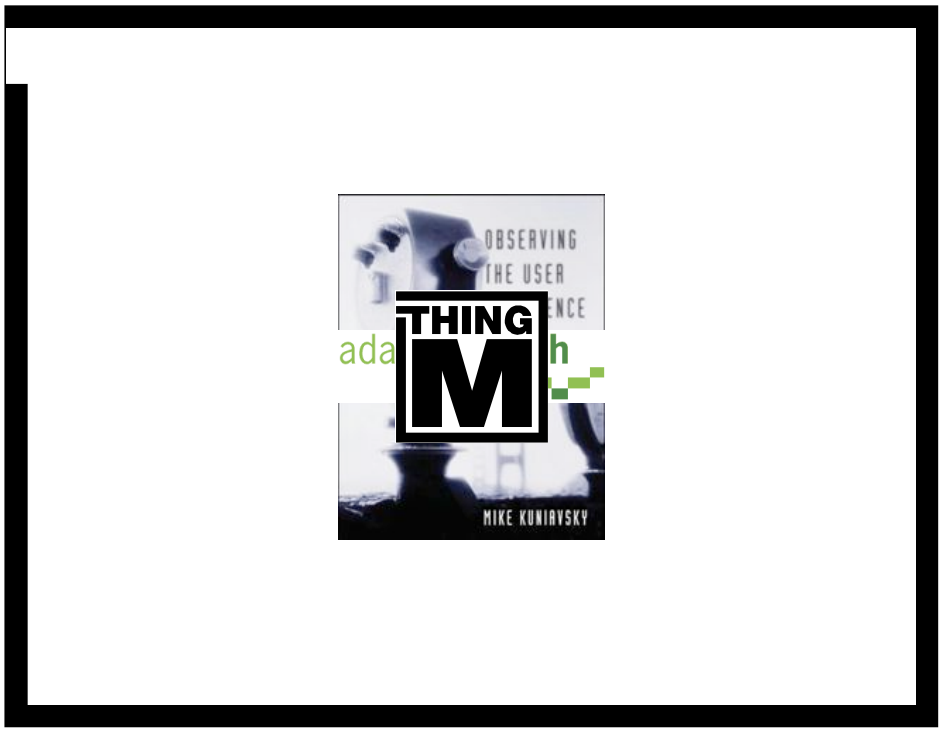
Mike Kuniavsky

July 10, 2007

I'd like to start by telling you a bit about who I am.

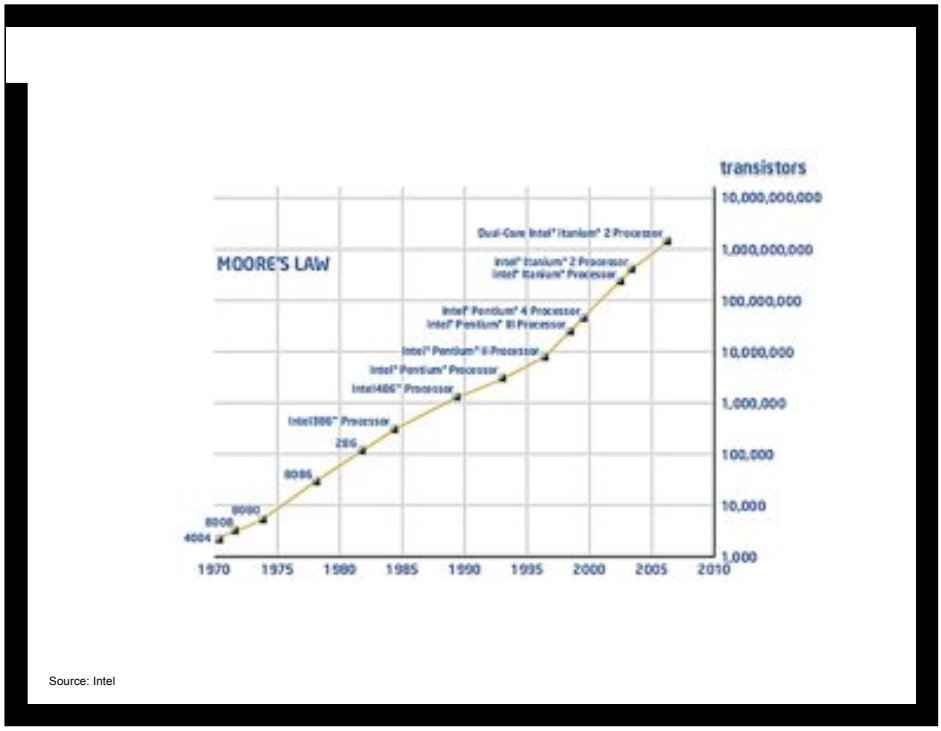
I am a consultant specializing in user experience design and user research. What that means is that I spend a lot of my time thinking about the relationship between technology and people. My core philosophy is that making technology is easy compared to figuring out what technology to make, and that's driven by understanding people.

Over the years, I've worked with many different organizations to help them develop technology with people in mind.



I wrote a book on that topic, focusing on user research techniques. I also co-founded a design company called Adaptive Path.

Last year I started a design and research company called ThingM. We call ThingM a device studio and we specialize in the relationship between information systems and objects. In other words, we're a ubiquitous computing design studio.



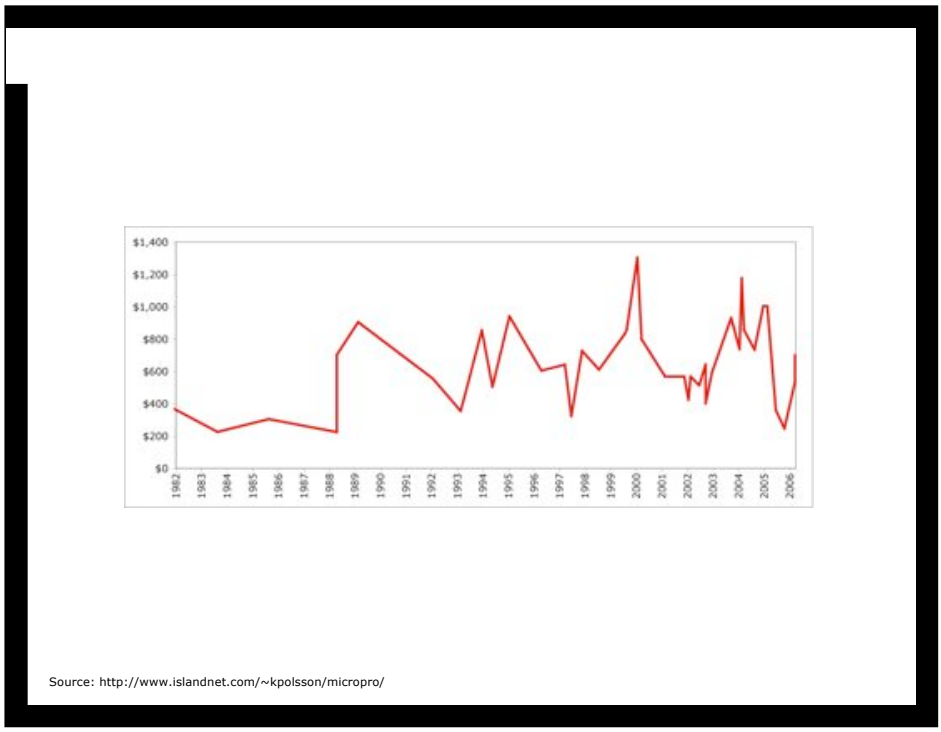
Source: Intel

Let me tell you about ubiquitous computing and why I think it's really important. And where that starts is in something that's close to every technologist's heart. Look, it's Moore's Law!

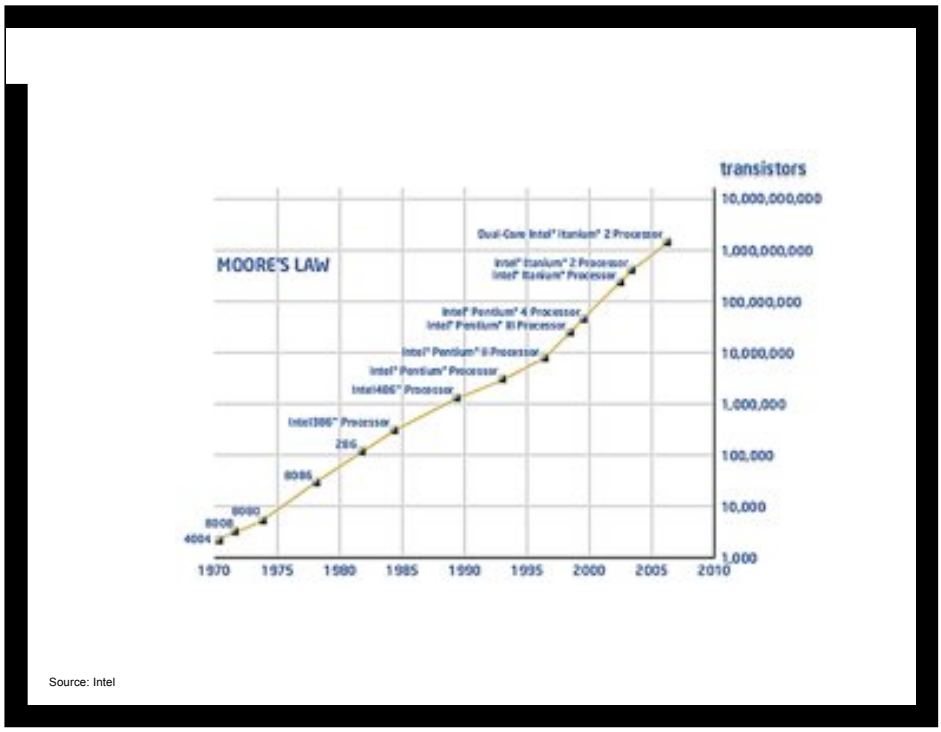
I know you've seen it a thousand times, but let's look at it again. People typically read this chart as a trend focusing on the number of transistors.

What's implicit in this trend, however, is that this is happening within the context of a marketplace.

This is not just the theoretically largest number of transistors that's possible to put on a on a CPU die. It's the number of transistors that can be sold at a specific price point.



However, the prices of CPUs on release have stayed the roughly same. This is a graph I made of the introductory price of many of the major chips at the time of their introduction. Even with the fluctuations in the price because of market positioning and the competition between Intel, Motorola and AMD, the price has remained pretty steady, generally between $500 and $1000 at the time of introduction.



Source: Intel

Taken in light of processor prices, the other way that you can read this chart is that the price of older processor technology decreases proportionately to the increase in transistor density. And although people tend to concentrate on the right side of the curve, I'd like to draw your attention a little to the left, to what I call the Hidden Middle of Moore's Law.

This range of processor power can do an immense amount and the price of it has dropped to near-disposable commodity levels. I think it starts right around the 486.

1989: $900

Quantities of 1000

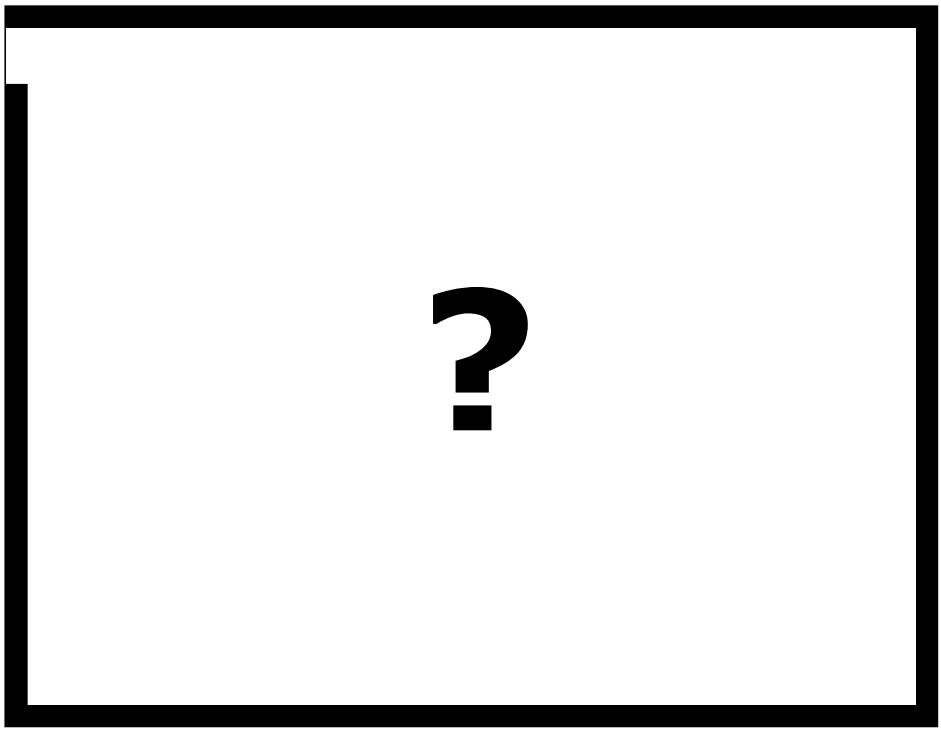
33MHz, 20MIPS

2007: $0.53

Quantities of 1

20MHz, 20MIPS

Sure enough, you can see that the actual chip prices reflect that price drop.Yeah, an Atmel CPU isn't the same as an i486, but it certainly shows that the trend is roughly correct.



What does this mean? This means is that embedded information processing becomes a cost-effective competitive advantage, much as new kinds of materials are.

And, like any fundamentally new material, when added to the design of an object, information processing and wireless networking fundamentally changes the capabilities of the object, It's akin to deciding to make something out of rubber rather than plastic. Or steel versus bamboo.

Say you want to sell more toy monkeys in an already crowded toy monkey market. Interactive behaviors, such as speech, memory, maybe a little servo control under a silicone skin may just be the differentiator that sells more of your monkeys. Doing that purely mechanically or with basic electronics is prohibitively expensive, but now that CPU power is cheap enough, you can in essence throw information processing at an otherwise difficult physical problem and it becomes a competitive calculation. You can put it on a spreadsheet like you would choose between latex and silicone or different kinds of fur.

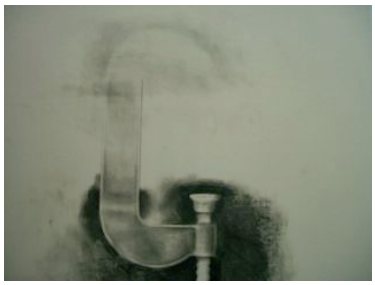
[Thanks to Rafi Haladjian for the example.]

This means that the vision of ubiquitous computing the late Mark Weiser had nearly 20 years ago at Xerox PARC is now a practical reality. The competitive advantages of systems with embedded information processing make ubiquitous computing an emergent byproduct of the decrease in chip prices.

I founded ThingM with Tod Kurt because I believe that ubiquitous computing holds amazing promise for making the world a better, happier and more interesting place. It is today where the Web was in 1992.

Source: Flickr, jurvetson

However, as you well know, between now and the glorious ubicomp future there's a lot of interaction design to be done. Designing user experiences for ubiquitous computing is largely terra incognita. We just started figuring out how to make Web pages not totally suck, and this is a whole new game. We don't have the tools.



(Jim Dine, *Untitled* from *Ten Winter Tools*, 1973)

I don’t know, but I decided to go back to the origin of all design tools to think about what qualities it could have. I went back to sketching. As Bill Buxton so correctly points out in his book, sketching is not prototyping. It is not the first step in solving a problem, it is the process by which we understand the design space so we can define the problem in the first place.

I see sketching on paper as having 3 key qualities.

1. Fast. The less time it takes to explore an idea, the more ideas can be explored. Time spent recreating a mental state where you had an idea after dealing with grungy details is often a big hurdle to making the right thing, versus just the most expedient thing.
2. Provisional. You know a sketch is not the final product. There are a bunch of indicators that say it's not the real thing to you and to others, That way everyone stays focused on the core ideas, and doesn’t get distracted by peripheral details.
3. Preserves history. Sketching shows you in one place the record of successful ideas, experiments and failures. You're constantly defining the envelope by being able to glance back to your dead ends and successes.

Bill Buxton has come up with many others.

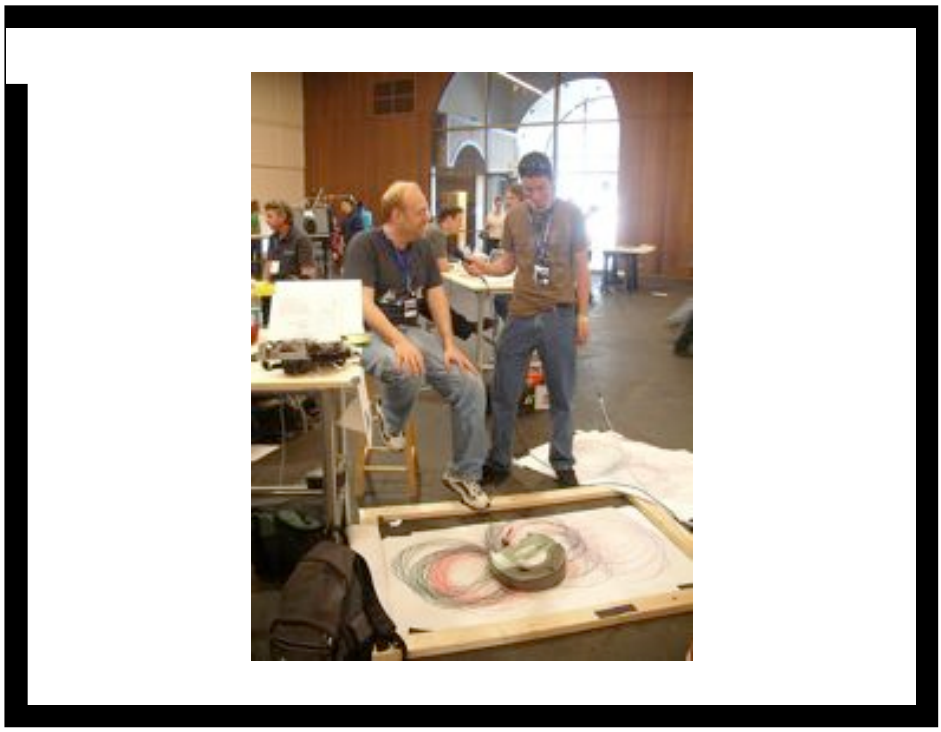
Medium	Speed	Provisional	History	Score
Drawing	5	5	5	125
Theater	5	5	3	75
Writing	5	4	3	60
Music	5	5	1	25
Software	2	3	4	24
Architecture	3	4	1	12
Interaction design	3	2	2	12
Information Architecture	4	2	1	8
Screen-level interfaces	2	3	1	6
Hardware	1	1	3	3

How does developing hardware compare with sketching and other activities? I did a completely unscientific scoring comparing media, and not surprising, hardware blows. The problem is not that making hardware isn't as easy as sketching, it's that it's *nowhere near as easy*. The problem is not moving it to the top, but moving up in that list, even a little.

As we're at the very beginning of the ubiquitous computing future, I have no idea what tools we're going to need to create good ubiquitous computing experiences.

When Tod and I cofounded ThingM I wanted the company to be based on sketching. I want to understand the space that we've found ourselves in by sketching in it a lot. We approach all design as part of a sketching process.

Let me show you some examples. This is an image drawn by a robot.

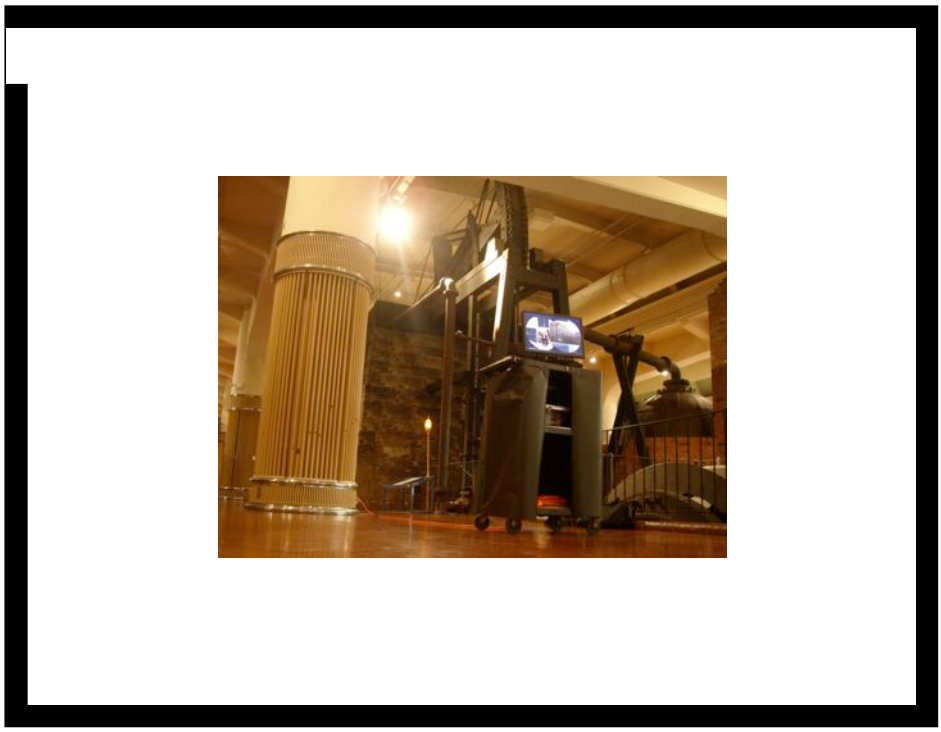


Specifically, a Roomba vacuum cleaner. That's Tod on the left at Maker Faire. Last summer he wrote a book about hacking Roombas to make them draw, play music, work as input devices, etc., all without damaging their core vacuuming function. Understanding the products of current tools is the first step to creating new ones.

Let me tell you about a client project we've been working on. We've been contracted by the Henry Ford to help them understand how to make their enormous collection more relevant. See, they're in this interesting position: they have a fantastic collection of millions artifacts related to the history of technology in America from the 17th century until today. If it was made of metal or wood between 1700 and today, they probaby have it. It was Walt Disney's model for Disneyland and it's enormous. It covers nearly a square mile. However, making those objects relevant is hard. A 65 year-old retired engineer is going to have a very different perspective on one of these artifacts than a 12 year-old on a class trip.

Our project for the museum is to arrange for a series of experiments exploring the idea of customizing content and then report back about what we had learned. It is explicitly not to create new technology, but to understand the boundaries that exist when trying to create a specific kind of experience with several new technologies. We started by doing an exhaustive literature search where we tried to gather everything we could about how others had create customized content experiences in museum using new technologies. This included everything from cell phone tours to extensive immersive experiences.

Then, once we knew what others had done, we started sketching. The way the project works is that once a month we show up in Dearborn, Michigan and spend a week creating a completely new experinece, with realistic content and hopefully semi-functional technology. This is April's. We used RFIDs and projected video to tell the story of several key artifacts in Henry Ford's life from several perspectives.

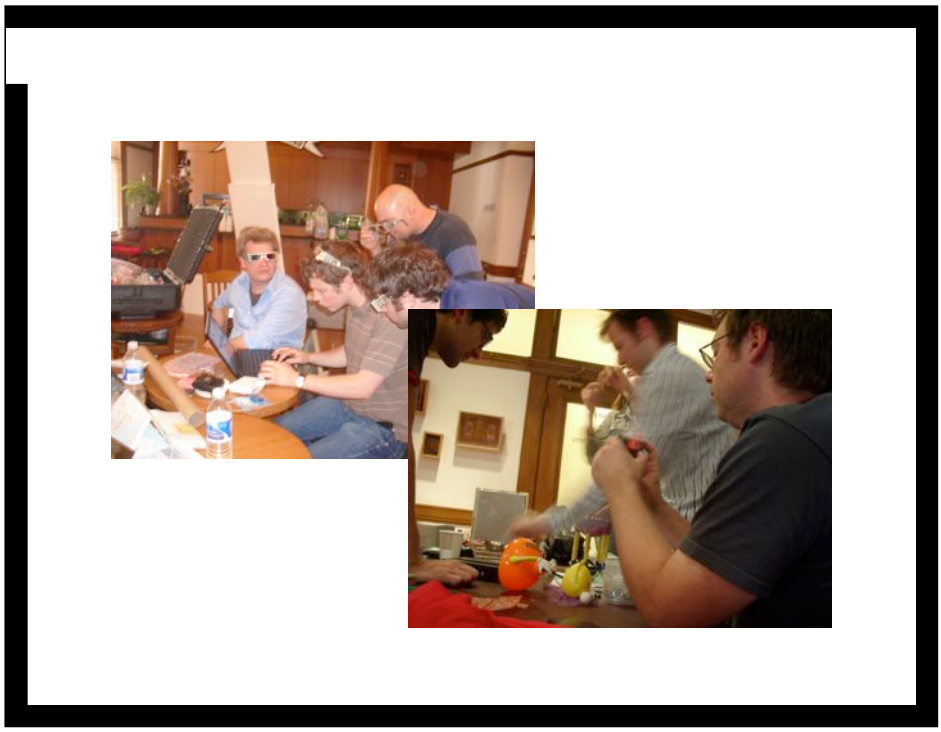


This is May's sketch. We tried to create a way to make these enormous steam engines relevant. Very few people even know how a steam engine works, and the relevance of the differences between the various engines in the Henry Ford collection are very difficult to explain to most people. We decided to recontextualize these machines by creating a kind of periscope that would allow you to see what they would have looked like in their original context or what they look like inside. We made it by putting an optical mouse under a lazy susan and then moving the image in the opposite direction of the mouse movement.

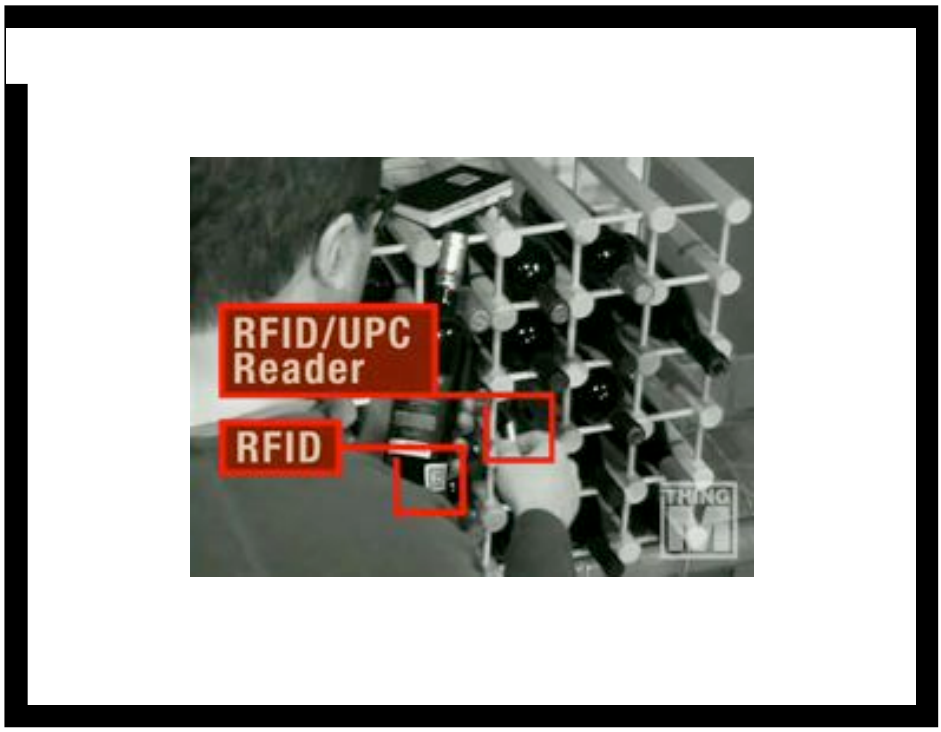
And here's the one we did the week before last. We used a pair of Wiimotes to create a kind of magic wand that allows people to point at certain objects and get extra information about them. Depending on the wand they use, they get different information.

We tested each of these prototypes with end users and although we don't have the final results from this project, I've written 60 pages of report about them. I'm going to be distilling that this week into what will hopefully be some guidelines for how to create systems such as this, or at least what to be aware of.

We also do another sketching-related practice. We run a conference called Sketching in Harware, where we bring together 30 people who are involved in the design of toolkits for rapid hardware prototyping to talk about issues in the design of tools for rapidly developing products. My feeling is that the design of tool is an incredibly powerful leverage point and that if we can discuss the creation of these tools at this early stage in the field, we're much more likely to have better tools in the long run.

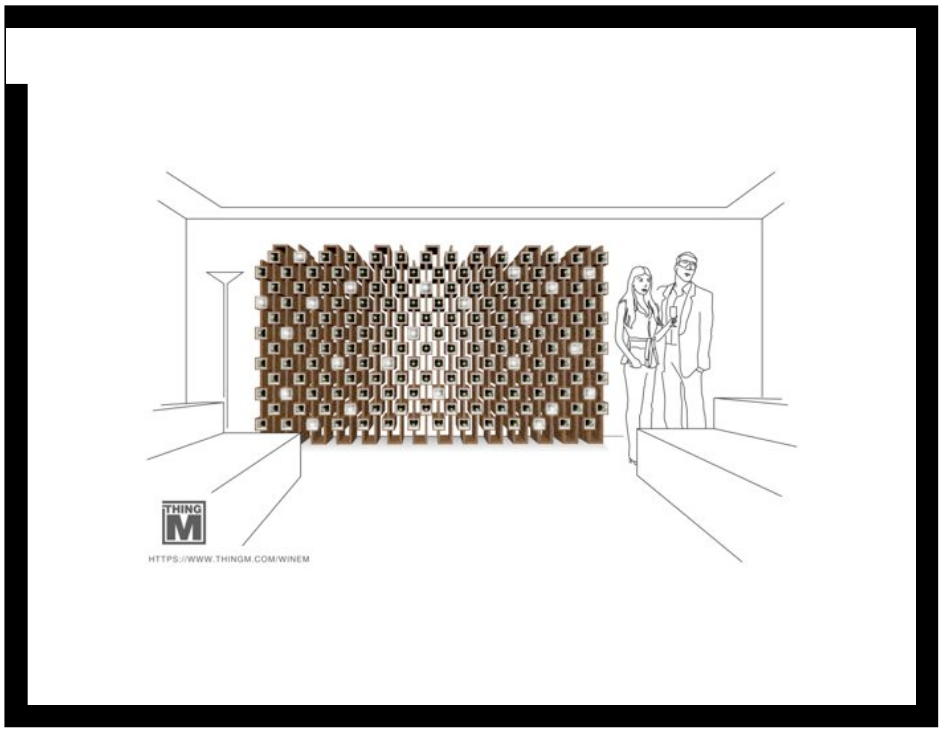


And of course there's actual sketching in the conference. We have a great time and some interesting projects come out of it. You'll be able to see many of the presentations from it online in the next week or so.

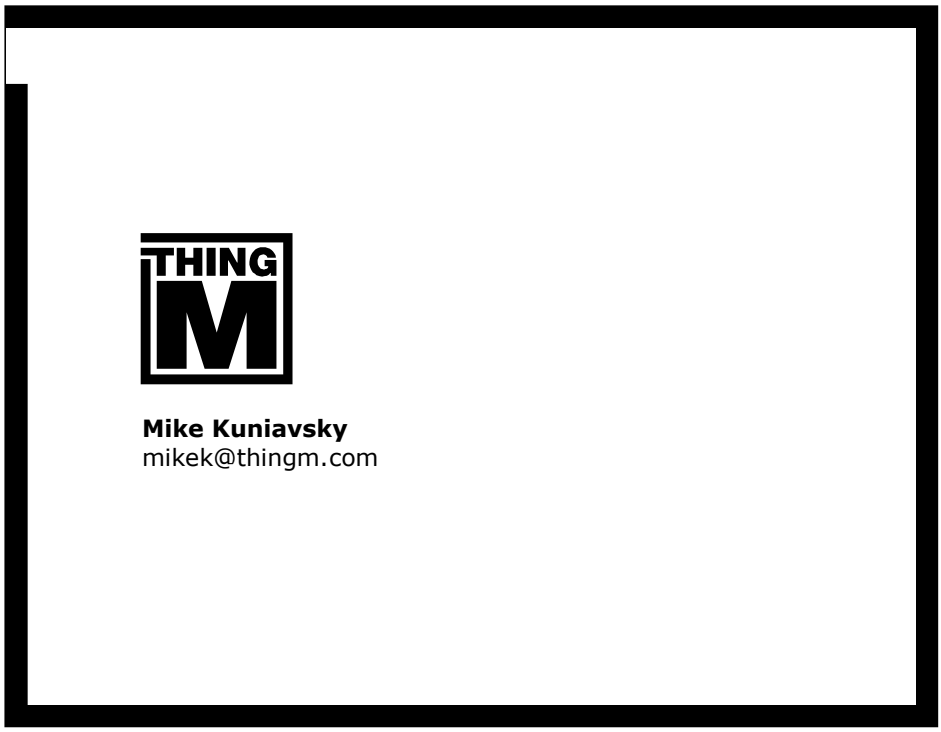


Finally, I'd like to talk about a project that we're in the middle of, which can be considered to be an extended high-fidelity sketching exercise. It started in January with what we call a Technology Sketch. This is a way for us to imagine how a technology could work. We try to do one of these per month to stretch ourselves. In a day, we come up with an idea, and then do a video that show how it could work. The production of the video is a key point of the sketching process. In the process of creating the video we have to face a lot of our assumptions about the experience we're trying to create. The January one was about RFIDs and furniture, and we chose wine racks.

We learned a lot about how interaction with RFID-enabled devices work. However, we weren't prepared for the response.



This is yesterday's rendering of the idea. It's number 38 out of 60 different designs and interaction models, but I think it's the one we're going to actually build in the next couple of weeks. We still consider this level of fidelity to be a kind of sketching. We're not prototyping. We don't feel that this is necessarily the right way to satify the needs and desires of our target audience. Maybe this particular technological solution can satisfy those needs and desires. Maybe it can't. But we're working quickly and in the process of developing it, we will discover a lot. If it's not the right solution, that's fine. One of the point of sketches is that they're disposable, even if they took a couple of weeks to contruct and weigh 500 pounds. In the end, we believe that profits come from solving THE RIGHT problem, not just a problem that we happened to have thought of, and we're willing to sketch until we know what that problem is.



Thanks!