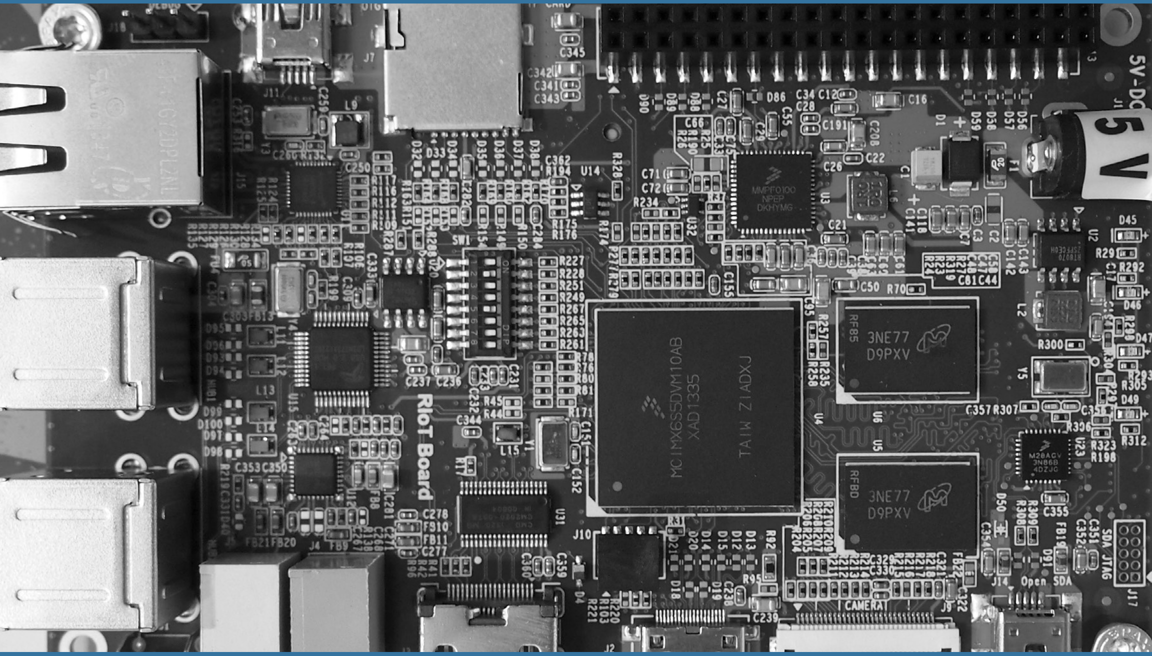
O'REILLY®

When Hardware Meets Software

How the Internet of Things Transforms Design and Manufacturing



Mike Barlow

When Hardware Meets Software

How the Internet of Things Transforms Design and Manufacturing

Mike Barlow

Beijing · Cambridge · Farnham · Köln · Sebastopol · Tokyo

O'REILLY®

When Hardware Meets Software

by Mike Barlow

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://my.safaribooksonline.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Editor: Mike Loukides

November 2014: First Edition

Revision History for the First Edition:

2014-11-12: First release

While the publisher and the author(s) have used good faith efforts to ensure that the information and instructions contained in this work are accurate, the publisher and the author(s) disclaim all responsibility for errors or omissions, including without limitation responsibility for damages resulting from the use of or reliance on this work. Use of the information and instructions contained in this work is at your own risk. If any code samples or other technology this work contains or describes is subject to open source licenses or the intellectual property rights of others, it is your responsibility to ensure that your use thereof complies with such licenses and/or rights.

While every precaution has been taken in the preparation of this book, the publisher and authors assume no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.

ISBN: 978-1-491-90256-1

[LSI]

Table of Contents

When Hardware Meets Software

In the Internet of Things, what's old is new, and what's new happens much faster.

When most of us first caught up with the Internet, it seemed like a magical door to a virtual reality played out in the infinite reaches of cyberspace. Today, it seems as though everyone is talking about the Internet in terms of its ability to get things done in the physical world of meat and machinery. We hear people talking about the Internet of Things, the Industrial Internet, and the Internet of Everything.

Is the Internet a wonderful rabbit hole into the infinite mind of the cosmos or the unsexy front end of a new industrial age? As Al Pacino puts it so eloquently in *The Godfather Part III*, "Just when I thought I was out, they pull me back in."

For those of us in the trenches of the Internet counterrevolution, the apparent shift in interest from virtual to physical benefits translates to a blurring of the boundaries between software and hardware. In a global economy driven by consumer technologies, software developers need to think more like product designers and product designers need to think more like software developers.

Until relatively recently, there was no bright-line demarcation between "hardware people" and "software people." Techies were techies, and everyone who was interested in electronics learned how to read a circuit diagram and use a soldering iron.

The rise of programmable computers spawned a new kind of techie—the software engineer. Each twist and turn of the modern industrial economy leads to increasing specialization. Technology was not exempt from this rule, and soon enough you had software developers who had never seen a finished product roll off an assembly line.

Welcome to the Age of Indie Hardware

All trends rise and fall. A new generation of smart techies has emerged to challenge the false duality of the hardware versus software paradigm. The spiritual heirs of the ham radio operators and homemade rocket enthusiasts of the 1940s and 50s have coalesced to form a maker culture that is quietly subverting the standard industrial model of product design and development.

Even if they aren't the actual grandsons and granddaughters of the original hobbyists, they apply the same kind of grit, smarts and do-it-yourself confidence as earlier generations of inventors and tinkerers who labored in basements, backyards, and garages all over the world.

Unlike their predecessors, whose audiences were limited mostly to friends and family members, the new generation is sharing its inventiveness globally and selling gadgets through maker-friendly ecommerce markets such as Tindie, Make:, and Grand St.

"We're increasingly seeing people who are able to support themselves —or who at least are close to supporting themselves—by selling technology they've created," says Julia Grace, head of engineering at Tindie. "Most of our sellers are people who started off as hobbyists, doing this on nights and weekends. As demand has grown, however, and as our sellers gain access to facilities and manufacturing techniques that were previously available only to people working in large companies, they're able to produce their items faster and sell more of them."

Julia compares indie hardware markets like Tindie to Etsy, the successful ecommerce website specializing in handmade artisanal products. On Etsy, you can shop for handcrafted ceramics and jewelry; on Tindie you can shop for Doomsday Clock Shield with Rotary Encoder or Tapster, a robot designed to play Angry Birds.

Indeed, Tapster is one of indie hardware's major success stories. Designed originally for gamers, it's been upgraded and adopted by large manufacturers as a low-cost machine for testing mobile apps.

Tapster was invented by Jason Huggins, a software developer who uses open source hardware technology such as Bitbeam, a modular building system similar to Lego Technic. What's especially interesting about

Bitbeam is that it can be made with a 3D printer—which means, of course, that it is potentially scalable. When something works and can be produced in large numbers easily, watch out.

For the moment, much of what's made by the maker community is consumed largely by other makers. But the winds of change are blowing. If a handful of visionary Fortune 500 firms are using Tapster, more will surely follow in their footsteps tomorrow.

Two or three years from now, Tapster (or something like it) might be seen in retrospect as the Altair 8800 of the indie manufacturing era. If that's the case, then the Commodore 64 and Apple I of the new era are already in the wings and waiting for their cue to take the stage.

It's also quite possible that the next cool thing produced by the indie manufacturing culture won't be a computer. Grand St., launched in late 2012, is also an online market for nifty products based on cutting-edge tech. In addition to providing a market for sellers, it serves as a platform for design and development.

"We offer three types of products," explains Amanda Peyton, Grand St.'s CEO and cofounder. "First, we have a traditional marketplace for products that are finished and ready to ship. Second, we have a 'beta' marketplace for products from makers who want feedback from people who aren't friends and family. Third, we have a pre-order marketplace for products that aren't ready for shipping, but will likely ship in the next six months. It's a way to test demand for a product before it hits the market. Sometimes there's an actual prototype and sometimes there's just a rendering."

Grand St.'s model offers potential consumers a view into the previously hidden universe of product design and development. In a very real sense, Grand St. pulls back the curtain, creating transparency in a process that is usually shrouded in secrecy.

The new approach isn't just about lifting the veil and showing consumers how it all works. It's based on shifting economics. "The capital requirements around product development have really changed," says Amanda. "Previously, you needed lots of money up front to fund an entire product development process. Now you can engage with your audience much earlier, and you can forecast the demand for products before they've hit the market."[1]

In the world of consumer electronics, for example, the impact of this new approach means that smaller teams—and fewer resources—are required for launching new products. When you only need a couple of designers and developers working on a project, time-to-market cycles can be compressed dramatically. More important, you don't need to raise gazillions in fresh capital to set up an assembly line.

Mindset and Culture

Julia Grace grew up playing first-generation Nintendo games. Her first computer was a used Commodore 64 and she learned programming from books she found at the public library. In today's world of ubiquitous broadband connectivity and instant streaming, that makes her seem very "old school." But her background and mindset are typical of the maker movement, which is essentially a 21st century variation on the pioneer cultures that stressed self-sufficiency and self-reliance in the 19th century.

It's fair to say that not everyone shares the retro-mindset and passionate enthusiasm for handcrafted small-batch technologies. A quick tour of any college dormitory reveals a startling lack of interest in any kind of technology that can't be assembled intuitively, immediately and without any tools. When was the last time you saw a college student with a soldering iron, wire strippers, or a circuit tester?

The phenomenon doesn't merely apply to geeks and nerds. In the 1950s, you couldn't drive two blocks without seeing someone working on a souped-up car with a customized chrome-plated exhaust system, tuck-and-roll leather upholstery, and an engine hood modified to accommodate a big nasty turbo charger. If you want to see the modern-day equivalent of 20th century gear-heads, you must travel to Shenz-

[1] In April 2014, after Amanda Peyton was interviewed for this report, Grand St. announced that it would be acquired by Etsy. "While this is a big step for us as a company, we plan to change very little about the site and your experience of Grand St. in the near term," Amanda Peyton wrote on her blog. "We exist to bring you the best in indie electronics from designers and makers all over the world, and we'll continue to do that at grandst.com. We will continue to launch new features and have a few coming out soon that we think you'll love."

hen, a large city in China's Guangdong Province, just north of Hong Kong.

Tigers Pacing in a Cage

Andrew "bunnie" Huang has a Ph.D. in electrical engineering from MIT, but he is most famous for reverse-engineering the Xbox, establishing his reputation as one of the world's greatest hardware hackers. He sees an evolving relationship between hardware and software.

"It used to be that products were limited solely by the capability of their hardware. Early radios, for example, had mechanical buttons that acted directly on the physics of the receiver," says bunnie. "As hardware becomes more capable, the user experience of the hardware is more dictated by the software that runs on it. Now that hardware is ridiculously capable—you basically have supercomputers in your pockets that cost next to nothing—pretty much the entire user experience of the product is dictated by the software. The hardware simply serves as an elusive constraint on the user experience."

Hardware is "a cage," says bunnie, and good software developers learn to work within the constraints of the hardware. "When I work with programmers on new products, I take the first prototype, put it on the desk and I say, 'Welcome to your new cage.' That's the reality. There's a hard wall. But we try to build the cage big enough so there are options for programmers. A quad core Android phone with a gigabyte of memory is a pretty big cage. Sometimes when programmers feel constrained, they're just being lazy. There's always more than one way to skin a cat in the software world."

For instance, if battery life is an issue—and when isn't it?—a programmer has a multitude of tools to optimize for power consumption: she can reduce CPU clock speed or shut down subsystems when the device is idling; she can optimize popular libraries and routines to take less energy to compute; or she can preload or precompute data images so energy consumption is shifted to the cloud.

Even the highest-level UI decisions impact battery life; an animated home screen wallpaper will burn batteries faster than a static background. On the other hand, no phone will ever contain a battery with the energy density required to brew a full pot of coffee.

While it seems unlikely that consumers will demand phones that also brew coffee, software designers routinely face challenging constraints, such as writing code for the embedded processors in low-end consumer products. These systems contain smaller CPUs and as little as a few dozen kilobytes of memory. “That’s when you see programmers pacing around like tigers in a cage,” says bunnie.

From bunnie’s perspective, the introduction and rapid proliferation of multitouch input digitizers on user interfaces (think of any smart phone screen) has had the greatest impact on the relationship between hardware and software. “Before that, and without that, the experience of that product would have been not nearly as interesting. The pinch zoom, the whole scrolling action that we are now very familiar with was very revolutionary. They redesigned the whole user experience around that input sensor,” says bunnie.

Certainly, one of the most compelling features of the iPhone was its touch screen. Most of the same tasks could be performed with buttons or a stylus, but the user experience was so completely different that the input digitizer became an indisputable competitive advantage. At the same moment that it became something everyone wanted, it also became the new cage.

But that’s the nature of product design. Yesterday it was novel, today it’s standard, and tomorrow it feels old-fashioned. Designers and developers can curse at the whims of temperamental consumers, but they cannot ignore them.

“I’ve run into developers who make inaccurate assumptions about hardware and who start complaining loudly about the design of the hardware when they bump into problems,” says bunnie. “They want you to loosen the constraints. But it always boils down to cost. You can do anything, but it will cost you money. Then the company will tell you they can’t sell the product at a higher price point.”

Occasionally, says bunnie, he meets developers who are “truly proactive in thinking through the whole process. But it’s rare to find people with the skills required to develop products that are great from front to back.”

Understanding the interplay between software, hardware, and the existing supply chain is important, he says. And that’s why many companies are heading to Shenzhen—because it’s one of the few places in the world where there is a critical mass of people with the multiple skill sets required for meeting the endless demands of a growing global consumer economy that constantly hungers for exciting new products.

The stalls, booths, shops, and mini-factories of Shenzhen are filled with younger and even scrappier versions of bunnie, and they are all looking for the big break that will elevate them into the spotlight. You might not find someone to refit your '55 Chevy Bel Air with an air intake, but if you're looking for someone to customize your smart phone, you've come to the right place.

Hardware Wars

Joichi Ito is the director of the MIT Media Lab.[2] Joi (pronounced "Joey") recalls sending a group of MIT students to Shenzhen so they could see for themselves how manufacturing is evolving. "Once they got their heads around the processes in a deep way, they understood the huge differences between prototyping and manufacturing. Design for prototyping and design for manufacturing are fundamentally different," says Joi. The problem in today's world, according to Joi, is that "we have abstracted industrial design to the point where we think that we can just throw designs over a wall" and somehow they will magically reappear as finished products.

The trip to Shenzhen helped the students understand the manufacturing process from start to finish. "In Shenzhen, they have a $12 phone. It's amazing. It has no screws holding it together. It's clearly designed to be as cheap as possible. It's also clearly designed by someone who really understands manufacturing *and* understands what consumers want."

Joi also sees a significant difference between what's happening on the factory floors in Shenzhen and the maker movement. "We're not talking about low-volume, DIY manufacturing," he says. Instead, Joi's students are working through the problems and challenges of a real, live paradigm shift—the kind of gut-wrenching upheaval described in Thomas S. Kuhn's seminal book, *The Structure of Scientific Revolutions*.

From Kuhn's point of view, a paradigm shift isn't a cause for celebration or blithe headlines—it's a sharp and unexpected blow that topples old theories, wrecks careers, and sweeps aside entire fields of knowledge.

The existing paradigm—the status quo—favors companies whose products and services are based mainly on software, because software

[2] Joichi Ito is also co-chair of the upcoming O'Reilly Solid Conference.

can be scaled rapidly at minimal cost. The emerging paradigm favors companies whose products and services are based mainly on hardware, because the cost of developing and manufacturing hardware is dropping precipitously.

Software and Moore's Law go together like soup and sandwiches. Put less kindly, software made Moore's Law look like a real law. Hardware is a different kettle of fish, since it involves orchestrating and synchronizing dozens (and sometimes hundreds) of complex interlocking processes, such as product design, prototyping, sourcing raw materials, managing supply chains, setting up assembly lines, building or leasing factories, setting up distribution channels, packaging, shipping, and much more.

The innovators in places like Shenzhen are showing the world that hardware startups can look a lot like software startups. They don't necessarily need tons of seed money or venture capital, they can be spun up relatively quickly, and if they fail, they can be broken down and sold for spare parts. That's the paradigm shift—and the people who control large portions of the global economy and decide where to invest trillions of dollars, yen, yuan, won, or rupees are beginning to see hardware as the coolest new shiny object.

Does This Mean I Need to Buy a Lathe?

The Internet of Things doesn't presage a return to the world of smoke-belching factories and floors covered with sawdust. But it does signify that change is afoot for any business or activity related to the information technology or communications industries.

"Not everyone will become a hardware designer," says Joi. But many students, software engineers, and entrepreneurs will see the advantages of learning how to work with hardware. "It's never too late to learn this stuff," says Joi, "if you decide that you want to do it."

At minimum, software engineers should learn as much about design and manufacturing as possible. "Buy an Arduino[3] and start building. Everything you need to learn is on the Web," urges Jordan Husney, an

[3] According to its homepage, "Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, designers, hobbyists and anyone interested in creating interactive objects or environments."

avid hardware hacker who serves as strategy director at Undercurrent, an organizational transformation firm and digital think tank in lower Manhattan.

In the same way that software people will have to reconfigure their modes of thinking, hardware people will need to learn new technical skills and new ways of looking at problems, says Jordan. “They will have to become more comfortable with uncertainty occurring later and later in the process,” he says. “Hardware engineers will keep things in the realm of bits (as opposed to committing them to atoms) by sharing designs using digital collaboration and simulation tools virtually, while testing multiple physical prototypes. I think we’re going to see the supply chain start to shift around these concepts.”

The idea of rapid and continuous iteration (some prefer to call it “continuous improvement”) dovetails with the notion that designers will have to become accustomed to higher levels of uncertainty farther into the manufacturing process. “It used to be that you’d ship a product and it was considered finished. Now, folks are shipping beta products (with unfinished hardware and software) to gather more feedback from the field,” says Jordan. “We’re seeing this with Coin [a startup with a card-like device that stores information about the credit cards you would normally carry in your wallet] and Energy Aware, which has just begun shipping its beta whole-home energy monitors.”

Designers now see their products as platforms, says Jordan. “With remote connectivity and remote updates, they’re able to iterate and add value to products that their customers already own. My iPhone, for example, gets better every day. Apple continuously produces new operating system enhancements and adds new functionality. The app platform allows software makers to increase the value of the hardware in my pocket.”

Obstacles, Hurdles, and Brighter Street Lighting

More than one observer has noted that while it’s relatively easy for consumers to communicate directly with their smart devices, it’s still quite difficult for smart devices to communicate directly, or even indirectly, with each other. Bill Ruh, a vice president and corporate officer at GE, drives the company’s efforts to construct an Industrial

Internet that will enable devices large and small to chat freely among themselves, automatically and autonomously.

Bill is the opposite of a mad scientist. From his perspective, the Industrial Internet is a benign platform for helping the world become a quieter, calmer, and less dangerous place.

"In the past, hardware existed without software. You think about the founding of GE and the invention of the light bulb—you turned it on and you turned it off. Zero lines of code. Today we have street lighting systems with mesh networks and 20 million lines of code," says Bill. "Machines used to be completely mechanical. Today they are part digital. Software is part of the hardware. That opens up huge possibilities."

A hundred years ago, street lighting was an on or off affair. In the future, when a crime is committed at night, a police officer could raise the intensity of the nearby street lights by tapping a smart phone app. This would create near-daylight conditions around a crime scene, and hopefully make it harder for the perpetrators to escape unseen. "Our machines are becoming much more intelligent. With software embedded in them, they're becoming brilliant," says Bill.

But the Internet we've grown accustomed to isn't reliable, robust or anti-fragile enough for what Bill envisions, which is a world of smart devices that are constantly sharing data and making real-time decisions.

Networks of smart devices already exist in the aviation industry. The locomotives and wind turbines GE builds have smart components that report their status and warn of impending breakdowns. The rest of the world, however, gets by on an Internet designed mostly for sharing noncritical information.

For example, if your Internet connection goes down while you're sending an email to a friend, posting a status update on Facebook, or watching someone's cat riding a Roomba vacuum on YouTube, it's not really a big deal. Almost all of the technology we associate with our PCs and mobile devices is really consumer technology, which means that it's reasonably okay but not the absolute best it can be. And for 99.9 percent of what we do in our daily lives, that's fine. Even the dreaded "blue screen of death" wasn't really fatal.

All of that changes when the machines we depend on for matters of life and death are added to the mix. When jet engines, municipal water systems, and heart-lung machines join the conversation, we need an

Internet (and all of the information and communication technologies supporting it) that performs flawlessly, every second of every day. Today, when your LinkedIn page freezes, it's irritating, but not dangerous. Tomorrow, when your Google driverless car freezes (let's not use the word "crash"), it's a different story.

As a culture, we're going to have to figure out how to make hardware that works all the time. "People get testy when their cell phone drops a call, but they get *really* testy when their electricity stops working," says Bill. When actual lives are at stake, concepts such as reliability, scalability, and security become incredibly relevant. They become more than just buzzwords used by marketers and industry journalists.

"We have a visceral response when the machines we depend on stop working. We expect the software in our machines to be as good as the hardware. Meeting those expectations requires a different approach to software design, a different mentality," says Bill.

The "old" mentality, says Bill, meant designing products for "rocket scientists" and assuming that they would eventually, through trial and error, figure out how to use it. As a result, most products had features that people never used. Microsoft Word is the classic example of the "old" software design mentality, and it's probably fair to say that most flat screen televisions are shipped with more features than will ever be used by the typical consumer. Earlier in this paper, Joichi Ito spoke of throwing a design over a wall and expecting someone on the other side to turn it into a finished product. Bill offers a similar thought, suggesting that many software designers neglect to consider the consequences of their designs on the end user. He predicts that more companies will require designers to come up with "zero installation time" products and services that work right out of the box, requiring no setup or fiddling to deliver expected levels of performance.

"People today assume they have a certain set of inalienable rights, such as the right to see water flowing when they turn on the tap and the right to see the lights go on when they touch a switch. People feel very strongly about those rights and they will get upset when they feel that their basic needs aren't met," he says.

Is the Industrial Internet—the Internet of Things, the Internet of Everything—prepared to deliver that kind of service? Are products designed for the new age of smart devices capable of delivering on expectations such as continuous improvement, 24/7 reliability, and zero installation?

In a new book, *The Zero Marginal Cost Society: The Internet of Things, The Collaborative Commons, and The Eclipse of Capitalism*, author Jeremy Rifkin notes that in 2007, "there were 10 million sensors connecting every type of human contrivance to the Internet of Things. In 2013, that number was set to exceed 3.5 billion, and even more impressive, by 2030 it is projected that 100 trillion sensors will connect to the IoT."

It seems highly unlikely that the Internet—which was neither conceived nor designed to function as an industrial-strength communications system—will ever fulfill the role of a universal medium for connecting everything to everything else. What seems more likely, however, is that machines will become smarter and more autonomous. We will design machines that we can trust to do the right thing, even when we're not keeping an eye on them.

The concept of trustable machines returns us to our earlier conversation about the evolving relationships between software and hardware, and between design and manufacturing. Ideas such as "zero installation time," "zero user interface," and "invisible buttons" require ditching the old paradigms of software versus hardware and design versus manufacturing. As we become increasingly dependent on our gadgets and devices, our primary concerns will narrow down to safety, reliability, redundancy, and survivability. In a world of brilliant machines and 100 trillion networked sensors, the main question will be, "Does this thing actually work?"

About the Author

Mike Barlow is an award-winning journalist, author, and communications strategy consultant. Since launching his own firm, Cumulus Partners, he has represented major organizations in numerous industries.

Mike is coauthor of *The Executive's Guide to Enterprise Social Media Strategy* (Wiley, 2011) and *Partnering with the CIO: The Future of IT Sales Seen Through the Eyes of Key Decision Makers* (Wiley, 2007).

He is also the writer of many articles, reports, and white papers on marketing strategy, marketing automation, customer intelligence, business performance management, collaborative social networking, cloud computing, and big data analytics.

Over the course of a long career, Mike was a reporter and editor at several respected suburban daily newspapers, including *The Journal News* and the *Stamford Advocate*. His feature stories and columns appeared regularly in *The Los Angeles Times*, *Chicago Tribune*, *Miami Herald*, *Newsday*, and other major US dailies.

Mike is a graduate of Hamilton College. He is a licensed private pilot, an avid reader, and an enthusiastic ice hockey fan. Mike lives in Fairfield, Connecticut, with his wife and two children.