O'REILLY®

Industrial Internet

The Machines are Talking



Jon Bruner

Sponsored by

www.dbooks.org



Stay up-to-date with the O'Reilly Solid Newsletter

Learn about the latest in the rapidly emerging area of the connected world, Internet of Things, and disruptive innovation.

See a sample:

oreilly.com/go/**solidnews**

O'REILLY®

Solid

SOFTWARE/HARDWARE EVERYWHERE

©2014 O'Reilly Media, Inc. The O'Reilly logo is a registered trademark of O'Reilly Media, Inc. 14651

Industrial Internet

Jon Bruner

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo O'REILLY®

www.dbooks.org

Industrial Internet

by Jon Bruner

Copyright © 2013 O'Reilly Media. All rights reserved.

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*.

March 2013: First Edition

Revision History for the First Edition:

2013-03-27: First release

See *http://oreilly.com/catalog/errata.csp?isbn=9781449368258* for release details.

Nutshell Handbook, the Nutshell Handbook logo, and the O'Reilly logo are registered trademarks of O'Reilly Media, Inc.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and O'Reilly Media, Inc. was aware of a trademark claim, the designations have been printed in caps or initial caps.

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: 978-1-449-36825-8

Table of Contents

Acknowledgments v

1. The industrial internet 1
Characteristics 2
Internet architecture and practice applied to industrial settings 2
Software abstraction makes the physical world accessible 3
Optimization above the level of a single machine 4
Substitution of software for assets 5
Substitution of software for labor 5
Everything becomes a sensor 7
Machines built nightly 7
Ultra-transparent markets replace regulation 8
Security problems arise from systems that were built without connectivity in mind 9

2. Security 11

3. Industry Focus 15
Energy 15
Building controls and demand response 17
Utilities 19
Generation 21
Automotive 23
Transportation 26
Aviation 27
Railroads 32
Health care 34

www.dbooks.org

Manufacturing 35

4. The role of Silicon Valley in creating the industrial internet. 39
Silicon Valley and industry adapting to each other 40

5. Conclusion. 43

Acknowledgments

I am grateful to more than 50 experts from the industries discussed in this report for speaking with me, hosting me on illuminating visits, and introducing me to their helpful contacts. Many of them gave me permission to share their insights here but asked that their names and affiliations not be published.

General Electric has generously underwritten this report, and it put its industry experts at my disposal while I wrote it. Nevertheless, the thematic development of the industrial internet in this report is strictly my own, formed from discussions with my colleagues at O'Reilly and outside experts.

We've been developing our framing of the industrial internet since the fall of 2012, and we will continue to cover this space with blog posts, interviews, and videos at *http://oreil.ly/industrial-internet*.

— Jon Bruner

www.dbooks.org

CHAPTER 1
The industrial internet

The barriers between software and the physical world are falling. It's becoming easier to connect big machines to networks, to harvest data from them, and to control them remotely. The same changes in software and networks that brought about decades of Silicon Valley innovation are now reordering the machines around us.

Since the 1970s, the principles of abstraction and modularity have made it possible for practically anyone to learn how to develop software. That radical accessibility, along with pervasive networks and cheap computing power, has made it easy to create software solutions to information problems. Innovators have responded, and have reshaped practically any task that involves gathering information, analyzing it, and communicating the result.

Something similar is coming to the interfaces between software and the big machines that power the world around us. With a network connection and an open interface that masks its underlying complexity, a machine becomes a Web service, ready to be coupled to software intelligence that can ingest broad context and optimize entire systems of machines.

The industrial internet is this union of software and big machines — what you might think of as the enterprise Internet of Things, operating under the demanding requirements of systems that have lives and expensive equipment at stake. It promises to bring the key characteristics of the Web — modularity, abstraction, software above the level of a single device — to demanding physical settings, letting innovators break down big problems, solve them in small pieces, and then stitch together their solutions.

www.dbooks.org

The foundational technologies of the industrial internet are available now to anyone from big industrial firms to garage inventors. These technologies include: pervasive networks; open-source microcontrollers; software that can analyze massive amounts of data, understand human preferences, and optimize across many variables; and the computing power needed to run this intelligence, available anywhere at little cost.

Anyone who can recast physical-world problems into software terms now has access to the broad world of "stuff that matters": conserving energy and reducing our impact on the environment; making our world safer, faster, and more comfortable; improving the productivity and well-being of workers; and generating economic opportunity.

Characteristics

The industrial internet[1] is an approach to bringing software and machines together, not a particular group of technologies. These are the principles driving its development.

Internet architecture and practice applied to industrial settings

The industrial internet isn't necessarily about connecting big machines to the public Internet; rather, it refers to machines becoming nodes on pervasive networks that use open protocols. Internet-like behavior follows: machines publish data to authorized recipients and receive operational commands from authorized senders.

Think of the difference between an airplane built 40 years ago and a modern design like the Boeing 787. Older airplanes have direct linkages between systems — from the landing-gear switch to the landing gear, for instance. Newer airplanes use standard networks, in which the landing gear is a node that's accessible to any other authorized part of the system — not only the landing-gear switch, but also safety, autopilot, and data-logging systems. Software can understand the status of the airplane in its entirety and optimize it in real-time (and, with a data connection to dispatchers and the air-traffic control system, soft-

1. We use lowercase *internet* to refer generically to a group of interconnected networks, and uppercase *Internet* to refer to the public Internet, which includes the World Wide Web.

ware can also understand the airplane's relationship to other planes and to the airspace around it).

The infrastructure of the Internet is highly flexible and scalable. Once a system of machines is brought together on a network, it's easy to add new types of software intelligence to the system, and to encompass more machines as the scope of optimization expands.

Software abstraction makes the physical world accessible

Web services mask their underlying complexity through software interfaces. Need to convert an address to latitude and longitude? Google's geocoder API[2] will make the conversion almost instantaneously, masking the complexity of the underlying process (text parsing, looking up possible matches in a database, choosing the best one). Geolocation thus becomes accessible to anyone building a Web site — no expertise in cartography needed. These services become modules in Web applications, which are designed with minimal assumptions about the services they use so that a change or failure in one module won't break the entire application.

In the same way, the industrial internet presents machines as services, accessible to any authorized application that's on the network. The scope of knowledge needed to contribute to a physical-world solution becomes smaller in the process.

Making a furnace more efficient, for instance, might involve some combination of refining its mechanical and thermal elements (machine design) and making it run in better relation to the building it's in and the occupants of that building (controls). The industrial internet makes it possible to approach these challenges separately: connect the furnace to a network and give it an API that guards against damaging commands, and the control problem becomes accessible to someone who knows something about software-driven optimization, but not much about furnaces.

In other words, the industrial internet makes the physical world accessible to anyone who can recast its problems in terms that software can handle: learning, analysis, system-wide optimization.

2. *https://developers.google.com/maps/documentation/geocoding/*

www.dbooks.org

At the same time, this transfer of control to software can free machines to operate in the most efficient ways possible. Giving a furnace an advanced control system doesn't obviate the need for improvements to the furnace's mechanical design; a machine that anticipates being controlled effectively can itself be designed more efficiently.

Optimization above the level of a single machine

With machines connected in Internet-like ways, intelligence can live anywhere between an individual machine's controller and the universal network level, where data from thousands of machines converges. In a wind turbine, for instance, a local microcontroller adjusts each blade on every revolution. Networked together, a hundred turbines can be controlled by software that understands the context of each machine, adjusting every turbine individually to minimize its impact on nearby turbines.

Optimization becomes more effective as the size of the system being optimized grows, and the industrial internet can create systems that are limitless in scope. Upgrades to the American air-traffic control system, for example, will tie every airplane together into a single system that can be optimized at a nationwide level, anticipating a flight's arrival over a congested city long before it approaches. (The current system is essentially a patchwork of space controlled at the local and regional level.)

Software intelligence, which relies on collecting lots of data to build models, will become smarter and more granular as the scope of data collection increases. We see this already in the availability of traffic congestion data gathered by networked navigation systems and smartphone apps. The next step might be cloud-level software that gathers, analyzes, and re-broadcasts other machine data from networked cars — the state of headlights and windshield wipers to detect rain, for instance.

Optimization can go beyond a single kind of machine to take into account external market conditions. "Each silo has achieved its highest possible level of efficiency," says Alok Batra, the CTO and chief architect for GE Global Research.[3] "If we don't break down silos, we can't generate more efficiency. Nothing operates in isolation anymore. If

3. Note: GE has sponsored this paper. See the acknowledgements section.

you operate a manufacturing plant, you need to know about wind and power supplies."

Substitution of software for assets

The industrial internet will, as Astro Teller[4], Captain of Moonshots at Google[x], suggests, "trade away physical complexity for control-system problems." As machines deliver their work more efficiently, we'll need fewer of them and the machines themselves will become simpler.

Consider, for instance, that California's state-wide electricity demand stays below 30 gigawatts about 80% of the time. For about 20 hours every year, though, it surges past 47 gigawatts.[5] Utilities must build out massive capacity that's only used during peak hours a few days each summer.

Flattening out those peaks could dramatically reduce the capacity needed to reliably serve the state's electricity needs, and that's a control-system problem. An interconnected stack of software that extends all the way from power plants to light bulbs — parts of which are sometimes called the "smart grid" — could gather system-wide context as well as local preferences to gently control demand, dimming lights during peak hours and letting temperatures drift slightly in buildings whose owners accept a financial incentive in return for flexibility.

Substitution of software for labor

Given a high-volume stream of accurate machine data, software can learn very fast. And, by transmitting what it learns back into a network, it can accumulate knowledge from a broad range of experiences. While a senior pilot might have 10,000 to 20,000 hours of flying experience, a pilotless aircraft operating system might log hundreds of thousands of hours in just a year, with each of many planes transmitting anomalies back to a universal learning algorithm.

4. *http://astroteller.net/*

5. *http://www.epa.gov/cleanenergy/documents/suca/ee_and_dr.pdf*

www.dbooks.org

U.S. manufacturing productivity grew by 69% in real terms between 1977 and 2011[6], in part because machines automated many low-level human tasks. In health care, similar gains have been elusive: productivity grew by just 26% in real terms over the same period as spending nearly quadrupled (and productivity — economic output divided by number of employees — is itself an imperfect measure for what we want from our health care system).

The kind of automation that has revolutionized manufacturing has so far failed to revolutionize health care. Doctors and nurses spend much of their time reading machine data from sensors (everything from blood-pressure cuffs to MRI machines), matching patterns of symptoms to likely diagnoses, and prescribing medication within formal guidelines. As routinized as that work is, it still requires a great deal of human judgment and discretion that automation tools have so far not been able to provide.

The industrial internet will make the health care sector more efficient by providing intelligence on top of machine data. Software will ingest sensor readings and perform real-time analysis, freeing doctors and nurses to do work that requires more sophisticated and nuanced patient interaction. Progress is already well underway in home monitoring, which lets patients who just a few years ago would have needed constant monitoring in a hospital bed recover at home instead.

As automation did to factory workers, the industrial internet will undoubtedly obviate the need for certain types of jobs. If information is seamlessly captured from machines as well as people, we'll need fewer low-level data shepherds like medical transcriptionists (ironically, the demand for these types of jobs has increased with the introduction of electronic medical records, though that's largely due to the persistence of poor user interfaces and interoperability barriers). The industrial internet will automate certain repetitive jobs that have so far resisted automation because they require some degree of human judgment and spatial understanding — driving a truck, perhaps, or recognizing a marred paint job on an assembly line.

6. See *http://www.bea.gov/industry/xls/GDPbyInd_VA_NAICS_1998-2011.xls* and *http://www.bea.gov/industry/xls/GDPbyInd_VA_NAICS_1947-1997.xls* for output by industry and employment from 1998 to 2011; see *http://www.bea.gov/industry/xls/GDPbyInd_FTPT_1948-1997.xls* for employment before 1998. The health care statistics here refer to combined ambulatory, hospital, and residential care, NAICS codes 621, 622, and 623.

In fast-growing fields like health care, displaced workers might be absorbed into other low- or medium-skill roles, but in others, the economic tradeoffs will be similar to those in factory automation: higher productivity, lower prices for consumers, continued feasibility of manufacturing in high-cost countries like the United States — but also fewer jobs for people without high-demand technical skills.

Everything becomes a sensor

Any machine that registers state data can become a valuable sensor when it's connected to a network, regardless of whether it's built for the express purpose of logging data. A car's windshield-wiper switch, for example, can be a valuable human-actuated rain sensor if it's connected to the vehicle's internal network.

Software operating across several machines can draw from aggregate data conclusions that can't be drawn from local data. One car running its windshield wipers doesn't necessarily indicate rain, but a dozen cars running their windshield wipers in close proximity strongly suggests that it's raining.

Software operating across several types of machine data can also draw out useful systemic insights. Combined with steering-wheel, speed, GPS, and accelerator-pedal readings, a sensor-driven rain indication could warn a driver that he's moving too fast for road conditions, or help him improve his fuel economy by moderating his acceleration habits.

Machines built nightly

The Web brought about the end of the annual software release cycle.[7] Provided as a loosely-coupled service on the Internet, software can be improved and updated constantly. The industrial internet will bring about a similar change in the physical world.

Some of the value of any machine is in its controls. By replacing controls regularly, or running them remotely and upgrading them every night like a Web service, machines can be constantly improved without any mechanical modifications. The industrial internet means that machines will no longer be constrained by the quality of their on-board intelligence. Development timelines for certain types of machines will

7. See *http://oreilly.com/pub/a/web2/archive/what-is-web-20.html?page=4*.

www.dbooks.org

become shorter as software development and hardware development can be separated to some degree.

Automakers, for instance, build cars with mechanical services that are designed to last more than 10 years in regular use. Entertainment and navigation systems are outdated within two years, though, and the software running on those systems might be obsolete in a few months. Automakers are experimenting with ways to decouple these systems from the cars they're installed in, perhaps by running entertainment and navigation software on the driver's phone. This scheme effectively gives the car's processor an upgrade every couple of years when the driver buys a new phone, and it gives the car new software every time the driver upgrades his apps.

It's easy to imagine something similar coming to the mechanical aspects of cars. A software update might include a better algorithm for setting fuel-air mixtures that would improve fuel economy. Initiatives like OpenXC[8], a Ford program that gives Android developers access to drivetrain data, portend the coming of "plug and play intelligence," in which a driver not only stocks his car with music and maps through his phone, but also provides his own software and computational power for the car's drivetrain, updated as often as his phone. One driver might run software that adjusts the car's driving characteristics for better fuel economy, another for sportier performance. That sort of customization might bring about a wide consumer market in machine controls.

This could lead to the separation of markets in machines and in controls: buy a car from General Motors and buy the intelligent software to optimize it from Google. Manufacturers and software developers will need to think in terms of broad platforms to maximize the value of both their offerings.

Ultra-transparent markets replace regulation

The electricity market balances supply and demand on sub-second intervals, but data constraints prevent it from being truly transparent. As a result, efforts to reduce electricity demand (and its consequent impact on the environment) have typically been regulatory — mandating the phase-out of incandescent lightbulbs, for instance. Two el-

8. *http://openxcplatform.com*

ements are lacking: data-transmission infrastructure, which would send instantaneous price data from power producers to distributors, local utilities, and, ultimately, consumers; and some sort of intelligent decision making, which would take into account both instantaneous electricity prices and human preferences to decide, for instance, whether to run a dishwasher now or in 10 minutes.

The industrial internet promises to provide both data transmission and intelligent decision making, and in doing so it will create highly transparent, efficient, and comfortable markets down to the individual household level.

Security problems arise from systems that were built without connectivity in mind

Security vulnerabilities in the industrial internet often arise from the assumption that some system is isolated. Contraband connectivity invariably makes its way into any system, though. The best way to approach security is to assume connectivity and plan for it, not to avoid it entirely. Counterintuitively, Internet Protocol and other open, widespread internet technologies, by virtue of their having been under attack for decades, can be more secure than specialized, proprietary technologies.

Security issues are discussed in detail in the next section.

www.dbooks.org

CHAPTER 2
Security

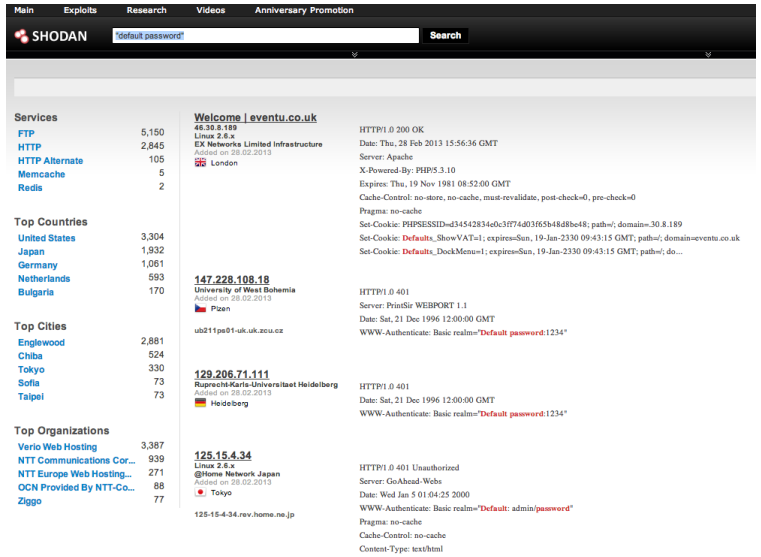
Adding software on top of machines and connecting them to networks creates tempting targets for malicious hackers. The evolution of industrial internet security is much like the evolution of PC security: many systems that are now being networked have historically enjoyed security by isolation, and, just as the original generation of PC operating systems didn't anticipate connections to the Internet, many industrial systems were not built with outside contact in mind.

The inherent scalability of software means that a single exploit can propagate fast; once discovered, an exploit can be used against lots of machines. Think of a car's odometer: the move to digital mileage counts, stored in software, makes it more difficult to tamper with the readout, but it expands the prospective target of an exploit from just one car (for mechanical odometers) to every car that uses the same software.

Tools like Shodan[1], a search engine for the Internet of Things, and Digital Bond's Basecamp[2], a database of industrial control exploits, illustrate the scale of the industrial internet and its vulnerabilities.

1. *http://www.shodanhq.com/*
2. *http://www.digitalbond.com/tools/basecamp/*

www.dbooks.org



Shodan is a search engine for Internet-connected devices, including some industrial control systems and Internet switches. Here it reveals several computers that return a default password field in their HTTP responses.

Industrial-control security is a fast-growing discipline with many parallels to the early PC security industry, but also some crucial advantages: connected infrastructure generally operates within tightly-defined networks, with consistent transmission and control patterns. As tools to handle big data improve, it becomes easier to apply deep-packet inspection and anomaly detection to the industrial internet, where these are particularly effective techniques.

Education will be a crucial part of the effort to keep the industrial internet safe. Industrial security conferences abound with stories of bored employees working the night shift at a power plant who circumvent their own security measures to play online games. Even highly technical employees are susceptible to spear phishing, in which an attacker sends a very specific email message with malware cloaked as a plausible attachment or Web link.

Air gaps — complete isolation of sensitive networks from the Internet — have long been part of industrial security, but they are becoming increasingly unworkable as the value of machine data becomes apparent to managers and as contraband connectivity finds its way in. Sys-

tems that rely on air gaps to avoid attacks will be compromised as connections are inevitably made across them.

"I don't think it's really possible to run a plant without bringing outside information in," says Eric Byers, the CTO of industrial-security firm Tofino, who has given a conference presentation entitled "Unicorns and Air Gaps — Do They Really Exist?"[3] Adds Byers: "As for management analytics, that horse ain't going back in the barn. Even on the plant floor, workers want iPads for both documentation and inventory — *what's in our spare closet?* — and that can make the difference between starting up in 15 minutes and starting up in two minutes."

The value of connectivity is high enough, and the stakes perilous enough, that the antivirus firm Kaspersky Lab sees a need for an industrial operating system that is "constructed with security in mind," says Roel Shouwenberg, who is part of the team developing Kaspersky's industrial operating system[4]. He figures that true air gaps at industrial facilities impose a productivity hit of 20-30%, and security approaches that are analogous to physical plant security systems (if you're standing in front of a machine on the factory floor, you're authorized to be there) are misguided. A better approach, he says, is to "trust no one, trust nothing" — that is, scrutinize what goes on rather than walling off parts of the system. "You can argue that the cause of nearly every security vulnerability that we can see is that some code is assumed to be trusted," he says.

Openness may be a key counterintuitive solution. The most widespread operating systems and protocols have been the subject of so many attacks and so much counter-research that they've become more secure than some smaller technologies that have never been challenged. "I can break your proprietary system in two days," says J.P. Vasseur, a fellow at Cisco Systems who has studied networks of big machines. "Because [Internet Protocol] has been under attack for decades, there's nothing more secure than IP."

3. *http://web.tofinosecurity.com/download-the-presentation-unicorns-and-air-gaps---do-they-really-exist/*

4. *http://radar.oreilly.com/2013/03/industrial-internet-security-kaspersky.html*

www.dbooks.org

CHAPTER 3

Industry Focus

Following is a handful of studies drawn from industries that will be particularly affected by the rise of the industrial internet. The accessibility of these examples varies; building the smart grid, with dynamic electricity prices calculated instantaneously as electricity supply and demand shift, will take years of stack development, entailing careful collaboration between power plant operators, distributors, independent system operators, and local utilities, and drawing in the seasoned engineering bases of all those participants.

Even so, some elements of the smart grid stack have been standardized and are now open to innovators from any background. Modularity means that an innovator doesn't need access to the *mechanism* of pricing in order to to build a responsive electric-car charger; she just needs to anticipate that dynamic pricing will eventually emerge as a service to which her machine can connect.

Energy

Ten years ago, San Diego Gas & Electric (SDG&E) had 20 generators on its network — big power plants that produced reliable electrical output at the command of human operators. Now, says Michael Niggli, the company's president and COO, it's got 2,000 generating sources, and could have 60,000 in another decade. Those include every solar installation and wind turbine connected to SDG&E's grid — power sources whose output fluctuates rapidly from minute to minute, flipping thousands of homes and businesses from net electricity producers to net electricity consumers every time the sun goes behind a cloud.

www.dbooks.org

At the same time, the adoption of plug-in electric vehicles promises to complicate electricity demand. The Nissan Leaf can draw as much as 3,300 watts, and a fully-charged battery holds 24 kilowatt-hours of electricity[1] (the average American household uses 31 kilowatt-hours in a day[2]). In one configuration, the Tesla S electric sedan can draw 20,000 watts and its battery can store 85 kilowatt-hours of electricity.[3]

Handling these new demands will involve both physical changes to the grid (especially the construction of small fast-dispatch power plants that can start in as little as 30 minutes when the wind stops blowing) as well as better controls based on software intelligence — the so-called "smart grid." Any improvement in these controls can substitute directly for new physical capacity.

The smart grid will require pervasive network connections to everything from coal turbines to clothes dryers, and an interoperable software stack to go with it. The core function of the smart grid will be dynamic electricity pricing that reflects supply and demand on a minute-by-minute basis. Fully dynamic pricing hasn't arrived yet, but peak-use surcharges are common in some markets today, and the sorts of responsive, intelligent controls that will work with it are useful even in the absence of dynamic pricing. Treated modularly, we can build and implement parts of the stack long before dynamic pricing arrives.

A layer of software intelligence on top of the world's electrical equipment could have a big impact on energy efficiency and the environment. Smart machines will not only use less electricity overall, they'll also use electricity in cleaner ways — drawing power when it's being produced in abundance by efficient sources and cutting back when rising demand forces utilities to switch on their dirtier generators.

Eventually, networks and abstracted systems might deliver services, optimized across entire systems, rather than assets and raw ingredients. Sunil Cherian, founder of Spirae[4], which produces electricity-distribution software, observes: "I'm not interested in electricity. I'm

1. *http://www.nissanusa.com/electric-cars/leaf/charging-range/battery/*
2. Averaged across the year. See table CE2.1, *http://www.eia.gov/consumption/residential/data/2009/index.cfm?view=consumption#fuel-consumption*
3. *http://www.teslamotors.com/models/facts*
4. *http://www.spirae.com/*

interested in illumination. I'm interested in comfort in the room. The delivery of those services, and who delivers those services, is really the fundamental problem. How do you take a copper infrastructure and transform that into a system that delivers the services or the end results that you're actually interested in?"

Building controls and demand response

Buildings — heating them, cooling them, lighting them, filling them with entertainment — make up 74% of U.S. electricity demand[5] and 56% of natural gas usage[6]. Much of that energy is used to heat, cool, light and entertain rooms much more than their occupants need.

The industrial internet will connect to building controls to moderate the relationship between people and the buildings they inhabit, balancing the sometimes-conflicting goals of reducing energy usage and keeping occupants comfortable. Software that sits on top of building controls will build thermal models and learn about occupants' preferences, then gently manage buildings.

At an abstract level, coordinating buildings with their occupants is a familiar problem. "Lots of appliances and other building systems are more or less asynchronous with occupants," says Mary Ann Piette, director of the Demand Response Research Center at Lawrence Berkeley National Laboratory. Home air conditioners run while we're at work, offices are cooled to uncomfortably low temperatures, lights stay on when we leave the room. Bringing these under better control will make it possible to form buildings to their occupants. "We have to go from components to systems," she adds. "That is an IT issue."

Efforts to moderate energy consumption through building controls have sometimes backfired because they're uncomfortable or inconvenient. Utilities have tried to introduce electricity discounts in return for the ability to remotely turn off a customer's air conditioner when demand is very high — which, in practice, means the air conditioner shuts down on the hottest days of the year when it's needed most.

The future of building controls, though, is much more moderate: controls will be informed and enabled by big data techniques that can assemble preference profiles and deliver energy savings at minimal

5. *http://buildingsdatabook.eren.doe.gov/TableView.aspx?table=1.1.9*
6. *http://buildingsdatabook.eren.doe.gov/TableView.aspx?table=1.1.10*

www.dbooks.org

inconvenience. Building controls will rely on broad, system-wide context and extremely granular control of building systems. "We're working toward having every light fixture be independently controllable," says Piette.

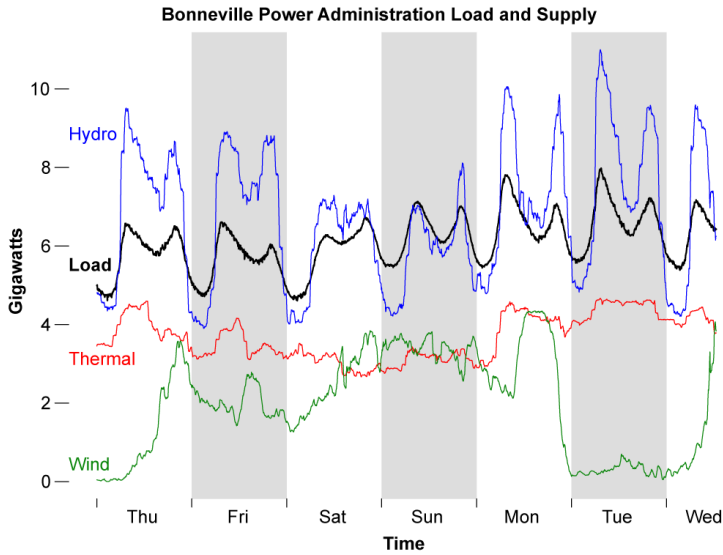
Instead of shutting off an air conditioner entirely on a hot day, a building's thermostat might consult the next day's forecast, run predicted conditions through a thermal model of the building, predict hourly changes in electricity prices, and then create an operating plan for the day that minimizes electricity costs while keeping the building within its occupants' range of tolerance.

The result might be that the building would run its air conditioners during the morning when electricity prices are low, then let building temperatures drift up imperceptibly during the mid-afternoon as prices rise. If the weather forecast calls for overcast skies and falling temperatures during the afternoon, the building could avoid the air-conditioning run-up during the morning.

Such systems will depend on local intelligence to run these models; internal networks to gather sensor data and control heating, air conditioning, and lights; and external networks to supply pricing and other real-time data from the grid.

At Lawrence Berkeley, Piette and her colleagues are developing ways to build the models that will sit at the heart of these systems, gathering sensor and machine data — interior and exterior temperatures, light glare, furnace settings, flue positions, occupancy, and a host of others — and determining the impact of each of these on interior comfort. These are the kinds of models that have become familiar to much of the software industry in the era of big data and predictive analytics, and software thinkers will have much to contribute to the intelligence that underlies smart buildings.

Utilities



Both electricity supply and demand (load) vary enormously from day to day and hour to hour. This graph shows a sample week of electricity supply and demand in the Bonneville Power Administration (BPA) dispatching area (supply is greater than demand for the most part because BPA exports some electricity to other areas). When wind power dies down, as it does on Monday afternoon in this graph, other generators must stand in. In this case, BPA was able to ramp up hydropower output quickly.[7]

The impact of this sort of control, coupled to dynamic electricity prices, could be massive. Electricity demand is highly irregular over the course of the day — it's not unusual for a household to use 40% of its daily electricity usage during its peak hour — and in most of the electrical system, every marginal watt of electricity is less efficient to produce than the previous watt.

Peak-hour output, which relies on expensive and dirty power plants that can be switched on quickly, is vastly more expensive to produce than the baseline power that comes from always-on sources like nuclear plants, and it consumes large amounts of capital investment than can't be widely amortized. (In California, for instance, state-wide elec-

7. *Data from http://transmission.bpa.gov/business/operations/wind/baltwg.aspx*

www.dbooks.org

tricity demand stays below 30,000 megawatts about 80% of the time. For about 20 hours every year, though, it exceeds 47,580 megawatts[8] — capacity that must be built and maintained for use only a few times every summer.) The object of demand response is to flatten the demand curve along the course of each day, week, and year, which in turn means less capacity will be necessary. It's an example of better controls standing in for machines.

Utilities and their customers will both benefit from better connections between buildings and the grid. Advanced meters already improve operations for utilities and help customers understand and reduce their electricity usage; they might eventually become the data interfaces between utilities and their customers.

Dennis Sumner, senior electrical engineer at Fort Collins (Colo.) Utilities, figures that his $36 million investment in advanced meters will pay off in 11 years from operational savings (meter readers no longer have to walk from door to door), but the data that they produce provides additional value to the utility. Reading electricity usage every 15 minutes — a 2,880-fold increase in resolution from the monthly data it was getting from human meter-readers — the utility can detect power outages and quality problems immediately, and have detailed data on scale and location.

In one case, Sumner says, meters in one neighborhood started to show voltage drops that suggested a transformer needed to be replaced. It was early spring and electricity demand was low; without smart meters, the problem would have manifested itself in the summertime when customers turned on their air conditioners. "Had we not done anything with it, we would have had a catastrophic failure," he says.

"Previously, we didn't know what was going on at the customer level," Sumner says. "Imagine trying to operate a highway system if all you have are monthly traffic readings for a few spots on the road. But that's what operating our power system was like."

The utility's customers benefit, too — an example of the industrial internet creating value for every entity to which it's connected. Fort Collins utility customers can see data on their electric usage through a Web portal that uses a statistical model to estimate how much electricity they're using on heating, cooling, lighting and appliances. The

8. *http://www.epa.gov/cleanenergy/documents/suca/ee_and_dr.pdf*

site then draws building data from county records to recommend changes to insulation and other improvements that might save energy. Water meters measure usage every hour — frequent enough that officials will soon be able to dispatch inspection crews to houses whose vacationing owners might not know about a burst pipe.

Green Button[9], a public-private initiative modeled on the Blue Button[10] program for health records, aims to give consumers more complete and more useful access to their own utility data by specifying what is essentially an API for utilities. Announcing the program in 2011, then U.S. Chief Technology Officer Aneesh Chopra wrote, "With this information at their fingertips, consumers would be enabled to make more informed decisions about their energy use and, when coupled with opportunities to take action, empowered to actively manage their energy use."[11] It's effectively an effort to reduce consumption not by edict, but by making markets more transparent and giving consumers the tools they need to react quickly to market conditions.

As promising as these initiatives are, the full "smart grid" as futurists imagine it will take years of careful collaboration between utilities, independent system operators, regulators, and software and hardware developers. Proposals for smart-grid standards abound, and big investments by any individual utility won't reach their full potential until every adjacent component is also modernized and connected.

"There are some dangerous conceptual ideas coming out of Internet companies saying the power system is like the Internet," says Dan Zimmerle, who runs a power-systems lab at Colorado State University[12] and directs research on grid technologies there. "The danger is in making policymakers think that utilities are as easily modernized as the Web."

Generation

Electric utilities and big power consumers around the world will spend more than $1.9 trillion on green-energy projects in the next five

9. *http://www.greenbuttondata.org/*
10. *http://bluebuttondata.org/*
11. *http://www.whitehouse.gov/blog/2011/09/15/modeling-green-energy-challenge-after-blue-button*
12. *http://www.eecl.colostate.edu/*

www.dbooks.org

years[13], and they are building more renewable capacity than ever before. Utilities and their customers installed 83.5 gigawatts of new renewable energy capacity worldwide in 2011[14], roughly equivalent to 12 Grand Coulee Dams[15], installing as much solar capacity in 2011 as existed in the entire world in 2009.

The industrial internet will make power plants more dynamic and easier to maintain. As renewable power sources and electric cars become more popular, stability in both supply and demand will be a crucial challenge. The same types of tools that will help smooth out demand will also help utilities produce stable power from wind and solar energy, matched to demand — namely, software intelligence connected directly to the machines that produce and consume electricity.

Wind farms are already loaded with sensors — weather sensors on each turbine as well as other machine sensors that monitor performance minute-to-minute. These sensors help power producers develop highly granular models that can forecast power production; with generators linked by pervasive networks, these forecasts will help utilities set dynamic electricity prices — data that will filter all the way back down to intelligent light bulbs.

This sensor data also helps power producers make the most of their assets. Software that interprets sensor data can alert crews that maintenance is needed and then help them schedule maintenance for times when impact on operations will be minimal. Newer wind turbines use software that acts in real-time to squeeze a little more current out of each revolution, pitching the blades slightly as they rotate to compensate for the fact that gravity shortens them as they approach the top of their spin and lengthens them as they reach the bottom.

Power producers use higher-level data analysis to inform longer-range capital strategies. The 150-foot-long blades on a wind turbine, for instance, chop at the air as they move through it, sending turbulence to the next row of turbines and reducing efficiency. By analyzing performance metrics from existing wind installations, planners can rec-

13. *http://www.pewenvironment.org/uploadedFiles/PEG/Newsroom/Press_Release/Innovate,%20Manufacture,%20Compete.pdf*

14. *http://www.pewtrusts.org/uploadedFiles/wwwpewtrustsorg/Reports/Clean_Energy/Clean%20Energy%20Race%20Report%202012.pdf*

15. *http://www.usbr.gov/projects/Powerplant.jsp?fac_Name=Grand+Coulee+Powerplant*

ommend new layouts that take into account common wind patterns and minimize interference.

Automotive

Google captured the public imagination when, in 2010, it announced that its autonomous cars had already driven 140,000 miles of winding California roads without incident. The idea of a car that drives itself was finally realized in a practical way by software that has strong links to the physical world around it: inbound, through computer vision software that takes in images and rangefinder data and builds an accurate model of the environment around the car; and outbound, through a full linkage to the car's controls. The entire system is encompassed in a machine-learning algorithm that observes the results of its actions to become a better driver, and that draws software updates and useful data from the Internet.

The autonomous car is a full expression of the industrial internet: software connects a machine to a network, links its components together, ingests context, and uses learned intelligence to control a complicated machine in real-time. Google hasn't announced any plans to make its cars available to the public, but elements of the industrial internet are widely visible in new cars today.

The car in the era of the industrial internet will be a platform — an environment that links software to the car's physical machinery, that understands conditions outside the car, and that serves as a safe interface to the driver. The platform will know something about the driver's preferences, control the car's internal environment, and feed the driver the information he needs when he needs it. Software will integrate and handle services previously handled by hardware components, which will speed development timelines, keep cars up-to-date once they're on the road, and facilitate customization (an important feature if, as some automakers expect, car-sharing will supplant ownership in many cases).

Entertainment and navigation systems are an obvious place to integrate multiple functions within one software environment; cars already make wide use of dynamic user interfaces and softkeys to control software-defined features. The Mercedes-Benz DriveStyle app with DriveKit Plus takes this integration a step further: it lets the driver's iPhone handle Internet connectivity and processing. The entirety of the navigation system, as well as Internet radio and social-media apps,

www.dbooks.org

runs on the iPhone, but the app displays on the dashboard screen and takes input from a knob near the gearshift.

As a component of the car, this iPhone-based system is asynchronous and modular. It shortens development timeframes for the automaker by decoupling the development of the entertainment system from that of the rest of the car. It also increases refresh frequency: a consumer who replaces her car every eight years might replace her smartphone annually, upgrading the entertainment system's processor with it.

"We've always had this challenge that once we put the head-unit in the car, it'll be there for years," says Kal Mos, senior engineering director at Mercedes-Benz R&D. "It's already been in development for a while before it goes in the car, then it lasts six to seven more years on the road. Why not take advantage of updates on the phone?"

By integrating its software with hardware, the car is also able to draw on more context to make better decisions — everything from its current location (and, say, nearby traffic conditions and businesses) to drive-train data that might help a driver save fuel or, in the case of Mercedes' high-end AMG sports car, improve her track-driving skills. "We're trying to provide the data you need, when you need it. The car almost becomes like a friend," says Mos.

The car's contextual awareness also enables safety features in the layer between the human and the software. Mercedes provides templates to developers that suggest changes to the way an app works when the car is moving and when it's parked; apps are forced into a simplified, low-clutter mode when the car goes into gear. Mercedes calls its design philosophy "guided openness."

Ford Motor is throwing open its doors to outside developers with OpenXC[16], an open-source hardware and software interface to its cars' drivetrain data. Car systems are already linked internally by the CAN bus, a near-universal vehicle control protocol, but Ford's effort opens that system to Android developers in a read-only capacity.

It's the start of what you might call "plug and play intelligence" — you carry around not only a preference profile and personal data on your phone, but also your own software stack and processor. Suppose OpenXC eventually gains write access to the drivetrain. Want to save gas? Run an app on your phone that coaches you to be a greener driver

16. *http://openxcplatform.com/*

and intermediates and optimizes the operation of your car — adjusting your automatic transmission's shift points, perhaps, or blunting your lead-food tendencies by easing acceleration.

But to get to that point, cars must become standardized software platforms. "In a physical sense, the notion of the platform has been with us as long as we've had mass production," says K. Venkatesh Prasad, senior technical leader for open innovation at Ford. The next step is to tie together a car's systems and make them available to developers.

Prasad suggests an illustration: for every car with a rain sensor today, there are more than 10 that don't have one. Instead of an optical sensor that turns on windshield wipers when it sees water, imagine the human in the car as a sensor — probably somewhat more discerning than the optical sensor in knowing what wiper setting is appropriate. A car could broadcast its wiper setting, along with its location, to the cloud. "Now you've got what you might call a *rain API* — two machines talking, mediated by a human being," says Prasad. It could alert other cars to the presence of rain, perhaps switching on headlights automatically or changing the assumptions that nearby cars make about road traction.

The human in this case becomes part of an API *in situ* — the software, integrated with hardware, is able to detect a strong signal from a human without relying on extractive tools like natural-language processing that are often used to divine human preferences. Connected to networks through easy procedural mechanisms like If This Then That (IFTTT)[17], human operators even at the consumer level can identify significant signals and make their machines react to them.

"I'm a car guy, so I'm talking about cars, but imagine the number of machines out there that are being turned on and off. In each case, the fact that a human is turning it on and off tells you something very interesting; it's human-annotated data," says Prasad.

"We want to allow the best of what's in the firm to innovate, and create a stable platform for the outside world to interact with us," says Prasad. "The key thing is going open with toolkits. All things open are closed at some level, and all things closed will — if they're really interesting — be opened at some point." As he speaks, a prototype box for an OpenXC USB hub sits on the coffee table in his office, with "open-source hardware" and "open-source interface" logos stamped on it.

17. *https://ifttt.com*

www.dbooks.org

Referring to *The Cathedral and the Bazaar* [18], Eric Raymond's seminal essay on open-source software, Prasad adds, "This is our offering to the bazaar."

Among the difficulties in creating truly integrated automotive networks are the long period it takes to refresh the national fleet (it takes about 15 years to refresh 95% of American cars), and the informal means by which they're maintained and upgraded — in contrast to industrial applications. "The mechanic down the street will need new skills," says Prasad. "Maybe this is a matter for Code for America[19]."

Prasad doesn't think that the addition of third-party intelligence will commoditize Ford's cars; instead, it lets Ford focus on building excellent machines. "You have to have excellent hardware and excellent software," he says. "The software needs a good operating system, the operating system needs good hardware, and the good hardware needs to be connected to an excellent car."

"There are lots of lessons to be learned for designs of internetworked platforms that attract others to add layers," says Prasad. "My collaborator Peter Semmelhack has often reminded me that Steve Jobs didn't think up Angry Birds."

Transportation

Transportation companies were early to recognize the value of computers for handling orders and coordinating complex systems. The airline industry started using computerized reservation systems in the 1950s, and the descendants of those early programs live on in the Sabre and Amadeus distribution systems. Airlines later recognized that logistical software could improve their capital utilization rates, structuring timetables and routes to make the best use of expensive equipment.

The transportation industry is now embracing the industrial internet with full, automated linkages between intelligent software and the big machines that move people and cargo. Trains, trucks, and airplanes gather detailed operational data and send it to system-level software

18. *http://shop.oreilly.com/product/9780596001087.do*

19. *http://codeforamerica.org/*

that optimizes routes, anticipates maintenance needs, and tweaks operations in real-time to improve fuel efficiency.

Eventually, the industrial internet will support broad use of automation to replace human operators with software that is safer, more reliable, and more efficient, completing the tie between global networks and machines. A single software stack will extend from the network planning and demand management level all the way down to throttles and brakes.

Aviation



The flight deck of a Boeing 787 Dreamliner, which uses Internet-like architectures to control flight systems and harvest data.[20]

Commercial airlines make up a complex network of intelligent devices, with authority distributed between ground-based dispatchers and air-traffic controllers and pilots in the sky, who exchange data mostly via ultra-low-bandwidth voice radios. Fuel is the biggest cost for every airline, and, spread across a large fleet, tiny refinements in flight paths, climbs, and descents can have an enormous impact on fuel consumption. Labor and capital are also big costs to anyone that operates an

20. *Photo by Alex Beltyukov, 2011, licensed under Creative Commons Attribution-ShareAlike 3.0 Unported. Via http://commons.wikimedia.org/wiki/File:Boeing_787-8_N787BA_cockpit.jpg*

www.dbooks.org

airplane; high capital utilization and effective maintenance management are crucial.

The industrial internet is coming to aviation in the form of high-bandwidth connections within airplanes, between airplanes, and from airplanes to ground controllers. These connections aren't being built as a unified system; rather, they're independently-developed networks that might eventually fit together as modules. They will enable more efficient flight plans, optimized maintenance regimes, and better utilization of airplanes and labor, and, if public opinion can be satisfied, they might eventually enable widespread use of pilotless aircraft in the domestic airspace.

The top-most of these networks is the Federal Aviation Administration's Next Generation air-traffic control system (NextGen). Slated to roll out over the next decade, it will replace voice-based communication between pilots and air-traffic controllers with digital streams. For the first time, air-traffic controllers will have a complete picture not only of the location of every airplane in the United States, but also of its flight plan, updated in real-time. Airplanes will also communicate with each other via ADS-B transponders, opening the door to the use of pilotless aircraft domestically, where they're currently prohibited.

The FAA's current air-traffic control system routes flights from ground beacon to ground beacon, giving pilots a series of targets for position and altitude as they cross the country on indirect paths. Planes descend toward airports in controller-mandated stair-step patterns, dropping a few thousand feet and then revving their engines to level off before dropping again, burning fuel and generating noise.

"Our flight-management computer knows the most efficient path and exactly when to start coasting toward an airport," says Gary Beck, the vice president for operations at Alaska Airlines. "ATC [air-traffic control] doesn't have that information, so their directions interfere with our optimized flight plans."

Beck says his airline saved $19 million in 2011 by using a satellite-based navigation program on its Alaska routes called Required Navigation Performance. Now it's taking part in a pilot program in Seattle that prescribes optimal-profile descents, which let an airplane essentially coast as much of the way as possible from cruising altitude to runway.

Using these descents will save the airline 2.1 million gallons of fuel every year.[21]

NextGen also promises to accommodate the collection and distribution of weather data. "All of our airplanes are basically weather sensors," Beck told a conference audience last fall. "At the minimum they give off things like altitude, winds aloft, temperature, and G-forces, which can then be translated into turbulence, but the problem is that data doesn't go anywhere, so ATC is not aware of what's going on in a real-time manner."

With airliners on a common, high-bandwidth network, this real-time weather data creates value for every airline and takes on one of the classic features of big data: information becomes more valuable when it's compiled broadly and shared back.

Within aircraft, too, better connections and sensors promise to cut operating costs. Every time a Boeing 787 pulls up to an airport gate, it disgorges a stream of data alongside its passengers. A wireless transceiver on the jet bridge connects to the plane's computer and downloads detailed flight data gathered by its engines, avionics, navigation system and other sensors distributed throughout the aircraft. The 787's systems can compile upwards of 100 gigabytes of sensor data per hour of flight; its GEnx jet engines alone collect and analyze 5,000 datapoints every second to detect problems and optimize performance.

The scale of this data exchange is enabled partly by a new design feature of commercial airliners — what we might call platformization — meant to optimize distribution of both data and energy resources. Whereas an airplane of a previous generation uses its engines to pump hydraulic fluid, pressurize cabin air, and generate electricity, new-generation airliners use their engines only for electrical generation and then use electrical pumps and compressors to pressurize the cabin and move flight surfaces.

Similarly, flight systems operate on common buses, making controls like throttles accessible to many different systems in the flight deck: the physical controller that a pilot operates; autopilot systems that try to minimize fuel usage; safety systems that monitor problems in the flight deck; and data-collection networks that inform maintenance, upgrade designs, and pilot training.

21. *http://www.faa.gov/nextgen/snapshots/slides/?slide=6*

www.dbooks.org

This model is analogous to many new-generation architectures in other areas of the industrial internet. Airliners are bundles of interchangeable systems — jet engines, avionics, seats, entertainment systems — that are carefully integrated and operate as services. And they all produce extraordinary amounts of data.

At the fleet level, one of the most promising applications of the industrial internet is in health maintenance — carefully planning, coordinating, and carrying out maintenance and upgrades across many aircraft and many maintenance bases.

Lockheed-Martin's forthcoming F-35 fighter jet, for instance, will use what its maker calls the Automated Logistics Information System (ALIS) to manage maintenance automatically. Sensors in the planes detect mechanical wear and other problems and automatically requisition replacement parts before the plane even lands. The system keeps track of maintenance certifications and composes checklists for mechanics who service the planes — work that, if done by hand, would have the undesirable combination of high stakes and repetitiveness. It also reduces the very high costs of maintaining specialized parts inventories around the world.

Real-time aviation networking tends to be constrained by bandwidth, which forces airlines to rely on a continuum of intelligence — processing some data locally, streaming some data, and waiting until a plane is on the ground and connected to a high-bandwidth pipeline to use the rest of it. Richard Ross, vice president for IT at Atlas Air, an air-cargo company that operates one of the world's biggest Boeing 747 fleets, says "we can get quite a rich data stream from the plane — to the point where we had to become more selective about what we streamed real-time versus what we gathered once the plane landed, because satellite communications can quickly become too expensive."

Integrated data collection is important because airlines can build nuanced models out of collective maintenance and performance data. "Older 747s have been in the air for more than 25 years, and we know a lot about the maintenance they need," says Ross. "Epidemiology is a useful model for thinking about the analysis of such a large dataset and for drawing conclusions about cause and effect over years of elapsed time and across different populations of planes."

The Centaur, an "optionally-piloted" airplane made by Aurora Flight Sciences, exemplifies the balance between remote and local intelligence in its operations. It can be flown directly by a pilot sitting in its

cockpit, by a pilot in a remote control center, or by a pilot in the cockpit *via* the plane's ground link. (In other words, Aurora has so comprehensively captured the mechanism of flight in its software that a pilot might as well fly the airplane he's sitting in through the digital pipeline rather than directly through the flight deck's physical links.)

The airplane itself is something like a Web service, accessible to any authorized user — on board or on the ground — and controlled through an API. The Centaur accounts for sometimes-weak ground connections by loading the plane with enough local intelligence to execute a flight plan without controller contact. John Langford, Aurora's founder, characterizes the Centaur as "masking the complexity of the machine"[22] to present a simplified interface to users.

As in commercial aviation, Langford says that the outdated air-traffic control system is holding back the deployment of more machine intelligence in aviation. "The node — the airplane — is very well developed," he says. "The problem is that it needs a more sophisticated network to work on than the air-traffic control system. The evolution of ATC is the limiting factor."

The model of airplane as intelligent Web service has profound implications for the accessibility of aviation. Langford points out that while a senior pilot might have 10,000 to 20,000 hours of flying experience, his pilotless operating system already has hundreds of thousands of hours of flying experience. "Every anomaly gets built into the memory of the system," he says. "As the systems learn, you only have to see something once in order to know how to respond. The [unmanned aircraft] has flight experience that no human pilot will ever build up in his lifetime."

Langford adds, "What we think the robotic revolution really does is remove operating an air vehicle from the priesthood that it's part of today, and make it accessible to people with lower levels of training." You might imagine a future in which pilotless aircraft, approved for domestic use, push down the cost of air transportation and make fast-dispatch airplanes available to anyone.

22. *http://radar.oreilly.com/2013/02/masking-the-complexity-of-the-machine.html*

www.dbooks.org

Railroads

A train can be so long that its locomotives start to climb one hill while its mile of coal cars are still descending the last one. Cruise control that anticipates terrain can save lots of fuel (just like a driver who practices "hypermiling"[23] to save gas). It can save even more fuel if it also knows something about the urgency of a train's schedule and the likelihood that the train will need to pull onto a siding to wait for another train to pass.

The industrial internet promises to encompass entire railroads in integrated models that optimize everything from the placement of freight cars within a train to small variations in throttle. Delivered as a service, software can take into account an enormous range of contextual data to inform every decision, then control big machines in real-time.

GE's Trip Optimizer software, a kind of autopilot for locomotives, observes the entire context of a journey — the consist of a train, grades along its route, the urgency of its delivery — and controls its throttle in real-time (or, in the case of many big freight trains, it controls throttles on several locomotives independently). Movement Planner, another piece of GE software, can act either as an advisor or as a controller. It sits between transportation managers and dispatchers to, for instance, slow a train and save fuel if it will need to wait on a siding anyway.

Deborah Butler, Norfolk Southern's chief information officer, says her railroad has seen a 6.3% reduction in fuel usage and 10-20% increases in velocity by installing Movement Planner on its network. Better velocity also means better capital utilization — when trips are faster, the railroad needs less capital equipment to operate them. "If it works out as we think it will … we're going to need [fewer] locomotives than we need right now," she told a conference audience last fall, adding, "That being said, our business will grow and we'll eventually need a lot more locomotives, so it's all a good thing."

Like many companies, railroads have been amassing huge data stores without specific plans for analyzing them. Butler says Norfolk Southern has used helicopters to map every mile of its network in detail. "We know where every tree is growing beside the track," she says. "We

23. *http://en.wikipedia.org/wiki/Hypermiling*

aren't even beginning to use that data in the way that we could." Software that brings networked intelligence onto trains can use that data, though, optimizing dispatching at the system-wide level and second-to-second throttle settings at the locomotive level.

The impetus for some of this investment is a mandate to install a signal upgrade called positive train control (PTC). In 2008, a commuter-train engineer in Los Angeles ran through a red signal while apparently sending a text message, killing 25 people when his train collided with a freight train. In the ensuing uproar, Congress required railroads to install positive train control on tracks that carry passengers and chemicals by 2015. Like NextGen on airplanes, PTC joins trains to dispatchers and automated safety systems through high-bandwidth data connections. Locomotives and signals become nodes on a wide network, making it possible for a remote dispatcher or automated system to stop a locomotive when it fails to obey a signal.

As a pure investment in safety, positive train control offers abysmal returns. The Federal Railroad Administration's own estimates put the ratio of costs to benefits as high as 22 to 1.[24] And the costs are enormous: Union Pacific, the biggest railroad in the U.S., forecasts that its investment in PTC will total about $2 billion.[25]

Railroads must get much more than a modest safety improvement out of PTC, so they are treating it as a data backbone on which other systems can be built, and are using the congressional deadline as a target for resolving a host of interoperability problems that have built up over decades of piecemeal IT investment. Butler says that Norfolk Southern uses more than 5,000 messaging protocols between trains and dispatchers, along rights-of-way, and between back offices. On top of that, her railroad must accommodate messages from other railroads that interchange traffic with Norfolk Southern.

That's the sort of problem that Internet architectures are good at solving, and once railroads have turned their IT systems into modular networks, they'll find new opportunities for layering intelligence on top of their systems.

24. *https://www.federalregister.gov/articles/2010/01/15/E9-31362/positive-train-control-systems#t-1*

25. *http://www.sec.gov/Archives/edgar/data/100885/000119312513045658/d477110d10k.htm*

www.dbooks.org

Health care

Here is health care in software terms: connect medical expertise asynchronously to patients who need it, draw in many streams of data to formulate a diagnosis, and control many treatment methods simultaneously while carefully measuring their performance. Doctors and technologists have foreseen for a long time the potential for computers to ease communication bottlenecks and provide analytical insight, hoping that electronic medical records might be layered with intelligent algorithms that can develop diagnoses, identify risk, and suggest preemptive treatment.[26]

The industrial internet will help doctors make better use of the enormous volumes of machine data that modern health care tools produce (a single MRI session, for instance, can produce more than three gigabytes of data[27]), break down treatment silos by disseminating diagnostic data where it's needed, and create external connections that will aid in monitoring and treating patients outside of hospital settings.

The result, as in other areas of the industrial internet, will be pervasive intelligence, available any time and anywhere, that takes into account broader context than what is available locally.

Medical practices have invested billions of dollars in electronic medical records (EMRs) in pursuit of more intelligent care and in response to government incentives, and more than 70% of doctors now use some sort of electronic record-keeping system.[28] Many of these systems have failed to live up to their potential, though. Doctors complain that user interfaces are poor and that encoding their work consumes too much time (this, in turn, hurts data integrity because doctors take shortcuts). Data formats are complex and often proprietary, making it difficult to share data and to build collaborative systems on top of EMRs.

A computation chief at a large university hospital says that when he set out to build some predictive tools for his doctors, he reverse-engineered the 15,000 tables in his own EMR system, analyzing them

26. A full description of the impact of data on health care is beyond the scope of this paper. For more information, see this primer: *http://radar.oreilly.com/2012/08/data-health-care.html*

27. *http://hmi.ucsd.edu/pdf/HMI_Case_Neuroimaging.pdf*

28. *http://www.cdc.gov/nchs/data/databriefs/db111.pdf*

as though they belonged to a competitor, rather than endure the expense and further lock-in of commissioning the same tools from his EMR vendor.

That illustrates the value of data in health care, and points to the need for standardized data structures and open interfaces. Academic hospitals and other advanced users of medical data have begun to build specialized analytical layers on top of their systems, and startups have appeared that hope to apply successful big-data lessons from other industries to health care.

The industrial internet will help in several respects. Better use of machine data and software intelligence will reduce the drag of doctor-computer interactions and improve data integrity. Rather than taking blood pressure, reading the cuff's dial, and entering the result into an electronic medical record — or, even worse, writing down the result on paper for later encoding by a typist — a technician should simply be able to attach the cuff to a computer and upload the result in real-time.

Better network connections and software-machine interfaces will also improve what we might call "asynchronous treatment" — interaction between doctors and patients that doesn't require both to be in the same room together, or even available at the same time. Networked machines, and software at the provider level that controls them and analyzes their data, make intensive home monitoring and treatment possible.

Finally, hospitals face capital utilization problems much like those of airlines: they own very expensive equipment that must be operated constantly in order to earn an attractive return, and that equipment must be operated by highly-paid experts deployed as part of a network. The industrial internet can improve availability of machines and staff by integrating both into hospital-management systems, and might someday be able to substitute inexpensive long-term data gathering for very expensive hospital tests, relying on machine learning algorithms to tease diagnostic insights out of everyday data from common sensors.

Manufacturing

Manufacturing is becoming broadly accessible to innovators operating at small scale. Sophisticated prototyping facilities are available at

www.dbooks.org

minimal cost in maker spaces across the country, where anyone with a modestly technical mindset can make use of newly simple tools — not only microcontrollers like the Arduino, but also 3D printers, laser cutters, and CNC machine tools. Powerful computer hardware — controllers, radios, and so forth — has become so inexpensive that, at least at the outset, nearly any problem can be reduced to a control challenge that can be solved with software.

Large-scale manufacturing will benefit from similar trends that will make it ever easier to bring intelligence to big machines. Intelligent software will make manufacturing more accurate and more flexible. Processors that are powerful enough to handle real-time streams of sensor data and apply machine-learning algorithms are now cheap enough to be deployed widely on factory floors to support such functions as machine-wear detection and nuanced quality-control observation. And logistics tools that transmit real-time data on shipments and inventory between manufacturers, shippers, and customers will continue to reduce inventory costs.

In this manner, software running on an inexpensive processor, reading data from inexpensive sensors, can substitute for more expensive capital equipment and labor. Better understanding of maintenance needs means better allocation of equipment — since the timing of maintenance can be optimized if it's proactive rather than reactive — and workers can similarly avoid being idled or having their time absorbed in detecting maintenance needs.

Large manufacturers have invested billions of dollars in SCADA (supervisory control and data acquisition — the low-level industrial control networks that operate automated machines). Comprehensive stacks of specialized software link these systems all the way to management dashboards, but many of these systems have their roots in automation, not high-level intelligence and analysis. Factory managers are understandably conservative in managing these systems, and demand highly-robust, proven technologies in settings where the functioning of a big machine or assembly line is at stake.

Factory settings can be extremely difficult environments for computing. J.P. Vasseur, at Cisco, says that it's not uncommon to see 40% packet-loss rates in factory networks due to humidity and electromagnetic interference. Current systems often depend on simplified software — derived from ladder logic and implemented on programmable logic controllers — that is easy for workers to learn and use.

Internet Protocol-based architectures have found their way into industrial plants and have brought about modularity that makes these systems vastly more flexible and easier to update. Richard Ross, the head of IT at Atlas Air, was previously CIO at Hess Oil, where large networks of sensors were integrated with supply-chain systems and personnel databases to schedule preventative maintenance on oil platforms and refineries. "It used to be that the totality of the sensor network was proprietary to a given vendor. Now, with TCP/IP technology, the barriers to entry to put these things in are much lower, the costs to install and maintain them are much lower and there is much more vendor competition," he says. In airplanes, too, says Ross, "You're no longer making a commitment to a single vendor for the rest of the life of the plane." Modularity means that "software can be treated like a part — just like you'd change out an engine part."

Another former oil company CIO says he worked hard to build interoperability between his systems. "Much of our approach was to bust closed systems," he says. "We hated the model that would screw you forever on maintenance fees. We tried to go with the open-source model for life."

Modularity will also allow changes in the type of computing that factories use. In environments where hardened, simplified, and robust computers are called for, programmable logic controllers (PLCs) will remain the foundational computer for industrial processes. But pervasive network connections and flexible networks will distribute intelligence along a continuum from PLC to cloud, placing immediate, real-time processes at the level of the industrial control system and putting analytics and optimization processes at higher levels, where they will benefit from wider context and more powerful computers.

Rather than replace humans directly, the industrial internet will make them more productive, speeding the flow of information and giving workers tools for better decision-making. Industrial engineers say that they have retained old control systems based on simplified ladder logic because they're easy for workers to grasp without a technical background. Intelligent software can interact with humans intuitively, giving these workers access to powerful analytics and control systems.

www.dbooks.org

CHAPTER 4

The role of Silicon Valley in creating the industrial internet

A new kind of hardware alpha-geek will approach those areas of the industrial internet where the challenges are principally software challenges. Cheap, easy-to-program microcontrollers; powerful open-source software; and the support of hardware collectives and innovation labs[1] make it possible for enthusiasts and minimally-funded entrepreneurs to create sophisticated projects of the sort that would have been available only to well-funded electrical engineers just a few years ago — anything from autonomous cars to small-scale industrial robots.

In the same way that expertise in software isn't necessary to create a successful Web app, expertise barriers will fall in software-machine interfaces, opening innovation to a big, broad, smart community.

Neil Gershenfeld, director of the Center for Bits and Atoms at MIT, compares the development of the amateur hardware movement to the development of the computer from mainframe to minicomputer to hobbyist computer and then to the ubiquitous personal computer. "We're precisely at the transition from the minicomputer to the hobbyist computer," he told a conference audience recently. He foresees a worldwide system of fabrication labs that produce physical objects locally, but are linked globally by information networks, enabling expertise to quickly dissimilate.

1. *http://radar.oreilly.com/2013/02/masking-the-complexity-of-the-machine.html*

www.dbooks.org

In complex, critical systems, clients will continue to demand the involvement of experienced industrial firms even while they ask for new, software-driven approaches to managing their physical systems. Industrial firms will need to cultivate technological pipelines that identify promising new ideas from Silicon Valley and package them alongside their trusted approaches. Large, trusted enterprise IT firms are starting to enter the industrial internet market as they recognize that many specialized mechanical functions can be replaced by software.

But the job of these firms will increasingly be one of laying foundations — creating platforms on which others can build applications and connect nodes of intelligence. These will handle critical functions and protect against dangerous behavior by other applications, as we already see in automotive platforms.

The industrial internet will make machine controls easier to develop in isolation from machines and easier to apply remotely. It's apparent, then, that markets in controls will arise separately from the markets in their corresponding machines. Makers of machines might reasonably worry that value will move from machines to software controls, leaving them with commodity manufacturing businesses (think of the corner case in which consumers buy a car from an automaker and then run practically all of its electronic services from their phones). Collaboration between machine makers and control makers is crucial, and the quality with which machines accommodate and respond to intelligent controls will become a key differentiator.

Silicon Valley and industry adapting to each other

Nathan Oostendorp thought he'd chosen a good name for his new startup: "Ingenuitas," derived from the Latin for "freely born" — appropriate, he thought, for a company that would be built on his own commitment to open-source software.

But Oostendorp, earlier a co-founder of Slashdot, was aiming to bring modern computer vision systems to heavy industry, where the Latinate name didn't resonate. At his second meeting with a salty former auto executive who would become an advisor to his company, Oostendorp says, "I told him we were going to call the company Ingenuitas, and he immediately said, 'bronchitis, gingivitis, inginitis. Your company is a disease.'"

And so Sight Machine[2] got its name — one so natural to Michigan's manufacturers that, says CEO and co-founder Jon Sobel, visitors often say "I spent the afternoon down at Sight" in the same way they might say "down at Anderson" to refer to a tool-and-die shop called Anderson Machine.

It was the first of several steps the company took to find cultural alignment with its clients — the demanding engineers who run giant factories that produce things like automotive bolts. The entire staff of the company, which is based in Ann Arbor, Mich., has Midwestern roots, and many of its eight employees have worked in the automotive industry. Sight Machine's founders quickly realized that they needed to sell their software as a simple, effective, and modular solution and downplay the stack of open-source and proprietary software, developed by young programmers working late hours, that might make tech observers take notice. They even made aesthetic adaptations, filling a prototype camera mount with pennies to make it feel heftier to industrial engineers used to heavy-duty equipment.

Heavy industry and the software community will both need to adapt their approaches and cultures in order to make the most of the industrial internet.

The technology industry can easily overreach when it begins to think of everything as a generic software problem. Physical-world data from machines tends to be dirty, and it's often buried in layers of arcane institutional data structures. (One airline-servicing company found that its first client had 140 tail numbers in its database — but only 114 planes.) Processes in established industries are often the result of decades (or centuries) of trial and error, and in many cases they're ossified by restrictive labor agreements and by delicate relationships with regulatory bureaucracies.

The demands of the industrial world mean that some of Silicon Valley's habits developed over years of introducing new services to consumers will need to change. Industrial systems can tolerate downtime only at enormous cost, and their administrators are only willing to install new services if they've been thoroughly proven. "I've bought systems from startups," says an industrial engineer who works on fruit-juice processes. "They asked us to report bugs — they should be paying *us* for

2. *http://sightmachine.com*

www.dbooks.org

that service! Can you imagine running an industrial process on beta software?"

Many of the successful software firms that I spoke with operate as a blend of software startup — drawing bright developers from any background — and industrial firm with specialized engineers. The former bring the agility and innovation that's driving the industrial internet's transformation; the latter bring the credibility that these firms need in order to develop business with more conservative industrial companies.

As Sight Machine found, startups need to show industrial firms that they're serious in terms that their customers will understand. Dan Zimmerle, from the power-systems lab at Colorado State University, says he's approached at least once a month by a self-styled entrepreneur bearing a design for a perpetual-motion machine. "The skepticism of the industrial buyer is in some sense well-founded," he observes drily.

Industry, too, will need to change its approach in order to take full advantage of the industrial internet, perhaps by changing incentive structures in order to reward mid-level plant managers for controlling costs as well as for keeping systems running smoothly. As in much of information technology, the reward for saving money is small and incremental, and the punishment for a new system breaking is somewhat more dramatic.

Some real-time applications of machine learning, in particular, can sound informal and can spook industrial managers with their implicit promises to learn from mistakes. Managers would, of course, prefer to avoid mistakes altogether, but machine learning has enormous promise in holding down labor costs, speeding output, and enabling flexibility. When capital budgets and responsibility for smooth operations sit with different people, the right level of risk-taking is unlikely to emerge.

CHAPTER 5
Conclusion

The industrial internet is the expression of the Internet's structure and practice in the parts of the physical world where it stands to have the greatest impact on "stuff that matters." Like the Internet, it invites the wide participation of anyone who cares to contribute expertise or ingenuity in solving a modular problem and scaling it across the world.

A century and a half ago, the machines that our lives now depend on were being invented and refined in basement workshops (as well as a few prototypical corporate research operations). As machines improved, the costs and expertise needed to improve them further grew, and the age of garage foundries for big machines largely passed.

Software, with the industrial internet, stands to reinvigorate machine innovation. Every few years, the software industry is reshaped by dorm-room entrepreneurs; the industrial internet will bring some of their spirit to the physical world at the same time that it affirms the value of big machines.

www.dbooks.org