DZone
RESEARCH

DZONE RESEARCH PRESENTS

2014 GUIDE TO INTERNET OF THINGS

BROUGHT TO YOU IN PARTNERSHIP WITH

Electric Cloud
redhat.
salesforce 1 Platform
WSO2

WELCOME

Dear Reader,

This Sunday I got up a little earlier than the rest of my family, so I grabbed my iPad and a coffee and went out to the back porch to think. A few clicks later I had awakened the stereo receiver in the living room, set the volume on the porch speakers, and selected a favorite station from my net radio bookmarks. I needed some information from one of the whiteboards at the office, so I connected to our telepresence robot and drove it into the conference room to snap a picture and email it to myself.

The Internet of Things isn't coming, it is already here. Think about what you can do today that you couldn't do ten years ago, and imagine what it will be like ten years from now. From the quantified self to home automation and geofencing, human lives are rapidly changing as a result of connected devices that entertain us, save us time and money, and may even help us live longer and stay healthier.

What we've seen so far is just a prelude to a sea change that will profoundly affect society and generate unprecedented opportunity for developers. Cisco projects the economic impact of IoT will exceed $15 trillion annually by 2020. Nothing in the history of the world has ever been so big, and those who can design and program this multitude of new devices will enjoy a big slice of the new wealth. Being a developer in the web era has been great so far, but it's about to get even better for those who are shrewd enough to gain early advantage in IoT.

Download some SDKs and get started programming for this new world of connected devices. In a few years, when you're telling your phone OS how to manage your investments as your driverless car takes you to work, you'll be glad you did.

RICK ROSS
CEO
research@dzone.com

TABLE OF CONTENTS

CREDITS

DZONE RESEARCH

JAYASHREE GOPALAKRISHNAN
DIRECTOR OF RESEARCH

MITCH PRONSCHINSKE
SENIOR RESEARCH ANALYST

BENJAMIN BALL
RESEARCH ANALYST

MATT WERNER
MARKET RESEARCHER

JOHN ESPOSITO
REFCARDZ EDITOR, AUTHOR

ALEC NOLLER
CONTENT CURATOR

MIKE LENTO
GRAPHIC DESIGNER

Special thanks to our topic experts Matt Butcher, Andreea Borcea, Sean Lorenz, and our trusted DZone Most Valuable Bloggers for all their help and feedback in making this report a great success.

DZONE MARKETING AND SALES

ALEX CRAFTS
SENIOR ACCOUNT MANAGER

HUNTER MURPHY
ACCOUNT MANAGER

KELLET ATKINSON
DIRECTOR OF MARKETING

ASHLEY SLATE
DIRECTOR OF DESIGN

CHRIS SMITH
MARKETING ASSOCIATE

CHELSEA BOSWORTH
MARKETING ASSOCIATE

DZONE CORPORATE

RICK ROSS
CEO

MATT SCHMIDT
PRESIDENT & CTO

BRANDON NOKES
VP, OPERATIONS

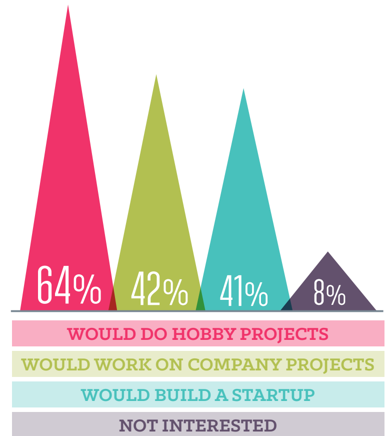
HERNÂNI CERQUEIRA
LEAD SOFTWARE ENGINEER

SUMMARY & KEY TAKEAWAYS

The Internet of Things is going to change the world, including our development and IT strategies for building many types of applications. Some firms are already predicting that the Internet of Things (IoT) will generate over $19 trillion in value over the next decade [1]. While much of the future for IoT is still unknown, we are now starting to see an early wave of emerging products and the new challenges associated with building them. *DZone's 2014 Guide to Internet of Things* is an early mover's map for navigating this bleeding edge space and finding your place in it. The data and content in this guide and on dzone.com/research/iot will provide you with:

- The most promising IoT product lines and the information you need to start developing on their platforms.
- The programming and networking challenges of building IoT applications, and some strategies for overcoming these challenges.
- A review of IoT hardware and development platforms for do-it-yourself IoT projects.
- Over 2,000 IT professionals' opinions about IoT issues and domains.
- An outlook for the next 2-3 years in IoT.

DEVELOPER INTEREST IN IoT DEVELOPMENT

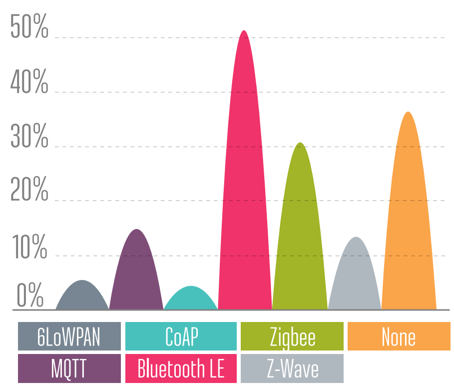


KEY TAKEAWAYS

DEVELOPER INTEREST IN IoT IS SIGNIFICANT

IoT product development already has significant interest in the software development community, and forward-thinking developers are already taking steps to familiarize themselves with this technological shift. One quarter of respondents have actually worked on an IoT project, so these types of products are already a significant factor in the development space. It starts as a hobby for most developers, with 64% of our survey respondents indicating their interest in hobby IoT development; but they know that a simple project started on a whim can become a flourishing business. Of the respondents, 41% were interested in building an IoT startup if the opportunity arose, and only 8% weren't interested in IoT projects at all.

WHICH OF THESE POPULAR IoT TECHNOLOGIES ARE YOU AWARE OF?



THERE IS NOT ENOUGH AWARENESS OF IoT TECHNOLOGY TRENDS

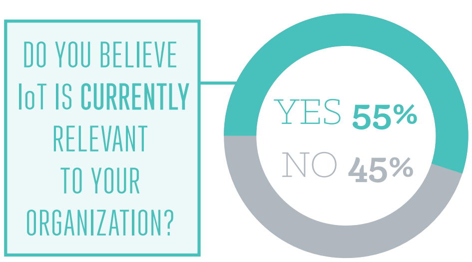
Despite growing levels of interest, a majority of the surveyed developers have not even looked at the major IoT communication protocols outside of WiFi and Bluetooth, and very few have worked with new technologies such as iBeacons. While over half the respondents reported that they knew about Bluetooth Low Energy (52%), it is possible that many are not aware of the differences between the Bluetooth and Bluetooth Low Energy protocols. The second highest answer for protocol awareness was None with 36%.

SMART HOME TECHNOLOGY CURRENTLY ATTRACTS THE MOST DEVELOPER INTEREST

Smart home technology is the highest ranking IoT domain according to respondent interests. With innovative products like Nest and new platforms like Apple HomeKit, developers see a big opportunity to make money by building apps on top of technologies that they would gladly buy themselves. In fact, 29% of respondents own smart home products or have built their own. Most of these products are used for smart utilities, security, or lighting.

THE IMPORTANCE OF IoT IS WELL UNDERSTOOD IN IT ORGANIZATIONS

With the amazing IoT systems that technologists are now imagining, it's no surprise that VCs, IT firms, and developers themselves are hyped about IoT. 55% of respondents believe IoT is relevant to their organization right now, and 83% of respondents are confident that IoT will be relevant to their organization in the future. That kind of confidence from the technologists leading us into our IoT future is hard to ignore.



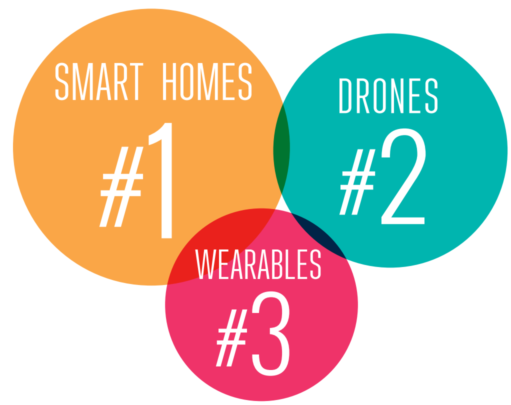
KEY RESEARCH FINDINGS

Over 2,200 IT professionals responded to DZone's survey for the *2014 Guide to Internet of Things*. The resulting data reveals insights about who's working on IoT, what IoT domains are interesting to developers, and what they have actually built for the IoT space. The largest demographic segments are developers (40%) and development team leads (25%). 52% of respondents come from large organizations (100 or more employees) and 48% come from small organizations (under 100 employees). The majority of respondents are headquartered in the US (35%) or Europe (38%).

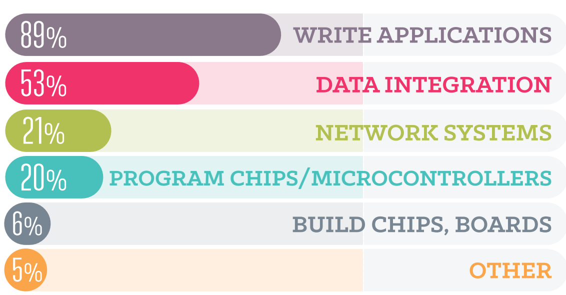
SMART HOME TECH HAS THE HIGHEST DEVELOPER INTEREST

One fourth of survey respondents have worked on an IoT project, and the most popular domain for those projects is in smart home technology. When ranking their IoT use case interests, smart home technology was ranked the highest, followed by drones and wearables. When indicating which domains they have worked in (or would like to work in), 17% of respondents said they had built smart home apps, while 62% said they would like to try building those applications in the future. Many respondents also want to work on drones (53%), wearables (48%), and several other technologies, while less want to work on geofencing (35%) and smart supply chain tech (38%). Despite its high popularity in the survey, very few respondents have actually worked on wearables, but that might change quickly with the new Android Wear platform.

IoT USE INTEREST RANKINGS



WHAT TASKS ARE YOU MOST LIKELY TO HANDLE WHEN DEVELOPING AN IoT PROJECT?

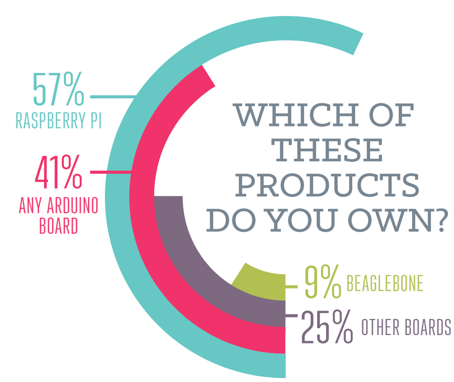


MOST DEVELOPERS WANT TO STICK TO THE SOFTWARE OF IoT

89% of respondents say that their most likely task on an IoT project would be building the applications that handle sensor data and perform other tasks. Over half of the respondents (53%) would also expect to help with data integration. Few respondents wanted to assemble the hardware (6%). Respondents also preferred to work with Linux if possible (69%), rather than real-time operating systems or direct microcontroller programming. If respondents were given a choice between working on an existing IoT project and starting a new one from scratch, the preferences were split almost evenly with 52% opting for existing projects and 48% hoping to start from scratch.

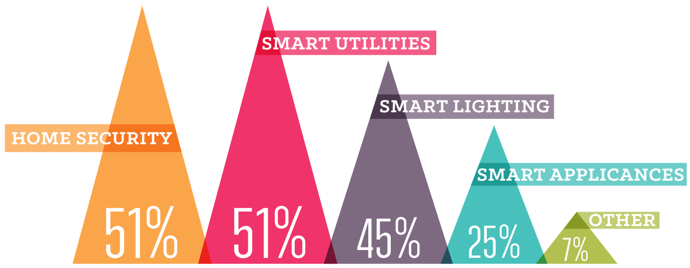
RASPBERRY PI AND ARDUINO LEAD THE PACK IN HARDWARE HACKING

57% of the respondents own a Raspberry Pi, which is a very popular single-board computer (SBC) that can be used as a compact IoT device hub, or as the sensor devices themselves. 41% own some type of Arduino board (an open source single-board microcontroller), 9% own a BeagleBone Black (another SBC), and 25% own other prototyping boards. When asked about consumer products that they are generally interested in hacking on, respondents were most interested in Google Glass and smartwatches.

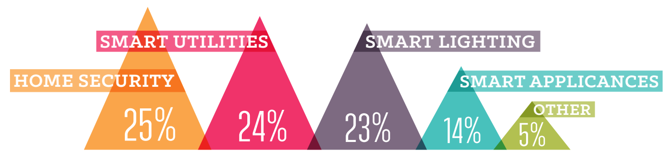


MANY DEVELOPERS HAVE DIY SMART HOME PRODUCTS

29% of respondents own a smart home device, and out of that group 51% own a smart utilities device, 51% own smart security devices, and 45% own smart lighting devices. Being a highly technical audience, the survey respondents built their own devices in these areas about half of the time. Of the smart home product owners, 25% built their own smart security devices, 24% built their own smart utilities, and 23% built their own smart lighting.



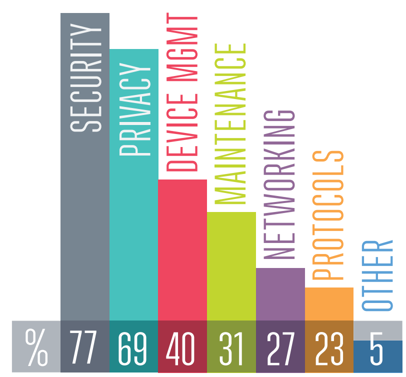
WHAT SMART HOME DEVICES DO YOU OWN?



WHAT TYPES OF SMART HOME PRODUCTS HAVE YOU BUILT OR DEVELOPED, IF ANY?

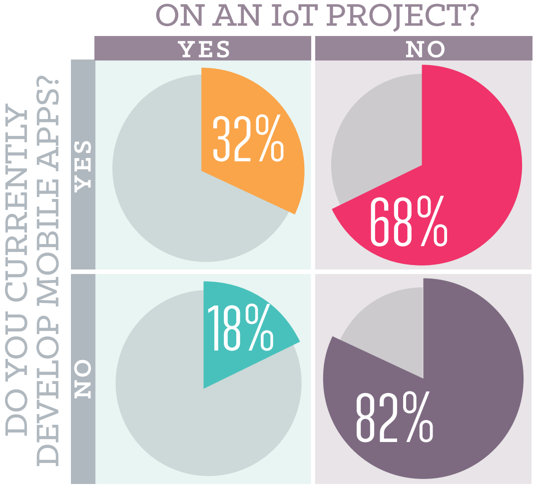
SECURITY AND PRIVACY ARE THE BIGGEST CONCERNS FOR IoT

With every new technology paradigm comes new security concerns, and in this case, privacy concerns as well. IoT systems are built on sensors that are frequently collecting data, and sometimes those systems are collecting personal information that is being shared without people's knowledge. It's not a guarantee that IoT will usher in a new era of significant privacy invasion, but it certainly has the potential to do that. That's why 36% of respondents are very worried about privacy invasion from technologies like geofencing, smart cameras, and wearables. A larger percentage (53%) are only slightly worried, but very few (11%) are not worried at all. The security of IoT data also has a higher risk of being intercepted because in many cases there are a massive number of low-security nodes where infiltrators can enter the network. Security may be one of the biggest problems to solve before IoT becomes more widespread. In the chart below, you can see that security (77%) and privacy (69%) are the largest concerns regarding IoT.



WHAT ARE YOUR CONCERNS REGARDING IoT?

HAVE YOU EVER WORKED ON AN IoT PROJECT?



MOBILE DEVELOPERS ARE MORE LIKELY TO WORK ON IoT

Smartphones are compact computers with fairly good performance, which keeps improving due to consumer demand. Their user interfaces are usually top notch, and they are packed with all kinds of sensors. There's no doubt that mobile devices are already a key component for IoT systems and products. In many cases, mobile touchscreens will be the window into IoT product control and data visualizations. It's no surprise that mobile developers from our survey are more likely to have worked on IoT projects. Almost twice as likely, as you can see in the graph to the left.

AN EARLY MOVER'S GUIDE TO IoT

BY ANDREEA BORCEA

The creation of the Internet was a significant shift in the way people acquire information, interact with each other, and make decisions. Now, the Internet is expanding its reach to a range of devices that can gather and analyze physical data and react to that data in a variety of applications that we've never seen before. This "Internet of Things" marks another dynamic shift in the history of technology.

This new stage in the Internet's evolution is changing it from a tool that we actively need to engage with—deliberately using a browser to access it—to one that passively endows the world around us with a "mind" of its own. We are developing a world where things interact intelligently and cooperate to achieve goals without explicit guidance from human operators.

DEFINING THE INTERNET OF THINGS

First we need to define the Internet of Things (also called "The Internet of Everything" by Cisco). A system falls under the Internet of Things definition if it meets the following criteria, known as the 3 Cs:

1. It must **Connect** – to the physical world around itself collecting information, to other things in order to interact with them effectively, to the internet or a network, etc.

2. It must **Compute** – by processing the inputs it receives in some way and making them meaningful to other systems.

3. It must **Communicate** – with the network, with other things, and with the user if necessary (more often than not, as you'll see, communicating to the user may be an unnecessary burden).

CHALLENGES FOR THE INTERNET OF THINGS

EFFICIENCY

Devices within the Internet of Things (IoT) only need to do the bare minimum necessary to effectively work within the existing ecosystem. Many of the newest products rely heavily on the power of your smartphone to connect to the Internet and orchestrate devices, but there is also extensive pressure to reduce the size, energy consumption, and cost of the processing entities within IoT devices. In order to reduce power consumption and manage node outages, there is a concept of daisy-chaining across a network of devices into a more powerful central hub. This is known as mesh networking, and it's becoming quite popular for IoT systems.

SECURITY, PRIVACY & THE NEED TO SHARE

A core requirement of a well-functioning IoT device is to collect, transfer, and store data from a wide variety of sources. As more sensors arrive in cities and healthcare institutions, that increasingly connected information will unavoidably lead to more concern about security and privacy.

The debate is still raging over balancing the clear benefits of new discoveries from processing Big Data with the strong personal fear of losing privacy. With IoT now in the picture, there is concern about devices that continuously and passively collect information on users. One recent clash over always-on sensors came with the release of Microsoft's Xbox One Kinect console, which has a camera that is constantly pointed at your living room. Although the camera itself is not always on, the backlash over that possibility was fierce [1]. Finding this balance will quickly become a requirement for continued progress.

Furthermore, the very nature of IoT and the connectivity network necessary for its success does make it particularly vulnerable in certain instances. Devices are especially vulnerable when connected over WiFi, because low tech sensor nodes with minimal computing power tend to be less secure, making them the ideal point of entry for infiltrators.

STANDARDS

As with all new technologies, the battle over standards is always a struggle. Nest, the company that developed the most popular smart home thermostat, and its new owner, Google, are now making significant strides trying to establish the Nest platform as the foundation for all consumer-based IoT devices and their software counterparts [2]. Cisco, Qualcomm, IBM, Microsoft, and most other major players have a similar strategy for creating standard models for approaching the Internet of Things.

The pressure to standardize is especially clear when new devices are appearing weekly. ZigBee already has extensive reach as an established standard for many household IoT devices. However, as a preferred codebase has yet to emerge as the standard of choice, it is recommended to connect with major standardization organizations like the IEEE, IETF, and the ZigBee Alliance [3]. Currently, the most common sensor networks use protocols such as Bluetooth Low Energy (BLE), RFID tags, ZigBee, and Wi-Fi. There are also iBeacons, which allow devices like smartphones to better identify their location and potential needs with NFC-powered micro-location and GPS technology.

OPPORTUNITIES FOR THE INTERNET OF THINGS

There are numerous prospects to consider when looking to develop IoT products. Given the multi-trillion dollar projections for the future IoT economy, we should take a look at these emerging markets for IoT tech [4].

CONSUMERS

The consumer IoT space has bred a small but growing segment of followers that have invested early into "smart" tech. At this year's CES, we saw everything from the Babolat Tennis Racket that becomes your personal tennis coach to the Kolibree Toothbrush that monitors your gum health while you brush. The fastest growing consumer IoT segment seems to be in smart home technology, with products such as self-managing refrigerators and resident-sensing door locks.

COMMERCIAL

Retailers have already proven adept at collecting a consumer's shopping history. With the functionality of NFC-powered beacons, these retailers are eager to personalize your shopping experience in a whole new way. Essentially, each physical shopping trip can now be as littered with targeted ads as any typical online search, much like a scene from the 2002 sci-fi film ***Minority Report.***

Walk into a store and instantly the advertising screens on the wall change to address your particular demographic, income level, and shopping preferences. If you've connected your Google calendar to certain applications, these screens would show outfits targeting your next big event. Signs on clothing racks sense you coming near and change prices, fully leveraging a custom pricing model that would have economists drooling. And as you try on outfits, the smart mirror in the dressing room recommends accessories or comments on alternatives that might be a better fit for your body type. After all of these IoT events have helped you with your purchase, there's no need to checkout. You've registered with the store and there's a beacon at the exit that registers what you picked up and charges your card automatically as you leave.

HEALTHCARE

With the recent U.S. mandate that all health records must be digital, there has been an explosion in the marketplace of new, patient-centered, smart health devices. The excitement of a healthcare revolution among top innovative companies, incubators, and startups predicts that this trend is not likely to taper off anytime soon. The key areas of focus so far have been: monitoring technologies like wearables (especially passive monitoring), function-improving technologies, education, and notification technologies.

THE MAIN OBJECTIVE IS TO COLLECT AS MUCH DATA AS POSSIBLE, MAKE IT AVAILABLE VIA OPEN APIS, AND ENCOURAGE MOTIVATED DATA ANALYSTS TO FIND OPPORTUNITIES FOR IMPROVEMENT

Wearables are generally the first consumer touch point in the IoT health sphere. With the popularity of Fitbit pedometers and Withings scales, the market is starting to experiment with internal monitoring and potentially replacing some organs completely in the near future. A study at Boston University has had incredibly positive results creating an artificial pancreas for Type 1 diabetics by inserting an insulin and glucagon pump that responds when an attached glucometer goes below a certain level, just like an actual pancreas.

Proteus, a promising startup out of San Francisco, has created an all-natural microchip in a pill that the patient swallows in order to monitor whether they are remembering to take their medication. The pill sends data to an armband that the user is wearing, which then can send notifications to family members regarding the patient's status. The most impressive feature is the fact that these chips are powered by the energy in the patient's digestive system.

CITIES, INFRASTRUCTURE, AND INDUSTRY

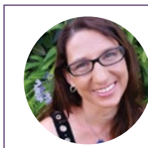
The long-term vision of the future includes technology such as self-driving cars and city lights that alert police when there's been an accident. In this stage of development, the majority of value is coming from technologies that monitor and collect data in urban settings. From an evolutionary perspective, the IoT city as a whole is still in what many would consider a learning phase. The main objective is to collect as much data as possible, make it available via open APIs, and encourage motivated data analysts to find opportunities for improvement in utility usage, environmental impacts, and service management for larger populations.

This is one area where being an industrial country like the U.S. may actually impede the ability to progress as quickly as our less established counterparts. Third world countries that haven't yet built a solid infrastructure allow for the creativity and flexibility to implement sophisticated solutions unfettered by generations of previous development. Silicon Valley powerhouses like Facebook and Google are actively engaged in projects to create a free global Wi-Fi network, and key locations in Africa have allowed them to experiment with these projects.

BEING AN EARLY MOVER

In the very near future, as more and more things connect to the Internet, Internet connectivity from IoT devices will dwarf the amount of traditional web browsing. The core standards and assumptions that will drive this next revolution in computing technology are still being established and, as a result, building anything that can add value to this exploding industry (software, hardware, devices, sensors, beacons, etc.) is a remarkable opportunity for the right developer. Right now is the time to start contributing to the development of these technologies if you want to be an early mover in IoT.

[1] http://www.businessinsider.com/xbox-one-kinect-privacy-issues-2013-5
[2] http://9to5google.com/2014/06/23/nest-to-share-some-user-data-with-google-open-up-third-party-development-platform/
[3] http://www.elsevier.com/__data/assets/pdf_file/0010/187831/The-Internet-of-Things.pdf
[4] https://internetofeverything.cisco.com/sites/default/files/docs/en/ioe_public_sector_vas_white%20paper_121913final.pdf
[5] Image of XBox One Kinect: http://news.xbox.com/2013/06/privacy
[6] Image of Nest Thermostat: https://nest.com/press/#product-images



WRITTEN BY
Andreea Borcea
Andreea Borcea has a passion for digital innovation and is currently leading the development of patient-empowering solutions for the healthcare industry. Keep up-to-date on the latest in IoT with Farstuff: The Internet of Things podcast, hosted by Andreea Borcea and Charles Wiltgen. Check it out at Farstuff.com

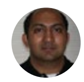
EVOLVING THE INTERNET OF THINGS THROUGH INFRASTRUCTURE

The Internet of Things (IoT) is changing the way businesses use technology. The Internet has become a medium to connect people, share information, find solutions, and report problems. With the advancement and availability of smart devices and smart machines, the Internet will evolve to the Internet of Things—billions of interconnected devices and machines measuring, collecting, analyzing, monitoring, and sharing useful information. This web of interconnected devices and machines gathers information from sensors in cornfields and bio-feedback chips in cows; it is embedded in smart manufacturing machinery, telemetrics, and smart meters. The flow of data brings new levels of insight to help organizations efficiently pursue opportunities. But to make a complex system of devices communicate with one another in a meaningful way—and do so profitably—requires infrastructure that is reliable, stable, and secure.

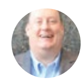
> RED HAT'S FLEXIBLE INFRASTRUCTURE AND ELASTIC SOLUTIONS BRINGS THE INTERNET OF THINGS TO LIFE.

For enterprises capitalizing on this promise of a truly interconnected world, Red Hat provides open source solutions that meet rigorous technology requirements through a combination of physical and virtual architecture implementations. With applications spanning from device, to field controllers, to the datacenter, Red Hat technologies enable reliable and secure information sharing, connection, and movement. Doing more than just transporting data, Red Hat helps you rapidly transform, process, and act on the data where and when it's needed. Red Hat's flexible infrastructure and elastic solutions bring the Internet of Things to life quickly, with lower risk, and cost-effectively. Visit http://www.redhat.com/embedded for more details.

WRITTEN BY



Sameer Parulkar
Product Marketing Manager, **Red Hat**



James Kirkland
Chief Architect for the Internet of Things and Intelligent Systems, **Red Hat**

IoT PLATFORM MIDDLEWARE

Red Hat JBoss A-MQ BY RED HAT

redhat.

JBoss A-MQ is extensible, and can either be deployed alone or with embeddable deployments while supporting multiple protocols and platforms.

TOP USE CASES

1. Transportation
2. Smart Service Sector/Retail
3. Smart Grid/City/Utilities
4. Smart Supply Chain/Manufacturing
5. Medical

STRENGTHS

1. Support for multiple protocols (MQTT, XMPP, STOMP, AMQP and more)
2. Support for multiple platforms (Red Hat Enterprise Linux, Windows, Solaris, AIX)
3. Lightweight footprint: stand-alone or embeddable deployment options
4. Open Source - community powered innovation

NOTABLE CUSTOMERS

- Etrade
- Specsavers
- Vodafone
- Sabre
- Meteorcomm

LANGUAGES

JAVA C C++ JAVASCRIPT .NET

MESSAGING PROTOCOLS

- MQTT
- AMQP
- STOMP
- OpenWire
- WebSockets

HOSTING OPTIONS

ON-PREMISE

OPEN SOURCE

YES

NATIVE CAPABILITIES

- Messaging
- Web services
- Analytics services

FULL PROFILE LINK dzone.com/r/UPJf

WEBSITE redhat.com

TWITTER @RedHatNews

THE PROGRAMMING CHALLENGES OF IoT

BY JOHN ESPOSITO

Pragmatic developers can look at the Internet of Things in two ways:

- **This is amazing.** I can only begin to imagine how I can directly improve the world outside the set of networked computer boxes.
- **This is terrifying.** If something goes wrong, then it's on me—and this time the system affected extends outside the set of networked computer boxes.

IoT is amazing in the way it bridges physical and virtual environments, but even the phrase "Internet of Things" should give a developer pause. Computers are pretty smart. Things are stupid. IoT tries to put Things online and tries to make them into inter-networked computers.

That's pop-philosophy, but you want to develop in the real world. So what real-world challenges will you face when you shoot for the IoT moon?

TWO TYPES OF CHALLENGES

It seems there are two types of programming challenges for the Internet of Things:

- Data and control (the comp-sci and networking stuff)
- Information and business logic (the info-sci and human-computer interaction stuff)

For this article, we're going to talk about the programming problems we can solve around IoT. We'll start at the bottom (data and control) and work our way up to the big picture (information and business logic).

TYPE 1: DATA AND CONTROL

CHALLENGE 1.1: POWER

This one is pretty obvious. Many IoT devices are wireless, and no one has invented thumbnail fusion reactors yet. One solution is equally obvious: pick your algorithms carefully. If you can save cycles to perform a given task, then do it. Libraries for implementing power-optimized algorithms will presumably spring up in greater numbers, but even so, you may need to inject some heavy-duty comp-sci know-how into IoT app development.

The second solution is more complex than the first. Higher-level developers will have to think more about Dynamic Power Management (DPM), which just means shutting down devices when they don't need to be on and starting them up when they do. Normally the operating system worries about this, but an IoT application that orchestrates wearables and phones, for example, will know things that each device's OS won't—and therefore will be able to switch things on and off more intelligently than each device's individual OS. Another option is to write or customize an embedded OS.

CHALLENGE 1.2: LATENCY

Latency on IoT sits in two places: at the source and in the pipes. The basic problem is a physical one. Thing-chips often have to be small, which means that the chip can only be as powerful as current transistor technology allows. Another problem is power. Many small devices transmit and receive data in discrete active/sleep cycles (think TDMA) in order to save bandwidth and power, but this increases latency inversely to power saved.

Another tradeoff is that network topologies optimized for IoT can involve more hops over slower devices. Mesh networks, for example, are immune to the failure of a few nodes. Similarly, "fog" and "edge" computing paradigms relieve Internet infrastructure by doing as much as possible without hub-nodes. The downside is that each node (a) can't do very much on its own and (b) can only talk to neighboring nodes.

COMPUTERS ARE PRETTY SMART. THINGS ARE STUPID.

The problem in the pipes is a matter of network infrastructure. Simply: the more Things, the less available bandwidth. Infrastructure technology will get faster, but cell networks won't catch up overnight. And Things, unlike fancier computers, are often supposed to transmit blindly—that is, without anyone necessarily asking them to. This means there's a massive potential for wasted bandwidth.

CHALLENGE 1.3: UNRELIABILITY

The third challenge flows from the first two. Devices are unreliable–"Things" even more so. The distributed and decentralized virtues of IoT bring their own reliability problems. Here are just a few:

- **Ubiquitous devices are cheap, so they fail more often.**
- **Truly ad-hoc connectivity implies ephemeral SLA, so uptime and recovery time may be unclear.**
- **Loosely controlled devices may have better things to do than give you their data (or computing resources), so concurrency may grow very complex.**
- **Less-reliable hardware generates less-reliable information ('does my outlying datapoint just signify device failure?'), so you may need to chew your data more thoroughly at the application level.**

THE SIZE OF DATA OVER IoT IS ITSELF A PROBLEM. WIRELESS SENSORS BEGET TONS OF DATA. ALL THE PROBLEMS (AND OPPORTUNITIES) OF BIG DATA CASCADE NATURALLY FROM IoT.

In a sense, IoT decouples low-level (the sub-session layer) from high-level channel capacity, because the distribution of error-sources on IoT is more heavily weighted toward originating or remote nodes. This means more error-correcting for application developers.

TYPE 2: INFORMATION AND BUSINESS LOGIC

CHALLENGE 2.1: VAST & THIN DATA

Sensors on smartphones are already generating oceans of raw data. These sensors are pretty sophisticated. Every major mobile OS provides a unified, simple API to access clean sensor and geo data. But start grabbing this data and it's not immediately clear what to do with it. Try to think of killer applications for barometric data—besides weather and elevation (with GPS)—off the top of your head. Raw sensor data is extremely thin. It doesn't explain itself, and we haven't yet produced a complete mapping from physical measurements to business logic—let alone software design.

Even if you know what to do with sensor/geo data eventually, you may have to learn new algorithms and data structures to process immediately. Geo-graphs aren't CS101 graph data structures (for one thing, edge length is a first-class citizen of geo-graphs).

The size of data over IoT is itself a problem. Wireless sensors beget tons of data. All the problems (and opportunities) of Big Data cascade naturally from IoT. Massively distributed computing on IoT devices is an exciting thought, but the toolchain for splitting calculations over a thousand idle Fitbits just isn't here yet.

2. CONTEXT-SENSITIVITY

Consider the term "ubiquitous computing," defined as: what happens when wirelessly connected sensors and actuators, placed more or less everywhere, allow software to interact with much larger swaths of the physical world than just hardware or bare metal. Put ubiquitous computing on the Internet, and IoT makes the software context much larger. This has implications at two basic levels.

At a high computer-architectural level: IoT extends the concept of computing environment well outside the von Neumann machine and weakens the concept of peripheral I/O. In an IoT-world interface, sensors are input and actuators are output. As IoT devices process increasingly at the edge (within individual nodes), the devices that appear peripheral to other nodes are actually doing an awful lot of computation.

At a high business-logic level: the more stuff outside the computer-box affects the program, the less predictable the program behavior becomes at runtime. The same bizarrely-birthed memory leak might slow down the UI in a smartphone context but contribute to a cascading electrical grid failure in an IoT context. This means that IoT demands more self-monitoring and self-repairing code.

TWO TYPES OF SOLUTIONS

Plenty of researchers are working on ambitious solutions to the programming challenges presented by IoT. Two of the more exciting examples include:

- Abstract Task Graph—**a data-driven model that maps the network graph to an application graph [1]**
- Computational REST—**replaces content resources with computation resources [2]**

REACTIVE PROGRAMMING

This general purpose paradigm responds to all major application-level challenges and embraces opportunities presented by IoT. The four definitive attributes of a reactive application are: event-driven, scalable, resilient, and responsive [3]. These four are excellent guiding principles for IoT applications at a high, cross-stack level.

FLOW-BASED PROGRAMMING AND THE ACTOR MODEL

Both present strongly independent components where only messages can affect processes. Both are deeply amenable to concurrency (for example, shared state is discouraged), nondeterminism, and scaling without exponential complexity growth, because components are black boxes. FBP is a bit more pragmatic and restrictive while the actor model is less restrictive and a bit harder to implement. FBP has already been implemented in Javascript (NoFlo), and the actor model has been implemented in Java (Akka) [4][5][6].

THE SAME BIZARRELY-BIRTHED MEMORY LEAK MIGHT SLOW DOWN THE UI IN A SMARTPHONE CONTEXT BUT CONTRIBUTE TO A CASCADING ELECTRICAL GRID FAILURE IN AN IoT CONTEXT.

What's important to remember is that there are already tools and techniques that can help you build IoT applications. FBP, actors, and reactive programming all have key attributes for creating applications that leverage the strengths of IoT to overcome its challenges.

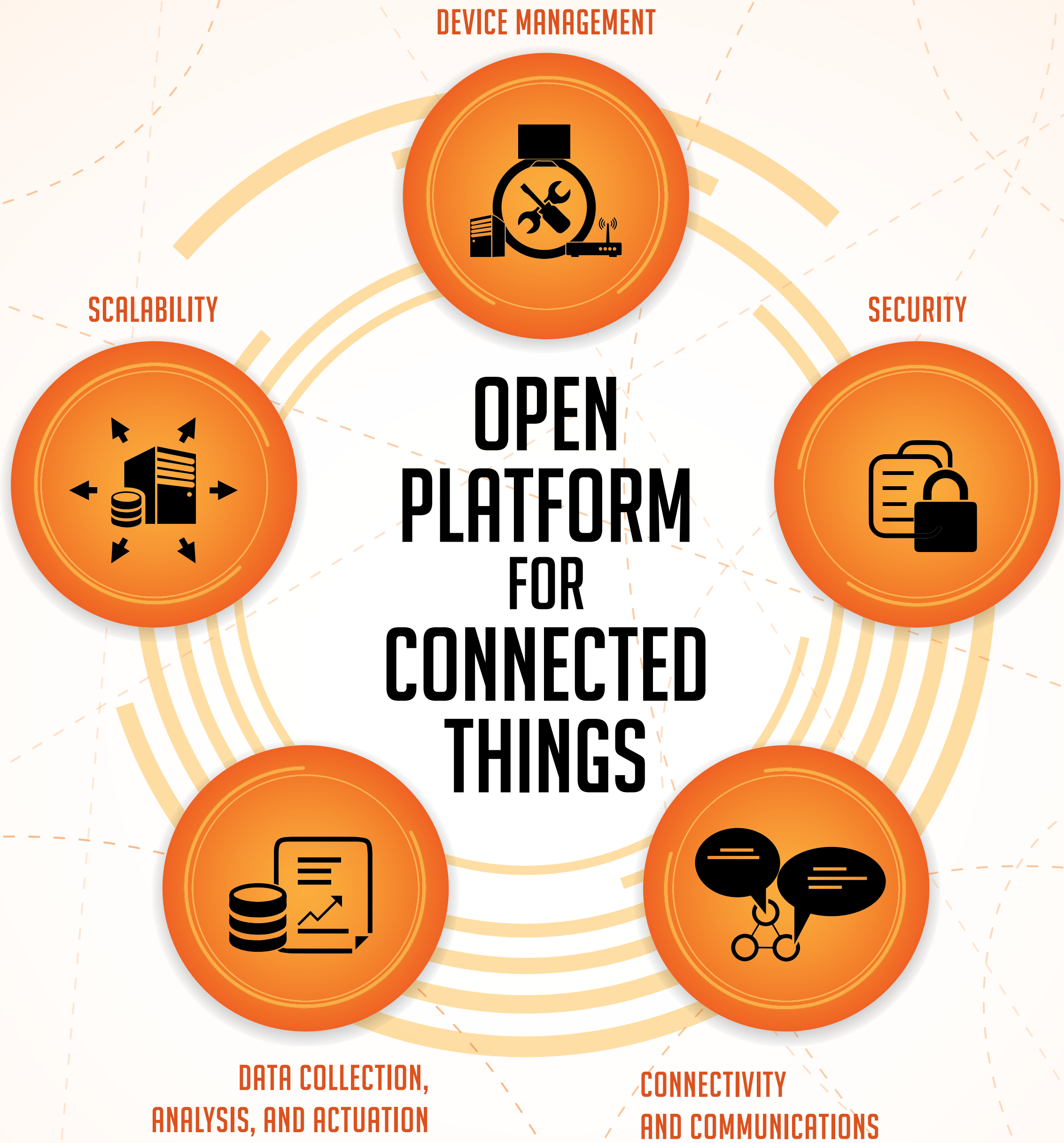
[1] https://www.usenix.org/legacy/event/mobisys05/eesr05/tech/full_papers/bakshi/bakshi.pdf
[2] http://isr.uci.edu/tech_reports/UCI-ISR-10-3.pdf
[3] http://www.reactivemanifesto.org/
[4] http://jpaulmorrison.com/fbp/
[5] http://noflojs.org/
[6] http://akka.io/

WRITTEN BY

John Esposito

John Esposito is a researcher and the Refcardz manager at DZone. In a previous life he was a database developer and network administrator. He is also writing a dissertation on ancient Greek philosophy.

OPEN PLATFORM FOR CONNECTED THINGS



The rapidly growing number and variety of Internet of Things (IoT) devices is driving the need for a highly scalable architecture with the flexibility to add or subtract capabilities, as well as support the many requirements of Internet-connected devices. With its 100% open source, comprehensive, componentized middleware platform, WSO2 is your ideal partner to get started. Contact us for a demo or visit:

http://wso2.com/landing/internet-of-things

for more details.

WHY YOU NEED **WSO2'S SCALABLE ARCHITECTURE** FOR IoT

The Internet of Things includes many categories of interaction, from wireless sensor networks to internet-connected wearables, low power embedded systems, and the use of mobile phones to interact with the real world, smart homes, connected cars, and more.

However, there is no single architecture model will suit all these categories and meet all their requirements. The result is that a modular, scalable reference architecture (RA) that supports adding and subtracting capabilities is useful and valuable.

The RA should cover multiple aspects including the cloud or server-side architecture that allows monitoring, managing, interacting with, and processing the data from IoT devices. It should also include the networking model to communicate with the devices, the agents and code on the devices, as well as the requirements for what devices can support this RA.

The overall requirements can be summarized into the following key categories:

- Data collection, analysis, and actuation
- Connectivity and communications
- Device management
- Scalability
- Security

> THE INTERNET OF THINGS COVER MULTIPLE DIFFERENT CATEGORIES. THE RESULT IS THAT NO SINGLE ARCHITECTURE WILL SUIT ALL THESE AREAS AND THE REQUIREMENTS EACH AREA BRINGS.

Details of the proposed RA can be found in the white paper, "A Reference Architecture for the Internet of Things," designed to provide architects and developers with an effective starting point for addressing the major requirements of IoT projects and systems.

The WSO2 platform is a completely modular, open source enterprise platform that provides all the capabilities needed to implement the server-side of this RA. An important aspect of the WSO2 platform is that it is inherently multi-tenant; this means that it can support multiple organizations on a single deployment with isolation between organizations (tenants). This is a key capability for deploying this RA in a Software-as-a-Service (SaaS) offering. It is also used on-premise by some organizations to support different divisions or departments within a group.

WRITTEN BY

Paul Fremantle

CTO, **WSO2**

IoT PLATFORM MIDDLEWARE

WSO2 Platform BY WSO2

WSO2 provides a modular, open source platform for device management and analytics, based on a single consistent codebase.

TOP USE CASES

1. Environmental
2. Transportation
3. Medical
4. Smart Service Sector/Retail
5. Smart Supply Chain/Manufacturing

STRENGTHS

1. Create and monitor managed APIs that interact with IoT devices
2. MQTT support: lightweight M2M communications integrated into ESB and APIs
3. Real-time event processing up to 1 million events per second on commodity hardware
4. Device Management for Raspberry Pi, Linux, Android, and portable to other platforms

NOTABLE CUSTOMERS

- Trimble
- T-Systems Connected Car
- Pacific Controls

LANGUAGES

JAVA

JAVASCRIPT

MESSAGING PROTOCOLS

- MQTT
- AMQP
- XMPP

HOSTING OPTIONS

HOSTED OR ON-PREMISE

OPEN SOURCE

YES

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Analytics
- Web Services

FULL PROFILE LINK dzone.com/r/**uVhy**

WEBSITE wso2.com

TWITTER @wso2

THE IoT PROTOCOL WARS

BY MATT BUTCHER

Transformational technologies are inevitably the battleground for competing implementations and protocols, and the Internet of Things (IoT) is no exception.

However, the evolutionary trajectory of two different markets makes this battle an interesting and unpredictable one. IoT is the nexus of Internet computing (web technologies, tablets, smartphones, etc.) and an older breed of consumer electronics such as air conditioners, light switches, and security systems. While these two technology markets are working increasingly well together, there is one big battle that has been taking shape for the last 18 months, and it may come to a head this year.

Droves of "smart" devices are hitting the market. They can be controlled from an app, from a remote server, or from other devices. Almost all of the communication for these devices is wireless, which sounds innocuous enough. We've been controlling things without wires for a very long time, but it turns out that wireless communication strategies are shaping up to be the most contentious aspect of IoT development.

THE SIDES

On one side of the field is the familiar face of 802.11, which we all affectionately call WiFi. Like a dashing knight, it is reliable, battle-tested, and continuously improving. We understand WiFi. Our coffee shops and libraries sport little WiFi stickers on their front doors. Its bandwidth facilitates rapidly moves large amounts of data. It's clearly the most visible protocol in our everyday computing environment. Proponents of WiFi view its claim to the IoT throne as a truism. How could it possibly be otherwise?

> WIRELESS COMMUNICATION STRATEGIES ARE SHAPING UP TO BE THE MOST CONTENTIOUS ASPECT OF IoT DEVELOPMENT.

On the other side of the field stands a small band of contenders. They don't have the stature, the mass appeal, or the hefty experience of WiFi. You might condescendingly call them "upstarts." But this rag-tag band has something WiFi doesn't: specialization.

Among the dozen or so contenders against WiFi, there are some notable names. Z-Wave and Zigbee protocols are the most recognizable, and the new Bluetooth Low Energy (BTL or BTLE) protocol is also in this camp. Not to be overlooked, Lutron's RadioRA and the Insteon protocol are also in this camp, carrying the advantage of a different bloodline: the decades-old home automation industry. Devices with these protocols are showing up at the neighborhood big-box consumer electronics and home improvement stores (e.g. Home Depot, Lowes). With their promise of energy efficiency and mesh networking, the devices leveraging these technologies are well-suited to the market.

> IT'S CLEAR FROM MY EXPERIENCE THAT DEVELOPERS ARE MORE COMFORTABLE USING IP-BASED PROTOCOL STACKS THAN SPECIAL-PURPOSE LOW-LEVEL PROTOCOLS.

Are these non-WiFi devices in it for the long run, or will they simply vanish overnight? Will one protocol really rule them all? And if so, will it be WiFi? While it is too early to predict, there are some potential weaknesses in WiFi's armor. First, let's look at its strengths.

WIFI'S STRENGTHS

Those backing WiFi do so with good reason. WiFi has some clear strengths.

UBIQUITY: The biggest thing WiFi has going for it is its ubiquity. The market that is going to buy into the IoT space overlaps strongly (if not entirely) with the market that already has existing WiFi technologies. The fact of the matter is that we've already got WiFi routers in our homes. The infrastructure is there.

OPEN AND STANDARDIZED: A tremendous advantage for WiFi is the fact that it is an open, unencumbered standard that has been implemented by a variety of chip manufacturers. Healthy competition drives innovation among makers, but WiFi's ubiquity counterbalances this by enforcing de facto compatibility.

DEVELOPER FRIENDLY: It's clear from my experience that developers are more comfortable using IP-based protocol stacks than special-purpose low-level protocols. This is where the Internet ecosystem really influences the Internet of Things. Most developers won't want to spend time on unfamiliar networking technologies if they can just use simple WiFi tools and focus more of their time on other features.

CHINKS IN THE WIFI ARMOR

WiFi's competitors have their own strong points. Because they are specialized, these technologies have definite advantages that general purpose WiFi simply cannot compete with.

MESH NETWORKING: Many of the low-power protocols use some type of mesh networking. In a nutshell, mesh networking refers to a network's ability to relay messages through nodes on the network. For example, a light switch may not only receive messages intended for it, but also pass on messages intended for other devices. This boosts message reliability and also network distance. Some varieties, like Insteon, even support dual-mode mesh networking, where messages can be sent wirelessly and over the power line.

> WHAT HAPPENS IF A WIFI LIGHTBULB IS COMPROMISED? WHAT DATA SUDDENLY BECOMES EXPOSED ON THE WIFI NETWORK?

EFFICIENCY: WiFi is not a low-power solution, nor is its protocol stack lightweight. We know this simply from using our laptops and phones with and without WiFi enabled. This is important to recognize, because many embedded devices have smaller power supplies, less memory, and fewer CPU cycles than a laptop. WiFi simply consumes too many resources for some embedded devices to handle. However, protocols like Z-Wave are designed to run with minimal power consumption and fewer processor and memory resources.

SECURITY: Think about all the data you can access once you're logged into a WiFi network. There are laptops, phones, servers, and media devices that send vast amounts of data over the local WiFi network. Now add some low-power, resource strapped devices that need logins and passwords to use WiFi, but don't have significant security measures. What's the new weakest point on your network? What happens if a WiFi light bulb is compromised? What data suddenly becomes exposed on the WiFi network? A well-publicized problem with Belkin Wemo earlier this year brought this issue to the IoT world's attention.

Contrast that security scenario with a limited, special-purpose protocol like Zigbee. In the absolute worst case, an attacker who manages to get physically close enough to a device and somehow manages to compromise it may have the ability to send messages to other Zigbee devices. In most cases, this wouldn't be as bad as an attacker having access to all the data on the network.

THE COMING SHOWDOWN

When it comes to the networking and protocol layer, what can we expect in the next few years? The final outcome is far from certain, but based on the current market trajectories here are some relatively safe bets.

WIFI FOR ENTERTAINMENT, CAMERAS, AND GATEWAYS: WiFi's biggest foothold is in the media-rich IoT space. Video cameras, streamed media (including voice-interactive technologies), and data intensive apps will likely always be WiFi. Many systems also have Internet-connected hubs or gateways that make it possible to control a device or group of devices from the cloud or from a mobile device. WiFi has no current competitor for these features.

TWO PROTOCOLS ENTER, ONE PROTOCOL LEAVES: Z-Wave and Zigbee are both intensely competing for the home automation space. Often times a market segment is wide enough for competing protocols, but home automation is not one of them. Consumers will quickly tire of trying to determine whether a device's network protocol is compatible with their other stuff. Right now, Zigbee is fragmented with incompatible implementations, but they can (and probably will) fix this situation right away.

STILL ROOM FOR AN OPEN STANDARD: The downside to Lutron, Z-Wave, Zigbee, and Insteon is that all of them are proprietary. Unlike WiFi, which relies upon open standards, these protocols are tightly bound to vendors or consortiums. If one of these protocols becomes the first to break away from this tight coupling, it may quickly emerge as the de facto protocol.

Finally, it is unlikely that any one protocol will ultimately "win out" and be the sole wireless platform for IoT. Already, Belkin's Wemo line is moving toward a combination of WiFi and Zigbee to control their new lighting solution. The reasoning is straightforward: WiFi is an easy bridge to existing home networks, but among the devices, Zigbee is more efficient. Philips Hue is also built this way. Hubs, those ugly little hockey pucks that seem to ship with every IoT solution, are the distasteful side-effect of multi-protocol solutions.

Thus, while multiple protocols will inevitably survive and even flourish, one challenge of the budding IoT space may be clearing out enough closet space for hubs and dongles.

[1] Image of Belkin Wemo: http://www.belkin.com/us/pressreleases/8799962858556/
[2] Image of Philips Hue Tap: http://www.newscenter.philips.com/main/standard/news/press/2014/20140328-introducing-hue-tap-the-worlds-first-kinetic-powered-web-enabled-light-switch.wpd

WRITTEN BY
Matt Butcher

Matt Butcher is the Lead Cloud Engineer at Revolv, a Boulder, CO builder of custom smart home automation solutions that unify next-generation homes with customers' lifestyle devices such as smart phones and tablets. Matt also teaches Computer Science and Philosophy at Loyola University Chicago.

This blog post was written by DZone's in-house research analyst Alec Noller, who actively writes for DZone's IoT Zone. You can stop by DZone's IoT Zone for daily news and articles: http://bit.ly/1ruFmpL

THE IOT SECURITY NIGHTMARE SCENARIO

BY ALEC NOLLER

As the Internet of Things becomes a ubiquitous idea and a fact of life, what happens to all the aging and increasingly insecure Things? According to Wired's Robert Mcmillan, responding to a recent question on the security of IoT from Dan Geer, this may be a problem [1][2]. The solution, Mcmillan suggests, is to design these devices with an expiration date. In other words: they need to be programmed to die.

The problem may not be too severe now, but the future of the Internet of Things will look different than it does now. Security will likely loosen, because software will be a part of everything, and it tends to be the case that things mass produced to that degree experience a bit of a drop in quality. That, Mcmillan argues, presents a problem:

> "...ALL CODE HAS BUGS, AND IN THE COURSE OF TIME, THESE BUGS ARE GOING TO BE FOUND AND THEN EXPLOITED BY A DETERMINED ATTACKER. AS WE BUILD MORE AND MORE DEVICES LIKE THERMOSTATS AND LIGHTBULBS AND SMART TRASHCANS THAT ARE EXPECTED TO LAST MUCH LONGER THAN A PC OR A PHONE, MAYBE WE NEED TO DESIGN THEM TO SIGN OFF AT THE POINT WHERE THEY'RE NO LONGER SUPPORTED WITH SOFTWARE PATCHES. OTHERWISE, WE'RE IN FOR A SECURITY NIGHTMARE."

A similar argument came from Bruce Schneier's interview with Scott Berinato about how future bugs like Heartbleed could impact IoT [3]. Schneier's conclusion is that processes must be built in to IoT devices and development to allow for regular patching and securing of embedded systems. How practical is that, though?

Mcmillan points to some recent scenarios where these fears have already come true: the lack of support for Linksys routers infected with Moon Worm, for example. Long-term patching would solve these issues, but will the increasing number of organizations developing IoT products be forward-thinking enough to care?

It's also not as if the problem will fade as the products become less popular, Mcmillan says:

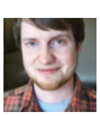
> "RESEARCHERS HAVE STUDIED THE WAY THAT SECURITY VULNERABILITIES ARE DISCOVERED, AND WHAT THEY'VE FOUND IS THAT SECURITY BUGS WILL KEEP CROPPING UP, LONG AFTER MOST SOFTWARE IS RELEASED... IN FACT, THEY'LL ONLY GET WORSE."

Open sourcing technology as it ages may also be a solution, Mcmillan says. However, even that is imperfect and requires a lot of cooperation from companies who may not be enthusiastic about such cooperation, as well as a base of developers interested enough in the technology to maintain it.

So, creating devices with an expiration date may be one of the most practical solutions. Otherwise, what happens when IoT is everywhere? What happens when we stop taking care of the things that we build?

[1] http://www.wired.com/2014/05/iot-death/
[2] http://geer.tinho.net/geer.secot.7v14.txt
[3] http://java.dzone.com/articles/heartbleed-iot-how-much-worse

FIND THIS ARTICLE ONLINE: http://bit.ly/1z45kCY



Alec Noller is the Senior Content Curator at DZone. When he's not creating and curating content for DZone, he spends his time writing and developing Java and Android applications.

FIND MORE AT DZONE'S IoT ZONE:
dzone.com/mz/iot

This blog post was written by DZone's in-house research analyst Benjamin Ball, who actively writes for DZone's IoT Zone. You can stop by DZone's IoT Zone for daily news and articles: http://bit.ly/1ruFmpL

A CRITIQUE OF MQTT FROM A SOFTWARE ARCHITECT

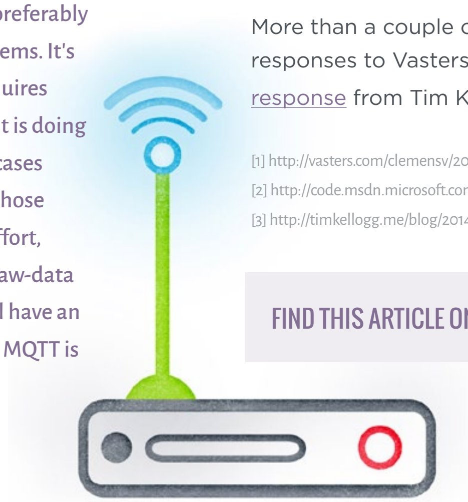
BY BENJAMIN BALL

Most people would say that MQTT has proven to be fairly successful as a lightweight messaging protocol. It is often boasted about for its small code footprint and limited bandwidth usage, which has led to a growing adoption of MQTT from developers creating Internet of Things (IoT) solutions. Next to CoAP, it's probably the most talked about of the messaging protocols in the space right now. However, Clemens Vasters, a Microsoft architect, wasn't so pleased when he sat down to write an implementation of MQTT 3.1.1.

Vasters writes about the topic at great length and he is nothing short of incredibly, incredibly thorough [1]. His lengthy critique tops off at over 9500 words. I don't think anyone can accuse him of not doing his research. He starts off his critique by explaining why he chose not to use Paolo Patierno's existing M2Mqtt client library, which I can only assume would have saved him a lot of time [2]. Among his reasons for passing it up was a difference in server implementation, but also just a desire to better understand the MQTT protocol.

He starts off with a bit of a summary of what he will ultimately conclude many thousands of words later:

> "MQTT is not a messaging protocol; I would call it a funnel protocol serving to move binary frames of data, preferably from constrained clients into data collection systems. It's too limited for actual messaging work, which requires message metadata that a broker can work with. It is doing reasonably well at a very, very narrow set of use-cases and it is terrible at everything that goes beyond those use-cases. What it's reasonably good at is best-effort, raw-data ingestion from clients and best-effort raw-data delivery to clients using a solicit-push pattern (I'll have an explanation later). And as it turns out, the things MQTT is good at can be done in much simpler ways, while retaining more flexibility at the same time."



He also has a few harsh words for IBM on their role in the message queuing information trade, accusing them of trying to gain face by pretending to be an "open-protocol champion" while they keep the MQ wire protocol on lockdown. Vasters is pretty straightforward though that he considers IBM a competitor, and you may have to take his word at face value. It's interesting to note that he really only talks about IBM as an implementor of early MQTT, whereas it's pointed out in the comments of his blog that Eclipse has invested quite a bit in the protocol.

He comes to a number of conclusions, but primary amongst them is that MQTT is just not the protocol that everyone wants it to be:

> "MQTT is an old, recycled, and often weirdly inconsistent mess. It's not a good protocol, and certainly not a good protocol for the Internet of Things where we will look connect devices with long-haul links with unpredictable network conditions, and I believe it's unfixable without becoming something different entirely. We ought to know better, and OASIS also ought to know better."

More than a couple of people have come out with responses to Vasters' critique, including a well-written response from Tim Kellogg at 2lemetry [3].

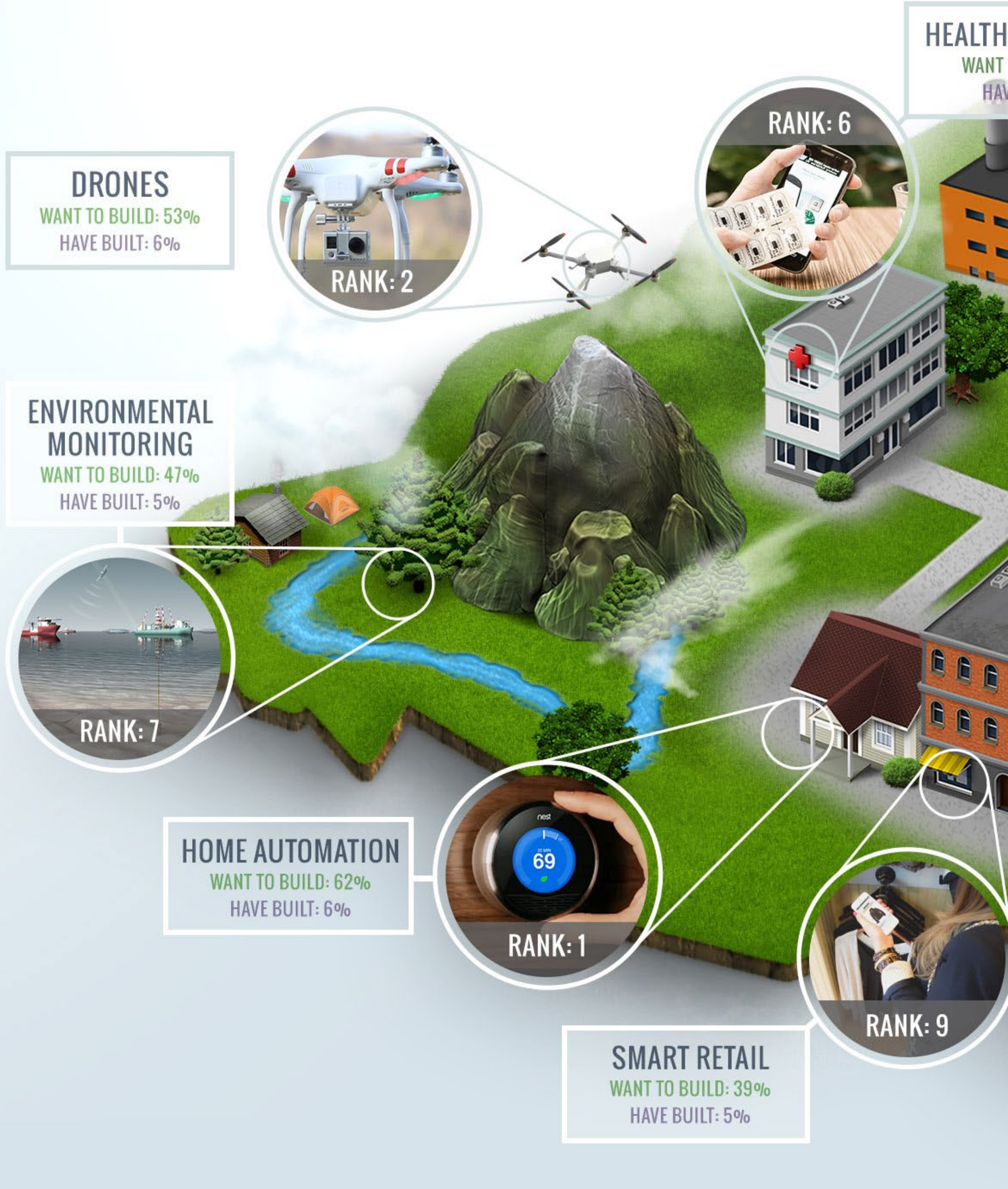
[1] http://vasters.com/clemensv/2014/06/02/MQTT+An+Implementers+Perspective.aspx
[2] http://code.msdn.microsoft.com/windowsdesktop/M2Mqtt-MQTT-client-library-ac6d3858
[3] http://timkellogg.me/blog/2014/06/02/MQTT-another-implementors-perspective/

FIND THIS ARTICLE ONLINE: http://bit.ly/1nCOXLh

Benjamin Ball is a Research Analyst and Content Curator, and leader of the Internet of Things Zone at DZone. When he's not creating and curating content for DZone, he is an avid reader, gadget collector, and live music enthusiast.

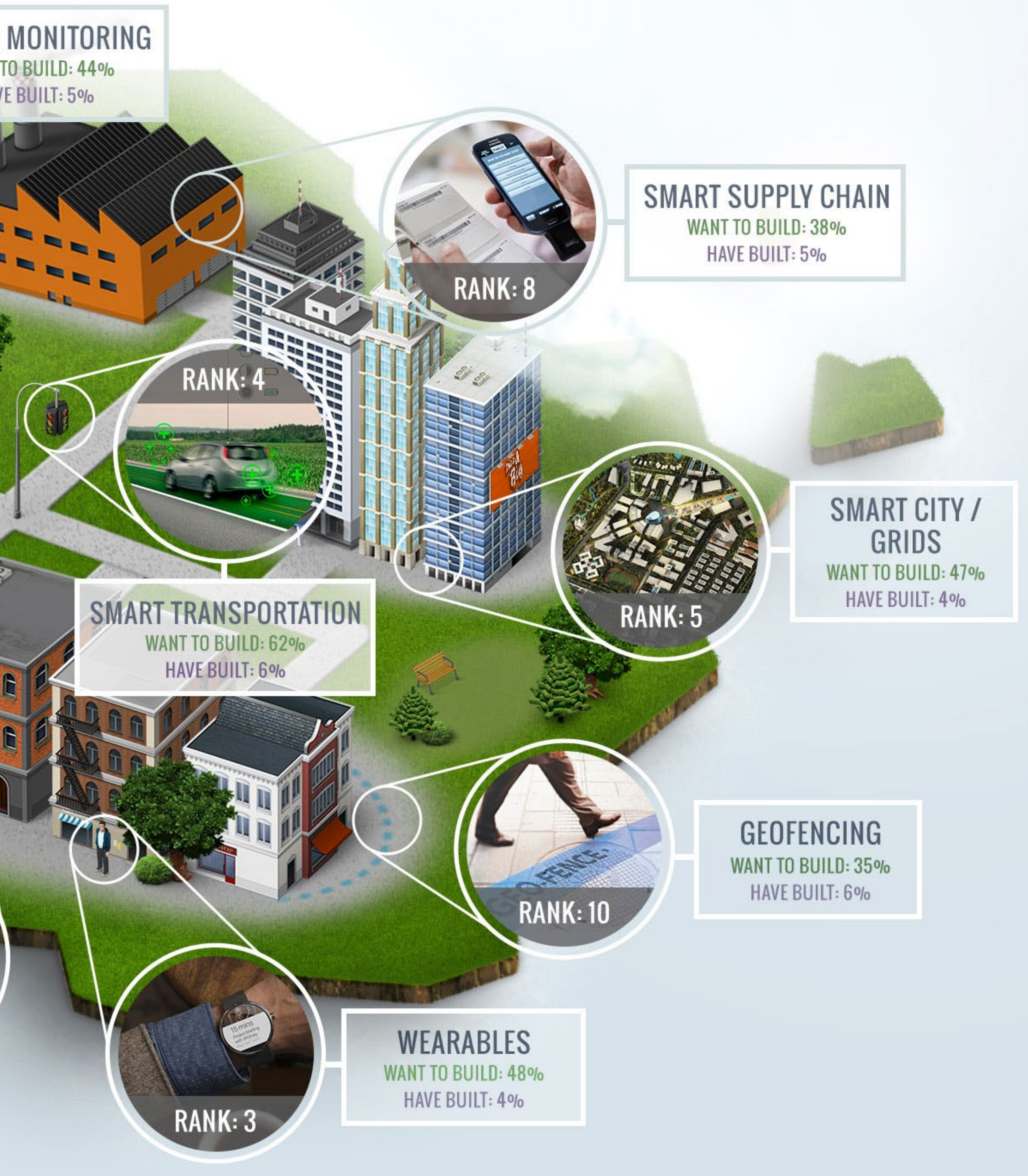
FIND MORE AT DZONE'S IoT ZONE:
dzone.com/mz/iot

IoT DEVELOPER MINDSHA



RE

What are the domains in the Internet of Things landscape? And which ones are developers and IT professionals most interested in? See each IoT domain's interest ranking and the percentage of developers who have actually worked on applications in those domains.



SIX ESSENTIAL INGREDIENTS FOR **BUILDING WEARABLE APPS**

Here at Salesforce, we see the huge potential of wearables to transform the world of applications. In fact, analyst firm IDC suggests that wearable devices will grow by 80% YOY over the next 4 years, with 112 million units shipped in 2018. So we recently released a set of open source starter apps, called the Salesforce Wear Developer Pack, to help developers quickly design and build wearable apps that connect to the Salesforce1 platform. In the process, we learned that there are a number of challenges unique to the wearable context.

- **Security and privacy:** There are security and privacy issues unique to certain wearable devices and scenarios. For example, notifications sent from an Android phone to a Android Wear watch have no built in security layer. Anyone with access to the watch can see the notification and take action (e.g approve/reject a quote discount).
- **Identity:** One of the most challenging parts about developing for any wearable device is figuring out how user identity and authentication works. The way users authenticate with the wearable app is very different across the various devices, and often very challenging to setup.
- **UI/UX:** The old mantra for building mobile apps was that mobile UX/UI does not equal desktop UX/UI. The same can be said for wearables. New UX patterns and best practices must be developed for engaging with these new wearable devices.

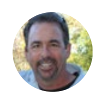
> WE LEARNED THAT THERE ARE A NUMBER OF CHALLENGES UNIQUE TO THE WEARABLE CONTEXT.

- **Data flow:** Some devices like Android Wear and Google Glass support two-way data flow to and from the device. Other devices like the Myo and Nymi only support one-way data flow from the device to the app. The potential use cases/scenarios for a wearable device are therefore dependent on the supported direction(s) of data flow.
- **Testing:** Testing apps for wearable devices is more challenging than for mobile. Emulators/simulators don't exist for some devices while others have emulators, but require tethering to a physical phone.
- **Connectivity:** Some devices like Google Glass can connect to WiFi directly, but most devices currently require to be paired with a phone app via Bluetooth Low Energy (BLE) for internet connectivity. For developers, this necessitates a unique app architecture whereby the phone app is responsible for aggregating multiple API calls and returning a consolidated and minimal data stream (often in JSON format) to the wearable device.

PLATFORM PROLIFERATION

Today, the wearable space is very similar to the early days of smartphones and mobile. There are lots of development platforms with no clear leaders, so developers have to build apps in multiple, diverse programming environments. Developers need a lot of patience, perseverance and trial-and-error to develop apps for these nascent platforms. Despite the technical challenges, wearable development is a ton of fun! What's not to love about that!

WRITTEN BY



Michael Floyd
Editor in Chief, Salesforce Developers, **salesforce.com**

IoT PLATFORM

Salesforce1 Platform BY SALESFORCE.COM

salesforce1 Platform

The Salesforce1 Platform allows developers to build secure, data-driven applications that can be run in the cloud and accessed from web and mobile devices anywhere.

TOP USE CASES

1. Wearables
2. Smart Supply Chain/Manufacturing
3. Medical
4. Transportation
5. Envrionmental

NOTABLE CUSTOMERS

- Nebraska Crossing Outlets
- Philips
- New England Biolabs
- Ventana Medical Systems, Inc.
- EMC Corporation
- ATEK Companies, LLC

LANGUAGES

JAVA
JAVASCRIPT APEX
OBJECTIVE-C RUBY

STRENGTHS

1. The platform is built around a robust and flexible security architecture, providing you with a fine degree of control over the users, network, and data
2. Provides programmatic access to your organization's information using simple, powerful, and secure APIs
3. Combine HTML5, native or hybrid apps with rich device features and your enterprise data to create engaging mobile apps
4. The REST-based Analytics API gives you programmatic access to your report and dashboard data as defined in the report builder and dashboard buildercommodity hardware

MESSAGING PROTOCOLS

- REST
- SOAP

HOSTING OPTIONS

CLOUD HOSTED

OPEN SOURCE

NO

NATIVE CAPABILITIES

- Data Services
- Web Services
- Analytics

FULL PROFILE LINK dzone.com/r/Y3Ww

WEBSITE salesforce.com

TWITTER @salesforcedevs

THE NEAR FUTURE OF IoT

BY SEAN LORENZ

Pundits within the technology sphere have been calling 2014 the year of the Internet of Things (IoT). The market revenue potentials are forecasted into the trillions and it's a Fortune 500 land grab with major companies moving quickly to stake their claims [1]. If this sounds a bit like pages from an American Wild West history book, a frontier analogy isn't too far off. This is an exciting turning point in technology that—thanks to advances in plummeting sensor costs, wireless communication, and chip size reduction—will soon make today's futuristic IoT concepts seem humorous in retrospect.

While it's difficult to see where the market is going, given the exponential rate of change in IoT technology, I have noticed several key trends emerging. As a fellow IoT prospector on the frontier, this is my account of the most evident trends as well as some educated predictions for the future.

TRENDS

1. BUSINESS VALUE OVER TECHNOLOGY FOCUS

Like any promising new technology still in its infancy stage, the true innovation stems from tech-savvy researchers and tinkerers that build fascinating devices that sometimes have no consumer base–I'm looking at you, robotics market. We have all heard about the smart toothbrushes and smart egg trays coming to market and thought: "Interesting! I wouldn't buy one, but... sure!"

Perhaps the biggest trend is a shift from thinking, "let's build it because we can" to "what business problem are we solving here?" IoT developers are getting wise to this mentality and building user-focused MVPs (Minimum Viable Products) that will begin hitting the market in late 2014 and early 2015.

2. KEEPING IT REAL

At my company, Xively, we often get asked what are the real use cases for the IoT. Many times our customers walk in the door with a vague idea of how connecting their product or service to the Internet would be potentially interesting, but need a little help with seeing how an IoT-enabled product can transform their business—internally and externally. The reason for this is that most of the exciting, transformative elements happen under the hood. Right now, the true "wow" moments in the industry are far from sexy: energy savings in enterprise complexes, CRM & ERP integration, service and support, supply chain efficiencies, product part failure and alert, and so on... you get the idea.

Smart homes that respond to our every whim are really great ideas, but these products aren't integral to our lives yet. Large manufacturing companies and enterprises are using the Internet of Things to manage internal operations and efficiency while also engaging their customers more fully with new IoT data sources aggregated in existing services like Salesforce1 or SAP.

3. PUBLISH-SUBSCRIBE

The IoT protocol wars are heating up, but allegiances aside, publish-subscribe messaging is what the bulk of implemented models use for connecting devices to the cloud. Pub-sub protocols such as MQTT, CoAP, and AMQP are attractive for connected product development thanks to their ease of scalability and many-to-one/one-to-many possibilities.

Given the massive variance of the IoT market, there is bound to be more than one protocol that wins in the end; yet before we get to that point, there are plenty of bugs and vulnerabilities to patch across all of the thriving, open IoT protocols out there.

> RIGHT NOW, THE TRUE "WOW" MOMENTS IN THE INDUSTRY ARE FAR FROM SEXY.

4. SECURITY PANIC!

Hacked refrigerators, big box stores, and security cameras... oh my! There has been no shortage of concern for privacy, security, and compliance in the Internet of Things space. Like any news story, some of this attention is warranted and some overblown.

Just like your pre-IoT old-fashioned Internet, creating specific application keys and advanced permissioning systems for hardware connecting to the cloud is essential. The amount of nodes at the edge connecting to services across the Internet will be far larger than anything we see now, but IoT platforms are already addressing these complex device lifecycle management issues that are crucial for protecting personal and enterprise information in a connected world.

NEAR-TERM PREDICTIONS

Now lets hop in the DeLorean and look into the future. Rather than focus on five, ten, or twenty years into the future, let's focus only on the next few years. Why? As I mentioned in the beginning, the IoT landscape changes on a day-to-day basis, so even a prediction looking forward six months from now can be unreliable. This list contains no self-driving cars or sentient AIs. Instead, it makes some pretty sure bets for what to expect over the horizon.

1. A HOUSEHOLD NAME

Usually the second question after "what's your name?" at a dinner party is the inevitable "so what do you do?" Mentioning the Internet of Things to non-techies still draws blank stares and looks of confusion. Those looks are justified given the not-so-great marketing name of IoT and the myriad definitions trying to explain what it actually is. Whether it's called the Internet of Things, Internet of Everything, or just the good ol' Internet, the concept of connecting any and everything to the Internet will begin to make sense for everyday consumers.

2. CONSUMERS SLOW TO ADOPT

Many IoT products are still just toys in many people's minds. Startups are building products that address problems which most consumers don't see as a problem yet. This isn't to say the consumer IoT market will evaporate. It just means we need to get smarter about what customers actually want from smart devices.

Today's wearable products remind me of the Newton—Apple's infamous PDA. The problem wasn't the idea, but rather the timing. The Apple Newton seemed clunky, not very powerful, and low on the usability scale. Years later, the iPhone and iPad came along with a set of features and a form factor that customers were looking for. The same feels true of wearables right now—they may need a few more years to incubate before the general public gives two thumbs up.

Other consumer IoT markets such as the smart home or driverless cars seem to be in the same situation as the wearables market, but this is changing quickly with major players like Apple and Google moving into these arenas. For example, in the home automation space, frameworks like Apple HomeKit will be essential for unifying disparate protocols and clouds into one application that can handle various products' data, automating much of the technology and pushing it into the background. I am sure there is a brilliant developer learning Swift and building the first killer smart home app as we speak.

3. ANALYTICS AND AUTOMATION

This prediction probably comes as no surprise, but it is worth stating. Most companies willing to foray into the IoT unknown are, for now, happy with connecting their devices to an external application or cloud service. Having a place to send the data is usually the first step in constructing an IoT system. So what do you do with all this data once you have it?

Reporting tools for IoT are just starting to become available, but this is just the tip of the iceberg. The real magic lies in the ability to use exploratory and predictive algorithms to make actionable intelligence a reality. These insights are beneficial to both businesses understanding their customers and to the customers themselves.

One could imagine closing the feedback loop between sensor, cloud, and actuator by adding some beautiful supervised machine learning code into the cloud platform at some point in the chain. There are currently a handful of analytics startups focusing on IoT specifically, but this market is about to explode from both platform and application perspectives.

MENTIONING THE INTERNET OF THINGS TO NON-TECHIES STILL DRAWS BLANK STARES AND LOOKS OF CONFUSION.

4. IoT STARTUPS GALORE

For any developers out there interested in the IoT with a real customer pain that needs solving, now is the time to get coding and building that pitch deck. With hardware back en vogue, venture capital funding of IoT-centric companies is on the rise [2]. Having been to a number of IoT events, the amount of enthusiasm by VC and angel investors is palpable. There's a definite need for developers with great, connected product and service ideas; so, if you haven't already, I strongly suggest putting on your favorite prospecting gear and exploring the untamed wild west of the Internet of Things.

HAVING BEEN TO A NUMBER OF IoT EVENTS, THE AMOUNT OF ENTHUSIASM BY VC AND ANGEL INVESTORS IS PALPABLE.

[1] https://internetofeverything.cisco.com/sites/default/files/docs/en/ioe_public_sector_vas_white%20 paper_121913final.pdf
[2] http://www.cbinsights.com/blog/internet-of-things-investing-snapshot
[3] Image of Revolv Hub: https://brandfolder.com/revolv/#!image/

WRITTEN BY
Sean Lorenz

Sean Lorenz is a Technical Product Guy for the Xively IoT platform from LogMeIn. He holds a PhD in computational neuroscience and has worked on projects in Internet of Things, UI/UX, robotics, neural and natural language algorithms, context-aware computing, web and print design, and customer & business development.

HOW TO IoT YOUR LIFE : THE COMPLETE SHOPPING LIST

Whether you're interested in modding and building on a platform for a product, or purchasing a pre-built setup, we've collected everything you could ever want in one place. This exhaustive list of IoT use cases contains all the products you'd need to completely immerse your day-to-day life in the Internet of Things. Use this list to create your own customized shopping list. Click the links to learn more about the products and platforms in each space.

WEARABLES

- ☐ **Eyewear Computing** (Google Glass)
- ☐ **Smartwatch Computing** (Android Wear, Pebble)
- ☐ **Fitness Tracking** (FitBit, Jawbone, Google Fit)
- ☐ **Item Tracking** (TrackR, Kensington Proximo)
- ☐ **Audio Recording** (Kapture)
- ☐ **Stress Sensor** (Neumitra)

HOME AUTOMATION

- ☐ **Smart Outlets** (WeMo)
- ☐ **Smart Devices Orchestration Hub** (Revolv, SmartThings)
- ☐ **Voice Activated Smart Devices Orchestration** (Ivee)
- ☐ **Smart Lights** (Philips Hue, Lutron, LIFX)
- ☐ **Smart Light Controller** (Goldee)
- ☐ **Smart Locks** (Goji, Kwikset, Yale, Lockitron)
- ☐ **Smart Security Video** (DoorBot, Dropcam, Canary)
- ☐ **Smart Security Alarm System** (Scout, Viper)
- ☐ **Wireless Speakers** (Korus, Sonos)
- ☐ **Smart Refrigerator** (Samsung, LG)
- ☐ **Smart Washing Machine** (GE, Maytag)
- ☐ **Smart Climate Control** (Google Nest, Honeywell Lyric, Radiator Labs)
- ☐ **Smart Air Quality Monitor** (Alima, Birdi)
- ☐ **Smart Weather and Indoor Climate Monitoring** (Netatmo)
- ☐ **Smart Water Metering** (LeakSmart, Sprav)
- ☐ **Smart Cooking** (Mellow, ICA Kitchen)
- ☐ **Door Notifications** (Notifon)
- ☐ **Internet Connection and Automation Service** (IFTTT)
- ☐ **Finger Remote Controls** (ThumbTrack, Nod)

TRANSPORTATION

- ☐ **Vehicle App Platforms** (Android Auto, Tesla)
- ☐ **Smart Car Diagnostics** (Dash, Mojio)
- ☐ **iBeacon Kits** (Estimote, RadiusNetworks)
- ☐ **Smart Cycling** (Helios, Vanhawks Valor, Hammerhead One)

LIFESTYLE

- ☐ **Pet Tracking** (Gibi, FitBark, Whistle)
- ☐ **Pet Feeding and Fitness** (Petnet, PetCube)
- ☐ **Sport Performance Tracking** (MiCoach Soccer Ball, Smash Tennis Wristband, Reebok Checklight)
- ☐ **Smart Toys** (Ubooly, Sphero)
- ☐ **Smart Gardening** (Edyn, HarvestGeek)
- ☐ **Smart Sprinkler System** (GreenBox, Iro, Lono)
- ☐ **Smart Plant Sensor** (Parrot, PlantLink)
- ☐ **Entertainment Media Hub** (Blumoo)
- ☐ **Smart Baby Monitor** (Owlet, Withings)
- ☐ **Wireless Sensor Pack** (Node+, Relayr WunderBar)
- ☐ **Connected Toothbrush** (Kolibree)

HEALTH

- ☐ **Health Hygiene and Prescription Management** (IntelligentM, AdhereTech, Lively)
- ☐ **Blood Pressure Monitor** (iHealth, Withings)
- ☐ **Blood Glucose Monitor** (Healbe GoBe, Telcare)
- ☐ **Smart Food Scale** (SITU, BlueAnatomy, Prep Pad, NutriCrystal)
- ☐ **Smart Scale** (Withings, BlueAnatomy)
- ☐ **Smart Sleep Monitor** (Withings, Beddit)

THE SOLUTIONS DIRECTORY

This section of the guide will provide a curated catalogue of the most interesting IoT platforms, hardware, and product lines available for technologists who want to start working on IoT projects right now.

*For IoT hardware products, light-colored tags represent protocols that are supported through add-ons.

To view an extended profile of any product, you can use the short-code link found at the bottom of each profile, or simply go to dzone.com/research/iot and enter in the short code at the end of the link.

MORE INFO dzone.com/r/7fuh

Get easy access to full product profiles with this URL.

IoT PLATFORM

AirVantage M2M Cloud BY SIERRA WIRELESS

The AirVantage platform enables M2M and IoT development with historical data storage, data analytics, customizable dashboards, and centralized device management including 3rd-party ALM tools.

LANGUAGES

JAVA PYTHON JAVASCRIPT RUBY

MESSAGING PROTOCOLS

MQTT AMQP OMA LIGHTWEIGHT M2M HTTP

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Analytics
- Web Services
- Networking Hardware

TOP USE CASES

1. Transportation
2. Home Automation/Smart Home
3. Smart Supply Chain/ Manufacturing
4. Environmental
5. Smart Grid/City/Utilities

HOSTING OPTIONS

Cloud Hosted

WEBSITE sierrawireless.com

TWITTER @AirVantage

MORE INFO dzone.com/r/7fuh

DEVELOPER API

Android Wear BY GOOGLE

Community-driven, open source program to connect existing and new Android apps with a hands-free platform.

PRODUCT TYPE WEARABLES

STRENGTHS

- Notifications on handhelds and smartphones can automatically be synced with wearables
- Apps can be registered to process voice commands
- Send data and actions between handhelds and wearables with data replication APIs and RPCs
- Partner ecosystem includes Samsung, Qualcomm, Broadcom, and Asus

ROADMAP

Android Wear's APIs are Google's initiative to engage existing Android developers to integrate Android smartphone apps with upcoming smart watches from Samsung, LG, Motorola, and others. Google has already made the APIs available after their announcement at Google I/O in June 2014, and has published design guidelines to help developers best work with the new medium.

PROMOTIONS & INCENTIVES

No promotions offered

WEBSITE android.com

TWITTER @android

MORE INFO dzone.com/r/Lf7u

DEVELOPER SDK

Apple HomeKit BY APPLE

HomeKit is a new framework that enables users to discover devices in their home and configure them, or create actions to control those devices and trigger them using Siri.

PRODUCT TYPE HOME AUTOMATION ORCHESTRATION

STRENGTHS

- Integrated feature of iOS 8, to be released in 2014
- Discover accessories and add them to a persistent, cross-device home configuration database
- Display, edit, and act upon the data in the home configuration database
- Communicate with configured accessories and services to get them to perform actions, such as turning on the lights in the living room

ROADMAP

Apple's primary goal with HomeKit is to allow developers to create iOS apps that control appliances and utilities in the home. Partners already include Honeywell, Philips Hue, Texas Instruments, and iHome. There is also a program, called MiFi, for hardware manufacturers who want to make HomeKit-compatible accessories.

PROMOTIONS & INCENTIVES

No promotions offered

WEBSITE apple.com

TWITTER @apple

MORE INFO dzone.com/r/wLdP

HARDWARE: Microcontroller

Arduino Uno BY ARDUINO

Features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter instead of the FTDI USB-to-serial driver chip.

PROTOCOLS: USB ZIGBEE WIFI

LANGUAGES
C C++

BASE PRICE	ADD-ON SUPPORT
$27.00	YES

INCLUDES PROPRIETARY HARDWARE
NO

PROCESSOR:	ATmega328
CPU:	16 MHz
RAM:	2 KB SRAM
STORAGE:	32 KB flash memory
PERIPHERALS:	6 Analog inputs, 1 USB Host, 2 UART, I2C, SPI
GPIO PINS:	14

WEBSITE arduino.cc
TWITTER @arduino
MORE INFO dzone.com/r/Tkv6

IoT PLATFORM

Ayla IoT Platform BY AYLA NETWORKS

Ayla's flexibility and modularity enables rapid changes to any type of device, cloud, and app to take advantage of in-field acquired data.

LANGUAGES
C C++ C# JAVA PYTHON

MESSAGING PROTOCOLS
SMS SMTP HTTPS

NATIVE CAPABILITIES
- Device Management
- M2M Gateway
- Data Services
- Networking Hardware
- Analytics
- Web Services

TOP USE CASES
1. Home Automation/Smart Home
2. Environmental
3. Wearables
4. Smart Service Sector/Retail
5. Smart Grid/City/Utilities

HOSTING OPTIONS
Cloud Hosted

WEBSITE aylanetworks.com
TWITTER @aylanetworks
MORE INFO dzone.com/r/JNfV

HARDWARE: Single-Board Computer (SBC)

BeagleBone Black Rev C BY BEAGLEBOARD

BeagleBone Black is a low cost, low power, high performance prototyping board with several I/O capabilities supported out of the box.

PROTOCOLS: ETHERNET USB BLUETOOTH BLUETOOTH LOW ENERGY WIFI 3G/4G

LANGUAGES
C C++ JAVASCRIPT PYTHON RUBY

BASE PRICE	ADD-ON SUPPORT
$50.00	YES

INCLUDES PROPRIETARY HARDWARE
NO

PROCESSOR:	TI Sitara AM3358 ARM Cortex-A8
CPU:	1000 Mhz
RAM:	512MB DDR3@800MHz
STORAGE:	4GB on-board eMMC, [2 TI PRU (32-bit MCU)]
PERIPHERALS:	1 mini-USB client, 1 standard USB host, 110/100 Mbps Ethernet
GPIO PINS:	65

WEBSITE beagleboard.org
TWITTER @beagleboardorg
MORE INFO dzone.com/r/49ks

DEVELOPER SDK

Belkin WeMo SDK BY BELKIN

Belkin's WeMo system allows users to control and customize connected devices in new ways through their local connection.

PRODUCT TYPE	HOME AUTOMATION ORCHESTRATION
STRENGTHS	• Modify what the app does at the local level • Through integration with rules and IFTTT, room for customization is huge • Allows you to design rules to govern your connected devices from the outlet • The switch can be controlled through any Internet connection on your phone • Devices will be controlled through a user's local connection
ROADMAP	In the near future, Belkin will be releasing a demo app for the SDK on their their official GitHub page. This will give developers the opportunity to experiment with the API before delving into the full kit.
PROMOTIONS & INCENTIVES	No promotions offered

WEBSITE belkin.com
TWITTER @belkin
MORE INFO dzone.com/r/xCYW

IoT PLATFORM

Carriots BY CARRIOTS

Powerful rule engine allows arbitrary code execution with full database access. Event oriented approach, easy REST API, and device management.

LANGUAGES

JAVA PYTHON GROOVY C C#

MESSAGING PROTOCOLS

MQTT HTTP

NATIVE CAPABILITIES

- Device Management
- Data Services
- Analytics
- Web Services
- Integration

TOP USE CASES

1. Smart Grid/City/Utilities
2. Smart Supply Chain/ Manufacturing
3. Transportation
4. Environmental
5. Smart Service Sector/Retail

HOSTING OPTIONS

Hosted or On-Premise

WEBSITE carriots.com

TWITTER @Carriots

MORE INFO dzone.com/r/YCWw

DEVELOPER SDK

Control4 DriverWorks SDK BY CONTROL4

Features a Simple Device Discovery Protocol (SDDP) that is provided in base-level firmware that allows Control4 to know exactly what driver to load to control that device.

PRODUCT TYPE HOME AUTOMATION ORCHESTRATION

STRENGTHS

- Uses the Lua programming language for runtime interpretation of driver commands
- Lua is a standard language that Control4 uses for driver development, rather than a proprietary language
- 3rd-party drivers can be certified, allowing Control4 to test driver functionality
- Automation solution available that allows drivers to work with multiple UI platforms

ROADMAP

Control4 is partnering with manufacturers and trying to grow the ecosystem and build relationships. An automation system needs to provide integration with popular devices as they emerge. Control4 is trying to build their interoperability capabilities, and they need developers to create these solutions and help manufacturers build in interoperability programs.

PROMOTIONS & INCENTIVES

No promotions offered

WEBSITE control4.com

TWITTER @control4

MORE INFO dzone.com/r/97zr

IoT PLATFORM

DeviceHive BY DATAART APPS

An open source platform that encapsulates the complexity of device management, connectivity, and logging through a set of RESTful interfaces.

LANGUAGES

JAVA PYTHON C C++ C#

MESSAGING PROTOCOLS

PROTOCOL PLUG-IN ARCHITECTURE

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Web Services

TOP USE CASES

1. Smart Grid/City/Utilities
2. Wearables
3. Medical
4. Environmental
5. Home Automation/Smart Home

HOSTING OPTIONS

Hosted or On-Premise

WEBSITE devicehive.com

TWITTER @DeviceHive

MORE INFO dzone.com/r/9Tzr

IoT PLATFORM

Dweet.io BY BUG LABS

Dweet.io boasts simplicity and ease of use, and allows users to use any device, API, or language.

LANGUAGES

JAVA PYTHON C C++ C# NODE.JS JAVASCRIPT

MESSAGING PROTOCOLS

HTTPS

NATIVE CAPABILITIES

- Device Management
- Web Services

TOP USE CASES

1. Smart Service Sector/Retail
2. Smart Supply Chain/ Manufacturing
3. Transportation
4. Smart Grid/City/Utilities
5. Environmental

HOSTING OPTIONS

Hosted or On-Premise

WEBSITE buglabs.net

TWITTER @dweet_io

MORE INFO dzone.com/r/Lk7u

IoT PLATFORM

Electric Imp BY ELECTRIC IMP

Electric Imp includes Agents, managed middleware that act on behalf of devices, BlinkUp, an optical setup process, and a low power mode.

LANGUAGES

SQUIRREL

MESSAGING PROTOCOLS

HTTP

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Analytics
- Web Services
- Networking Hardware
- Sensors/Acuators/ Controllers

TOP USE CASES

1. Home Automation/Smart Home
2. Environmental
3. Smart Grid/City/Utilities
4. Smart Service Sector/Retail

HOSTING OPTIONS

Cloud Hosted

WEBSITE electricimp.com

TWITTER @electricimp

MORE INFO dzone.com/r/u3hy

IoT PLATFORM

Etherios Device Cloud BY ETHERIOS

An IoT Platform for today's connected world, including comprehensive device management capabilities and an IoT application enablement framework.

LANGUAGES

JAVA PYTHON C C++

MESSAGING PROTOCOLS

HTTP TCP/IP

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Analytics
- Web Services
- Networking Hardware
- Sensors/Acuators/ Controllers

TOP USE CASES

1. Smart Grid/City/Utilities
2. Smart Supply Chain/ Manufacturing
3. Smart Service Sector/Retail
4. Medical
5. Transportation

HOSTING OPTIONS

Cloud Hosted

WEBSITE etherios.com

TWITTER @Etherios

MORE INFO dzone.com/r/r6Np

IoT PLATFORM

Everywhere Device Cloud BY EUROTECH

An open source and standards-based platform with remote device management, minimal bandwidth usage, REST APIs, and a schemaless database.

LANGUAGES

JAVA

MESSAGING PROTOCOLS

MQTT

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Analytics
- Networking Hardware
- Sensors/Acuators/ Controllers

TOP USE CASES

1. Smart Grid/City/Utilities
2. Smart Supply Chain/ Manufacturing
3. Transportation
4. Environmental
5. Smart Service Sector/Retail

HOSTING OPTIONS

Hosted or On-Premise

WEBSITE eurotech.com

TWITTER @Eurotech

MORE INFO dzone.com/r/yCX9

IoT PLATFORM

EVRYTHNG Engine BY EVRYTHNG

EVRYTHNG offers a robust, scalable, and user-friendly cloud solution to create apps for the Internet of Things.

LANGUAGES

JAVA PYTHON C C++ C#

MESSAGING PROTOCOLS

MQTT WEBSOCKETS

NATIVE CAPABILITIES

- Device Management
- Data Services
- Analytics
- Web Services

TOP USE CASES

1. Smart Service Sector/Retail
2. Smart Supply Chain/ Manufacturing
3. Wearables
4. Medical
5. Transportation

HOSTING OPTIONS

Cloud Hosted

WEBSITE evrythng.com

TWITTER @evrythng

MORE INFO dzone.com/r/z7rN

DEVELOPER API

Fitbit BY FITBIT

The Fitbit API is a set of protocols that can be used to read and write data for a user's tracker collections, profile data, social resources, fetch status of devices and statistical data.

PRODUCT TYPE: WEARABLES

STRENGTHS

- Features integrations with partners like Google Health, About.me, and Microsoft HealthVault
- Supports several languages including Java, PHP, and .NET
- Comes packaged with several client libraries

ROADMAP

Fitbit's primary goal is to let developers create more integrations with apps, websites, or devices they use with their Fitbit devices. They are also relying on their community to offer feedback regarding what features are necessary for the API to grow. Fitbit has also partnered with HTC to preload Fitbit software on HTC devices, and plans to expand and improve its product line to best serve their established market.

PROMOTIONS & INCENTIVES

Partner program available, allowing app developers to make money from promotion and sales of Fitbit trackers

WEBSITE fitbit.com

TWITTER @fitbit

MORE INFO dzone.com/r/73uh

HARDWARE: System on a Chip (SoC)

Freedom K64 BY FREESCALE

A development board compatible with Arduino R3 hardware, including an mbed web IDE, free Kinetis Development System, and MQX RTOS software; including Ethernet, wireless connectivity, and an integrated accelerometer and magnetometer.

PROTOCOLS: ETHERNET USB BLUETOOTH WIFI ZIGBEE

LANGUAGES: C C++

BASE PRICE: $29.00

ADD-ON SUPPORT: YES

INCLUDES PROPRIETARY HARDWARE: NO

PROCESSOR:	MK64FN1M0VLL12 MCU
CPU:	120 MHz
RAM:	256 KB RAM
STORAGE:	1 MB Flash
PERIPHERALS:	1 Micro-USB connector, 1 Micro-USB Power, 1 USB host, 1 10/100 Mbps Ethernet, 2 UART, 2 PWM, SD card, Integrated debugger
GPIO PINS:	43

WEBSITE freescale.com

TWITTER @Freescale

MORE INFO dzone.com/r/RkL7

HARDWARE: Single-Board Computer (SBC)

Intel Galileo BY INTEL

Intel Galileo is a multitasking, multiuser board that runs Linux and is compatible with Arduino hardware.

PROTOCOLS: ETHERNET WIFI 3G/4G ZIGBEE

LANGUAGES: C C++

BASE PRICE: $65.25

ADD-ON SUPPORT: YES

INCLUDES PROPRIETARY HARDWARE: NO

PROCESSOR:	400 mHz i832 Pentium Processor
CPU:	400 Mhz
RAM:	1Gb RAM
STORAGE:	SPI Boot Flash, and Mini-SD card slot up to 32 GB
PERIPHERALS:	1 USB host, 1 USB Client, 1 10/100 Ethernet
GPIO PINS:	22

WEBSITE intel.com

TWITTER @intel

MORE INFO dzone.com/r/ahAt

IoT PLATFORM

Jasper Control Center BY JASPER TECHNOLOGIES, INC.

Jasper is a cloud-based platform that automates the delivery and management of mobile services to connected devices for enterprises worldwide.

LANGUAGES: HTML XML

MESSAGING PROTOCOLS: SMS TCP/IP PROPRIETARY WIRELESS PROTOCOLS

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Analytics
- Web Services
- Networking Hardware
- Sensors/Acuators/ Controllers

TOP USE CASES

1. Transportation
2. Smart Grid/City/Utilities
3. Home Automation
4. Medical

HOSTING OPTIONS

Cloud Hosted

WEBSITE jasper.com

TWITTER @Jasper_IoT

MORE INFO dzone.com/r/6bxY

DEVELOPER API

Jawbone UP BY JAWBONE

The UP platform extends and complements the UP experience by letting users connect their UP with other services they use and love.

PRODUCT TYPE WEARABLES

STRENGTHS

- Authentication supported by OAuth2
- Users have the option to terminate connections with any partners that data is shared with
- Able to track and analyze body events such as mood, workout, and nutrition
- iOS SDK is available to provide an Objective-C interface for integrating iOS apps with the UP API

ROADMAP

Jawbone UP already supports integrations with services like IFTTT, LoseIt!, MyFitnessPal, and Withings, and wants to allow developers the power to create more integrations with other services, and to use their fitness data in new exciting ways. Jawbone has included role-based permissions, and has made its extensive documentation available to best use the API, along with a style guide.

PROMOTIONS & INCENTIVES

No promotions offered

WEBSITE jawbone.com

TWITTER @jawbonedev

MORE INFO dzone.com/r/QW4k

HARDWARE: IoT Prototyping Product

Kinoma Create BY MARVELL

Fully assembled and powered by JavaScript, you'll say "hello world" in 10 minutes, and display sensor values on the screen within 30.

PROTOCOLS: BLUETOOTH WIFI ANALOG UART

LANGUAGES

C C++

BASE PRICE $149.00

ADD-ON SUPPORT YES

INCLUDES PROPRIETARY HARDWARE YES

PROCESSOR:	800 mHz ARM Marvell Aspen SoC
CPU:	800 Mhz
RAM:	128 MB
STORAGE:	16 MB SPI onboard flash, microSD slot
PERIPHERALS:	1 USB 2.0, including USB OTG (On-The-Go), 1 Microphone input, Analog inputs, PWM
GPIO PINS:	36

WEBSITE kinoma.com/create/

TWITTER @kinoma

MORE INFO dzone.com/r/AytU

IoT PLATFORM

Mojio BY MOJIO

Mojio offers an open standards-based platform and API to write applications for cars with rich developer tools.

LANGUAGES

C# JAVASCRIPT PHP

MESSAGING PROTOCOLS

WEBSOCKETS

NATIVE CAPABILITIES

- Device Management
- Data Services
- Analytics
- Web Services

TOP USE CASES

1. Transportation
2. Smart Supply Chain/ Manufacturing
3. Environmental
4. Home Automation/Smart Home

HOSTING OPTIONS

Cloud Hosted

WEBSITE moj.io

TWITTER @getmojio

MORE INFO dzone.com/r/d6PM

MIDDLEWARE

MuleSoft Anypoint Platform BY MULESOFT

The Anypoint Platform enables businesses to connect to applications, data sources, APIs, or devices, whether in the cloud or on-premises.

LANGUAGES

JAVA PYTHON JAVASCRIPT

MESSAGING PROTOCOLS

MQTT AMQP WEBSOCKETS HTTP TCP

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Analytics
- Web Services

TOP USE CASES

1. Smart Grid/City/Utilities
2. Smart Supply Chain/ Manufacturing
3. Environmental
4. Medical
5. Transportation

HOSTING OPTIONS

Hosted or On-Premise

WEBSITE mulesoft.com

TWITTER @mulesoft

MORE INFO dzone.com/r/C4KT

DEVELOPER API

Nest Developer Program BY GOOGLE

The Nest API gives users access to Home and Away states, smoke and CO alerts, and peak energy rush hour events in order to build more integrations and home automation tools.

PRODUCT TYPE SMART CLIMATE CONTROL

STRENGTHS

- OAuth 2.0 and SSL security
- Testing tools available through the program
- Only a limited amount of data is shared, and allows users to revoke access of any integrations they choose
- Partners include Mercedes-Benz, Whirlpool, Jawbone, and Logitech

ROADMAP

With the Google acquisition, Nest is becoming the control center for many smart home devices that are quickly joining the "Works with Nest" program. Nest plans to expand their program beyond their smart thermostat in order to interact with even more consumer devices and to build platforms that automatically anticipate homeowner needs.

PROMOTIONS & INCENTIVES

Google ventures and Kleiner Perkins offer investment opportunities

WEBSITE nest.com

TWITTER @nest

MORE INFO dzone.com/r/N6pH

DEVELOPER SDK

Pebble SDK BY PEBBLE

Pebble provides a rich set of APIs to seamlessly integrate into everyday life, as well as a community appstore.

PRODUCT TYPE WEARABLES

STRENGTHS

- UI design framework included
- Bi-directional communication channel between Pebble and apps running on your smartphone
- Option to build apps through SimplyJS, in a web browser, or through the SDK on a desktop

ROADMAP

With the Pebble SDK 2.0, the SDK has added integration with phone hardware, such as accelerometers and GPS, and has added persistent storage and data logging capabilities. Pebble is currently looking to expand its reach with game developers to create new, original games for the Pebble app store. Pebble has also introduced a cloud platform for developing apps: CloudPebble.

PROMOTIONS & INCENTIVES

No promotions offered

WEBSITE getpebble.com

TWITTER @PebbleDev

MORE INFO dzone.com/r/fMVb

MIDDLEWARE

ProSyst mBS BY PROSYST

mBS is a middleware stack, based on OSGi specifications that supports many IoT relevant communication and messaging protocols.

LANGUAGES

JAVA

MESSAGING PROTOCOLS

MQTT COAP OMA LIGHTWEIGHT M2M

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Web Services
- Sensors/Acuators/ Controllers

TOP USE CASES

1. Home Automation/Smart Home
2. Smart Grid/City/Utilities
3. Medical
4. Smart Supply Chain/ Manufacturing
5. Transportation

HOSTING OPTIONS

On-Premise

WEBSITE prosyst.com

TWITTER @ProSystSoftware

MORE INFO dzone.com/r/AWtU

HARDWARE: Single-Board Computer (SBC)

Raspberry Pi Model B BY RASPBERRY PI

The Model B is the higher-spec variant of the Raspberry Pi, with 512 MB of RAM, two USB ports, and a 100mb Ethernet port.

PROTOCOLS: ETHERNET 3G/4G WIFI ZIGBEE NFC

LANGUAGES

C C++

BASE PRICE $35.00

ADD-ON SUPPORT YES

INCLUDES PROPRIETARY HARDWARE NO

PROCESSOR:	Broadcom BCM2835 700MHz ARM1176JZFS
CPU:	700 Mhz
RAM:	512 MB RAM
STORAGE:	8 GB Micro SD card
PERIPHERALS:	4 USB hosts, 1 10/100 Ethernet, 1 RPi camera connector, 1 Micro-USB Power, 1 Micro SD card slot
GPIO PINS:	40

WEBSITE raspberrypi.org

TWITTER @Raspberry_Pi

MORE INFO dzone.com/r/3NjG

DEVELOPER SDK

Razer Nabu SDK BY RAZER

The Nabu open SDK enables your app to communicate with the Razer Nabu environment and access Nabu user information.

PRODUCT TYPE WEARABLES

STRENGTHS

- Offers testing and validation tools to submit apps to the app store
- Design principles handbook available to help developers create the best apps for the platform
- Nabu can be controlled through simple hand motions that developers can capitalize on

ROADMAP

Razer's upcoming Nabu wristband prompted over 10,000 developers to sign up for their program within 24 hours of its debut at CES 2014. The wristband is currently in a limited beta, but Razer has made it clear that they will be focusing on fostering a strong developer community to build 3rd-party apps, which is why developer editions are available well before the upcoming general release.

PROMOTIONS & INCENTIVES

$49 developer edition device available for those who register for the program

WEBSITE razerzone.com

TWITTER @Razer

MORE INFO dzone.com/r/GQRL

MIDDLEWARE

RESEARCH PARTNER

Red Hat JBoss A-MQ BY RED HAT

JBoss A-MQ is extensible, and can either be deployed alone or with embeddable deployments while supporting multiple protocols and platforms.

LANGUAGES

JAVA C C++ JAVASCRIPT .NET

MESSAGING PROTOCOLS

MQTT AMQP STOMP OPENWIRE WEBSOCKETS

NATIVE CAPABILITIES

- Messaging
- Web Services
- Analytics

TOP USE CASES

1. Transportation
2. Smart Service Sector/Retail
3. Smart Grid/City/Utilities
4. Smart Supply Chain/ Manufacturing
5. Medical

HOSTING OPTIONS

On-Premise

WEBSITE redhat.com

TWITTER @RedHatNews

MORE INFO dzone.com/r/UPJf

IoT PLATFORM

RESEARCH PARTNER

Salesforce1 Platform BY SALESFORCE.COM

The Salesforce1 Platform allows developers to build secure, data-driven applications that can be run in the cloud and accessed from web and mobile devices anywhere.

LANGUAGES

JAVA JAVASCRIPT APEX OBJECTIVE-C RUBY

MESSAGING PROTOCOLS

REST SOAP

NATIVE CAPABILITIES

- Data Services
- Web Services
- Analytics

TOP USE CASES

1. Wearables
2. Smart Supply Chain/ Manufacturing
3. Medical
4. Transportation
5. Envrionmental

HOSTING OPTIONS

Cloud Hosted

WEBSITE salesforce.com

TWITTER @salesforcedevs

MORE INFO dzone.com/r/Y3Ww

DEVELOPER IDE

SmartThings IDE BY SMARTTHINGS

SmartThings Developers can write their own SmartApps and create new types of connected devices.

PRODUCT TYPE HOME AUTOMATION ORCHESTRATION

STRENGTHS

- Allow users to create and develop new ways to automate their home devices and security
- Active community site for users to learn about integrations and other projects
- Web-based IDE allows users to manage locations, hubs, and devices from a browser

ROADMAP

SmartThings began supporting a "cloud-first" approach to their platform, meaning there is a dependency on running SmartThings devices and apps in the SmartThings cloud. However, moving forward, SmartThings plans on launching the SmartThings Hub, which will allow apps and devices to connect to each other locally, without requiring an Internet connection.

PROMOTIONS & INCENTIVES

No promotions offered

WEBSITE smartthings.com

TWITTER @smartthings

MORE INFO dzone.com/r/pxHQ

HARDWARE: Single-Board Computer (SBC)

Tessel BY TECHNICAL MACHINE

JavaScript-programmable, Node-compatible microcontroller, Wifi built in, plug-and-play add-ons, and uses less power than Raspberry Pi.

PROTOCOLS: WIFI NFC RFID USB

LANGUAGES

C JAVASCRIPT COFFEESCRIPT LUA

BASE PRICE: $99.00

ADD-ON SUPPORT: YES

INCLUDES PROPRIETARY HARDWARE: NO

PROCESSOR:	180mhz ARM Cortex-M3 LPC1830
CPU:	180 mhz
RAM:	32mb SDRAM
STORAGE:	32mb Flash
PERIPHERALS:	1 Micro-USB Power, 4 module ports, JTAG, CC3000 debug pins, 1 External antenna port, Vin headers
GPIO PINS:	18

WEBSITE tessel.io

TWITTER @technicalhumans

MORE INFO dzone.com/r/MyaA

IoT PLATFORM

ThingFabric IoT Platform BY 2LEMETRY

The ThingFabric platform helps companies make sense of captured data by offering actionable intelligence through predictive computational models and a configurable rules engine.

LANGUAGES

JAVA PYTHON C C++ C#

MESSAGING PROTOCOLS

MQTT AMQP XMPP COAP DDS OMA LIGHTWEIGHT M2M

NATIVE CAPABILITIES

- Device Management
- M2M Gateway
- Data Services
- Analytics

TOP USE CASES

1. Home Automation/Smart Home
2. Smart Supply Chain/ Manufacturing
3. Smart Service Sector/Retail
4. Wearables
5. Smart Grid/City/Utilities

HOSTING OPTIONS

Hosted or On-Premise

WEBSITE 2lemetry.com

TWITTER @2lemetry

MORE INFO dzone.com/r/4zks

IoT PLATFORM

Thingworx BY THINGWORX

Thingworx is an IoT application development platform with a drag-and-drop interface and built-in analytics tool.

LANGUAGES

JAVA

MESSAGING PROTOCOLS

MQTT AMQP XMPP COAP DDS WEBSOCKETS

NATIVE CAPABILITIES

- Data Services
- Web Services
- Analytics

TOP USE CASES

1. Smart Supply Chain/ Manufacturing
2. Medical
3. Smart Service Sector/Retail
4. Environmental
5. Transportation

HOSTING OPTIONS

Hosted or On-Premise

WEBSITE thingworx.com

TWITTER @thingworx

MORE INFO dzone.com/r/xGYW

HARDWARE: Single-Board Computer (SBC)

TI TM4C Connected LaunchPad BY TI

Tiva features an on-chip 10/100 Ethernet MAC and PHY, USB 2.0, hibernation module, motion control pulse-width modulation, and a multitude of simultaneous serial connectivity.

PROTOCOLS: ETHERNET USB BLUETOOTH BLUETOOTH LOW ENERGY

LANGUAGES

C C++

BASE PRICE: $19.99

ADD-ON SUPPORT: YES

INCLUDES PROPRIETARY HARDWARE: YES

PROCESSOR:	TM4C1294NCPDTI with 32-bit ARM Cortex M4 & Floating Point Unit
CPU:	120 Mhz
RAM:	256 KB
STORAGE:	1MB On-chip flash and 6KB EEPROM
PERIPHERALS:	1 10/100 Ethernet, 1 USB 2.0 with HS ULPI, 4 Quad SSIs, 2CANs, 8UARTs, and 10 I2C ports
GPIO PINS:	90

WEBSITE ti.com

TWITTER @TXinstruments

MORE INFO dzone.com/r/JdfV

HARDWARE: System on a Chip (SoC)

Waspmote Mote Runner Kit BY LIBELIUM

Developed in partnership with IBM, the Mote Runner is built for developing and researching industrial-level sensor networks. It is sold as a kit with over 20 add-on modules included.

PROTOCOLS: USB RADIORA ZIGBEE ETHERNET I2C PWM

LANGUAGES
C C++

PROCESSOR:	ATmega1281
CPU:	14 MHz
RAM:	8 KB SRAM,
STORAGE:	128 KB Flash Memory
PERIPHERALS:	7 analog inputs, 1 USB host, 1 mini-USB power
GPIO PINS:	18

BASE PRICE
$3,491.00

ADD-ON SUPPORT
YES

INCLUDES PROPRIETARY HARDWARE
YES

WEBSITE libelium.com

TWITTER @libelium

MORE INFO dzone.com/r/vs6x

MIDDLEWARE

RESEARCH PARTNER

WSO2 Platform BY WSO2

WSO2 provides a modular, open source platform for device management and analytics, based on a single consistent codebase.

LANGUAGES
JAVA JAVASCRIPT

MESSAGING PROTOCOLS
MQTT AMQP XMPP

NATIVE CAPABILITIES
- Device Management
- M2M Gateway
- Data Services
- Analytics
- Web Services

TOP USE CASES
1. Environmental
2. Transportation
3. Medical
4. Smart Service Sector/Retail
5. Smart Supply Chain/ Manufacturing

HOSTING OPTIONS
Hosted or On-Premise

WEBSITE wso2.com

TWITTER @wso2

MORE INFO dzone.com/r/uVhy

IoT PLATFORM

Xively BY LOGMEIN

Includes a highly-efficient data communications backbone, trust and permissioning engine, and a data and directory service.

LANGUAGES
JAVA PYTHON C C++ OBJECTIVE-C

MESSAGING PROTOCOLS
MQTT AMQP XMPP COAP

NATIVE CAPABILITIES
- Device Management
- Data Services
- Analytics
- Web Services

TOP USE CASES
1. Smart Supply Chain/ Manufacturing
2. Medical
3. Smart Service Sector/Retail
4. Transporation
5. Environmental

HOSTING OPTIONS
Cloud Hosted

WEBSITE xively.com

TWITTER @XivelyIOT

MORE INFO dzone.com/r/xbYW

IoT PLATFORM

Zatar BY ZEBRA TECHNOLOGIES

Zatar features a standards-based interface including CoAP and HTTP, and is optimized for low-bitrate channels.

LANGUAGES
JAVA JSON PYTHON

MESSAGING PROTOCOLS
COAP OMA LIGHTWEIGHT M2M

NATIVE CAPABILITIES
- Device Management
- M2M Gateway
- Data Services
- Analytics
- Web Services

TOP USE CASES
1. Smart Supply Chain/ Manufacturing
2. Medical
3. Smart Service Sector/Retail
4. Smart Grid/City/Utilities
5. Transportation

HOSTING OPTIONS
Cloud Hosted

WEBSITE zatar.com

TWITTER @Zatar_IoT

MORE INFO dzone.com/r/4dks

GLOSSARY OF TERMS

A

ACTUATOR: A mechanism that performs a physical task based on input from a connected system.

ADVANCED MESSAGE QUEUING PROTOCOL (AMQP): An open application layer protocol for message-oriented middleware with a focus on queuing, routing (P2P, PubSub), security, and reliability.

B

BLUETOOTH LOW ENERGY (BLE): A wireless personal area network (PAN) aimed at devices with reduced power consumption and cost while maintaining a similar communication range to regular Bluetooth.

C

CONSTRAINED APPLICATION PROTOCOL (COAP): An application layer protocol used in resource-constrained devices that allows internet connectivity and remote control.

E

EMBEDDED DEVICE/SYSTEMS: An embedded system is a computer with a dedicated function within a larger mechanical or electrical system; it is embedded as part of a complete device.

ENDPOINT DEVICE: An internet-capable device on a TCP/IP network.

F

FLOW-BASED PROGRAMMING: A type of programming that defines applications as networks of process that exchange data across defined connections by message passing, where the connections are specified externally to the processes.

G

GATEWAY: A data communication device that connects a host network to a remote network.

GEOFENCING: A technology that creates virtual boundaries around a physical area in order to trigger an action on a connected device, usually through a combination of GPS and RFID tags.

H

HOME AUTOMATION: A combination of hardware and software solutions that allow for the control and management of electronics, appliances, and devices within a home.

I

IBEACON: A small network transmitter used to identify, track, and interact with connected systems using Bluetooth low energy. It's an Apple trademark, but it is also available on Android devices.

INTERNET OF THINGS (IOT): A network of objects (such as sensors and actuators) that can capture data autonomously and self-configure intelligently based on physical-world events, allowingthese systems to become active participants in various public, commercial, scientific, and personal processes.

INTERNET PROTOCOL SUITE (TCP/IP): The language a computer uses to access the Internet. It consists of a suite of protocols designed to establish a network of networks to provide a host with access to the Internet.

INTERNET PROTOCOL VERSION 4 (IPV4): An internet layer protocol that provides end-to-end transmission across multiple IP networks, and can utilize 32-bit IP addresses.

INTERNET PROTOCOL VERSION 6 (IPV6): An Internet layer protocol that provides end-to-end transmission across multiple IP networks, and can utilize 128-bit IP addresses.

L

LIGHTWEIGHT PROTOCOL: Lightweight protocol refers to any protocol that has a lesser and leaner payload when being used and transmitted over a network connection.

LONG RANGE COMMUNICATION PROTOCOLS: Used to refer to universal long range radio frequencies for multi-generation wireless standards such as 2G, 3G, 4G, and 4G LTE.

IPV6 OVER LOW-POWER WIRELESS PERSONAL AREA NETWORKS (6LOWPAN): This refers to technology that uses IPv6 for a diverse range of hardware applications, including resource-restricted devices for the Internet of Things.

M

MACHINE-TO-MACHINE (M2M): This refers to a network setup that allows connected devices to communicate freely, usually between a large number of devices; M2M often refers to the use of distributed systems in industrial and manufacturing applications.

MESH NETWORK: A type of network topology in which a device transmits its own data and also serves as a relay for other nodes by providing the most efficient data path through routers.

MICROCONTROLLER: A small computer on a single integrated circuit designed for embedded applications and used in automatically controlled embedded systems.

MESSAGE QUEUING TELEMETRY TRANSPORT (MQTT): A lightweight messaging protocol that runs on TCP/IP protocol. It is designed for communicating with small devices in remote locations with low network bandwidth.

N

NEAR-FIELD COMMUNICATION (NFC): A feature, based on technical standards, that allows devices to establish radio communication with other nearby systems or mobile devices.

P

PERSONAL AREA NETWORK: A network created through the interconnection of informationtechnology devices within the context of a single user.

R

RADIO FREQUENCY IDENTIFICATION (RFID): A technology that incorporates uses electromagnetic coupling and radio frequency to identify objects and persons. It consists of three components:
an antenna, transceiver, and transponder.

S

SENSOR: A device or component that perceives and responds to physical input from the environment.

SENSOR NETWORK: A group of sensors with a communications infrastructure intended to monitor and collect data from multiple locations.

SINGLE-BOARD COMPUTER: A complete computer built on a single circuit board with all the components required of a functional computer.

SYSTEM ON A CHIP: An integrated chip that is comprised of electronic circuits of multiple computer components to create a complete device.

T

TRANSMISSION CONTROL PROTOCOL/ INTERNET PROTOCOL (TCP/IP): A basic client/server model of communication protocol for the Internet and private networks.

W

WEARABLES: Connected devices that can be equipped with different types of sensors and are worn on a person's body. They are meant to monitor, collect, and quantify data about a person's life and environment, and allow them to interface with that data.

WIFI (802.11): is defined as wireless local area network (WLAN) that uses radio waves to provide wireless high-speed Internet and network connections.

Z

ZIGBEE: An open standard for wireless communication designed to use low-power digital radio signals for personal area networks (PAN); it is used to create networks that require a low data transfer rate, energy efficiency, and secure networking.

Z-WAVE: A wireless protocol for home automation that communicates using a low-power radio frequency technology specifically designed for remote control applications.

DELIVER **IoT** FASTER

Electric Cloud: Powering Continuous Delivery.
We help industry leaders deliver release-ready software by automating and accelerating build-test-deploy processes at scale.

9X faster builds

75% faster tests

10X faster development

Electric Cloud
electric-cloud.com/iot