FEABHAS

EMBEDDING SOFTWARE COMPETENCE

IoT: Gold Rush or Wild West?

A perspective of what the IoT means today and the opportunities and challenges it brings for rugby players, fridges, smart cities and more.

Author: Niall Cooling, CEO at Feabhas

IoT: Gold Rush vs Wild West

Introduction

Okay, so here's a question, how many times have you come across the term IoT today?

Without doubt, IoT is certainly this year's buzzword and while many are keen to jump on the bandwagon, not everyone actually understands the wagon they're jumping onboard and whether it'll lead to striking gold.

This white paper is for embedded product managers and anyone looking to learn more about the IoT. It will guide you through the wild west of the IoT, the practical ways to connect devices and how the IoT is being used in different sectors.

IoT, of course, stands for "Internet of Things"; the term most often being accredited to Kevin Ashton while running the Auto-ID center at MIT. The original work was conceptualizing the idea that one day all 'things' will be connected to the Internet and thus have to ability to convey information about themselves and their immediate environment. This wealth of information would then allow society to adapt and change in ways never feasible before – so it's no wonder that everyone is keen to join the 'Gold Rush'.

There are numerous 'visions' of the future with all our Things on the Internet; such as the 'smart' fridge that automatically reorders produce, the 'smart' city where given an accident, all traffic is managed in a seamless, automated fashion. These positional pieces tend to include phrase such as "Imagine…" or "In the future…" and "…revolutionise how we live"; which are all well and fine (in 2020 or 2050) but tend to lack any practicality or relationship to where we are today.

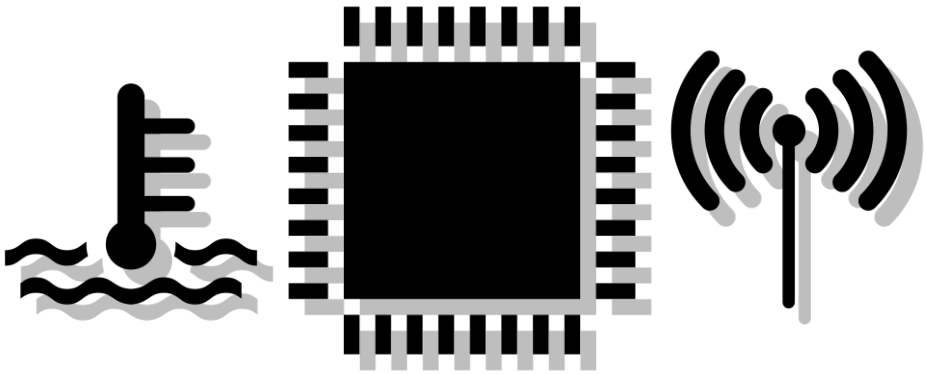
So when companies claim to have the latest IoT solution; is it really IoT or more likely is it, as the old phrase goes, just "Putting lipstick on a pig"? Looking at what comes across my inbox I'd say there is a lot of lipstick being used out there!

The Principles of IoT

Let's start by trying to define the principles of what encompasses IoT, especially as many of the current IoT solutions don't follow Ashton's original conceptual model.

Things

First, what are our Things? In the broadest terms, we're dealing with sensor based systems. If we can 'embed' one or more sensors into an artifact then we typically start to call the artifact 'smart'. To manage the sensors, we can expect a small microcontroller based system. Finally, if we have to convey this sensor information then we need some form of connectivity (e.g. networking).



Sensors

The cost, size and availability of sensors have changed dramatically in the past decade, very much on the back of the introduction of Apple's iPhone and then further smartphones.

Standard in most modern smartphones are sensors that measure:

- acceleration and rotational forces along three axes. This includes accelerometers, gravity sensors, gyroscopes, and rotational vector sensors (Motion sensors)
- environmental parameters, such as ambient air temperature and pressure, illumination, and humidity. This includes barometers, photometers, and thermometers (Environmental sensors)
- the physical position of a device. This includes orientation sensors and magnetometers (Position sensors)

Add to this devices that support GPS, Bluetooth Smart (BLE), Wi-Fi and NFC (Near Field Communication), our Thing can directly benefit from these sensors and devices becoming commoditised. Naturally it opens up an amazing array of opportunities for the data our Thing can collect and report on in an ever-decreasing package size.

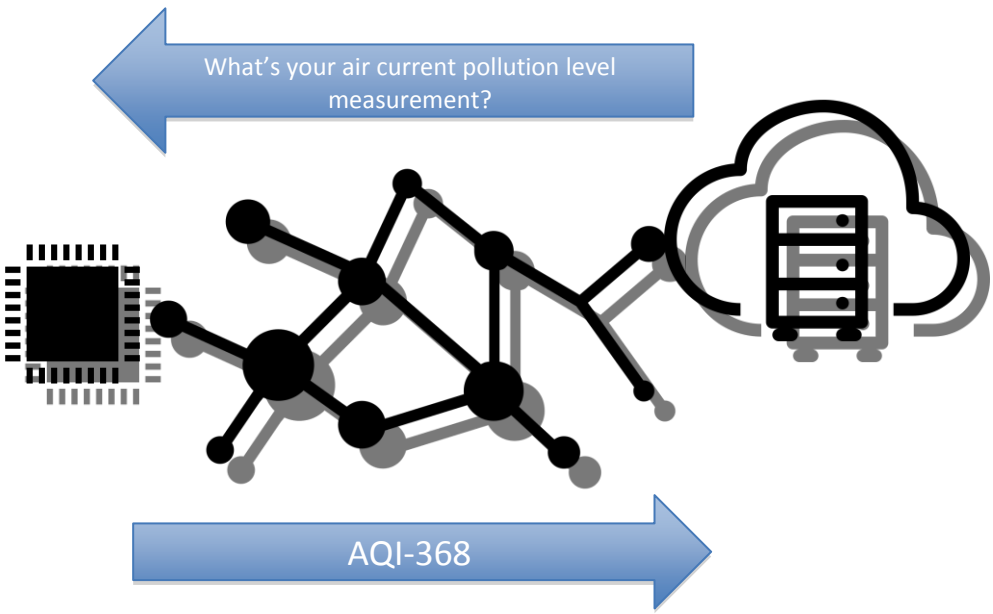
Freescale Semiconductor's Kinetis KL03 processor, shown here nestled inside a dimple of a golf ball. Freescale Semiconductor.

This means the cost model of sensing is changing; suddenly it's becoming cost effective to start sensing data that couldn't have been necessarily justified or even imagined before. Not only because of the decreasing cost of sensors, but also the further development of embedded architectures, such as ARM's Cortex-M family, which offers an incredible price/performance/power solution, enables a significant cost reduction in the overall bill-of-materials and all being efficiently run by a coin-cell battery.

So what's new?

But, I hear you say, haven't we (in embedded systems) been doing this since forever? Yes, the principles of data acquisition are so well established that there is a whole industry based around it; namely SCADA (Supervisory Control And Data Acquisition) systems. In addition, when everyone was getting excited about B2B (Business-to Business), the embedded field jumped on that bandwagon, using M2M (Machine-to-Machine) to describe connected systems. So does that make SCADA and M2M systems automatically IoT? Yes, no, maybe… Think lipstick!

In our conceptual IoT model, our 'smart' devices are directly connected to the Internet. This means, in the same way we query a webpage at a webserver, we (or any system) should be able to query a smart device for its sensor data (i.e. what's your current pollution level measurement at your location?). I'll refer to this as the "pull-model", i.e. we can pull the sensor data from the Thing. The diagram below shows an example for air pollution.



To query a webserver, first it must have an accessible IP (Internet Protocol) address. This IP address is normally hidden from us because we use DNS (Domain Name System) to translate, for example, www.feabhas.com to an IP address (e.g. 178.79.144.243). Second, it

must support the TCP/IP family of protocols (notably TCP and UDP). So, in our 'ideal' model our Things have an IP address.

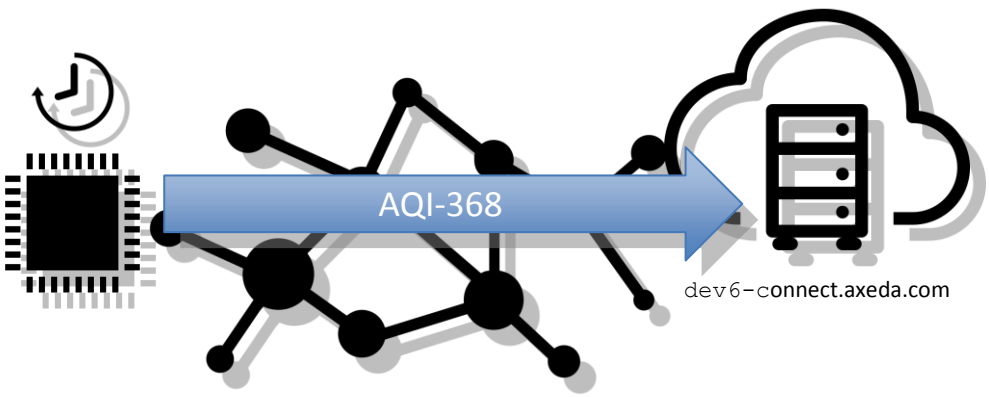
Herein lies our first challenge; the IP address above is based on IPv4 and, unfortunately, there just aren't enough unique IPv4 address to go around (. Forecasts vary wildly of how many 'Things' will be connected to the Internet by 2020, but it's clear it's likely to exceed the 4.3 billion addresses theoretically available using IPv4. There is, however, a very simple solution – IPv6 (Internet Protocol version 6) which, it may surprise you, has been around since 1996 and that all modern operating systems support running IPv4 and IPv6 side-by-side.

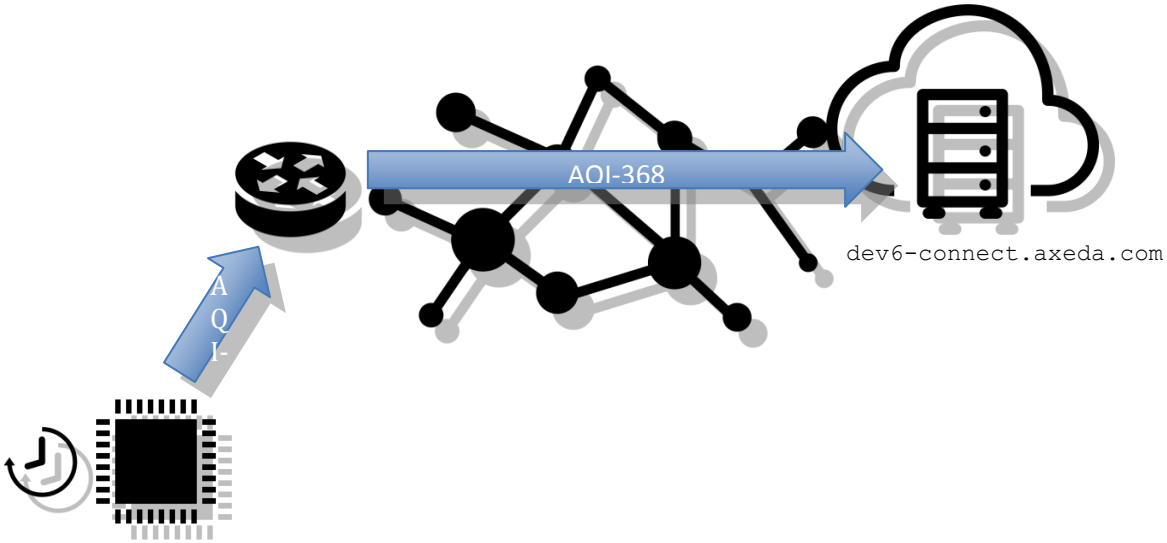
IPv6 has had very slow adoption because, simply, we haven't needed to move away from IPv4. However, the IoT may very well be that 'Tipping Point' to move us over to wider IPv6 usage.

However, today, most 'IoT Solutions' aren't using IPv6 but instead rely on an Internet Gateway. This is typically an Internet connected platform, commonly running embedded Linux, which then communicates with the Things. In one form it's simply acting like your home-router, thus allowing the Things to use internal networking address (e.g. the common 192.168.1.X addressing). Alternatively, many 'IoT Solutions' use a gateway to translate Internet traffic into and from a non-TCP/IP protocol (quite often a proprietary one). In either case, there is one Internet facing IP address mapping onto multiple Things.

One major drawback of the gateway solution is that it makes addressing the Things from the Internet much more complex, as they don't simply have a unique IP address to query. There are approaches that can be taken, however, as an alternative the majority of 'IoT Solutions' use a "push-model" rather than the pull-model described earlier.

In the push-model, the Thing is sending (pushing) its data to the Internet, typically to some form of cloud-service. This approach has a number of advantages; first is that the Thing only needs to know the IP address of the cloud-server and the rest of the world does not need to address the Thing. Once we have the data on the cloud, back end analytics can be applied to ultimately lead to a decision process. Further, existing Internet standards can be used, such as RESTful web services combined with JSON representation of data. Finally, it suits a gateway model where the Thing may communicate to the gateway using, for example, legacy non-TCP/IP protocols.





Hopefully you can appreciate that mixing TCP/IP and non-TCP/IP networks in the push-model makes the IoT landscape more confusing, and this is why there is so much current bewilderment about exactly what IoT is.

However, there is one final, and key aspect of IoT; that it is not just about the collection of sensor data, that data must drive a process of decision making. This is where the "Imagine if…" visionaries look to predict what we could be doing with IoT based information.

We can, therefore, stipulate some key principles of IoT:

1. There is a collection of sensor based system (our smart Thing)
2. Our smart Thing is connected, either directly or indirectly, to the Internet
3. The Thing's data, along with many other Things data, is analyzed and the results applied to make 'valuable' decisions

If we use this set of principles, we can rule out certain solutions as not being in the spirit of IoT (a car's keyfob automatically locking/unlocking the doors based on proximity is not IoT for example).

Decisions, decisions, decisions…

Central, then, to IoT is the collection and analysis of real-world sensor data where upon which 'valuable' decisions can be made.

In terms of today's IoT, there are three distinct areas, each having a quite different viewpoint of the 'value' of the IoT information:

1. Personal / Household
2. Public Sector / Government
3. Private Sector

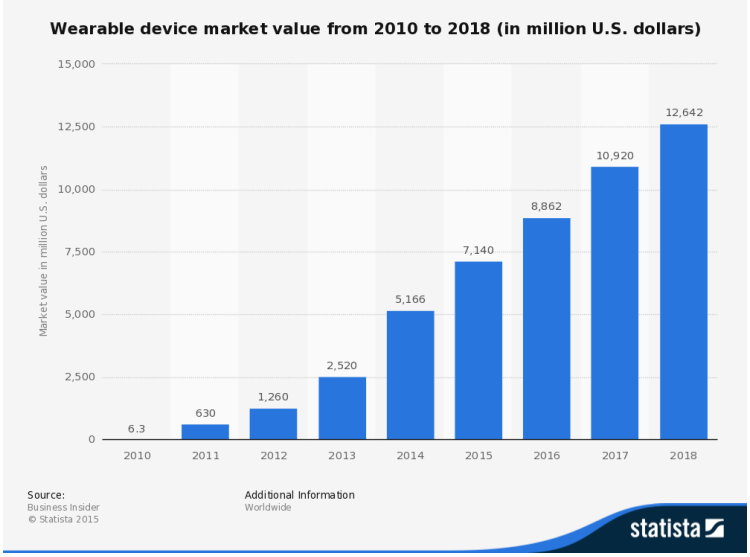
There are, naturally, areas where these overlap; for example, do I want to automatically and seamlessly share my personal fitness data with my doctor or my insurance company? We'll come back to privacy later.

Each of these areas have a different perception the value of the data and the decision making process.

Personal / Household

Wearables / Fitness

The wearable market has exploded recently going from $630M in 2011 to a predicted $7B this year and onwards to over a $12B in 2018.



Technologies have existed in this area for many years (for example Garmin's backed ANT+ networking), but with Bluetooth Smart (also called Bluetooth Low Energy [BLE]) going into smartphones, the phone suddenly becomes the Internet gateway for wearable IoT devices, such as the FitBit and Jawbone UP.

To-date, these have mainly been limited to glorified pedometers, where you can track and analyze, via an app or a website, you daily movement. You can buy a basic pedometer for about £3 ($5), but a basic FitBit is £44 ($70). Is it worth it? With the personal fitness market, the decision making process is very, unsurprisingly, personal, as there is no clear monetary RoI.

The key selling point of the wearable is helping with most people's biggest issue; motivation. As the data (e.g. steps taken) is automatically uploaded (via the smartphone) you can create friendly competitive 'leagues' of, say, other FitBit users so you can see how you're doing compared to your peers or you may choose to make the data available to a personal trainer who can adapt a training programme. There are other nice touches, such as an idle alert (you been sedentary for too long) and sleep monitoring.

Home Automation

Away from fitness, the other area of personal decision-making is around the home. IoT devices hit the headlines when Google paid $3.2 billion for Nest Labs, at the time, a 3 year-old company with 300 employees and makers of a smart thermostat. A basic thermostat costs about £20 whereas the NEST costs £179 (including installation). The decision making process here is different to the wearables, as we can start to see a potential financial RoI. If, through the NEST's "Auto-away" feature I'm going to save £X/year, it will pay for itself in N years.

Home management is nothing new, technology such the X10 has existed for many years (e.g. the ability to turn on and off lights and wall sockets), but it had limited proximity sensor models and tended to be managed by a PC (no Internet here). This is all set to change,

assuming technology such as Apple's HomeKit gain traction, coupled with the new Apple Watch, opens up a potential smorgasbord of home devices to be managed from a smart watch, phone or the Internet.

Public Sector / Government

Decision making

The RoI discussion for Public Sector/Government projects is naturally biased towards efficiency and cost-savings and this is where much of the larger scale IoT work has been focusing.

'Smart City' is a general catchall term and this area has shown some interesting developments. Two clear here-and-now projects based on IoT concepts are the management of street lighting and parking in urban areas.

Street lighting

Street light management is a great poster child for IoT. Combining lighting management (when to turn on/off) with notification of failed lights (making the servicing element far more cost efficient) it is easy to see where this can be of real benefit to any urban area, with claims of up to 70% cost savings on the previous systems (e.g. Intellistreets™ from Illuminating Concepts[1]).

The Intellistreets system is a good example of a gateway model, using proprietary networking between the lamposts aggregated at an Internet gateway. But of course, once you start putting a small computer in a street light we now have the potential for doing much, much more. For example the Intellistreet system can already include digital signage, CCTV, speakers for announcements, panic buttons, etc. (some marketed as 'Homeland Security' features).

City parking

Reports, from companies such as Streetline[2] (a company selling a smart parking system), claim that:

- 20 minutes is the average time spent looking for a parking space globally
- 30% of city traffic consists of people looking for parking
- 60% of drivers have given up on an activity recently due to the difficulty of finding parking
- Parking is usually the second or third largest source of revenue in a city, however, it is exceedingly difficult to manage due to a lack of granular or continuous data on demand and behavior

So, naturally, this is great fodder for IoT RoI case studies. Parking also adds one extra IoT attraction, the ability to use a smartphone App as part of the equation.

Smart city

At the moment, however, many of these systems are IoT silos, the street lighting gives us savings, the parking management shows real RoI (by utilizing the available parking space will naturally lead to higher revenues). For true smart cities we want to see a completely joined up system, including links to other related systems such as transportation and emergency services. This is where the marketing people can start to "Imaging a future…", but we're still some way away from that vision due to one overriding problem, the initial investment costs.

Unfortunately, the RoI for these systems can take many years to repay and given current public funding, without substantial Government investment most regional/city councils cannot the initial investment to get a programme running. This means Smarter cities are likely only going to develop in regions with good government funding, such as parts of the Middle East and South East Asia.

Healthcare

Away from urban street management, the cost of healthcare is still one of the largest government spends each year, so any potential savings here are going to be of great interest and where novel IoT solutions are most likely secure funding.

Using sensors attached to the body, readings can be automatically transmitted to an interested party is classified as Remote Patient Monitoring (RPM). The key benefit of RPM is the ability to send patients home from hospital much earlier than currently. Firstly, this frees up beds and reduces direct costs. In addition, if patients can manage their condition at home, potential costly major medical treatments are less likely to occur. Not surprisingly, the development if 'IoT' systems have received a huge amount of venture capital funding in the last couple of years.

As an aside, the reducing cost of wearables is allowing them to be used as part of public healthcare programmes. For example, studies have shown that overweight people will engage in a wellness/weight loss programme more readily when wearables are incorporated.

Private Sector

Investment decision in IoT for the private sector is much easier as it driven by increased profit. IoT for private sector solutions is where the greatest abuse of the IoT term comes in.

The initial IoT offerings for the private sector have tended to supplement or look to replace existing proprietary systems. For example, companies have for many years implemented various forms of asset tracking using barcode technology. Using NFC replaces the difficulties of traditional barcodes, but in principle is adding nothing new (and is not IoT). If however, we add GPS, networked communication (e.g. 3G connectivity) and sensors (e.g. temperature and humidity) then a different set (potentially proactive) decision can be made. This capability already exists but the cost model has yet to drop to a point where it is truly a commodity and existing ways of tracking assets are sufficient.

For many private sector companies, the ability to have an App to augment their product is enough to class it as IoT, when there isn't any sensing going on, it's just mobile data management (lipstick on a pig). Take, for example, the much-publicized rollout last year by a leading hotel chain, allowing you to unlock your hotel room via their App (using Bluetooth). The major benefit touted is improved security, arguing that traditional locks can be easily bypassed. But this, as before, is just improving an existing system with modern technology; that doesn't make it IoT. However there very well may be new types of data the hotel chain is collecting from the lock allowing it to make differing business decisions; if that is the case, you can argue it is IoT.

There are good examples of where sensor based systems are being used to make far-reaching decisions. For example, professional sport is the leading the way into new and different ways IoT can be used. Major contact sports such as Rugby Union now use

sensor-based systems not only to track players movements around a field (i.e. accelerations, decelerations, different changes of direction, etc.) but in addition, detect collision information. This then enables the coaching staff to adjust training and recovery based on the data analytics in a way they've not been able to do before. The collection of this data is also opening up further research into the key areas such as concussion.

Personal/Private/Public – Smart Meters

One area at the forefront of IoT discussions is the introduction, use and benefits of smart meters. What makes smart metering interesting is that it overlaps all three decision areas:

- Personal – It puts the consumer in control of their energy use
    - a smart meter allows them, for example, to monitor energy usage and receive customized reports, so they can then attempt to adapt their) daily habits to reduce the household's energy bill.
    - Additionally, the automatic reporting of the meter reading means it eliminates that issue of a provider trying to predict usage and the underpay/overpay scenario.
- Private - There are a number of significant benefits for the energy suppliers, including
    - The elimination of manual meter reading, which not only improves accuracy of the bills, but more significantly saves labour costs.
    - Allows the provider to offer more advanced/complex tariffs that can be tuned and customised towards personal household use, encouraging off-peak usage.
    - It is possible to 'remotely disconnect' customers which is not only more cost effective but also overcomes the significant legal barriers when trying to gain physical access to the meter.
    - It helps protects the supplier against 'energy theft' through remote monitoring and analysing usage patterns.

What changed with smart metering was that governments believe that by implementing a smart meter initiative it can meet key carbon emissions targets. For example, the UK Government's goal is that all homes and small businesses will have smart meters by 20203. The savings from the use of smart meters are tied in with the concept of the Smart Grid, where energy provision move away from the existing technologies (Coal, Gas and Nuclear) towards a diverse supply where renewables play a large contribution.

The involvement of the Government suddenly changed the cost model (thus RoI) of smart metering as, at the personal level, it's free and for the utilities they are available subsidies. On a side note, the state of California had the first major smart meter rollout, which has lead to major opposition of their installation as people complain about issues from privacy through to health4.

The Architecture of IoT

In the first section we tried to define the essential model that defines an IoT system. Next we looked at the types of systems that may be considered using IoT technology. In this final section we shall look the wild west of the IoT technology.

So, architecturally, what makes something IoT? There are, of course, myriads of sensors and microcontrollers out there, ranging from the traditional 8-bit microprocessor such as the

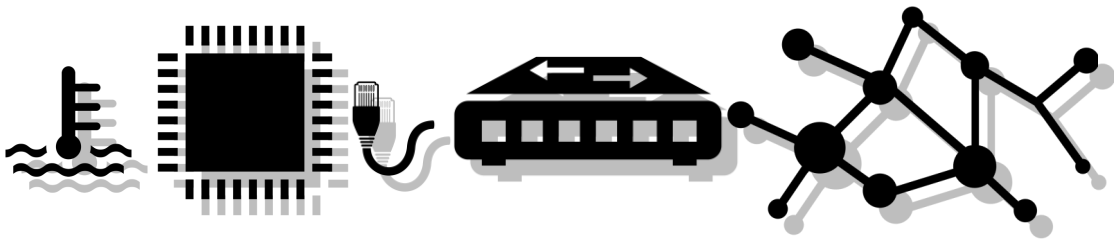
very popular AVR based Arduino systems1 through to what many see as the core enabling technology for IoT; microcontrollers build around the ARM Cortex-M0+ core. These microcontrollers need to combine low cost with good power management.

The first big architectural differentiator is how is the sensor system connects to the Internet; wired or wireless. Certain systems dictate the approach, e.g. a wearable wouldn't be of much use if it has to be tethered in some way, so only wireless options are feasible. But for a static Thing, such as a streetlamp, then there are both wired and wireless options.

Static Things - Wired

With wired IoT, one would naturally look towards the standards used in the home and office. We tend to call this 'Ethernet' and that will suffice for this discussion. This has many benefits:

- The supporting device hardware is relatively cheap (RJ45 connector, etc.)
- It is well established (just plug you IoT device into your router) and can use standard Internet protocols (TCP/IP)
- It tends to be simpler (and more reliable) to configure
- It simplifies the security model as it is part of the physical network



Unfortunately it has some major drawbacks:

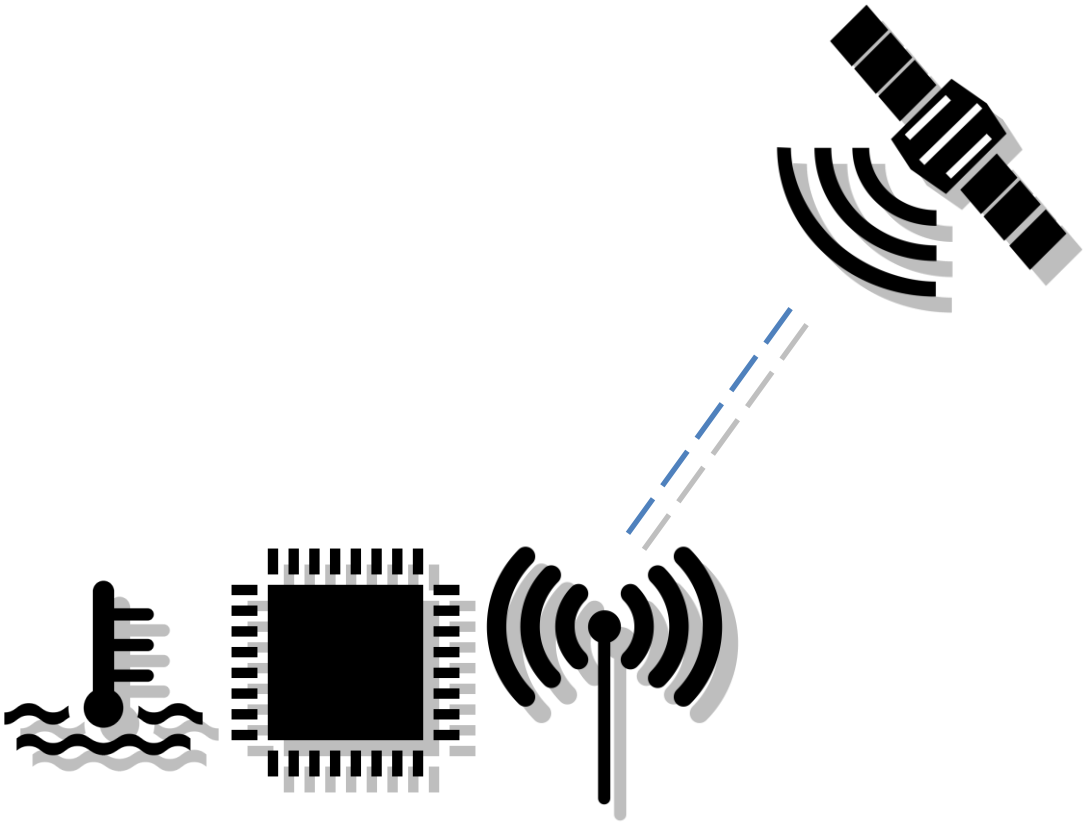
- The cost of installation. Unless cabling already exists this can extend to digging up roads, etc. for cable trunking
- A full TCP/IP stack can require excessive memory and processing needs from the small microprocessor
- A reduced TCP/IP stack may make the device unsecure and open to attacks
- Commercial Ethernet may not be rugged enough, requiring extra costs associated with using Industrial grade components (such as connectors)

For industrial systems, Ethernet has other overheads making unsuitable for many real-time systems typified by SCADA systems. These systems have historically been using other networking systems such as Modbus, Profibus, CANopen, DeviceNet and Fieldbus.

Static Things - Wireless

As typified by the Intellistreets system, even when a Thing is not mobile, it is still may be beneficial to use wireless technology to communicate to the Internet. Ease of installation and reduce maintenance costs make wireless attractive.

[1] Ignoring the Arduino Due that is based on an ARM Cortex-M



We have a number of existing options how we choose for our Thing to communicate with the Internet which depend the remoteness of the location of the device:

- Satellite - if we were monitoring a remote location (e.g. for river pollution) our only option may be to use satellite communication such as the Inmarsat services. Understandably the value of the data must clearly exceed the cost of the satellite communication.
- 2G/3G/4G – As many sensor systems do not generate high volumes of data (compared to video and voice) then technology, such as 2G SMS is sufficient. This option, though, can be problematic due to different network operators and costs across national borders.
- WiMAX (IEE 802.16) / Wi-Fi (IEEE 802.11) / ZigBee (IEEE 802.15.4) – Once we move to urban areas, there is potential to add our own infrastructure. This has the huge benefit of not relying on external network operators. Depending on range, Wi-Fi may be an option, but has relatively short range. Where Wi-Fi can be used, typically ZigBee can as well.
- In building – Wi-Fi (IEEE 802.11) / ZigBee (IEEE 802.15.4) - These are more suitable to shorter distances, such as in-building, but Wi-Fi networks regularly require extenders, whereas mesh networks such as ZigBee can offer benefits here. ZigBee systems will, at some point, require a gateway to the Internet.
- Short range RF (434MHz/915MHz) – This spectrum has historically been used for things such as car remote controls, but due to their low-cost and low-power, gateway based IoT systems do utilise the spectrums.

There are a number of other existing technologies out there and some promising up and coming ones, most notably Weightless5 specifically targeting IoT/M2M communications.

Mobile Things

If our Thing is mobile (a tracked asset), or attached to a mobile object (a wearable) then we can add short-range wireless network, dominated by Bluetooth Smart to the list above. Other short range options exist, most notably ANT+ in the fitness market, but with the integration of BLE into smartphones, BLE is set to dominate.

NFC has existed for many years, used significantly as access token mechanism. However, with the introduction of Apple's Pay touch-to-pay system, NFC may be an option for gateway-based systems. Its major benefit over BLE is cost (so for large scale asset tracking BLE may never be economical), but its downside is proximity (typically somewhere in the 10cm range, whereas BLE can be many meters).

However, where a Thing is collecting data, it must store up sensor readings locally until a gateway unit is in range to then 'dump' the data to the Internet. A good example here is where out Thing uses the smartphone's capability (via Bluetooth) to connect through to the Internet. This is certainly a grey area when defining it as part of the conceptual IoT model, as our Things are not addressable and not part of the Internet.

Wireless Network Stacks

If my Thing uses standard Wi-Fi with TCP/IP and IPv4, getting the sensor data to a cloud-based server is relatively straightforward. Assuming its IP settings can easily be configured, the appropriate security passphrase for the network set up2, then the device just needs somewhere to send the data too in a well-defined format.

Here is where the main battlefield for IoT lies; there are many competing approaches and solutions, mainly involving a centralised cloud-based service for storage and analysis for the data, typified by an offering such as xively6 (there are currently at least 20+ different offerings in this space alone). These approaches look to build on and utilize existing web technology. Sensor data values are typically packaged using the JSON (JavaScript Object Notation) format using RESTful (Representational State Transfer) APIs. Both JSON and REST are well-established technologies in traditional web solutions.

More recently, ARM have collaborated with IBM to offer a 'plug-and-play' IoT starter kit7. The starter kit enables developers to channel data from Internet-connected devices directly into IBM's Bluemix cloud platform using a similar model.

However, as previously mentioned, supporting a full TCP/IP IPv4 stack on a small embedded microcontroller may require too many resources, possibly adding to the overall cost (requiring more memory) and impacting battery life.

The major driver of using prevailing web technology for IoT solutions, built upon TCP/IP IPv4 is that there are both existing standards and accepted de-facto standards. If, for technical reasons, we cannot use IPv4 we are therefore entering non-standard territory, where today,

[2] As most Things don't have displays and keyboards, this is quite often configured using BLE or NFC

many different consortiums (often supported by the same major players) present and champion competing offerings.

There are a number of promising technologies built around IPv6, such as 6LoWPAN (IP6 over Low power over Wireless Personal Area Networks). But this has then lead to further competing efforts to standardize different parts of the end-to-end architecture.

So, currently it is incredibly confusing when starting to discuss what the software of an IoT architecture is. Many key companies are a part of, what appear to be, competing pseudo-standardisation bodies, all vying for their solution to become the IoT solution. In addition, some are trying to address the whole end-to-end solution, whereas others are addressing just one or two layers of the communications stack, but the most significant projects are:

- Thread – backed by Google, ARM and Samsung, is looking to be the alternative to Wi-Fi for IoT networking, and is built upon IPv6 and 6LoWPAN
- AllJoyn8 – Backed by Qualcomm and managed by the AllSeen Alliance is an open-source software framework aimed at making it easy for devices and apps to discover and communicate with each other. Significantly Microsoft recently joined the AllSeen Alliance
- HomeKit – Apples software platform it claims will allow devices, such as locks, lights and thermostats, to be unilaterally controlled from one app
- HyperCat9 – Developed by a group of 40 UK-based companies, including IBM, ARM and BT. The HyperCat specification IoT clients to discover what data an IoT server has available. It is built on the same Web standards that are now common for that interface, i.e. HTTPS, REST, JSON

IoT Security

The security of IoT systems warrants its own complete paper. Needless to say, it ultimately will be what makes or breaks the IoT dream.

First, there is the issue of personal information security; it's probably not a big issue if someone hacked into the Withings website and had access to my current weight automatically uploaded from my Wi-Fi enable smart scales. However, as IoT devices become more widely used in healthcare, the sensitivity of this collected data is naturally very high. Cases, such as the Anthem Data Breach10, raise major concerns about the use of IoT in healthcare11.

Second, there is the issue of physical security. Weak security on an IoT device opens it up for various attacks, from simple Denial-of-Service (DoS) through to complete hijacking. Recent cases have included a report that a German steel mill was targeted. The attack led to an uncontrolled shutdown of a blast furnace, bringing it to an uncontrolled state that led to massive damages12.

Finally, weak security offers new attack vectors to use the device to hop onto a secure network for further exploitation. An example was demonstrated by Nitesh Dhanjani, who hacked into a Philips Hue 'smart' lightbulb installation13. Further, another example was the use of a car's telematics system to gain access to the internal car network to override control14.

Unfortunately, embedded systems have, to a greater-or-lesser extent, ignored many aspects associated with security, relying on either security-through-obscurity (i.e. using proprietary protocols) or being part of an unconnected system. This is all set to change, and at the moment much of the embedded software community is not security savvy.

Summary

With the “Imagine a future…” hat on, we can foresee a future where no industry is left untouched by IoT. For much of this to be realised will require the ongoing cost reduction of the bill-of-materials to build Things and their associated infrastructure.

Unfortunately, at the moment we are still in the ‘Wild West’ when attempting to define an IoT architecture. Natural selection (e.g. Betamax vs. VHS) will hopefully lead us towards an accepted solution, but one concern is that, with technology such as Apple’s HomeKit, we may end up with a number of competing siloed solutions.

Finally, the rush to apply IoT to existing markets is well ahead of our knowledge and understanding of building secure, reliable Things. This could, ultimately, be IoT’s Achilles heel.

About Feabhas

Feabhas improves the competence of embedded software developers through on-site team development, public training for individual engineers, consultancy and mentoring, as well as pre- and post-course assessments.

Feabhas was formed in 1995 and has trained over 15,000 engineers worldwide to date, helping them to improve their embedded software competency and reduce software development times and costs.

As an ARM Approved Training Centre and provider of ARM Accreditation Training, Feabhas is one of two ARM accredited training partners that offers on-site ARM Accredited Engineer (AAE) programmes in Asia, the Americas and Europe.

Feabhas help with developing software standards e.g. DO-178C, ISO 26262, IEC 62304, EN 50128 etc., graduate training program and re-skilling from other disciplines.

Niall Cooling is a Chartered Engineer and CEO at Feabhas, the UK’s leading independent provider of training and consultancy for real-time embedded systems development and software competency.

Niall delivers training and provides consultancy and mentoring to a wide variety of electronics companies ranging from smart metering, industrial control, telecommunications and defence.

His training repertoire includes courses on the ARM Cortex family and Niall was a member of the steering group for the AAE/AAME accreditation programme.

He is also a regular guest speaker at industry conferences and events and has particular interest in establishing a competency framework for Embedded Software Engineering.

References

1 http://www.illuminatingconcepts.com/
2 http://www.streetline.com/
3 https://www.gov.uk/government/policies/helping-households-to-cut-their-energy-bills/supporting-pages/smart-meters
4 http://www.pbs.org/newshour/bb/science-july-dec13-meters_08-27/
5 http://www.weightless.org
6 https://xively.com
7 http://www.arm.com/about/newsroom/arm-connects-a-new-world-of-intelligent-devices-to-the-cloud.php
8 https://allseenalliance.org/
9 http://www.hypercat.io/
10 http://www.latimes.com/business/la-fi-anthem-data-breach-20150224-story.html
11 https://www.ftc.gov/system/files/documents/reports/federal-trade-commission-staff-report-november-2013-workshop-entitled-internet-things-privacy/150127iotrpt.pdf
12 http://catless.ncl.ac.uk/Risks/28.43.html#subj1.1
13 http://www.dhanjani.com/blog/2013/08/hacking-lightbulbs.html
14 http://www.cbsnews.com/news/car-hacked-on-60-minutes/