

What is the Internet of Things?

Michael J Koster
April 29th 2013

So far in these articles, I've discussed in broad terms, and in relative depth, some ideas around architecture for the Internet of Things. This time I want to take up a specific question that helps to frame the differences between various standards and platforms. That question is about the fundamental distinction between Machine to Machine communication, known as M2M, and where I think we are going with the Internet of Things

What is M2M and what is IoT?

Of course I don't have a hard definition, but I'm going to offer a broad contrast, based on what I see to be different, complementary optimization points. Let's start with what we expect to get out of IoT as our context for the discussion. What is it we expect IoT to do for us?

IoT Value Proposition and Value Model

IoT will be a lot of things to a lot of people. Rather than try to list them all, I think there are some common patterns that define the basic value model delivered by the Internet of Things.

First, connecting things to software results in a big gain in the manageability of things and starts to enable the life cycle management of everyday objects. For business this means that capital assets are converted to managed services. The value begins to shift from a scarcity model to a usage model, with huge incremental increase in the return on investment (ROI).

Next, and even more exciting, is the *network effect*. As more and more things are connected and made available as managed services, whole collections of resources can be managed as systems. It's clear that this enables the smart home and smart city convenience scenarios that many use to portray IoT. Beyond that, there is another, more exponential, increase in created value when resources are managed together as a system. This is the network effect applied to the physical world, and what I think generates the ultimate value of the Internet of Things.

To realize the full value of the IoT, then, things need to be connected to software, and they also need to be made available to be used together as a system. This is an important distinction that helps frame the discussion of the original question around M2M vs. IoT.

M2M enables connection of objects to software

Machine to Machine communication has been used for decades, in experimental form since the days of Tesla, and in practice since teletype. Nowadays, M2M connects sensors or connected objects to software services and to each other. Industrial automation has been handled by computer software since the 1970s.

M2M standardization has been a goal of industry for most of this time. M2M standards are focused on protocols and message formats, and recently, to some extent, software APIs. They provide for vertical integration, end-to-end, of systems. M2M standards allow for message level connectivity between endpoints in systems.

What's missing from M2M is the high level mechanisms to build out horizontally and connect diverse resources together as a system to create and exploit the network effect. The result is *silos*, which are separately controlled platforms and standards that are incompatible with one another, making it difficult to integrate devices and services from multiple providers.

The Internet of Things connects M2M systems together

M2M communication provides the essential underlying mechanism for IoT. But simply creating one standard for M2M is not sufficient. We think of IoT as adding a layer of discovery, interoperability, and autonomy on top of M2M communication services, using open internet-style standards. The focus is on data models and APIs that enable horizontal integration across different M2M systems. IoT standards enable collections of sensors and other connected objects using different M2M protocols to be managed as larger systems, creating and exploiting the network effect. IoT is built on M2M. An IoT platform connects across silos and discourages separate data models that lead to the creation of silos.

What about MQTT and other M2M Standards? How do they relate to IoT?

MQTT is at its heart a message transport protocol. It moves messages from one endpoint to another, through a message queue. A publish/subscribe interface is provided to associate endpoints. An endpoint having data to send publishes the availability of the data stream, and any interested endpoint registers a subscription. After that any messages or, in the case of sensors, updates are pushed from the publisher to all the subscribers.

The value for IoT is efficient data push to potentially a large number of subscriber endpoints. Also, the MQTT service can itself be responsible for guaranteed delivery in the case of lossy networks. There are some different QOS models you can specify.

MQTT is a standard protocol that can be used to send data messages or updates from one endpoint to another. I have seen XMPP and PubSub also proposed to do this. There are some advantages to MQTT that make it a very good M2M standard.

HTTP and CoAP can also be used for M2M communication, based on RESTful interfaces at the endpoints. Efficient data push is native to CoAP using an extended GET operation, and for small systems and low data bandwidth, can easily be done in HTTP based systems with websockets and webhooks. RESTful interfaces have their own advantages.

RPC based systems based on .NET and SOAP are also being promoted as M2M standards, and getting some adoption. So as I see it, there are many ways to accomplish M2M communication with different strengths and weaknesses.

CoAP is a good example of both M2M and IoT, as it can be run on different sensor net radio stacks, while still providing interoperable APIs and data models across them all.

The open source platform we're building for IoT will be able to use MQTT "under the covers" to update exposed RESTful endpoints. The idea is to provide standard abstractions for the endpoints that can use different underlying M2M systems as needed to adapt to existing platforms or exploit unique advantages.

So, you can build systems with MQTT as the M2M layer. But MQTT is one component of many that will make up the Internet of Things.

Broad Interoperability has other benefits

Lack of interoperability tends to lead to distorted business models that attempt to monetize the basic connectivity and communication channels, creating a "fat middle" ecosystem full of multiple incompatible competing information pipes, and distracting developers from the human touch points where real value is created.

The Interoperability problem also makes it difficult to reuse and repurpose existing sensing capabilities in new applications. Much of the value will be created from repurposing existing data, as in the well-known example of using GE turbine and commercial aircraft sensor data for real-time weather forecasting. Existing sensors need to be made available for use in new systems.

IoT is an Internet

Just as the Internet is made of many platforms, standards, and networks, so is IoT made of many Sensor Nets, M2Ms, Application platforms, user platforms, and so on, creating a community ecosystem.

Meaningful IoT standards will be built in the manner of Internet standards, based on community process and open contribution, drawn from diverse use cases brought by participants from industry, research, the DIY and Maker community, and others.

We already have common internet languages and protocols, including HTTP, CoAP, MQTT, XMPP, and more. There are plenty of M2M options, but there is not yet a well known architecture for interoperability, discovery, and autonomy. This is, by definition, something we will all use and will need to build together.

Building a shared common platform will create a richer ecosystem for everyone, by enabling development of new systems to focus on the value-added human physical touch points of connected objects and the services that manage them. New services based on high value horizontal integration of resources will be enabled.

It will accelerate the deployment of the IoT at scale, leading to near term economic and human social benefits for all.