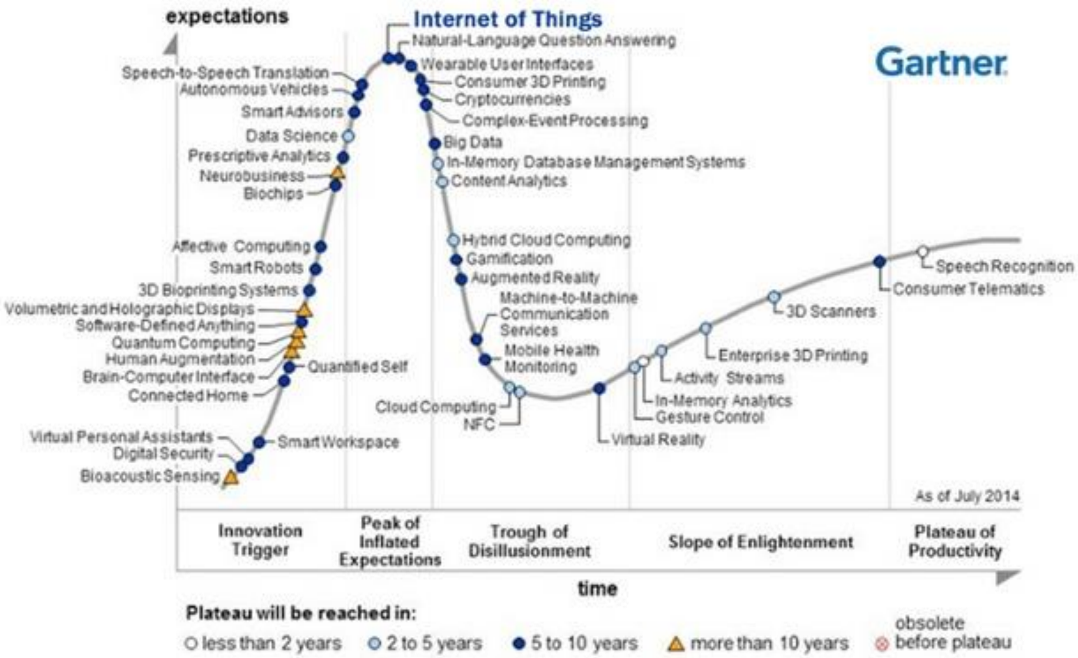
Open Source Strategy and the Internet of Things: a Reality Check

Bill Weinberg | Senior Director and Analyst, Open Source Strategy | The Linux Foundation | August. 5, 2015

Every time I turn around, a company or consortium announces another Internet of Things (IoT) platform. And, while only some of the touted IoT underpinnings are open source, there is a popular consensus that the Internet of Things will only rise if built upon pillars of open source software (OSS).

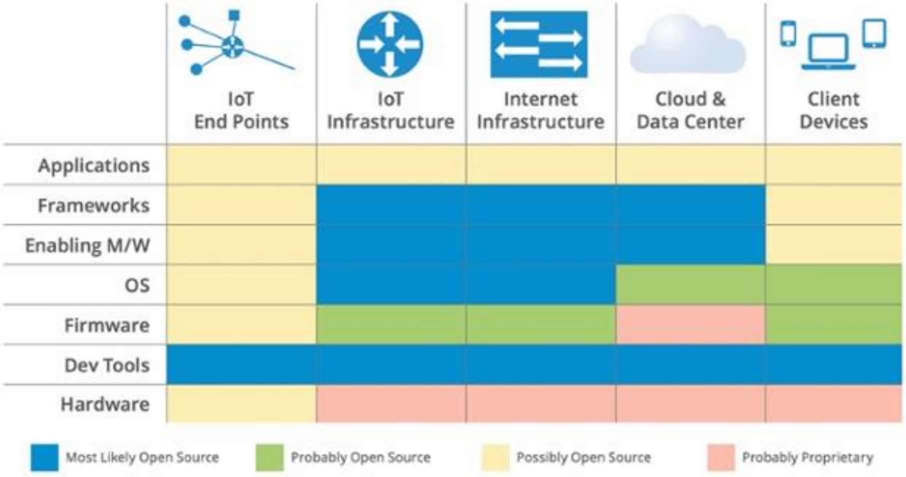
The Gartner 2014 Hype Cycle places the Internet of Things at the very Peak of Inflated Expectations:



While open source is instrumental for building out the IoT, its role is also overhyped. The presence and utility of open source is not universal, nor uniform, across all elements and layers of the emerging IoT.

The purpose of this blog is not to dampen the enthusiasm for open source in IoT, but rather to de-hype the discussion with a reality check.

The following diagram lays out five major IoT "zones" and explores the likelihood of OSS domination over key layers of the software stack for each:



	IoT End Points	IoT Infrastructure	Internet Infrastructure	Cloud & Data Center	Client Devices
Applications	🟨	🟨	🟨	🟨	🟨
Frameworks	🟨	🟦	🟦	🟦	🟨
Enabling M/W	🟨	🟦	🟦	🟦	🟨
OS	🟨	🟦	🟦	🟩	🟩
Firmware	🟨	🟩	🟩	🟥	🟩
Dev Tools	🟦	🟦	🟦	🟦	🟦
Hardware	🟨	🟥	🟥	🟥	🟥

🟦 Most Likely Open Source 🟩 Probably Open Source 🟨 Possibly Open Source 🟥 Probably Proprietary

Let's walk through the zones and peruse the presence of open source.

IoT Endpoints – Simple to Sophisticated

Dumb Devices

Often overlooked in IoT discussions are "dumb devices" – smart labels, inductive slugs, and other RFID devices, used in manufacturing, inventory control, and other areas to track item location. These passive devices report an ID and small amounts of data when energized by scanning equipment. The role of open source here lies not in the RFID tags and slugs themselves, but in equipment that energizes and scans them, and in supporting applications that act upon the data.

Leaf Nodes

One step up are "leaf nodes" of the prototypical Internet of Things – sensors and actuators. Ubiquitous and free-standing, with low power consumption and lower cost, they include light switches and sockets, thermostats and HVAC controls, motion sensors and perimeter alarm switches, and so on. Leaf nodes deploy minimal software, supporting core functionality for sensing or affecting its environment and communicating state or status information upstream. Leaves may deploy embedded OSes or simple main program loops and device service code. Open source in such devices is tactical: developers use OSS tools to create device software, semiconductor suppliers provide OSS drivers and other elements to support them, etc. However, applications running on them will likely remain closed. OEMs today (and in the future) see more value in retaining rights to differentiating technology, in hardware and software, vs. in sharing development and maintenance responsibilities.

Peer-level Endpoints

Peer-level leaf devices serve many of the same functions as simple endpoints, with two key differences:

1. Better provisioning, with 32- or even 64-bit CPUs, additional RAM and storage
2. Bundling routing and gateway functionality into a single package, realizing multiple functions, with potential to deploy enterprise-peer operating systems – Linux, BSD, Windows, etc.

These devices represent more interesting opportunities for open source, from system software up through middleware and applications frameworks, as well as routing software.

IoT Infrastructure

There are really two distinct types of IoT infrastructure

1. Routers, gateways, and aggregators that bridge between the existing Internet, and IoT endpoints
2. Access points, LAN router and edge routers, backbone and core switches, and routers that constitute the Internet.

IoT Edge

At the Edge, the IoT greatly resembles its conceptual predecessor, Machine-to-Machine (M2M) networking. Mission-specific devices transmit context-dependent information across point-to-point or mesh networks, aggregated, buffered, and conditioned by application-specific gateways and routers. In the M2M paradigm, devices communicate over a LAN to computers tasked with control, data analysis, etc. With the IoT, they bridge to the larger Internet and the cloud. Edge nodes provide ample opportunities for OSS deployment and for the evolution of new open source implementations.

Internet Infrastructure

The broader infrastructure of the Internet, from local wireless networks to broadband and mobile baseband access, to edge and core networking, is already teeming with open source software:

- Embedded Linux / Carrier Grade Linux in access points, routers, gateways, firewalls, media gateways, and other networking and telecommunications equipment
- Routing packages, security libraries, network management toolkits, high-availability enablers, and other network-related middleware
- BSDLite-derived TCP/IP stacks paired with proprietary embedded OSes

Embedded web application components and servers used to support configuration and management interfaces

The rollout of SDN (Software Defined Networking) and NFV (Network Functions Virtualization) also provides ample opportunities for open source development to support IoT infrastructure.

The Cloud

As with Internet infrastructure, the cloud is substantially built on OSS components. Linux, virtualization platforms, orchestration and management software, application support libraries and other types of cloud middleware, and the tools and frameworks developers use to write and deploy code – all open source.

Not all cloud software is open (e.g., Microsoft Azure), nor is software that implements Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) readily available as open source. And, while code that implements IoT applications and IoT-centric Software-as-a-Service (SaaS) solutions leverage OSS, there is no impetus for that code to be open source itself.

Android provides an apt analogy. While the platform derives from hundreds of open source components and is itself mostly open, with development tools and libraries published as open source, the majority of the applications distributed through Google Play are closed and proprietary.

End-User Software

End-user IoT software supports monitoring, control, and configuration of IoT devices, and analytics consuming data generated by IoT endpoints. Apps also provide domain-specific functionality relating to the functioning of IoT devices, such as medical diagnosis or soil analysis trending for crop yields.

End-user IoT applications are typically manifested as web applications or mobile apps, but not as open source.

The Open Source Sweet Spot

So yes, open source does have potential to power a vast swatch of IoT zones and node types. Where I believe it has the greatest impact lies in device drivers and enabling middleware. At the edge and in the cloud, IoT aggregators and applications will need to communicate with hundreds of device types and brands today distributed in OEMs branded-siloes. By developing and freely distributing code that lets IoT app developers tip over these silos and abstract away arbitrary differences among comparable devices and across device classes. Look to projects like IoTivity and OpenIoT for the greatest potential for rapid ROI and real-world deployments in 2015.