

IoT: Yes You Can!

How and where to get started in IoT (and why this is such a good idea right now)

Gloria W.

IoT: Yes You Can!

My Background:

26 years of software development, *NIX systems admin, dev-ops, DB design and development, and the list goes on (mostly in *NIX, which is an important part of this talk).

IoT: From buzzword to boom, finally.

Internet of Things: small autonomous devices connected directly to the Internet, or to bigger devices (which are connected to the Internet).

Most are in the form of sensors, which sense movement, measure temperature, record or sense sound or vibration, sense or measure light, take photos and sense changes between images, and much more.

IoT: From buzzword to boom, finally.

List of sensors, by type:

https://en.wikipedia.org/wiki/List_of_sensors

How to know where to start? Follow the protocols:

<https://en.wikipedia.org/wiki/ZigBee>

https://en.wikipedia.org/wiki/Home_automation#Protocols

Communication and Power, the two Barriers to Entry.

Of course, a “drop and go” type of installation is ideal (even if sensor is eventually thrown away).

Wireless and GSM (Global System for Mobile Communications, originally Groupe Spécial Mobile) are preferred.

Battery power is preferred, although at times impossible.

Mesh networking is essential, so that repeaters/amplifiers/controllers are not needed through many parts of an installation.

Often construction and environment come into play: brick, concrete, communication around corners, exposure to elements, electrical or signal interference.

Mesh Networking Who's Who

We'll focus on the ZigBee and Bluetooth Mesh protocols for this discussion, but know that there are more wireless mesh network protocols.

Zigbee Protocol:

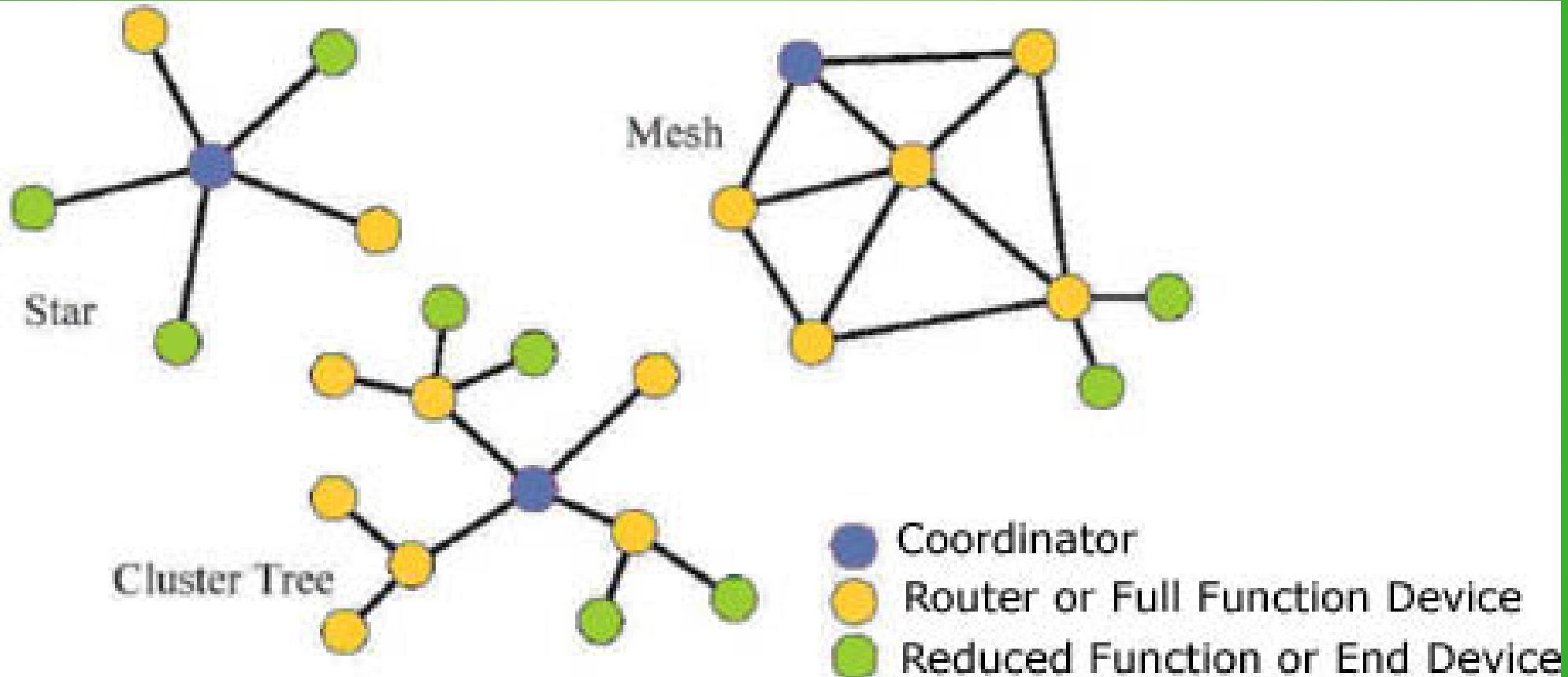
Based on IEEE 802.15.4 spec, with the added functionality of routing, ad-hoc network creation, and self healing mesh functions.

ZigBee is NOT GPL! It's owned and licensed by The ZigBee Alliance. Membership is required for commercial use.

XBee is not ZigBee! XBee is a particular chip set built on the ZigBee protocol, licensed by Digi. There are other chipsets that speak ZigBee:
https://en.wikipedia.org/wiki/Comparison_of_802.15.4_radio_modules

LoRaWAN (Low Range Wide Area Network) is a new Bluetooth protocol owned and licensed by the LoRa alliance, based on an LPWAN standard (Low Power). It allows Bluetooth devices to communicate in a many-to-one fashion. Mesh is still experimental.

Mesh Networking Who's Who



DIY Issues for Businesses

Home automation companies often build their products on top of a well defined commercial protocol. Sometimes they build their own (closed) protocol, such as HA (Home Automation), built on top of ZigBee.

Some products, such as the Smart Things[©] product line, allow developers into a very narrow, limited HA interface. This is no good beyond experimentation.

Implementing your own protocol, using something like a Silicon Labs chip with basic ZigBee support, would require you to add all of the conveniences XBee already does for you re: configuration of networking and sensors, over-the-air updates of sensors, and more. Once done, this implementation would need FCC approval, if it is not built on an already FCC-approved module. These are the barriers to entry for DIY IoT.

But even this is changing rapidly, and should not discourage you from learning and trying out new things.

Open Source Hardware and Software, Protocols and Standards

Open Source is quickly infringing on the corporate stronghold in IoT.

The Raspberry Pi board plays a huge role in acting as a “hub”, which receives data from an attached controller.

It transmits this data over the Internet (usually using the MQTT (MQ Telemetry Transport or Message Queue Telemetry Transport) or AMQP (Advanced Message Queue Protocol). MQTT is a publish/subscribe protocol (two-way), while AMQP can either be a point-to-point or publish/subscribe protocol.

The Pi runs a variant of Debian called Raspbian. Other free and commercial OSes are available for the Pi, but this is the officially supported one.

Competitors to the Raspberry Pi board are appearing like wild right now, from many hardware vendors. Sadly, many try to lock you into closed licensed OS and protocols, and no access to their source code, making you beholden to one company for your IoT solutions. Some run Ubuntu/Debian, and do not restrict access to onboard chips, but this market is still very new.

Open Source Hardware and Software, Protocols and Standards

On the sensor side, the Arduino is a great Open Source option, allowing either C++ or Prototype code to be written which can read data from accelerometers, LiDar and Sonar sensors, and more.

Then there's the Matrix Voice, which does voice recognition, image processing and visual detection, and comes with an onboard ZigBee chip, and FPGA (Field Programmable Gate Array) chip, as well as a bunch of different sensors. Matrix is Open Source, and was crowd-funded and specifically aimed at the hobbyist/enthusiast. It was developed and released into production in 16 months:
<http://www.matrixlabs.ai/labs/hardware.html>

Who is skilled enough to develop products and software for IoT? You!

The good news:

*NIX dev-ops and system admin skills are essential here, and are in high demand in IoT.

Most dev kits that accompany IoT hardware kits give you interfaces in Python and NodeJS. C++ is often not required to get started.

In cases where C++ is required, the C++11 standard is quite advanced, and will look easier to you than you think: standard data structure support, better type casting rules and standards, plus built-in concurrency support make C++ easier to use than in the past.

Go Forth and Play!

XBee sensor kit: \$99:

<https://www.digi.com/products/models/xkb2-z7t-wzm>

Matrix Voice and Computer Vision Kits:

<https://www.indiegogo.com/projects/matrix-voice-open-source-voice-platform-for-all#/>

Raspberry Pi 3B:

<https://www.element14.com/community/community/raspberry-pi/raspberrypi3>

Free Linux online courses and testing:

<https://www.cybrary.it/course/comptia-linux-plus/>