Technology Behind IoT

The following entities provide a diverse technology environment and are examples of technologies, which are involved in IoT.

● Hardware (Arduino Raspberry Pi, Intel Galileo, Intel Edison, ARM mBed, Bosch XDK110, Beagle Bone Black and Wireless SoC)

● Integrated Development Environment (IDE) for developing device software, firmware and APIs

● Protocols [RPL, CoAP, RESTful HTTP, MQTT, XMPP (Extensible Messaging and Presence Protocol)]

● Communication (Powerline Ethernet, RFID, NFC, 6LowPAN, UWB, ZigBee, Bluetooth, WiFi, WiMax, 2G/3G/4G)

● Network backbone (IPv4, IPv6, UDP and 6LowPAN)

● Software (RIOT OS, Contiki OS, Thingsquare Mist firmware, Eclipse IoT)

● Internetwork Cloud Platforms/Data Centre (Sense, ThingWorx, Nimbits, Xively, openHAB, AWS IoT, IBM BlueMix, CISCO IoT, IOx and Fog, EvryThng, Azure, TCS CUP)

● Machine learning algorithms and software. An example of machine-learning software is GROK from Numenta Inc. that uses machine intelligence to analyze the streaming data from clouds and uncover anomalies, has the ability to learn continuously from data and ability to drive action from the output of GROK's data models and perform high level of automation for analyzing streaming data.

The following five entities can be considered for the five levels behind an IoT system

1. Device platform consisting of device hardware and software using a microcontroller (or SoC or custom chip), and software for the device APIs and web applications
2. Connecting and networking (connectivity protocols and circuits) enabling internetworking of devices and physical objects called things and enabling the internet connectivity to remote servers
3. Server and web programming enabling web applications and web services
4. Cloud platform enabling storage, computing prototype and product development platforms
5. Online transactions processing, online analytics processing, data analytics, predictive analytics and knowledge discovery enabling wider applications of an IoT system

SOURCES OF IoT

Examples of hardware sources for IoT prototype development are Arduino Yún, Microduino, Beagle Board and RasWIK. Hardware prototype needs an IDE for developing device software, firmware and APIs.

M2M COMMUNICATION

Machine-to-machine (M2M) refers to the process of communication of a physical object or device at machine with others of the same type, mostly for monitoring but also for control purposes. Each machine in an M2M system embeds a smart device. The device senses the data or status of the machine, and performs the computation and communication functions. A device communicates via wired or wireless systems. The communication protocols are 6LowPAN, LWM2M, MQTT, and XMPP. Each communication device is assigned 48-bits Ipv6 address.

M2M to IoT

IoT technology in industry involves the integration of complex physical machinery M2M communication with the networks of sensors, and uses analytics, machine learning, and knowledge discovery software.

M2M technology closely relates to IoT when the smart devices or machines collect data which is transmitted via the Internet to other devices or machines located remotely. The close difference between M2M and IoT is that M2M must deploy device to device, and carry out the coordination, monitoring, controlling of the devices and communicate without the usage of Internet whereas IoT deploys the internet, server, internet protocols and server or cloud end applications, services or processes.

M2M has many applications in fields such as industrial automation, logistics, smart grid, smart cities, health and defence. Initial applications of M2M were found in automation and instrumentation only, but now these include telemetric applications and Industrial Internet of Things (IIoT) as well.

M2M Architecture

M2M architecture consists of three domains

1. M2M device domain
2. M2M network domain
3. M2M application domain

M2M device communication domain consists of three entities: physical devices, communication interface and gateway. Communication interface is a port or a subsystem, which receives the input from one end and sends the data received to another. M2M network domain consists of M2M server, device identity management, data analytics and data and device management similar to IoT architecture (connect + collect + assemble + analyse) level.

M2M application domain consists of application for services, monitoring, analysis and controlling of devices networks.

Software and Development Tools

Examples of M2M software and development tools are as follows:

- Mango is an open source M2M web-based software. It supports multiple platforms, multiple protocols, databases, meta points, user-defined events and import/export.20
- Mainspring from M2MLabs is a development tool, and source framework for developing M2M applications.21 It enables:
  - Flexible modelling of devices and their configurations
  - Communication between devices and applications
  - Validation and normalisation of data
  - Long-term data storage and data retrieval functions

❍Programming in Java and Apache Cassandra

❍Usages of no SQL database.

●**DeviceHive** is an M2M communication framework. It is an M2M platform and integration tool. It enables connecting devices to the IoT. It includes web-based management software that creates security-rules-based e-networks and monitoring devices. The web software enables prototype projects built with DeviceHub and online tests to find out how it works.

● Following are the open protocols tools and frameworks for M2M:

❍XMPP, MQTT-OASIS standards group and OMA LWM2M-OMA standard group for protocol

❍Various projects of Eclipse M2M industry working groups' are Koneki, Eclipse SCADA for open standards for communication protocols, tools and frameworks

❍ITU-T Focus Group M2M global standardisation initiative for a common M2M service layer

❍3GPPP study group for security aspects of M2M equipment and automatic SIM activation covering remote provisioning and change of subscription.

❍Weightless (wireless communications) group for standards and using wireless spaces for M2M

EXAMPLES OF IoT

Wearable Smart Watch
Smart Home
Smart Cities