1

Introduction to Internet of Things

1.1 Characteristics of IoT

Internet of Things (IoT) can be used to design products for businesses. It facilities to add the valuable feature to the businesses, where the IoT framework is designed to connect the information from devices which are interconnected. The process has been classified into five phases. The first phase is the "create phase," where sensors collect the data from the environment. This data can generate the information for the business. Second is "communicate phase," where data generated in the first phase are communicated to the required destination. Third is the "aggregate phase," where collected data over the network are aggregated by the device itself. Fourth is "analyse phase," where aggregated data are used to generate patterns and use it to control and optimize the process. Fifth is the "act phase," where necessary actions are taken on the basis of information extracted from the aggregated data.

The characteristics of IoT may vary from one domain to other. A few characteristics are listed as follows:

1. **Intelligence:** IoT is treated as smart due to the integration of hardware, software, computational capability, and algorithms. The intelligence feature in an IoT system establishes a great capability of responding in an intelligent way to situations in order to carry out specific tasks. IoT provides a standard input method in the form of a graphical user interface, which also makes it user friendly.
2. **Connectivity:** Connectivity brings objects together over IoT. It is important, as connectivity contributes to the collective intelligence of the system. It enables network accessibility and compatibility of the objects. New opportunities could be generated in the recent market by connecting smart devices by the networking.
3. **Dynamic nature:** IoT is dynamic in nature, as it is capable of collecting data from devices, which may change dynamically, for example, a change in temperature or speed.

4. **Enormous scale:** The number of connected devices over IoT is very large. The data management for such large number of devices is even more critical. But the complexity does not affect the number of objects connected to IoT day by day.
5. **Sensing:** The sensor is the most important part of an IoT network. It detects or measures the environmental changes to generate the data. The sensing technologies provide true information regarding physical quantities in the environment.
6. **Heterogeneity:** IoT devices designed using various hardware frameworks and networks can communicate over different networks. Features like modularity, scalability, extensibility, and interoperability play key design roles in IoT.
7. **Security:** IoT devices are susceptible to cyberattack. There is a high level of transparency and privacy issues with IoT. It is important to secure the end objects, the networks, and the data that are transferred over the network.

There is a wide range of technologies incorporated with IoT to support its successful functioning. IoT creates values and supports human beings to make their lives better.

1.2 Design Principles of IoT

In the near future, everyday lives will be filled with more intelligent devices. Designing IoT devices and networks has challenges to be addressed, which includes connecting different types of physical devices, collecting data, extracting meaningful information, and fulfilling different needs at the level of industries and home.

A few of the design principles for IoT devices and networks are as follows:

1. **Focus on value:** To start with IoT design, it is important to understand the types of features that need to be included. The challenges and barriers need to be understood before adopting new technologies. The designer needs to dig out the user's need and acceptability of the product. The order of features also needs attention in any design process.
2. **Holistic view:** IoT product consists of multiple devices with different capabilities. The solution may also have a cooperation with multiple service providers. It is not enough to design only one end device; the designer needs to take a holistic view across the complete system.

3. **Safety first:** The consequences of avoiding safety in IoT products can be very serious due to direct connection of devices with the real world. Also, building trust must be one of the main drives among designers. Because IoT is a combination of hardware, software, and network, any situation of occurring error needs to addressed. In the event an error situation can't be avoided, the feature of communicating the error to the user may build a trust. It is important to make users' data secure and safe, to build a trust towards IoT.
4. **Consider the context:** IoT solutions directly deal with the real world, where many unexpected things happen at a time when the user should feel safe. IoT solutions should be capable of handling changing environmental situations, such as a change in temperature. Also, an IoT device can have multiple users, unlike a smartphone, so this context needs to be addressed.
5. **Strong brand:** To handle adverse conditions such as device failure, building a strong brand is very important among users. When users feel connected with a brand, they are more forgiving and are more likely to keep the products.
6. **Prototyping:** An IoT solution is a combination of both hardware and software, and both have different life spans. But in IoT, the solution needs to be aligned. IoT hardware and software are hard to upgrade once they are placed at a location. So, the prototyping and its iteration are the solution before actual finalizing the product to launch.
7. **Use data responsibly:** An IoT solution generates tons of data during its life span. However, the idea is not to hold all the data but instead to identify the data points that are required to make the solution functional and useful. So, the possibility of data sciences comes in here. Data science provides a solution to reduce user friction. It can be used to interpret meaningful signals and automate repeated context-dependent decisions.

1.3 IoT Architecture and Protocols

1.3.1 IoT Architecture

IoT architecture is comprised of the following components:

1. **Thing:** IoT is interconnected with various sensors to collect the data and actuators to perform actions corresponding to the commands received from the cloud.
2. **Gateway:** It is used for data filtering, preprocessing, and communicating it to the cloud and vice versa (receiving the commands from the cloud).

3. **Cloud gateway:** It is used to transmit data between the gateways and IoT central servers.
4. **Streaming data processor:** It distributes the data coming from sensors to the relevant devices connected in network.
5. **Data lake:** It is used to store all defined and nondefined data.
6. **Big data warehouse:** It is used for collecting valuable data.
7. **Control application:** It is used to send commands to the actuators.
8. **Machine learning:** It is used to generate models by applying algorithms on data, which can be used to control applications.
9. **User application:** It enables the users to monitor the data and make decisions on controlling connected devices.
10. **Data analytics:** It is used for manual data processing.

1.3.2 IoT Protocols

1.3.2.1 OSI Model

The OSI (Open Systems Interconnection) model for IoT protocols, as shown in Figure 1.1, includes five layers: physical layer, link layer, internet layer, transport layer, and application layer.

The physical layer is comprised of devices, objects, and things. The link layer operates on protocols like IEEE 802.15.4, IEEE 802.11, IS/IEC 18092:2004, Bluetooth, ANT, NB-IoT, EC-GSM-IoT, ISA100.11a, EnOcean, and LTE-MTC. The internet layer protocols are 6LoWPAN, IPv6, uIP, and NanoIP. The transport layer protocols are CoAP, TCP, UDP, MQTT, XMPP, AMQP, LLAP, DDS, SOAP, and DTLS. The application protocols are JSON-IPSO, REST API objects, and binary objects.

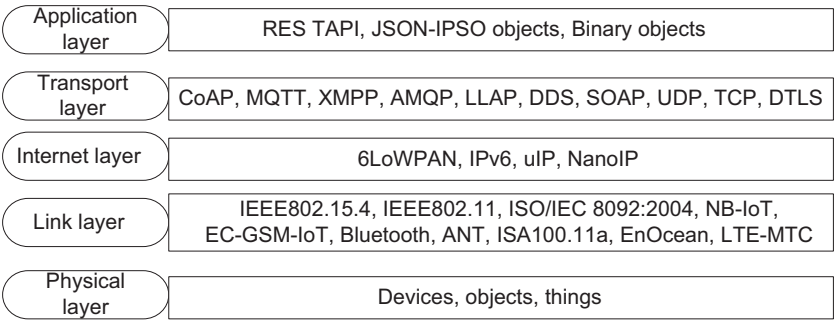

FIGURE 1.1
OSI model for IoT protocols.

1.3.2.2 Organizational Levels

IoT protocols can also be categorized on the basis of the organization levels, as follows:

1. Infrastructure (IPv4/IPv6, 6LowPAN, RPL)
2. Identification (EPC, IPv6, uCode, URIs)
3. Communication (Bluetooth, Wi-Fi, LPWAN)
4. Discovery (DNS-SD, mDNS, Physical Web)
5. Data Protocols (AMQP, MQTT, Websocket, CoAP, Node)
6. Device Management (TR-069, OMA-DM)
7. Semantic (Web Thing Model, JSON-LD)
8. Multi-layer Frameworks (Weave, IoTivity, Alljoyn, Homekit)

IPv6: IPv6 is popular as an internet layer protocol to transmit packets of information in an end-to-end transmission over multiple Internet Protocol (IP) networks.

6LoWPAN: 6LoWPAN stands for IPv6 over Low-power Wireless Personal Area Networks. It is an extended layer for IPv6 through IEEE802.15.4 links. It operates on 2.4 GHz of frequency with a data transmission rate of 250 kbps.

RPL: It is an IPv6-based routing protocol used in less power and lossy network.

UDP (User Datagram Protocol)**:** This protocol is meant for IP-based protocol networked between client/server. UDP is used in applications for real-time performance.

QUIC: It stands for Quick UDP Internet Connections. It supports the multiplexed connections between two endpoints over (UDP). It was designed for security protection to reduce latency in connection and transportation of information over a network.

µIP: The acronym is micro Internet Protocol. It is widely used due to open source TCP/IP stack, which can be used for tiny 8- and 16-bit microcontrollers.

DTLS (Datagram Transport Layer)**:** The DTLS protocol provides the communication privacy for datagram protocols. It is used for prevention of tampering, message forgery, or eavesdropping in a network.

NanoIP: Nano Internet Protocol is meant to establish communication among embedded sensors and devices without the overhead of TCP/IP.

Time-Synchronized Mesh Protocol (TSMP)**:** It is a communication protocol to establish communication among self-customized wireless sensor nodes called motes.

Physical Web: The Physical Web is an approach to interconnect devices and access them seamlessly.

HyperCat: It is an open source JSON-based lightweight hypermedia catalogue format for exposing collections of URIs.

MQTT (Message Queuing Telemetry Transport)**:** The MQTT is a lightweight protocol that enables a publish/subscribe messaging model. Used for remote connection in a network.

CoAP (Constrained Application Protocol): CoAP is an application layer protocol. It is designed to translate to HTTP for simplified integration with the web.

SMCP: It is a C-based CoAP stack, which can be used for embedded environments. It has fully asynchronous I/O and supports both UIP and BSD sockets.

STOMP: It stands for Simple Text Oriented Messaging Protocol used in communication networks.

XMPP: It stands for Extensible Messaging and Presence Protocol.

XMPP-IoT: It is the same as XMPP with an additional feature to establish a communication link between machine to people and machine to machine.

Mihini/M3DA: It acts as intermediate agent between the M2M server and embedded gateway. M3DA is an extended version to transport M2M binary data.

AMQP: The acronym is Advanced Message Queuing Protocol and is an open-source application layer used as middleware in a messaging application. It is reliable and more secure to use in routing and queuing.

DDS: It stands for Data Distribution Service for real-time systems. It is an open source and international standard to address communication among real time and embedded systems.

LLAP: It is elaborated as a lightweight local automation protocol. LLAP facilitates sending short and simple messages between intelligent objects.

REST: It stands for Representational State Transfer.

SOAP: It stands for Simple Object Access Protocol.

Websocket: It is a full-duplex socket used to communicate between server and client.

SensorML: It describes sensors and the measurement process by providing the standard models and an XML encoding.

RAML: The acronym is RESTful API Modelling Language. It is used to design and share the API.

IoTivity: It is founded by Linux Foundation to facilitate open-source projects and sponsored by OIC.

IEEE P2413: It is a standard for an architectural framework for the IoT.

OTrP (Open Trust Protocol)**:** This protocol is used to install, update, and delete applications. It manages the security configuration in a Trusted Execution Environment (TEE).

1.4 Enabling Technologies for IoT

In the present world, many wired and wireless technologies contribute to automation. IoT is the latest trend in technology. The networking part in IoT may involve more than one type of communication media or device.

1. **Short-Range Wireless Technology**

   Bluetooth networking in mesh: It is a Bluetooth low-energy (BLE) compatible mesh network with an increased number of nodes.

   Light-Fidelity (Li-Fi): This technology is almost similar to the Wi-Fi standard, but it uses the visible light spectrum.

   Near-field communication (NFC): It is a communication protocol that enables communication between two devices within a range of 4 cm.

   QR codes and barcodes: It is optical tag that can be read by machine; it stores the information for the item to which it is stacked to.

   Radio-frequency identification (RFID): It uses electromagnetic fields to read the information stored in tags on the other items.

   Thread: This network protocol is based on the IEEE 802.15.4 standard.

   Wi-Fi: It is for local area networking, which is based on the IEEE 802.11 standard.

   Z-Wave: It is a low-powered, low-latency, near-range communication protocol having better reliability than Wi-Fi.

   ZigBee: This protocol can be used for a personal area network; it is based on the IEEE 802.15.4 standard.

2. **Medium-Range Wireless Technology**

   HaLow: It is the variant of the Wi-Fi standard. It provides low data rate transmission over a wide range.

   LTE-Advanced: It is Long-Term Evolution technology meant to provide flawless communication with a high data rate.

3. **Long-Range Wireless Technology**

   Low-power wide-area networking (LPWAN): This wireless network facilitates a wide range of communication along with low bit rate and less power.

   Very small aperture terminal (VSAT): This communication is used in satellites using dish antenna for narrow-banded data.

4. **Wired Technology**

   Ethernet: It is a wired communication technique using a twisted pair and optical fiber with hubs or switches.

   Multimedia over Coax Alliance (MoCA): This technology enhances video quality over existing cable.

   Power-line communication (PLC): This communication technology uses the transmit of electrical power and data.

1.5 IoT Levels

Level 1 IoT: A level 1 IoT system performs sensing, actuation, storing, and analysis operations and is comprised of a single node/device. An example is a home automation system where a single node is designed to control the lights and appliances remotely.

Level 2 IoT: A level 2 IoT system performs sensing, actuation, and analysis and has a single node/device. This is suitable for big data analysis. The data is stored on the cloud. It is popular for cloud-enabled applications like smart farming.

Level 3 IoT: A level 3 IoT system is a single-node-based cloud platform. This type of system is suitable for big data needs that are computationally intensive. An example is the package tracking system. The system comprises of a single node (for a package), which monitors the vibration level of a package being shipped.

Level 4 IoT: A level 4 IoT system has multiple nodes that perform the analysis and data stored on the cloud. The system may have local and cloud-based server nodes that receive the information and upload on the cloud. Server nodes only process the information and perform no control action. It is suitable where multiple nodes are required and involve big data that is computationally intensive. An example is noise monitoring.

Level 5 IoT: A level 5 IoT system has multiple end nodes and a single coordinator node. The end node performs the sensing and/or actuation actions. Collections of data done by the coordinator node form the sensor nodes and communicates it to the cloud and is analyzed

ສອນໂດຍ ປອ. ສົມສັກ ອິນທະສອນ
tiny.one/IoT-BSc
facebook.com/somsacki

on the cloud. The system is suitable for a WSN-based solution with big data and computationally intensive requirement. An example is forest fire detection. The system is comprised of multiple nodes placed at different locations for monitoring temperature, humidity, and CO_2 levels in the forest.

Level 6 IoT: A level 6 IoT is comprised of sensor nodes and an actuator to perform sensing and controlling. It is a suitable cloud-based database designed for data analysis. The central controller knows the status of all end nodes and sends the control commands to the nodes. An example is a weather monitoring system. The system is comprised of multiple nodes that are placed at the different locations for monitoring temperature, humidity pressure, radiation, and wind speed. The sensor nodes are responsible for transmission of the data from end nodes to the destination via a websocket. The data is stored on the cloud-based server. The data analysis is done on the cloud to make the prediction by aggregating the data.

1.6 IoT vs M2M

IoT can be defined as a system where multiple objects communicate with each other and share data through sensors and digital connectivity. Machine-to-machine (M2M) solutions are comprised of linear communication channels between the machines to make them work in a cycle. Here, the action of one machine triggers the activity of other.

Differences between IoT and M2M

- A few experts define M2M as a subset of IoT, while others call the Internet of Things an evolved version of machine to machine. Either way, the conclusion is IoT is a broader area than M2M.
- Both the technologies work on the principle of connecting devices and make them to work together. While M2M relies on conventional connection tools like Wi-Fi, IoT has much flexibility and varied connectivity options.
- M2M solution has very limited scope and is confined to create a network of machines that work in synchronization. IoT creates 360° solutions for flexible responses and multi-level communication.
- The advantage of IoT over M2M is its ability to add interactivity amongst devices. Machine to machine operates by triggering responses based on an action. It is a one-way communication. In IoT-based systems, communication flows to and fro freely.