Department of Information Technology
19IT302 – Internet of Things
Unit 1 IoT INTRODUCTION AND APPLICATIONS

Overview and Motivations

The proliferation of an ever-growing set of devices able to be directly connected to the Internet is leading to a new ubiquitous-computing paradigm. Indeed, the Internet—its deployment and its use—has experienced significant growth in the past four decades, evolving from a network of a few hundred hosts (in its ARPAnet form) to a platform capable of linking billions of entities globally. Initially, the Internet connected institutional hosts and accredited terminals via specially developed gateways (routers). More recently, the Internet has connected servers of all kinds to users of all kinds seeking access to information and applications of all kinds. Now, with social media, it intuitively and effectively connects all sorts of people to people, and to virtual communities.

The next evolution is to connect all "things" and objects that have (or will soon have) embedded wireless (or wireline) connectivity to control systems that support data collection, data analysis, decision making, and (remote) actuation. "Things" include, but are not limited to, machinery, home appliances, vehicles, individual persons, pets, cattle, animals, habitats, habitat occupants, as well as enterprises. Interactions are achieved utilizing a plethora of possibly different networks; computerized devices of various functions, form factors sizes, and capabilities such as iPads, smartphones, monitoring nodes, sensors, and tags; and a gamut of host application servers.

In the IoT, commonly deployed devices and objects contain an embedded device or microprocessor that can be accessed by some communication mechanism, typically utilizing wireless links. The IoT aims at closing the gap between objects in the material world, the "things," and their logical representation in information systems. It is perceived by proponents asthe "next-generation network (NGN) of the Internet." Thus, the IoT is a new type of Internet application that endeavors to make the thing's information (whatever that may be) available on a global scale using the Internet as the underlying connecting fabric (although other interconnection data networks, besides the Internet, can also be used such as private local area networks and/or wide area networks).

The IoT has two attributes:

(i) being an Internet application and
(ii) dealing with the thing's information.

The "things" are also variously known as "objects," "devices," "end nodes," "remotes," or "remote sensors," to list just a few commonly used terms.
The IoT generally utilizes low cost information gathering and dissemination devices—such as

sensors and tags—that facilitate fast-paced interactions in any place and at any time, among the objects themselves, as well as among objects and people. Actuators are also part of the IoT.

Hence, the IoT can be described as a new-generation information network that enables seamless and continuous machine to machine (M2M) and/or human-to-machine (H2M) communication. One of the initial goals of the IoT is to enable connectivity for the various "things"; a next goal is to be able to have the "thing" provide back appropriate, application-specific telemetry; an intermediary next step is to provide a web-based interface to the "thing".

At the "low end" of the spectrum, the thing's information is typically coded by the unique identification (UID) and/or electronic product code (EPC); the information is (typically) stored in a radio frequency identification (RFID) electronic tag; and, the information is uploaded by noncontact reading using an RFID reader.

At the "mid range" of the spectrum, one finds devices with embedded intelligence (microprocessors) and embedded active wireless capabilities to perform a variety of data gathering and possibly control functions. On-body biomedical sensors, home appliance and power management, and industrial control are some examples of these applications.

At the other end of the spectrum, more sophisticated sensors can also be employed in the IoT: some of these sensor approaches use distributed wireless sensor network (WSN) systems that

(i) can collect a wide variety of environmental data such as temperature, atmospheric and environmental chemical content, or even low- or high resolution ambient video images from geographically dispersed locations;
(ii) can optionally pre-process some or all of the data; and
(iii) can forward all these information to a centralized (or distributed/virtualized) site for advanced processing. These objects may span a city, region, or large distribution grid.

Other "things" may be associated with personal area networks (PANs), vehicular networks (VNs), or delay tolerant networks (DTNs)

Below Figure 1.1 depicts the high level logical partitioning of the interaction space, showing where the IoT applies for the purpose of this text; the figure illustrates

- human-to-human (H2H) communication,
- M2M communication, H2M communications, and
- machine in (or on) humans (MiH) communications (MiH devices may include human embedded chips, medical monitoring probes, global positioning system (GPS) bracelets, and so on).

The focus of the IoT is on M2M, H2M, and MiH applications;

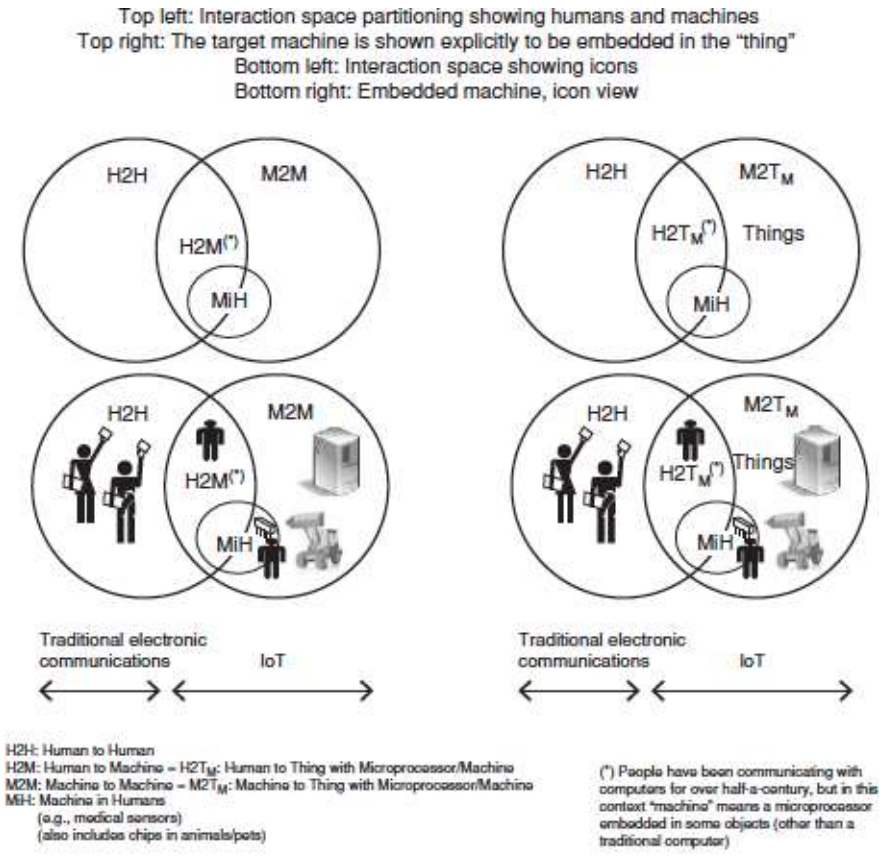


FIGURE 1.1 H2H, H2M, and M2M environment.

M2M services aim at automating decision and communication processes and support consistent, cost-effective interaction for ubiquitous applications (e.g., fleet management, smart metering, home automation, and e-health). M2M communications per se is the communication between two or more entities that do not necessarily need direct human intervention.

IoT applications range widely from energy efficiency to logistics, from appliance control to "smart" electric grids. Indeed, there is increasing interest in connecting and controlling in real time all sorts of devices for personal healthcare (patient monitoring and fitness monitoring), building automation (also known as building automation and control (BA&C)—for example, security devices/cameras; heating, ventilation, and air-conditioning (HVAC); AMRs), residential/commercial control (e.g., security HVAC, lighting control, access control, lawn and garden irrigation), consumer electronics (e.g., TV, DVRs); PC and peripherals (e.g., mouse, keyboard, joystick, wearable computers), industrial control (e.g., asset management, process control, environmental, energy management), and supermarket/supply chain management.

IPv6 Role

Using IPv6 with its abundant address spaces, globally unique object (thing) identification and connectivity can be provided in a standardized manner without additional status or address (re)processing—hence, its intrinsic advantage over IPv4 or other schemes.

It is both desirable as well as feasible for all physical (and even virtual or logical) objects to have

a permanent unique identifier, an object ID (OID). It is also desirable as well as feasible for all end-point network locations and/or intermediary-point network locations to have a durableunique network address (NAdr); the IPv6 address space enables the concrete realization of these goal.

When objects that have enough intelligence to (run a communication protocol stack so that they can) communicate are placed on a network, these objects can be tagged with an NAdr. Every object then has a tuple (OID, NAdr) that is always unique, although the second entry of the tuple may change with time, location, or situation.

In a stationary, nonvariable, or mostly static environment, one could opt, if one so chose, to assign the OID to be identical to the NAdr where the object is expected to attach to the network. However, there is a general trend toward object mobility, giving rise to a dynamic environment (e.g., for mobile or variable case); hence, to retain maximal flexibility it is best to separate, in principle, the OID from the NAdr and thus assign a general (OID, NAdr) tuple where the OID is completely invariant.

What was described above is not feasible in an IPv4 world, because in the 32-bit address space, only 2^{32}~10^{10} NAdr location can be identified uniquely. IPv6 offers a much larger 2^{128} space; hence, the number of available unique node addressees is 2^{128}~10^{39}. IPv6 has more than 340 undecillion (340,282,366,920,938,463,463,374, 607,431,768,211,456) addresses, grouped into blocks of 18 quintillion addresses. Already today many tags operate with a 128-bit OID field that allows 2^{128}~10^{39} ($\approx 3.4 \times 10^{38}$) unique identifiers, but the tuple (OID, NAdr = OID) could not be defined uniquely in the IPv4 world.

IPv6 was originally defined in 1995 in request for comments (RFC) 1883 and then further refined by RFC 2460, “Internet Protocol, Version 6 (IPv6) Specification,” authored by S. Deering and R. Hinden.

IPv6 embodies IPv4 best practices but removes unused or obsolete IPv4 characteristics; this results in a better-optimized Internet protocol. Some of the advantages of IPv6 include the following

- Scalability and expanded addressing capabilities: as noted, IPv6 has 128-bit addresses versus 32-bit IPv4 addresses. With IPv4, the theoretical number of available IP addresses is 2^{32}~10^{10}. IPv6 offers a much larger 2^{128} space. Hence, the number of available unique node addressees is 2^{128}~10^{39}.
- “Plug-and-play”: IPv6 includes a “plug-and-play” mechanism that facilitates the connection of equipment to the network. The requisite configuration is automatic; it is a serverless mechanism.
- Security: IPv6 includes and requires security in its specifications such as payload encryption and authentication of the source of the communication. End-to-end security, with built-in strong IP-layer encryption and authentication (embedded security support with mandatory IP security (IPsec) implementation), is supported.
- Mobility: IPv6 includes an efficient and robust mobility mechanism namely an enhanced support for mobile IP, specifically, the set of mobile IPv6 (MIPv6) protocols, including the base protocol defined in RFC 3775.

IoT Definitions

General Observations

Some applicable observations related to the definition of the IoT include the following:
Internet of Things is a twenty-first century phenomenon in which physical consumer products (meta products) connect to the web and start communicating with each other by means of sensors and actuators.

Originally the term "Internet of Things" was invented by the MIT Auto-ID Center in 2001 and referred to an architecture that comprises four elements,

– Passive radio frequency identification (RFIDs), such as Class-1 Generation-2 UHFRFIDs, introduced by the electronic product code (EPC) Global Consortium and operating in the 860– 960 MHz range1
– Readers plugged to a local (computing) system, which read the EPC
– A local system offering IP connectivity that collects information pointed by the EPC, thanks to a protocol called object naming service (ONS)
– EPCIS (EPC Information Services) servers that process incoming ONS requests and returns physical markup language (PML) files, for example, XMLdocuments carrying meaningful information linked to RFIDs.

ITU-T Views

The ITU-T is in the process of identifying a common way to define/describe the IoT.
So far, the ITU-T has not found "a good definition to cover all aspects of IoT as the IoT has quite big scope not only the technological viewpoints but also other views.

View A: IoT is just a concept (conceptual aspects of definition): the IoT does not refer to a network infrastructure; the IoT is not a technical term but a concept (or a phenomenon).

View B: IoT is an infrastructure: The IoT refers to an infrastructure.

The ITU-T is suggesting to define the IoT as a short definition with more general concept rather than as a technical definition.

ITU-T "strongly insists on a short definition as concept instead of a technical definition (long or detailed description of technology.

View A: IoT is just a concept (conceptual aspects of definition): the IoT does not refer to a network infrastructure; the IoT is not a technical term but a concept (or a phenomenon).

TABLE 2.1 Examples of Definitions for Case A (IoT is Just a Concept)

Candidate Definition	Reference
A technological revolution that represents the future of computing and communications, and its development depends on dynamic technical innovation in a number of important fields, from wireless sensors to nanotechnology	Source: ITU Internet Reports 2005: The Internet of Things, Executive Summary
The networked interconnection of objects—from the sophisticated to the mundane—through identifiers such as sensors, RFID tags, and IP addresses	Margery Conner, Technical Editor of EDN Magazine, "Sensors empower the 'Internet of Things' ", May 2010
The Internet of things *links the objects of the real world with the virtual world*, thus enabling anytime, anyplace connectivity for anything and not only for anyone. It refers to a world where physical objects and beings, as well as virtual data and environments, all interact with each other in the same space and time	Cluster of European Research Projects on the Internet of Things, "Vision and Challenges for Realizing the Internet of Things", March 2010
The IoT refers to as *ubiquitous networking or pervasive computing environments*, is *a vision* where all manufactured things can be network enabled, that is connected to each other via wireless or wired communication networks	European Network and Information Security Agency (ENISA)
The IoT is *a world where physical objects are seamlessly integrated into the information network*, and where the physical objects can become active participants in business processes. Services are available to interact with these "smart objects" over the Internet, query and change their state and any information associated with them, taking into account security and privacy issues. RFID, sensor networks, and so on are just enabling technologies	SAS

View B: IoT is an infrastructure: The IoT refers to an infrastructure.

TABLE 2.2 Examples of Definitions for Case B (Infrastructural Aspects of Definition)

Candidate Definition	Reference
A *global network infrastructure*, linking physical and virtual objects through the exploitation of data capture and communication capabilities. This infrastructure includes existing and evolving Internet and network developments. It will offer specific object identification, sensor and connection capability as the basis for the development of independent federated services and applications. These will be characterized by a high degree of autonomous data capture, event transfer, network connectivity, and interoperability	Coordination and Support Action (CSA) for Global RFID-related Activities and Standardization (CASAGRAS)
A global information and communication infrastructure enabling automated chains of actions (not requiring explicit human intervention) facilitating information assembly and knowledge production and contributing to enrichment of human life by interconnecting physical and logical objects based on standard and interoperable communication protocols and through the exploitation of data capture and communication capabilities supported by existing and evolving information and communication technologies NOTE: Physical objects may include sensors, devices, machines, and so on. Logical objects may include contents and so on	Originally produced by the discussion among China-Japan-Korea. ITU Q3/13 has made some modifications
A global ICT infrastructure linking physical objects and virtual objects (as the informational counterparts of physical objects) through the exploitation of sensor and actuator data capture, processing and transmission capabilities. As such, the IoT is an overlay above the "generic" Internet, offering federated physical-object-related services (including, if relevant, identification, monitoring, and control of these objects) to all kinds of applications.	Proposed by France Telecom on the IoT definition mailing list.

Working Definition

Generalizing from the published literature and the observations made thus far in this text, we characterize the IoT with a "working definition" as follows:

Definition: A broadly-deployed aggregate computing/communication application and/or application-consumption system, that is deployed over a local (L-IoT), metropolitan (M-IoT), regional (R-IoT), national (N-IoT), or global (G-IoT) geography, consisting of (i) dispersed instrumented objects ("things") with embedded one or two-way communications and some (or, at times, no) computing capabilities, (ii) where objects are reachable over a variety of wireless or wired local area and/or wide area networks, and, (iii) whose inbound data and/or outbound

commands are pipelined to or issued by a(n application) system with a (high) degree of (human or computer-based) intelligence.

Two other related "working definitions" are as follows:
Definition: Sensors are active devices that measure some variable of the natural or man-made environment (e.g., a building, an assembly line, an industrial assemblage supporting a process).

Sensor networks usually consider remote devices as belonging to two classes, based on device capabilities:
Full-function devices (FFDs) and
Reduced function devices (RFDs).

Sensors and actuators are part of a larger universe of objects. Objects in the IoT context can also be classified from a functionality perspective.

Definition: An actuator is a mechanized device of various sizes (from ultra-small to very large) that accomplishes a specified physical action, for example, controlling a mechanism or system, opening or closing a valve, starting some kind or rotary or linear motion, or initiating physical locomotion. An actuator is the mechanism by which an entity acts upon an environment.

The actuator embodies a source of energy, such as an electric current (battery, solar, motion), and a source of physical interaction such as a hydraulic fluid pressure or a pneumatic pressure; the device converts that energy into some kind of action or motion upon receipt of an external command or stimulus.

An object is a model of an entity. An object is distinct from any other object and is characterized by its behavior. An object is informally said to perform functions and offer services.

Objects have the following characteristics, among others:

- have the ability to sense and/or actuate
- are generally small (but not always)
- have limited computing capabilities (but not always)
- are energy/power limited
- are connected to the physical world
- sometimes have intermittent connectivity
- are mobile (but not always)
- of interest to people
- managed by devices, not people (but not always)

An M2M/H2M environment comprises three basic elements:
(i) the data integration point (DIP);
(ii) the communication network; and;
(iii) the data end point (DEP)

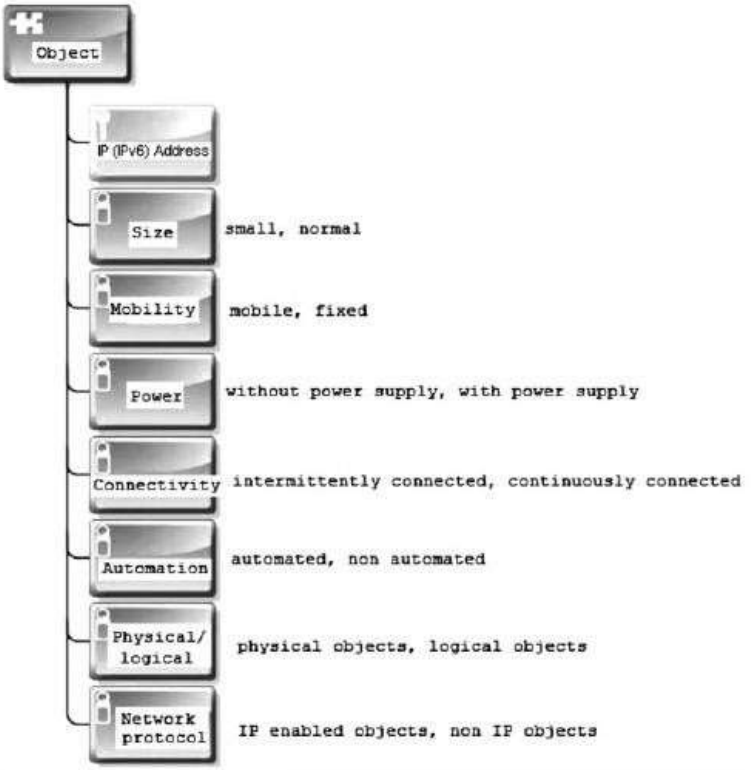
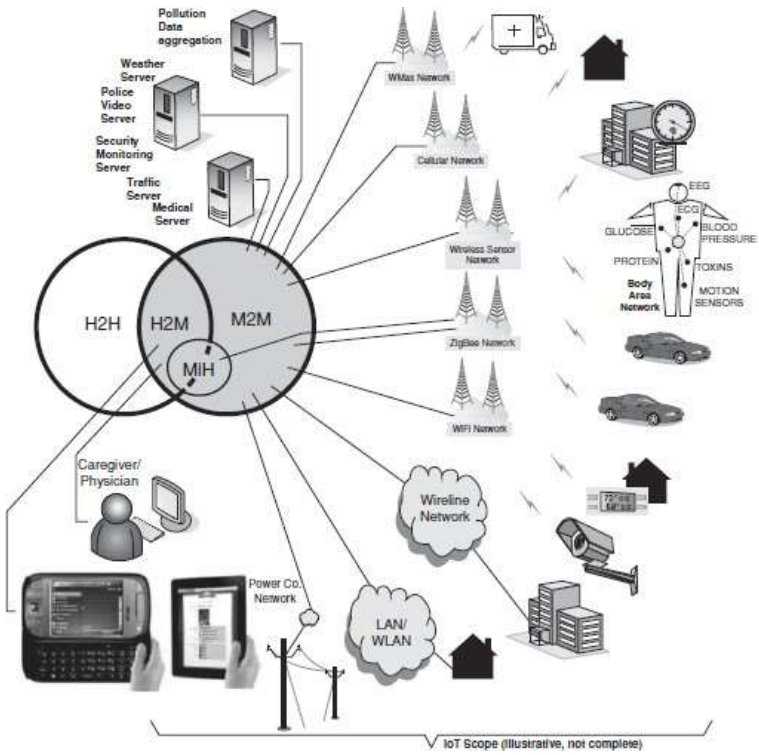


FIGURE 2.2 Object classification.



Scope of IoT

IoT FRAMEWORKS

A high level M2Msystem architecture (HLSA)

The HLSA comprises the device and gateway domain, the network domain, and the applications domain.

The device and gateway domain is composed of the following elements

1. M2Mdevice: A device that runs M2M application(s) using M2M service capabilities.
M2M devices connect to network domain in the following manners:

- Case 1 "Direct Connectivity": M2Mdevices connect to the network domain via the access network. The M2M device performs the procedures such as registration, authentication, authorization, management, and provisioning with the network domain. The M2M device may provide service to other devices (e.g., legacy devices) connected to it that are hidden from the network domain.

- Case 2 "Gateway as a Network Proxy": The M2M device connects to the network domain via an M2M gateway. M2M devices connect to the M2M gateway using the M2M area network. The M2M gateway acts as a proxy for the network domain toward the M2M devices that are connected to it.

2. M2M area network: It provides connectivity betweenM2Mdevices andM2M gateways. Examples of M2M area networks include personal area network (PAN) technologies such as IEEE 802.15.1, Zigbee, Bluetooth, IETF ROLL, ISA100.11a, among others, or local networks such as power line communication (PLC), M-BUS, Wireless M-BUS, and KNX.

3. M2M gateway: A gateway that runs M2M application(s) using M2M service capabilities. The gateway acts as a proxy betweenM2Mdevices and the network domain. The M2M gateway may provide service to other devices (e.g., legacy devices) connected to it that are hidden from the network domain. As an example, an M2M gateway may run an application that collects and treats various information (e.g., from sensors and contextual parameters).

The network domain is composed of the following elements:

1. **Access network**: A network that allows the M2M device and gateway domain to communicate with the core network. Access networks include (but are not limited to) digital subscriber line (xDSL), hybrid fiber coax (HFC), satellite, GSM/EDGE radio access network (GERAN), UMTS terrestrial radio access network (UTRAN), evolved UMTS terrestrial radio access network (eUTRAN), W-LAN, and worldwide interoperability for microwave access (WiMAX).

2. Core network: A network that provides the following capabilities (different core networks offer different features sets):
– IP connectivity at a minimum, and possibly other connectivity means
– Service and network control functions
– Interconnection (with other networks)
– Roaming

Core networks (CoNs) include (but are not limited to) 3GPP CoNs, ETSI TISPAN CoN, and 3GPP2 CoN.

3. M2M service capabilities:
– Provide M2M functions that are to be shared by different applications
– Expose functions through a set of open interfaces
– Use CoN functionalities
– Simplify and optimize application development and deployment through hiding of network specificities.

The "M2M service capabilities" along with the "core network" is known collectively as the "M2M core."

The applications domain is composed of the following elements:
1. M2M applications: Applications that run the service logic and use M2M service capabilities accessible via an open interface.

There are also management functions within an overall M2M service provider domain, asfollows:

1. Network management functions: Consists of all the functions required to manage the access and core networks; these functions include provisioning, supervision, fault management.

2. M2Mmanagement functions: Consists of all the functions required to manage M2Mservice capabilities in the network domain. The management of theM2M devices and gateways uses a specific M2M service capability.

Basic Nodal Capabilities
Consistent with the HLSA, a remote device generally needs to have a basic protocol stack that supports as a minimum local connectivity and networking connectivity; in addition, some higher layer application support protocols are generally needed, with varying degrees of computational/functional sophistication.

Distributed control/M2M typically entails continuously changing variables to control the behavior of an application. Typical requirements include the following capabilities

Retransmission
– Network recovers from packet loss or informs application
– Recovery is immediate: on the order of RTTs, not seconds
Network independent of MAC/PHY
Scale
– Thousands of nodes
– Multiple link speeds
Multicast
– Throughout network
– Reliable (positive Ack)

Duplicate suppression
Emergency messages
– Routed and/or queued around other traffic
– Other traffic slushed as delivered
Routine traffic delivered in sequence
Separate timers by peer/message
Polling of nodes
– Sequential
– Independent of responses
Paradigm supports peer-to-peer
– Not everything is client/server
Capabilities
– Discover nodes
– Discover node capabilities
– Deliver multisegment records (files)
Exchange of multisegment records
Network and application versioning
Simple publish/subscribe parsers
Security
– Strong encryption
– Mutual authentication
– Protection against record/playback attacks
– Suite B ciphers

Physical Design of IoT

The "Things" in IoT usually refers to IoT devices which have unique identities and can perform remote sensing, Actuating and monitoring capabilities. IoT devices can exchange data with other connected devices and applications (directly or indirectly), or collect data from other devices and process the data locally or send the data to Centralized servers or cloud based applications back ends for processing the data or from some task locally and other task within the IoT infrastructure, based on temporal and space constraints (ie: Memory, processing calibrators, communication latencies and speed and deadlines).

An IoT device may consist of several interfaces connections to other devices, both wired and wireless. These include

I) IoT interfaces for sensors
II) interfaces for internet connectivity
III) memory and storage interfaces
IV) audio video interfaces.

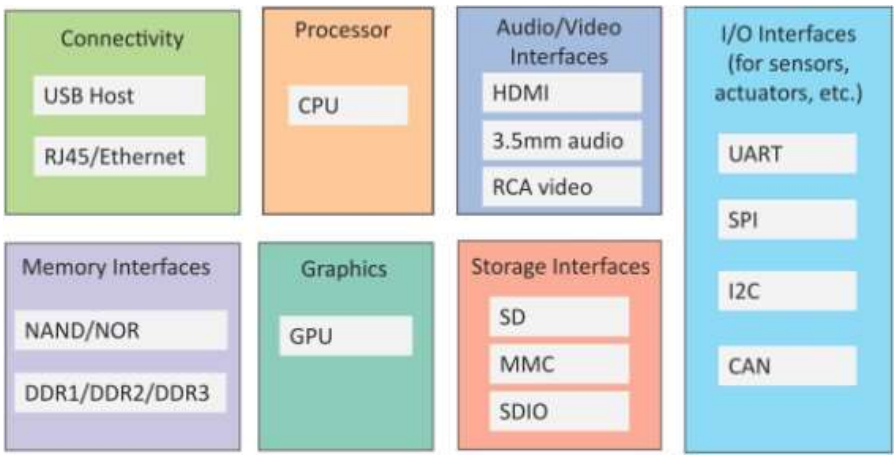
An IoT Device can collect various types of data from the the onboard or attached sensors, such as temperature e , humidity, light intensity.

IoT devices can also be varied types, for instance, wearable sensors, smart watches, LED light automobiles and industrial machines.

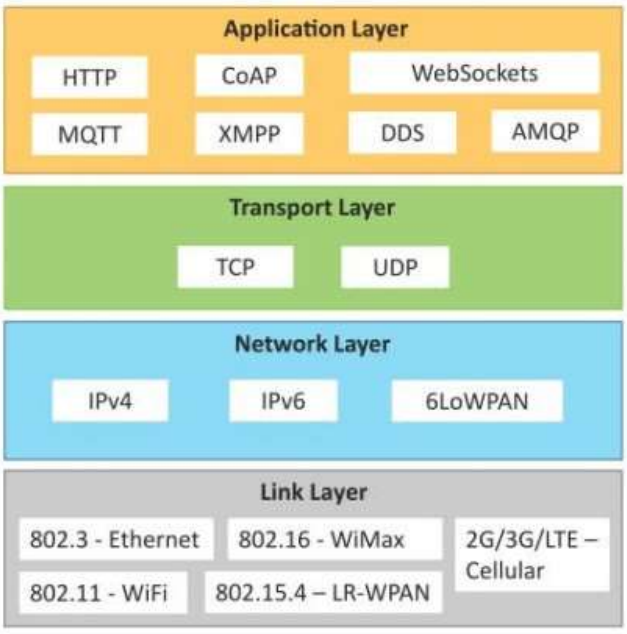
IoT Protocol

Link Layer:

Link Layer protocols determine how the data is physically sent over the networks physical layer or medium(example copper wire, electrical cable, or radio wave). The Scope of The Link Layer is the Last Local Network connections to which host is attached. Host on the same link exchange data packets over the link layer using the link layer protocol. Link layer determines how the packets are coded and signaled by the hardware device over the medium to which the host is attached.



Generic block diagram of an IoT Device



IoT Protocols

802.3 Ethernet:

802.3 is a collections of wired Ethernet standards for the link layer. For example 802.3 10BASE5 Ethernet that uses coaxial cable as a shared medium, 802.3.i is standard for 10 BASET Ethernet over copper twisted pair connection, Standards provide data rates from 10 Mb/s to 40 gigabits

per second and the higher. The shared medium in Ethernet can be a coaxial cable , twisted pair wire or and Optical fiber. Shared medium carries the communication for all the devices on the network.

802.1- WI-FI:
IEEE 802.3 is a collections of wireless Local area network.(WLAN) communication standards, including extensive descriptions of the link layer. For example 802.11a operate in the 5 GHz band, 802.11b and 802.11g operate in the 2.4 GHz band. 802.11ac operates in the 5G hertz band.

802.16 wiMAX:
IEEE 802.16 is a collection of wirless broadband and Standards, including extensive descriptions for the link layer also called WiMAX wimax standard provides a data rates from from 1.5 Mb/s to 1Gb/s the recent update provides data rates of hundred megabits per second for mobile station.

802.15.4 LR-WPAN:
IEEE 802.1 5.4 is a collections of standard for low rate wireless personal area network(LR-WPAN).These standard form the basis of specifications for high level communication Zigbee. LR-WPAN standards provide data rates from 40 k b/ s. These standards provide low cost and low speed Communications for power constrained devices.

2G / 3G / 4G mobile communications:
These are the different generations of mobile communication standards including second generation (2G including GSM and CDMA). 3rd Generation (3G including UMTS and CDMA2000) and 4th generation 4G including LTE.

Network / Internet layer :
The network layer are responsible for sending of IP datagrams from the source network to the destination network. This layer Performs the host addressing and packet routing. The datagrams contains a source and destination address which are used to route them from the source to the destination across multiple networks. Host Identification is done using the hierarchy IP addressing schemes such as ipv4 or IPv6.

IPV4: Internet protocol versions for open parents close (IPV4) is there most deployed internet protocol that is used to identify the device is on a network using a hierarchy latest schemes. It uses 32 bit addresses scheme that allows total of 2 32 address. As more and more devices got connected to the internet. The Ipv4 has succeeded by IPv6.

IPv6: It is the newest versions of internet protocol and successor to IPv4. IPv6 uses 128 bit address schemes that are lost total of 2 128 are 3.4* 10 38 address.

6LoWPAN:
IPv6 over low power wireless personal area networks brings IP protocol to the low power device which have limited processing capability it operate in the 2.4 GHz frequency range and provide the data transfer rate off to 50 kb/s.

Transport layer :
The Transport layer protocols provides end-to-end message transfer capability independent of the underlying network. The message transfer capability can be set up on connections, either using handshake or without handshake acknowledgements. Provides functions such as error control , segmentation, flow control and congestion control.

TCP: Transmission control protocol is the most widely used to transport layer protocol that is used by the web browsers along with HTTP , HTTPS application layer protocols email program (SMTP application layer protocol) and file transfer protocol. TCP is a connection Oriented and stateful protocol while IP protocol deals with sending packets, TCP ensures reliabletransmissions of packets in order. TCP also provide error deduction capability so that duplicate packets can be discarded and low packets are retransmitted. The flow control capability ensures that the rate at which the sender since the data is now to too to high for the receiver to process.

UDP: unlike TCP, which requires carrying out an initial setup procedure, UDP is a connection less protocol. UDP is useful for time sensitive application they have very small data units to exchange and do not want the overhead of connection setup. UDP is a transactions oriented and stateless protocol. UDP does not provide guaranteed delivery, ordering of messages and duplicate eliminations.

Application layer:
Application layer protocol define how the application interfaces with the lower layer protocols to send the data over the network. Data are typically in files, is encoded by the application layer protocol and encapsulated in the transport layer protocol .Application layer protocol enable process-to-process connection using ports.

HTTP: Hypertext transfer protocol is the application layer protocol that forms the foundations of world wide web http includes, ,commands such as GET, PUT, POST, DELETE, HEAD, TRACE, OPTIONS etc. The protocol follows a request-response model where are client sends request to server using the http, commands. Http is a stateless protocol and each http request is independent father request and http client can be a browser or an application running on theclient example and application running on an IoT device ,mobile mobile applications or other software.

CoAP: Constrained application protocol is an application layer protocol for machine to machine application M2M meant for constrained environment with constrained devices and constrained networks. Like http CoAP is a web transfer protocol and uses a request- response model,however it runs on the top of the UDP instead of TC CoAP uses a client –server architecture where client communicate with server using connectionless datagrams.It is designed to easily interface with http like http,CoAP supports method such as GET, PUT, DELETE .

Websocket: Websocket protocol allows full duplex communication over a single socket connections for sending message between client and server. Websocket is based on TCP and Allows streams of messages to be sent back and forth between the client and server while keeping the TCP connection open. The client can be a browser, a mobile application and IoT device

MQTT: Message Queue Telemetry Transport it is a lightweight message protocol based on public -subscribe model MQTT uses a client server Architecture by the clients such as an IoT device connect to the server also called the MQTT broker and publishers message to topic on the server. The broker forward the message to the clients subscribed to topic MQTT is well suited for constrained and environments.

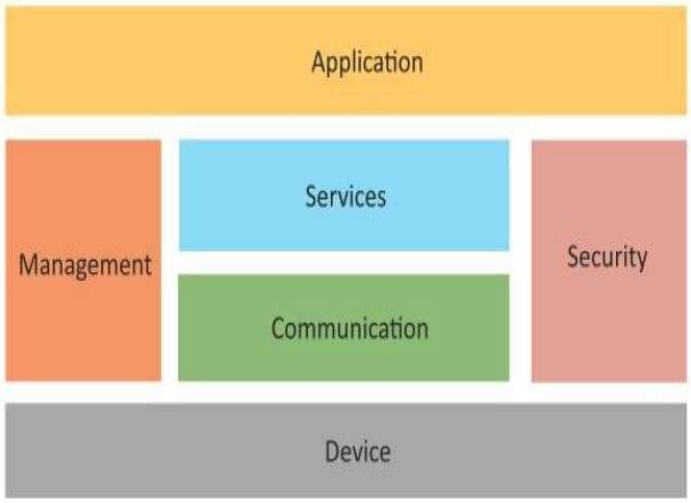
XMPP: Extensible Messaging and Presence Protocol it is a protocol for real-time communication and streaming XML data between network entities XMPP powers wide range of applications including messaging, presence, data syndication, gaming multiparty chat and voice / voice calls. XMPP Allows sending small chunks of XML data from one network entity to another in real time. XMPP supports both client to server and server –client communication path.

DDS: Data distribution service is the date centric middleware standard for device-to-device machine to machine communication DDS uses a publish subscribe model where publisherexample device that generate data create topics to which subscribers per can subscribe publisher is an object responsible for data distributions and the subscriber responsible for receiving published data. DDS provide quality of service (QoS) control and configurable reliability

AMQP: Advanced Message Queuing protocols. it is an open application layer protocol for business messaging. AMQP support point to point and publish - subscribe model routing and queuing. AMQP broker receive message from publishers example devices or applications that generate data and about them over connections to consumers publishers publish the message to exchange which then distribute message copies to queues.

Logical Design of IoT

Logical design of an IoT system refers to an abstract representation of the entities and process without going into low level specification of the implementations.



Functional Blocks of IoT

IoT functional block

An IoT system comprises of a number of functional blocks that provide the system the capabilities for identification , sensing, actuation ,communication and Management.
The function blocks are described as follows

Devices: An IoT system comprises of the devices that provide sensing, actuation, monitoring and control function

Communication: communication block handle the communication systems

Services: An IoT system uses various types of IoT services such as services for device monitoring ,device control services ,data publishing services and services for device Discovery.

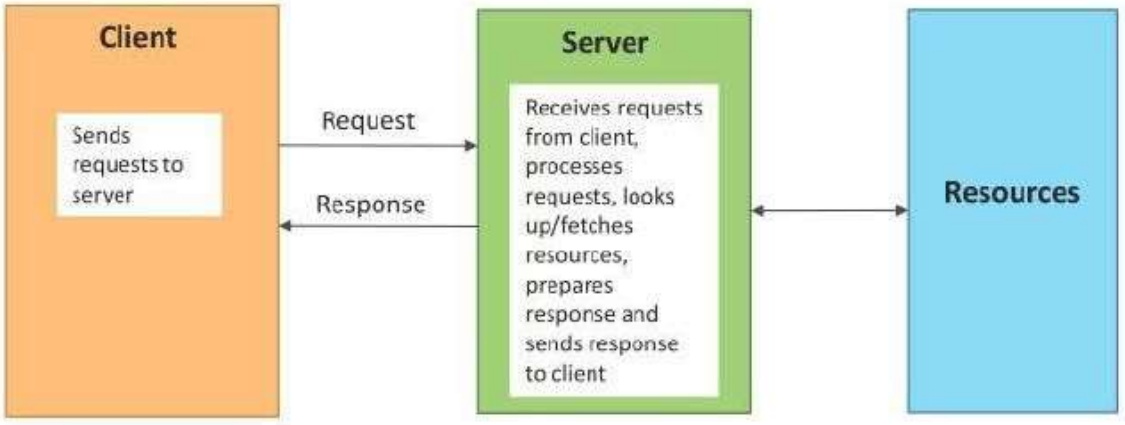
Management: Functional blocks provide various functions to govern the IoT system

Security: Security functional block security IoT system and by providing functions such as application authorization message and content integrity and data security.

Application: IoT application provides and interface that the user can used to control and monitor various aspects of the IoT system. Application also allow users to view the system status and view or analyze the processed to data.

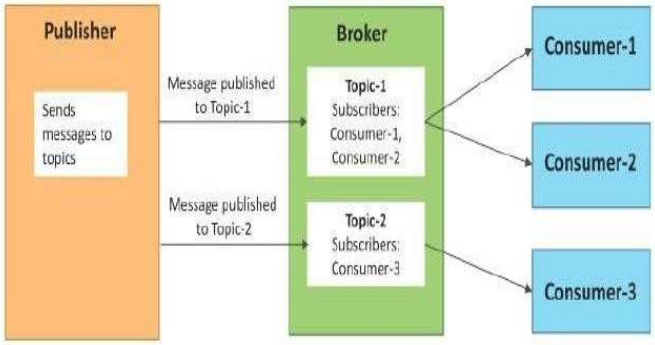
IoT communication model

1. **Request response**: Request-response is a Communications model in which the client sends request to the server and the server responds to the requests. when the server receives a request it decides how to respond, if it shows the data retrieved resources definitions for the response , and then send the response to the client. Access to response model is a stateless communication model and each request response per is independent of others the crime and server interactions inthe request response model.

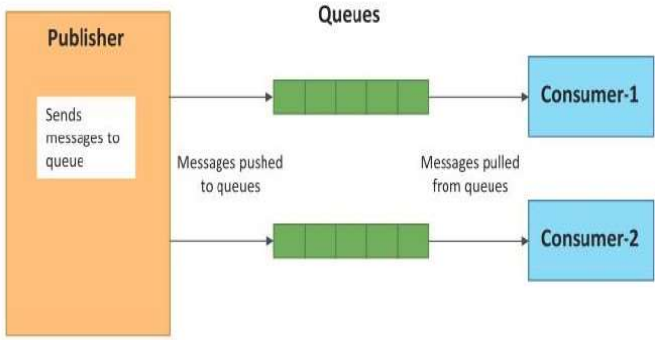


2. **Publish - Subscribe:** Publish - Subscribe is a communication model that involve Publishers brokers and consumers. Publishers are the source of data. Publishers send the data to the topics which is managed by the broker. Publishers are not aware of the consumer. Consumers Subscribe to the topic which are managed by the broker. When the broker receives the data for a topic from

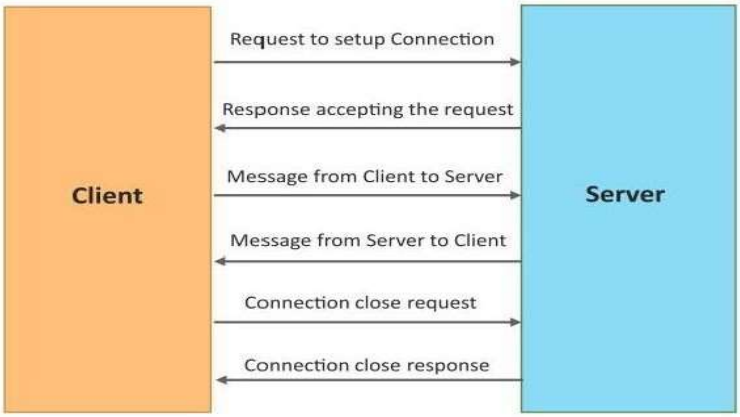
the publisher, it send the data to all the subscribed consumers.



3.Push pull: Push pull is communication model in which the data producers push the data to queues and the consumers pull the data from the queues. Producers do not need to be aware of the consumer. Queues help in decoupling the messaging between the Producers and Consumers . It also act as a buffer which helps in situations when there is a mismatch between the rate at which the produces push data and the rate at which the consumers full the data



4.Exclusive pair: Exclusive pair is a bi directional, fully duplex communication model that uses a persistent connections between the client and the server. once the condition is setup it remains open until the client sends a request to close the connection. client and server can send messages to each other after connection setup. Exclusive pair is a stateful Communications model and the server is aware of all the open connections.

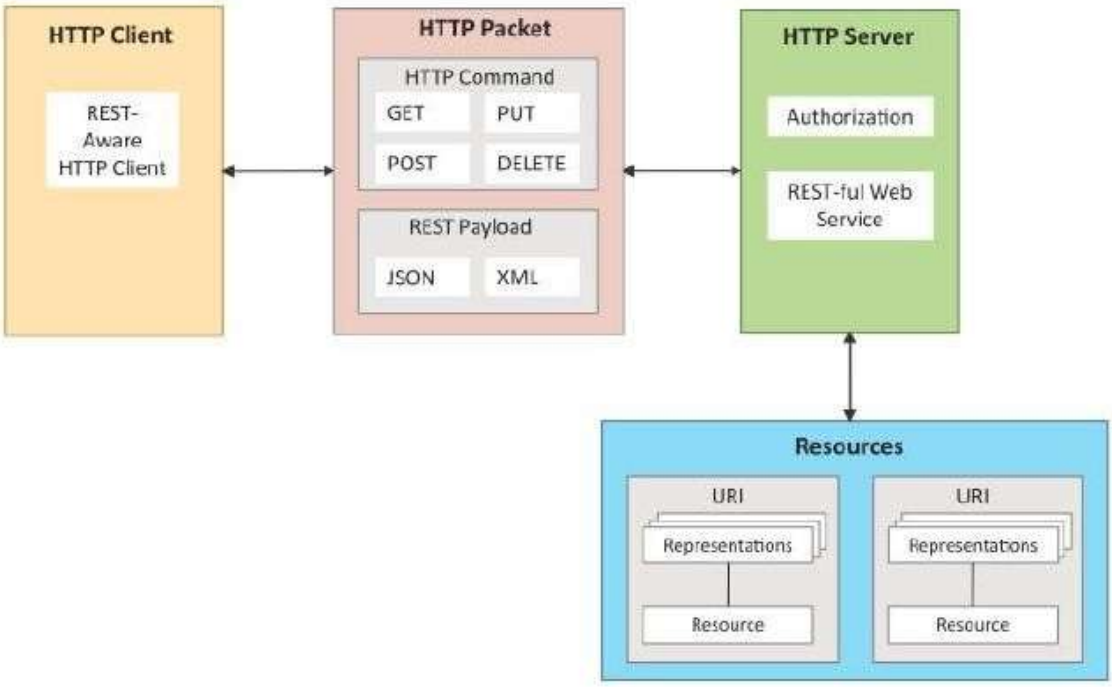


IoT communication APIs

REST- based communication API:

Representational state transfer is a set of architectural principles by which you can design web service and Web API that focus on a system resources and how resources states and addressed the transferred. REST API follow the request- response communication model.

The REST architectural constraints apply to the components, connectors, and data elements,



The REST architectural constraints are as the follows,

Client server: The principle behind the client-server conference separations of concerns for example client should not be concerned with the storage of data which is their concern of the server. Similarly the server should not be concerned about the user interface which is a concern of the client. separation allows client and server to be independently deployed and updated.

Stateless: Each request from client to server must contain all the information necessary to understand the request , and cannot take advantage of any stored context on the server .

Catchable: Catch constrain requires that the data within the response to a request be implicitly or explicitly labeled as catchable or non-catchable. Then a client cache is given the right to reuse that response data for later, equivalent requests. completely eliminate some attractions andimprove efficiency and scalability.

Layered system: System constraint come off constraints, constrains the behavior of components such that each component cannot see beyond the immediate layer with which they are interacting. Example client cannot tell whether it is connected directly to the end server or to an intermediary along the way system scalability can be improved allowing intermediaries to

respond to request instead of tender server.

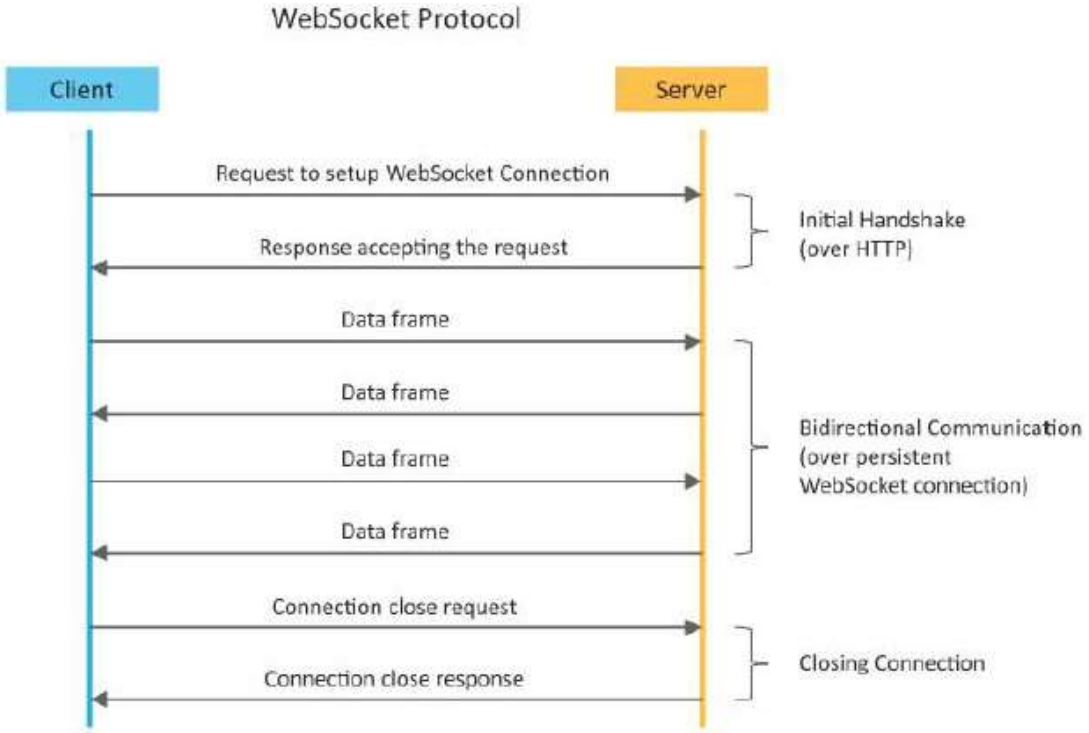
Uniform interface: Uniform interface constraints requires that the method of communication between client and server must be uniform. Resources are identified in the request and separate from the representation of the resource that are returned to the client. When climbing holds a representation of your resource it has all the information required to update or delete theresource.

Code on demand : Service can provide executable code script for clients to execute in their context.

WebSocket based communication API:
WebSocket API allow bi directional, full duplex communication between client and server. Unlike request-response API allow full duplex communication and do not require new connection to be set up for each message to be sent. Websocket communication begins with connection setup request send by the client to the server.

The request is sent over http and the server interprets it as an upgrade request. If the server support protocol response to the website handshake response after the connection setup the clientand the server can send data or messages to each other in full duplex model.

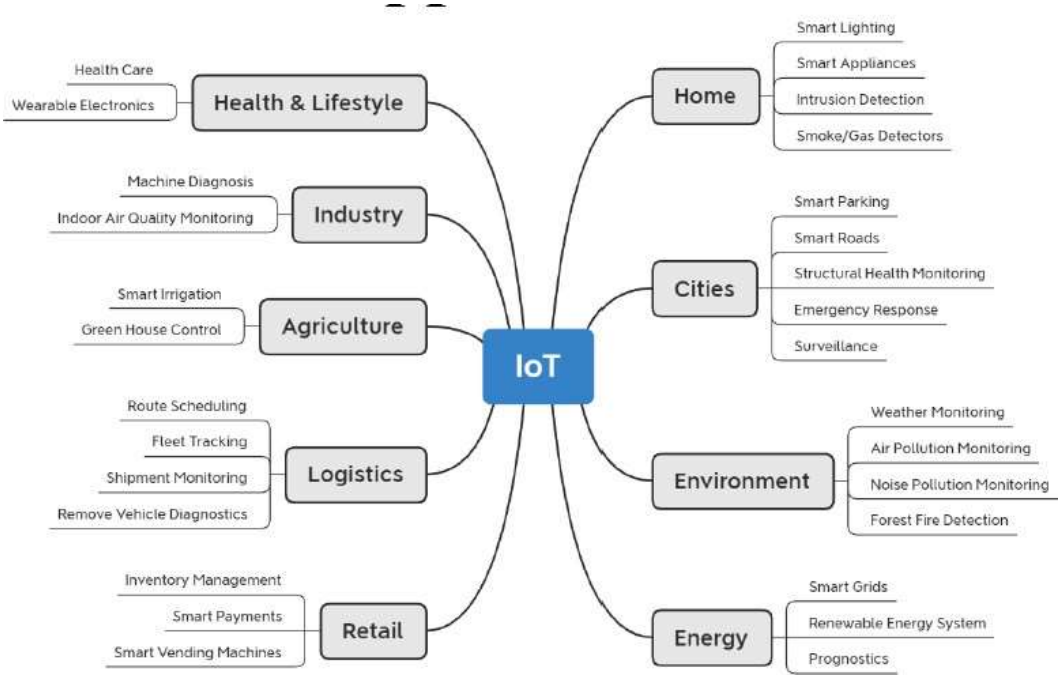
WebSocket API reduce network traffic and latency as there is no overhead for connection setup and determination records to each message.



IoT Applications

IoT used in Many areas. Some important applications are

- City Automation
- Automotive Applications
- Home Automation
- Environment
- Energy
- Retail
- Logistics
- Agriculture
- Industry
- Health and Life Style



CITY AUTOMATION

IoT application in city automation focuses on improving the infrastructure of the city with automated decision making ability to make it as a smart city.

Some applications in this domain include but are not limited to the following:

- Traffic flow management system in combination with dynamic traffic light control
- Street light control
- Passenger information system for public transportation
- Passive surveillance
- Smart Parking
- Structural Health Monitoring
- Emergency Response (Fire, Gas, Water Leakage Detection)
- Smart Waste Management

- Smart Buildings
- Smart Environment

Generic city sensors include environmental sensors and activity sensors. Environmental sensors include:

- Thermal (Temperature of solid, liquid, gas)
- Hygrometric (Humidity)
- Anemometric (Wind speed, pressure, velocity)
- Sound (Noise, Sound level)
- Gas (Detect different gases)
- Particles (Smoke and dust in air)
- light, other EM spectrum (intensity of light, infrared, ultraviolet, x-ray, gamma ray)
- Seismic (Vibration)

Activity sensors include:
– pavement/roadway pressure
– vehicle and pedestrian detection
– parking space occupancy

AUTOMOTIVE APPLICATIONS

IoT/M2M automotive and transportation applications focus on safety, security, connected navigation, and other vehicle services such as, but not limited to, insurance or road pricing, emergency assistance, fleet management, electric car charging management, and traffic optimization.

A brief description of applications are as follows

bCall (breakdown call): A bCall sends the current vehicle position to a roadside assistance organization and initiates a voice call.

Stolen vehicle tracking (SVT): A basic application for automotive M2M communications is tracking of mobile assets—either for purposes of managing a fleet of vehicles or to determine the location of stolen property. The goal of a SVT system is to facilitate the recovery of a vehicle in case of theft.

Remote diagnostics: Remote diagnostic services can broadly be grouped into the following categories:

– Maintenance minder—when the vehicle reaches a certain mileage (e.g., 90% of the manufacturer's recommended service interval since the previous service), the TCU sends a message to the owner or the owner's named dealership, advising the owner (or the dealership) that the vehicle is due for service.

– Health check—Either on a periodic basis or triggered by a request from the owner, the TCU compiles the vehicle's general status using inbuilt diagnostic reporting functions and transmits a diagnostic report to the owner, the owner's preferred dealership, or to the vehicle manufacturer.

– Fault triggered—When a fault (a diagnostic trouble code [DTC]) is detected with one of the vehicle systems, this triggers the TCU to send the DTC code and any related information to the owner's preferred dealer, or to the vehicle manufacturer.

– Enhanced bCall—When a manual breakdown call is initiated by the owner, the TCU sends both position data and DTC status information to the roadside assistance service or to the vehicle manufacturer.

Fleet management: The fleet owner wishes to track the vehicles—that is, to know, over time, the location and velocity of each vehicle—in order to plan and optimize business operations.

Vehicle-to-infrastructure communications - vehicle to roadside applications are less welldeveloped; in this case, vehicles have embedded M2M devices that can interface withlocation-determination technology and can communicate via a mobile telecommunication network to an entity (server). This application assumes that vehicles have been deployed with M2M devices installed that are able to:

Interface with sensors on the vehicle that measure velocity, external impacts

Interface with devices that can detect position

Establish a link with a mobile telecommunication network using appropriate network access credentials, such as a USIM

Upload or download traffic and safety information to a traffic information server

Insurance services: Pay-as-you-drive (PAYD) schemes offer insurers the opportunity to reduce costs based on actual risk and provide more competitive products to the end-user based on getting feedback from the vehicle as to when, where, how, or how far the vehicle is being driven.

HOME AUTOMATION

Home automation has received a lot of attention of late in the IoT/M2M context. Basic applications of the automated home include remote media control, heating control, lighting control (including low power landscape lighting control), and appliance control. Sensed homes, as examples of smart space, are seen as "next-step/nextgeneration" applications. Smart meters and energy efficiency (making use of the potential of SG), discussed above, also fit this category.Telehealth (e.g., assisted living and in-home m-health services) also can be captured under this set of applications; security and emergency services also can be included here.

M2M communications is expected to play a major role in residences, where instrumentation of elements supporting daily living (e.g., appliances), comfort, health, security, and energy efficiency can improve the quality of life and the quality of experience. Home control applications include but are not limited to:

Lighting control

Thermostat/HVAC

White goods/

Appliance control

In-home displays

Home security applications include but are not limited to:

- Door access phone
- Window locks
- Motion detector
- Smoke/ Gas / Fire alert
- Baby monitors
- Medical pendant

Energy efficiency at home is a key application of interest because of the possibility of monetary saving for the consumer. Occupancy sensors can be used to establish whether there is somebody in a room or not and when the room becomes unoccupied the lights are automatically switched off; other types of sensors can be used to control the energy consumption from different equipments (e.g., temperature, TVs, and so on).

The sensors and actuators can be autonomous (as in the case of light sensors), or can beconnected to anM2Mgateway control node (wirelessly or using wires, e.g., via PLC). The M2M system allows reducing energy consumption by automatically adapting the use of the house equipment to various short-term situations (people moving in and out of rooms, people going to work and retuning later) or long-term situations.

IoT levels and Deployment Templates

IoT system comprises of the following components:

Device : An IoT device allow identification, remote sensing, actuating and remote monitoring capabilities.

Resources : Resources are software components on the device for accessing and storing information for controlling actuator connected to the device also include software components that enable network access for the device .

Controller service: Controller Service is a native service that runs on the device and interact with the web services. Controller service sends data from the device to the web service receive command from the application from controlling the device.

Database: Database can be either local or in the cloud and stores the data generated by the IoT device.

Web service: Serve as a link between the device, application database and analysis components. Web Services can be implemented using HTTP and REST principles or using website protocol.

Stateless/stateful: Rest services stateless in nature. Each request contain all the information needed to process it. Request are independent of each other. Website on the other hand is stateful in nature where the server maintains the state and is aware of all the open connections.

Directional / Bi- directional: REST service operate over http and unidirectional. Request is

always sent by a client and the server response to the request. And other hand website is a bi directional product server to send message to each other.

Request response / full duplex: REST service follower request response Communicationsmodel where the client sends request and the server response to the request. Website and the other hand Allow full-duplex Communications between the client and server, it means both client and server can send messages to can independently.

TCP connections: For REST Service each http request involves setting up in a new TCP connections Websocket on the other hand involves a single TCP connection over which the client and server communicate in a full duplex mode.

Headache Overhead: REST service operate over http , and each request is independent of others . Thus each request carries http header which is an overhead. Due to the overhead of http headers, REST is not suitable for real time applications left hand does not involve overhead of headers. After the initial handshake the client and server exchange messages with minimal frame information.

Scalability: Scalability is easier in this case of the REST services of request are independent And no state information needs to be maintained by the server. Thus both horizontal out and vertical scaling solutions are possible for REST services.

Analysis component: The analysis component is responsible for analyzing the IoT data and generate results in the form which are easy for the user to understand. Analysis of IoT
data can be performed either locally or in the cloud. Analyzed results are stored in the local or cloud database.

Application: IoT applications provide an interface that the user can use to control and monitor various aspects of the IoT system. Applications also allow user to view the system status and view the processed data.

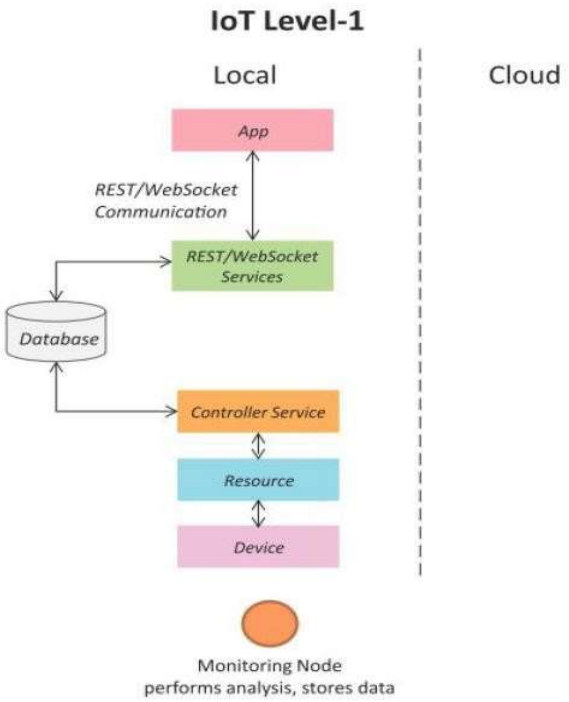
IOT LEVELS

IoT level 1:

Level One IoT system has a single node / device that performs sensing and/or actuation, stores data, reforms analysis and the host to the application. Level 1 IoT systems are suitable for modeling low cost and low complexity solutions where the data involving is not big and the analysis requirements are not computationally intensive.

Example

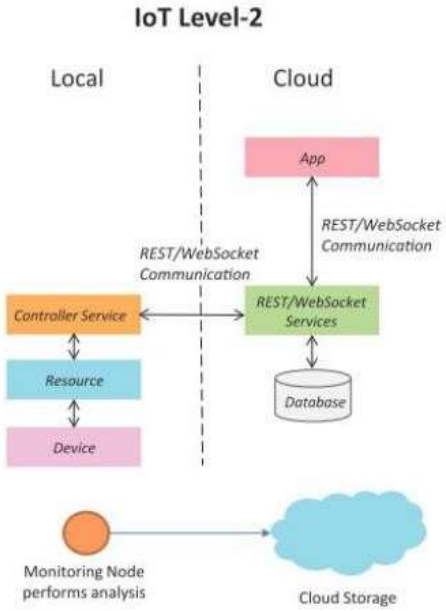
Let us now consider done example of Level 1 IoT system for home automation. This system consists of the single node that allows controlling the lights and appliances in your homeremotely . The device used in this system interface with their lights and appliances usingelectronic relay switches.



IoT level 2:

Level 2 IoT system has a single node that performs sensing and/or actuation and local analysis. Data is stored in the cloud and application is usually cloud based systems are suitable for solutions where the data in world is big, however the primary analysis requirement is notcomputationally intensive and can be done local itself.

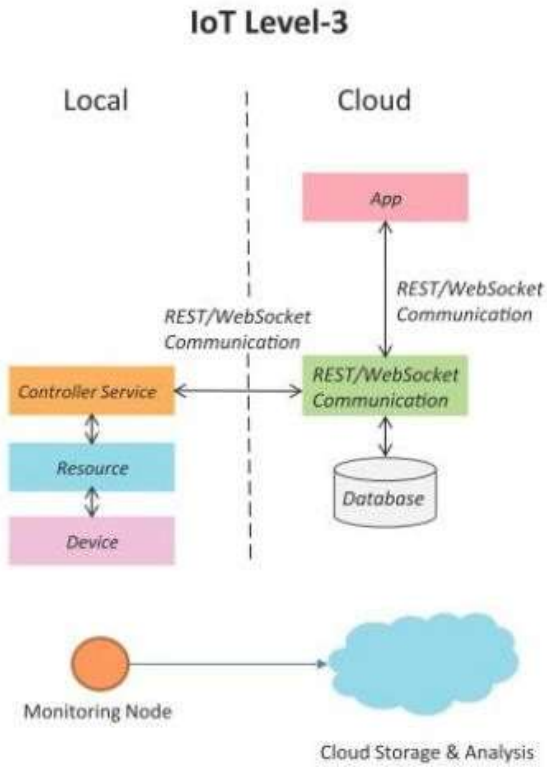
Example of Level 2 IoT system for smart irrigation.

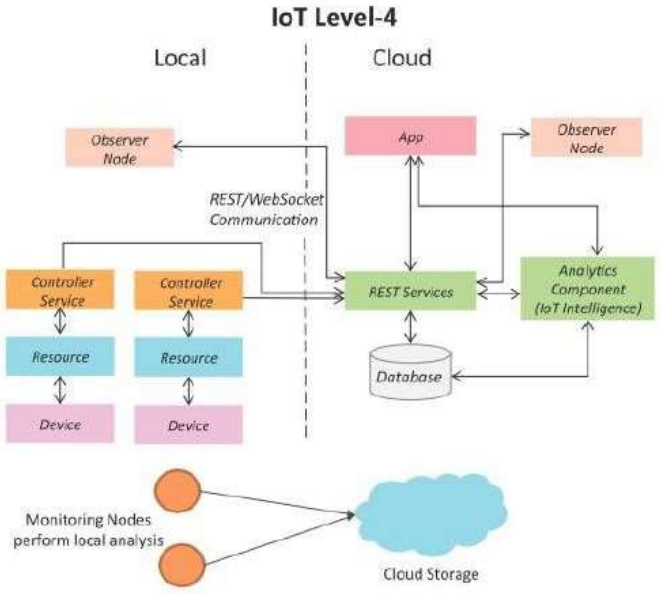


IoT Level 3:

Level 3 system has a single node . Data is stored and analyzed in the cloud application is cloud-based. Level 3 IoT system suitable for solutions where the data involved is big and analysis requirements computationally intensive.

Example of Level 3 IoT system tracking package handling



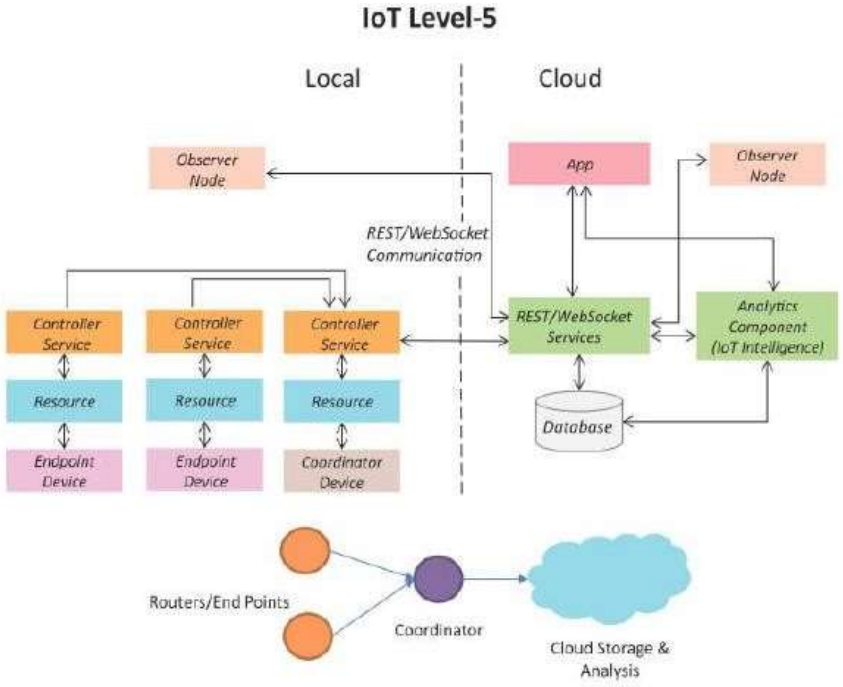


IoT level 4

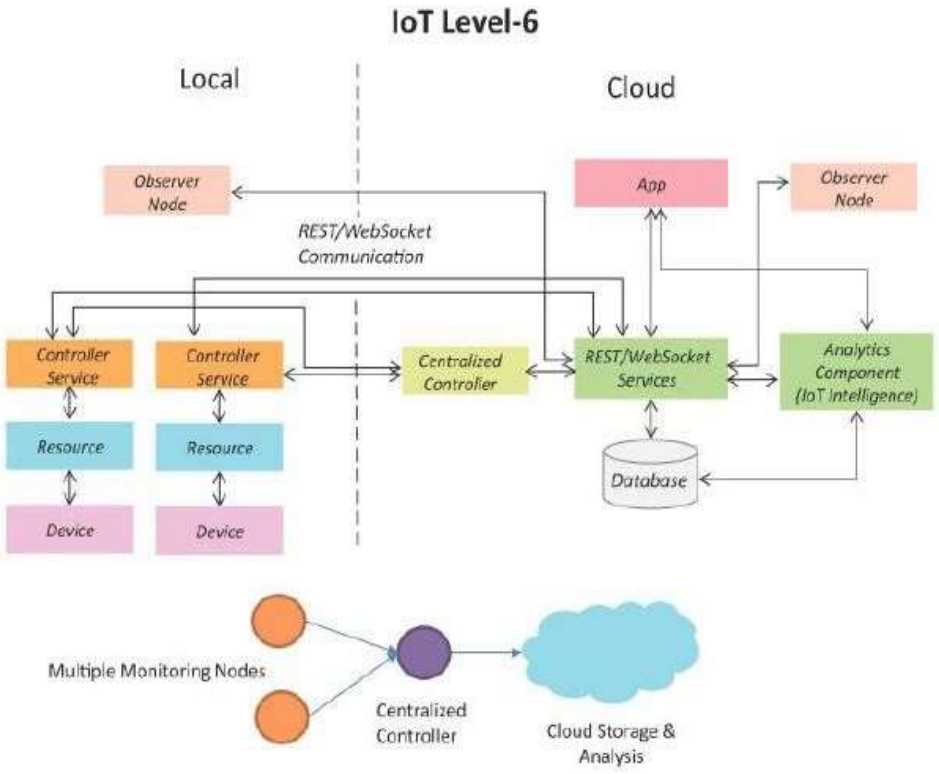
A level 4 IoT system has multiple notes that perform local analysis. Data is stored in the cloud and application is cloud based, level 4 contains local and cloud based observer notes which can

subscribe to and receive information collected in the cloud from IoT devices. Observer node can process information and use it for various applications, however observer notes do not perform any control function. level 4 IoT systems are suitable for solutions where multiple nodes are required the data involved is big and the analysis requirements are computationally intensive.
Example of level four IoT system for noise monitoring.

IoT Level 5:
IoT system has multiple end nodes and one coordinator notes and notes that perform sensing and / or actuation. Coordinator node collects data from the entry and send to the cloud. Data is stored and analyzed in the cloud and applications is cloud based.
Level 5 IoT system are suitable for forest fire detection. The system consists of multiple nodes placed in different locations for monitoring temperature, humidity and carbon dioxide levels in a forest. The endnotes in this example are equipped with various sensors such as temperature humidity and to CO2. The coordinator node collects the data from the end nodes and act as a Gateway that provides internet connectivity to the IoT system. The controller service on the coordinator device sends the collected data to the cloud .The data is stored in the cloud database. The analysis of the data is done in the computing cloud to aggregate the data and make prediction.



IoT Level 6:
IoT Level 6 system has multiple Independent and nodes that perform sensing and / or actuations and send data to the cloud. Data is stored in the cloud and applications is cloud based . The analytics component analyze the data and store the results in the cloud database. The results are visualized with the cloud based application. The centralized controller is aware of the status ofall the end notes and send control commands to the notes.
Example of the level 6 IoT system for weather monitoring

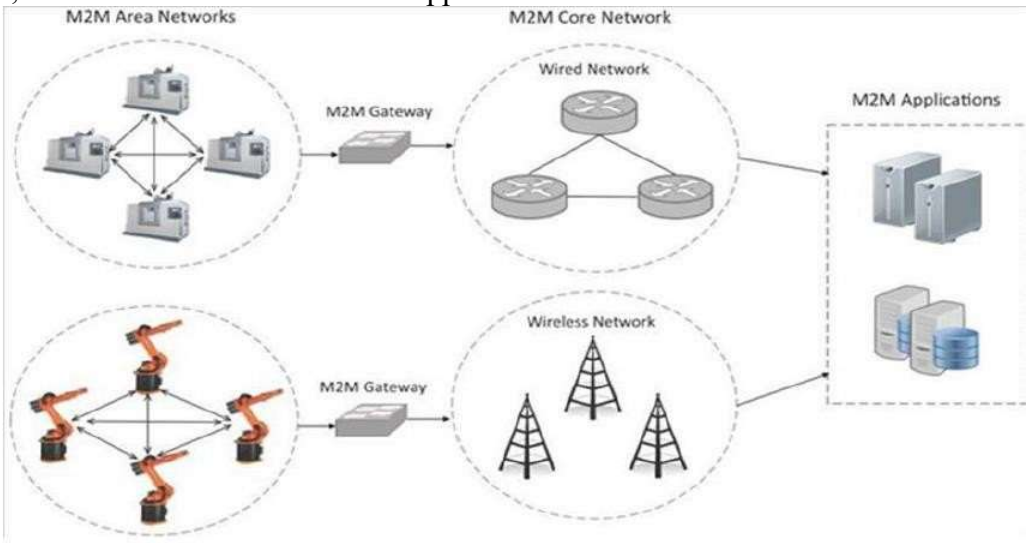


IoT and M2M

Machine-to-Machine (M2M) refers to networking of machines(or devices) for the purpose of remote monitoring and control and data exchange. The end-to-end architecture for M2M systems comprising of M2M area networks, Communications Network and application domain.

- Term which is often synonymous with IoT is Machine-to-Machine (M2M).
- IoT and M2M are often used interchangeably.

Below Figure Shows the end-to-end architecture of M2M systems comprises of M2M area networks, communication networks and application domain.



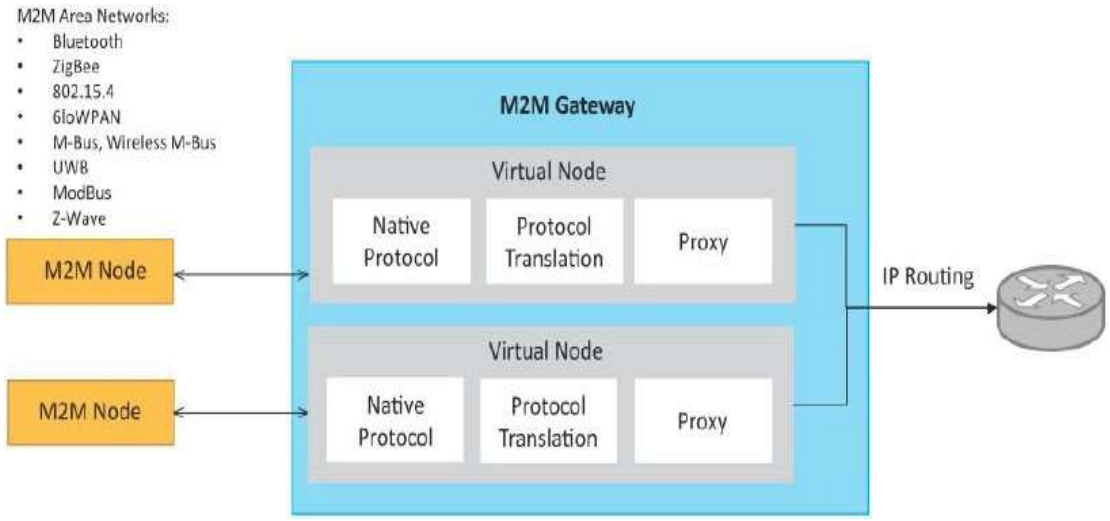
An **M2M area network** comprises of machines(or M2M nodes) which have embedded network modules for sensing, actuation and communicating various communication protocols can be used for M2M LAN such as ZigBee, Bluetooth, M-bus, Wireless M-Bus, 6LoWPAN etc., These protocols provide connectivity between M2M nodes within an M2M area network.

The **communication network** provides connectivity to remote M2M area networks. The communication network can use either wired or wireless network(IP based). While the M2M are networks use either properietorary or non-IP based communication protocols, the communication network uses IP-based network. Since non-IP based protocols are used within M2M area network, the M2M nodes within one network cannot communicate with nodes in an external network.

To enable the communication between remote M2M area network, **M2M gateways** are used.

The communication between M2M nodes and the M2M gateway is based on the communication protocols which are naive to the M2M are network.

M2M gateway performs protocol translations to enable Ip-connectivity for M2M are networks. M2M gateway acts as a proxy performing translations from/to native protocols to/from Internet Protocol(IP). With an M2M gateway, each mode in an M2M area network appears as a virtualized node for external M2M area networks.



Differences between IoT and M2M
1) Communication Protocols:
M2M and IoT can differ in how the communication between the machines and device happens. M2M uses other proprietary or not IP based communication protocol for communication with in the M2M area networks.
Commonly used M2M protocols include ZigBee, Bluetooth, ModBus, M-Bus, Wireless M-Bus.

The focus of communication in M2M is usually on the protocols below the network layer. Focus of communication in IoT is usually a protocol in network layer

IoT uses HTTP, CoAP, WebSocket , MQTT ,XMPP ,DDS ,AMQP etc.,

2) Machines in M2M Vs Things in IoT:
 Machines in M2M will be homogenous whereas Things in IoT will be heterogeneous.
The " things " IoT refers to Physical objects that have unique identifier and can sense and communicate with the external environment or their internal physical status. The unique identifiers the things in IoT are the IP addresses.

Things have software component for accessing processing and storing sensor information on controlling actuator connector. IoT system can include IoT devices of various types such as fire alarms , door alarms, lighting control devices.

3) Hardware Vs Software Emphasis:
 the emphasis of M2M is more on hardware with embedded modules, the emphasis of IoT is more on software.

4) Data Collection &Analysis
 M2M data is collected in point solutions and often in on-premises storage infrastructure.
 The data in IoT is collected in the cloud (can be public, private or hybrid cloud).

5) Applications
 M2M data is collected in point solutions and can be accessed by on-premises applications such as diagnosis applications, service management applications, and on- premisis enterprise applications.
 IoT data is collected in the cloud and can be accessed by cloud applications such as analytics applications, enterprise applications, remote diagnosis and management applications, etc.