INTRODUCTION OF IOT

IoT comprises things that have unique identities and are connected to internet. By 2020 there will be a total of 50 billion devices /things connected to internet.

IoT is not limited to just connecting things to the internet but also allow things to communicate and exchange data.

Definition: A dynamic global n/w infrastructure with self configuring capabilities based on standard and interoperable communication protocols where physical and virtual ―things have identities, physical attributes and virtual personalities and use intelligent interfaces, and are seamlessly integrated into information n/w, often communicate data associated with users and their environments

Characteristics:

1) **Dynamic & Self Adapting:** IoT devices and systems may have the capability to dynamically adapt with the changing contexts and take actions based on their operating conditions, user‘s context or sensedenvironment. Eg: the surveillance system is adapting itself based on context and changing conditions.

2) **Self Configuring:** allowing a large number of devices to work together to provide certain functionality.

3) **Inter Operable Communication Protocols:** support a number of interoperable communication protocols ans can communicate with other devices and also with infrastructure.

4) **Unique Identity:** Each IoT device has a unique identity and a unique identifier(IP address).

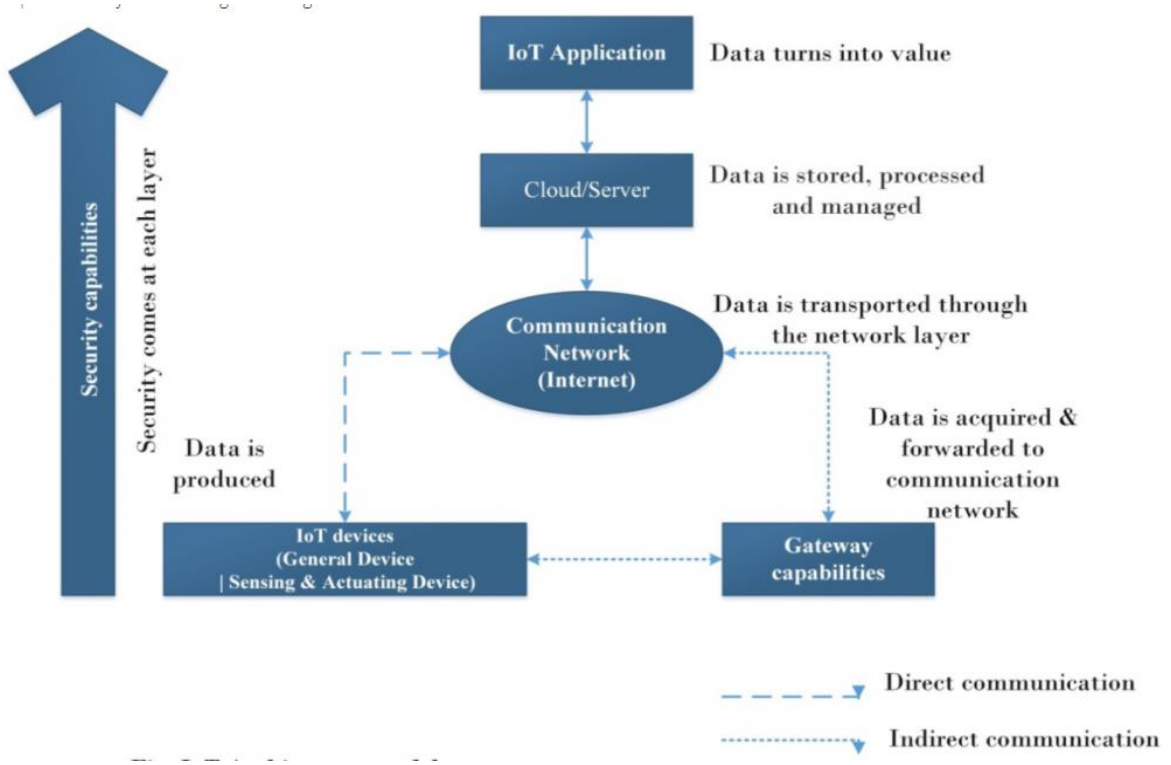
5) **Integrated into Information Network:** that allow them to communicate and exchange data with other devices andsystems.

Need of IoT

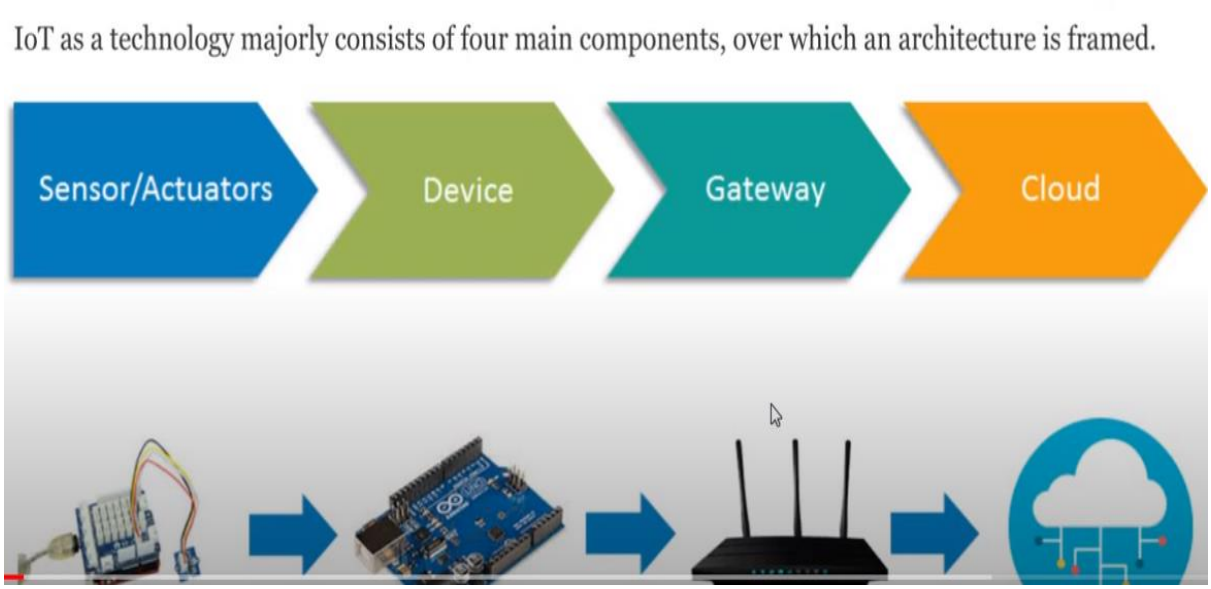
- In a nutshell IoT wants to connect all potential objects to interact each other on the internet to provide secure, comfort life for human .
- More data means better decisions
- Ability to track and monitor things
- Lighten the workload with automation
- Increases efficiency by saving money and resources

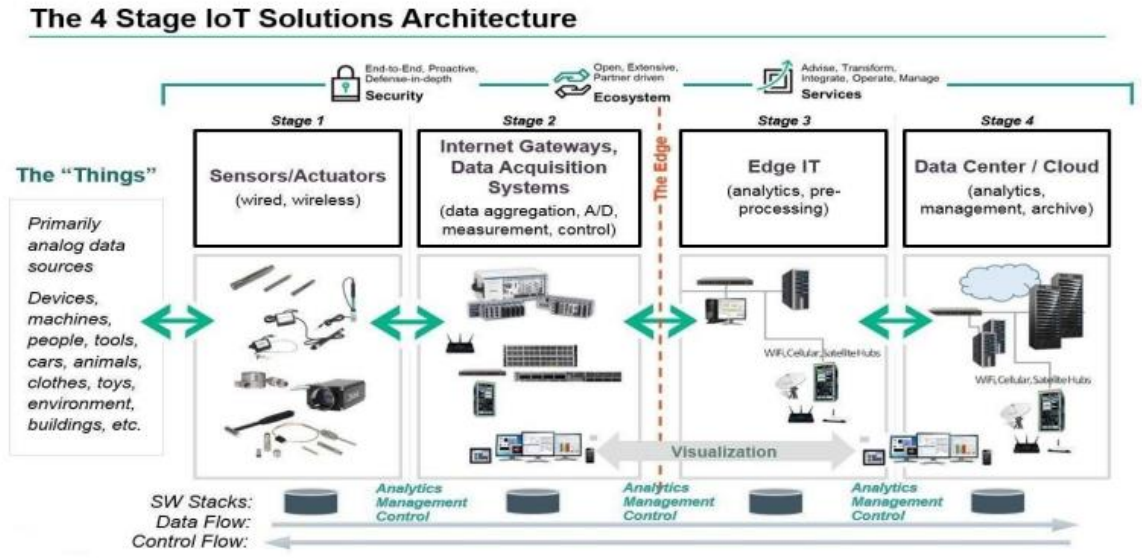
- Better quality of life
- Environmental Monitoring:
- Infrastructure management
- Industrial applications
- Building and home automation
- Transport system

Technical building blocks of IOT



IOT Architecture





Stage 1:- Sensors/actuators

Sensors collect data from the environment or object under measurement and turn it into useful data. Think of the specialized structures in your cell phone that detect the directional pull of gravity and the phone's relative position to the —thing‖ we call the earth and convert it into data that your phone can use to orient the device.

Actuators can also intervene to change the physical conditions that generate the data. An actuator might, for example, shut off a power supply, adjust an air flow valve, or move a robotic gripper in an assembly process.

The sensing/actuating stage covers everything from legacy industrial devices to robotic camera systems, water level detectors, air quality sensors, accelerometers, and heart rate monitors. And the scope of the IoT is expanding rapidly, thanks in part to low-power wireless sensor network technologies and Power over Ethernet, which enable devices on a wired LAN to operate without the need for an A/C power source

Stage 2:- The Internet gateway

The data from the sensors starts in analog form. That data needs to be aggregated and converted into digital streams for further processing downstream. Data acquisition systems (DAS) perform these data aggregation and conversion functions. The DAS connects to the sensor network, aggregates outputs, and performs the analog-to-digital conversion. The Internet gateway receives the aggregated and digitized data and routes it over Wi-Fi, wired LANs, or the Internet, to Stage 3 systems for further processing. Stage 2 systems often sit in close proximity to the sensors and actuators.

For example, a pump might contain a half-dozen sensors and actuators that feed data into a data aggregation device that also digitizes the data. This device might be physically attached to the pump. An adjacent gateway device or server would then process the data and forward it to the Stage 3 or Stage 4 systems. Intelligent gateways can build on additional, basic gateway functionality by adding such capabilities as analytics, malware protection, and data management services. These systems enable the analysis of data streams in real time.

Stage 3:- Edge IT

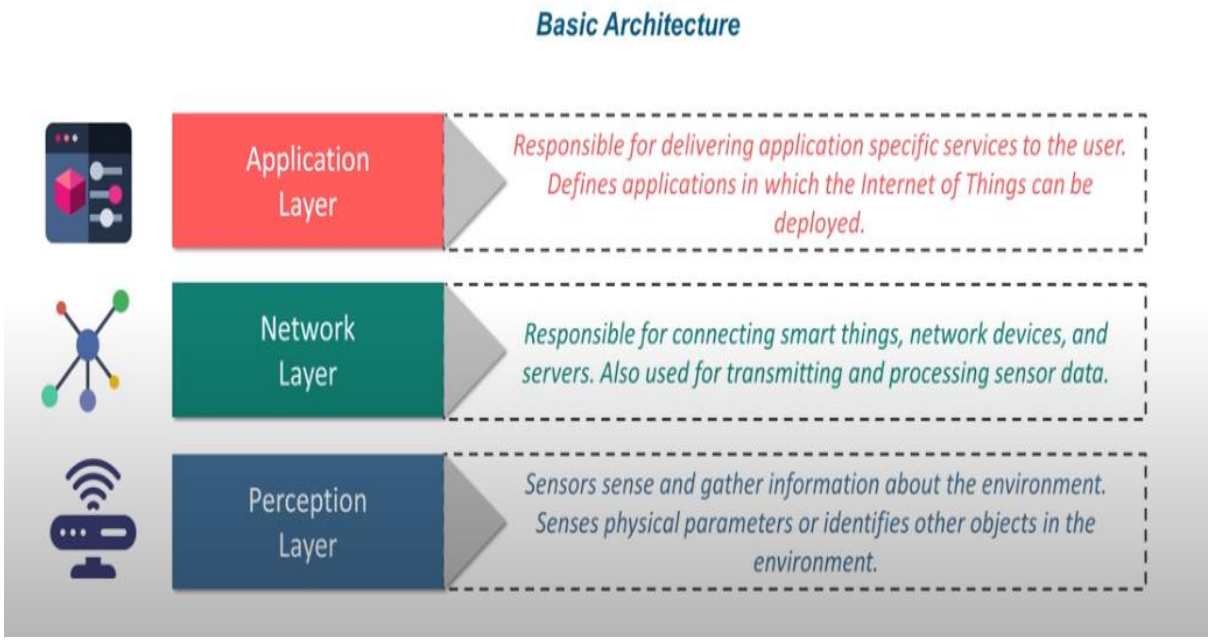
Once IoT data has been digitized and aggregated, it's ready to cross into the realm of IT. However, the data may require further processing before it enters the data center. This is where edge IT systems, which perform more analysis, come into play. Edge IT processing systems may be located in remote offices or other edge locations, but generally these sit in the facility or location where the sensors reside closer to the sensors, such as in a wiring closet. Because IoT data can easily eat up network bandwidth and swamp your data center resources, it's best to have systems at the edge capable of performing analytics as a way to lessen the burden on core IT infrastructure. You'd also face security concerns, storage issues, and delays processing the data. With a staged approach, you can preprocess the data, generate meaningful results, and pass only those on. For example, rather than passing on raw vibration data for the pumps, you could aggregate and convert the data, analyse it, and send only projections as to when each device will fail or need service

Stage 4:- The data center and cloud

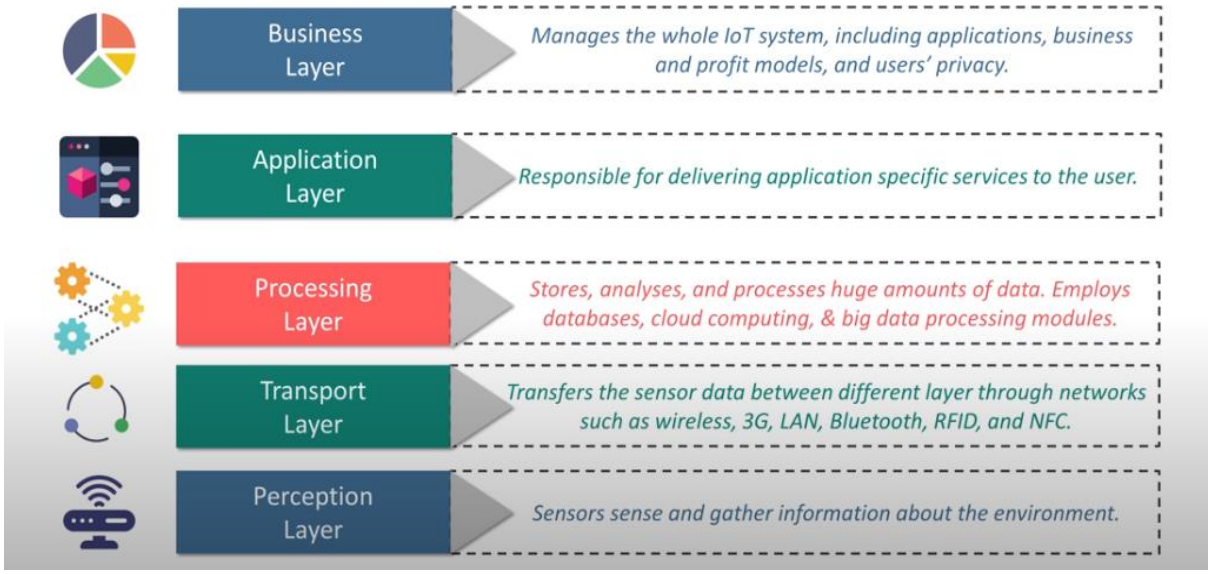
Data that needs more in-depth processing, and where feedback doesn't have to be immediate, gets forwarded to physical data center or cloud-based systems, where more powerful IT systems can analyze, manage, and securely store the data. It takes longer to get results when you wait until data reaches Stage 4, but you can execute a more in-depth analysis, as well as

combine your sensor data with data from other sources for deeper insights. Stage 4 processing may take place on-premises, in the cloud, or in a hybrid cloud system, but the type of processing executed in this stage remains the same, regardless of the platform.

Basic Architecture

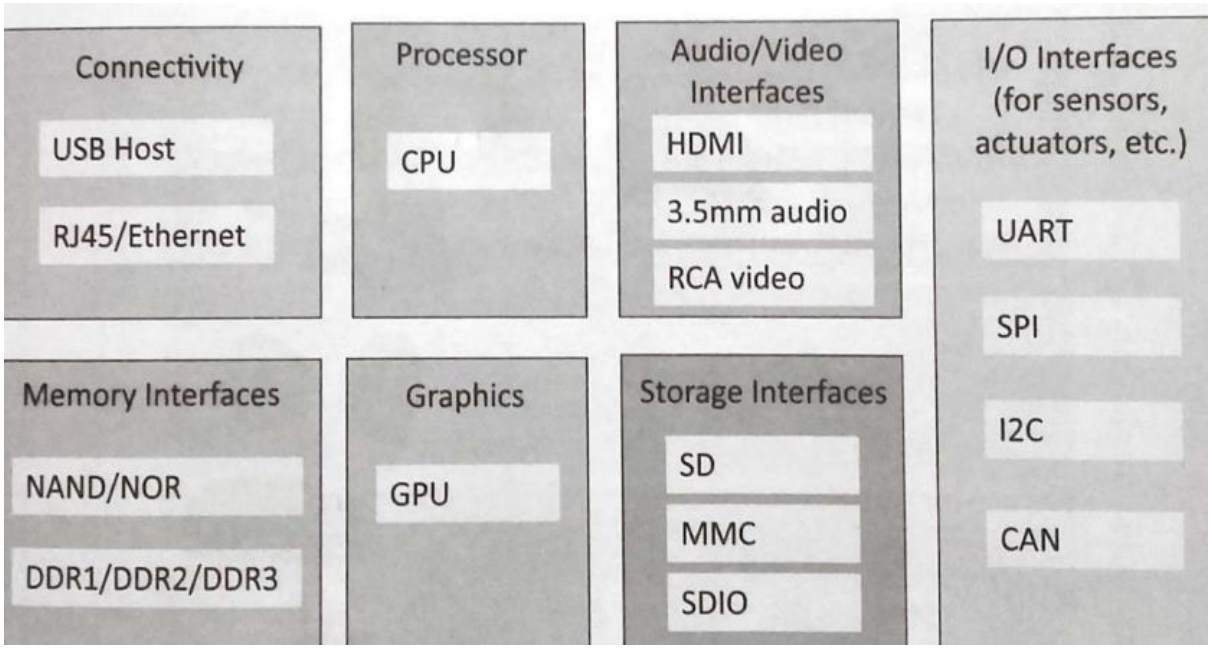


5 Layer Architecture



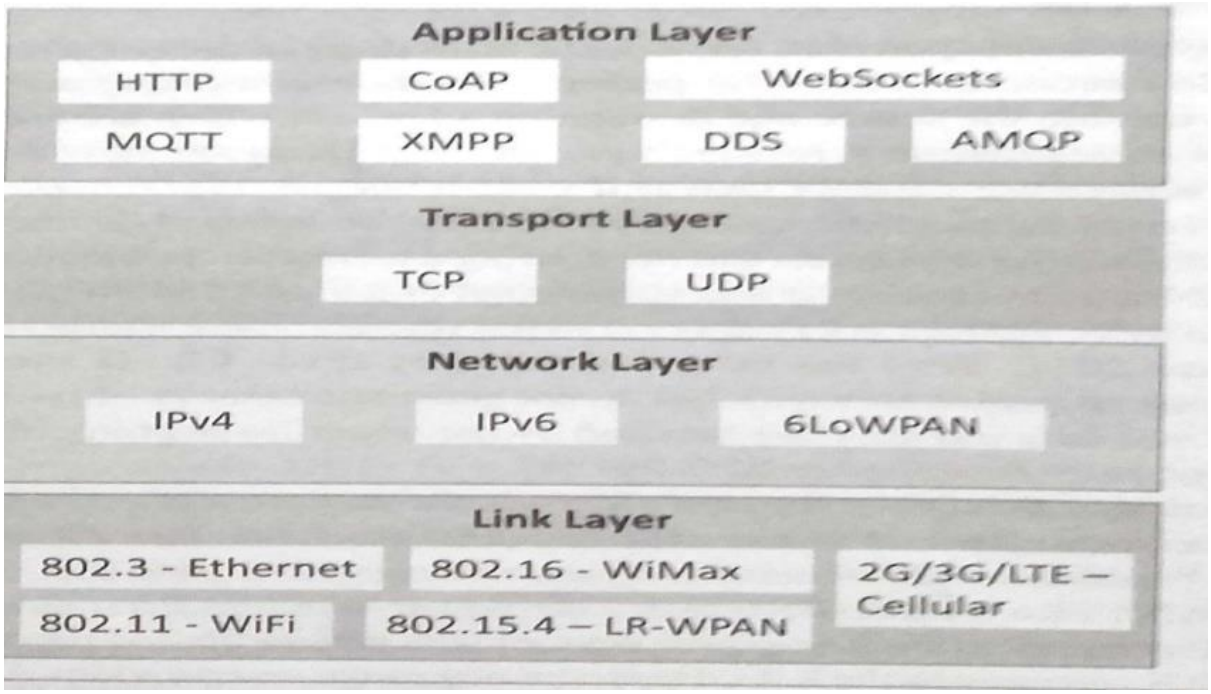
Physical design of IOT

Things in IoT:



The things in IoT refers to IoT devices which have unique identities and perform remote sensing, actuating and monitoring capabilities.

IoT devices can exchange data with other connected devices applications. It collects data from other devices and process data either locally or remotely. An IoT device may consist of several interfaces for communication to other devices both wired and wireless. These includes (i) I/O interfaces for sensors, (ii) Interfaces for internet connectivity (iii) memory and storage interfaces and (iv) audio/video interfaces



IoT Protocols:

a) Link Layer : Protocols determine how data is physically sent over the network‘s physical layer or medium. Local network connect to which host is attached. Hosts on the same link exchange data packets over the link layer using link layer protocols. Link layer determines how packets are coded and signalled by the h/w device over the medium to which the host is attached

Protocols:

- 802.3-Ethernet: IEEE802.3 is collection of wired Ethernet standards for the link layer. Eg: 802.3 uses co-axial cable; 802.3i uses copper twisted pair connection; 802.3j uses fiber optic connection; 802.3ae uses Ethernet overfiber.
- 802.11-WiFi: IEEE802.11 is a collection of wireless LAN(WLAN) communication standards including extensive description of link layer. Eg: 802.11a operates in 5GHz band, 802.11b and 802.11g operates in 2.4GHz band, 802.11n operates in 2.4/5GHz band, 802.11ac operates in 5GHz band, 802.11ad operates in 60Ghzband.
- 802.16 - WiMax: IEEE802.16 is a collection of wireless broadband standards including exclusive description of link layer. WiMax provide data rates from 1.5 Mb/s to 1Gb/s.
- 802.15.4-LR-WPAN: IEEE802.15.4 is a collection of standards for low rate wireless personal area network(LR-WPAN). Basis for high level communication protocols such as ZigBee. Provides data rate from 40kb/s to250kb/s.
- 2G/3G/4G-Mobile Communication: Data rates from 9.6kb/s(2G) to up to100Mb/s(4G)

B) Network/Internet Layer:

Responsible for sending IP datagrams from source n/w to destination n/w. Performs the host addressing and packet routing. Datagrams contains source and destination address

6LOWPAN:(IPv6overLowpowerWirelessPersonalAreaNetwork) operates in 2.4 GHz frequency range and data transfer 250 kb/s.

Protocols:

- IPv4: Internet Protocol version4 is used to identify the devices on a n/w using a hierarchical addressing scheme. 32 bit address. Allows total of 2**32addresses.
- IPv6: Internet Protocol version6 uses 128 bit address scheme and allows 2**128 addresses

C) Transport Layer:

Provides end-to-end message transfer capability independent of the underlying n/w. Set up on connection with ACK as in TCP and without ACK as in UDP. Provides functions such as error control, segmentation, flow control and congestion control.

Protocols:

- TCP: Transmission Control Protocol used by web browsers(along with HTTP and HTTPS), email(along with SMTP, FTP). Connection oriented and stateless protocol. IP Protocol deals with sending packets, TCP ensures reliable transmission of protocols in order. Avoids n/w congestion and congestion collapse.
- UDP: User Datagram Protocol is connectionless protocol. Useful in time sensitive applications, very small data units to exchange. Transaction oriented and stateless protocol. Does not provide guaranteed delivery

D) Application Layer:

Defines how the applications interface with lower layer protocols to send data over the n/w. Enables process-to-process communication using ports.

Protocols:

- HTTP: Hyper Text Transfer Protocol that forms foundation of WWW. Follow reques tresponse model Stateless protocol.

- CoAP: Constrained Application Protocol for machine-to-machine(M2M) applications with constrained devices, constrained environment and constrained n/w. Uses client server architecture.

- WebSocket: allows full duplex communication over a single socket connection.

- MQTT: Message Queue Telemetry Transport is light weight messaging protocol based on publish-subscribe model. Uses client server architecture. Well suited for constrained environment.

- XMPP: Extensible Message and Presence Protocol for real time communication and streaming XML data between network entities. Support client-server and server-server communication.

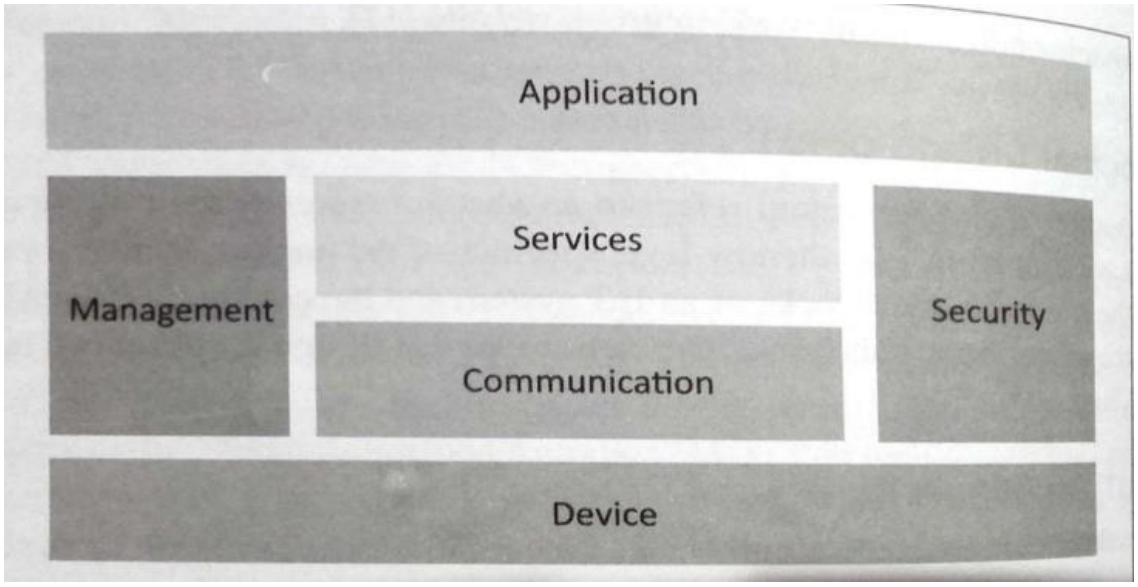
- DDS: Data Distribution Service is data centric middleware standards for device-to-device or machine-to-machine communication. Uses publish-subscribe model.

- AMQP: Advanced Message Queuing Protocol is open application layer protocol for business messaging. Supports both point-to-point and publish-subscribe model.

LOGICAL DESIGN of IoT

Refers to an abstract represent of entities and processes without going into the low level specifies of implementation. 1) IoT Functional Blocks 2) IoT Communication Models 3) IoT Comm. APIs

1) **IoT Functional Blocks**: Provide the system the capabilities for identification, sensing, actuation, communication and management



- Device: An IoT system comprises of devices that provide sensing, actuation, monitoring and control functions.

- Communication: handles the communication for IoT system.

- Services: for device monitoring, device control services, data publishing services and services for device discovery.

- Management: Provides various functions to govern the IoT system.

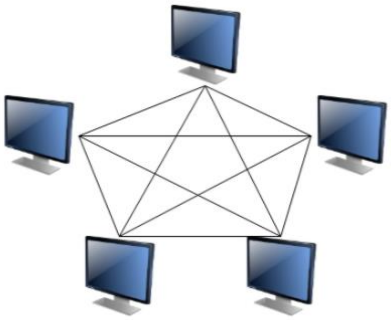
- Security: Secures IoT system and priority functions such as authentication, authorization, message and context integrity and data security.

- Application: IoT application provide an interface that the users can use to control and monitor various aspects of IoT system.

P2P (peer to peer)

A P2P (peer to peer) connection is a direct communication infrastructure between two peers:

- A client device (such as a smartphone or a laptop) and an IoT device (such as a surveillance camera, smart door lock, alarm system, heat controller, or anything else that can connect to the internet).
- The peer-to-peer computing architecture contains nodes that are equal participants in data sharing. All the tasks are equally divided between all the nodes. The nodes interact with each other as required to share resources.
- Here each computer acts as a node for file sharing within the formed network. Here each node acts as a server and thus there is no central server to the network.
- This allows the sharing of a huge amount of data. The tasks are equally divided amongst the nodes. Each node connected in the network shares an equal workload.
- For the network to stop working, all the nodes need to individually stop working. This is because each node works independently.



Characteristics of Peer to Peer Computing

The different characteristics of peer to peer networks are as follows −

- Peer to peer networks are usually formed by groups of a dozen or fewer computers. These computers all store their data using individual security but also share data with all the other nodes.
- The nodes in peer-to-peer networks both use resources and provide resources. So, if the nodes increase, then the resource sharing capacity of the peer-to-peer network increases. This is different than client-server networks where the server gets overwhelmed if the nodes increase.
- Since nodes in peer-to-peer networks act as both clients and servers, it is difficult to provide adequate security for the nodes. This can lead to denial of service attacks.
- Most modern operating systems such as Windows and Mac OS contain software to implement peer-to-peer networks.

Advantages and disadvantages of Peer to Peer Computing

Some advantages of peer to peer computing are as follows −

- Each computer in the peer-to-peer network manages itself. So, the network is quite easy to set up and maintain.
- In the client-server network, the server handles all the requests of the clients. This provision is not required in peer-to-peer computing and the cost of the server is saved.
- It is easy to scale the peer-to-peer network and add more nodes. This only increases the data-sharing capacity of the system.
- None of the nodes in the peer to peer network are dependent on the others for their functioning

Some disadvantages of peer to peer computing are as follows −

- It is difficult to back up the data as it is stored in different computer systems and there is no central server.
- It is difficult to provide overall security in the peer-to-peer network as each system is independent and contains its own data.

Machine-to-Machine(M2M)

M2M- Networking of machines(or devices) for the purpose of remote monitoring and control and data exchange.

Features:

- Large number of nodes or devices.
- Low cost.
- Energy efficient.
- Small traffic per machine/device.
- Large quantity of collective data.
- M2M communication free from human intervention.

Architecture of M2M consists of

- M2M area networks
- Communication network
- Application domains

M2M area networks

- Comprises of machines which are embedded hardware module for sensing, actuation and communication.
- Communication protocols provide connectivity between M2M nodes within an M2M area network.

Various communication protocols used for M2M local area network are

- Zigbee
- Bluetooth
- M- bus
- Wireless M- bus
- Power line communication(PLC)
- Modbus

Communication network

- Communication network can use either wired or wireless networks(IP based) while M2M area network use either proprietary or non-IP based Communication protocols.
- Since non-IP based protocols are used in 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.

M2M Gateway

- The communication between the M2M nodes and the M2M gateway is based on communication protocol which are native to the M2M area network.

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

M2M Applications

The M2M data is gathered into point solutions such as enterprise applications, service management applications, or remote monitoring applications.

M2M has various application domains such as

- Smart Metering
- Home Automation
- Industrial Automation
- Smart grids, etc.

Difference between M2M and IOT

M2M	IOT
It is about direct machine to machine communication	It is about sensor automation and internet platform
It supports point to point communication	It supports cloud based communication
Device not necessary relay on internet	Device necessary relay on internet
It is mostly based on hardware	It is based on both hardware and software
Machine normally communicates with single machine at a time	Many users can access at a time over internet
It uses either proprietary or non IP based protocols	It uses IP based protocols
Limited number of devices can be connected at a time	More number of devices can be connected at a time
It is less scalable	It is more scalable

IOT Framework

A high level M2Msystem architecture (HLSA) comprises of the device and gateway domain, the network domain, and the applications domain.

The **device and gateway domain** is composed of the following elements:

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.

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), 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

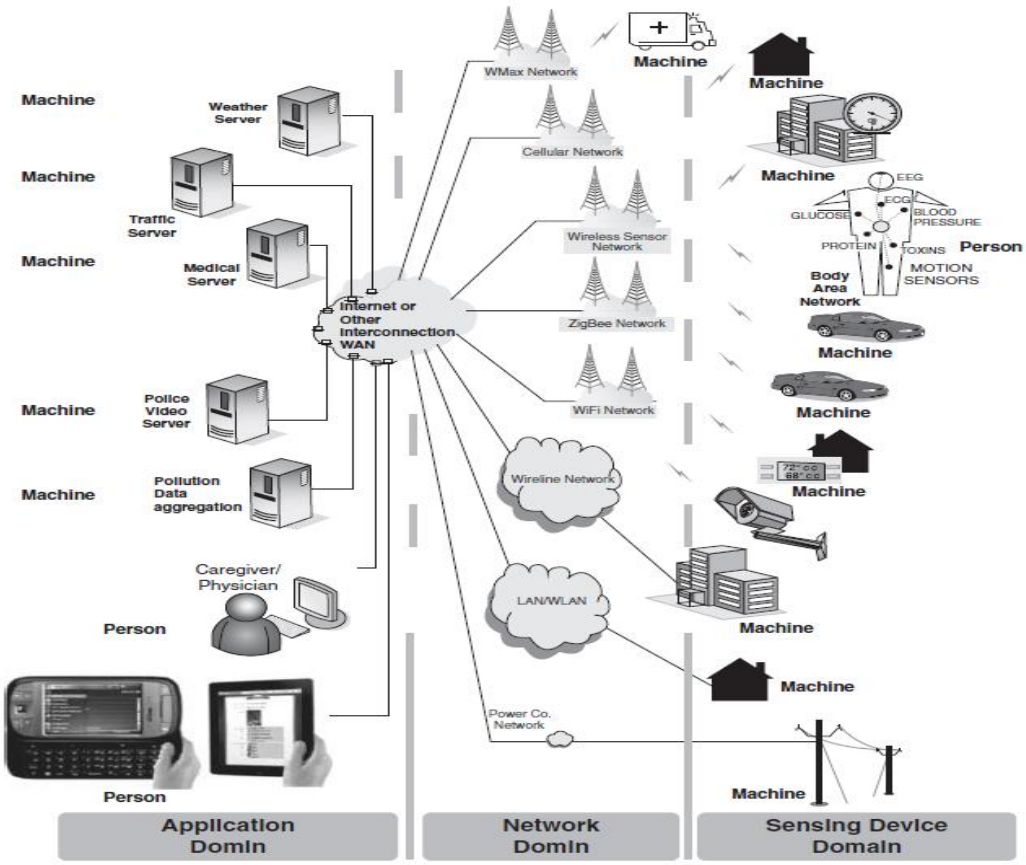
The **applications domain** is composed of the following elements:

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, as follows:

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. M2M management functions: Consists of all the functions required to manage M2Mservice capabilities in the network domain.



Challenges in IOT

Security challenges in IoT :

- Lack of encryption
- Insufficient testing and updating
- Brute forcing and the risk of default passwords
- IoT Malware and ransomware

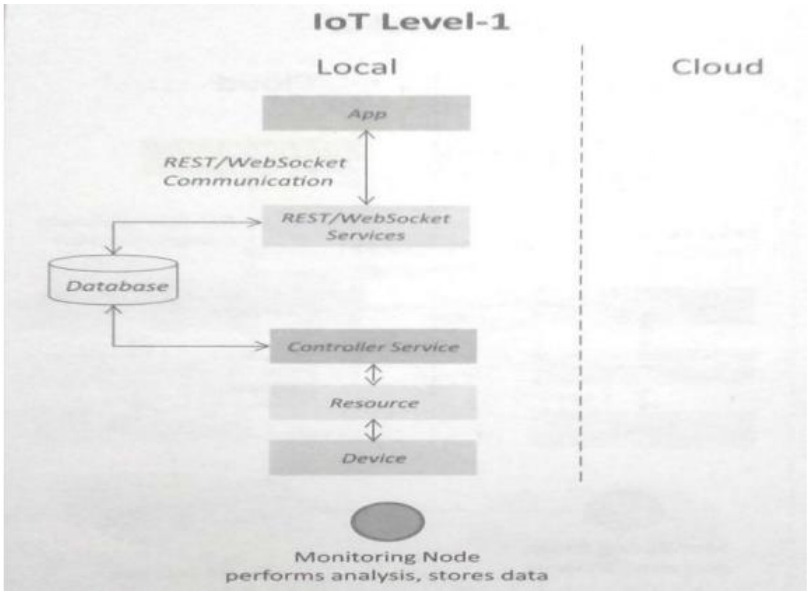
Design challenge in IoT

- Battery life is a limitation
- Increased cost and time to market
- Security of the system

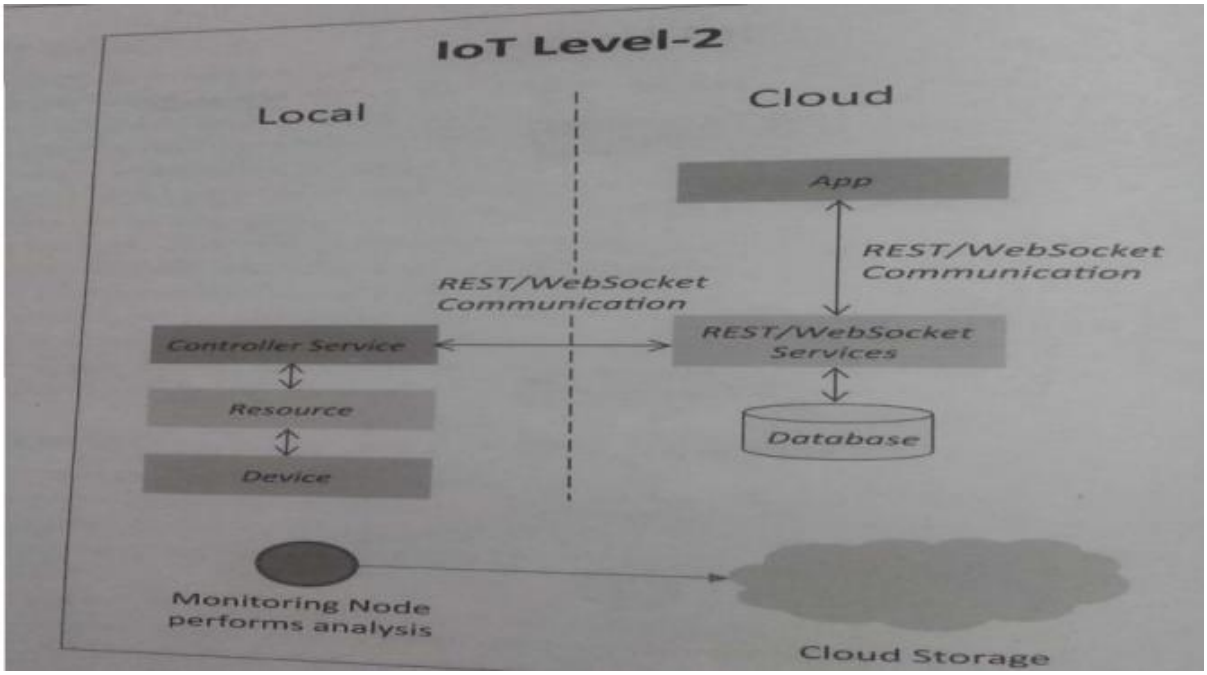
- Deployment challenges in IoT:
  - Connectivity
  - Cross platform capability
  - Data collection and processing
  - Lack of skillset
- Compatibility Challenges
- Bandwidth Challenges
- Customer Expectation Challenges

IoT Levels and Deployment Templates

1) **IoT Level 1**
   System has a single node that performs sensing and/or actuation, stores data, performs analysis and host the application as shown in fig. Suitable for modeling low cost and low complexity solutions where the data involved is not big and analysis requirement are not computationally intensive. An e.g., of IoT Level1 is Home automation.

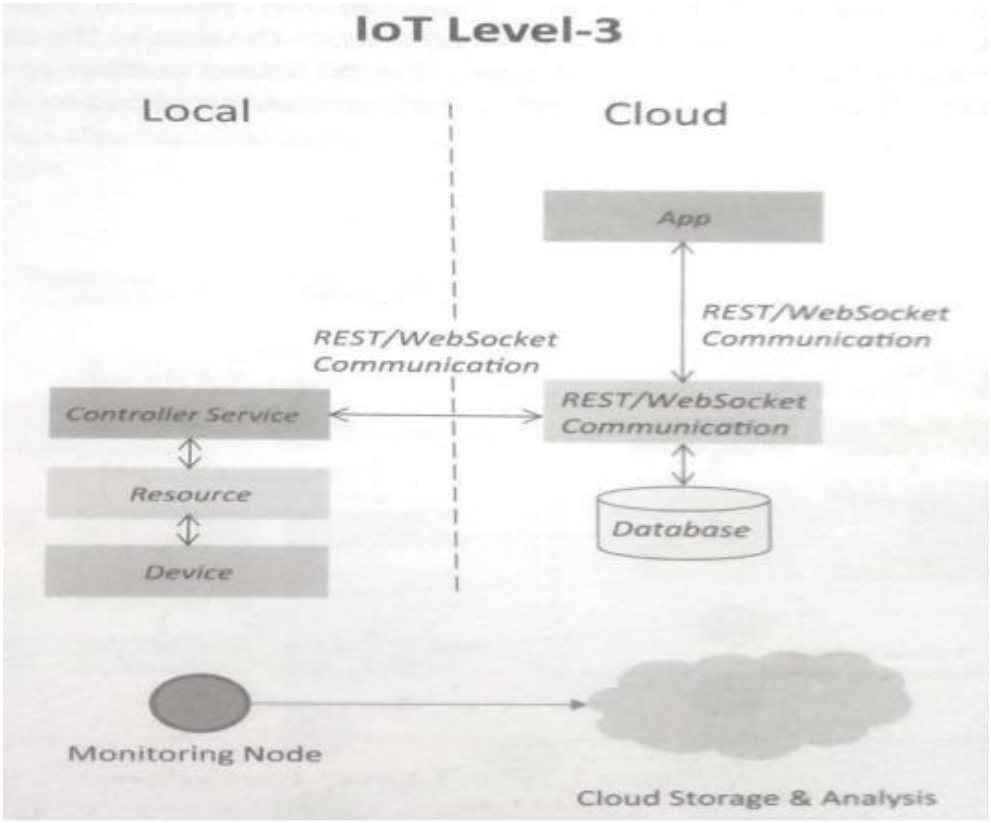


2) **IoT Level 2**
   It has a single node that performs sensing and/or actuating and local analysis as shown in fig. Data is stored in cloud and application is usually cloud based. Level2 IoT systems are suitable for solutions where data are involved is big, however, the primary analysis requirement is not computationally intensive and can be done locally itself. An e,g., of Level2 IoT system for SmartIrrigation.



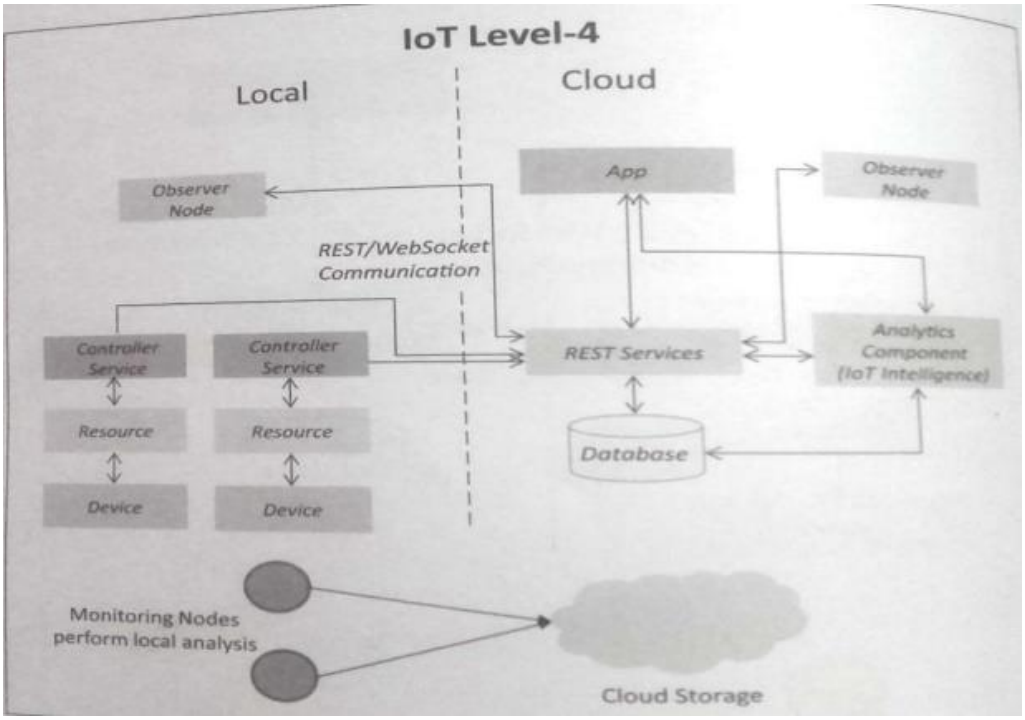
3) IoT Level 3

System has a single node. Data is stored and analyzed in the cloud application is cloud based as shown in fig. Level3 IoT systems are suitable for solutions where the data involved is big and analysis requirements are computationally intensive. An example of IoT level3 system for tracking packagehandling



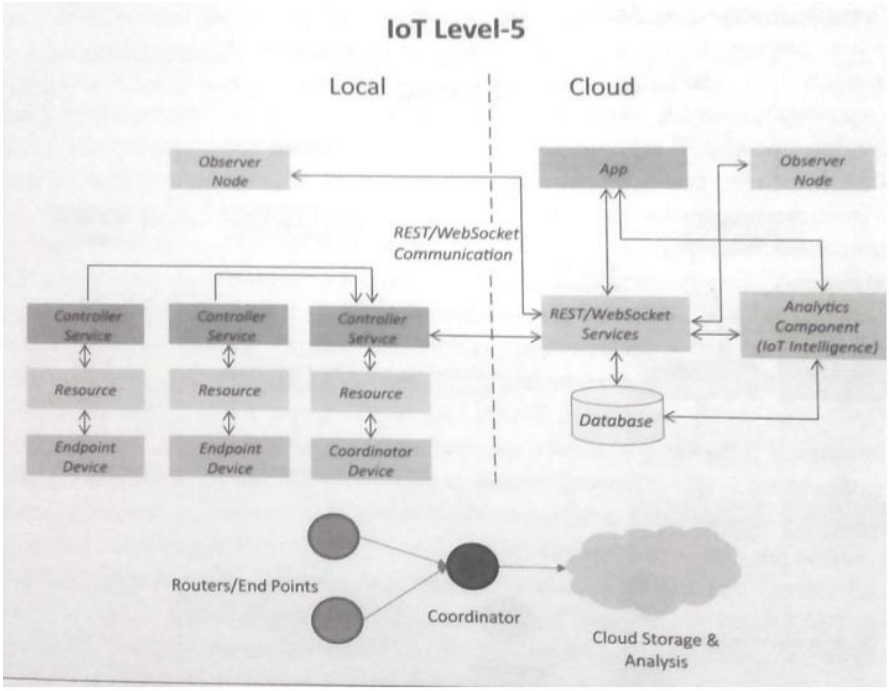
4) IoT Level 4

System has multiple nodes that perform local analysis. Data is stored in the cloud and application is cloud based as shown in fig. Level4 contains local and cloud based observer nodes which can subscribe to and receive information collected in the cloud from IoT devices. An example of a Level4 IoT system for NoiseMonitoring



5) IoT Level 5

System has multiple end nodes and one coordinator node as shown in fig. The end nodes that perform sensing and/or actuation. Coordinator node collects data from the end nodes and sends to the cloud. Data is stored and analyzed in the cloud and application is cloud based. Level5 IoT systems are suitable for solution based on wireless sensor network, in which data involved is big and analysis requirements are computationally intensive. An example of Level5 system for Forest Fire Detection.



6) IoT Level 6

System has multiple independent end nodes that perform sensing and/or actuation and sensed data to the cloud. Data is stored in the cloud and application is cloud based as shown in fig. The analytics component analyses the data and stores the result in the cloud data base. The results are visualized with cloud based application. The centralized controller is aware of the status of all the end nodes and sends control commands to nodes. An example of a Level6 IoT system for Weather Monitoring System