

Program: Master of Computer Applications

Curriculum Scheme: CBCGS

Examination: MCA First Year Semester II

Course Code: MCAE242 and **Course Name:** Internet of Things

Time: 2 HRS

Max. Marks: 80

Section I - MCQS (40 Marks) – 40 Minutes

Section II – Subjective (40 Marks) – 80 Minutes

The timings

If the Examination Time is 10:00 am to 12:00 noon

Section I – 10:00 am – 10:40 am

Section II – 10:40 am – 12:00noon

If the Examination Time is 2:00 pm to 4:00 pm

Section I – 2 :00 pm – 2:40 pm

Section II – 2 :40 pm to 4:00 pm

SECTION II

Q 2. Solve any Two questions out of three which carry 10 marks each respectively. 20 marks

- A. Explain the IoT Level 2 and IoT Level 3 Deployment templates.
- B. Explain ETSI and IETF state of art Architectures and Reference Models for IoT.
- C. Write short note on: IEEE-802.15.4 protocol and IoT vulnerability

Q 3. Solve any Two questions out of three which carry 10 marks each respectively. 20 marks

- A. Draw the schematic of Functional View of IoT reference architecture and explain various Functional Components (FC) of it.
- B. Use IoT design methodology steps and design Home automation system.
- C. Explain the Cloud of Things Architecture in brief.

12/8/21

Program: Master of Computer Applications

Curriculum Scheme: MCA 2 YEAR COURSE

Examination: MCA FH2021 SEMESTER II

Course Code: MCAE242 and **Course Name:** Internet of Things

Time:2 HRS Max. Marks:80

Section I - MCQS (40 Marks) – 40 Minutes

Section II – Subjective (40 Marks) – 80 Minutes

The timings

If the Examination Time is 10:00 am to 12:00 noon

Section I – 11:00 am – 11:40 am

Section II – 11:40 am – 1:00 pm

SECTION II

Q2. Solve any two questions out of three.

20 marks

A. Explain with appropriate diagram the deployment of any 4 IOT Levels.

B. What is functional model of IOT? Describe the functionality of 5 functional groups of the functional model.

C. What are the different efforts that have been taken for IOT Protocol Standardization?

Q3. Solve any two questions out of three.

20 marks

A. What are the steps involved in IOT system design methodology?

B. Elaborate the application of IOT in the following domains: Cities, Retail and Logistics.

C. Explain the Cloud of things architecture with suitable diagram.



Chapter 1

Introduction to IoT

INTERNET OF THINGS

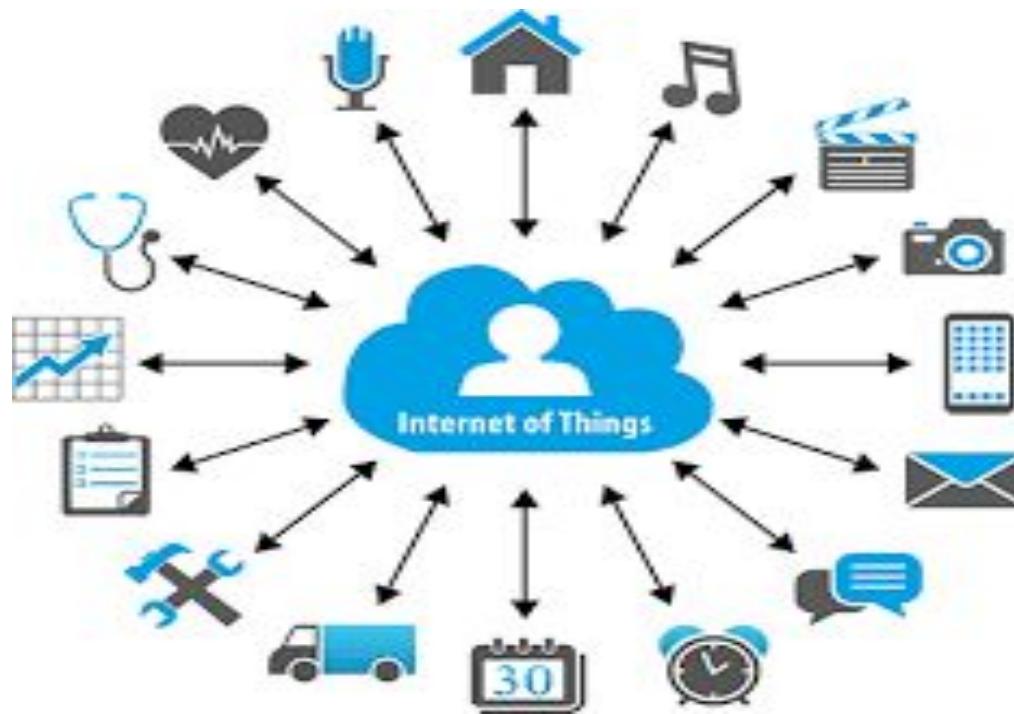
A Hands-On Approach



Outline

- Definition of IoT
- Characteristics of IoT
- Building Blocks of IoT
- Physical design of IoT
- Logical design of IoT
- IoT protocols
- IoT levels and deployment templates

IoT



IoT

- <https://www.youtube.com/watch?v=GIfWNtMfYvk&t=1s>
- <https://www.youtube.com/watch?v=Q3ur8wzzhBU>

IoT

- Internet Of Things is Fully Networked and Connected Devices sending analytics data back to cloud or data center.
- The definition of Internet of things is that it is the network in which every object or **thing is provided unique identifier** and data is transferred through a network without any verbal communication.
- **Scope of IoT is not just limited to just connecting things to the internet, but it allows these things to communicate and exchange data, process them as well as control them while executing applications.**

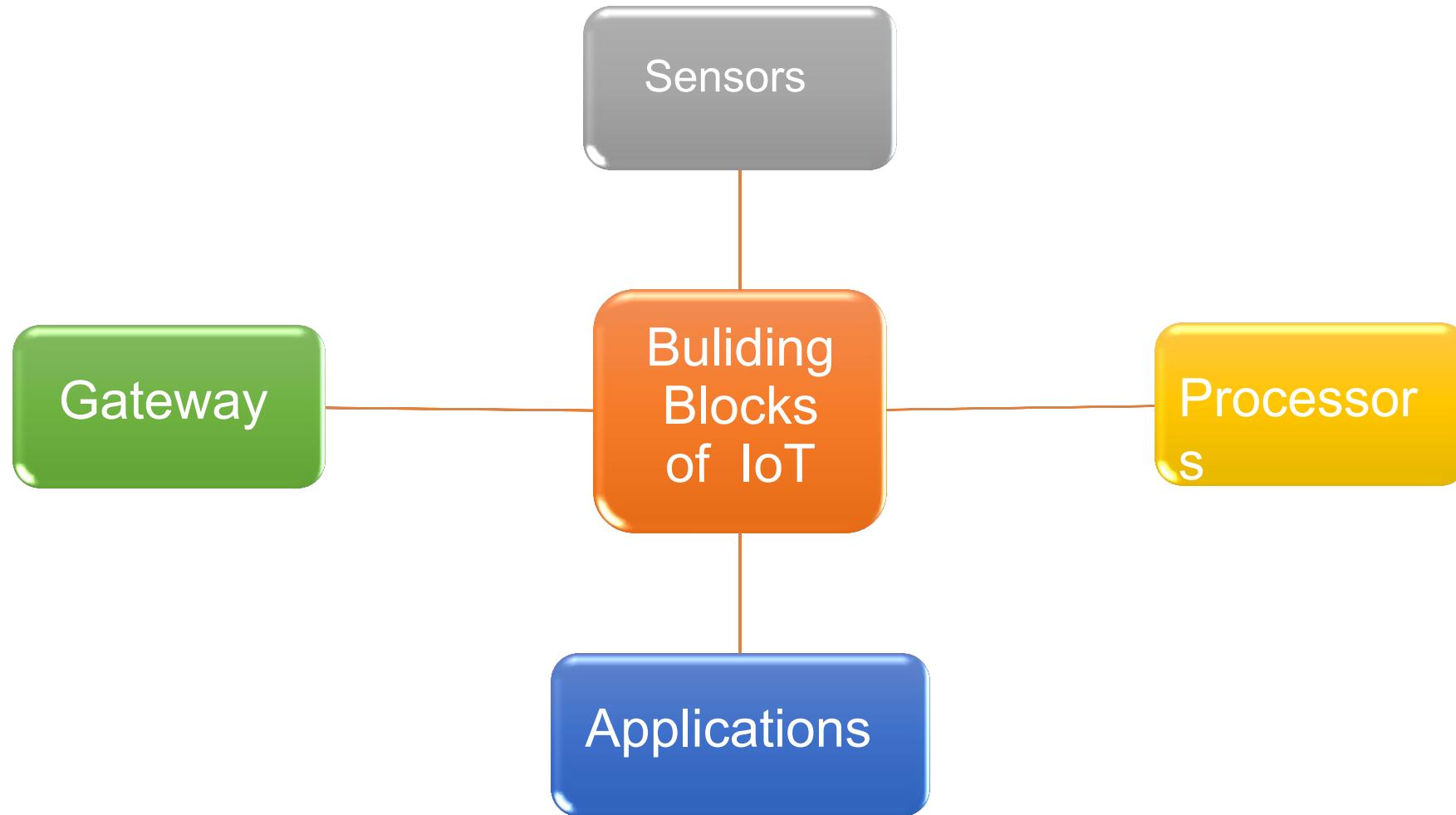
Formal Definition of IoT

- A **dynamic global network** infrastructure with **self- configuring capabilities** based on standard and **interoperable communication protocols**, where physical and virtual “things” have **identities**, physical attributes, and use intelligent interfaces, and are seamlessly **integrated** into **information network** that communicate data with users and environments.

Characteristics of IoT

- **Dynamic Global network & Self-Adapting** : Adapt the changes w.r.t changing contexts
- **Self Configuring** : Eg. Fetching latest s/w updates without manual intervention.
- **Interoperable Communication Protocols** : Communicate through various protocols
- **Unique Identity** : Such as Unique IP Address or a URI
- **Integrated into Information Network** : This allows to communicate and exchange data with other devices to perform certain analysis.

Building Blocks of IoT



Buliding Blocks of IoTSensors

- Sensors are the front end of the IoT devices.
- They really mean “things” in IoT.
- Their main task is to get necessary data from surroundings and pass it further to database or processing systems.
- They must be uniquely findable from there IP address because they are basic front end interface in the large network of other devices.
- Sensors collect real time data and can either work autonomous or can be user controlled.
- Examples of sensors are: gas sensor, water quality sensor, moisture sensor, etc.

Buliding Blocks of IoTProcessors

- Processors are the brain of the IoT system.
- The main job of processors it to process raw data collected by the sensors and **transforms** them **to some meaningful information** and knowledge. In short, we can say that its job is to **give intelligence to the data**.
- Processors are easily controllable by applications and their one more important job is **to securing data**. They perform encryption and decryption of data.
- Microcontroller, embedded hardware devices, etc can process the data using processors attached within the devices.

Buliding Blocks of IoTGateways

- Main task of gateways is to route the processed data and transfer it to proper databases or network storage for proper utilization.
- In other words, gateway helps in communication of the Communication data. and network connectivity are systems. essentials for IOT
- Examples of gateways are LAN, WAN, PAN, etc.

Buliding Blocks of IoTApplications

- Applications are another end of an IoT system. Applications **do proper utilization of all the data collected and provide interface to users to interact with that data.** These applications could be cloud based applications which are responsible for rendering data collected. Applications are user controllable and are delivery points of particular services.
- Examples of applications are: **smart home apps, security system control applications, industrial control hub applications, etc.**

Physical Design of IoT

- Things in IoT
- IoT Protocols

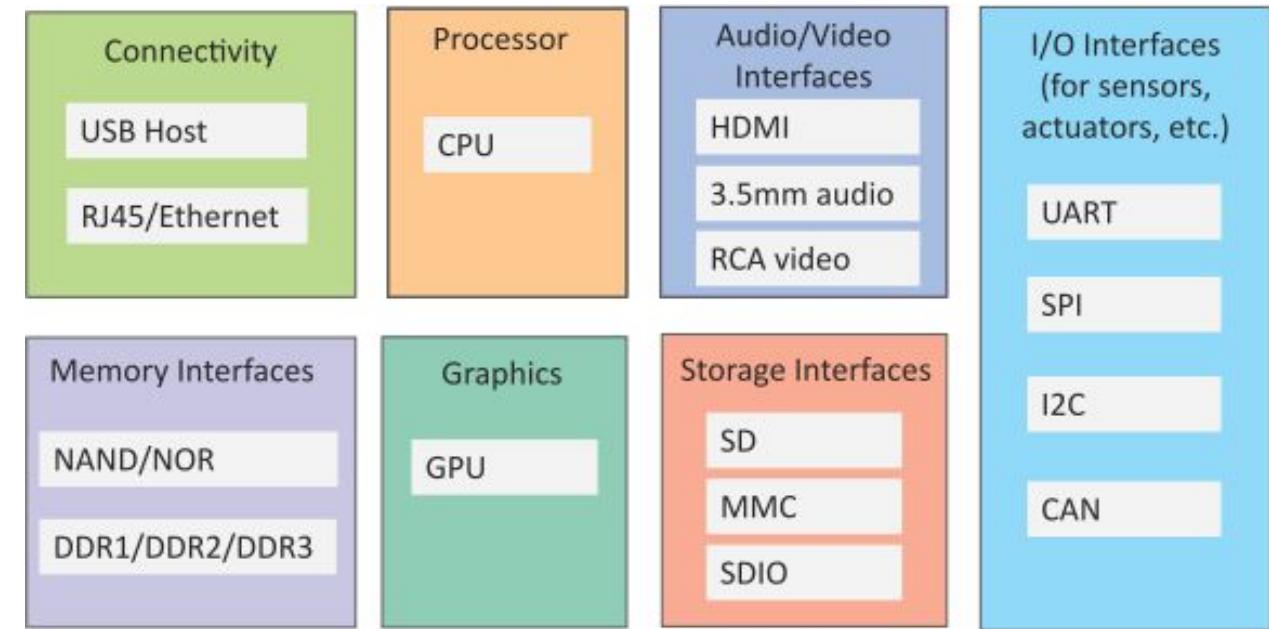
Things in IoT

- Refers to IoT devices which have unique identities that can perform sensing, actuating and monitoring capabilities.
- IoT devices can exchange data with other connected devices or collect data from other devices and process the data either locally or send the data to centralized servers or cloud – based application back-ends for processing the data.

Generic Block Diagram of an IoT Device

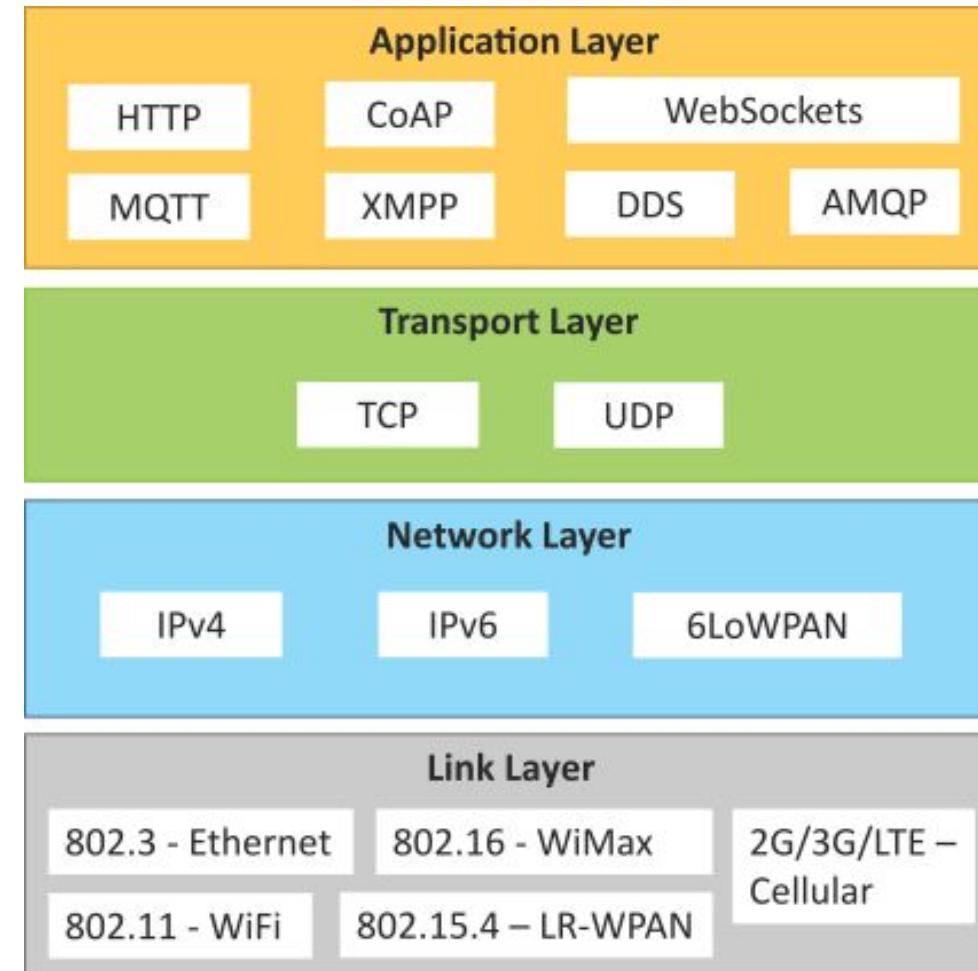
- An IoT device may consist of several interfaces for connections to other devices, both wired and wireless.

- I/O interfaces for sensors
- Interfaces for internet connectivity
- Memory and storage interfaces
- Audio/video interfaces

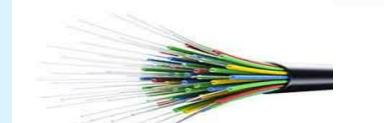


IoT Protocols

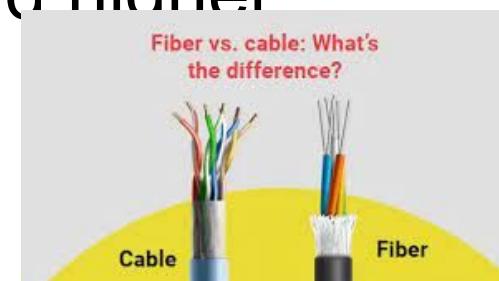
- Link Layer
 - 802.3 – Ethernet
 - 802.11 – WiFi
 - 802.16 – WiMax
 - 802.15.4 – LR-WPAN
 - 2G/3G/4G
- Network/Internet Layer
 - IPv4
 - IPv6
 - 6LoWPAN
- Transport Layer
 - TCP
 - UDP
- Application Layer
 - HTTP
 - CoAP
 - WebSocket
 - MQTT



IoT Protocols...Link Layer...Ethernet

Sr.No	Standard	Shared medium	
1	802.3	Coaxial Cable...10BASE5	
2	802.3.i	Copper Twisted pair10BASE-T	
3	802.3.j	Fiber Optic.....10BASE-F	
4	802.3.ae	Fiber.....10Gbits/s	

Data Rates are provided from 10Gbit/s to 40Gb/s and higher



IoT Protocols...Link Layer...WiFi

Sr.No	Standard	Operates in
1	802.11a	5 GHz band
2	802.11b and 802.11g	2.4GHz band
3	802.11.n	2.4/5 GHz bands
4	802.11.ac	5GHz band
5	802.11.ad	60Hz band

- Collection of Wireless LAN
- Data Rates from 1Mb/s to 6.75 Gb/s

IoT Protocols...Link Layer...WiMax

WiMAX stands for Worldwide Interoperability for Microwave Access

Sr.No	Standard	Data Rate
1	802.16m	100Mb/s for mobile stations 1Gb/s for fixed stations

- Collection of Wireless Broadband standards
- Data Rates from 1.5Mb/s to 1 Gb/s

IoT Protocols...Link Layer...LR-WPAN

- Collection of standards for low-rate wireless personal area networks
- Basis for high level communication protocols such as Zigbee.
- Data Rates from 40Kb/s to 250Kb/s.
- Provide low-cost and low-speed communication for power constrained devices.

IoT Protocols...Link Layer...2G/3G/4G –Mobile Communication

Sr.No	Standard	Operates in
1	2G	GSM-CDMA
2	3G	UMTS and CDMA 2000
3	4G	LTE

- Data Rates from 9.6Kb/s (for 2G) to up to 100Mb/s (for 4G)

IoT Protocols...Network/Internet Layer

- Responsible for sending of IP datagrams from source to destination network.
- Performs the host addressing and packet routing.
- Host identification is done using hierarchical IP addressing schemes such as IPV4 or IPV6.

Parameter	EtherNet	WiFi	WiMax	LR-WPAN(ZigBee)	Cellular
Used	Inside offices and houses	Outside offices and houses	Outside offices and houses	Outside offices and houses	Outside offices and houses
IEEE Standards:	802.3	802.11	802.16	802.15.4	
Range	100mtrs	100 mtrs	80-90kms	10-100 mtrs	1-5kms
Data Transfer Rate	10Mbps-100Mbps	54Mbps	40Mbps	250kbit/s	100Kbps-1MBps
Application	Houses, Offices, Industries	Mobile Applications, Video Conferencing	MetroPolitan Area Network	Smart Metering, Home Automation (Alexa), Smart Asset Tracking	Camera on Traffic Light, Video on Demand

IoT Protocols...Network Layer

- IPV4
 - Used to identify the devices on a network using hierarchical addressing scheme
 - Uses 32-bit address scheme
- IPV6
 - Uses 128-bit address scheme
- 6LoWPAN (IPV6 over Low power Wireless Personal Area Network)
 - Used for devices with limited processing capacity
 - Operates in 2.4 Ghz
 - Data Rates of 250Kb/s

IoT Protocols...Transport Layer

- Provide end-to-end message transfer capability independent of the underlying network.
- It provides functions such as error control, segmentation, flow- control and congestion control

IoT Protocols...TCP

- Transmission Control Protocol
- Connection Oriented
- Ensures Reliable transmission.
- Provides Error Detection Capability to ensure no duplicacy of packets and retransmit lost packets.
- Flow Control capability to ensure the sending data rate is not too high for the receiver process
- Congestion control capability helps in avoiding congestion which leads to degradation of n/w performance



IoT Protocols...UDP

- User Datagram Protocol
- Connectionless
- Does not ensure Reliable transmission
- Does not do connection before transmitting
- Does not provide proper ordering of messages
- Transaction oriented and stateless



IoT Protocols...Application Layer...Hyper Transfer Protocol

- Forms foundation of World Wide Web(WWW)
- Includes commands such as GET,PUT, POST, HEAD, OPTIONS, TRACE..etc
- Follows a request-response model
- Uses Universal Resource Identifiers(URIs) to identify HTTP resources



IoT Protocols...Application Layer...CoAP

- Constrained Application Protocol
- Used for Machine to machine (M2M) applications meant for constrained devices and n/w's
- Web transfer protocol for IoT and uses request-response model
- Uses client –server architecture
- Supports methods such as GET,POST, PUT and DELETE



Constrained Application Protocol
(Web Protocol for IoT)

Ananya Chakrabarti
Associate Vice President and Chief Architect, Digital Precision, Infineon
ananya.chakrabarti@infineon.com | <https://www.linkedin.com/in/ananya-chakrabarti/> | Twitter: [@AnanyaChakr](https://twitter.com/AnanyaChakr)

IoT Protocols...Application Layer...WebSocket

- Allows full-duplex communication over single socket
- Based on TCP
- Client can be a browser, IoT device or mobile application

IoT Protocols...Application Layer...MQTT

- Message Queue Telemetry Transport , light-weight messaging protocol
- Based on publish-subscribe model
- Well suited for constrained environments where devices have limited processing, low memory and n/w bandwidth requirement

IoT Protocols...Application Layer...XMPP

- Extensible messaging and presence protocol.
- For Real time communication and streaming XML data between n/w entities
- Used for Applications such as Multi-party chat and voice/video calls.
- Decentralized protocol and uses client server architecture.

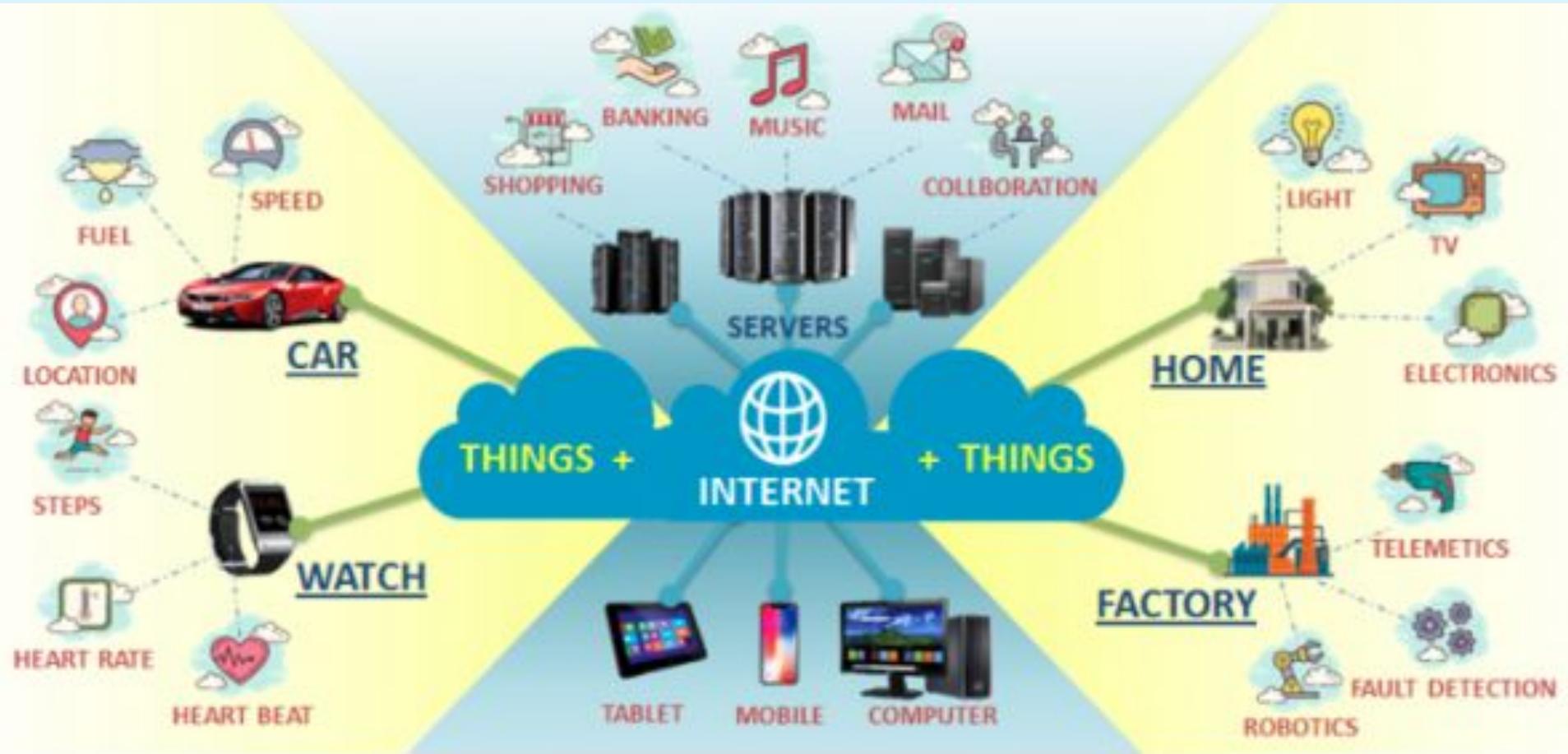
IoT Protocols...Application Layer...DDS

- Data Distribution service is a data-centric middleware standard for device-to-device or machine-to-machine communication.
- Publish subscribe model where publishers create topics to which subscribers can use.
- Provides Quality-of-service control and configurable reliability.

IoT Protocols...Application Layer...AMQP

- Advanced Messaging Queuing Protocol used for business messaging.
- Supports both point-to-point and publisher/subscriber models, routing and queuing
- Broker here receives messages from publishers and route them over connections to consumers through messaging queues.

Parameter	HTTP	CoAP	XMPP(Open XML)	DDS	AMQP	MQTT
Protocol	TCP	UDP	TCP	TCP and UDP	TCP	TCP
Network Layer	IP	6LowPAN	IP	IP	IP	IP
Architecture	Client-Server	Client-Server and Publish-Subscribe	Client-Server and Publish-Subscribe	Publish-Subscribe	Client-Server	Publish-Subscribe
Synchronization	Needed	No Need	Needed	Sometimes Needed, Sometimes Not	Needed	Needed
Designed for	Internet	IoT/M2M	IoT/M2M	Real Time Systems	M2M	IoT/M2M
Application	WWW	Retrieving Sensor Data	WhatsApp, Gaming, Google Talk	Volkswagen Smart Cars for Video Assistance	Google Cloud	Facebook Messenger

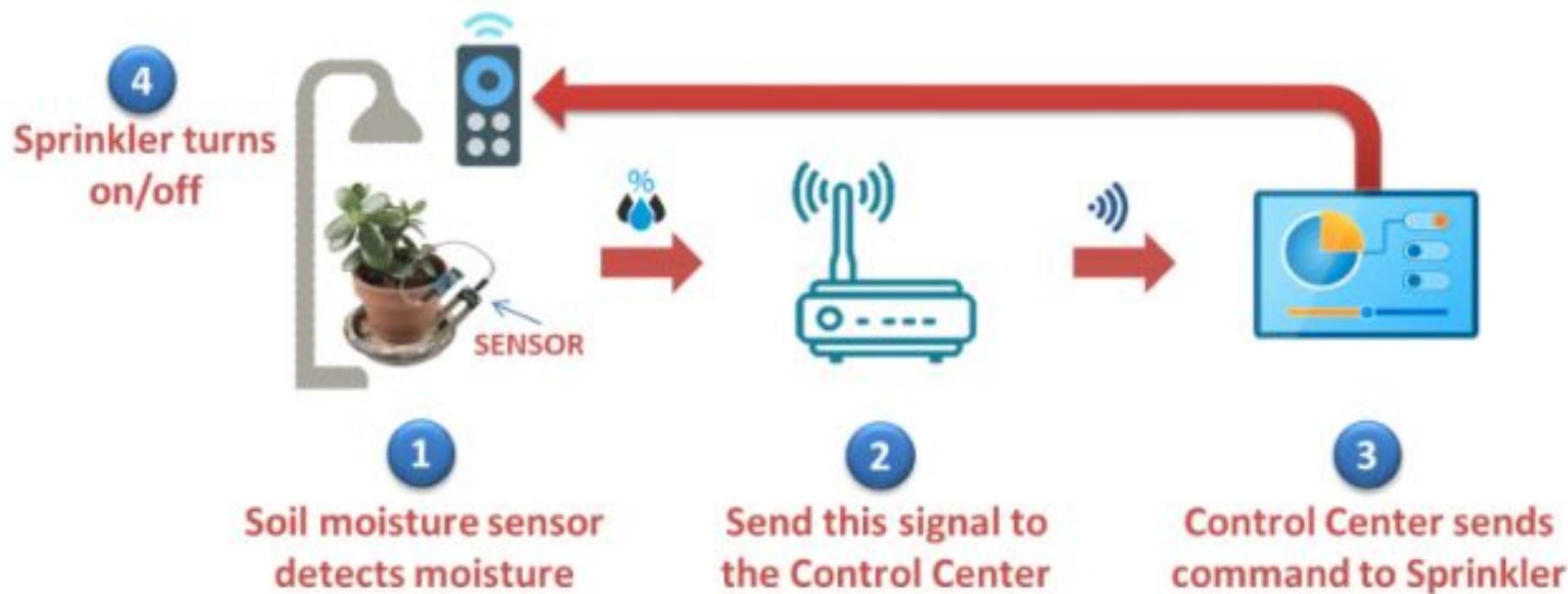


Internet of Things

Internet Of Things (IOT)

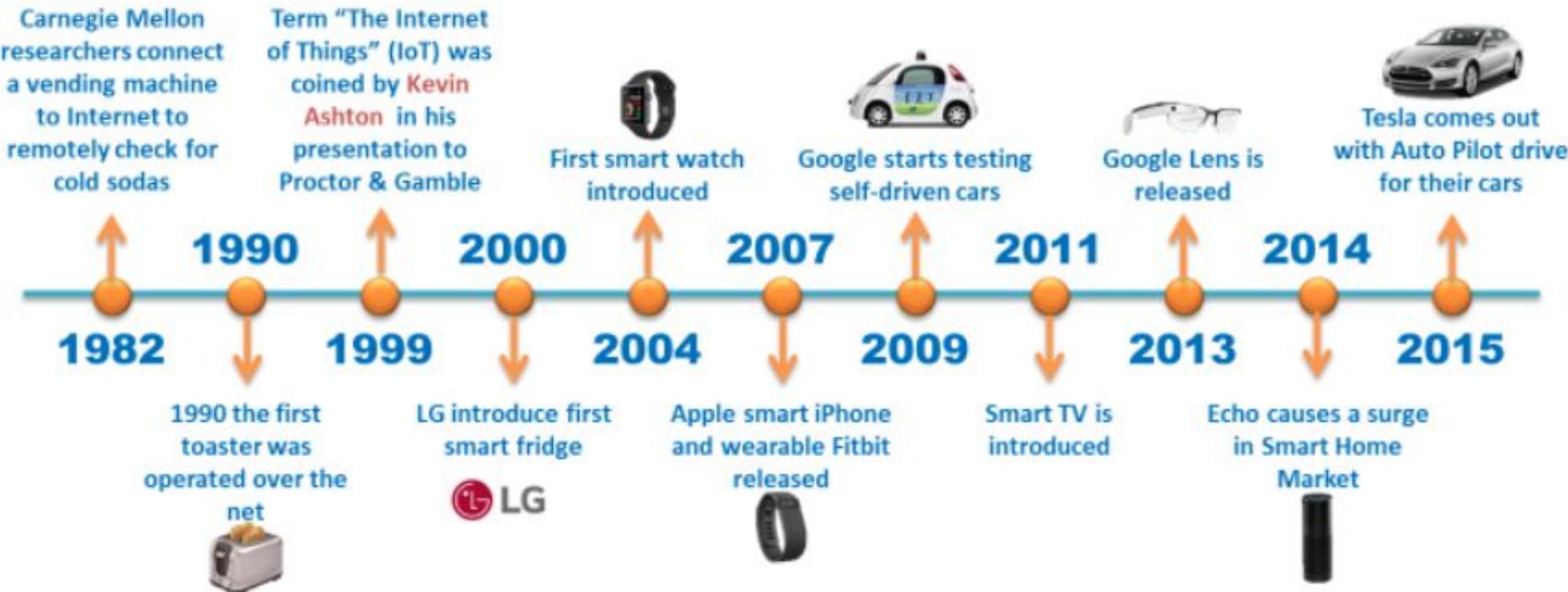
Taking everyday things, embedding them with electronics, software, sensors and then connecting them to internet and enabling them to collect and exchange data without human intervention is called as the Internet of Things (IoT)

Example of IOT Sprinkler System



Example of IOT Sprinkler System

History of IOT



History of IOT

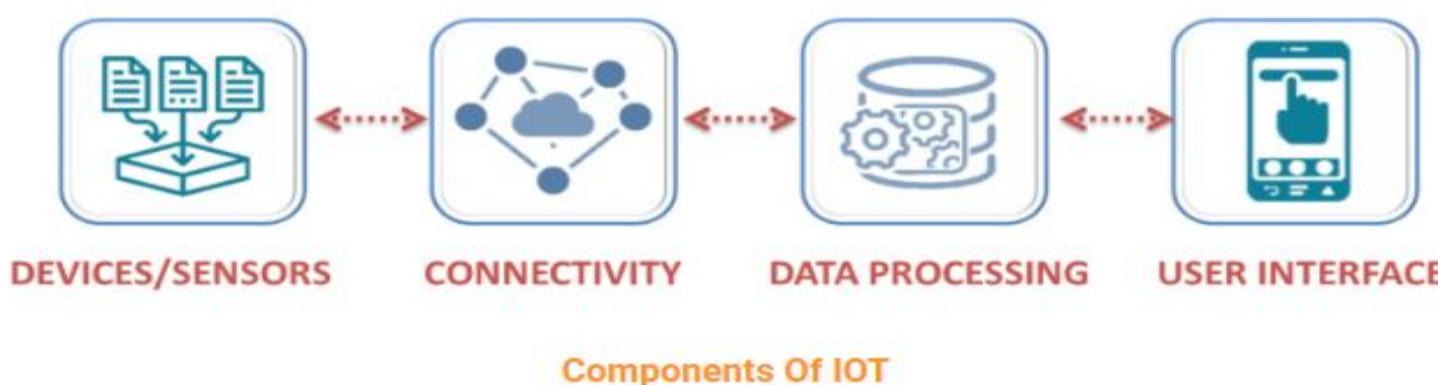
History of IOT

- The term “The Internet of Things” (IoT) was coined by Kevin Ashton in a presentation to Proctor & Gamble in 1999.
- However even before that after the internet was discovered, in 1982, Carnegie Mellon researchers connected a vending machine to Internet to remotely check for cold sodas.
- In 1990 the first toaster was operated over the net.
- In 2000 LG introduced first smart fridge.
- In 2004 smart watch was introduced and in 2007 smart iphone and wearable Fitbit were released.
- In 2009 Google started testing self-driven cars.
- In 2011 Smart TV was introduced.
- In 2013 Google Lens is released followed by Echo in 2014 which causes a surge in smart home market.
- IOT continues to grow dramatically with Tesla coming out with Auto Pilot in 2015, IOT continue to proliferate with cheaper devices and sensors

How does IoT work?

There are four main components based on which an internet of things ecosystem works on:

devices or Sensors, Connectivity, Data Processing and User Interface.



1. Sensors

1. Sensor

It is a device that measures physical input from its environment and converts it into data that can be interpreted by a computer.

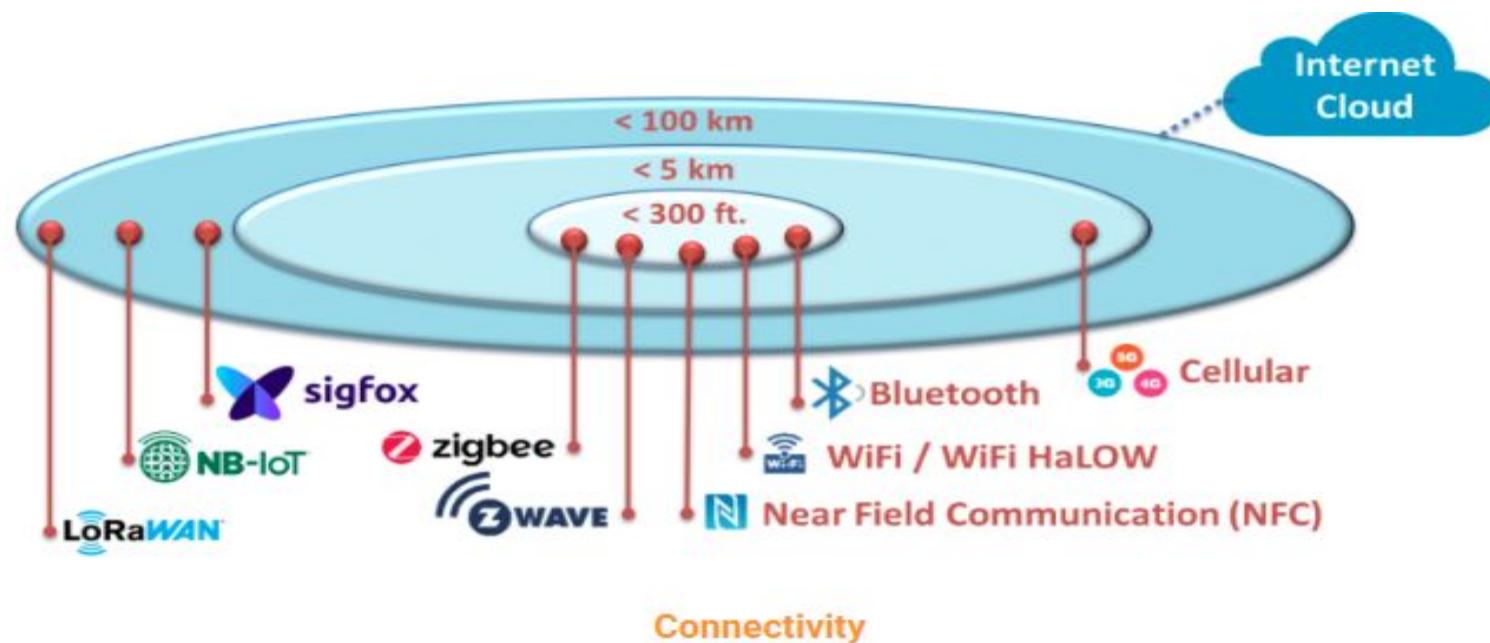
There are various types of sensors available now, For e.g. sensing motion, temperature, pressure, light, sound etc.

This sensor is typically integrated with a microprocessor based embedded system which can collate the data and connect to internet.



2. Connectivity

- Several Communication Protocols and Technology used in IOT.
- Depending upon Range, Power Usage, Cost, Data Rate etc. right one is used. E.g mobile, Bluetooth, WI-FI, LoraWAN, etc.
- All the collected data is sent via internet to a cloud infrastructure.



Advantages of IOT

Advantages

- ✓ Minimize human effort and save time
- ✓ Lead to more automation and technical optimization
- ✓ Help us to reduce waste and use our natural resources effectively

Disadvantages

- ✗ Security of confidential Data is a key concern
- ✗ Can lead to various types of network attacks
- ✗ Maintaining privacy is a challenge

3. Data Processing

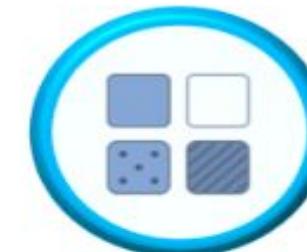
- In the processing stage, a computer transforms the raw data into information.
- The transformation is carried out by using different data manipulation techniques.
- This process can be just aggregating from multiple devices like AC or Light. Or it can be complex like extracting car number plates from video feed of speeding cars.
- It could be classifying the data or do real time analytics and identify patterns for human analysis.



Data Aggregation



Data Extraction



Data Classification



Data Analytics

Data Processing

4. User Interface

- The information processed is made available to the end-user in some way, like an app which can trigger alarm or send them notification through email or text message.
- It might provide the user with actual live feed or show trends etc.
- The application could also provide an interface to send instructions back as well, like resetting the temperature or releasing water to the plants based upon moisture reading etc.



Alerts



Notifications



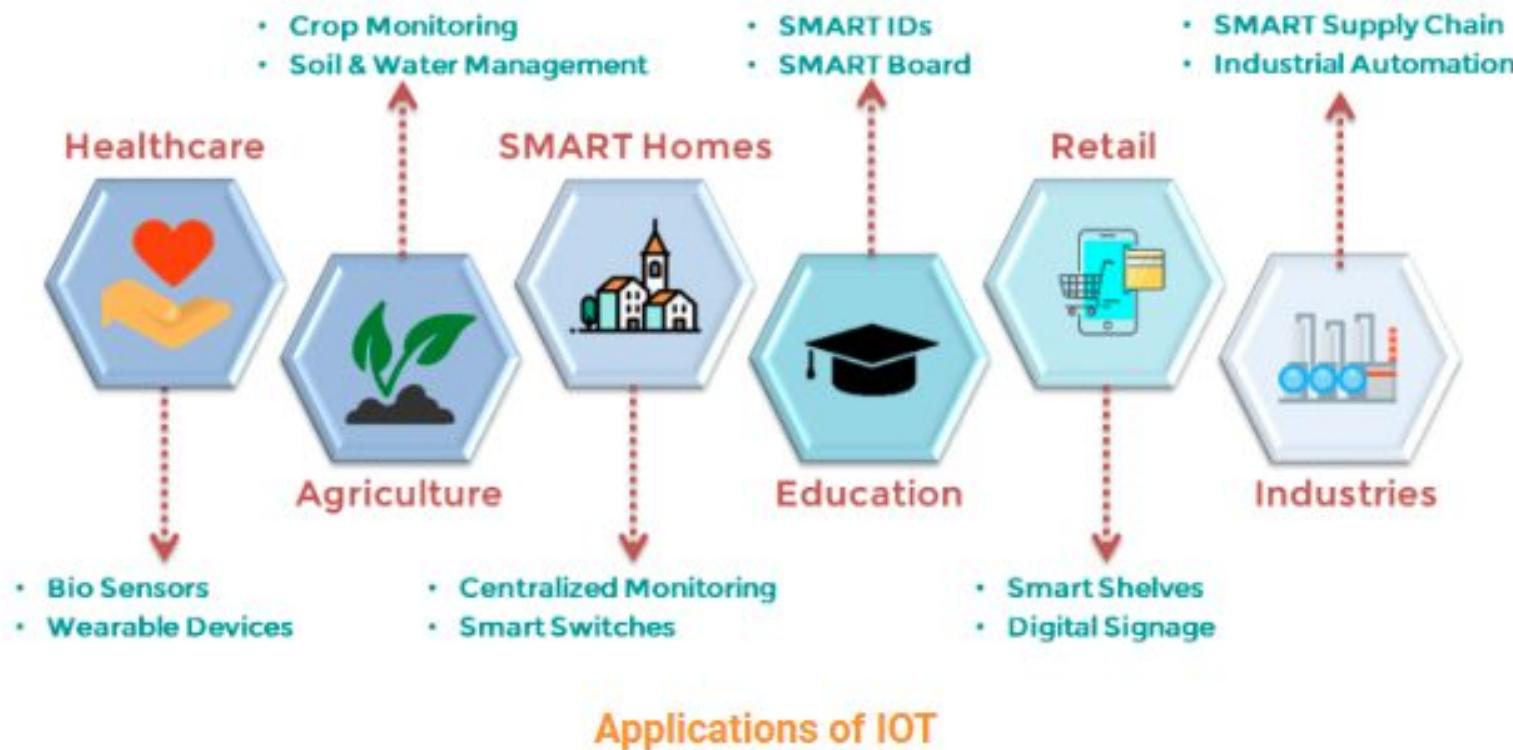
Live Trends



Remote Control

User Interface

Applications of IOT



Web Protocols for IoT



HTTP(HyperText Transfer Protocol)

- HTTP (Hypertext Transfer Protocol) is perhaps the most popular application protocol used in the Internet (or The WEB).
- HTTP is an *asymmetric request-response client-server protocol*
- HTTP is state-less

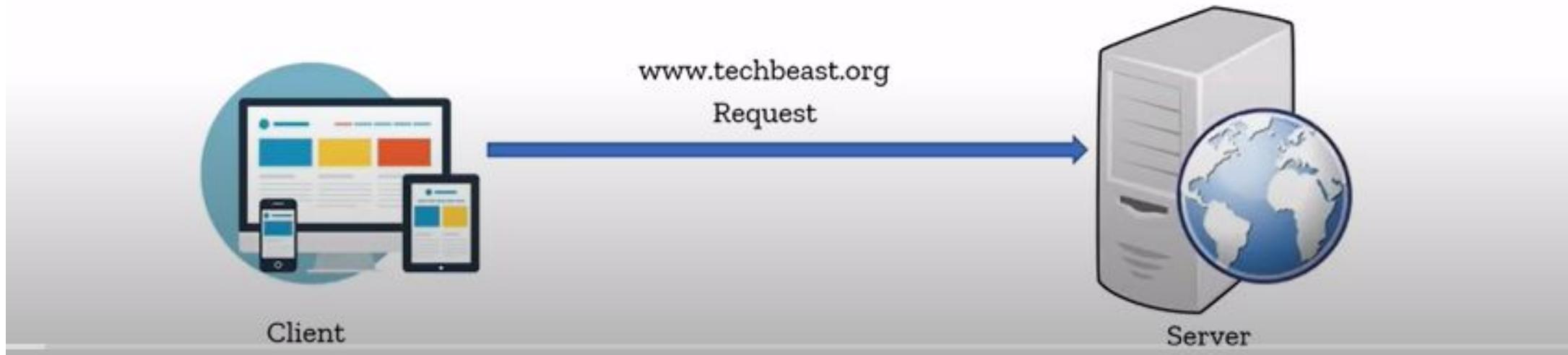


Client

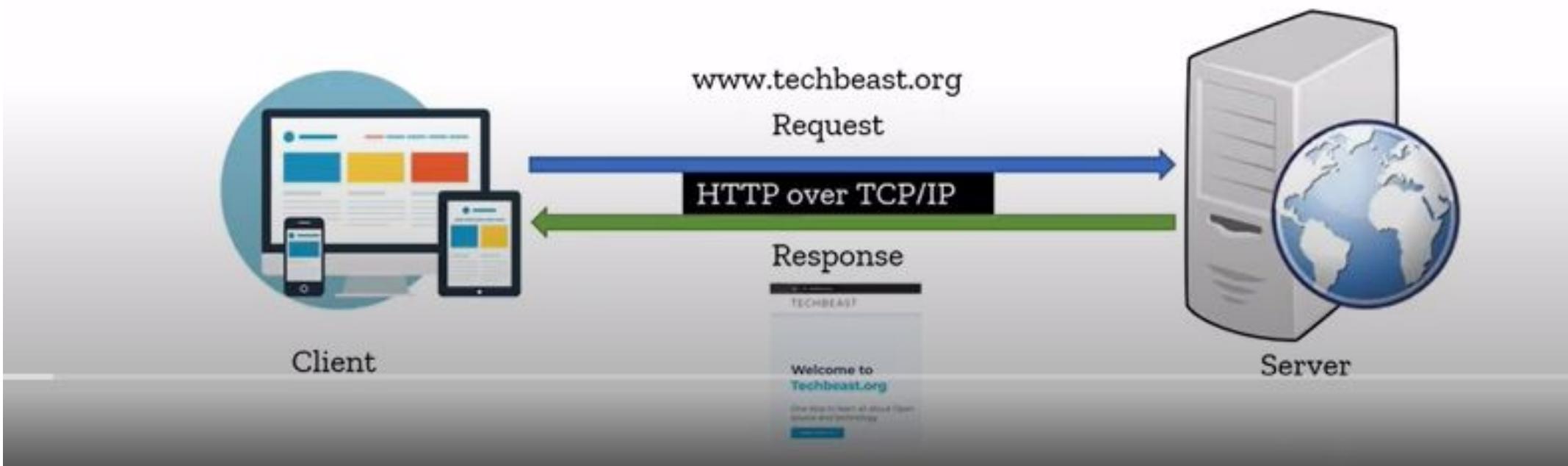


Server

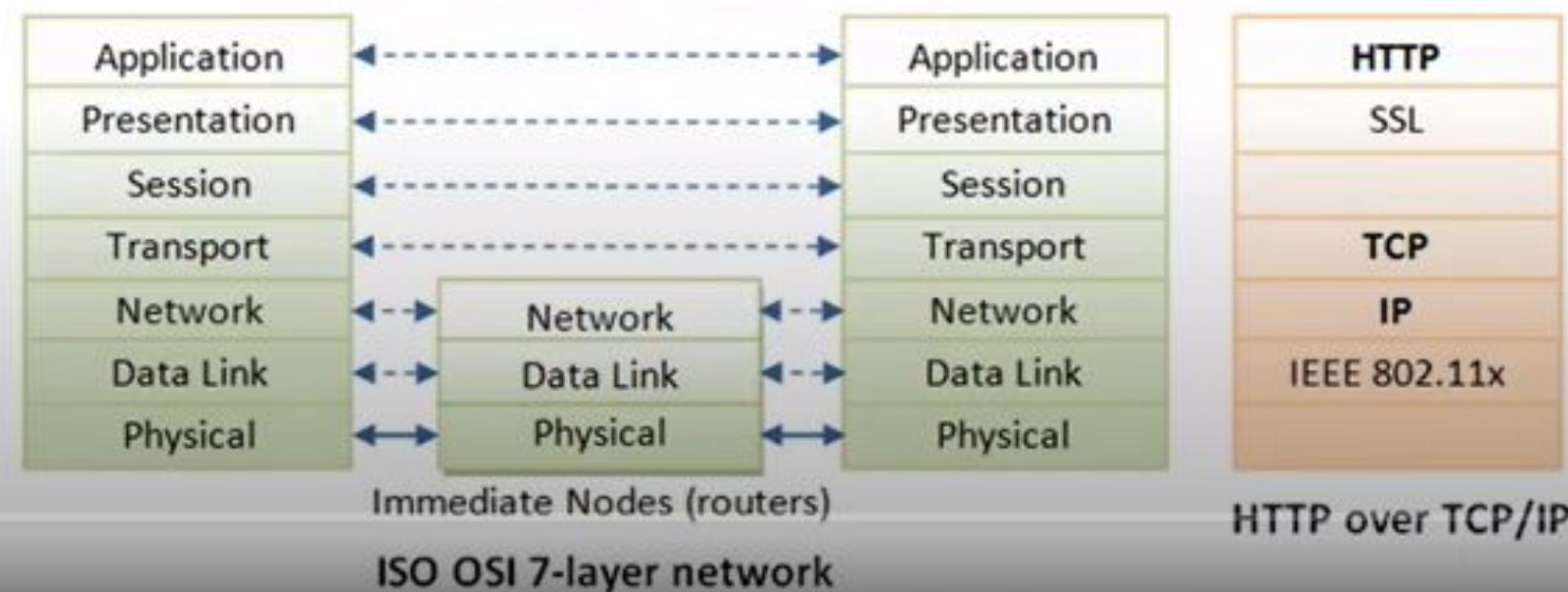
- HTTP (Hypertext Transfer Protocol) is perhaps the most popular application protocol used in the Internet (or The WEB).
- HTTP is an asymmetric request-response client-server protocol
- HTTP is state-less



- HTTP (Hypertext Transfer Protocol) is perhaps the most popular application protocol used in the Internet (or The WEB).
- HTTP is an asymmetric *request-response* client-server protocol
- HTTP is state-less

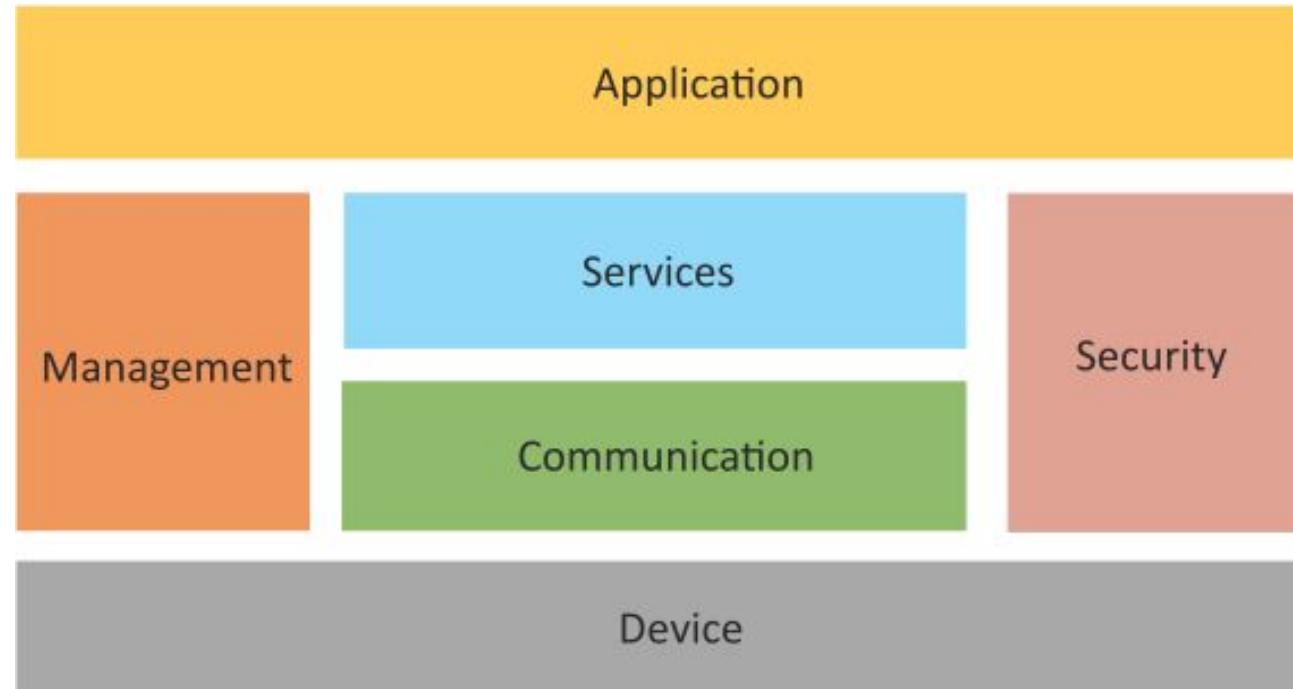


- HTTP runs over TCP/IP in most of the scenario to ensure packet delivery guarantee.
- TCP/IP is a Transport and Network Layer protocol used to communicate between two machines



Logical Design of IoT

- Logical design of an IoT system refers to an abstract representation of the entities and processes without going into the low-level specifics of the implementation.
- An IoT system comprises of a number of functional blocks that provide the system the capabilities for identification, sensing, actuation, communication, and management.



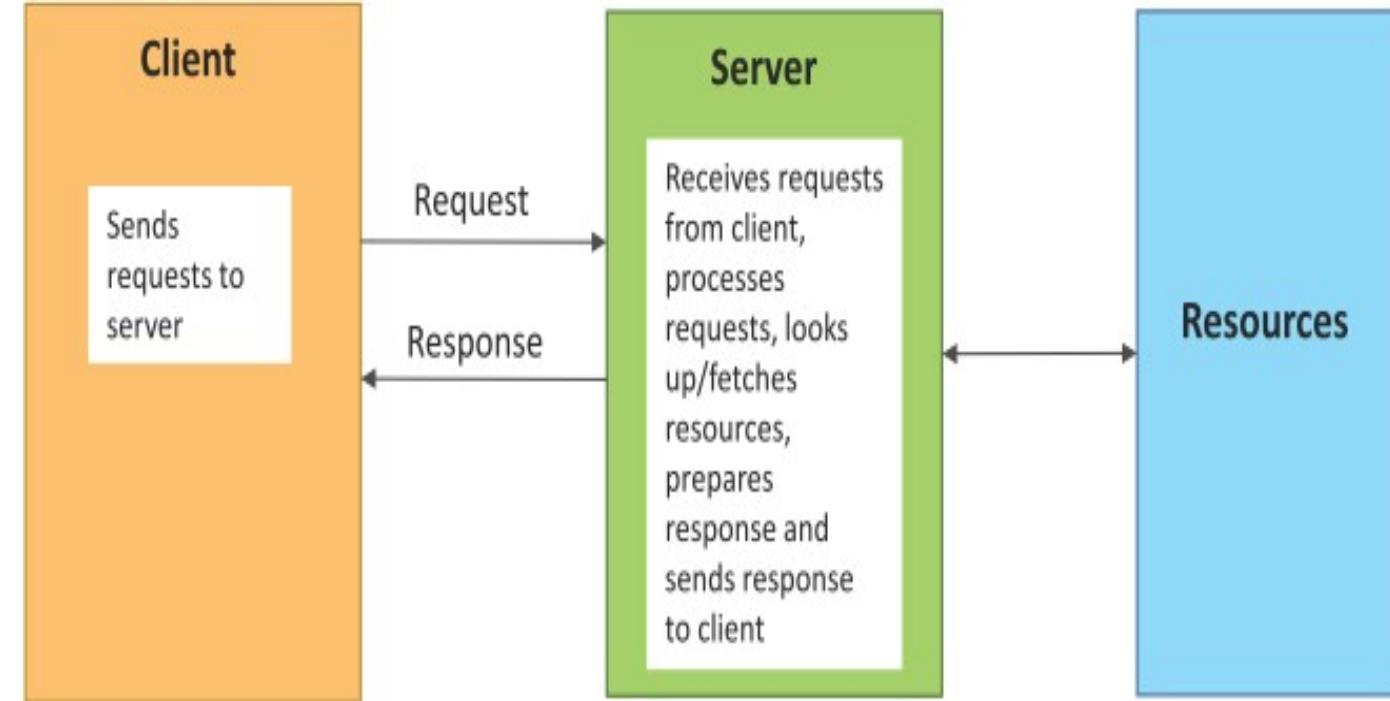
Logical Design of IoT

- Device : Devices such as sensing, actuation, monitoring and control functions.
- Communication : IoT Protocols
- Services like device monitoring, device control services, data publishing services and device discovery
- Management : Functions to govern the system
- Security : Functions as authentication, authorization, message and content integrity, and data security
- Applications

Communication Models

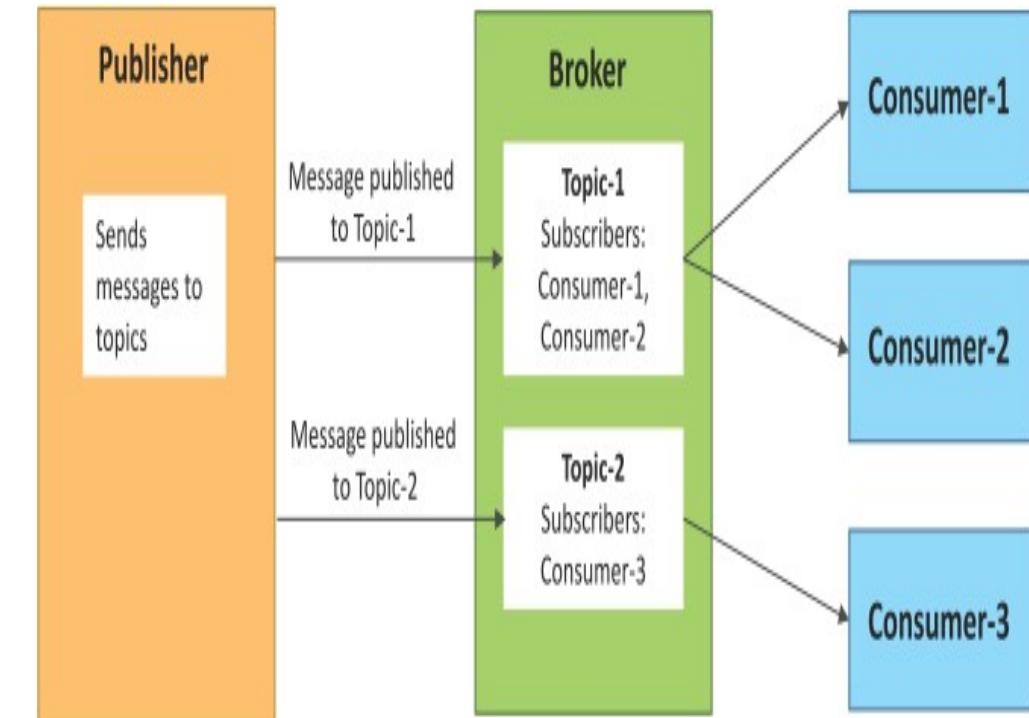
Request–Response Communication Model

- Request–Response is a communication model in which the client sends requests to the server and the server responds to the requests.
- When the server receives a request, it decides how to respond, fetches the data, retrieves resource representations, prepares the response and then sends the response to the client.
- Stateless communication model



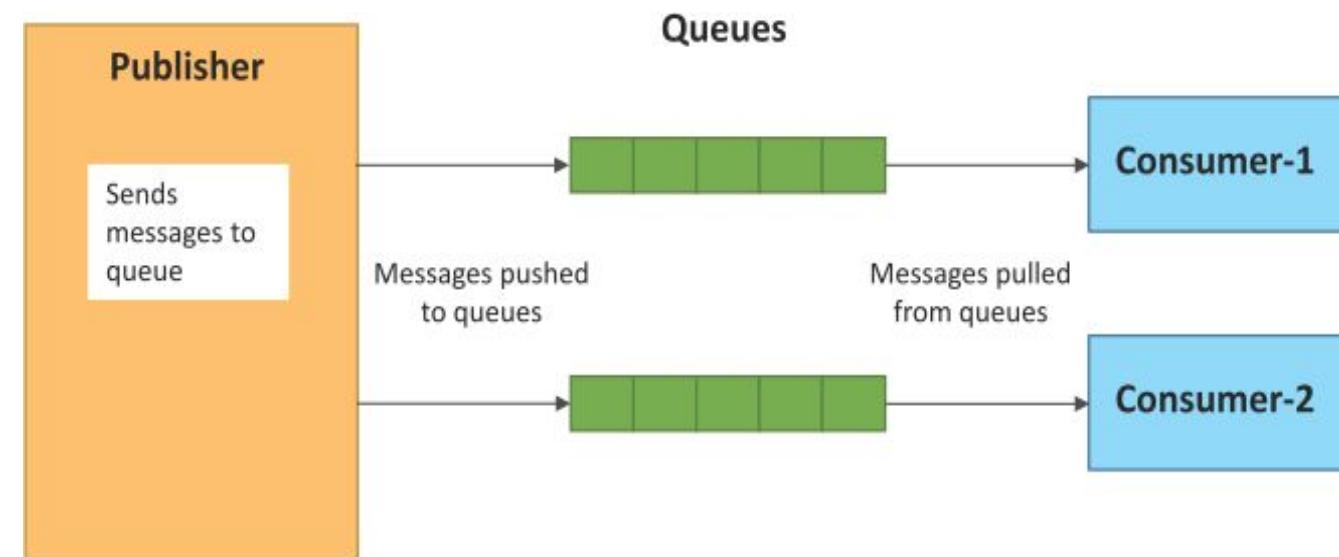
Publish–Subscribe Communication Model

- Publish–Subscribe is a communication model that involves publishers, brokers and consumers.
- Publishers are the source of data. Publishers send the data to the topics which are managed by the broker. Publishers are not aware of the consumers.
- Consumers subscribe to the topics which are managed by the broker.
- When the broker receives data for a topic from the publisher, it sends the data to all the subscribed consumers.



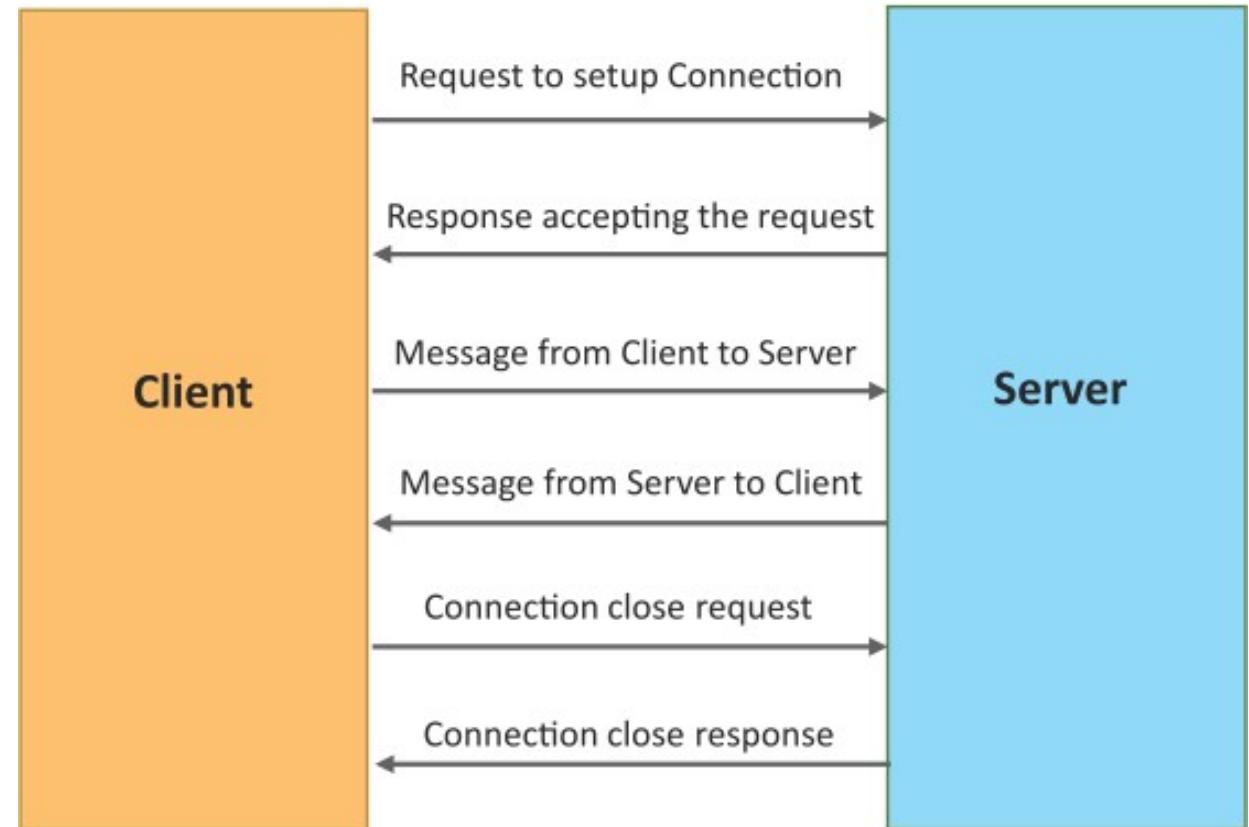
Push–Pull Communication Model

- Push–Pull is a 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 consumers.
- Queues help in decoupling the messaging between the producers and consumers.
- Queues also act as a buffer which helps in situations when there is a mismatch between the rate at which the producers push data and the rate at which the consumers pull data.



Exclusive Pair Communication Model

- Exclusive Pair is a bidirectional fully duplex communication model that uses a persistent connection between the client and the server.
- Once the connection is set up 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.



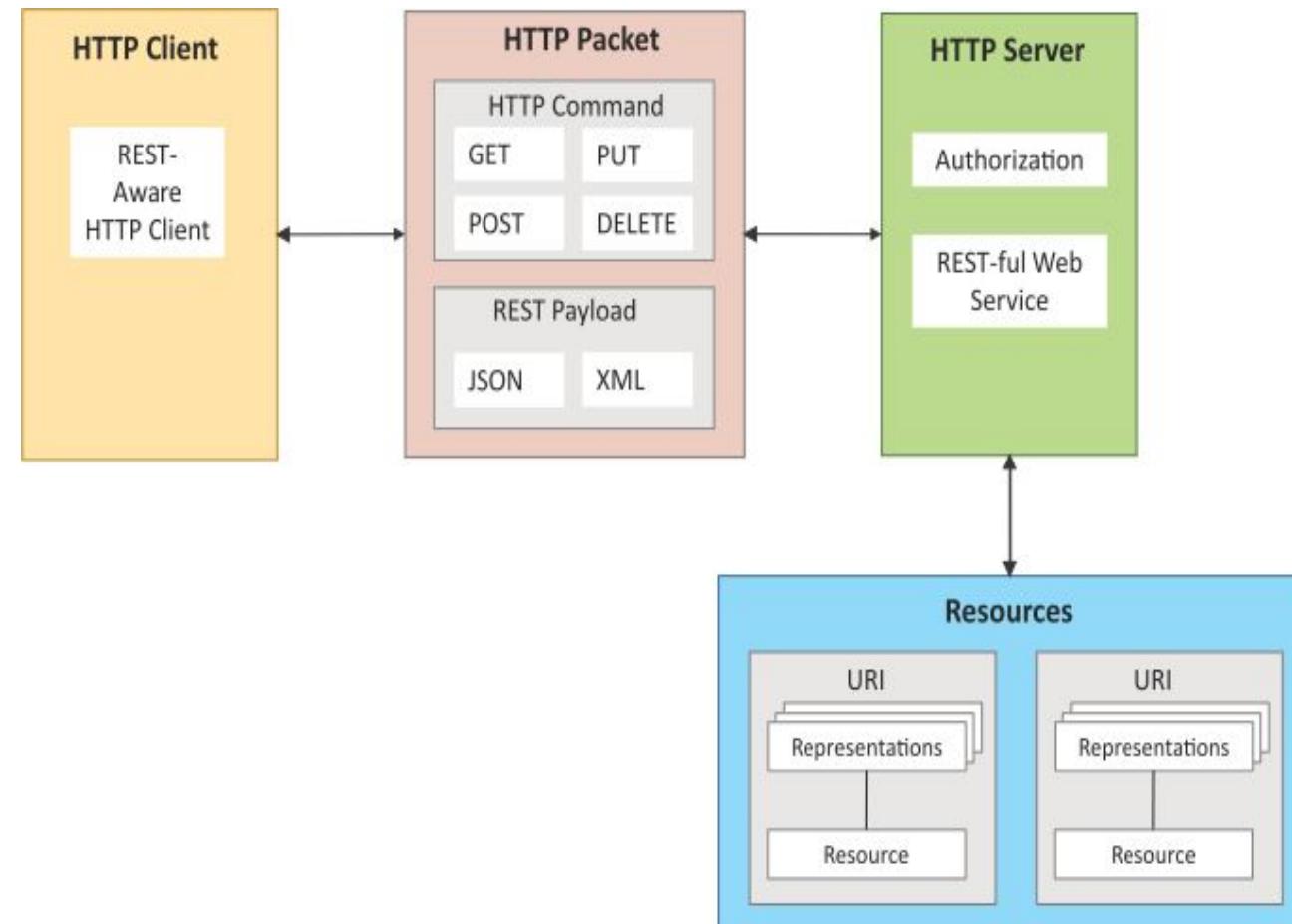
Communication APIs

IoT Communication APIs

1. An API is an interface used by programs to access an application.
2. It enables a program to send commands to another program and receive replies from the app.
3. IoT APIs are the interface points between an IoT device and the Internet and/or other network components.

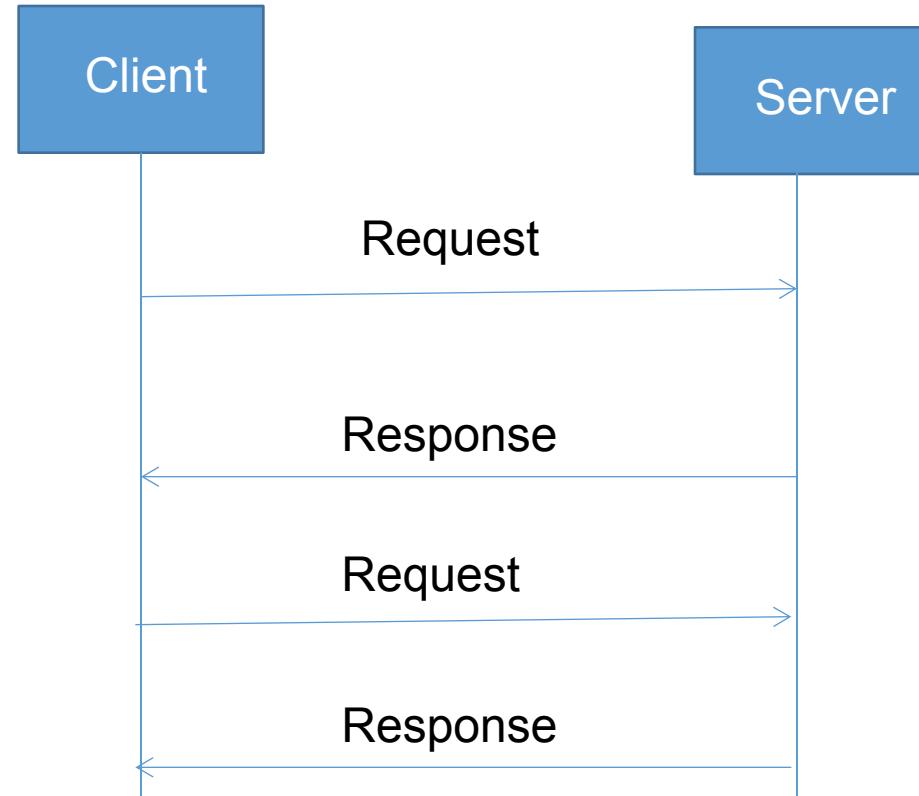
REST-based Communication APIs

- Representational State Transfer (REST) is a set of architectural principles by which you can **design web services and web APIs** that focus on a system's resources and how resource states are addressed.
- REST APIs **follow the request-response communication model.**
- REST architectural constraints apply to the components, connectors and data elements **hypermedia distributed**



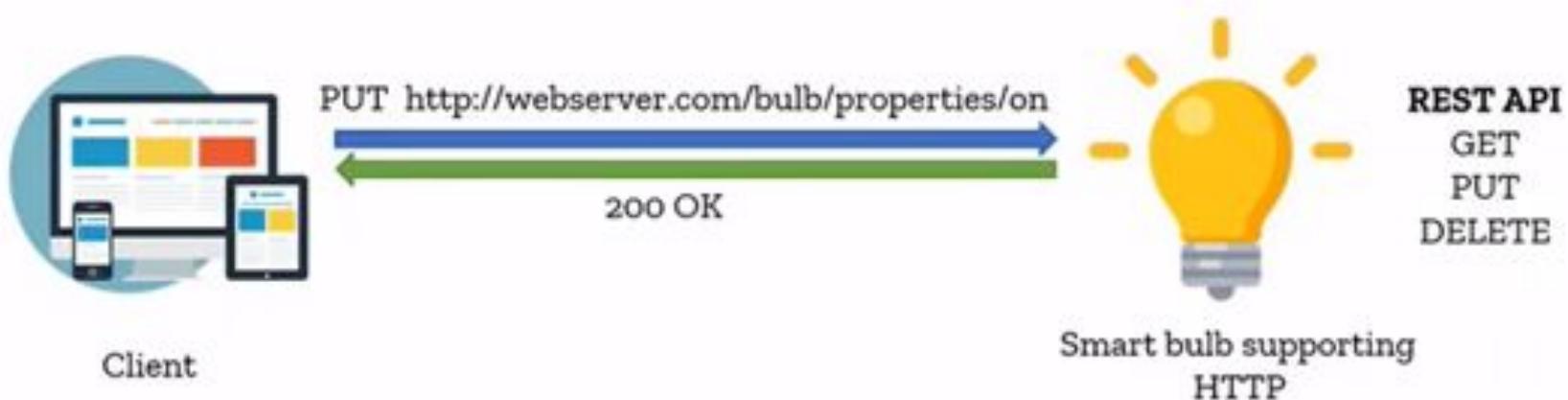
REST-based Communication APIs Constraints

- Client – Server
- Stateless
- Cacheable
- Layered System
- Uniform Interface
- Code on demand



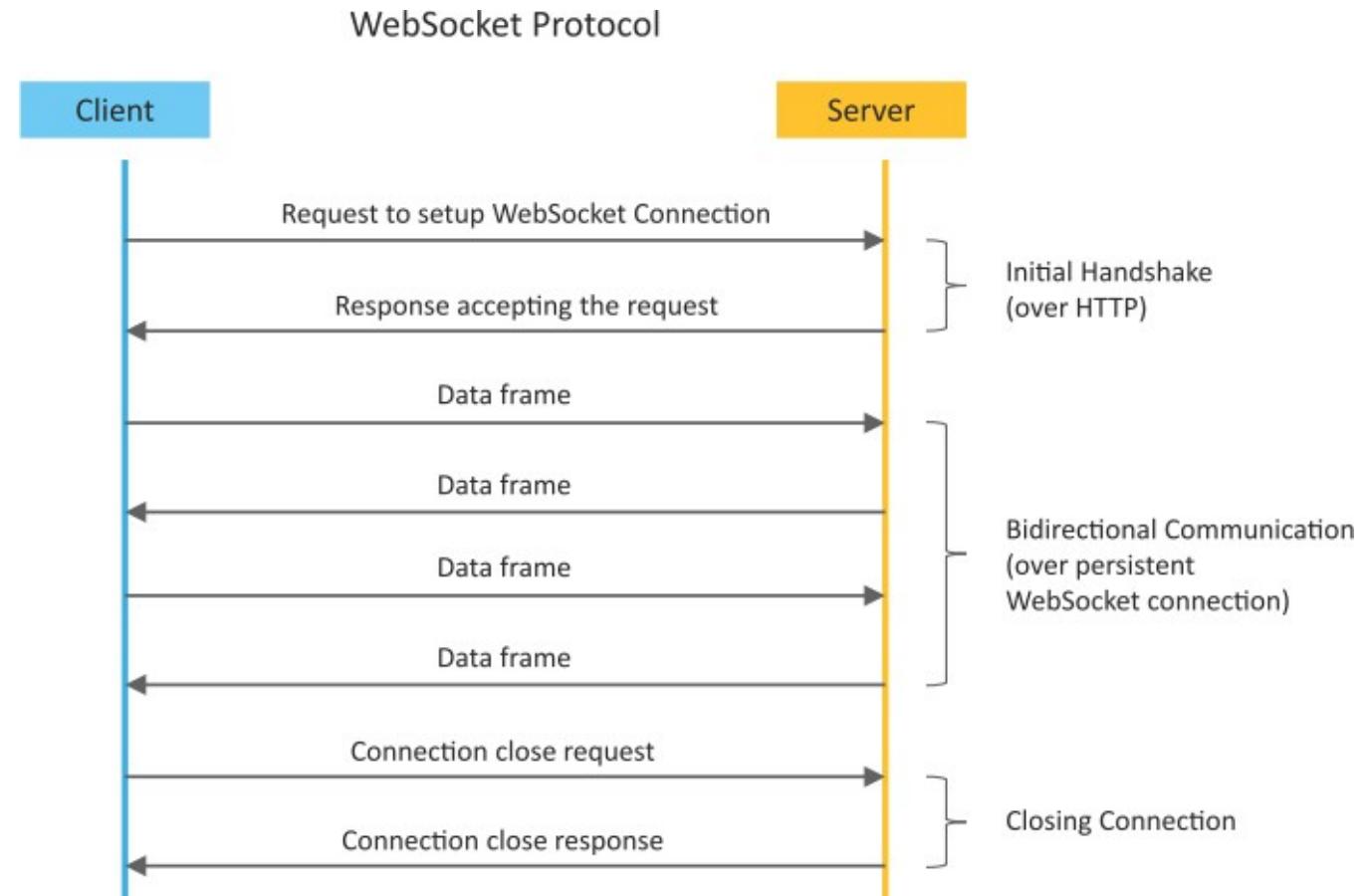
How HTTP works in IoT

- REST – REpresentational State Transfer is a standard/design to communicate between systems on the Web whereas **HTTP** is a implementation based on REST standard.
- If any architecture in the web follows REST standards it is said to be RESTful



WebSocket-based Communication APIs

- WebSocket APIs allow **bi-directional, full duplex communication** between clients and servers.
- WebSocket APIs follow the **exclusive pair communication model**.



Why WebSocket ?

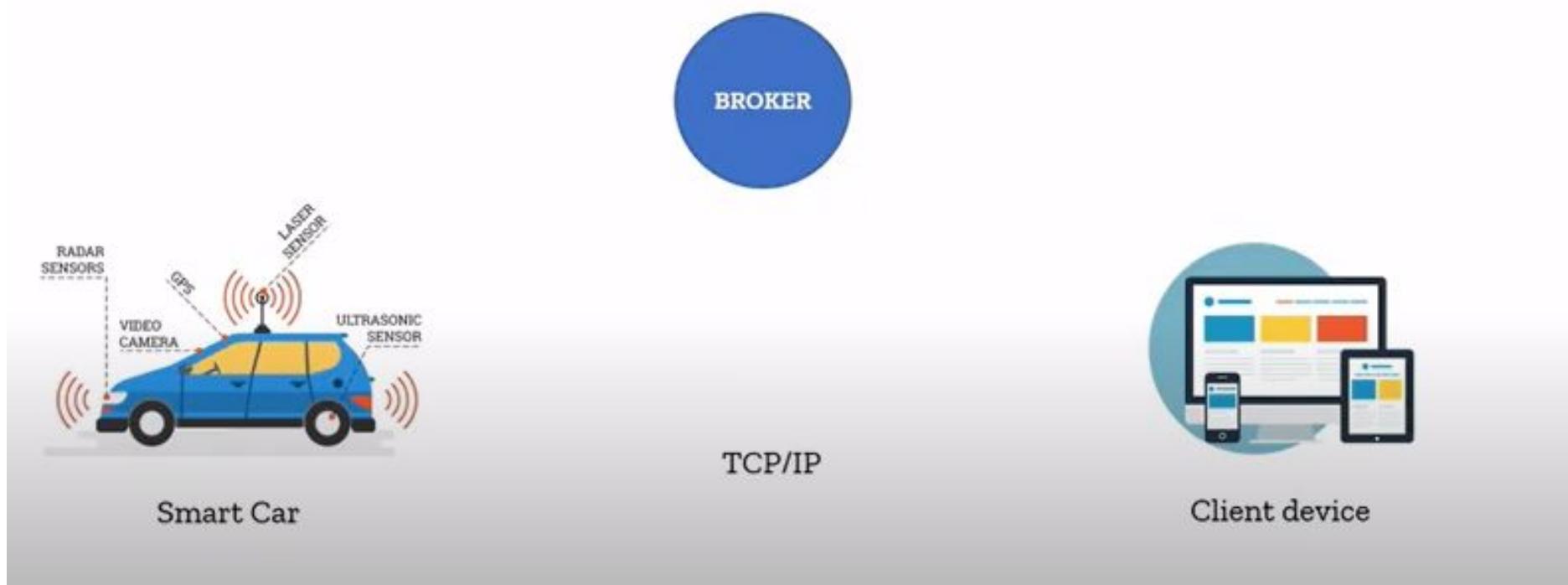
- It uses standard internet and web technologies
- WebSocket aren't blocked by firewalls and can traverse proxies
- Useful for real time communications without any delay in IoT applications
- Very less bandwidth than HTTP polling
- Faster than HTTP

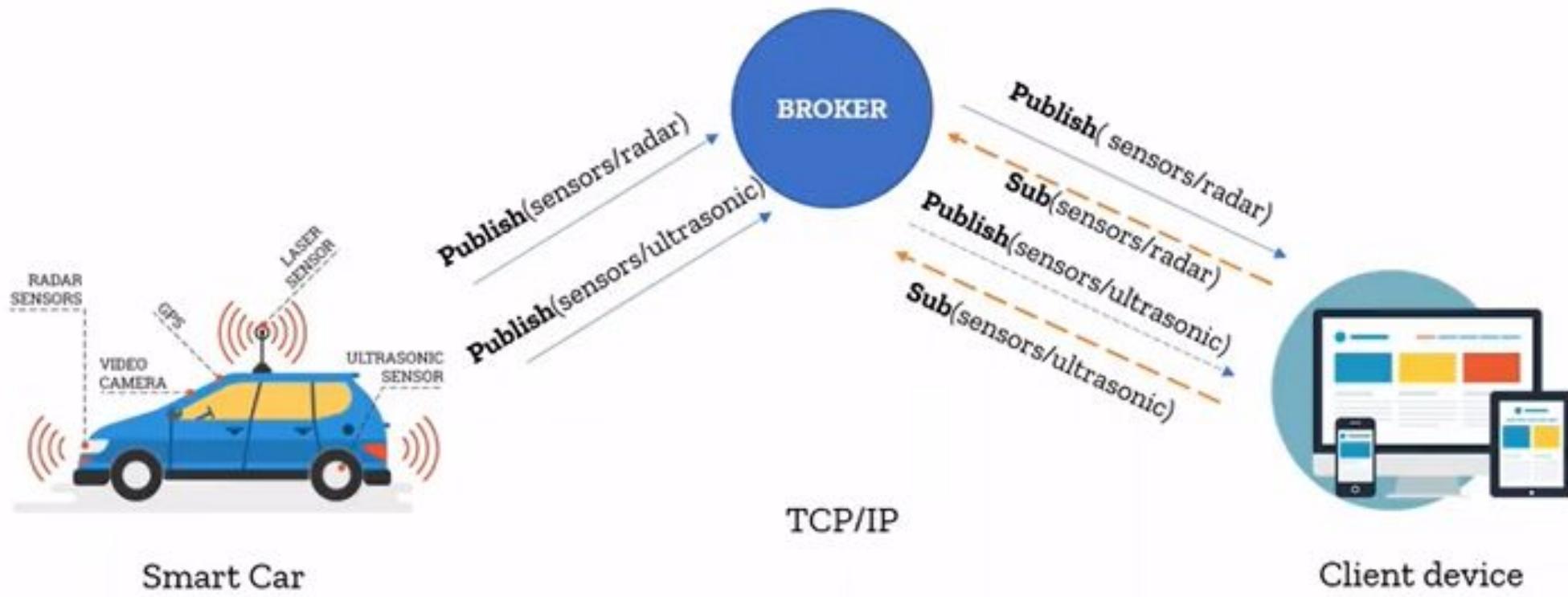
Difference between REST and WebSocket-based Communication APIs

Comparison Based on	REST	Websocket
State	Stateless	Stateful
Directional	Unidirectional	Bidirectional
Req-Res/Full Duplex	Follow Request Response Model	Exclusive Pair Model
TCP Connections	Each HTTP request involves setting up a new TCP Connection	Involves a single TCP Connection for all requests
Header Overhead	Each request carries HTTP Headers, hence not suitable for real-time	Does not involve overhead of headers.
Scalability	Both horizontal and vertical are easier	Only Vertical is easier

MQTT (Message Queuing Telemetry Transport)

- MQTT is an application Layer Protocol located at Layer 7 in OSI model which works on **Publish/Subscribe** concept
- Runs on the **top of TCP/IP**
- Messages are published and subscribed by a mediator called "**Broker**"
- Broker can be in **local** or in **internet**
- Offers **three** level of **QoS**
 - QoS 0 - Fire and Forget (no guarantee)
 - QoS 1 - Deliver at least once
 - QoS 2 - Deliver exactly once





Why MQTT

- Mainly used for Constrained devices with limited bandwidth
- Uses less battery power to publish and subscribe data
- Connection is secured using TLS where traffic is encrypted on the Web
- Can be used in wide range of IoT applications

IoT Enabling Technologies

- Wireless Sensor Network
- Cloud Computing
- Big Data Analytics
- Embedded Systems



WSN

- **Distributed Devices with sensors** used to monitor the environmental and physical conditions.
- Consists of several **end-nodes acting as routers or coordinators** too.
- **Coordinators collects data** from all nodes / **acts as gateway** that connects WSN to internet
- **Routers route the data packets** from end nodes to coordinators.

Example of WSNs in IoT & Protocols used

Example

- Weather monitoring system
- Indoor Air quality monitoring system
- Soil moisture monitoring system
- Surveillance systems
- Health monitoring systems

Protocols

- Zigbee

Cloud Computing

https://youtu.be/cBRstE_C3iQ

- **Deliver applications and services over internet.**
- Provides computing, networking and storage resources on demand.
- Cloud computing performs services such as IaaS, PaaS and SaaS (<https://youtu.be/36zducUX16w>)
- IaaS: Rent Infrastructure
- PaaS : supply an on-demand environment for developing, testing, delivering and managing software applications.
- SaaS : method for delivering software applications over the Internet, on demand and typically on a subscription basis.

Big Data Analytics

- Collection of data whose volume, velocity or variety is too large and difficult to store, manage, process and analyze the data using traditional databases.
- It involves data cleansing, processing and visualization
- Lots of data is being collected and warehouse
 - Web data, e-commerce
 - purchases at department/ grocery stores
 - Bank/Credit Card transactions
 - Social Network



Big Data Analytics

Variety Includes different types of data

- Structured
- Unstructured
- SemiStructured
- All of above

Big Data Analytics

Velocity Refers to speed at which data is processed

- Batch
- Real-time
- STreams

Big Data Analytics

Volume refers to the amount of data

- Terabyte
- Records
- Transactions
- Files
- Tables

IoT Levels and Deployment Templates

An IoT system comprises the following components:

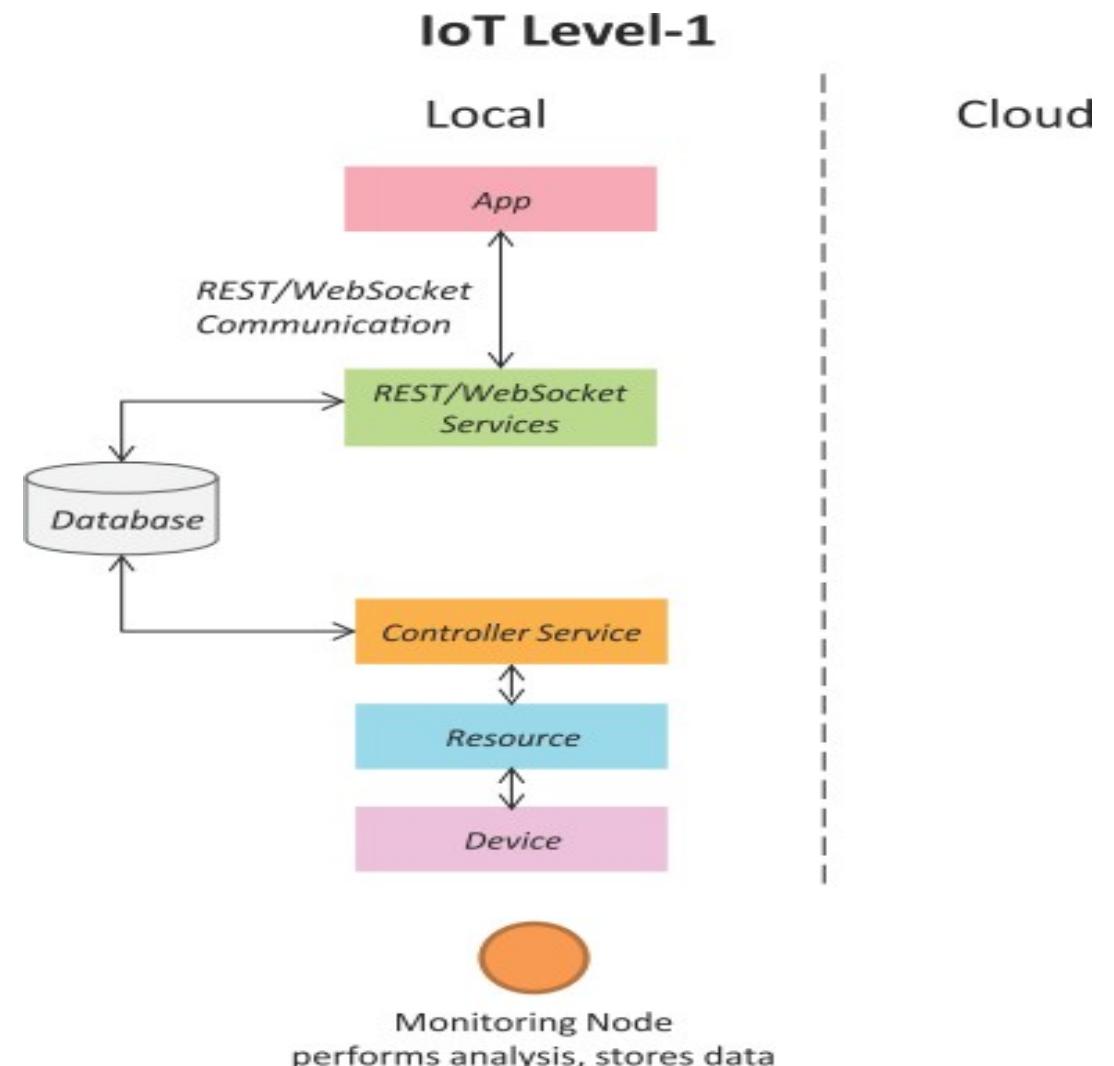
- **Device:** An IoT device allows identification, remote sensing, actuating and remote monitoring capabilities.
- **Resource:** Resources are software components on the IoT device for accessing, processing and storing sensor information, or for controlling actuators connected to the device. Resources also include the software components that enable network access for the device.
- **Controller Service:** Controller service is a native service that runs on the device and interacts with the web services. Controller service sends data from the device to the web service and receives commands from the application (via web services) for controlling the device.

IoT Levels and Deployment Templates

- **Database:** Database can be either local or in the cloud and stores the data generated by the IoT device.
- **Web Service:** Web services serve as a link between the IoT device, application, database and analysis components. Web service can be implemented using HTTP and REST principles (REST service) or using the WebSocket protocol (WebSocket service).
- **Analysis Component:** This is responsible for analyzing the IoT data and generating results in a form that is easy for the user to understand.
- **Application:** IoT applications provide an interface that the users can use to control and monitor various aspects of the IoT system. Applications also allow users to view the system status and the processed data.

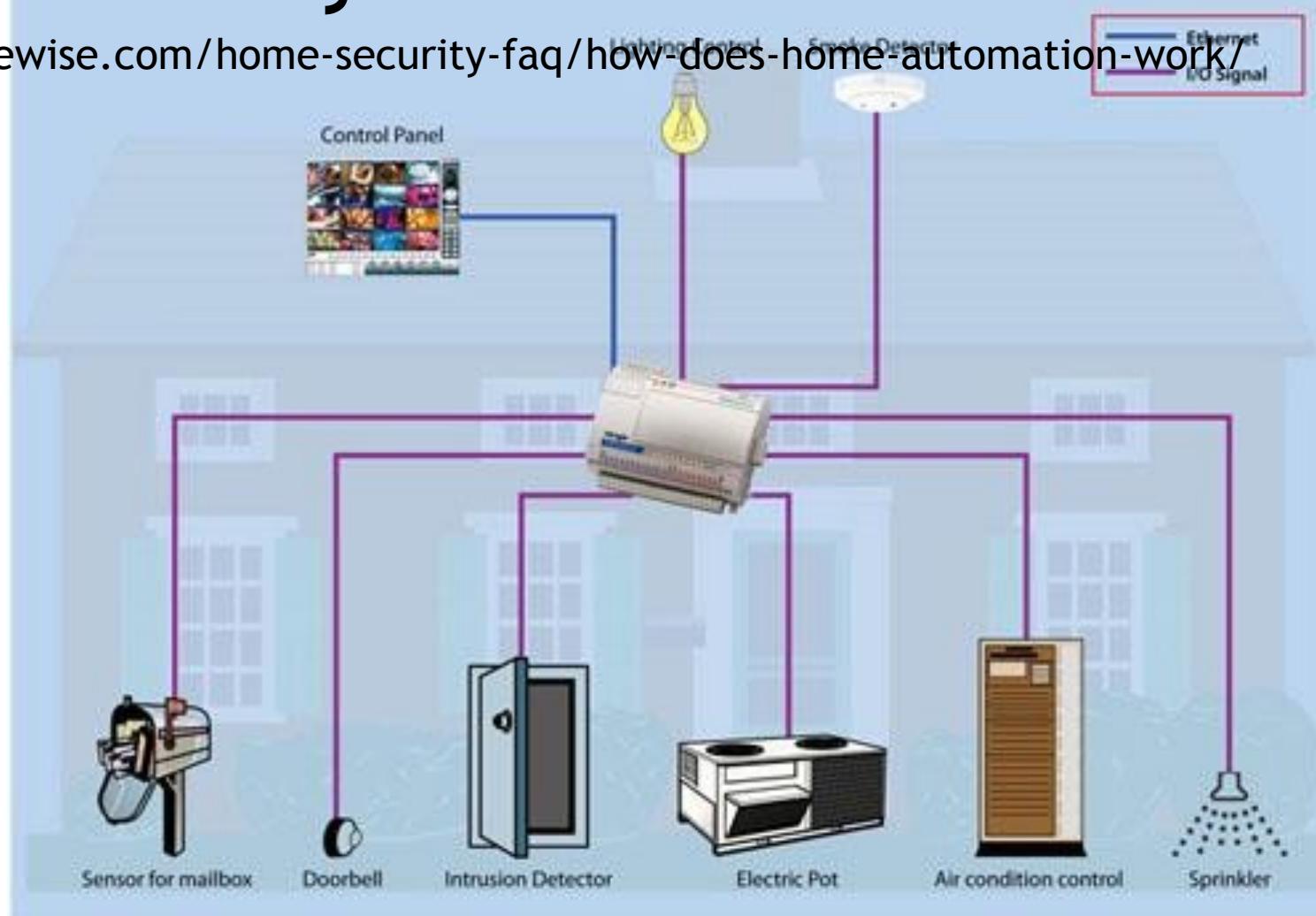
IoT Level-1

- A level-1 IoT system **has a single node/device** that performs sensing and/or actuation, stores data, performs analysis and hosts the application.
- Level-1 IoT systems are suitable for **modelling low-cost and low-complexity solutions** where the data involved is not big and the **analysis requirements are not computationally intensive**.



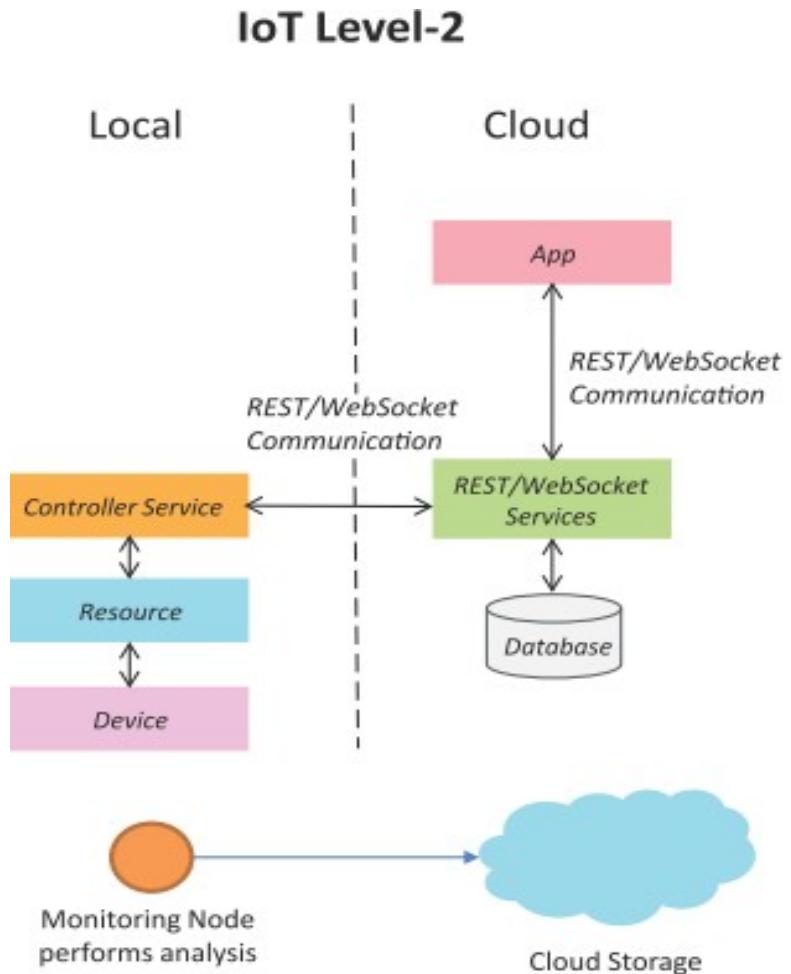
IoT - Level 1 Example ...Home Automation System

<https://www.safewise.com/home-security-faq/how-does-home-automation-work/>



IoT Level-2

- A level-2 IoT system has a **single node that performs sensing and/or actuation and local analysis.**
- **Data is stored in the cloud** and the application is usually cloud-based.
- Level-2 IoT systems are **suitable for solutions where the data involved is big;** however, the primary **analysis requirement is not computationally intensive** and can be done locally.

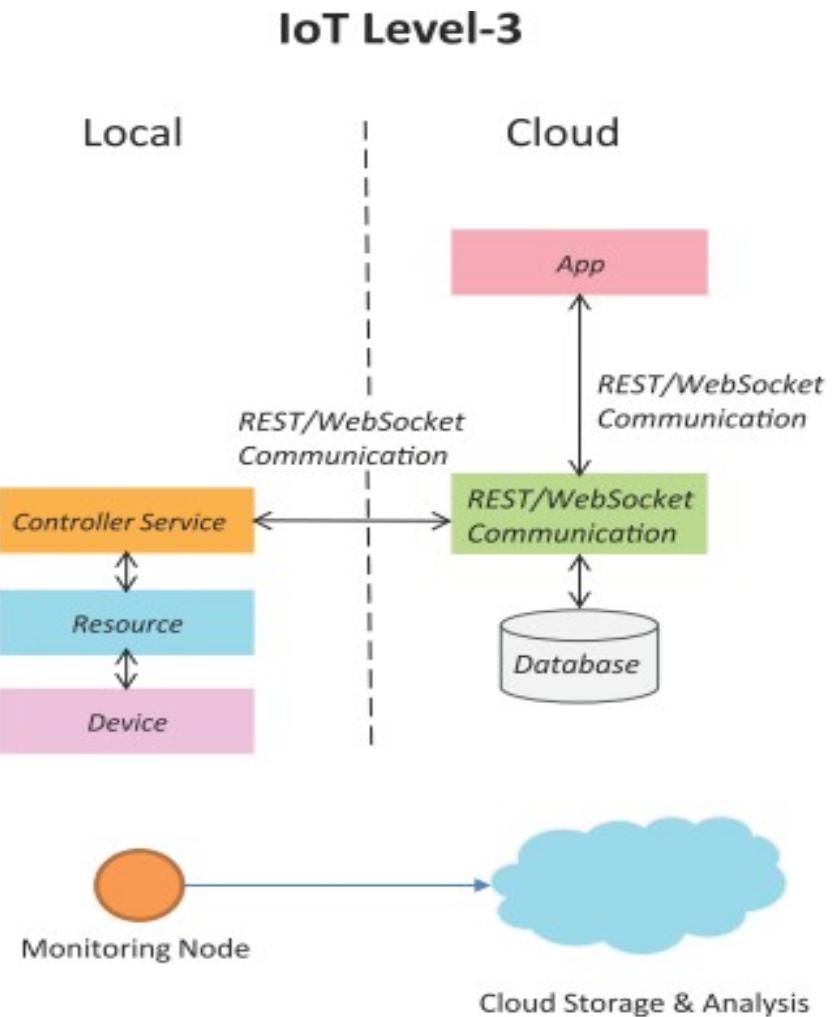


IoT - Level 2 Example ...Smart Irrigation



IoT Level-3

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



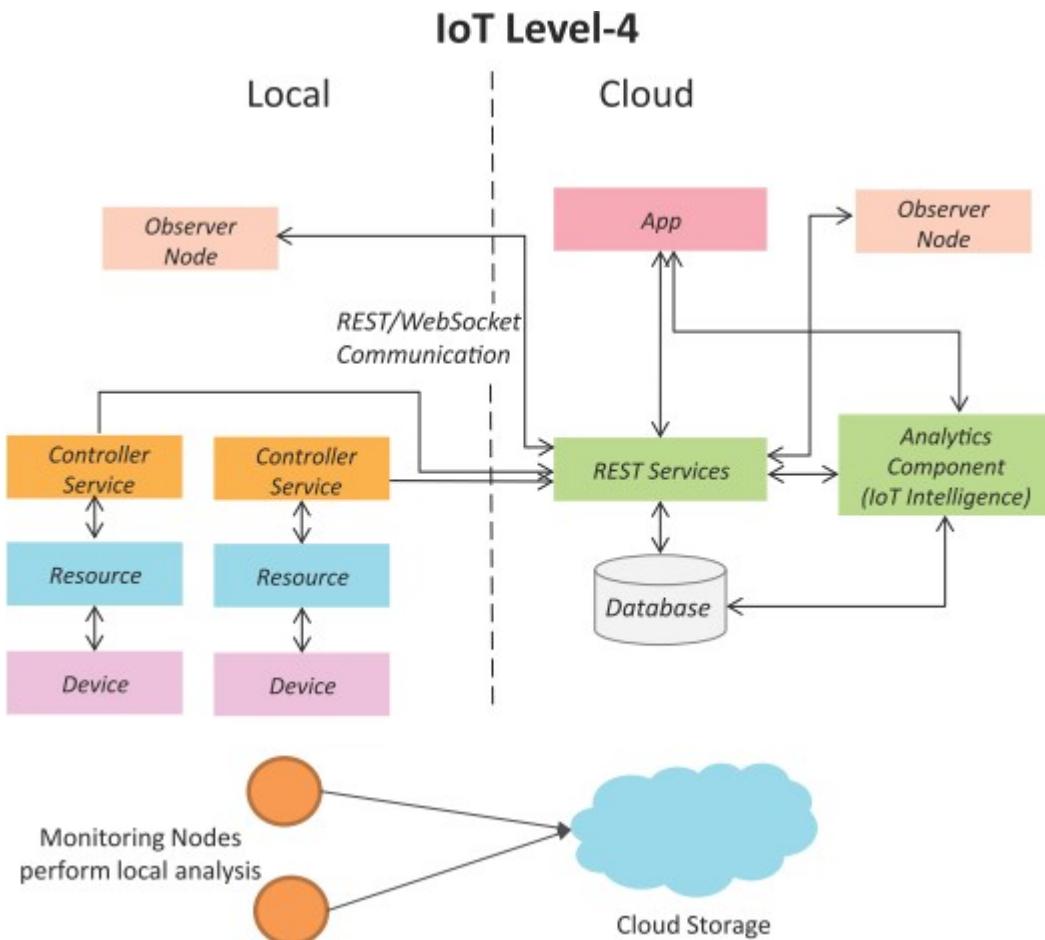
IoT - Level 3 Example ...Tracking Package Handling

Sensors used accelerometer and gyroscope



IoT Level-4

- A level-4 IoT system has multiple nodes that perform local analysis. Data is stored in the cloud and the application is cloud-based.
- Level-4 contains local and cloud-based observer nodes which can subscribe to and receive information collected in the cloud from IoT devices.
- 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.



IoT - Level 4 Example ...Noise

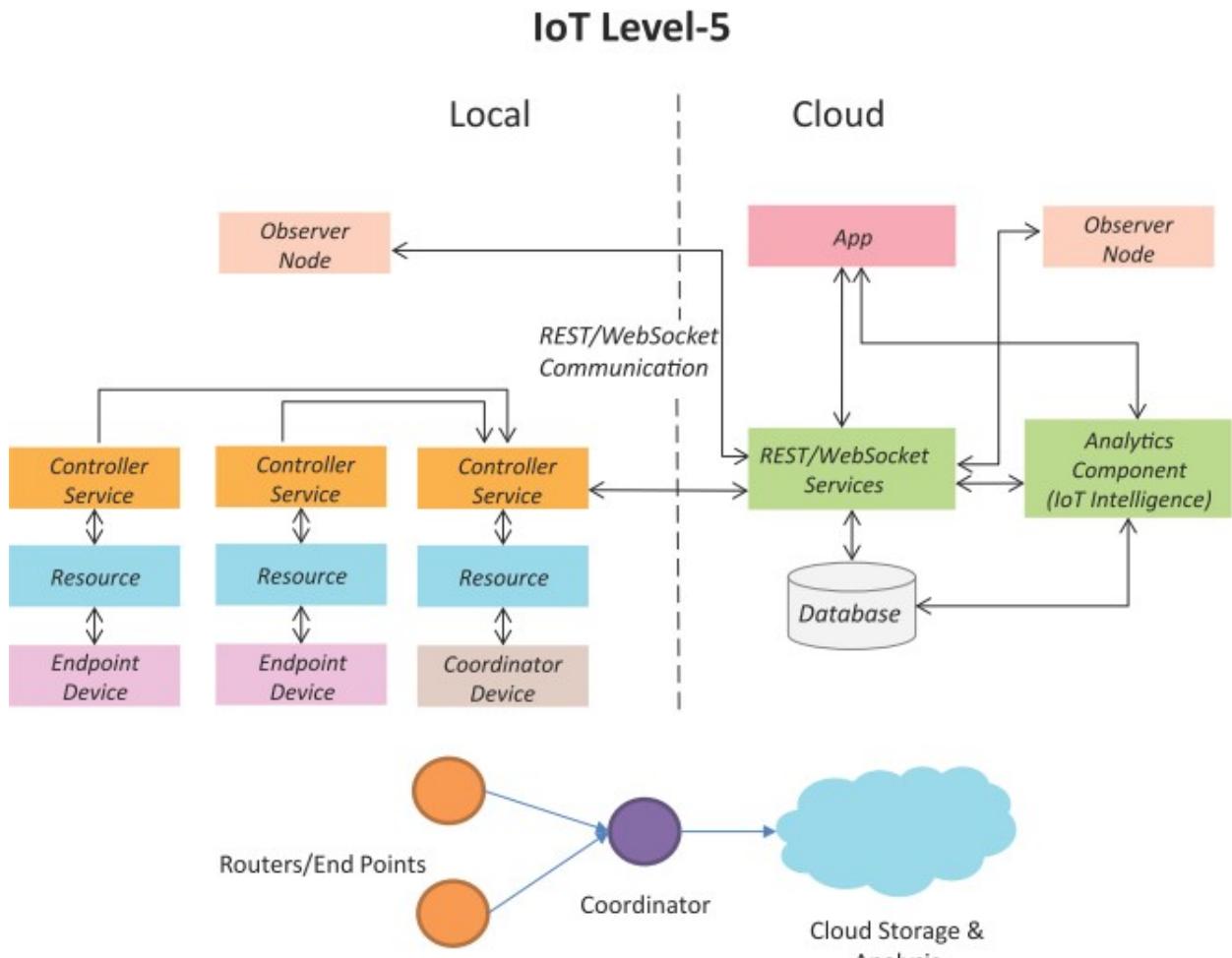
Monitoring

Sound Sensors are used



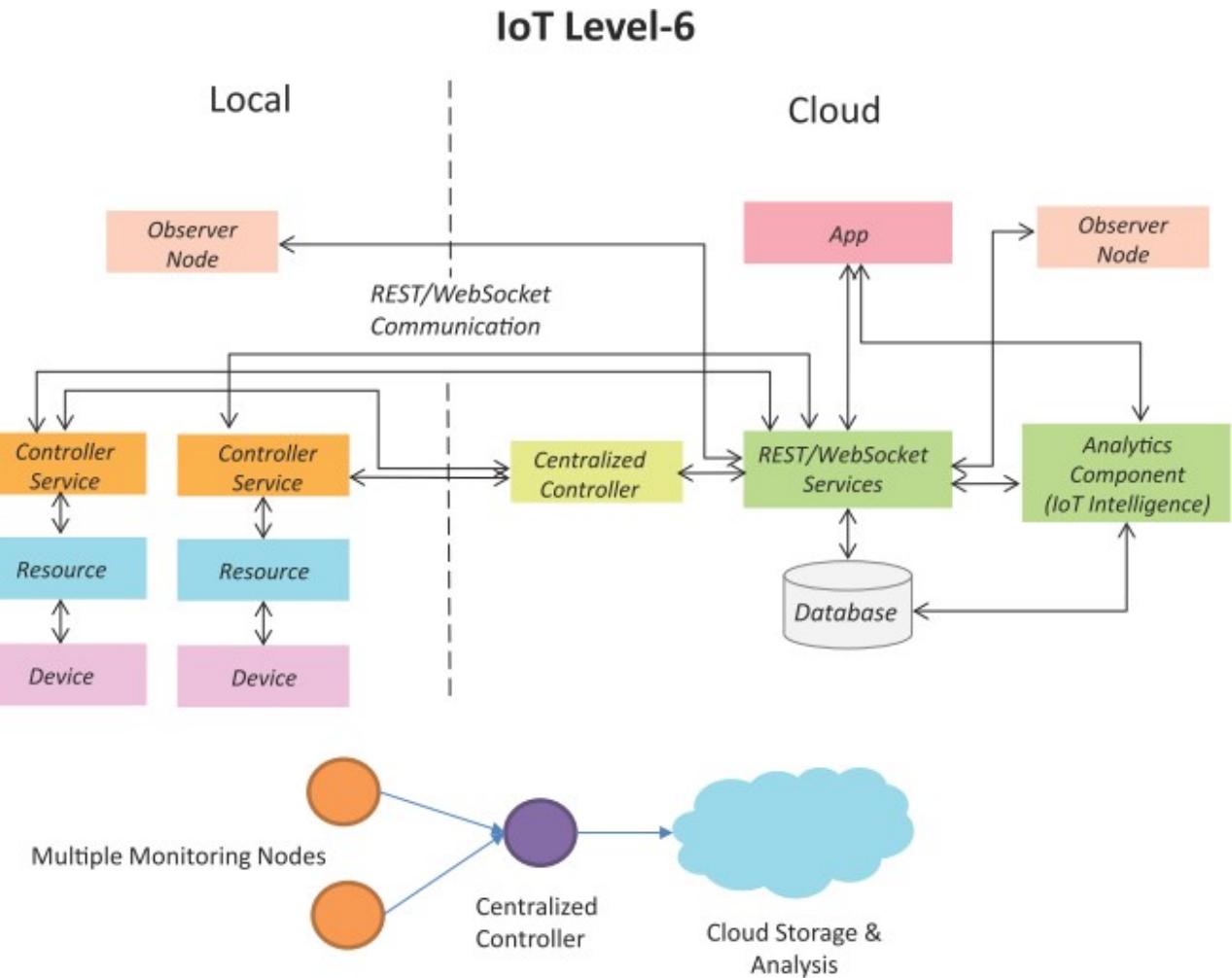
IoT Level-5

- A level-5 IoT system has multiple end nodes and one coordinator node.
- The end nodes perform sensing and/or actuation.
- The coordinator node collects data from the end nodes and sends it to the cloud.
- Data is stored and analyzed in the cloud and the application is cloud-based.
- Level-5 IoT systems are suitable for solutions based on wireless sensor networks, in which the data involved is big and the analysis requirements are computationally intensive.



IoT Level-6

- A level-6 IoT system has multiple independent end nodes that perform sensing and/or actuation and send data to the cloud.
- Data is stored in the cloud and the application is cloud-based.
- The analytics component analyzes the data and stores the results in the cloud database.
- The results are visualized with the cloud-based application.
- The centralized controller is aware of the status of all the end nodes and sends control commands to the nodes.



IoT Issues and Challenges

Security

- Cyber Attacks, Data Theft

Privacy

- Controlling access and ownership of data.

InterOperability

- Integration Inflexibility

Legality and Rights

- Data Protection laws be followed, Data Retention and destruction policies

Economy and Development

- Investment Incentives, Technical Skill Requirement

Emerging Trends of IoT

Artificial Intelligence

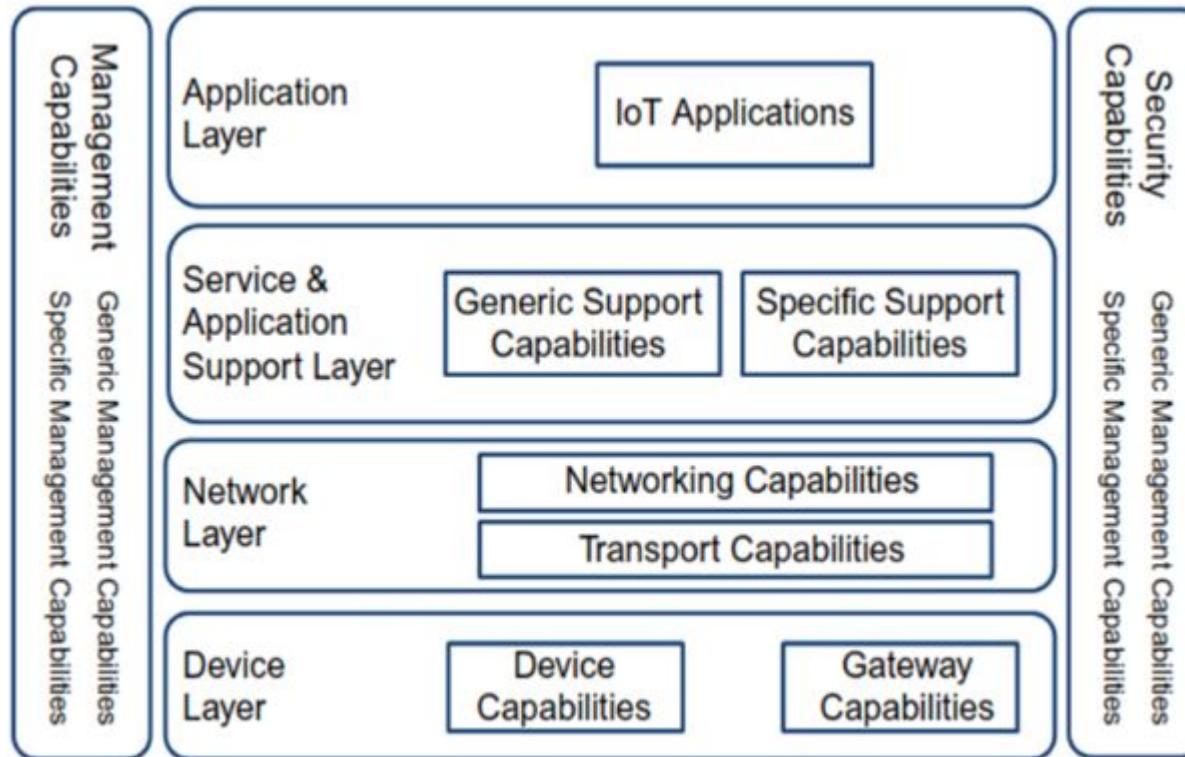
Block chain

Machine Learning

Data Analytics

IoT architecture

Architecture Reference Model

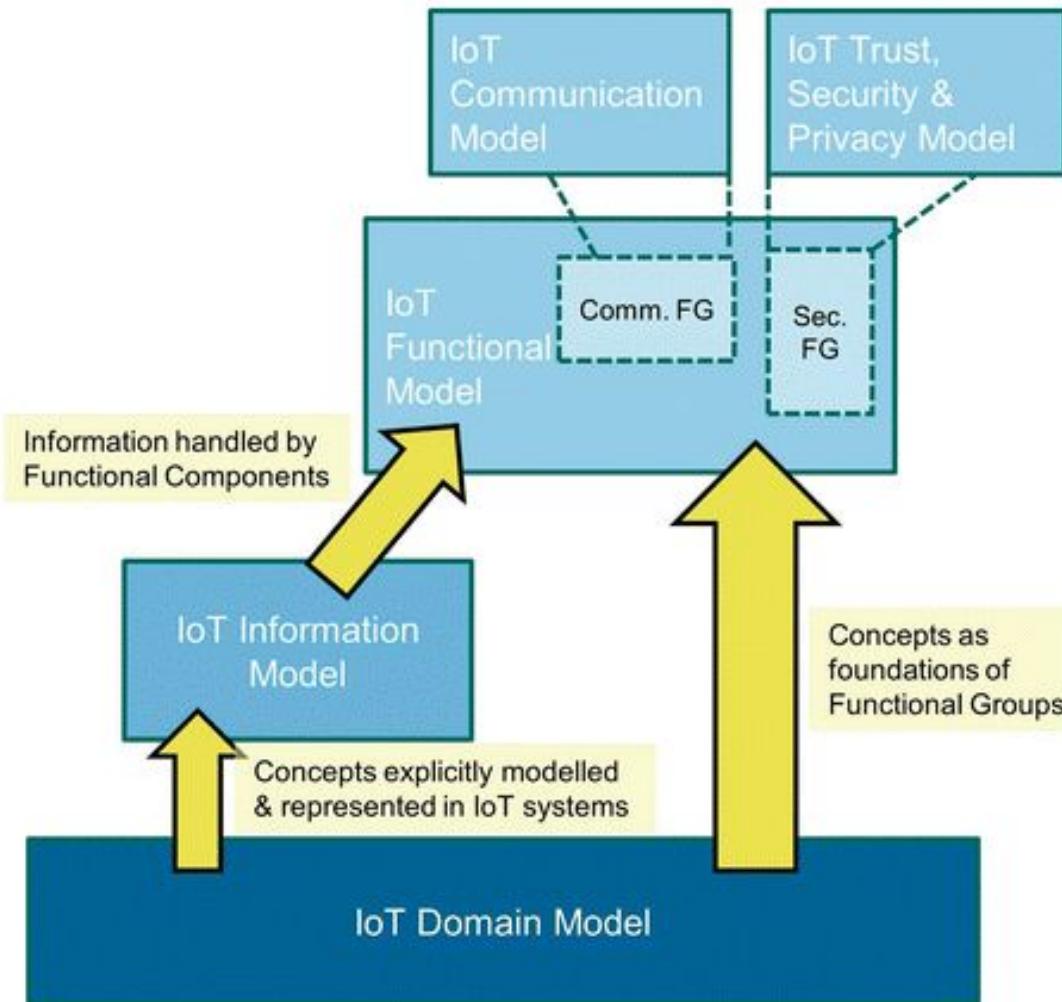


ITU-T IoT Reference Model P

Reference Model and Architecture

- An ARM consists of two main parts: a Reference model and a Reference Architecture.
- A reference model describes the domain using a number of sub-models

Reference Model



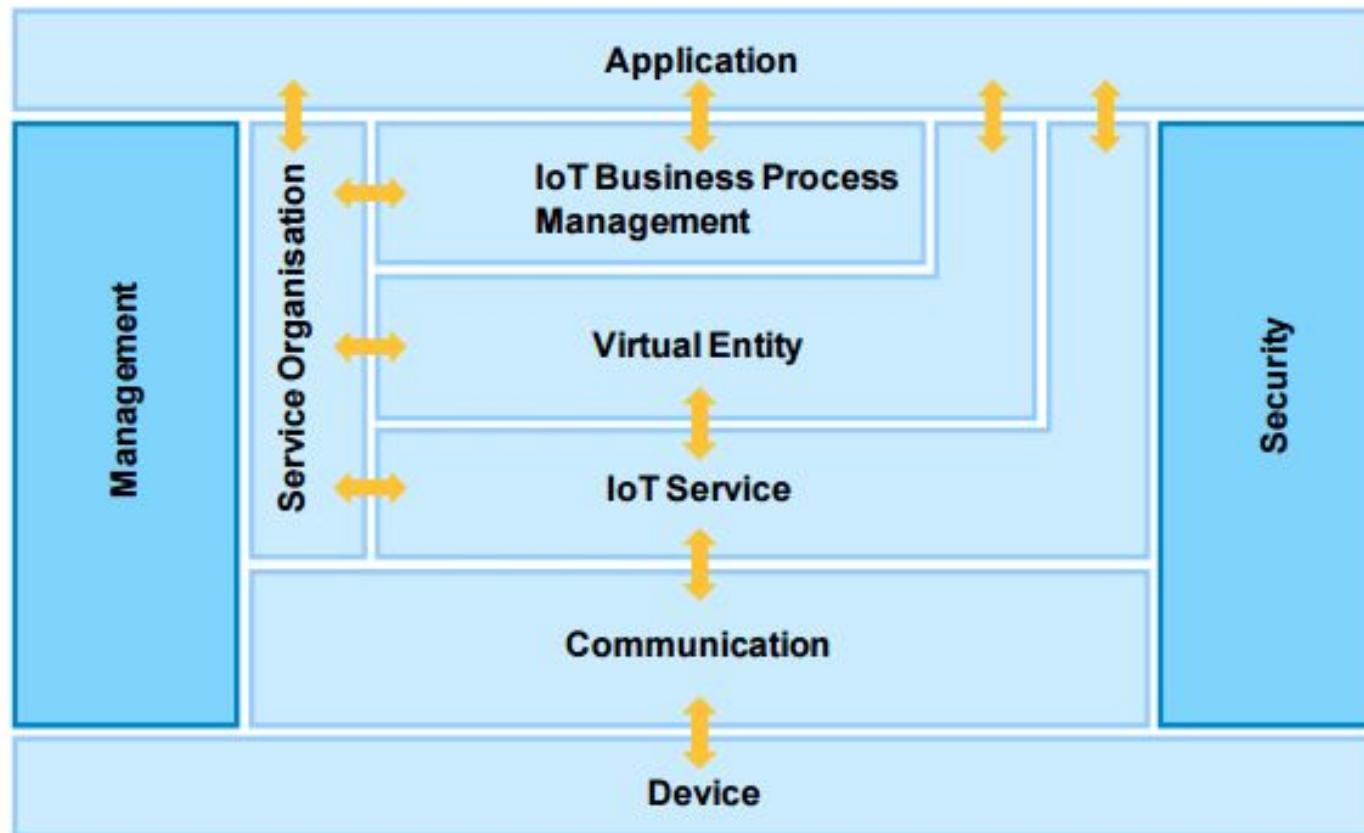
IOT Reference Model

IoT domain model

- The IoT-A project defines a domain model as **a description of concepts belonging to a particular area of interest**. The domain model also defines basic attributes of these concepts, such as name and identifier.
- **For the IoT Domain Model, three kinds of Device types are the most important:**
- Sensors: These are simple or complex Devices that typically involve a transducer that converts physical properties such as temperature into electrical signals.
- Actuators
- Tags

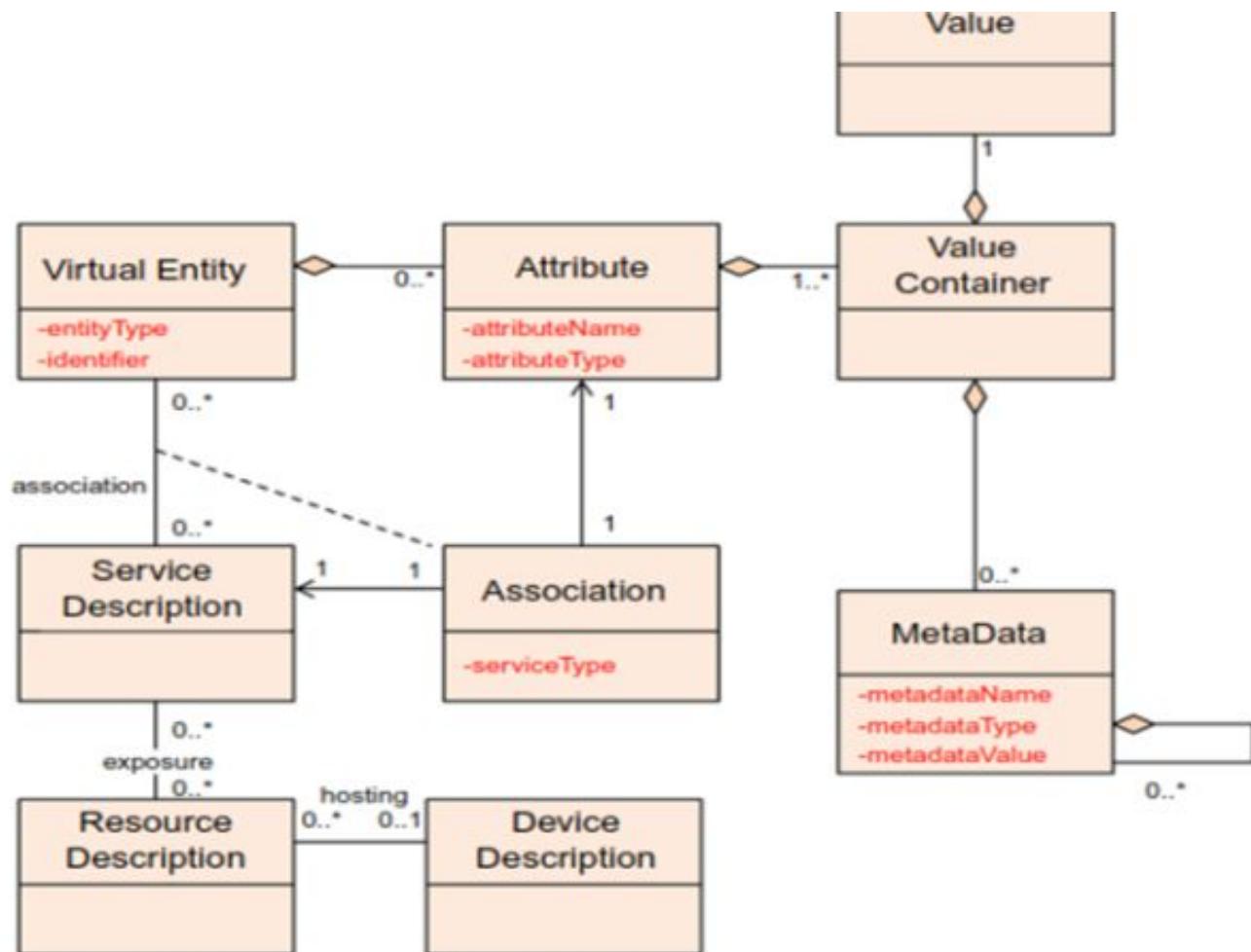
Functional model

The IoT Functional Model aims at describing mainly the Functional Groups (FG) and their interaction with the ARM, while the Functional View of a Reference Architecture describes the functional components of an FG, interfaces, and interactions between the components. The Functional View is typically derived from the Functional Model in conjunction with high-level requirements.



- ***Information Model***

- Virtual Entity in the IoT Domain Model is the “Thing” in the Internet of Things, the IoT information model captures the details of a Virtual Entity- centric model. Similar to the IoT Domain Model, the IoT Information Model is presented using Unified Modelling Language (UML) diagrams.



High-level IoT Information Model

Communication model

Safety

- the IoT Reference Model can only provide IoT-related guidelines for ensuring a safe system to the extent possible and controllable by a system designer.
Eg: smart grid.

Privacy

- Because interactions with the physical world may often include humans, protecting the User privacy is of utmost importance for an IoT system. The IoT-A Privacy Model depends on the following functional components: Identity Management, Authentication, Authorisation, and Trust & Reputation

Trust

- Generally, an entity is said to ‘trust’ a second entity when the first entity makes the assumption that the second entity will behave exactly as the first entity expects.”

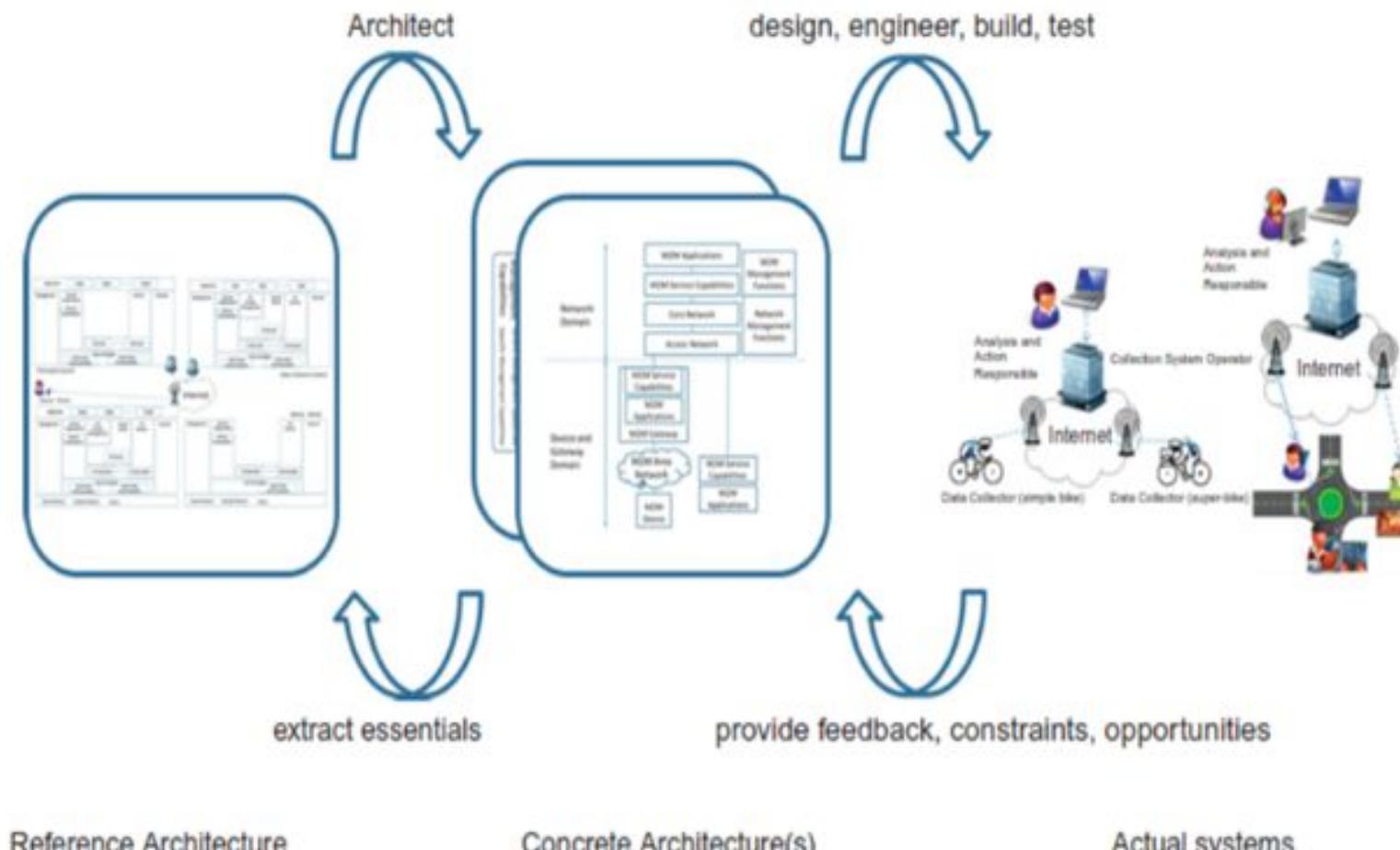
Security

- The Security Model for IoT consists of communication security that focuses mostly on the confidentiality and integrity protection of interacting entities and functional components such as Identity Management, Authentication, Authorisation, and Trust & Reputation.

Reference Architecture

Introduction

- The Reference Architecture is a starting point for generating concrete architectures and actual systems
- concrete architectures- concerns of multiple stakeholders of the actual system
- Views are useful for reducing the complexity of the Reference Architecture blueprints by addressing groups of concerns one group at a time



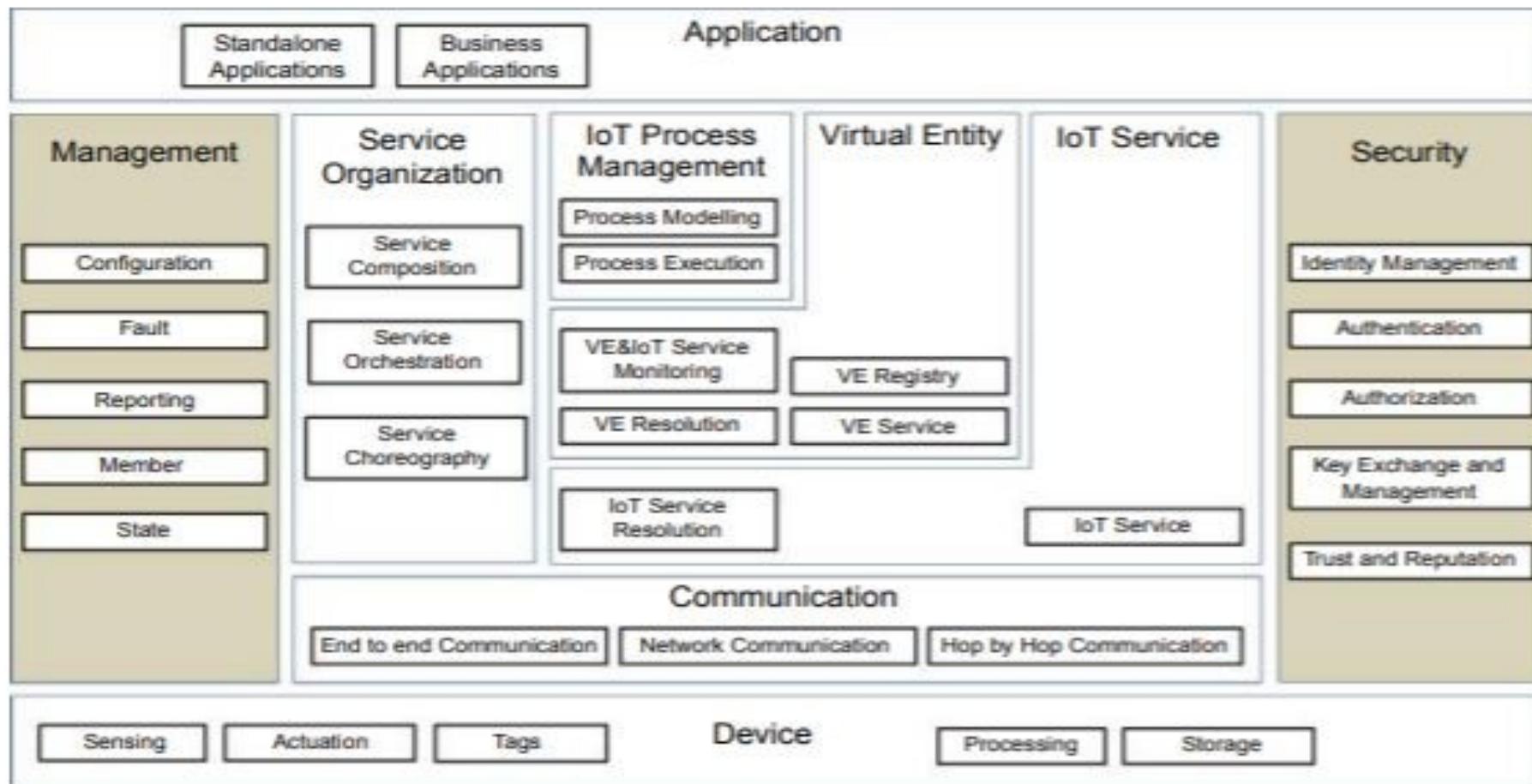
From Reference to concrete architecture and actual system

- Reference Architecture as a set of architectural views
- Functional View: Description of what the system does, and its main functions.
- Information View: Description of the data and information that the system handles.
- Deployment and Operational View: Description of the main real world components of the system such as devices, network routers, servers, etc

Functional view

- It consists of the Functional Groups (FGs) presented earlier in the IoT Functional Model, each of which includes a set of Functional Components (FCs).
- FCs are used in a concrete IoT architecture, and therefore the actual system
- It consists of
 - ✓ Device and application functional group
 - ✓ Communication functional group .
 - ✓ IoT Service functional group
 - ✓ Virtual Entity functional group
 - ✓ IoT process management functional group .
 - ✓ Service Organization functional group
 - ✓ Security functional group

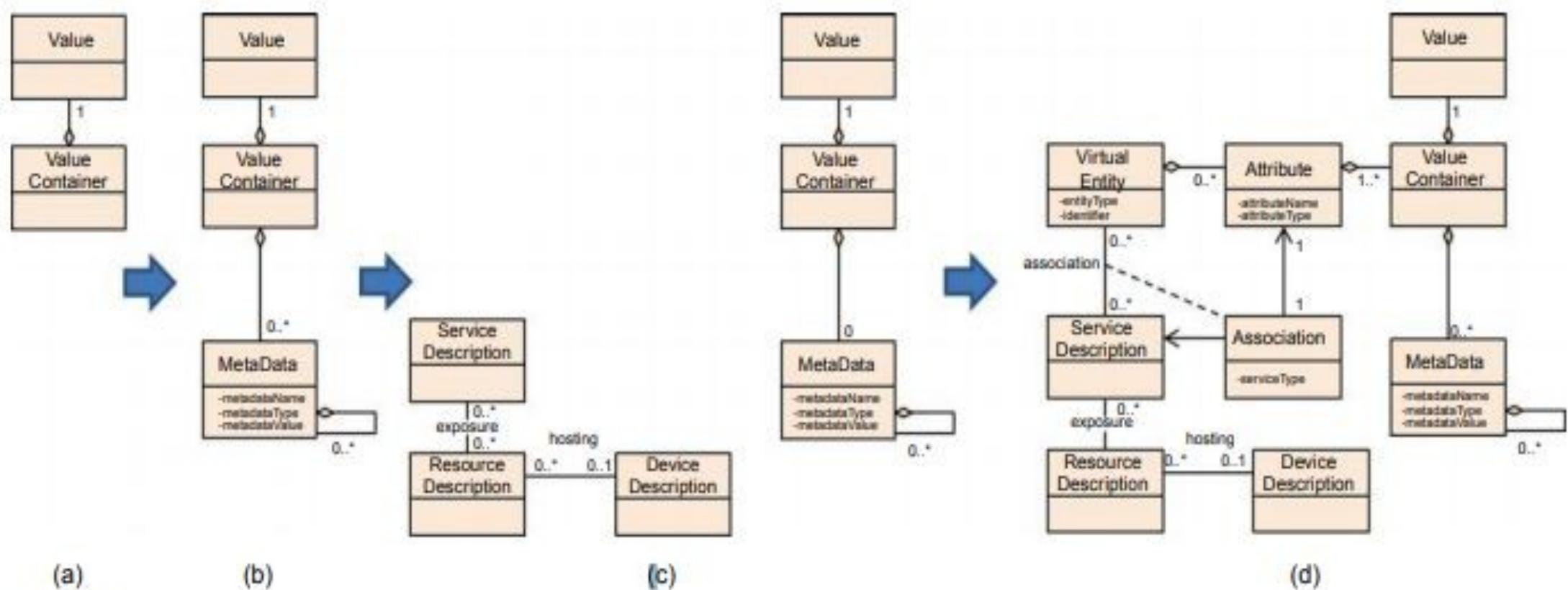
IoT Functional View



Information view

- The information view consists of
 - (a)the description of the information handled in the IoT System
 - (b)the way this information is handled in the system; in other words, the information lifecycle and flow
- It consists of
 - ✓ Information description
 - ✓ Information flow and lifecycle
 - ✓ Information handling

Information Enrichment Process



- Above figure shows the devices equipped with sensors transform changes in the physical properties of the Physical Entities of Interest into electrical signals.
- These electrical signals are transformed in one or multiple values (Figure a) on the device level.
- These values are then enriched with metadata information such as units of measurement, timestamp, and possibly location information (Figure b).
- These enriched values are offered by a software component (Resource) either on the device or the network. The Resource exposes certain IoT Services to formalize access to this enriched information (Figure c).

Deployment and operational view

- The Deployment and Operational View depends on the specific actual use case and requirements.
- Example- Parking Lot
- Below figure depicts the Devices view as Physical Entities deployed in the parking lot, as well as the occupancy sign.
- There are two sensor nodes each of which are connected to eight metal/car presence sensors.
- The payment station acts both as a user interface for the driver to pay and get a payment receipt as well as a communication gateway that connects the two sensor nodes
- The occupancy sign also acts as a communication gateway for the actuator node.

Parking Lot Deployment and Operational View, Devices.

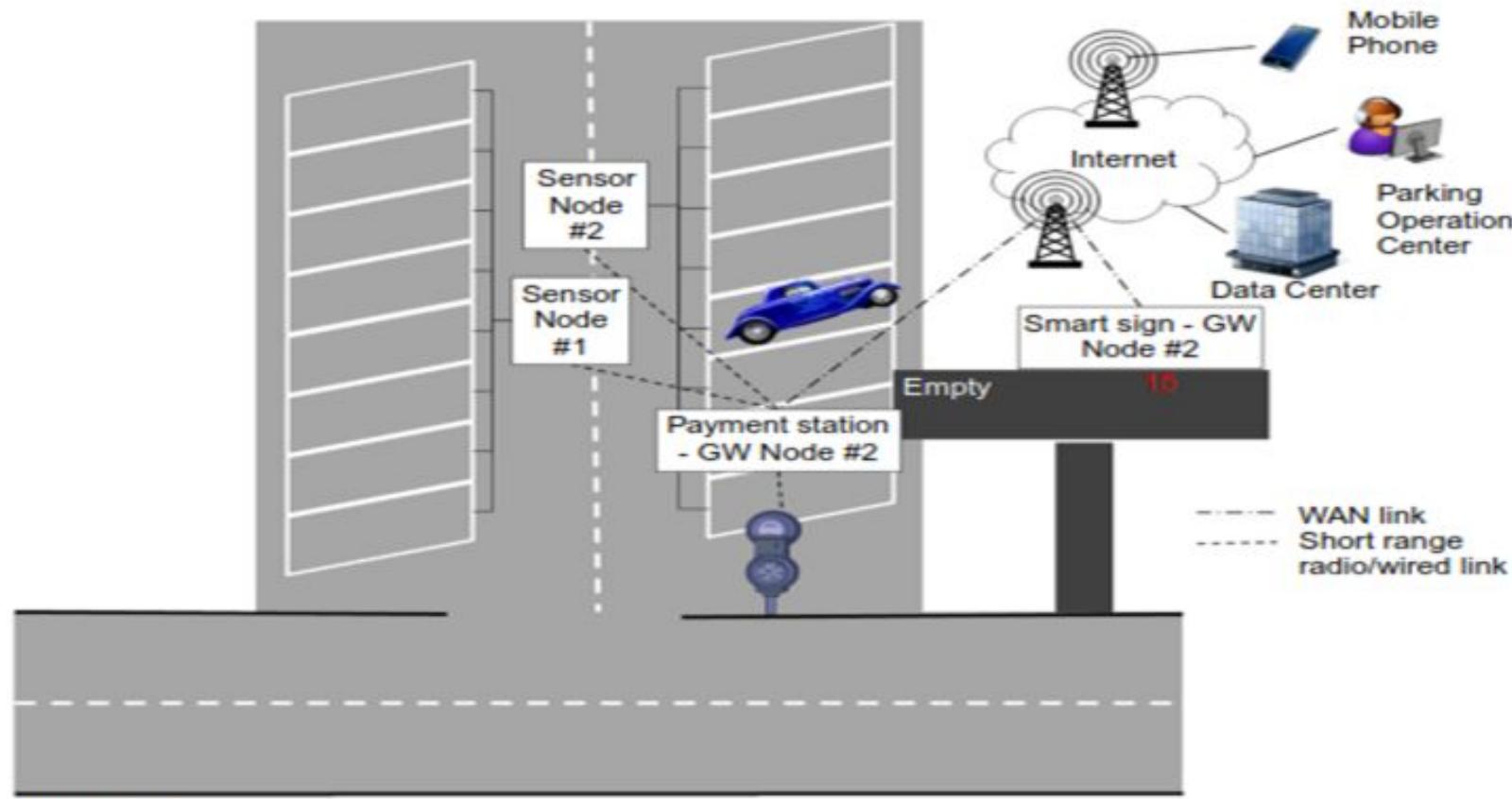


FIGURE 8.8

Parking Lot Deployment and Operational View, Devices.

Parking Lot Deployment & Operational View, Resources, Services, Virtual Entities, Users

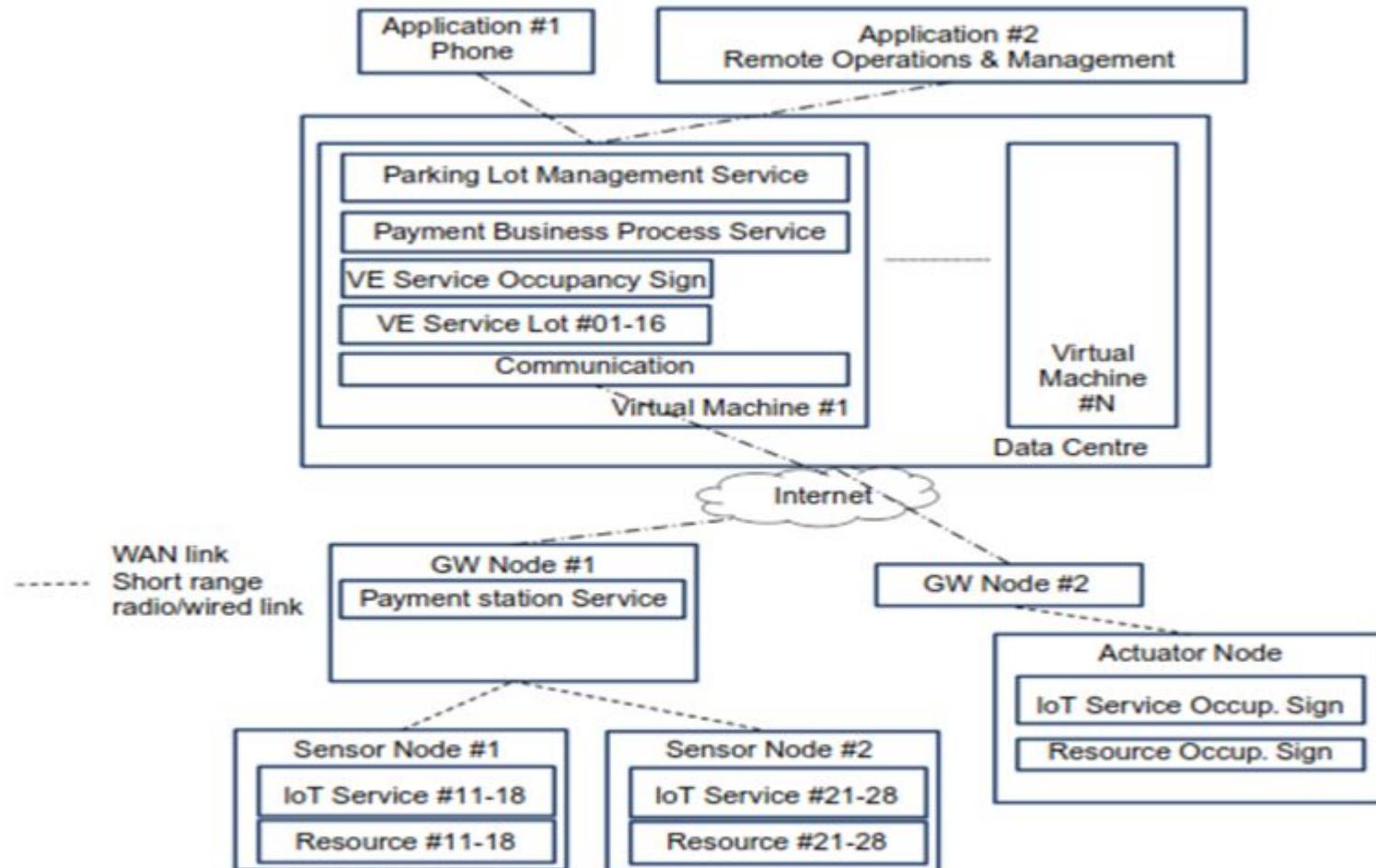
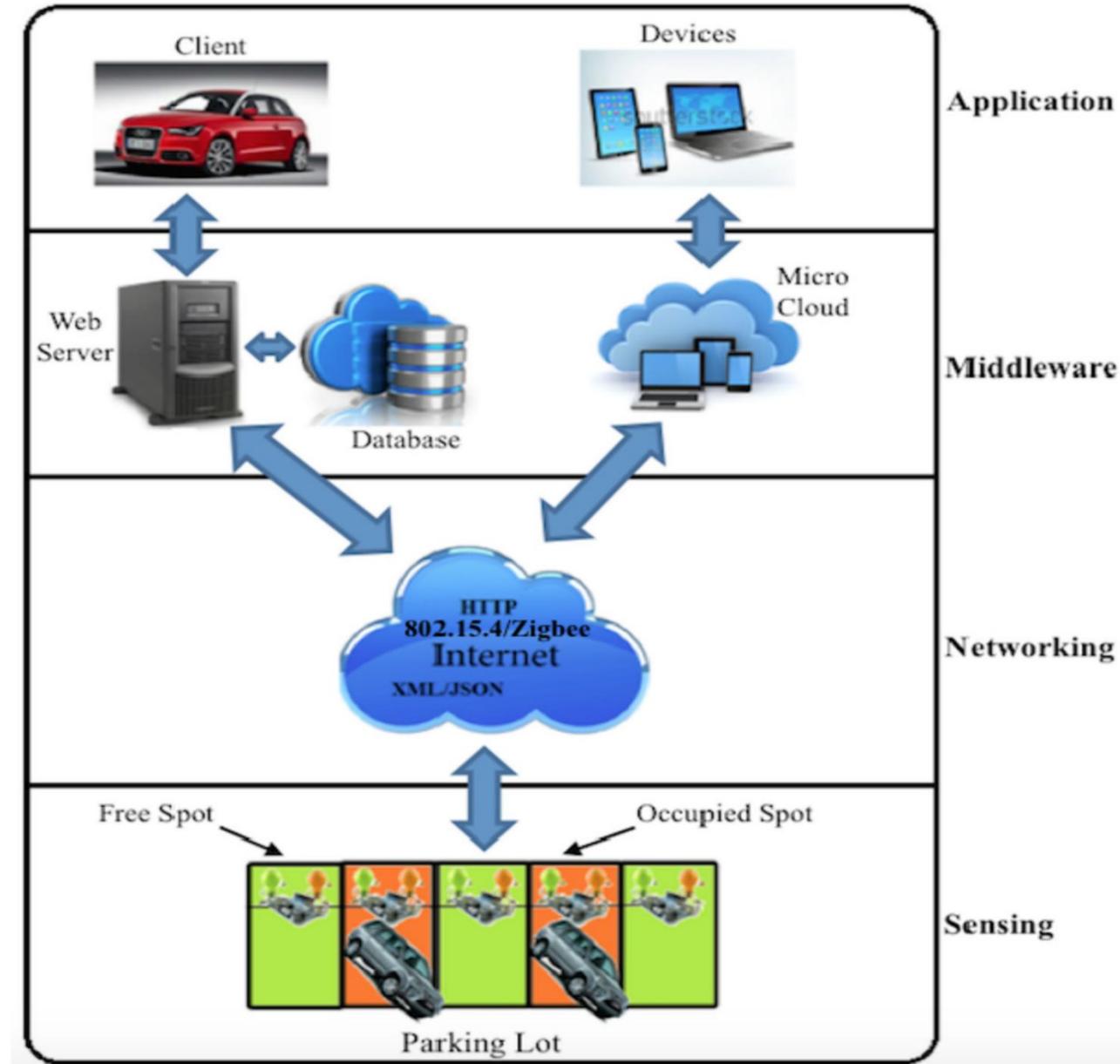


FIGURE 8.9

Parking Lot Deployment & Operational View, Resources, Services, Virtual Entities, Users.



Unit III

Pillars of Embedded IoT
and Physical Devices

INTERNET OF THINGS
A Hands-On Approach



Outline

- Horizontal and Vertical Applications of IoT
- Four Pillars of IoT
- M2M : Internet of Devices
- RFID : Internet of Objects
- WSN : Internet of transducer
- SCADA : Internet of Controllers
- DCM :
 - Device : Things that talk
 - Connect : Pervasive Network
 - Manage : Create Business Values

Outline

- IoT Physical Devices and Endpoints
 - Basic building blocks of IoT device
- Exemplary device: Raspberry Pi
- Raspberry Pi interfaces
- Programming Raspberry Pi with Python
- Beagle board and other IoT Devices

Horizontal Applications

- Provide solution to common problems
- These are not business specific
- Can be Used monitored and controlled by multiple companies
- Allows multiple providers to work together on a single platform

Advantages

- Robust
- Developed fast and less costly

Vertical Applications

- These are business specific
- Can be used monitored and controlled by only single company
- Does not allow multiple providers to work together on a single platform

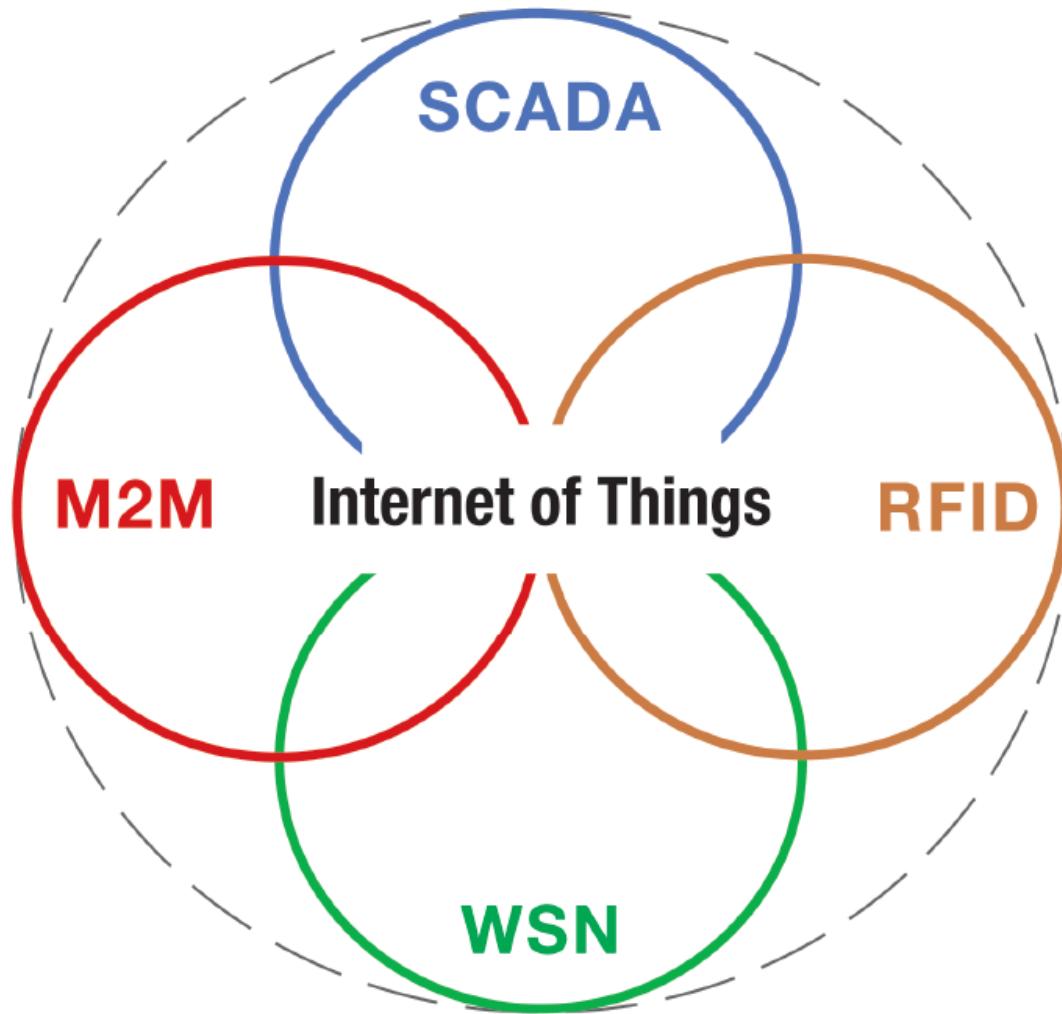
Advantages

- No compatibility issue as no other companies are involved

Disadvantages

- Depended entirely on a single vendor for modifications or upgrades

Four Pillars of IoT



M2M

- Machine to Machine
- Enables **flow of data between machines** which **monitors data** by means of **sensors** and at other end **extracts the information on gathered data and processes it.**
- Subset of IoT
- It uses WAN , GPRS, Cellular and Fixed N/w's

M2M Architecture

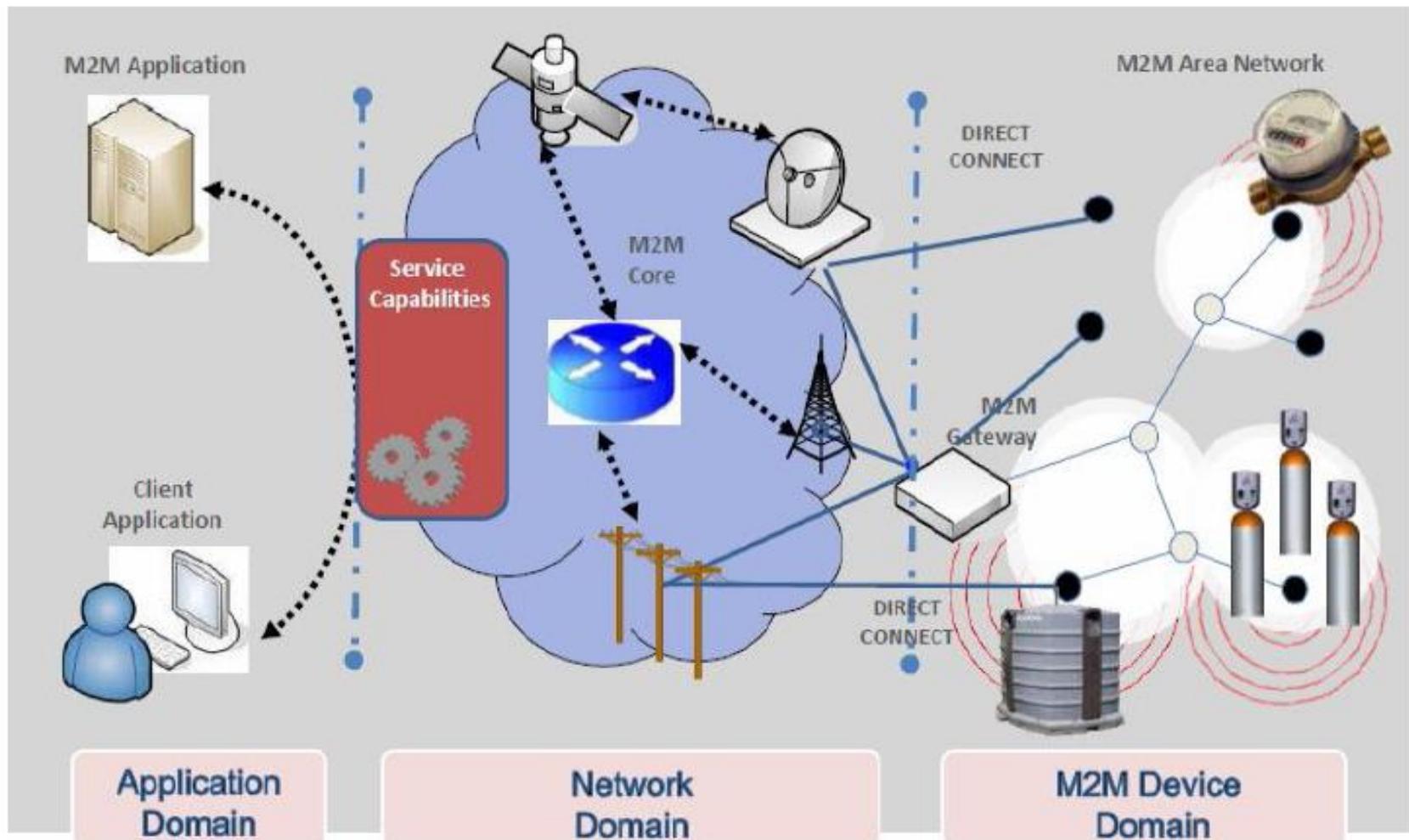


Figure 1: Architecture of M2M system

M2M Architecture

Components of M2M architecture are :

- 1) M2M Devices
- 2) **M2M Area Network i.e Device Domain**
- 3) M2M Gateway
- 4) **M2M Communication N/w's : Network Domain**
- 5) **M2M Applications i.e Application Domain**

M2M Devices

- Device that are capable of replying to request for data contained within those devices or capable of transmitting data autonomously are M2M Devices.
- **Sensors and communication devices** are the endpoints of M2M applications.

M2M Area Network

- Provide connectivity between M2M Devices and M2M Gateways.
- **E.g. Personal Area Network**

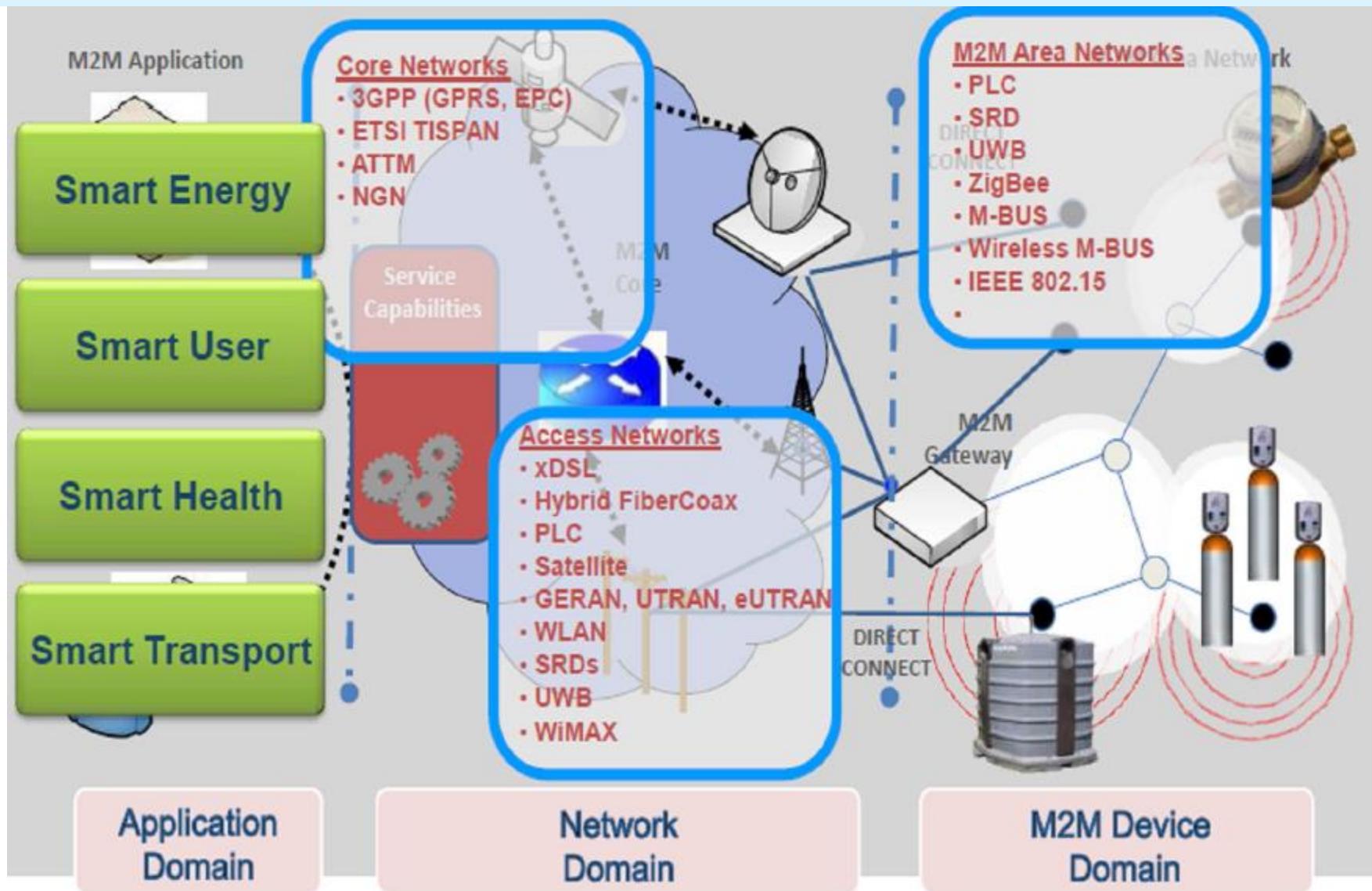
M2M Network Domain

- Communication between M2M Gateways and M2M Applications.
- **E.g. WiMax, WLAN, LTE**

M2M Application Domain

- It contains the middleware layer where data goes through various application services and is used by the specific business-processing engines.

Examples of M2M Components





RFID

- **Radio Frequency Identification**
- Uses **radio frequency** to **read** and capture information stored on a **tag** attached to an object.
- A tag can be read from up to several feet away and does not need to be within direct line-of-sight of the reader to be tracked.
- Uses NFC (Next Field Communication protocol), IC (Integrated Circuit) Cards, Radio Waves

RFID

What is **RFID**

Application of **RFID**

What is inside in **RFID**

How **RFID Works? (Operating Principle)**

What is RFID?



RFID



Objects can be books in library

RFID



Objects can be items in shopping mall

RFID



Objects can be inventory in the warehouse

RFID



Objects can be a car

RFID Vs Bar Code

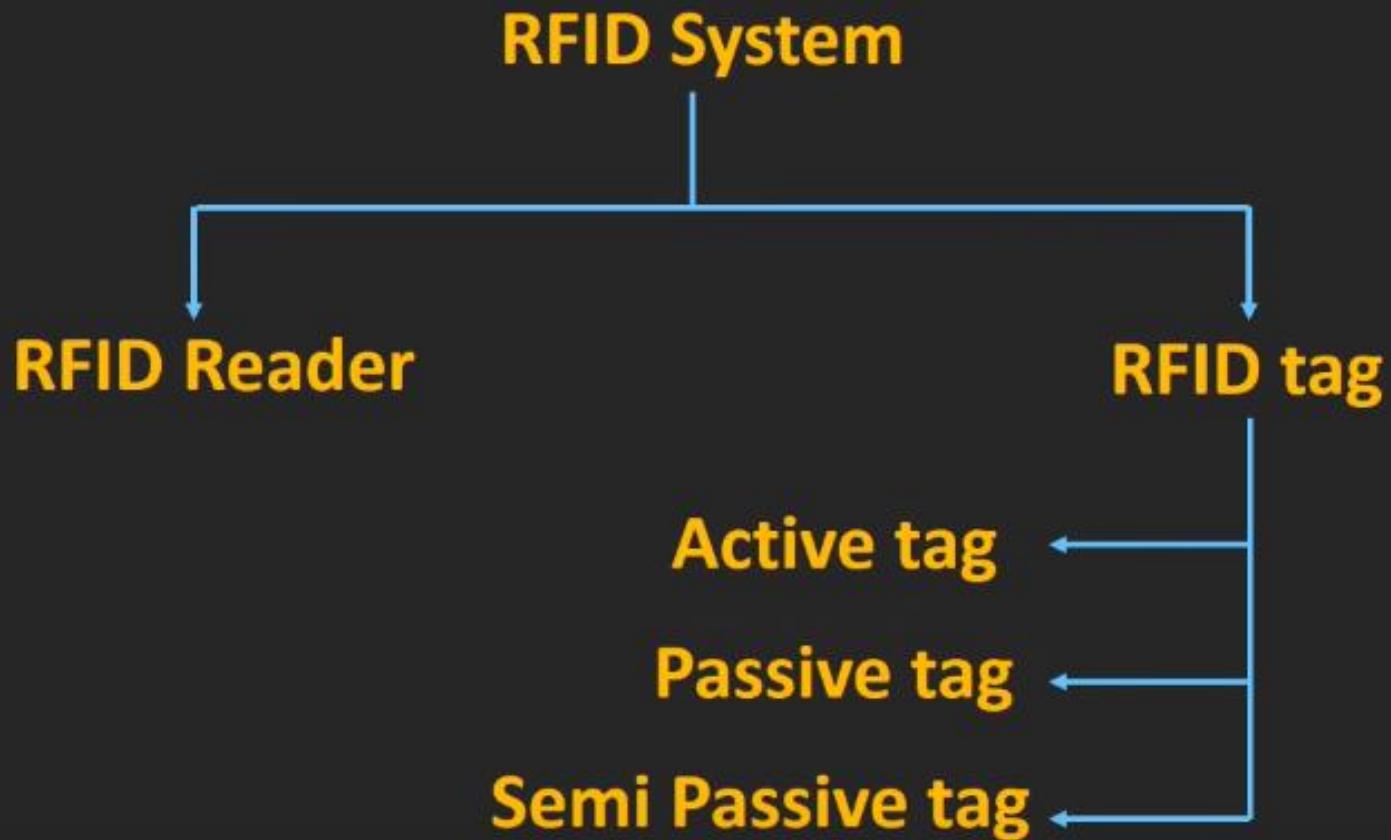


- In Bar code the scanner must be in line of sight...and this is not mandatory in RFID
- RFID can track multiple objects while Bar Code cannot

RFID Vs Bar Code Vs QR Code



RFID System



RFID

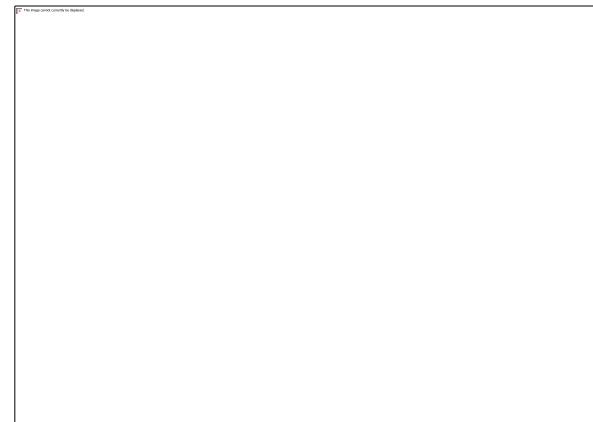
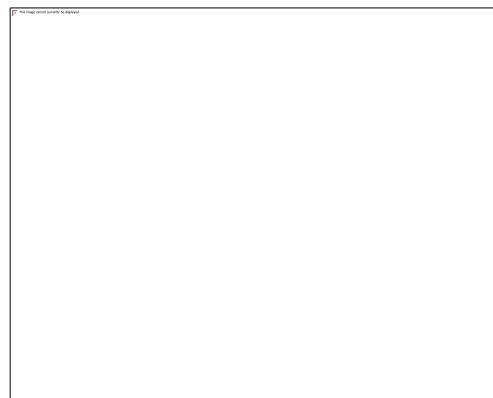
- Passive Tags do not have their own power supply, hence rely on radiowaves for source of energy
- SemiPassive Tag have their own power supply, but for transmitting back they rely on signals coming from RFID Reader
- Active Tag uses their own power supply for both transmitting and receiving
- Range of Passive Tags is less than that of Semi and Active Tags

RFID Tags

- Passive Tags do not have their own power supply, hence rely on radiowaves for source of energy
- SemiPassive Tag have their own power supply, but for transmitting back they rely on signals coming from RFID Reader
- Active Tag uses their own power supply for both transmitting and receiving
- Range of Passive Tags is less than that of Semi and Active Tags

RFID Passive Tags

- Passive Tags are cheaper
- Passive tags do not use any power source hence are compact



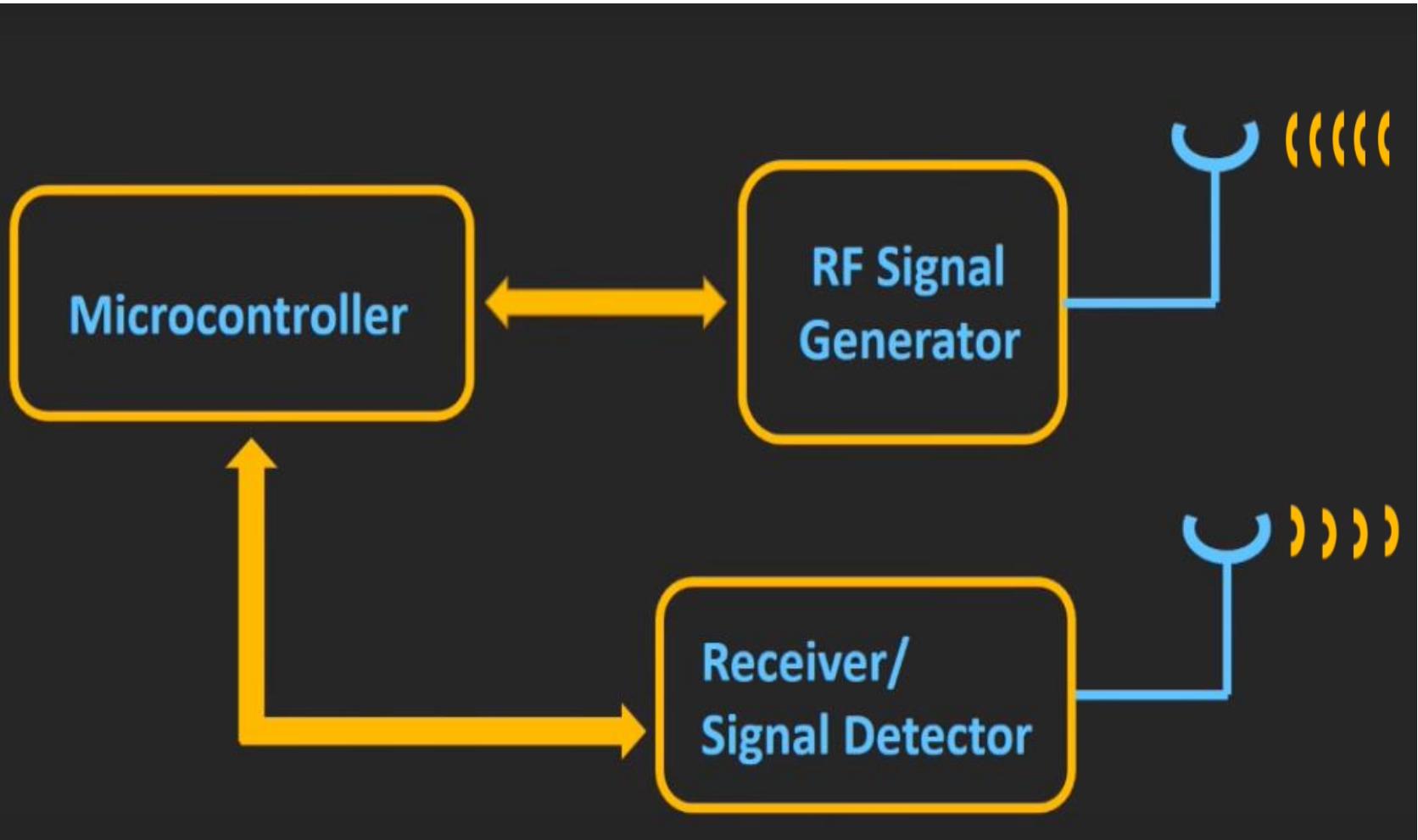
RFID Reader

- Come in many size and shapes



HandHeld Reader

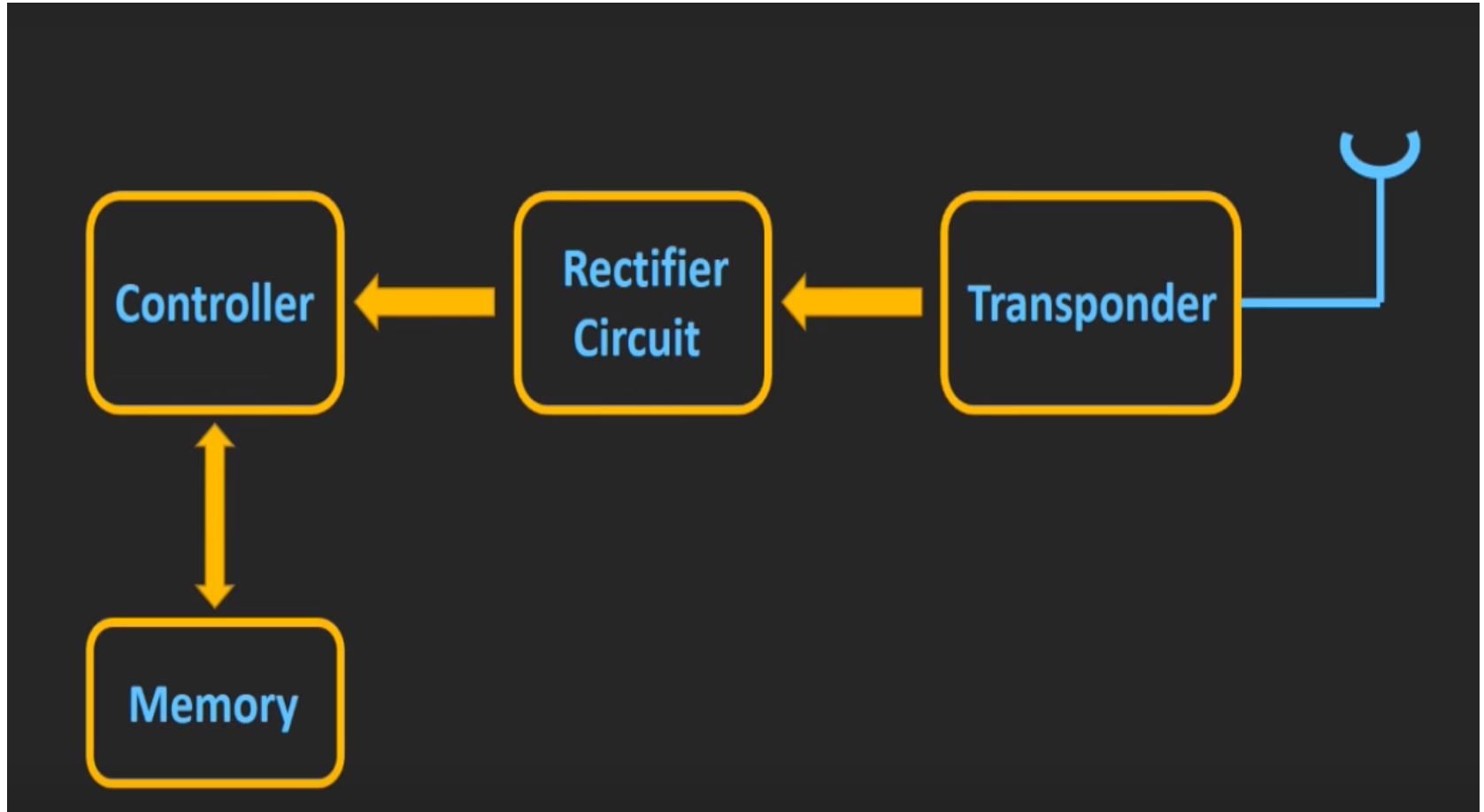
RFID Reader



RFID Reader

- RF Signal Generator Generates radio waves which are transmitted through the antenna
- Receiver or signal detector receives the signals coming from the object
- And to process these signals microcontroller is used

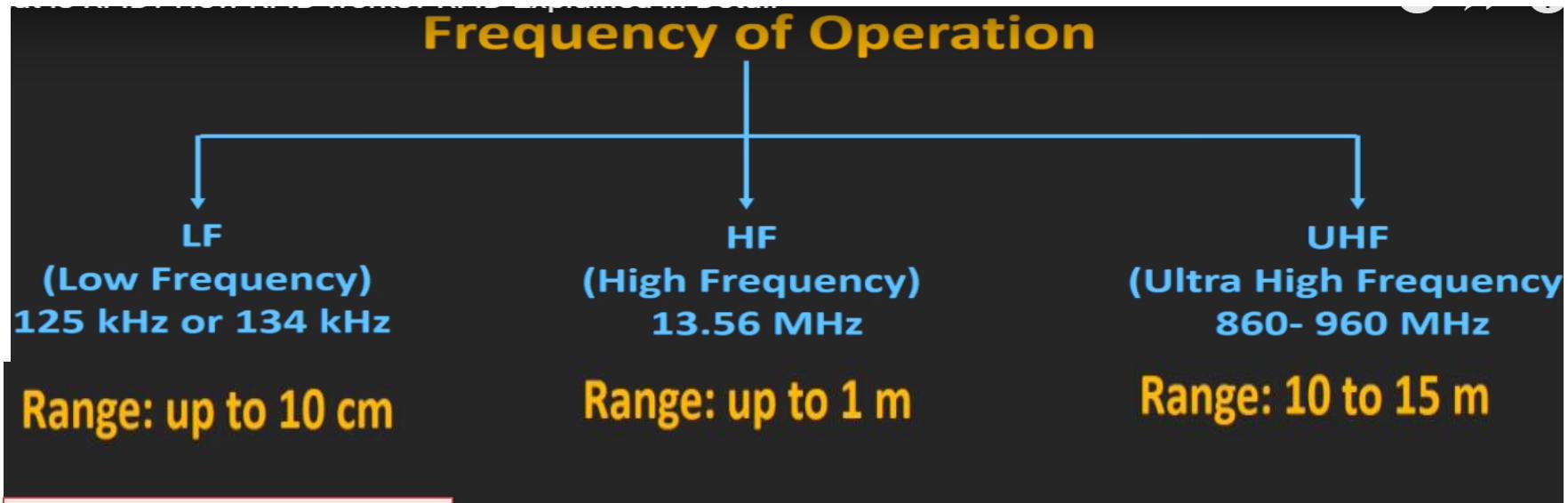
RFID Tag



RFID Tags

- Transponder receives signals from reader and sends back feedback to the reader
- The Passive Tags use the rectifier circuit to store the energy coming from the radio waves.
- This energy is used as the supply for the controller and the memory element

RFID Frequency Operation



Person Identification



Clothes at Shopping Mall



Vehicle Identification at Toll Plaza



RFID Frequency Operation

- Frequency Operation varies from country to country, but major countries

RFID Working Principle

LF and HF RFID Tags: Inductive Coupling (Near Field Coupling)

UHF RFID Tags: Electromagnetic Coupling (Far Field Coupling)

SCADA

- **Supervisory Control and Data Acquisition**
- These connect , monitor and control equipment's using short range n/w inside a building or an industrial plant
- Uses **BacNet** (communication protocol) , **CanBus** (Controller Ara N/w) and **Wired FieldBuses**(Industrial Computer Network Protocols)

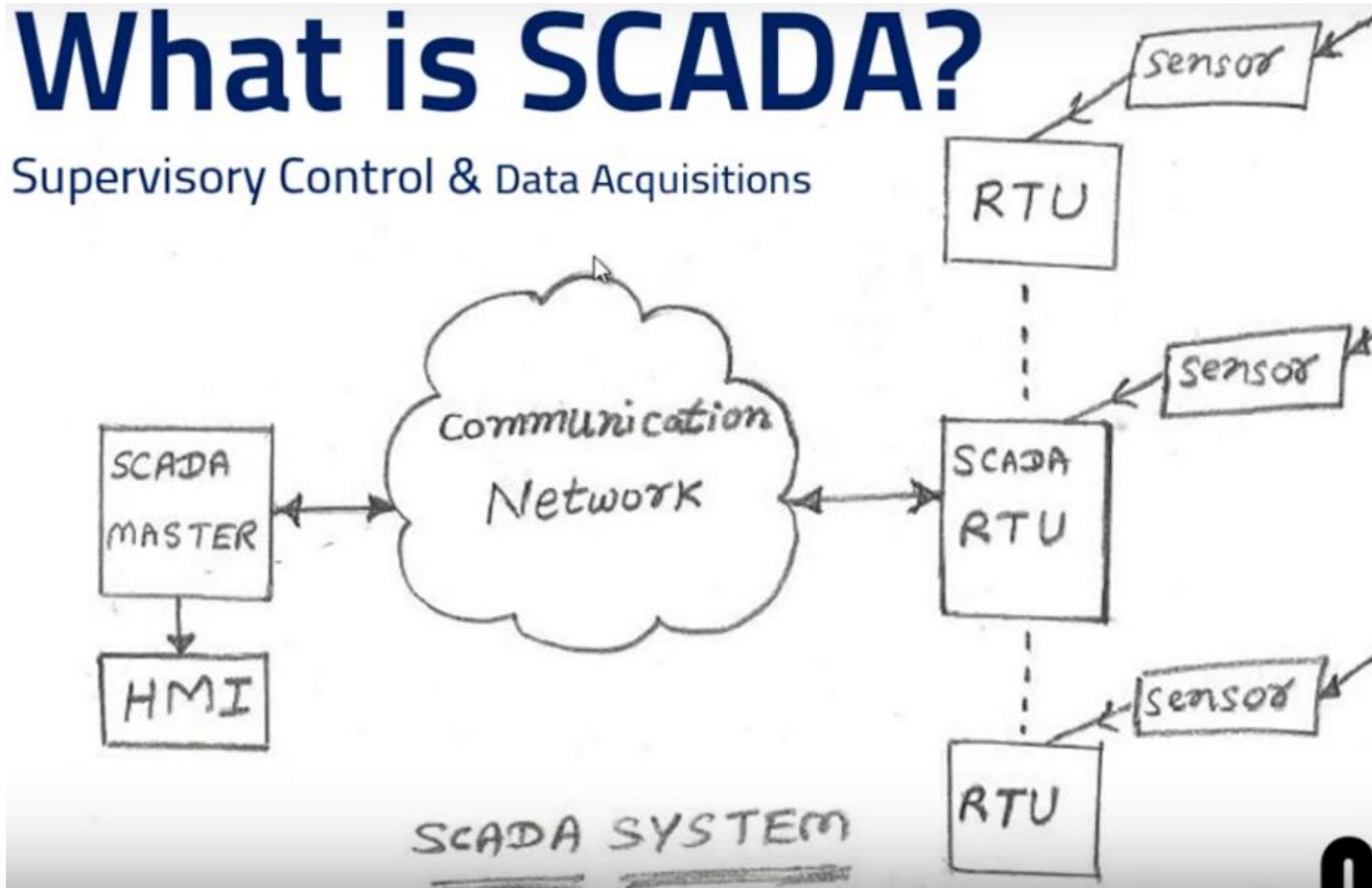
SCADA

- **Supervisory** means top level
- **Control** means controlling things
- **Data Acquisition** means acquiring the data / reading the data
- **SCADA** ia a s/w used to **control** the **hardware** i.e PLC, drives , servers , sensors and also **acquire the data** which is **stored on** the **personal computer** or **Human Machine Interface(HMI)**

SCADA

What is SCADA?

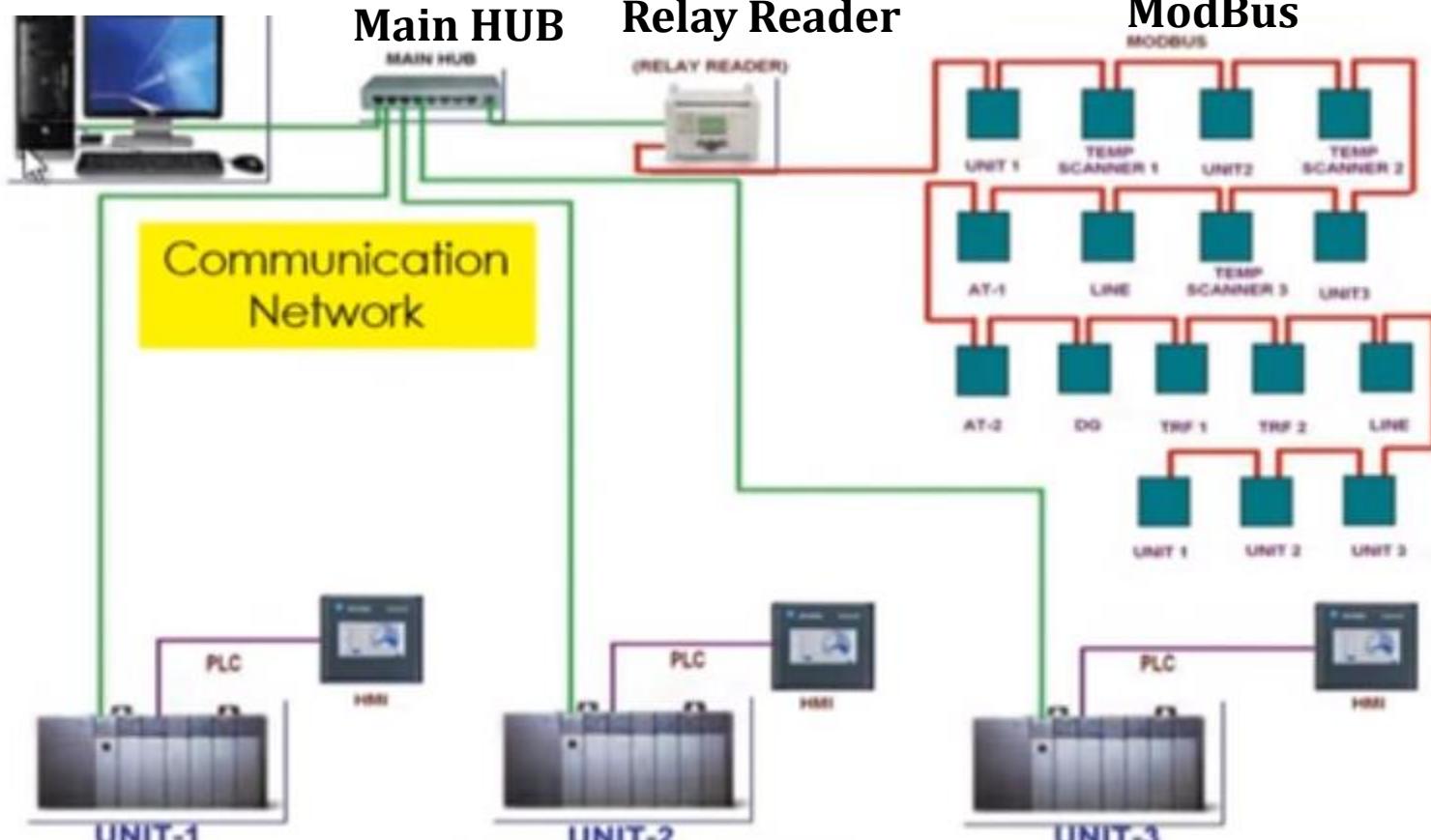
Supervisory Control & Data Acquisitions



SCADA Architecture

Control Centre

Field Instruments



Field Instruments

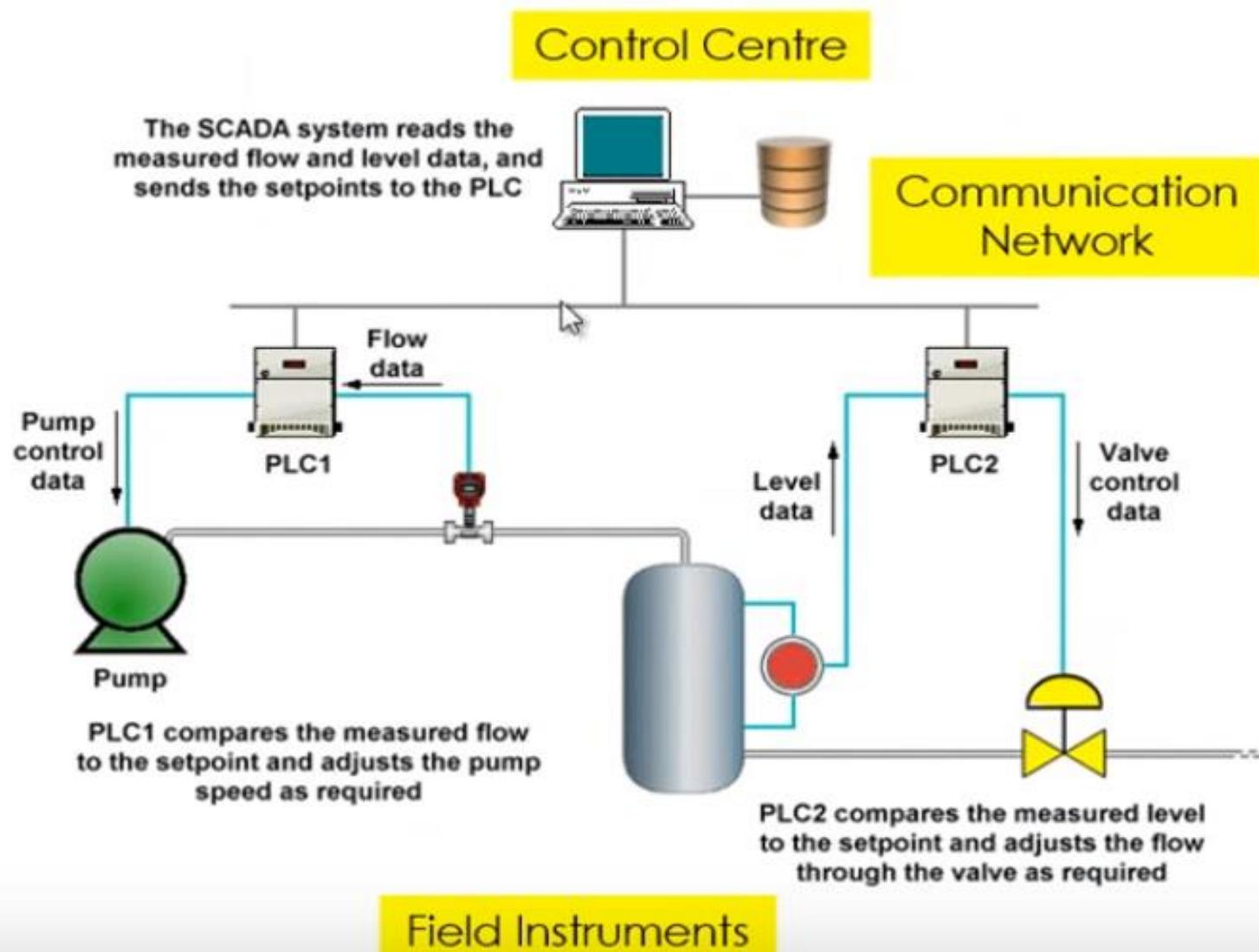
SCADA Architecture

- SCADA Architecture has a **control centre** connected to the main hub(i.e the ethernet port)
- The PLC(Relay Reader is **connected to the ethernet board** which is overall **connected to the CPU**.
- PLC on other hand is **connected to various field instruments** which can be the temperature sensors or actuators that can be analog or digital

SCADA Architecture

- The PC has a **SCADA s/w** which can **interact with the field instruments**
- PC is also connected to various other PLC's UnitsI , Unit II, as shown in Diagram.
- There is an Human Machine Interface(**HMI**) which is **individually connected to the PLC**
- The **HMI** individually **monitors and controls** the PLC
- To read information from all the units we need a SCADA system.
- **PLC (Programmable Logic Unit) monitor and control the data**
- **RTU(Remote Terminal Units)** receive data from sensors and convert this data to digital data and send to SCADA system

Example of SCADA Architecture



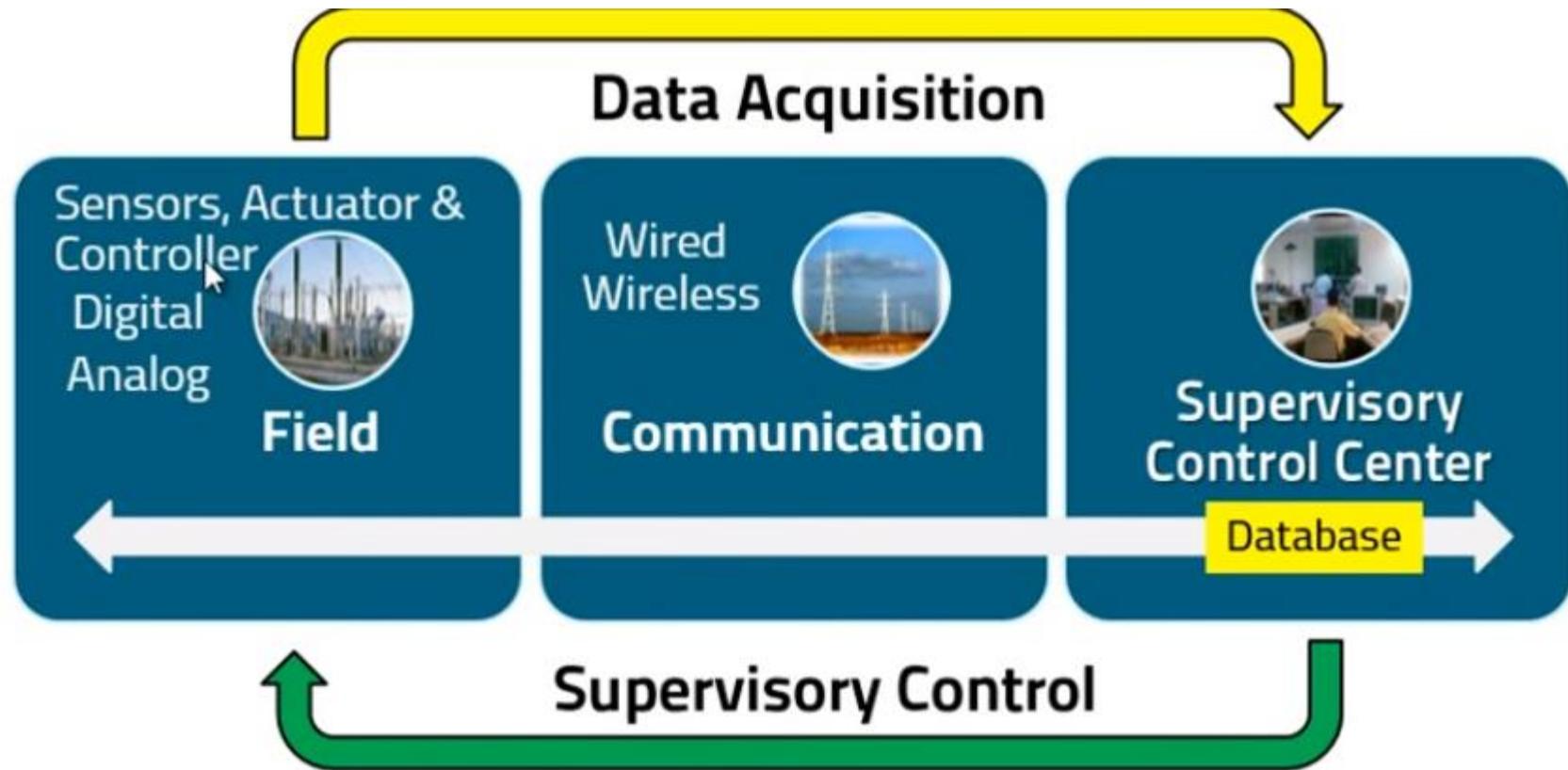
Example of SCADA Architecture

- As shown in the figure the Pump is controlled by the PLC1, which controls speed of the pump.
- There is a water level sensor that senses the level of water in water tank and gives the information to PLC2 that monitors the level of the water.
- The PLC1 and PLC2 are connected to the SCADA system.

Example of SCADA Architecture

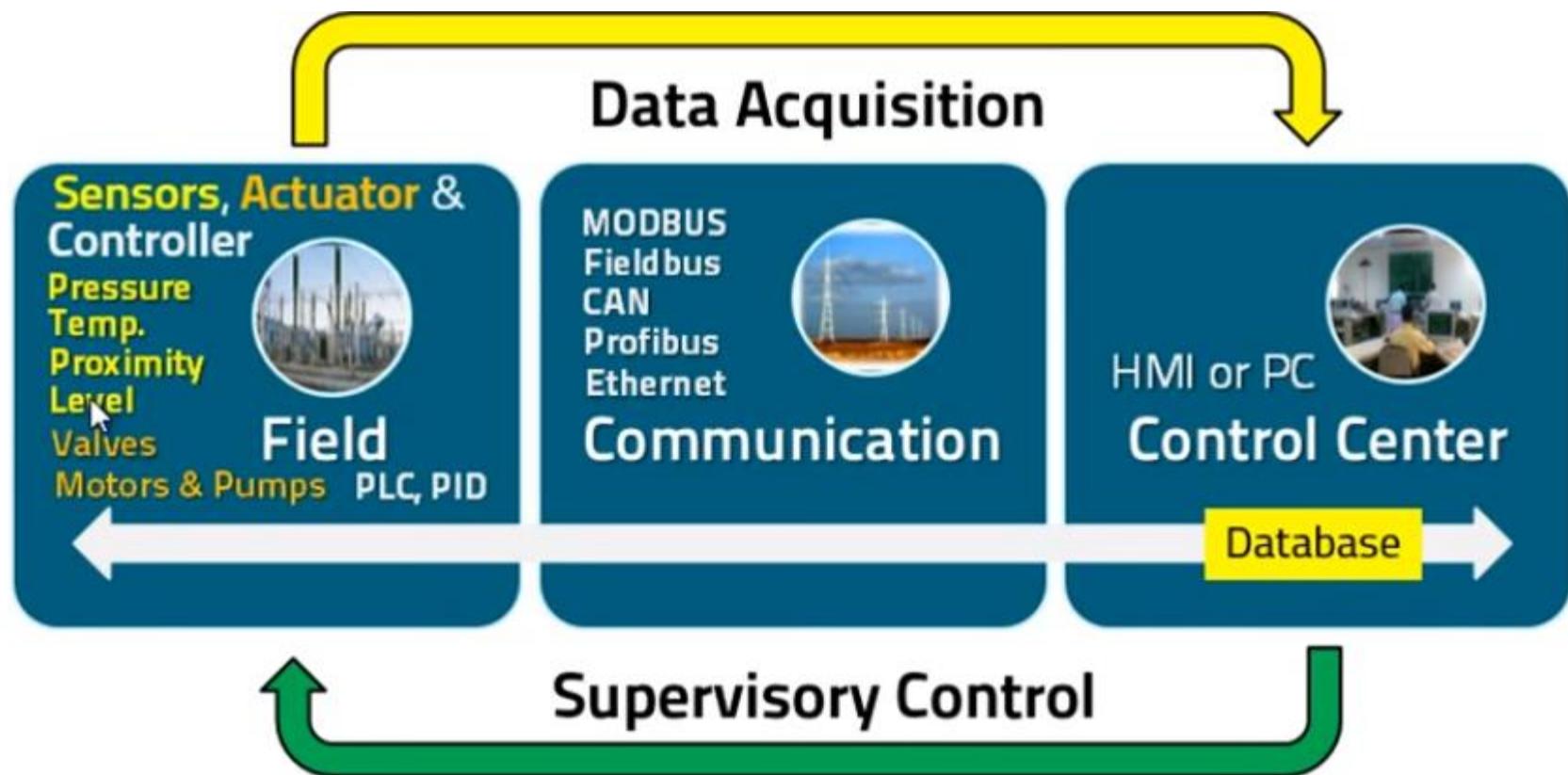
- The system contains the database which can store the data of speed of the pump and water level.

SCADA Architecture



- Information Display
- Supervisory Control
- Alarm & Tagging
- Data Logging

SCADA Architecture



► Information Display ► Supervisory Control ► Alarm & Tagging ► Data Logging

SCADA Applications

SCADA Applications



Food Processing
Industry



SCADA Applications

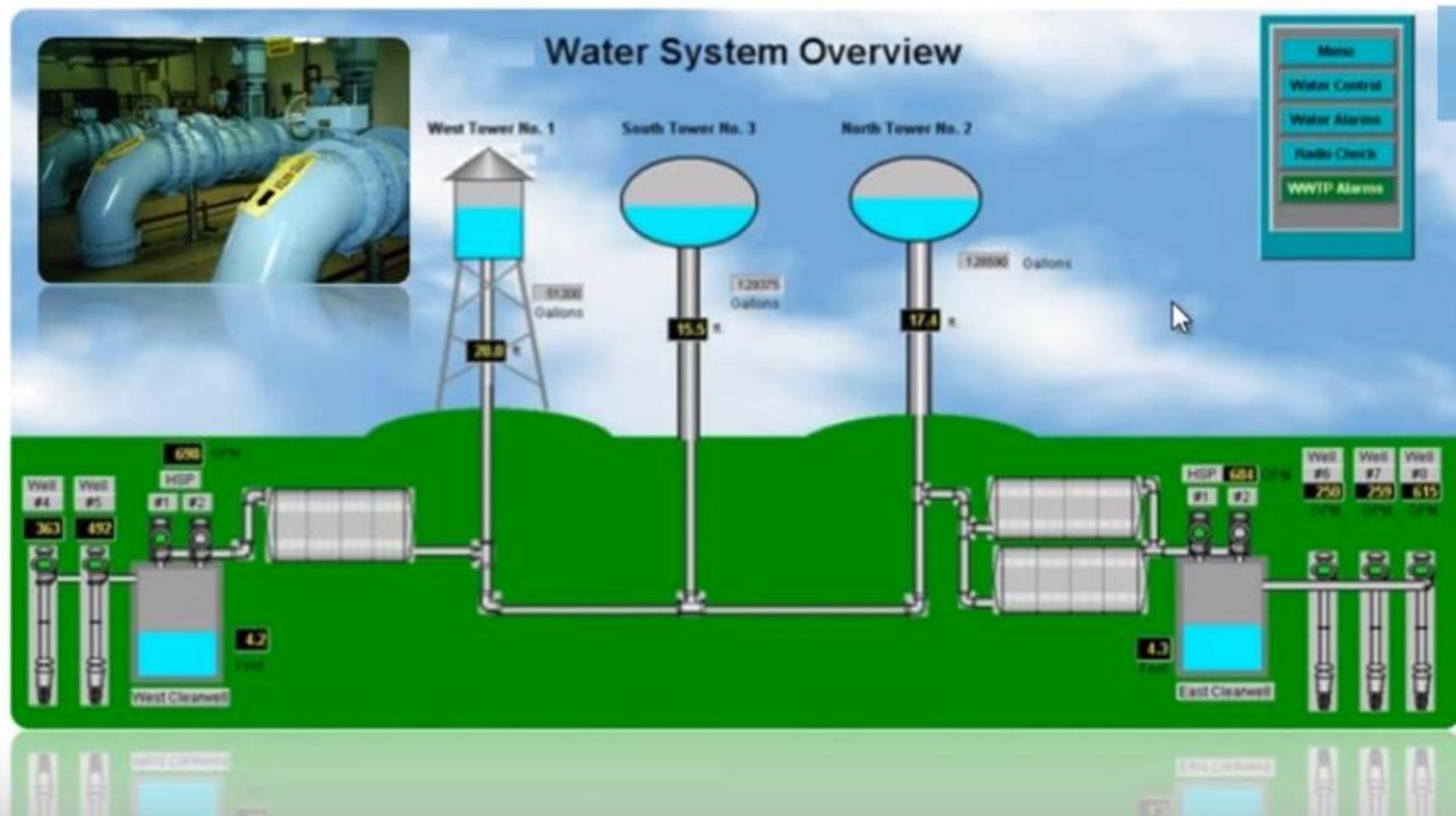
Chemical Industry

The image displays a SCADA system interface for a chemical plant. The background features a photograph of a brightly lit industrial facility at night. The interface is organized into several sections:

- Plant Status: ONLINE**: Shows Total Flow as 1047.6 GPM with a **Reset** button.
- Filter Pumps**: Displays ON status at 1200 GPM with **START** and **STOP** buttons.
- Backwash**: Shows **START** and **STOP** buttons for Alternate mode with Setpoints at 15.0 PSI and 10.0 Min.
- River Flow**: Indicated as 1744.5 GPM.
- Clarifier Mod Valve**: Setpoint at 85%.
- Alum Setpoint Control**: Output at 38.1% in Manual mode, with a Manual Selpoint at 80%.
- Blowdown**: Set to OFF with Auto and Manual options, and a Timer Preset of 00:00.
- Pump 1**: 350.8 GPM.
- Pump 2**: 347.5 GPM.
- Pump 3**: 361.1 GPM.
- Pressure**: 10 PSI.
- River Pumps**: Pump 1 at 1022.8 GPM and Pump 2 at 1029.5 GPM, both with **START** and **STOP** buttons.
- ALTERNATE**: A button at the bottom left.
- Filter Plant Flow Control**: GPM Control ON with Set % Open and Set GPM buttons, and a Mill Pressure of 10 PSI.
- Filter Plant**: Effluent Turbidity at 23 NTU, Filter Plant Flow at 105.94 GPM, Mill Pressure at 10 PSI, and pH at 5.6.

SCADA Applications

Water Treatment Plant



WSN

- **Wireless Sensor N/w**
- It senses and gathers data using sensors which are spatially distributed
- It collects this data into a centralized location with the help of wired / wireless connection

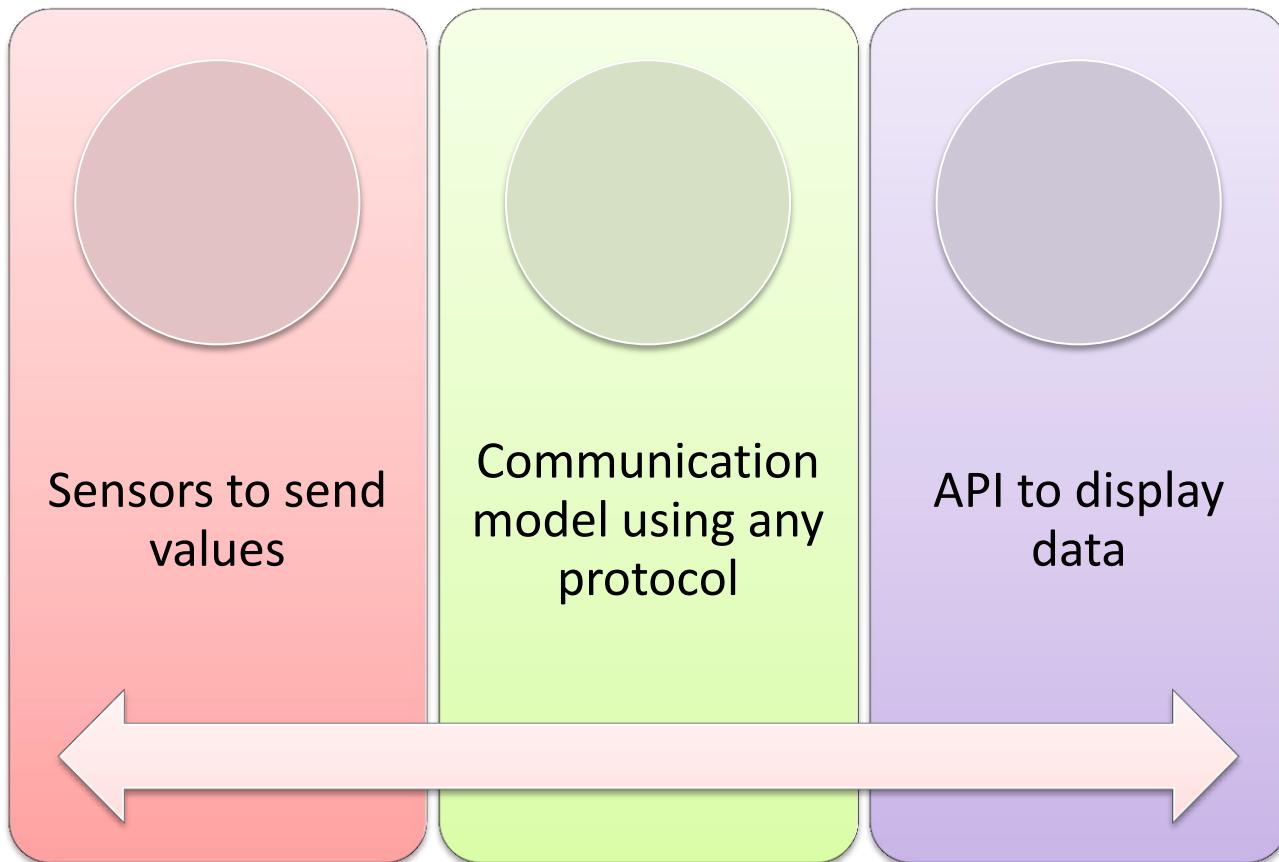
WSN

- * It is a deployment of several devices equipped with sensors that perform a collaborative measurement process



WSN

- **Consists of three basic things**



WSN

Send any value...

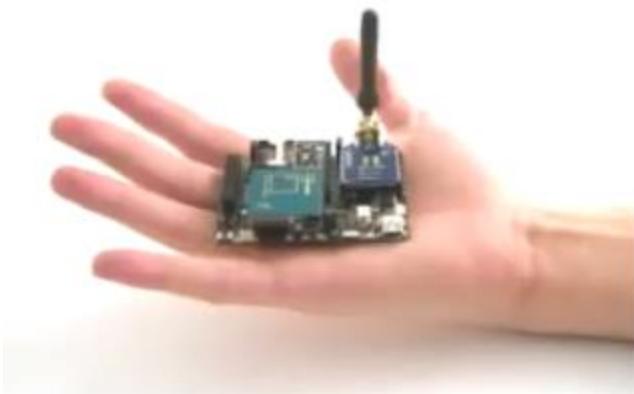
...using any protocol

...to any system

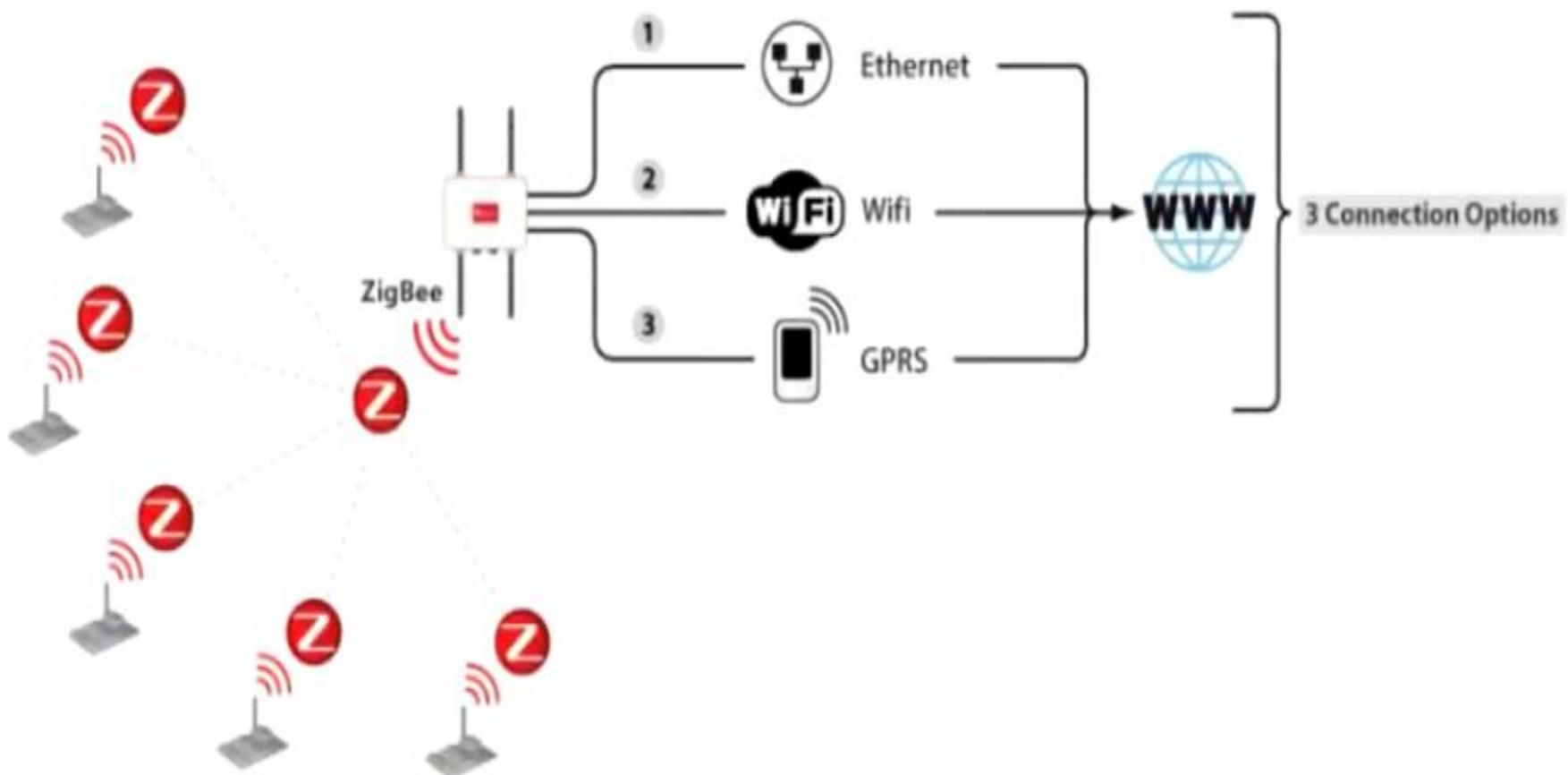


WSN Elements

- * **Node:** Autonomous sensor-equipped device
- * **Data gatherer:** Data capturer and gateway to external systems
- * **External systems:** Data storing and managing centres



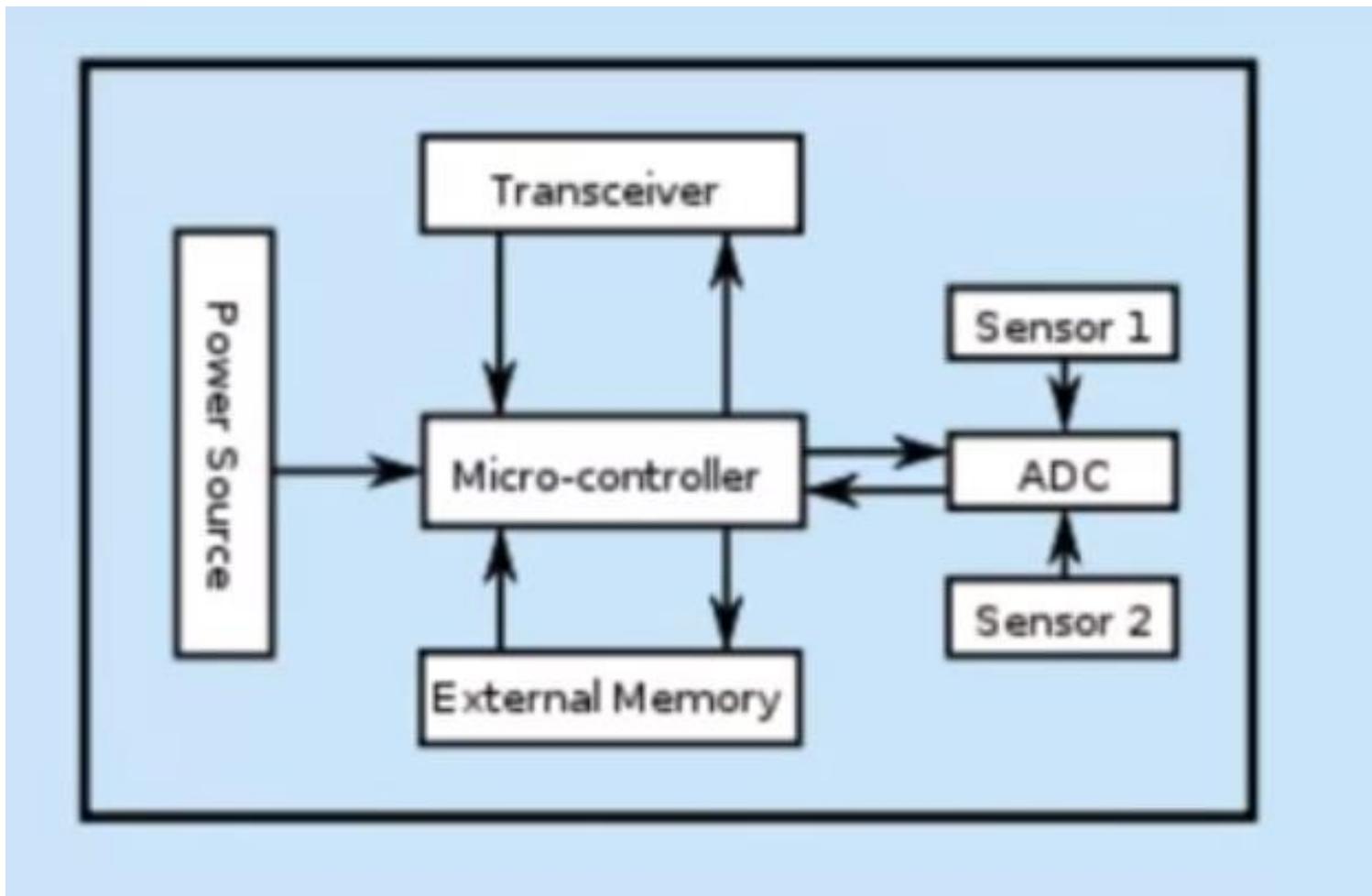
Working of WSN



Working of WSN

- In the above diagram we can see the **sensors (nodes)** are **sensing** the **device values**
- These **transmit** the **information** to the **measuring device (data gatherer)** which **transmits** the **values** to **external systems** using **Ethernet, Wifi, or GPRS.**

Parts of WSN



Difference between four Pillars of IoT in terms of communication

Communication	Wired		Wireless	
Pillars	Short Range	Long Range	Short Range	Long Range
M2M	No	Some	Some	Yes
RFID	No	Some	Yes	Some
WSN	No	Some	Yes	Some
SCADA	Yes	Yes	Some	Some

So.....



Internet of devices



Internet of Objects



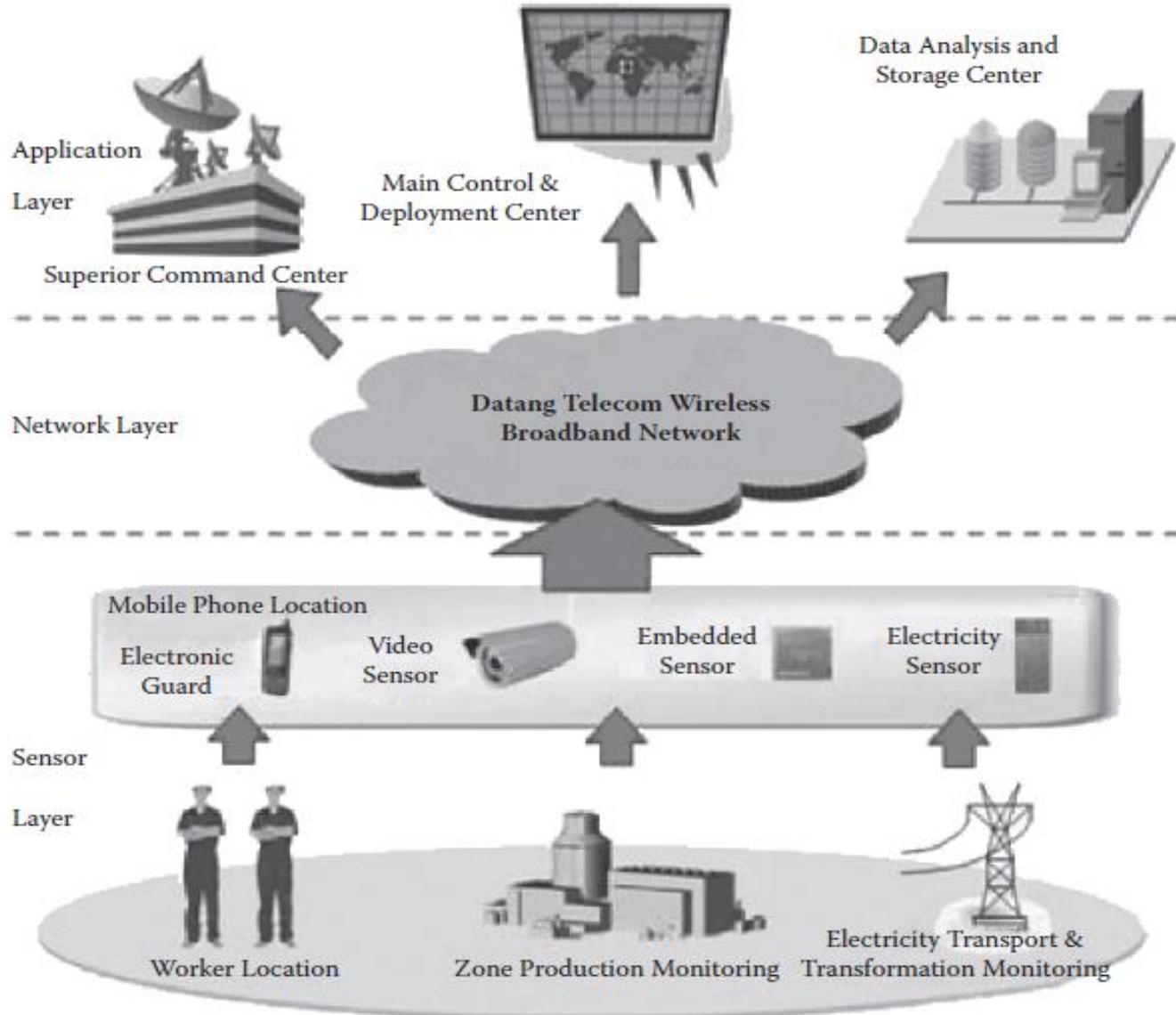
Internet of Transducer
(Transmitter and receiver)



Internet of controllers

Three Layer architecture of IoT

M/A



C/N

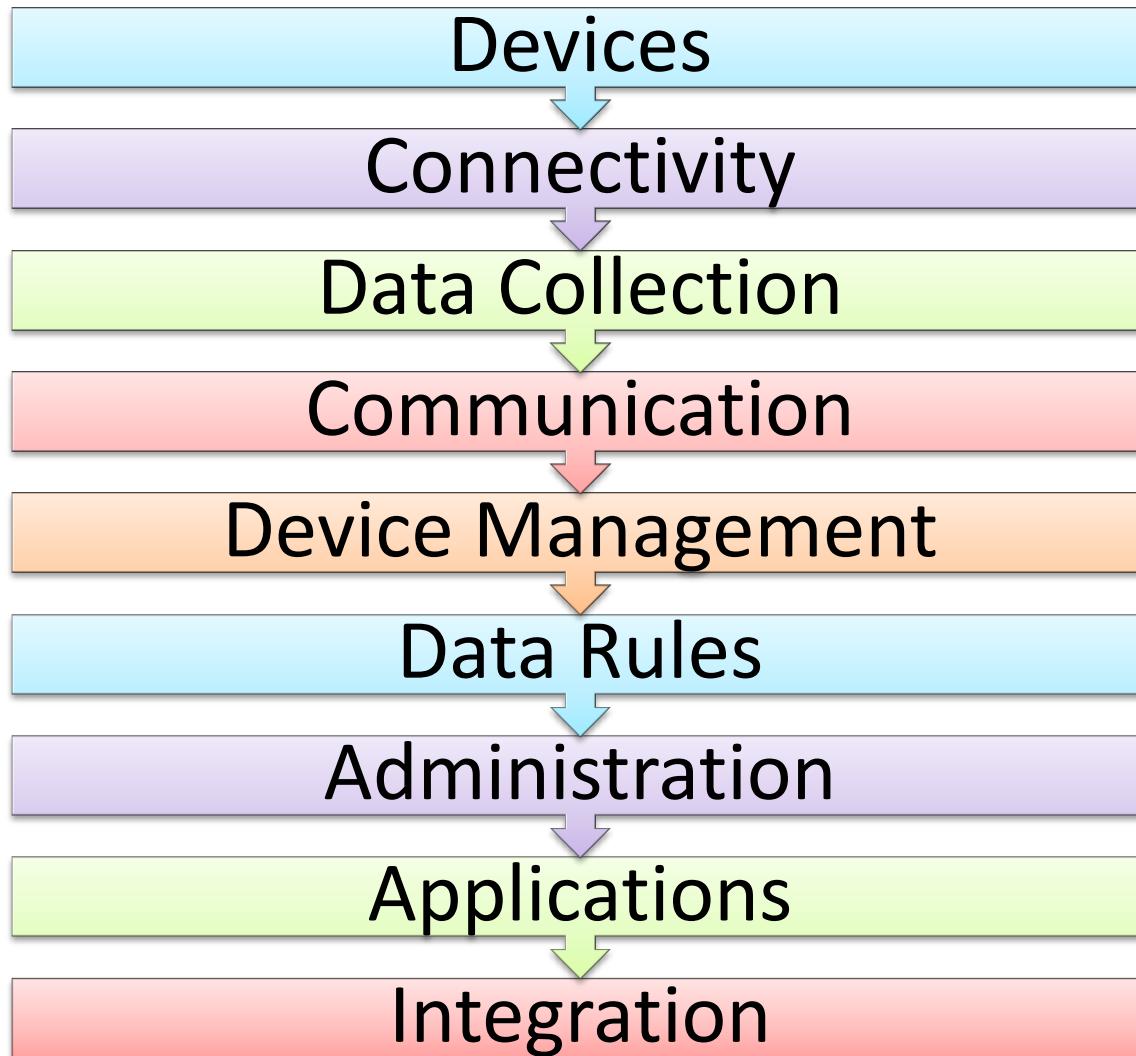
Manage/Application

D

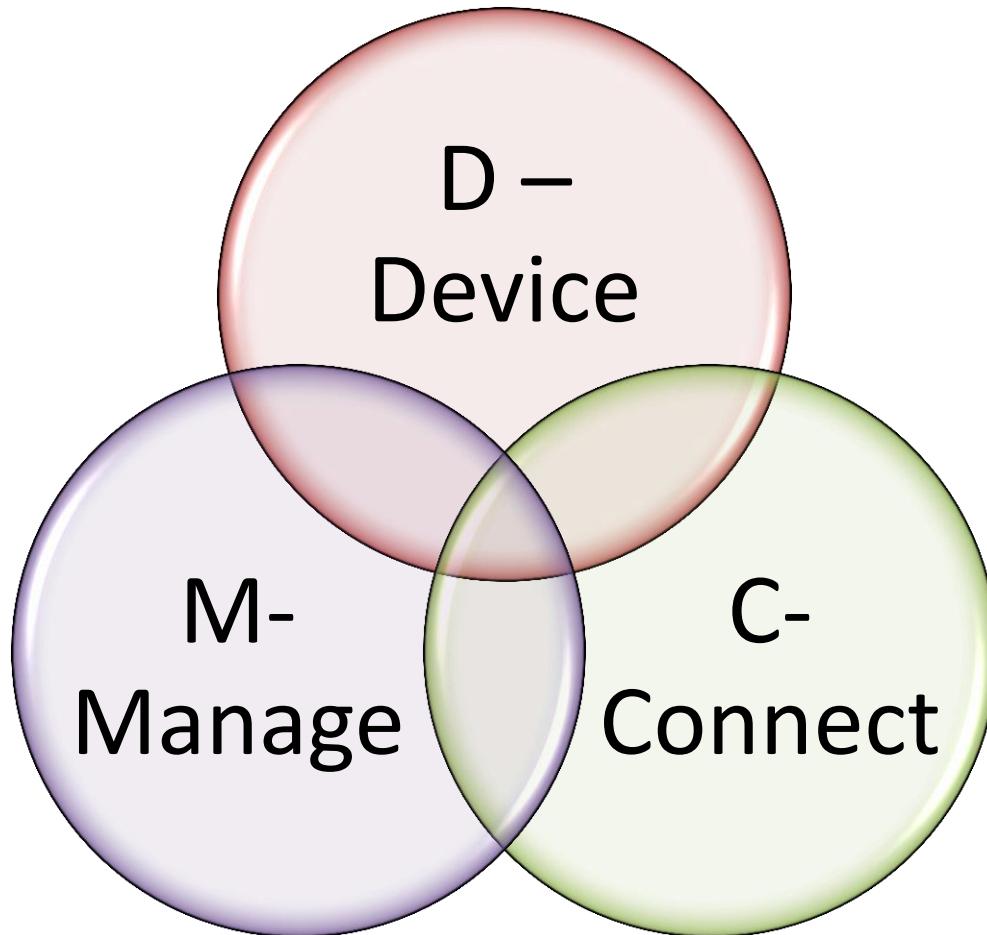
Connect/Network

Device

Nine Layer architecture of IoT



DCM



Devices : Things that Talk

Inherent Intelligent

- Means Inbuilt Intelligent
- Eg. Washing Machines, ventilation, and air-conditioning, (HVAC) controllers

Enabled Intelligent

- Means need to be made intelligent
- RFID tagged devices

Devices : Things that Talk

Sensors

- Perform **Input function**
- Device that **responds to a physical stimulus, measures** the physical stimulus quantity, and **converts it into** a signal, usually **electrical**, which can be read by an observer or by an instrument
- Also called **detector**
- A sensor is basically an **electrical device**. It could be an **M2M terminal**, an **RFID reader**, or a **SCADA meter**.
- A sensor can be very small and itself can be a trackable device

Devices : Things that Talk

Sensors

- Perform Input function
- Device that responds to a physical stimulus, measures the physical stimulus quantity, and converts it into a signal, usually electrical, which can be read by an observer or by an instrument

Actuators

- Performs Output function
- RFID tagged devices

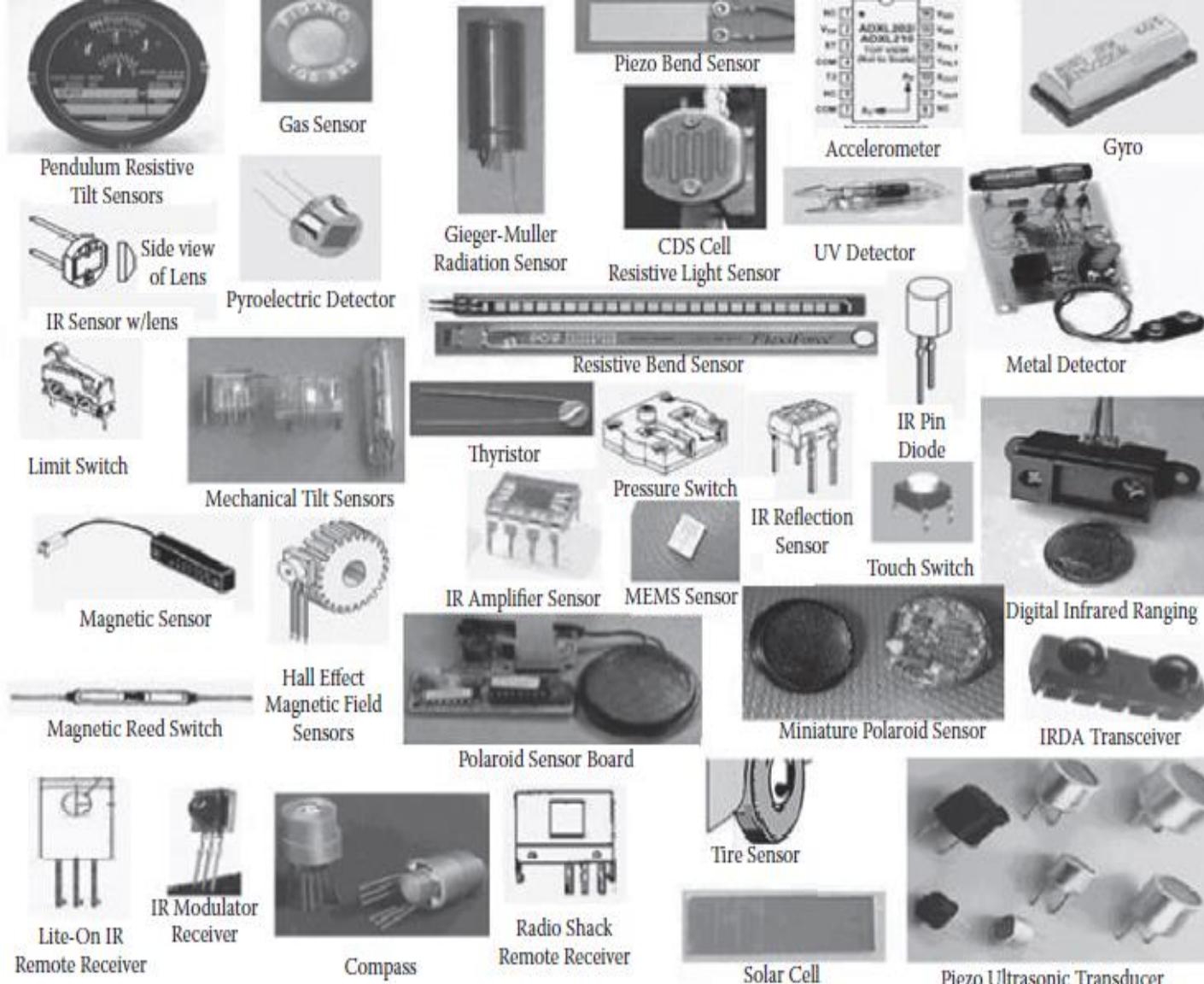


Figure 4.3 Examples of sensors.

Sensor Type and Example

Sensor Type	Example
Position, angle, displacement, distance, speed, acceleration	position sensor, Accelerometer, capacitive displacement sensor
Pressure	barometer, boost gauge, pressure sensor
Acoustic, sound, vibration	Geophone, hydrophone, lace sensor, microphone
Automotive, transportation	Air-fuel ratio meter, engine coolant temperature (ECT) sensor, parking sensors,

Sensor Type and Example

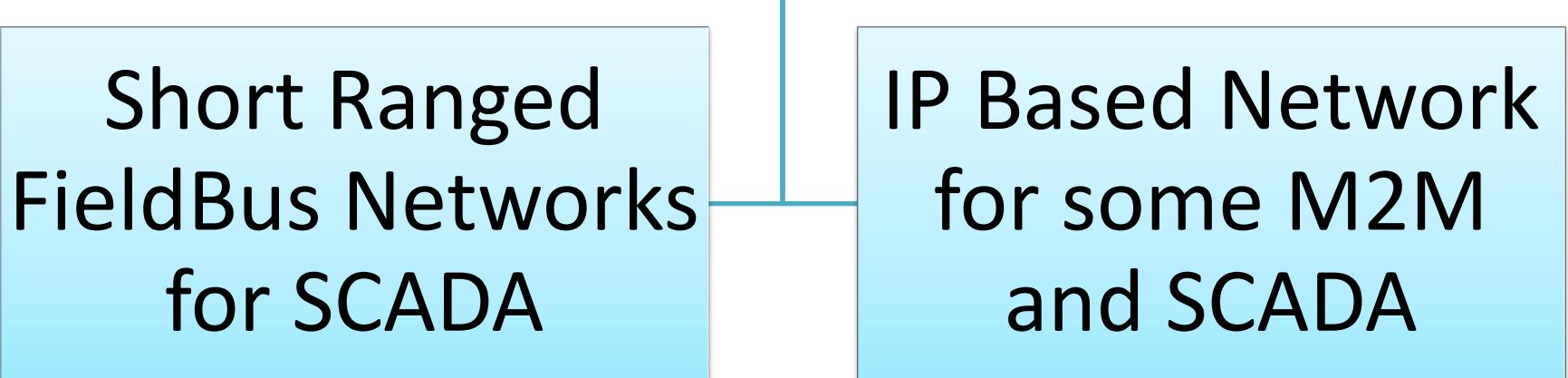
Sensor Type	Example
Environment, weather, moisture, Humidity	leaf sensor, rain sensor, soil moisture sensor
Flow, fluid velocity	Gas meter, water meter
Optical, light, imaging, photon	Contact image sensor, infrared sensor.....

Connect : Pervasive Network

- The communications layer is the foundational infrastructure of IoT.
- There are two major communication technologies: **wireless and wired** (or wireline).
- When talking about IoT, **wireless communications** is the topic most of the times, because three (M2M, RFID, and WSN) of the four IoT pillars are based on wireless

Wired Communication

Wired



Wired Communication : FieldBus

- Field bus is the name of a family of **industrial computer network protocols** used for **real-time distributed** control of nodes.
- Standardized as **IEC 61158**
- **Field** refers to **geographical area** or contextual area in industry where IOT devices are distributed
- **Bus** refers to the **electrical medium** that carries **data** through it.

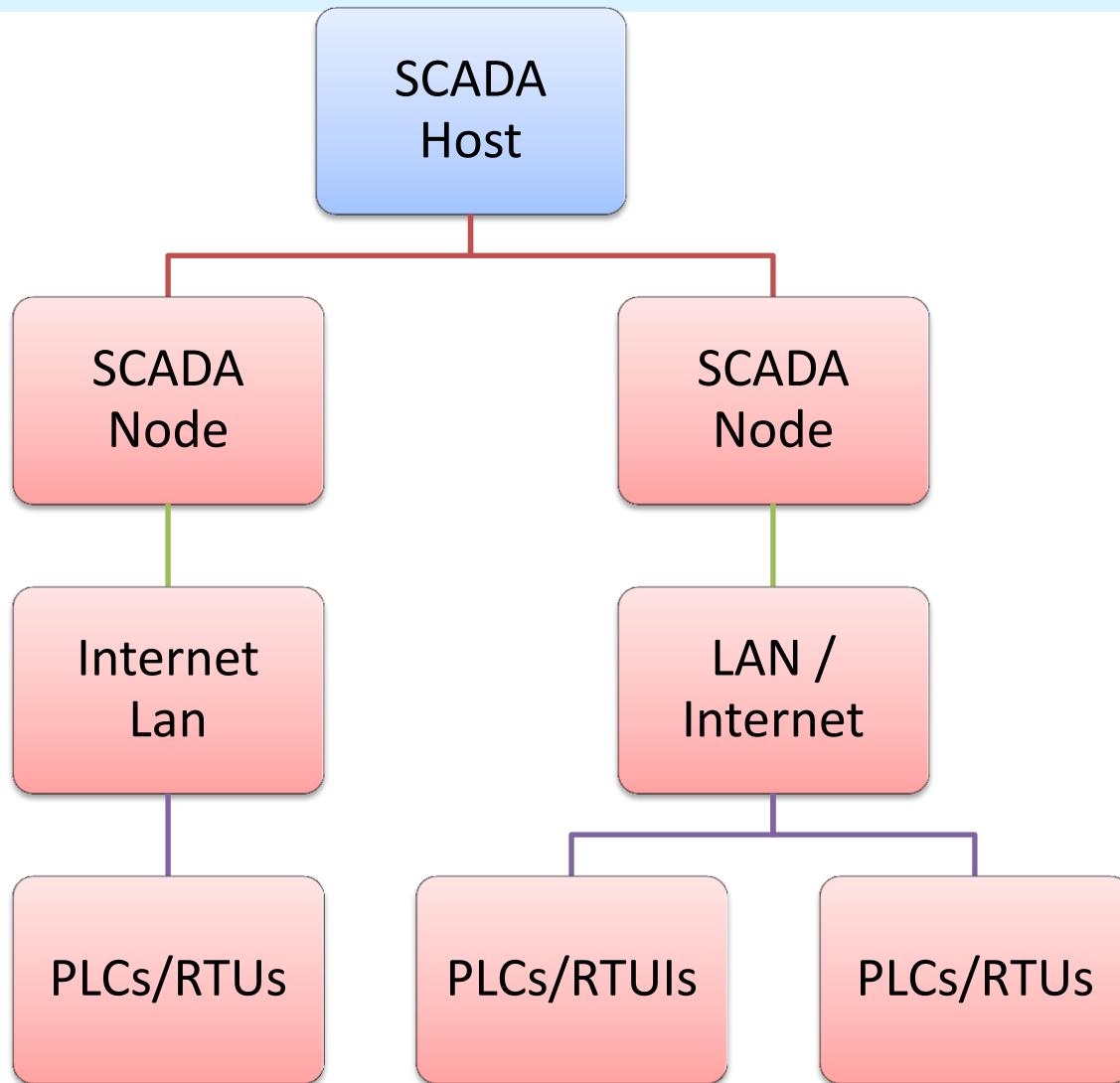
Wired Communication : FieldBus

- Field bus has Eight different protocol sets
 - Type 1: Foundation field bus H1
 - Type 2 ControlNet
 - Type 3 PROFIBUS
 - Type 4 P-Net
 - Type 5 FOUNDATION field bus HSE (high-speed Ethernet)
 - Type 6 SwiftNet (a protocol developed for Boeing, since withdrawn)
 - Type 7 WorldFIP
 - Type 8 Interbus

Wired Communication : FieldBus

- Most use **Twisted pair or optical fiber** as a physical communication medium
- **Uses bidirectional communication** between the field instruments and HMI..
- Mainly used for **SCADA Based Applications**

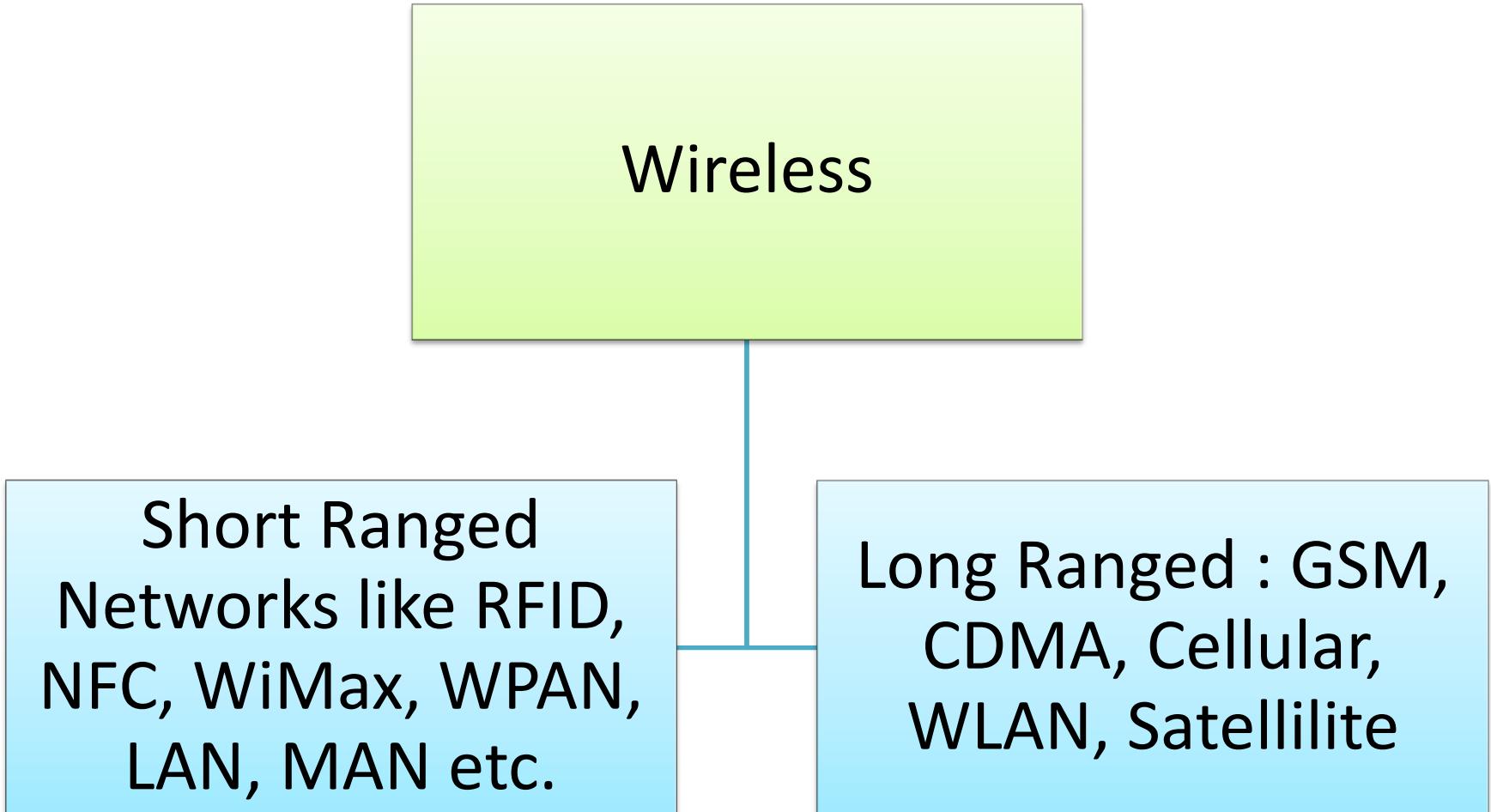
Wired Communication : FieldBus Communication



Examples for FieldBus

- Automatic meter Reading : ModBus
- Home Automation : Bacnet
- Power Automation : Profibus

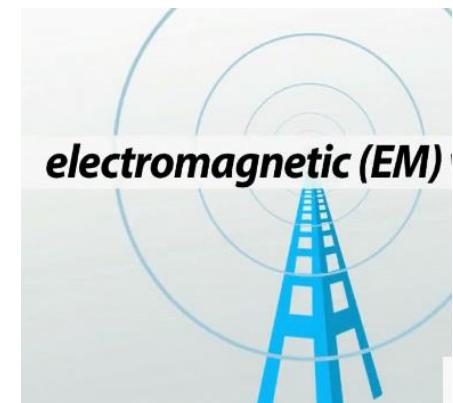
Wireless Communication



Wireless Network

any type of network that establishes connections without cables

- Wireless Communication **use electromagnetic waves** that travel through the air
- Simplest example of this is : **Radio**





- When we listen radio in a moving car we are actually receiving radio waves which is one type of electromagnetic wave.



- Electromagnetic waves are analog whereas information in computer is digital; **wireless systems therefore need adapter for analog to digital conversion.**

Categories of Wireless Communication

Short-range wireless

Bluetooth, infrared and Zigbee

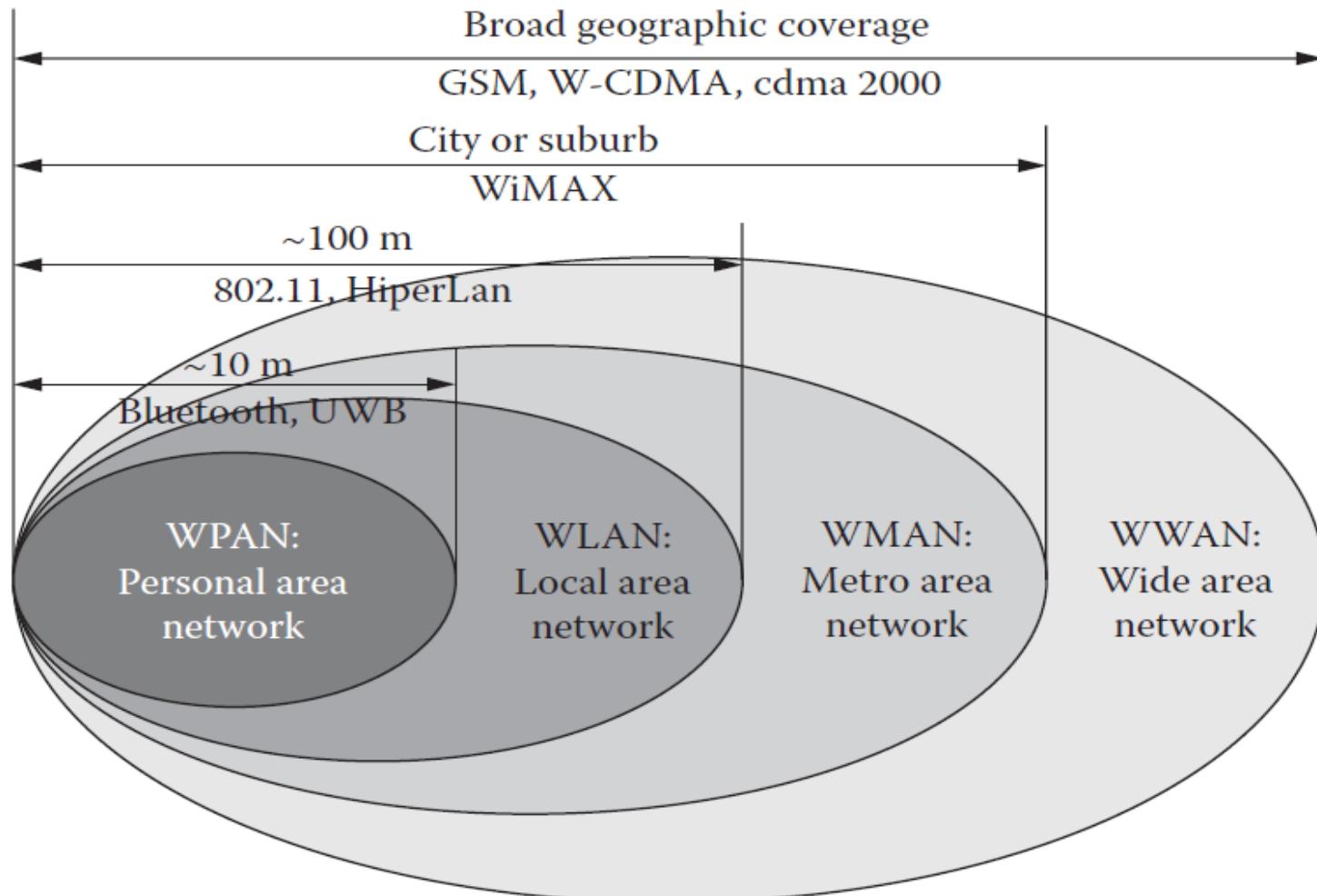
Medium-range wireless

Wireless Fidelity (WiFi)

Wide area wireless

cellular and satellite communications

Short and long range wireless n/w's



Standards of Wireless n/w

- RFID and NFC are parts of WPAN.
- 6LowPAN (IPv6 over low power wireless personal area networks)
- BSN (Body Sensor n/w)
- HomeIR: wireless IR home networking
- HomeRF: wireless RF home networking

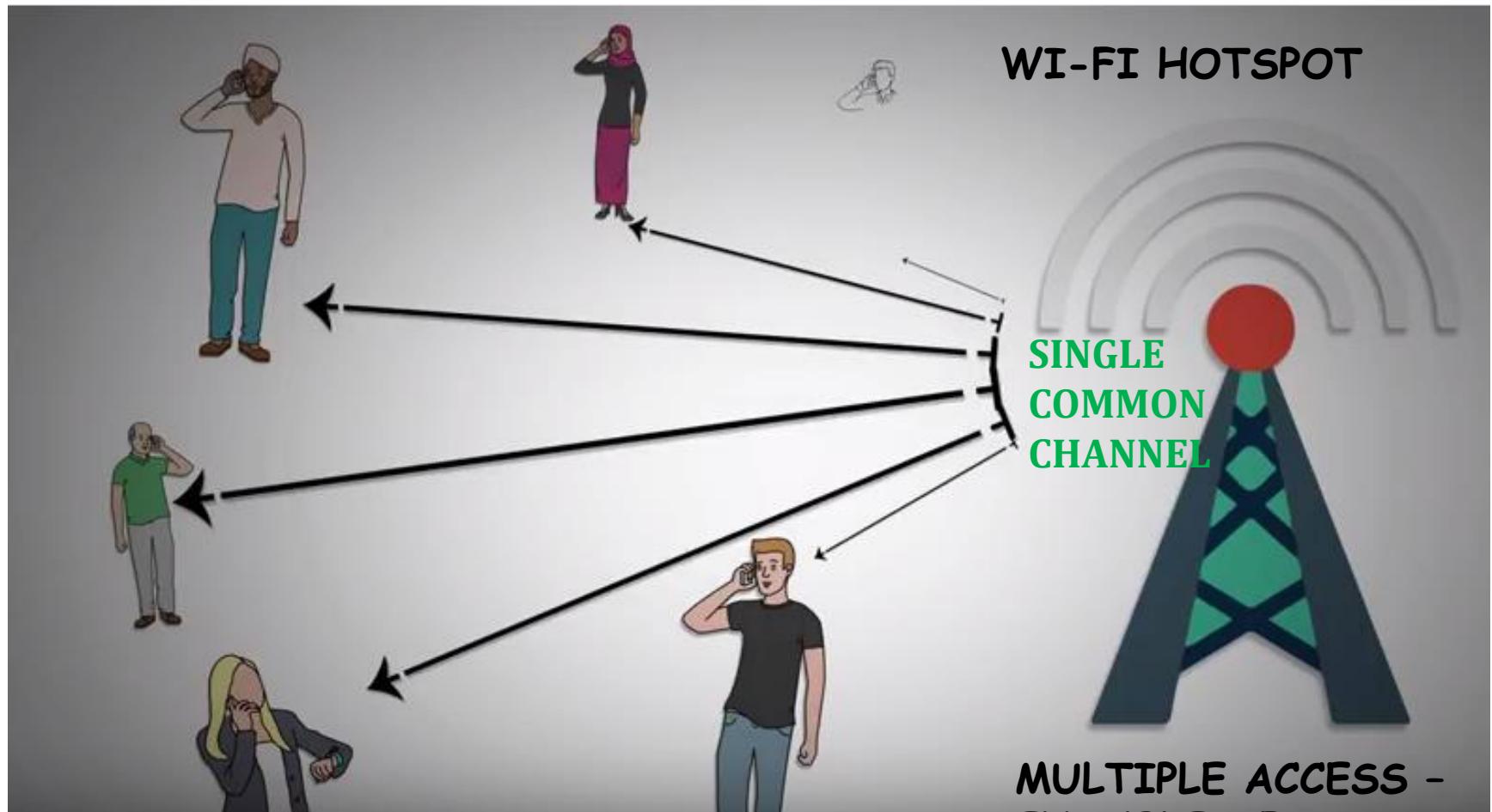
Communication Mechanisms of Wireless n/w

- Orthogonal frequency division multiplexing (OFDM) and orthogonal frequency division multiple access(OFDMA)
- Ad hoc sensor network
- Software defined radio (SDR)
- Cognitive radio (CR)

OFDM/OFDMA

- These are two different variants of the same **broadband wireless air interface**.
- **Long-Term Evolution (LTE)** the standard for **high-speed wireless communication** used for mobile devices and data terminal is an **OFDMA-based** technology **standardized in 3rd Generation Partnership Project (3GPP)**.
- Typically occupy roaming, fixed, and one-way transmission standards, ranging from TV transmission to Wi-Fi as well as fixed WiMAX and newer multicast wireless systems

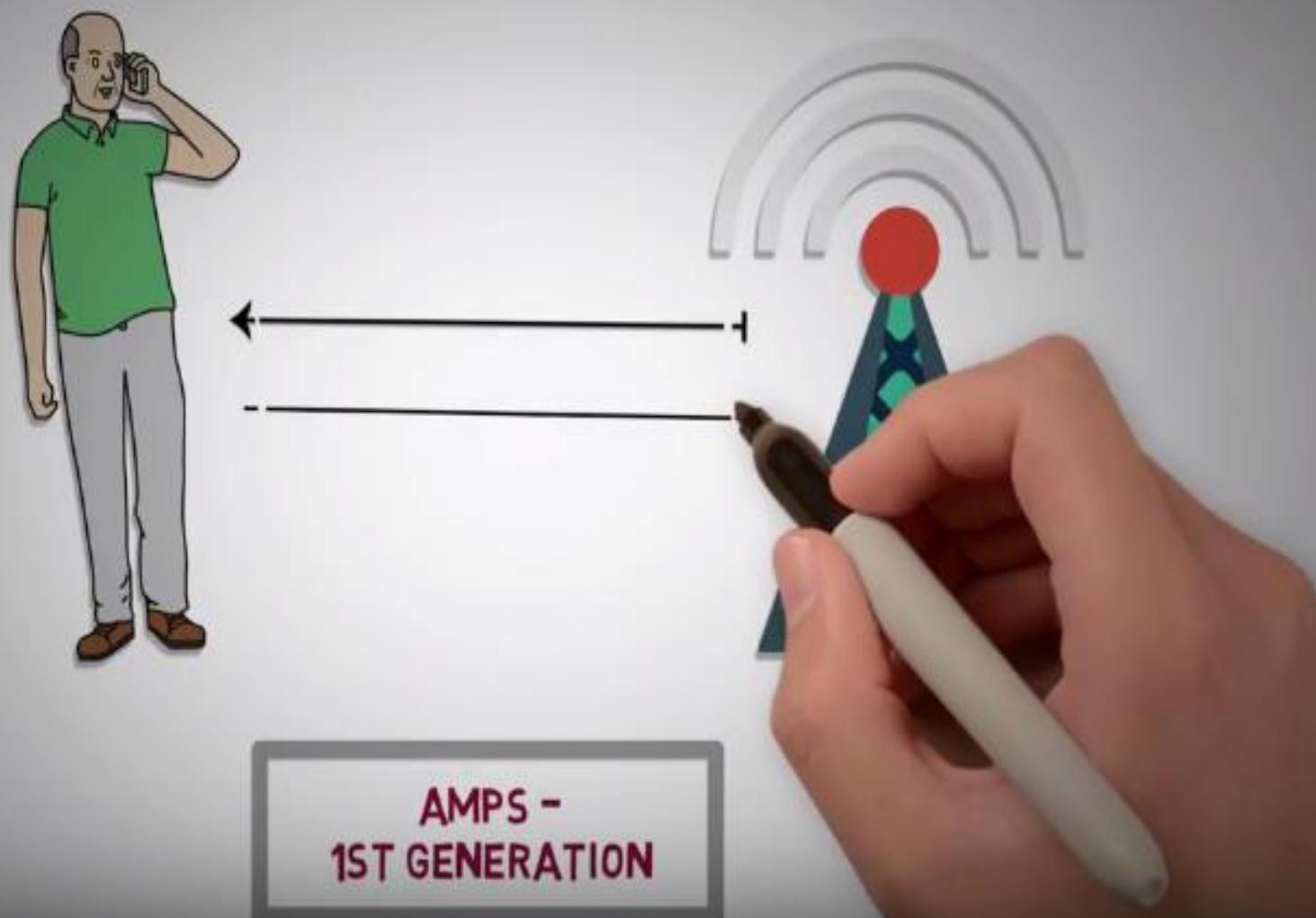
Multiple Access



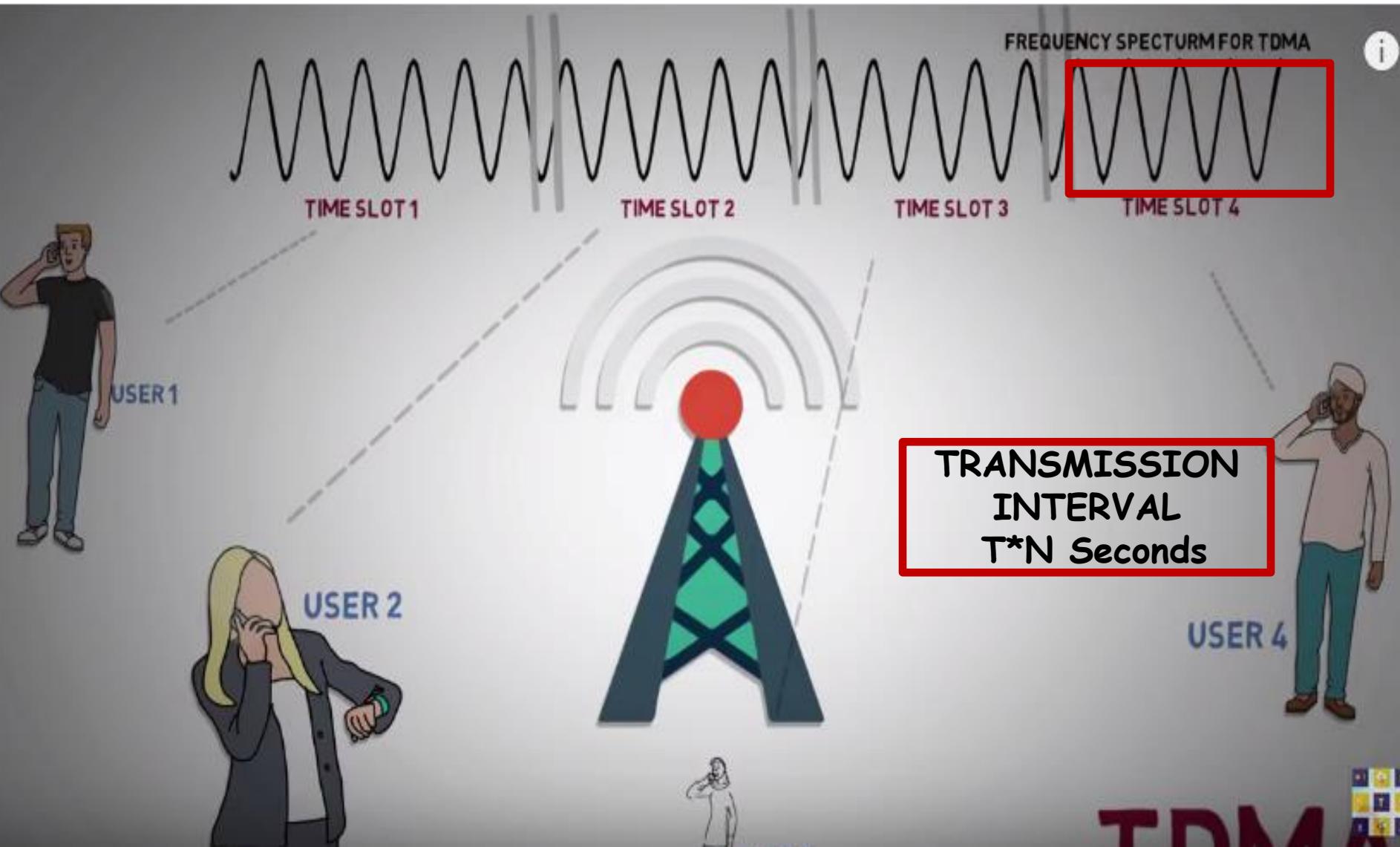
MULTIPLE ACCESS -
EXAMPLE OF
MULTIPLEXING

FDMA

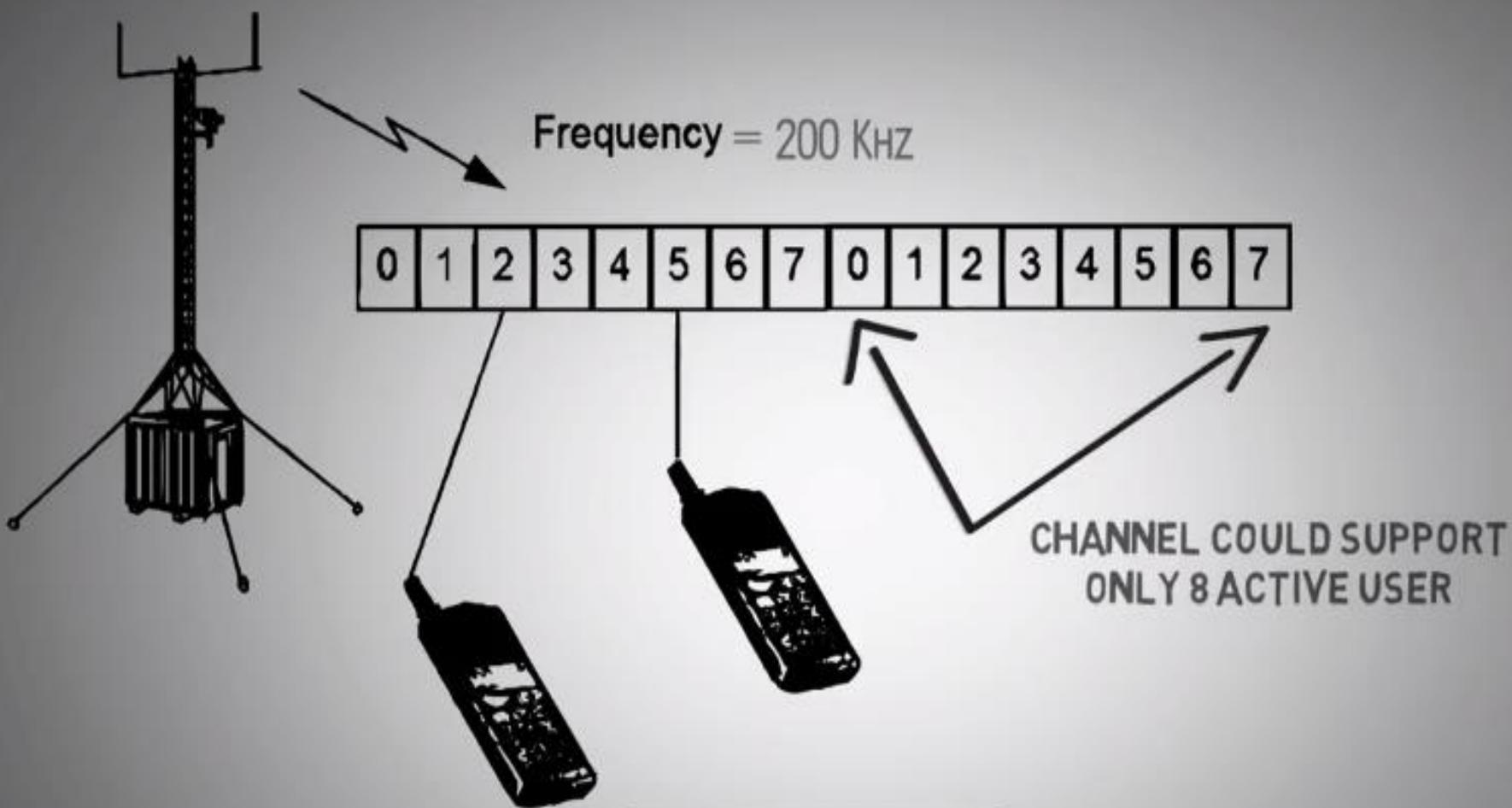
EXAMPLE OF FDMA



TDMA



EXAMPLE OF TDMA



2ND GENERATION
GSM, GPRS, EDGE

CDMA

Every User in CDMA get a unique code which is known as chipping sequence.

These codes are orthogonal which means if these codes are multiplied together , it will give zero. (consider 0 as -1)



10110

1 -1 1 1 -1 1

1 1 -1 1 1 1



11011

$$\begin{array}{r} 1 -1 1 1 -1 1 \\ \times \quad 1 1 -1 1 1 1 \\ \hline 1 -1 -1 1 -1 1 \end{array}$$

↓ ↓ ↓ ↓ ↓ ↓

Add → 0

CDMA ..>Encoding

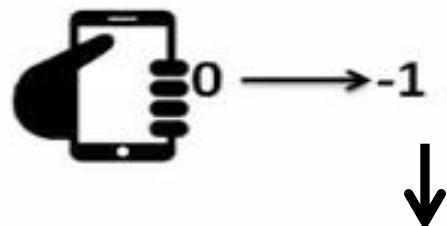
Encoding of Data : Data is encoded using chipping sequence.
In chipping sequence and data, consider 0 as -1

Chipping Sequence : 1 0 1 1 0 1

chipping sequence : 1 -1 1 1 -1 1

Saved Sequence : 1 0 1 1 0 1

chipping sequence : 1 -1 1 1 -1 1



Sender

$0 \longrightarrow -1 * (1 -1 1 1 -1 1)$



$-1 1 -1 -1 1 -1 \longrightarrow -1 1 -1 -1 1 -1$

Data bit will be multiplied with every bit of chipping sequence

CDMA...>Decoding

Decoding of Data : Decoding of data done by multiplying received signal with chipping sequence.

Chipping Sequence : 1 0 1 1 0 1

chipping sequence : 1 -1 1 1 -1 1



Sender

$$0 \longrightarrow -1 \\ \downarrow \\ -1 \ 1 \ -1 \ -1 \ 1 \ -1 \ * (1 \ -1 \ 1 \ 1 \ -1 \ 1)$$

Saved Sequence : 1 0 1 1 0 1

chipping sequence : 1 -1 1 1 -1 1



0

$$-1 \ 1 \ -1 \ -1 \ 1 \ -1$$

$$1 \ -1 \ 1 \ 1 \ -1 \ 1$$

$$\boxed{-1 \ -1 \ -1 \ -1 \ -1 \ -1}$$

Add

-6

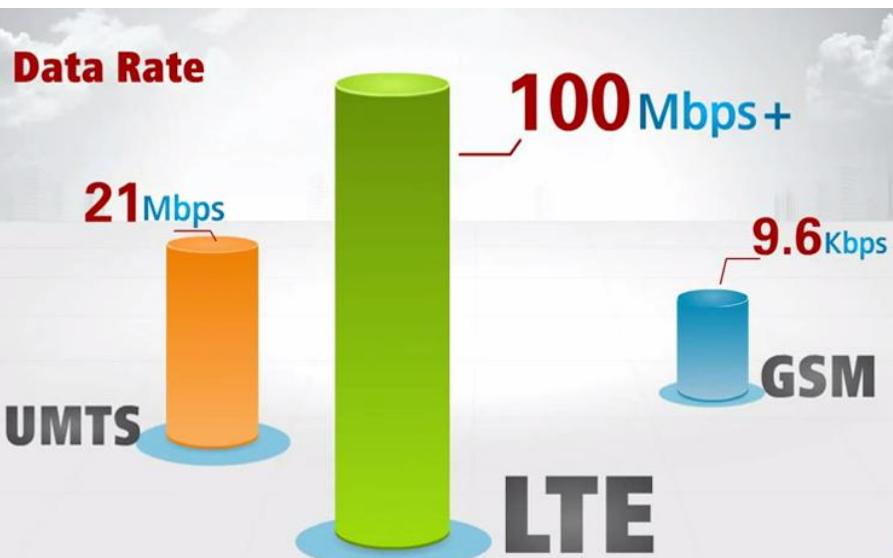
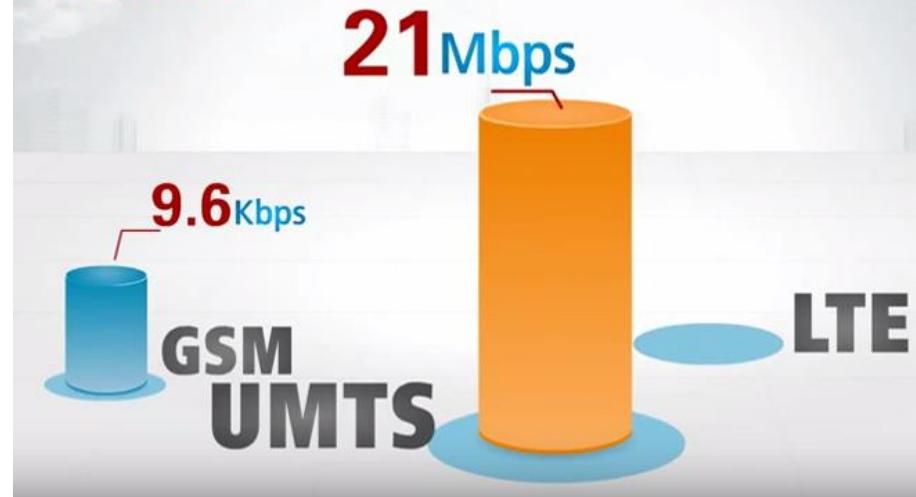
Data bit will be multiplied with
chipping sequence

OFDM

Data Rate



Data Rate



Data Rate



100 Mbps+

What kind of key technology
makes LTE so FAST?

LTE

OFDM

Orthogonal Frequency Division Multiplexing

FDM

- FDM Divides the bandwidth into many subcarriers
- And hence allows multiple users to access the system simultaneously



FDM

- Different users data gets transmits at different subcarriers
- Between Subcarriers there exists a **guard band** to avoid interference



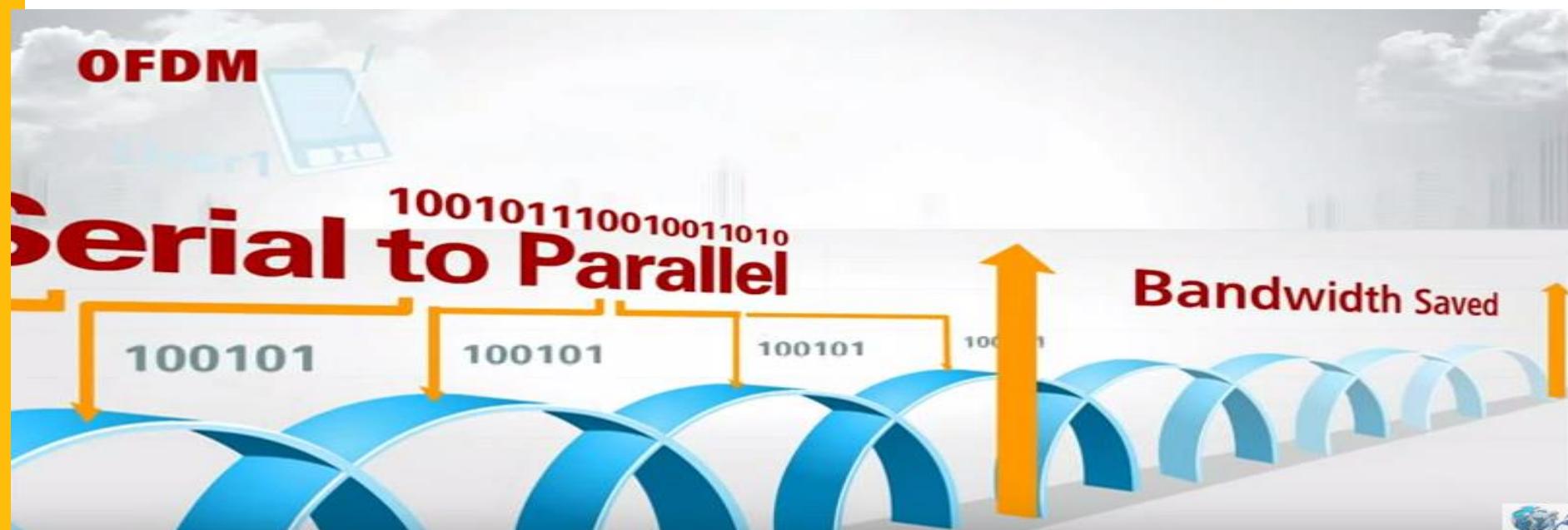
Multicarrier FDM

- In multicarrier FDM the data of a single user can be splitted into multiple sub streams and send them in parallel to make the data rate higher.



OFDM

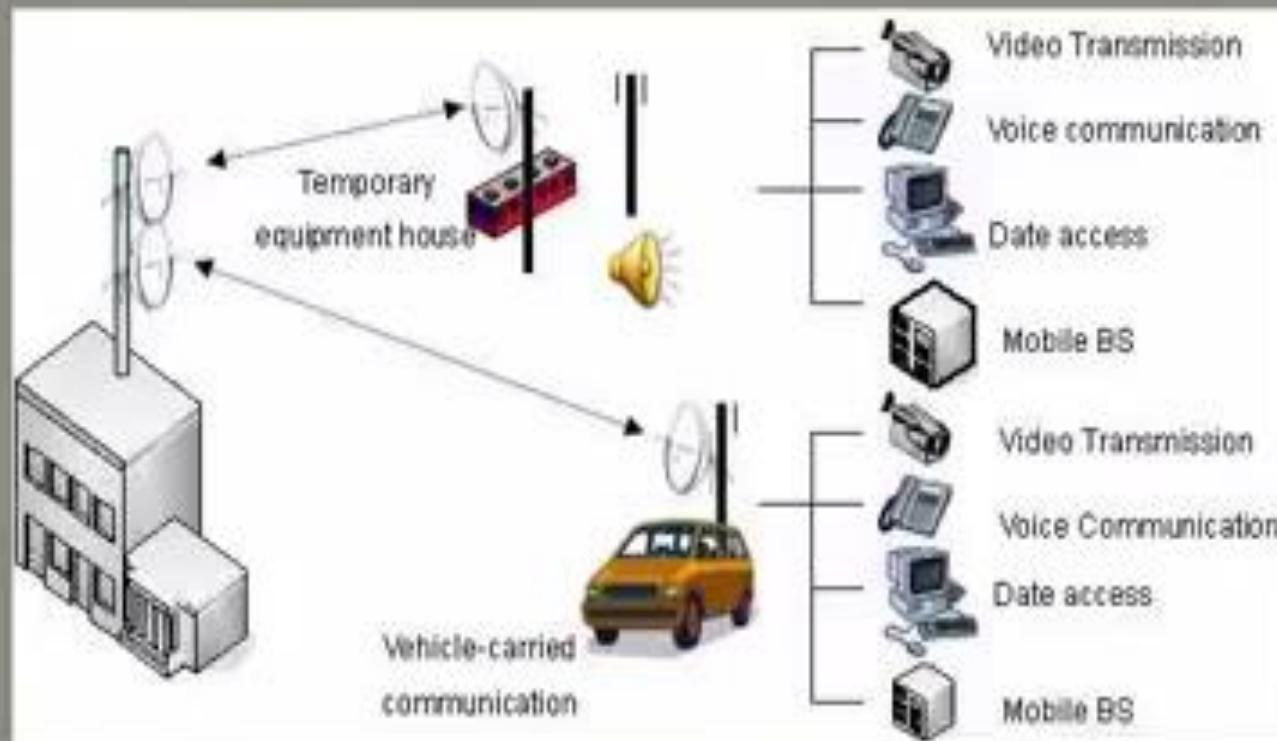
- In OFDM Subcarriers are orthogonal which allows subcarriers to overlap and hence save the bandwidth and achieving higher data rate.



Adhoc Sensor n/w

- a **short-lived network** of two or more mobile devices connected to each other without the help of intervening infrastructure
- an ad hoc network can be deployed in **remote geographical locations** and **requires minimal setup and administration costs.**
- **Said to be emergency and temporary network**

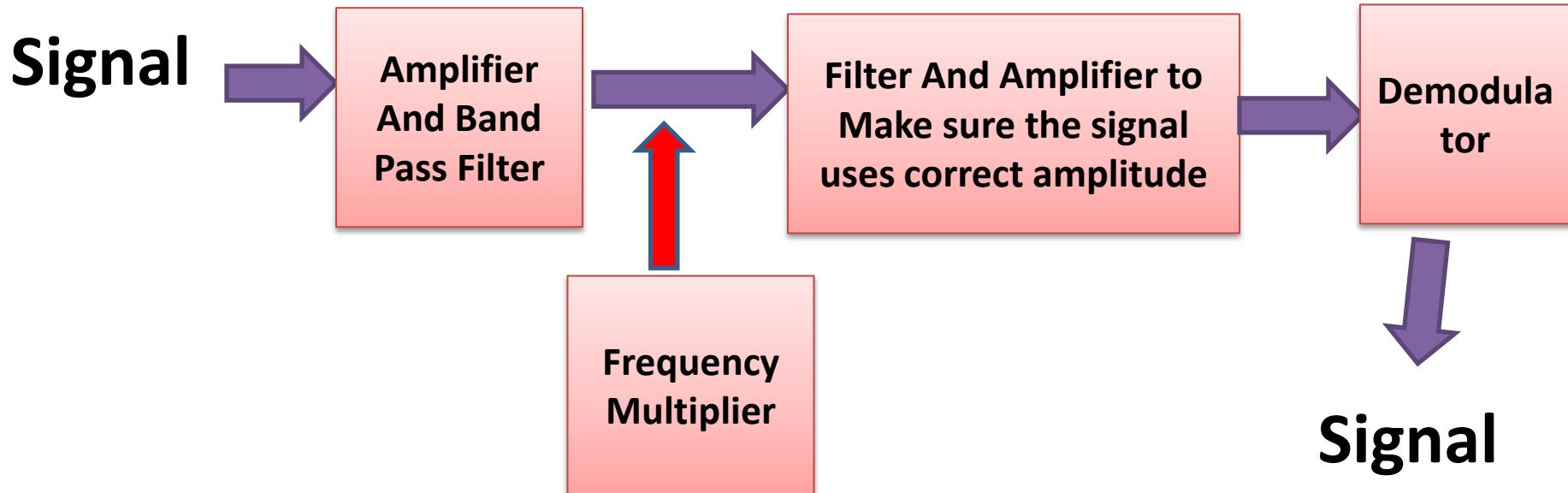
Adhoc Sensor n/w



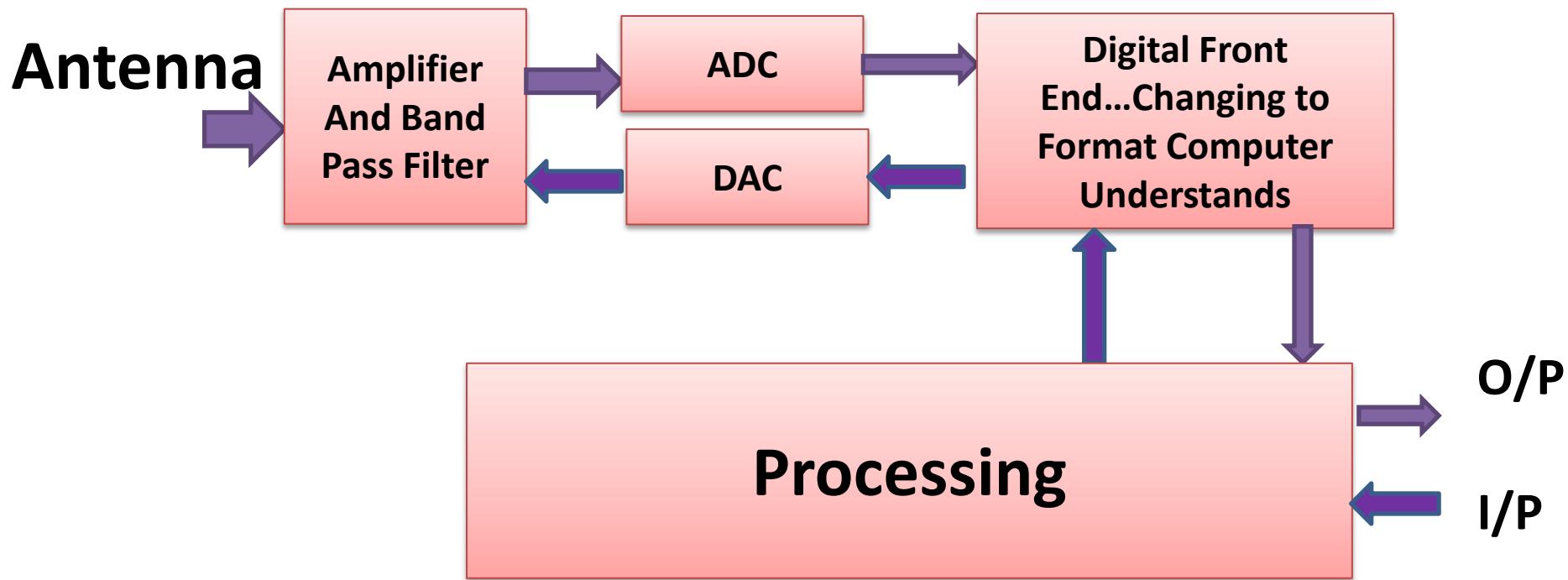
Software Defined Radio

- Using Software to replace Hardware to modulate the signals.
- **Hardware is Still needed for RF front end and ADC (Analog to Digital Converter)/ DAC (Digital to Analog Converter)**
- Advantage : We can receive many signals with single piece of hardware

H/w Radio Receiver



Software Defined Radio



Software Defined Radio

- SDR is the result of an evolutionary process from purely hardware-based equipment to fully software-based equipment.
- **All functions, modes, and applications, such as transmit frequencies, modulation type, and other RF parameters, can be configured and reconfigured by software**
- Software-defined refers to the use of software processing within the radio system or device to implement operating (but not control) functions

Cognitive Radio

- CR is a form of wireless communication in which a transceiver can intelligently detect which communication channels are in use and which are not, and instantly move into vacant channels while avoiding occupied ones.
- This optimizes the use of available RF spectrum while minimizing interference to other users
- SDR is a required basic platform on which to build a CR.
- Cognitive radios are radios that are aware of their environment and internal state and can make decisions about their radio operating behavior based on that information and predefined objective.

Satellite IoT

- <https://www.youtube.com/watch?v=hXa3bTclGPU>

Manage : TO Create Business Values

The business value of IoT comes from knowing how, when, and where to use the data in value-adding ways.

With IoT and analytics it is possible to make **business predictable** : E.g : predict when machinery would need maintenance

Manage : TO Create Business Values

Video analytics for b2c retail environment

- Installed equipment that use video cameras for detecting customers' features like age, gender and customer's mood. Based on the analysis the software determines which ad to display to each customer.
- These insights can be used to boost sales, help the customer find what they are looking for quicker and to test the attractiveness of different products, services and marketing messages.

Manage : TO Create Business Values

SnapSkan

- a service that detects tire tread depth when a customer drives into a parking hall.
- The depth of the tire tread is automatically scanned at the entrance and connected to the car's register plate number (only when customer makes query), ready to be presented to the customer when he makes the request via text message.

Manage : TO Create Business Values

Two Categories of IoT Business Value

- Increase Revenue
- Reduce Cost

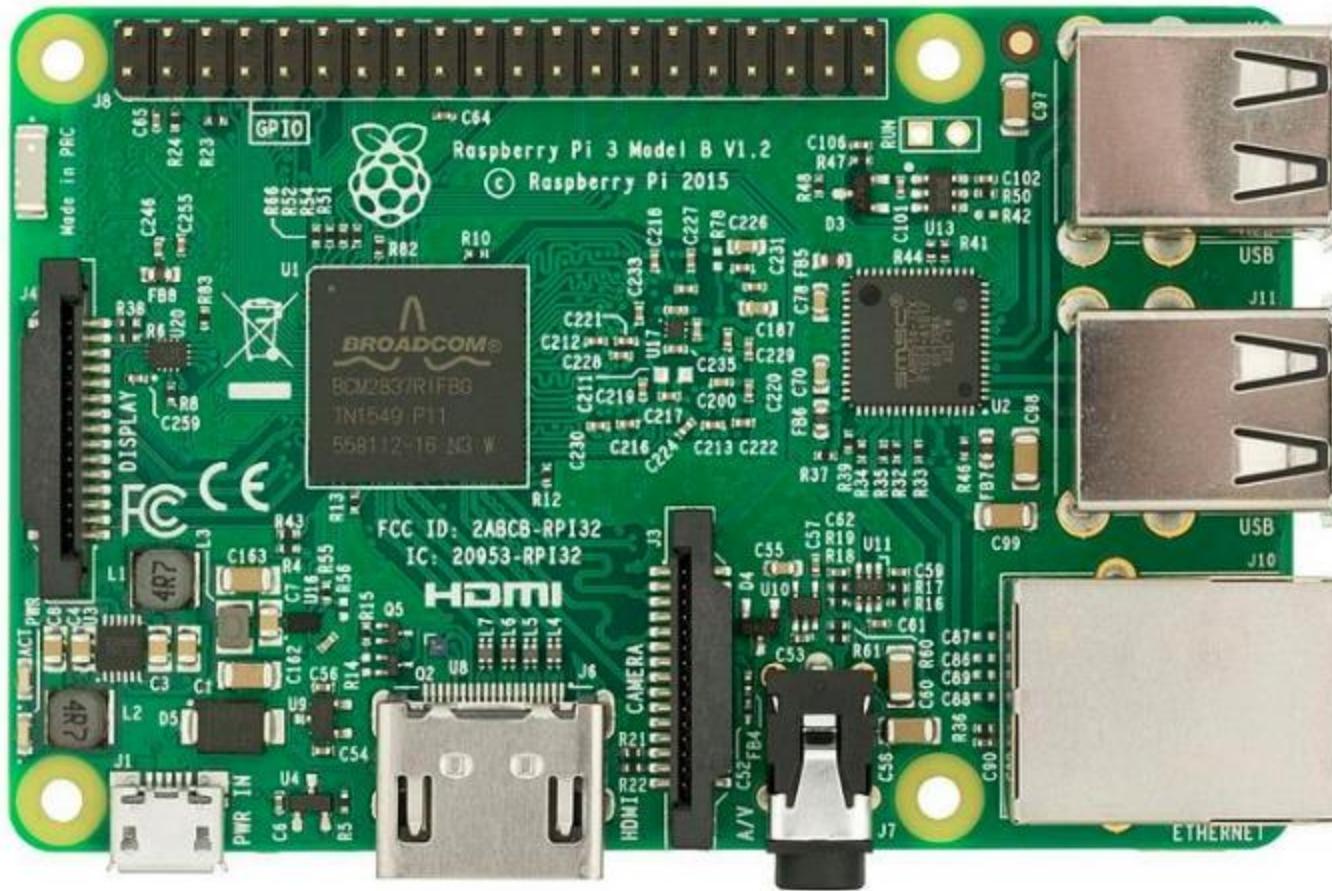
Manage : TO Create Business Values

- <https://www.youtube.com/watch?v=DfkGr8FurOs>

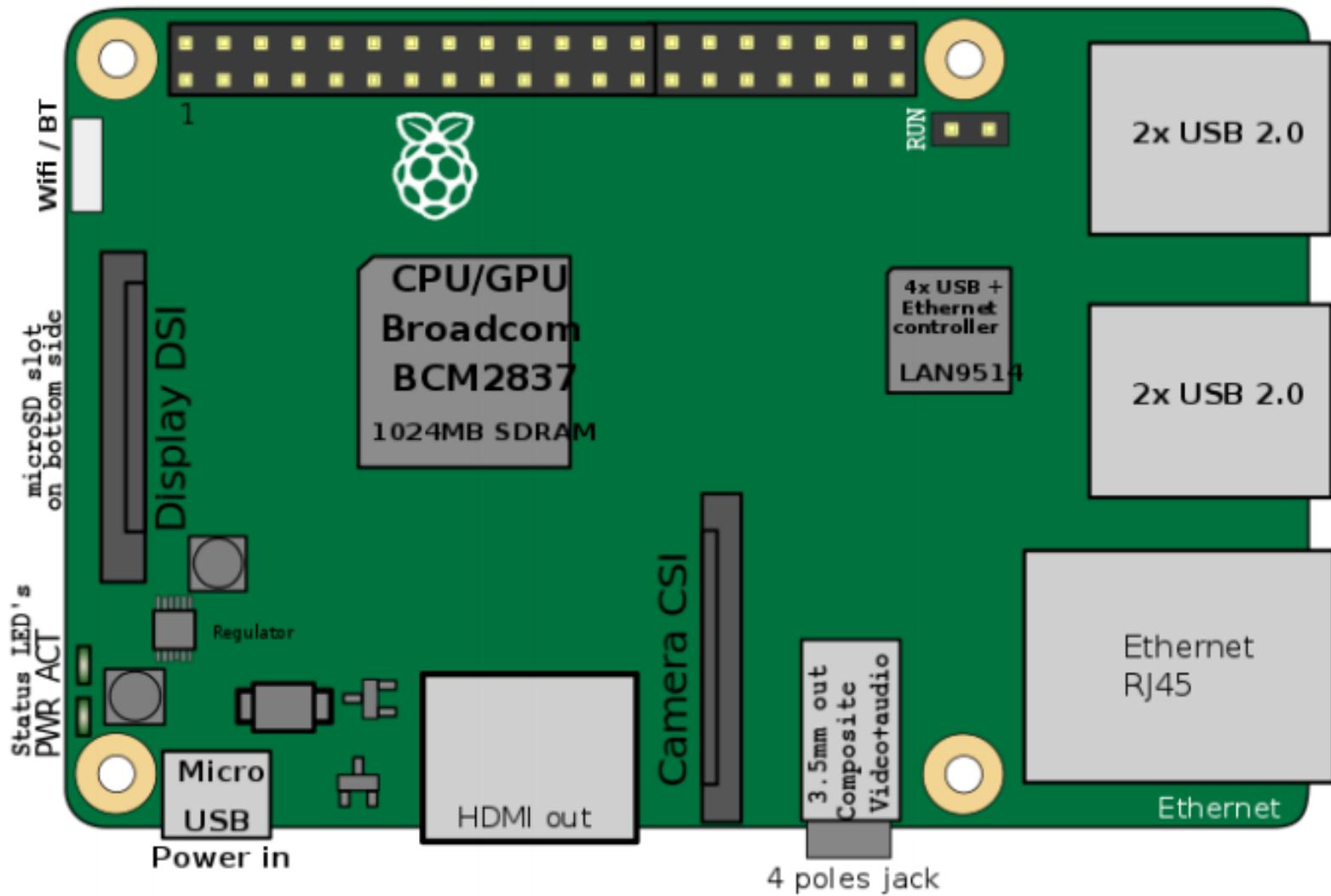
Raspberry Pi

- The Raspberry Pi is a **series of small single-board computers** developed in the **United Kingdom** by the **Raspberry Pi Foundation** to promote the teaching of basic computer science in schools and in developing countries.

Raspberry Pi 3 Model B



Raspberry Pi 3 Model B



Raspberry Pi ...TimeLine

- The first generation (Raspberry Pi 1 Model B) was released in February 2012.
- It was followed by a simpler and inexpensive model Model A.
- In 2014, the foundation released a board with an improved design in Raspberry Pi 1 Model B+. These boards are approximately credit-card sized and represent the standard mainline form-factor.
- Improved A+ and B+ models were released a year later. A "compute module" was released in April 2014 for embedded applications, and a Raspberry Pi Zero with smaller size and reduced input/output (I/O) and general-purpose input/output (GPIO) capabilities was released in November 2015 for US\$5.

Raspberry Pi ...TimeLine

- The Raspberry Pi 2 which added more RAM was released in February 2015.
- Raspberry Pi 3 Model B released in February 2016, is bundled with on-board WiFi, Bluetooth and USB boot capabilities.
- As of January 2017, Raspberry Pi 3 Model B is the newest mainline Raspberry Pi. Raspberry Pi boards are priced between US\$5–35.
- As of 28 February 2017, the Raspberry Pi Zero W was launched, which is identical to the Raspberry Pi Zero, but has the Wi-Fi and Bluetooth functionality of the Raspberry Pi 3 for US\$10.

Raspberry Pi ...Features

- Features a Broadcom **system on a chip** which includes an **ARM compatible** central processing unit (**CPU**) and an on-chip graphics processing unit (**GPU**).
- CPU speed ranges from **700 MHz to 1.2 GHz** for the Pi 3 and on board memory range from **256 MB to 1 GB RAM**.
- **Secure Digital (SD) cards** are used to store the **operating system** and program memory in either the SDHC or MicroSDHC sizes.

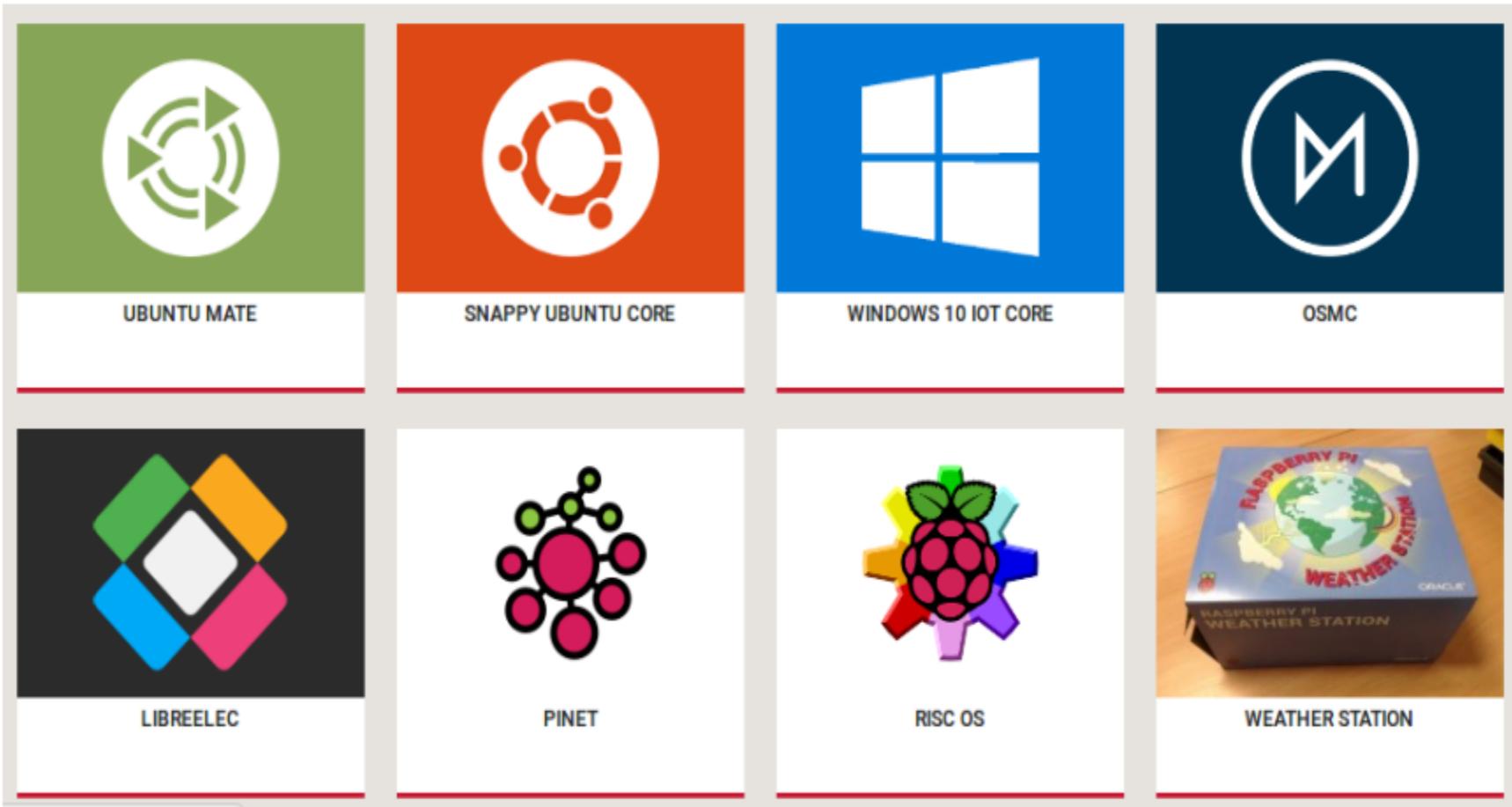
Raspberry Pi ...Features

- Most boards have between **one and four USB** slots, **HDMI and composite video output**, and a **3.5 mm phono jack for audio**.
- **Lower level output** is provided by a number of **GPIO pins** which support common protocols like I^2C .
- The B-models have an **8P8C Ethernet port** and the Pi 3 and Pi Zero W have **on board Wi-Fi 802.11n and Bluetooth**.

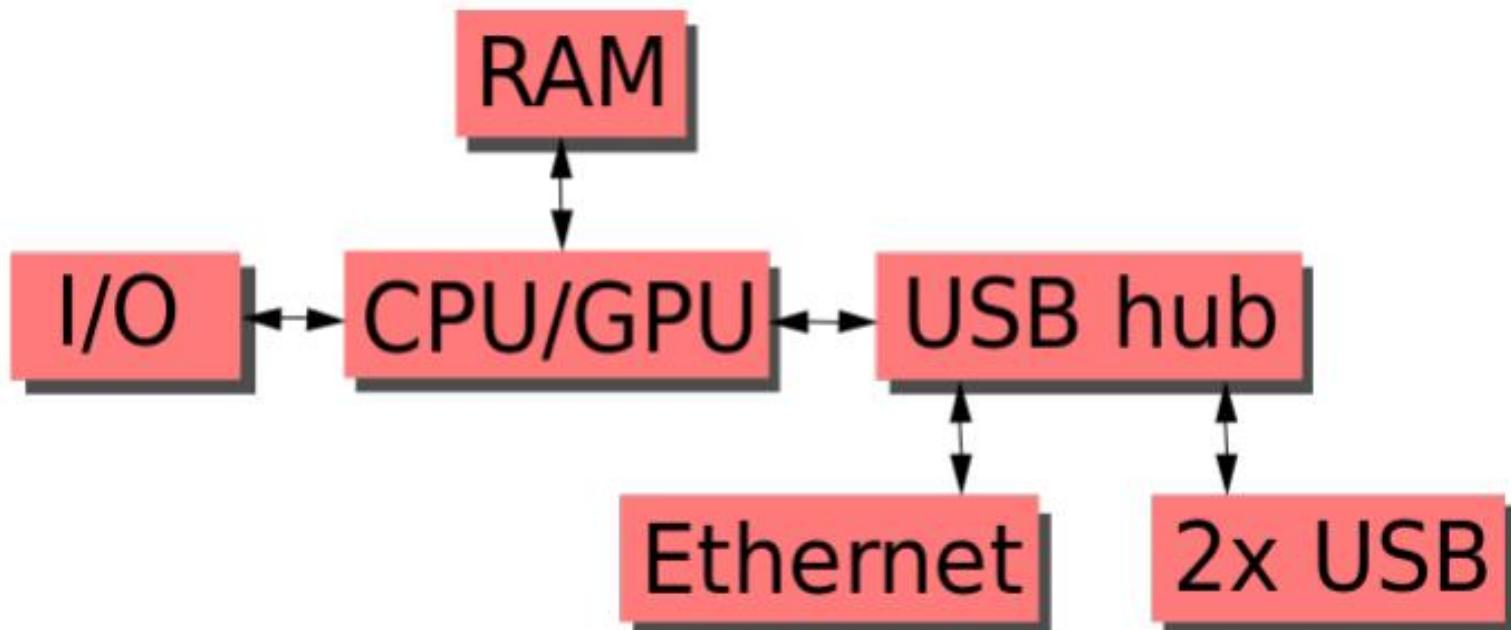
Raspberry Pi ...Operating System

- The Foundation provides **Raspbian**, a Debian-based Linux distribution for download, as well as third party **Ubuntu**, **Windows 10 IOT Core**, **RISC OS**, and specialised media center distributions.
- It promotes **Python** and **Scratch** as the main programming language, with support for many other languages.
- The default firmware is closed source, while an **unofficial open source is available**.

Raspberry Pi ...Operating System



Raspberry Pi ...Hardware



Raspberry Pi ...Hardware

	Raspberry Pi 1 Model A	Raspberry Pi 1 Model A+	Raspberry Pi 1 Model B	Raspberry Pi 1 Model B+	Raspberry Pi 2 Model B	Raspberry Pi 3 Model B	Raspberry Pi Zero
USB 2.0 Ports	1	1	2	4	4	4	1 (Micro-USB)
Ethernet	None	None	10/100 Mbit/s	10/100 Mbit/s	10/100 Mbit/s	10/100 Mbit/s	None
Bluetooth	None	None	None	None	None	4.1	None
WiFi	None	None	None	None	None	802.11n	None
Audio In	I ² S	I ² S					
Audio Out	I ² S, analog (3.5mm jack), digital (HDMI)	I ² S, analog (3.5mm jack), digital (HDMI)	I ² S, analog (3.5mm jack), digital (HDMI)	I ² S, analog (3.5mm jack), digital (HDMI)	I ² S, analog (3.5mm jack), digital (HDMI)	I ² S, analog (3.5mm jack), digital (HDMI)	Digital (mini-HDMI), analog GPIO PWM
Video In	CSI Camera Connector	None					
Video Out	HDMI, Composite (RCA)	HDMI, Composite (TRRS)	HDMI, Composite (RCA)	HDMI, Composite (TRRS)	HDMI, Composite (TRRS)	HDMI, Composite (TRRS)	Mini-HDMI, GPIO Composite
External Storage	SD	MicroSD	SD	MicroSD	MicroSD	MicroSD	MicroSD

Other Boards

- BeagleBone
- Banana Pi
- Orange Pi
- Ordroid

Other Boards

Parameter	Rpi	Orange Pi	Banana Pi	Odroid	Beaglebone
SOC Vendor	BroadCom	AllWinner	AllWinner	Amlogic	OMAP
CPU	Cortex A53	Cortex A7	Cortex A7	Cortex A53	CortexA8
CPU Freq	1.2 GHz	1.2-1.6GHz	1.8GHz	2GHz	1GHz
Memory	1GB	2GB	2GB	2GB	512MB
USB 2.0	400MHz	4+1 OTG	2+1OTG	4+1OTG	4+1
Ethernet	100Mb	1GB	1GB	1Gb	100Mb
Wireless	802.11n	802.11n	802.11n	None	None
Bluetooth	4.1	None	4.0	None	None
HDMI	1200P60	4KP30	1200P60	4K60	16b
Android	No	Yes	Yes	Yes	No

Unit – 4

IoT Protocols and

Security

Protocol Standardization for IoT

- IoT-Architecture one of the few efforts targeting a holistic architecture for all IoT sectors
- This consortium consists of 17 European organizations from nine countries
- Summarized current status of IoT standardization as
 - Fragmented architectures
 - No holistic approach to implement IoT has yet been proposed
 - Many island solutions do exist (RFID, sensor nets, etc.)
 - Little cross-sector reuse of technology and exchange of knowledge

M2M and WSN Protocols

- Most M2M applications are developed today in a highly customized fashion
- High-level M2M architecture from M2M Standardization Task Force (MSTF) does include fixed & other non cellular wireless networks
- Means it's generic, holistic IoT architecture even though it is M2M architecture
- M2M and IoT sometimes are used interchangeably in the United States

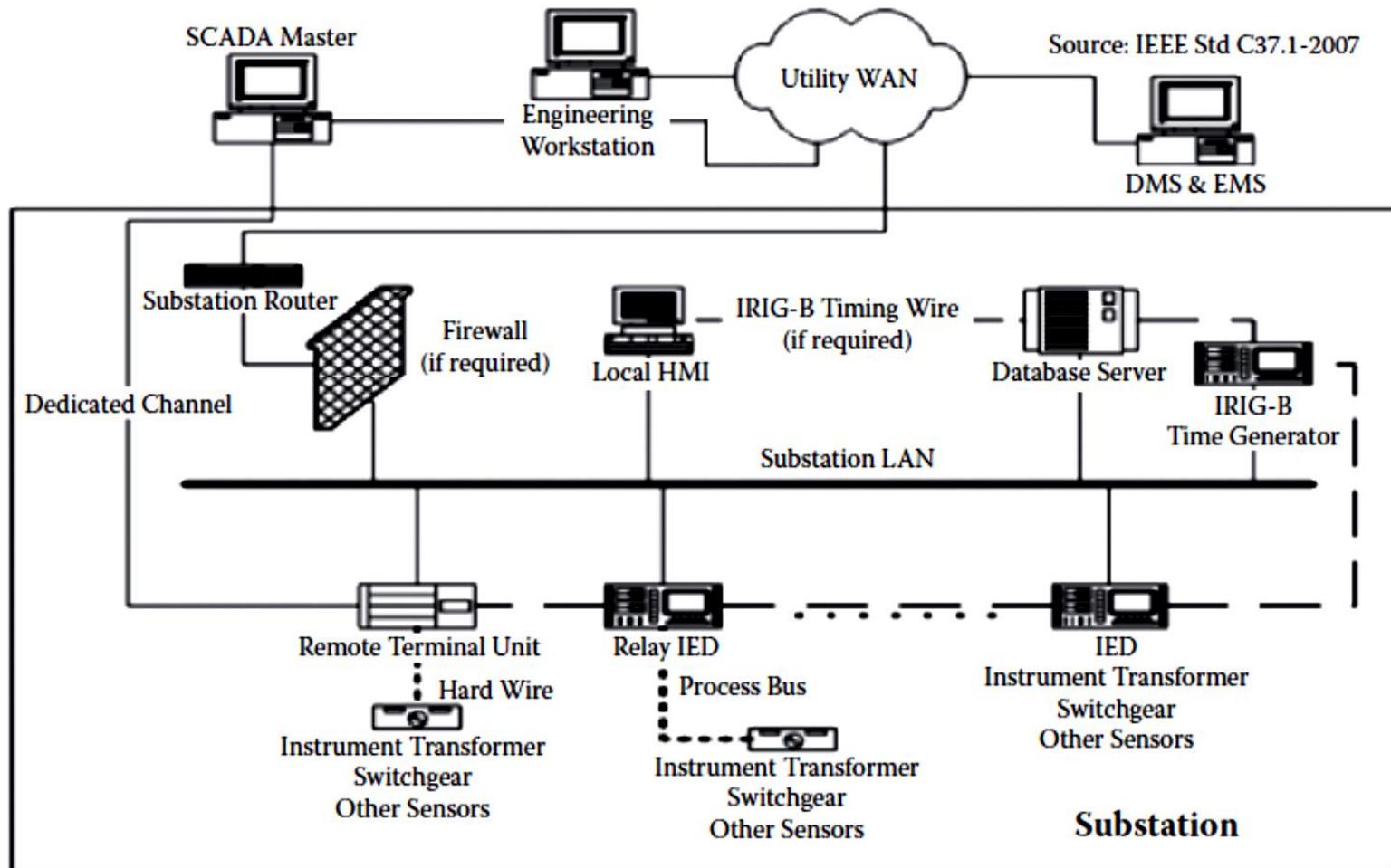
M2M and WSN Protocols

- Other M2M standards activities include:
 - Data transport protocol standards - M2MXML, JavaScript Object Notation (JSON), BiTXML, WMMP, MDMP
 - Extend OMA DM to support M2M devices protocol management objects
 - M2M device management, standardize M2M gateway
 - M2M security and fraud detection
 - Network API's M2M service capabilities
 - Remote management of device behind gateway/firewall
 - Open REST-based API for M2M applications

SCADA and RFID Protocols

- Supervisory Control And Data Acquisition
- One of the IoT pillars to represent the whole industrial automation arena
- IEEE created standard specification called Std C37.1™, for SCADA & automation systems in 2007
- In recent years, network-based industrial automation has greatly evolved
- With the use of intelligent electronic devices (IEDs), or IoT devices in our terms, in substations and power stations

SCADA and RFID Protocols



SCADA and RFID Protocols

- The processing is now distributed
- Functions that used to be done at control center can now be done by IED i.e. M2M between devices
- Due to restructuring of electric industry, traditional vertically integrated electric utilities are replaced by many entities such as
 - GENCO (Generation Company),
 - TRANSCO (Transmission Company),
 - DISCO (Distribution Company),
 - ISO (Independent System Operator), etc.

Issues with IoT Standardization

- It should be noted that not everything about standardization is positive
- Standardization is like a double-edged sword:
 - Critical to market development
 - But it may threaten innovation and inhibit change when standards are accepted by the market
- Standardization and innovation are like yin & yang
- They could be contradictory to each other in some cases, even though this observation is debatable

Issues with IoT Standardization

- Different consortia, forums and alliances have been doing standardization in their own limited scope
- For example, 3GPP covers only cellular wireless networks while EPCglobal's middleware covers only RFID events
- Even within same segment, there are more than one consortium or forum doing standardization without enough communication with each other
- Some are even competing with each other

Issues with IoT Standardization

- Some people believe that the IoT concept is well established
- However, some gray zones remain in the definition, especially which technology should be included
- Following two issues for IoT standardization in particular and ICT standardization in general may never have answers:

Issues with IoT Standardization

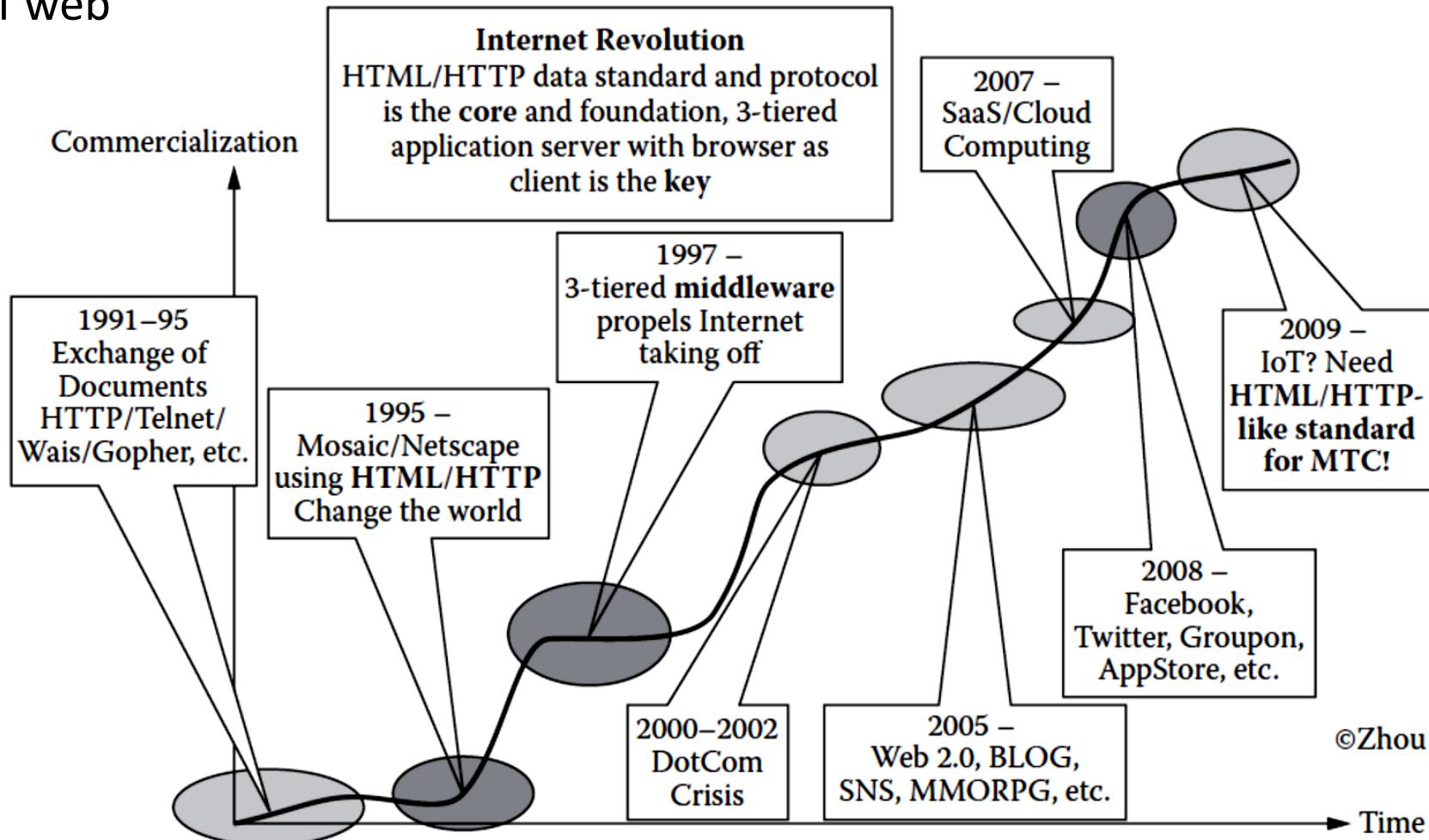
1. ICT standardization is a highly decentralized activity. How can the individual activities of the network of extremely heterogeneous standards-setting bodies be coordinated?
2. It will become essential to allow all interested stakeholders to participate in the standardization process toward the IoT and to voice their respective requirements and concerns. How can this be achieved?

Unified Data Standards

- Already discussed about two pillars of the Internet
- HTML/HTTP combination of data format and exchange protocol is the foundation pillar of WWW
- Described great number of data standards and protocols proposed for four pillar domains of IoT
- Many issues still impede the development of IoT and especially WoT vision

Unified Data Standards

Evolution of web



Unified Data Standards

- Many standardization efforts have been trying to define unified data representation, protocol for IoT
- Before IoT, Internet was actually an Internet of documents or of multimedia documents
- Two pillars of Internet including HTML/HTTP turned the Internet into WWW
- We need to turn the IoT into the WoT
- What will it take to make this to happen?

Unified Data Standards

- *Do we need a new HTML/HTTP-like standard for MTC and WoT? If there is no need to reinvent the wheel, what extensions do we need to build on top of HTML/HTTP or HTML5?*
- *Browser is intended for humans, so do we need new browser for machines to make sense of ocean of machine-generated data? If not, what extensions do we need to make to the existing browsers?*

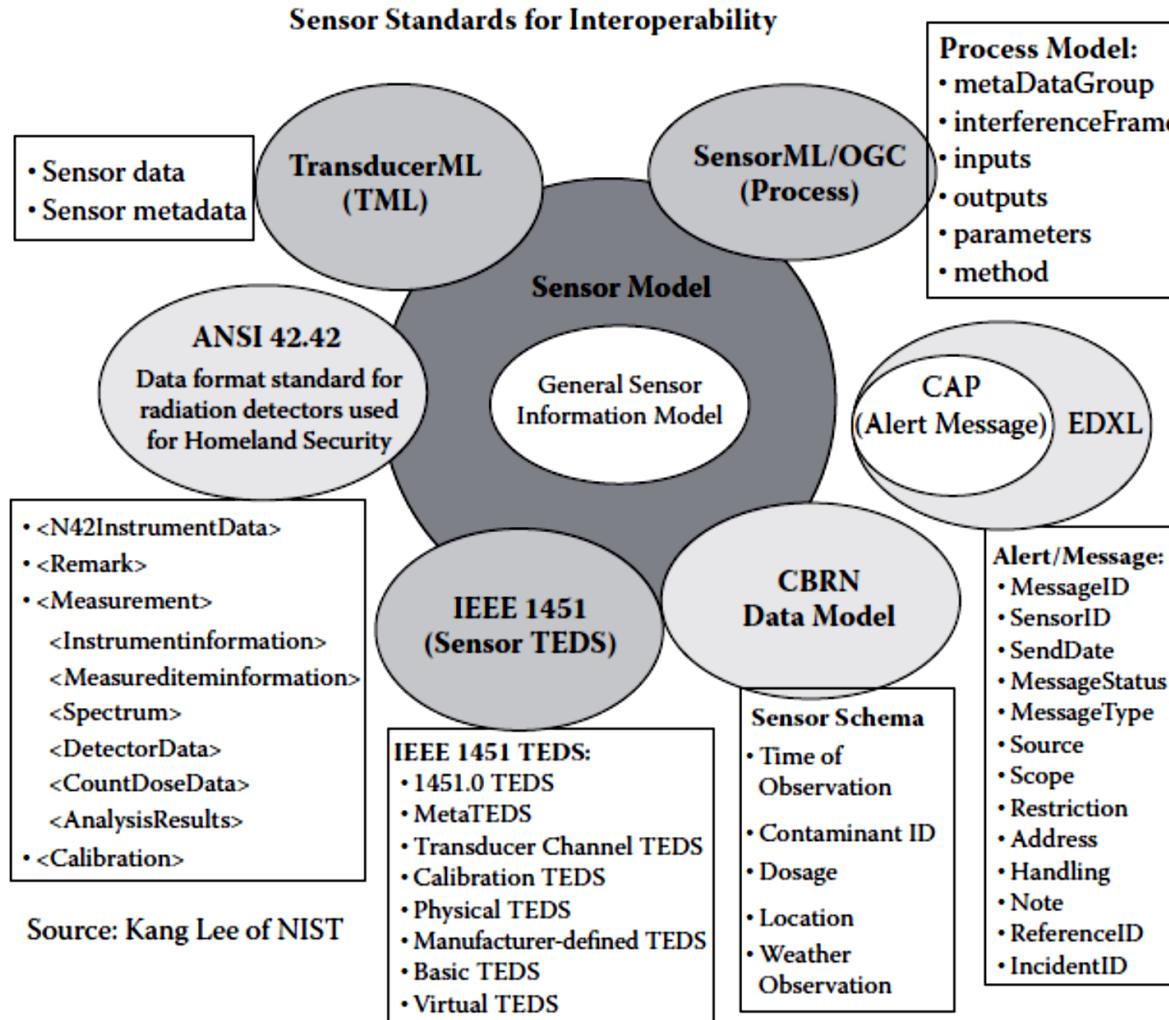
Unified Data Standards

- *Today, most new protocols are built on top of XML. For OS there must be XML-based data format standards or a metadata standard to represent the machine-generated data (MGD). Is it possible to define such a metadata standard that covers everything?*

Unified Data Standards

- There are many different levels of protocols
- But the ones that most directly relate to business and social issues are the ones closest to the top
- so-called application protocols such as HTML/HTTP for the web
- Web has always been visual medium, but restricted
- Until recently, HTML developers were limited to CSS & JavaScript in order to produce animations
- Or they would have to rely on a plug-in like Flash

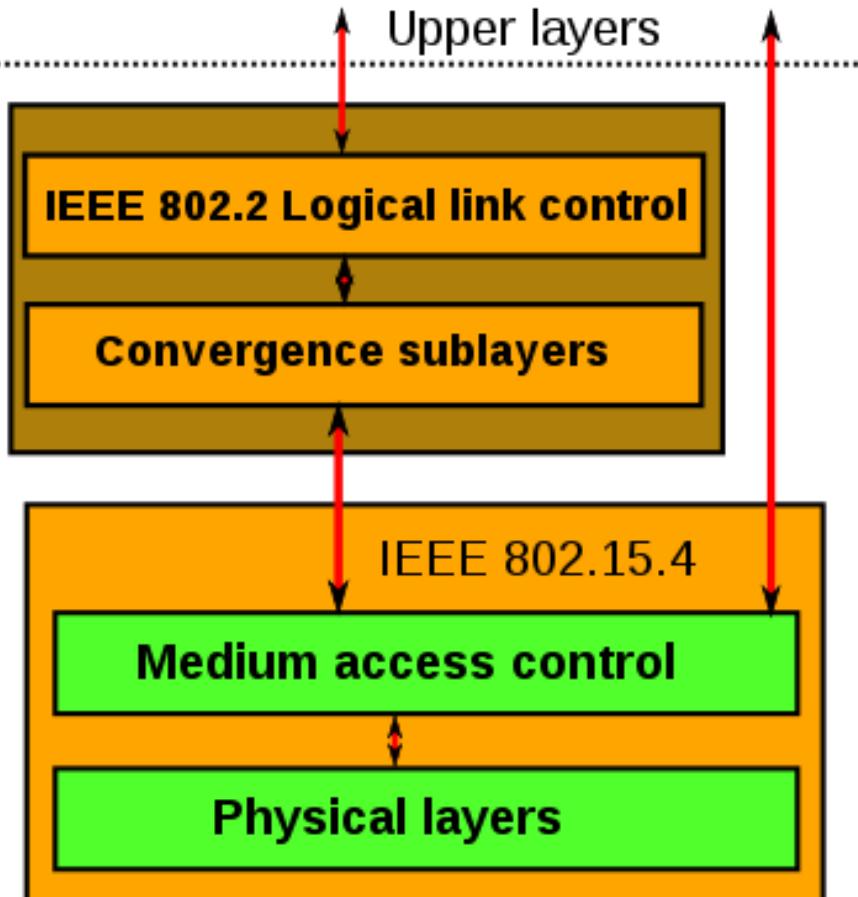
Unified Data Standards



Protocols – IEEE 802.15.4

- Defines operation of low-rate wireless personal area networks (LR-WPANs)
- Specifies physical layer and media access control for LR-WPANs
- Maintained by IEEE 802.15 working group, which defined the standard in 2003
- Basic framework conceives a 10m communications range with a transfer rate of 250 kbit/s

Protocols – IEEE 802.15.4



- *Physical Layer (PHY)* provides data transmission service & interface to *physical layer management entity*
- MAC enables transmission of MAC frames through the use of the physical channel

BACNet Protocol

- Communications protocol for Building Automation and Control (BAC) networks
- Provides mechanisms for computerized building automation devices to exchange information
- Designed to allow communication of building automation & control system for application like
 - Heating, Ventilating and Air-conditioning Control (HVAC)
 - Lighting Control, Access Control
 - Fire Detection Systems and their Associated Equipment

BACNet Protocol

- Defines a number of services that are used to communicate between building devices
- Protocol services include Who-Is, I-Am, Who-Has, I-Have which are used for Device & Object discovery
- Services such as Read-Property and Write-Property are used for data sharing
- Defines 60 object types that are acted upon by services
- Defines no. of data link/physical layers including

BACNet Protocol

- ARCNET,
- Ethernet,
- BACnet/IP,
- BACnet/IPv6,
- Point-To-Point over RS-232,
- Master-Slave/Token-Passing over RS-485,
- ZigBee
- LonTalk

Modbus

- Serial communications protocol originally published by Modicon (now Schneider Electric) in 1979
- Commonly available for connecting industrial electronic devices
- Reasons for use of Modbus in industrial environment:
 - Developed with industrial applications in mind
 - Openly published and royalty-free
 - Easy to deploy and maintain
- Enables communication among many devices connected to the same network

Modbus Object Types

Object type	Access	Size
Coil	Read-write	1 bit
Discrete input	Read-only	1 bit
Input register	Read-only	16 bits
Holding register	Read-write	16 bits

Protocol Versions

- Modbus RTU
- Modbus ASCII
- Modbus TCP/IP or Modbus TCP
- Modbus over TCP/IP or Modbus over TCP or Modbus RTU/IP
- Modbus over UDP
- Modbus Plus (Modbus+, MB+ or MBP)
- Pemex Modbus
- Enron Modbus

KNX Protocol

- Standardized (EN 50090, ISO/IEC 14543), OSI-based network communications protocol for automation
- Defines several physical communication media:
 - Twisted pair wiring (inherited from the BatiBUS and EIB Instabus standards)
 - Powerline networking (inherited from EIB and EHS - similar to that used by X10)
 - Radio (KNX-RF)
 - Infrared
 - Ethernet (also known as EIBnet/IP or KNXnet/IP)

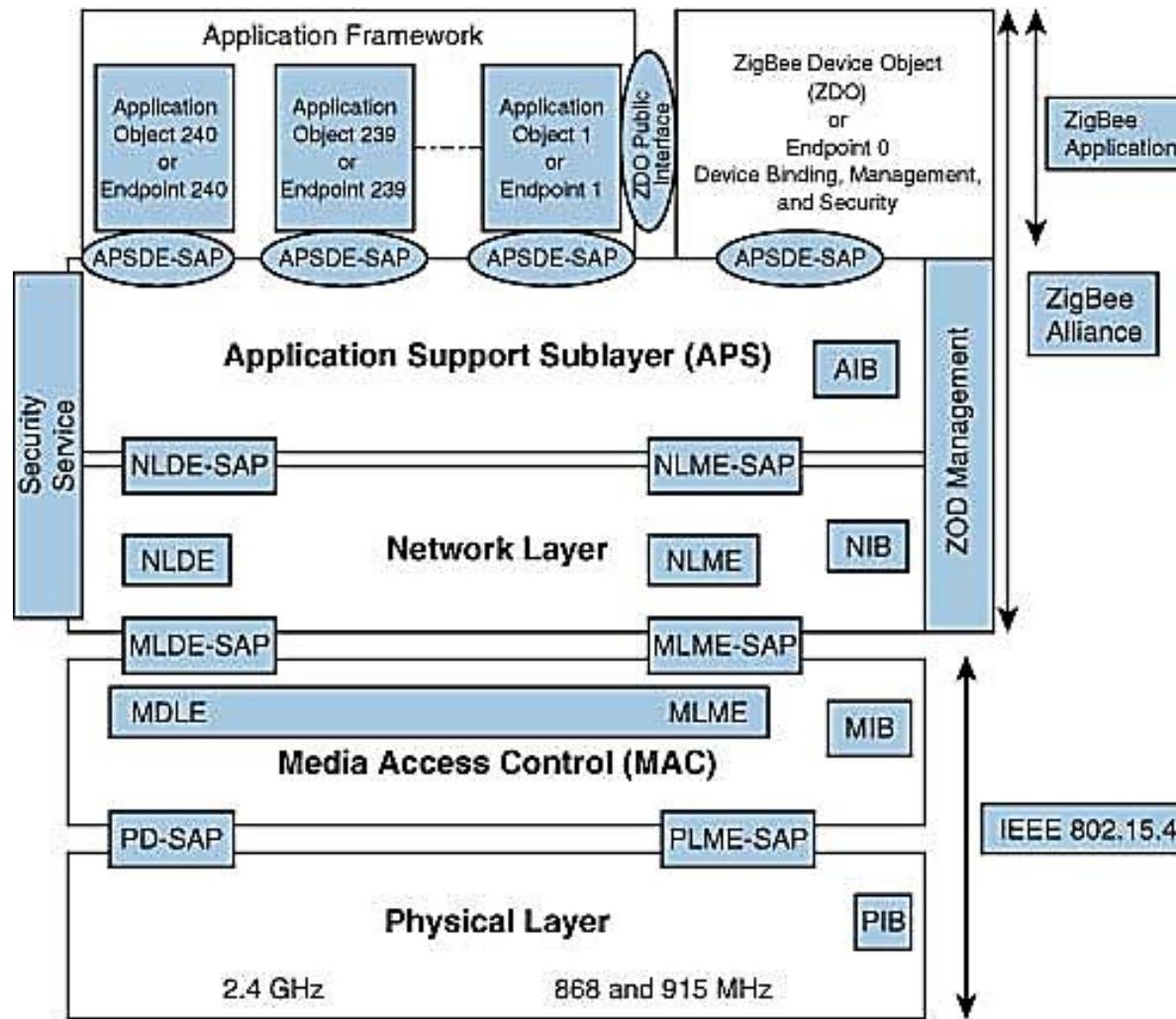
KNX System Components

- All the devices for a KNX installation are connected together by a two wire bus to exchange data
- Sensors
- Actuators
- System devices and components

ZigBee

- IEEE 802.15.4-based specification for a suite of high-level communication protocols
- Used to create personal area networks with small, low-power digital radios
- ZigBee based applications
 - Home Automation
 - Medical Device Data Collection
 - other low-power low-bandwidth

ZigBee Architecture



ZigBee Architecture

- Divided into three sections
 - IEEE 802.15.4 which consists of MAC and physical layers
 - ZigBee layers, which consist of the network layer, the ZigBee device object (ZDO), the application sublayer, and security management
 - Manufacturer application: Manufacturers of ZigBee devices can use the ZigBee application profile or develop their own application profile

Network Layer

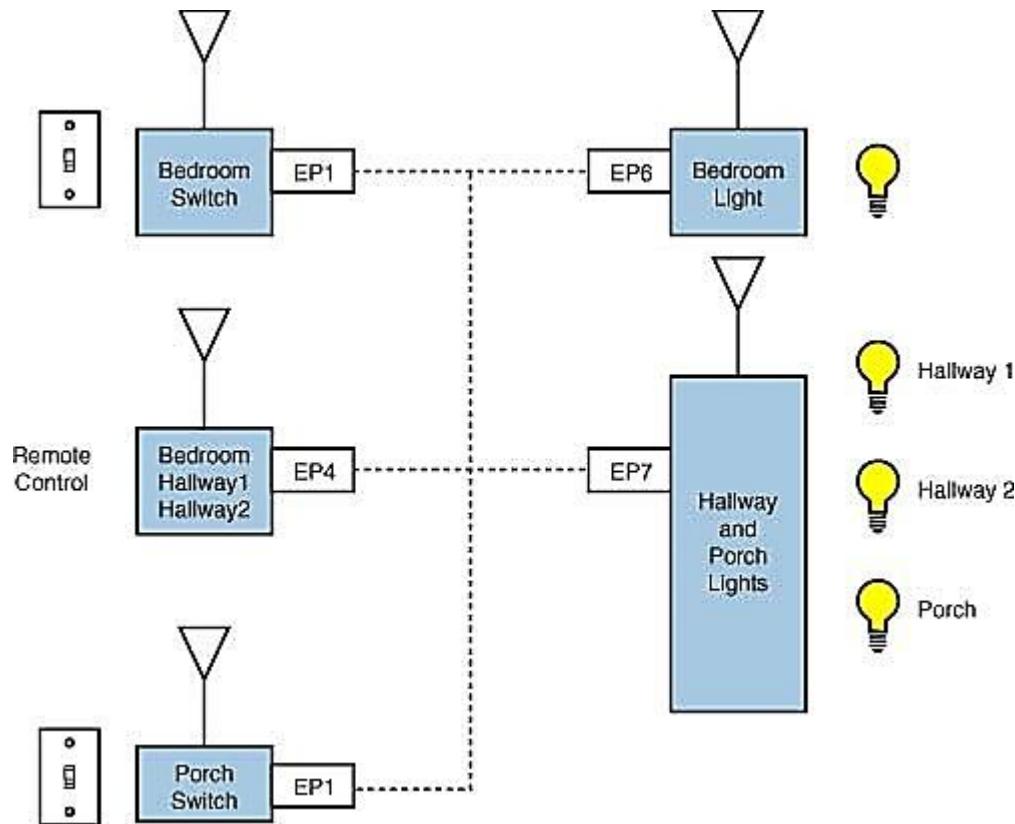
- Located between the MAC layer and application support sublayer
- Provides the following functions:
 - Starting a network
 - Managing end devices joining or leaving a network
 - Route discovery
 - Neighbor discovery

APS Layer

- Application Support Sublayer (APS)
- Provides services necessary for application objects (endpoints) and the ZigBee device object (ZDO)
- Some of services provided by the APS to the application objects for data transfer are
 - Request
 - Confirm
 - Response

APS Layer

- Application Object (endpoint)
 - Defines input and output to the APS
 - For example, a switch that controls a light is the input from the application object, and the output is the light bulb condition
 - Each node can have 240 separate application objects



APS Layer

- ZigBee Device Object (ZDO)
 - Control and management of application objects
 - Performs overall device management tasks:
 - Determines the type of device in a network (for example, end device, router, or coordinator)
 - Initializes the APS, network layer, and security service provider
 - Performs device and service discovery
 - Initializes coordinator for establishing a network
 - Security management
 - Network management

APS Layer

- End Node
 - Each end node or end device can have multiple EPs
 - Each EP contains an application profile, such as home automation
 - can be used to control multiple devices or single device
- ZigBee Addressing Mode
 - ZigBee uses direct, group, and broadcast addressing for transmission of information

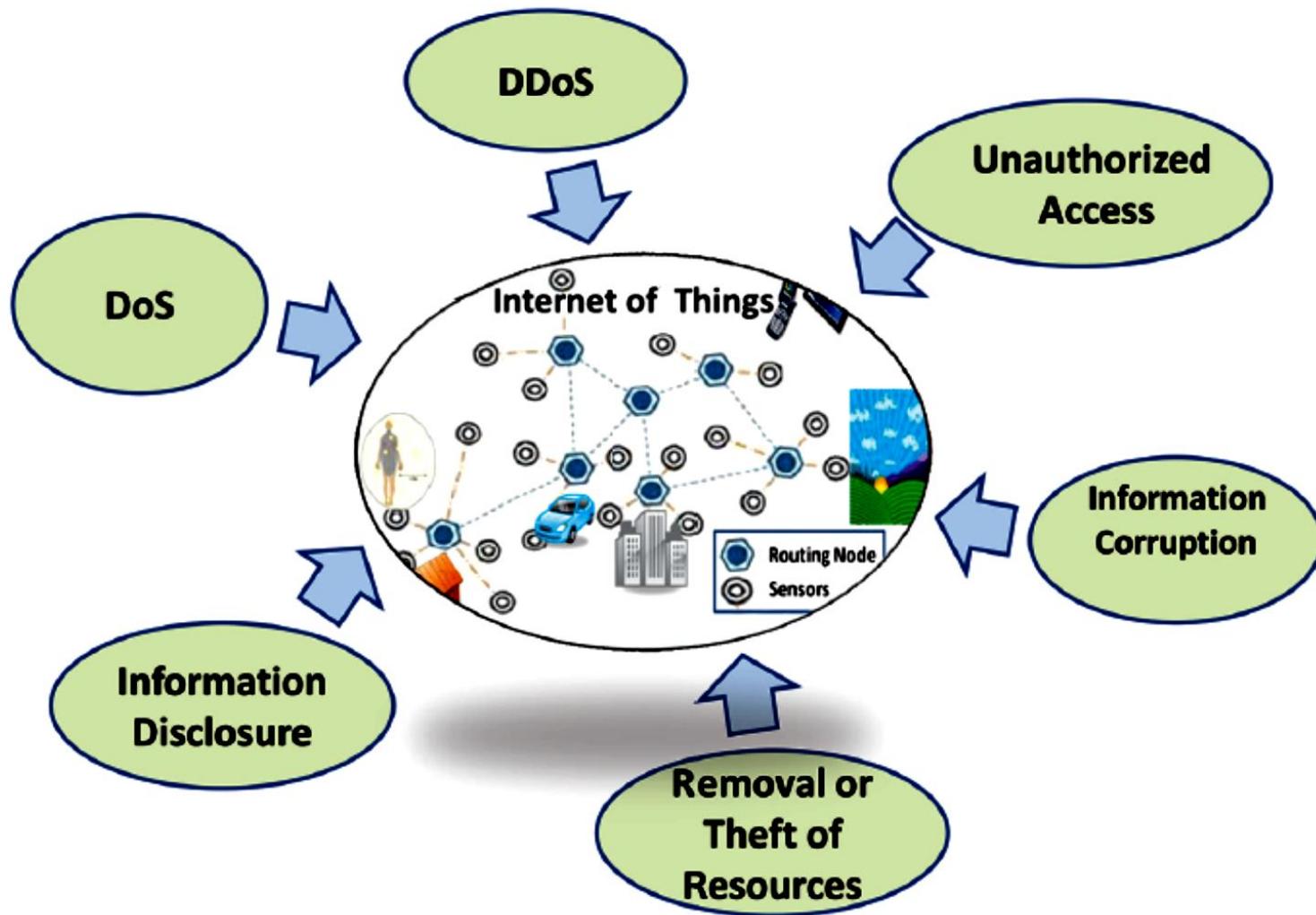
IOT Security

- Fundamental idea - IoT will connect all objects around us to provide smooth communication
- Economic of scale in IoT presents new security challenges for global devices in terms of
 - Authentication
 - Addressing
 - Embedded Security

IOT Security

- Devices like RFID and sensor nodes have no access control functionality
- Can freely obtain or exchange information from each other
- So authentication & authorization scheme must be established between these devices to achieve the security goals for IoT
- Privacy of things and security of data is one of the key challenges in the IoT

Vulnerabilities of IoT



Vulnerabilities of IoT

- Unauthorized Access
 - One of the main threats is the tampering of resources by unauthorized access
 - Identity-based verification should be done before granting the access rights
- Information corruption
 - Device credential must be protected from tampering
 - Secure design of access rights, credential and exchange is required to avoid corruption

Vulnerabilities of IoT

- Theft of Resources
 - Access of shared resources over insecure channel causes theft of resources
 - Results into man-in-the-middle attack
- Information Disclosure
 - Data is stored at different places in different forms
 - Distributed data must be protected from disclosure
 - Context-aware access control must be enforced to regulate access to system resources

Vulnerabilities of IoT

- DoS Attack
 - Denial of Service (DoS)
 - Makes an attempt to prevent authentic user from accessing services which they are eligible for
 - For example, unauthorized user sends too many requests to server
 - That flood the network and deny other authentic users from access to the network

Vulnerabilities of IoT

- DDoS Attack
 - Distributed Denial of Service
 - Type of DoS attack where multiple compromised systems are used to target single system causing DoS
 - Compromised systems – usually infected with Trojan
 - Victims of a DDoS attack consist of both
 - End targeted systems
 - All systems maliciously used and controlled by the hacker in the distributed attack

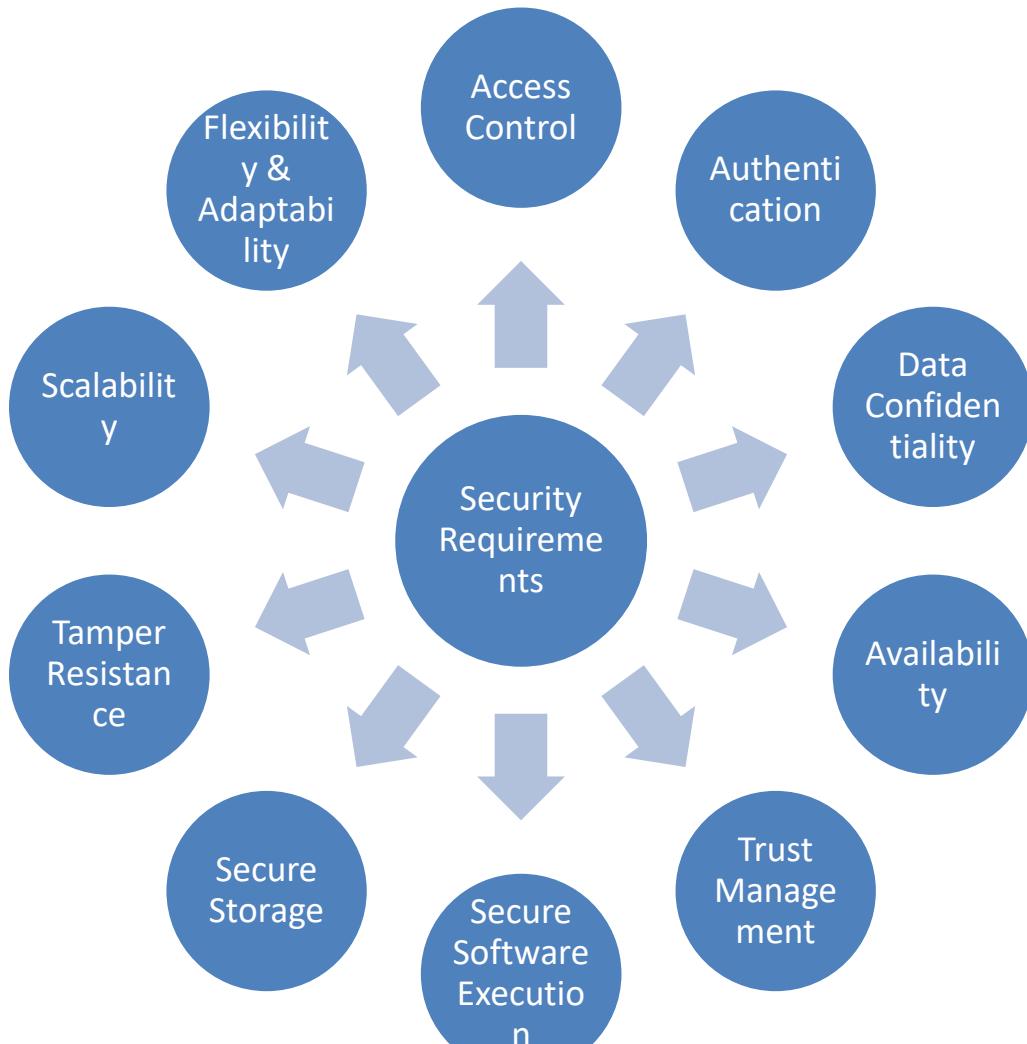
Vulnerabilities of IoT

- CyberBunker Launches “World’s Largest” DDoS Attack
- Slows down the Entire Internet
- CyberBunker - Dutch web hosting company
- Caused global disruption of the web
- Slowing down internet speeds for millions of users across the world, according to BBC report

Vulnerabilities of IoT

- Few real examples of attacks that hit the IoT:
 - Carna Botnet – 4,20,000 ‘things,’ such as routers, modems, printers were compromised
 - TRENDnet’s connected cameras were hacked, with feeds from those cameras published online
 - Linux.Darlloz - PoC IoT worm found in the wild by Symantec, 1,00,000 compromised systems including connected things such as TVs, routers and even a fridge

Security Requirements



Security Requirements

- Access Control
 - Provides authorized access to network resources
 - IoT is ad-hoc, and dynamic in nature
 - Efficient & robust mechanism of secure access to resources must be deployed with distributed nature
- Authentication
 - Identity establishment b/w communicating devices
 - Due to diversity of devices & end users, an attack resistant and lightweight solution for authentication

Security Requirements

- Data Confidentiality
 - Protecting data from unauthorized disclosure
 - Secure, lightweight, and efficient key exchange mechanism is required
- Availability
 - Ensuring no denial of authorized access to network resources

Security Requirements

- Trust Management
 - Decision rules needs to be evolved for trust management in IoT
- Secure Software Execution
 - Secure, managed-code, runtime environment designed to protect against different applications
- Secure Storage
 - Involves confidentiality and integrity of sensitive information stored in the system

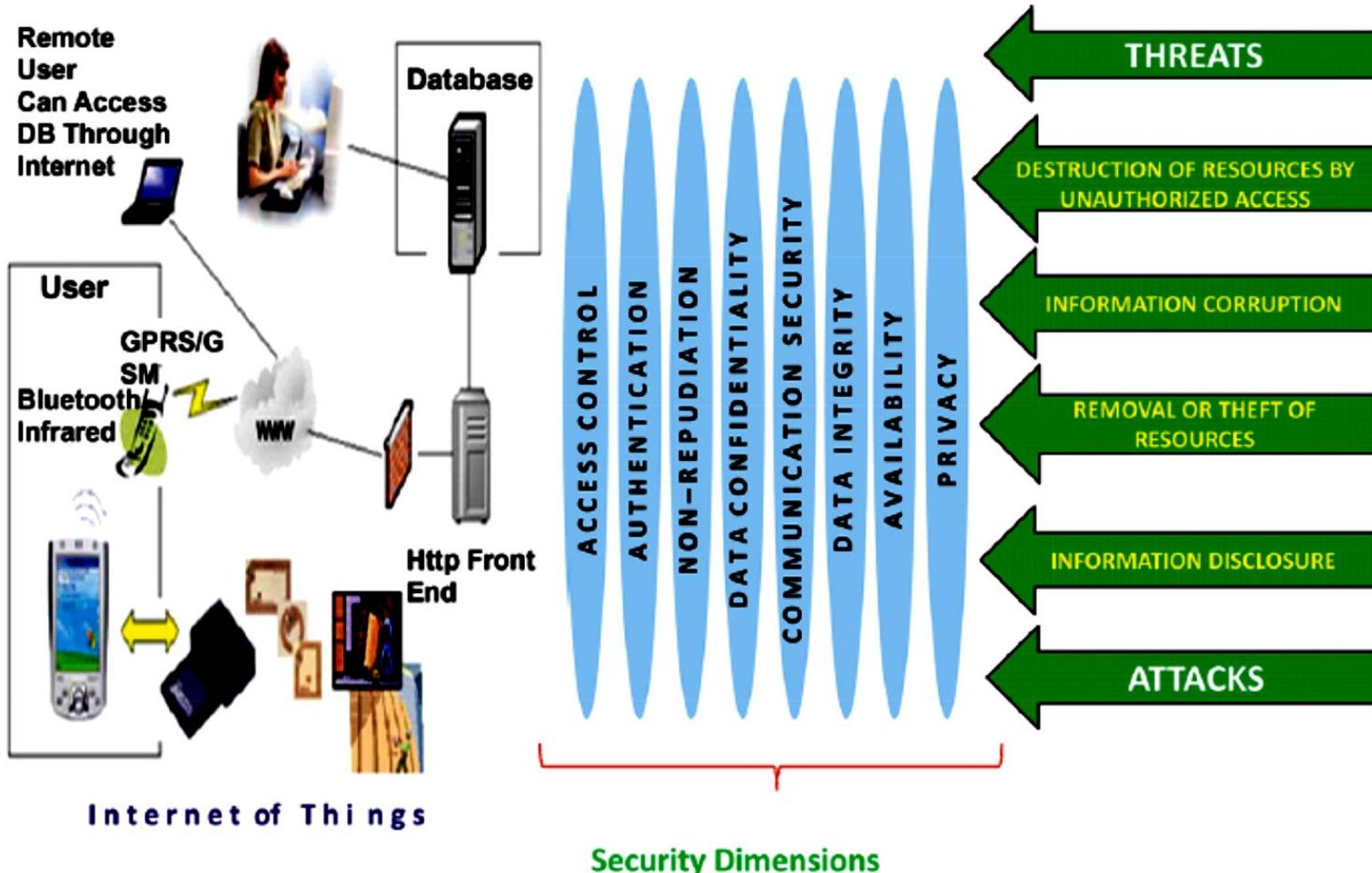
Security Requirements

- Tamper Resistance
 - Desire to maintain security requirements even when device falls into hands of malicious parties
 - Can be physically or logically probed
- Scalability
 - IoT consist of various types of devices with different capabilities from intelligent sensors and actuators, to home appliances
 - Communication (wire or wireless) & protocols (Bluetooth, ZigBee, RFID, Wi-Fi, etc.)

Security Requirements

- Flexibility and Adaptability
 - IoT will consist of mobile communication devices
 - Can roam around freely from one type of environment to others
 - With different type of risks and security threats
 - So users are likely to have different privacy profile depending on environment

Security Architecture for IoT



Threat Modeling

- Presented by first defining misuse case
- Means negative scenario describing the ways the system should not work
- And then standard use case
- Assets to be protected in IoT will vary with respect to every scenario case

Threat Analysis

- Assets needs to be identified to drive threat analysis process
- Smart home is localized in space, provide services in a household
- Devices in Smart Home are combined with n/w
- Provide means for entertainment, monitoring of appliances, controlling of house components and other services

Use Cases and Misuse Cases

- Actor in use case and misuse case in the scenario of smart home includes:
 - Infrastructure owner (smart home)
 - IoT entity (smartphone device or software agent)
 - Attacker (misuser)
 - Intruder (exploiter)

Use Cases and Misuse Cases

- Access rights granted to unauthorized entity
- Corruption of access credentials
- Unauthorized data transmission
- Denial of service (DoS) attack
- Man-in-the-middle attack

IoT Security Tomography

- Classified according to attacks addressing to different layers
 - Transport Layer
 - Network Layer
 - MAC layer
 - RF layer

IoT Security Tomography

Possible Threats	Layers	Possible Threats
	Transport Layer	Send wrong data Inject wrong control packets
Wormhole attack	Network Layer	Routing loop Network partitioning
Buffer overflows OS threat	MAC Layer	Spoofing Eavesdropping
Hardware threat Sensor threat	RF Layer	Complete jamming Eavesdropping Replay attacks

Key Elements of Security

- Authentication
- Access Control
- Data and Message Security
- Prevention from denial of taking part in a transaction

Identity Establishment

- Secure Entity Identification or Authentication
- Authentication is identity establishment between communicating devices or entities
- Entity can be a single user, a set of users, an entire organization or some networking device
- Identity establishment is ensuring that origin of electronic document & message is correctly identified

Access Control

- Also known as access authorization
- Principles is to determine who should be able to access what
- Prevents unauthorized use of resources
- To achieve access control, entity which trying to gain access must be authenticated first
- According to authentication, access rights can be modified to the individual

Data and Message Security

- Related with source authenticity, modification detection and confidentiality of data
- Combination of modification & confidentiality of message is not enough for data integrity
- But origin of authenticity is also important
- Location privacy is equally important risk in IoT
- Should not be any way for attacker to reveal identity or location information of device

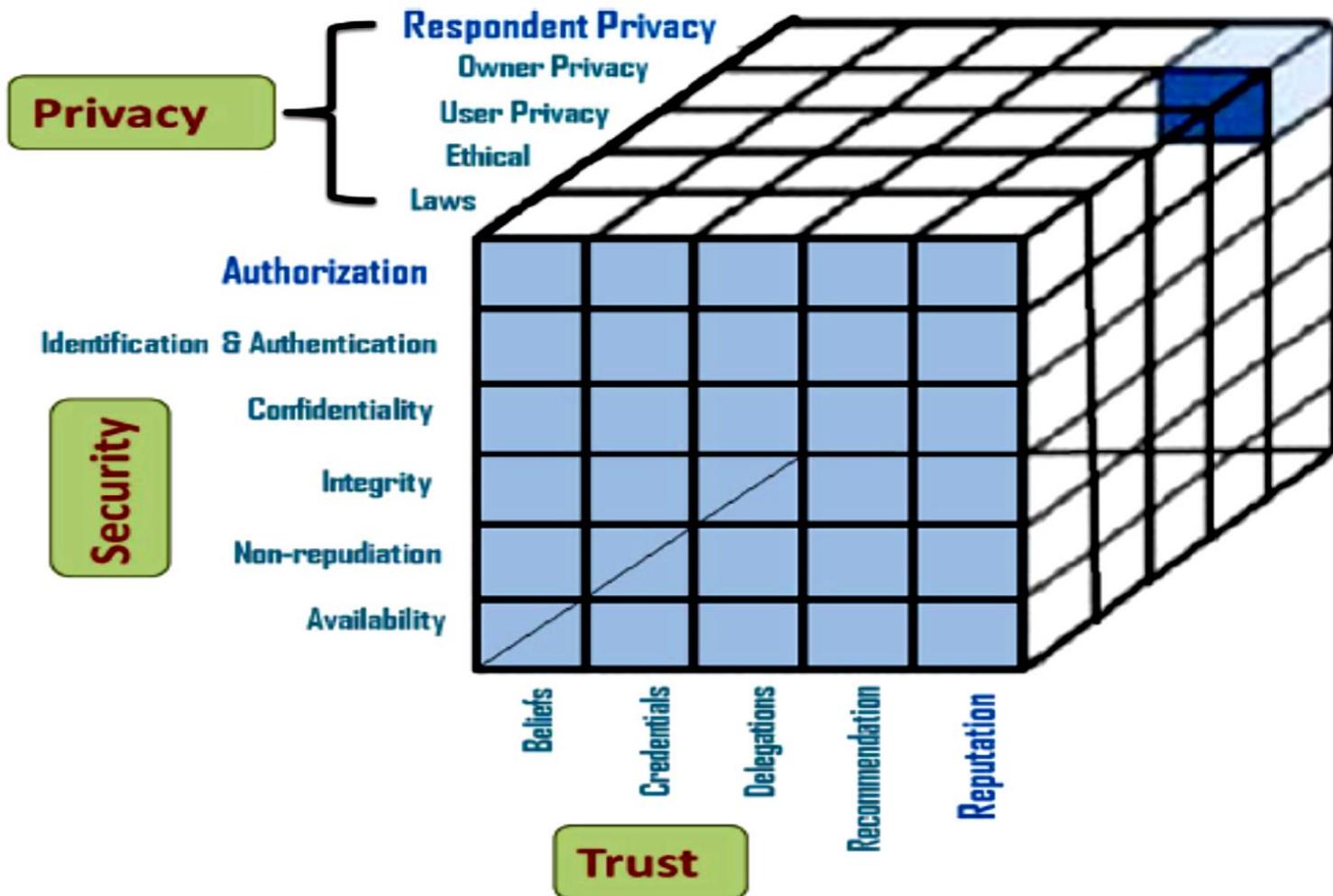
Non-repudiation and Availability

- Non-repudiation is the security services for point-to-point communications
- Process by which an entity is prevented from denying a transmitted message
- So when message is sent, receiver can prove that initiating sender only sent that message
- Sender can prove that receiver got message
- To repudiate means to deny

Non-repudiation and Availability

- Availability is ensured by maintaining all h/w, repairing immediately whenever required
- Also prevents bottleneck occurrence by keeping emergence backup power systems
- And guarding against malicious actions like Denial of Service (DoS) attack

Security Model for IoT



[Intel IoT -- What Does The Internet of Things Mean?](#)

<https://youtu.be/Q3ur8wzzhBU>

IoT Protocols And Security

IoT Standardization Efforts

- The IoT- A (Internet of Things architecture) is targeting a **holistic(universal) architecture for all IoT sectors.**
- 17 European organizations from nine countries are a part of IoT- A.
- **They summarize the current status of IoT standardization as :**

Current IoT Standardization acc to IoT-A

Fragmented architectures

No universal approach to implement IoT has yet been proposed

Many island solutions do exist (RFID, sensor nets, etc.)

Little cross-sector reuse of technology and exchange of knowledge

Current IoT Standardization is a
problem , SO

What could be Done to Solve this???

Proposed Solution By IoT-A for Standardization

Create Architectural foundation for IoT, that will be operable with future Internet

Use Existing technologies instead of creating new ones.

Demonstrating the applicability of IoT in a set of use cases

Establish a strong stakeholder group to remove the barriers and accept IoT on wide scale

Combine various IoT technologies into a single entity.

Groups doing IoT Standardization

Work Package Framework (**WPF**)

International Telecommunication Union
Telecommunication Standardization

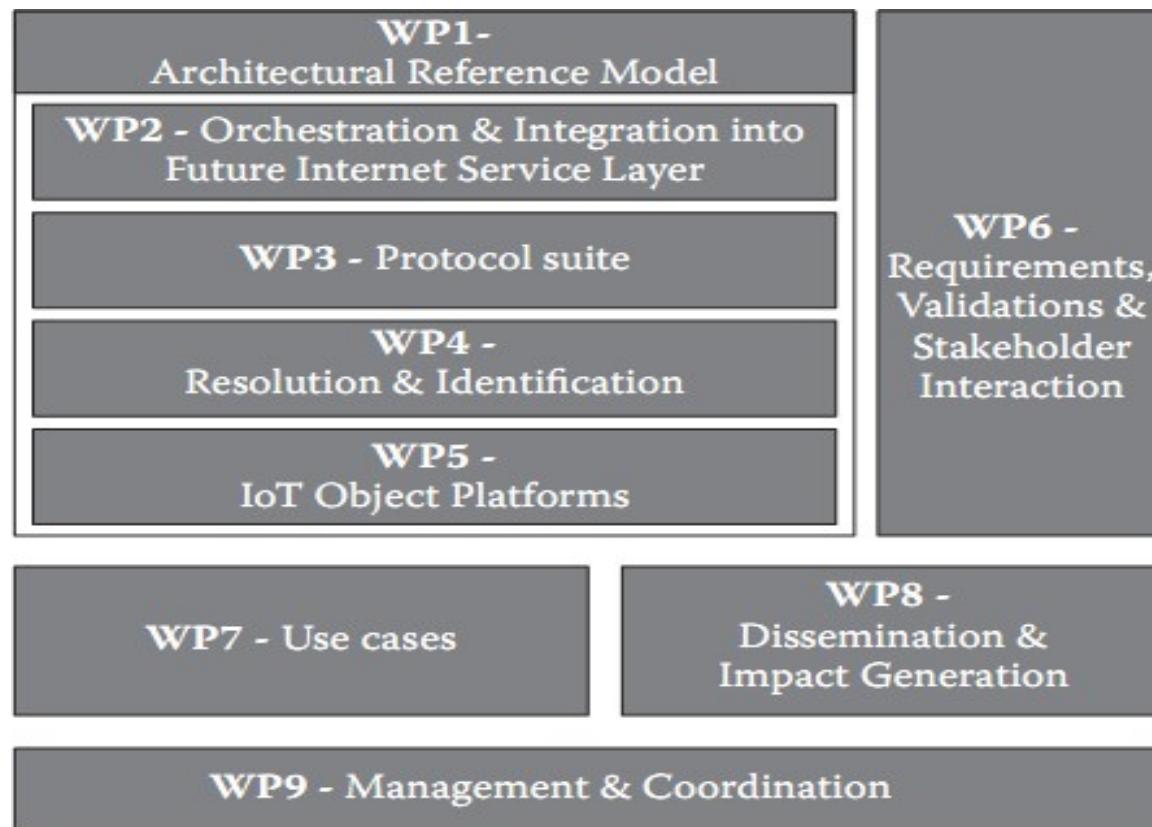
Sector(**ITU-T**)

Internet Protocol for Smart Objects (**IPSO**)

- Aim to form an open group of companies to market and educate about how to use IP for IoT smart objects based on an all- IP holistic

Work Package Framework

- The WPF divides the implementation standards of IoT into hierarchical groups of tasks as shown in the below



International Telecommunication Union

Telecommunication Standardization Sector

- The ITU-T mission is to ensure the **efficient and timely production of standards** covering all fields of **telecommunications** on a worldwide basis, as well as defining tariff and accounting principles for international telecommunication services.
- The technical work, the development of Recommendations, of ITU-T is managed by Study Groups (SGs).
- There are currently 11 SGs.

International Telecommunication Union

Telecommunication Standardization

	ITU-T Study Group	Study Group Name	Activities Related to IoT
Current Standards Activities	SG 2	Operational aspects of service provision and telecommunication management	Numbering, naming and addressing
	SG 3	Tariff and accounting principles including related telecommunication economic and policy issues	
	SG 5	Environment and climate change	
	SG 9	Television and sound transmission and integrated broadband cable networks	
	SG 11	Signalling requirements, protocols and test specifications	Testing architecture for tag-based identification systems and functions
	SG 12	Performance, QoS and QoE	
	SG 13	Future networks including mobile and NGN	NGN requirements and architecture for applications and services using tag-based ID
	SG 15	Optical transport networks and access network infrastructures	
	SG 16	Multimedia coding, systems and applications	Requirements and architecture for multimedia information access triggered by tag-based ID
Pre-standards	Focus Groups	SG 17	Security
		Smart Grid	Smart metering, M2M
		Cloud Computing	Cloud network requirements, e.g., for IoT
		Future Networks	Describe future networks underlying the IoT
		Car Communication	

Internet Protocol for Smart Objects

- The IPSO Alliance is an open, informal and thought-leading association of like-minded organizations and individuals that **promote the value of using the Internet Protocol for the networking of Smart Objects.**
- **IP Stack** can easily run on tiny, battery operated embedded devices as it is long-lived and stable technology.
- The role of the alliance is to ensure how IPv4, IPv6, and 6LoWPAN are used, deployed and provided to all potential users.

Internet Protocol for Smart Objects

- Mobile IP is an approach by IETF (Internet Engineering Task Force) which manages the movement of mobile devices over IPV4 and IPV6

M2M Standardization Efforts

M2M Standardization Task Force (MSTF) coordinate the efforts of individual **standards development organizations** (SDO) for M2M Applications

These **define a conceptual framework** for M2M applications and **specify a service layer** that will enable application developers to create applications that operate transparently across different vertical domains

M2M Standardization Efforts

M2M standards activities include the following

- Use JSON as Data Transport Format
- Resolve IP addressing issues for devices IPV6
- Use Open REST- based API for M2M applications
- Remote management of devices behind a gateway or firewall be done
- Fix the charging standars

WSN Standardization Efforts

There are number of standardization bodies in the field of WSNs

The IEEE focuses on the physical and MAC layers

IETF Internet Engineering Task Force works on layers 3 and above.

WSN Standardization Efforts

IEEE 1451 is a set of smart transducer interface standards developed by the IEEE Instrumentation and Measurement Society's Sensor Technology Technical Committee that describe a set of open, common, network- independent communication interfaces for connecting sensors or actuators) to microprocessors, instrumentation systems, and control/field networks

The goal of the IEEE 1451 family of standards is to allow the access of transducer data through a common set of interfaces whether the transducers are connected to systems or networks via a wired or wireless means

WSN Standardization Efforts ... IEEE 1451 Activities

1451.0-2007 Common Functions, Communication Protocols

1451.1-1999 Network Capable Application Processor Information Model

1451.2-1997 Transducer to Microprocessor Communication Protocols

1451.3-2003 Digital Communication Formats for Distributed Multi-drop Systems

WSN Standardization Efforts

...IEEE

1451 Activities

1451.4-2004 Mixed- mode Communication Protocols

1451.5-2007 Wireless Communication Protocols

1451.7-2010 Transducers to Radio Frequency Identification
(RFID) Systems Communication Protocols

SCADA Standardization Efforts

IEEE created a standard specification, called Std C37.1™, for SCADA and automation systems

The processing is now distributed, and functions that used to be done at the control center can now be done by the intelligent electronic devices (IED) that is, M2M between devices.

SCADA Standardization Efforts

The ISA100 was developed by the standards committee of the Industrial Society for Automation formed to **define procedures for implementing wireless systems in the automation and control environment with a focus on the field level.**

SCADA Standardization Efforts

OPC, which stands for Object Linking and Embedding (OLE) for Process Control standard specification developed by an **industrial automation industry task force (IAITF)**

The standard specifies the communication of **real-time plant data** between control devices from different manufacturers.

OPC was designed to provide a common bridge for Windows-based software applications and process control hardware.

What is SCADA

<https://www.youtube.com/watch?v=nlFM1q9QPJw>

[What is SCADA?](#)

RFID Standardization Efforts

The RFID protocols and data formats are relatively well defined, mostly by EPCglobal (Electronic Product Code)

The standard for contactless smart card communications is ISO/ IEC 14443 allows for communications at distances up to 10 cm.

ISO/ IEC 15693, which allows communications at distances up to 50 cm

International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC).

Summary

IoT-A

- WPF, ITUT, IPSO

M2M

- MSTF

WSN

- IEEE which developed
IEEE1451 , IETF

SCADA

- IEEE developed C37.1 , ISA ,
IAITF developed OPC

RFID

- EPCGlobal developed ISO/IEC
14443 and ISO/IEC 15693

Issues of IoT Standardization

Standardization is like a double-edged sword: critical to market development, but it may threaten innovation and inhibit change when standards are accepted by the market.

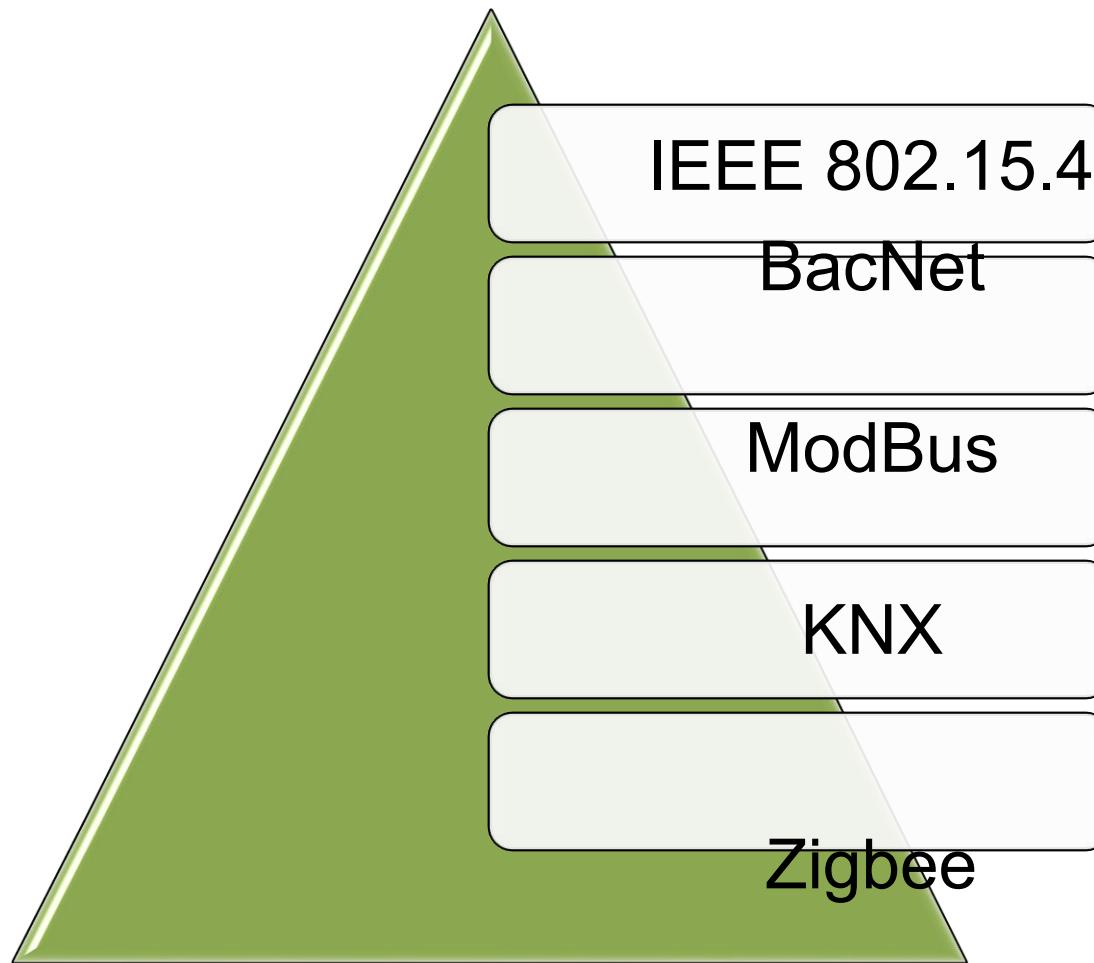
Issues of IoT Standardization

- Different consortia, forums, and alliances have been doing standardization
- Even within the same segment, there are more than one consortium or forum doing standardization
- ICT standardization is a highly decentralized activity.

Unified Data Standards in IoT for data Exchange

- Use **XML** representation for data exchange/transfer.
- **Resource Description framework** (RDF) can be used for modeling the information that is deployed as web resource.
- Use REST API
- **ebXML** can be used for e-commerce solutions
- IEEE 1451

IoT Protocols



Wireless

Wireless communication standards:

- IEEE 802.11 a/b/g
- Bluetooth
- GSM

What makes them unattractive for WSN:

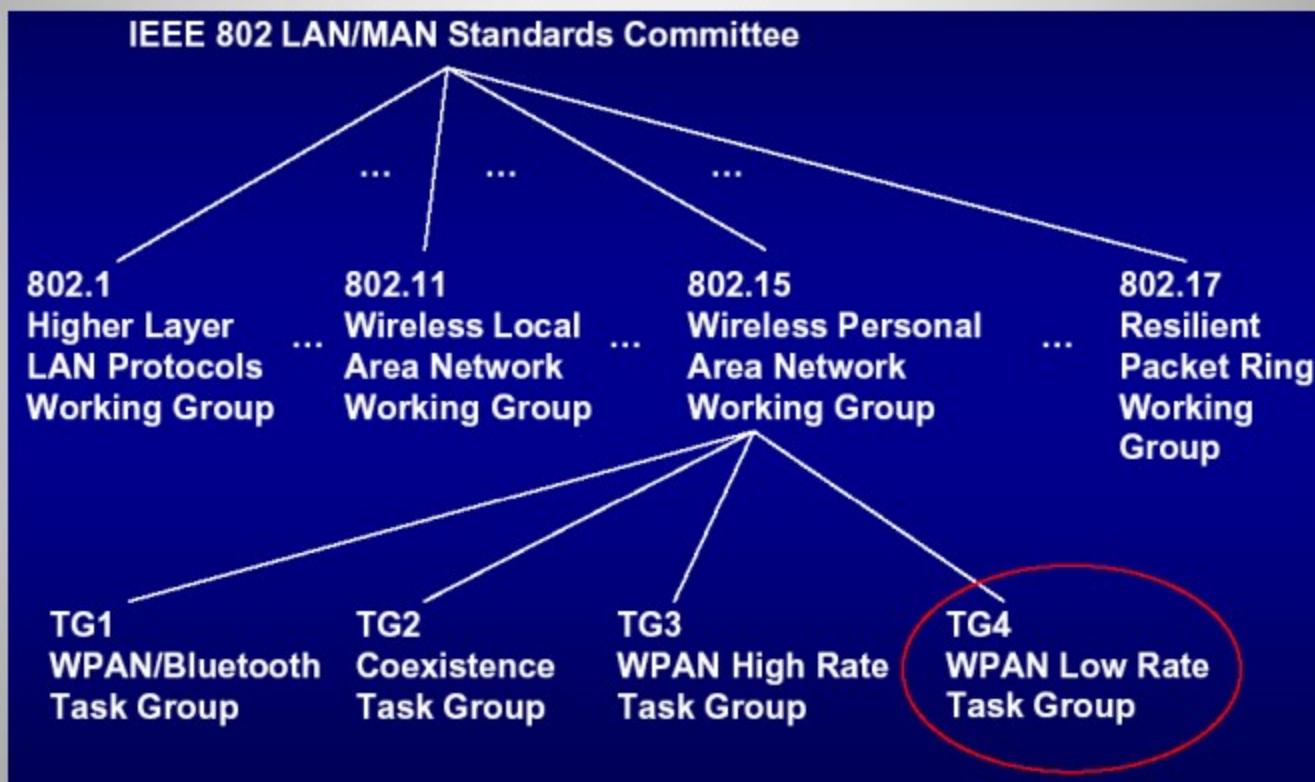
- Power hungry (need big batteries)
- Complexity (need lots of clock cycles and memory)

New protocol for WSN:

- 802.15.4
- Zigbee

802.15.4

IEEE 802.15 working group



802.15.4

- IEEE 802.15.4 task group began to develop a standard for LR-WPAN.
- The goal of this group was to provide a standard with ultra-low complexity, cost, and power for low-data-rate wireless connectivity among inexpensive fixed, portable, and moving devices.

802.15.4

Property	Range
Raw data rate	868 MHz: 20 kb/s; 915 MHz: 40 kb/s; 2.4 GHz: 250 kb/s
Range	10–20 m
Latency	Down to 15 ms
Channels	868/915 MHz: 11 channels 2.4 GHz: 16 channels
Frequency band	Two PHYs: 868 MHz/915 MHz and 2.4 GHz
Addressing	Short 8-bit or 64-bit IEEE
Channel access	CSMA-CA and slotted CSMA-CA

Approaches for Low Power

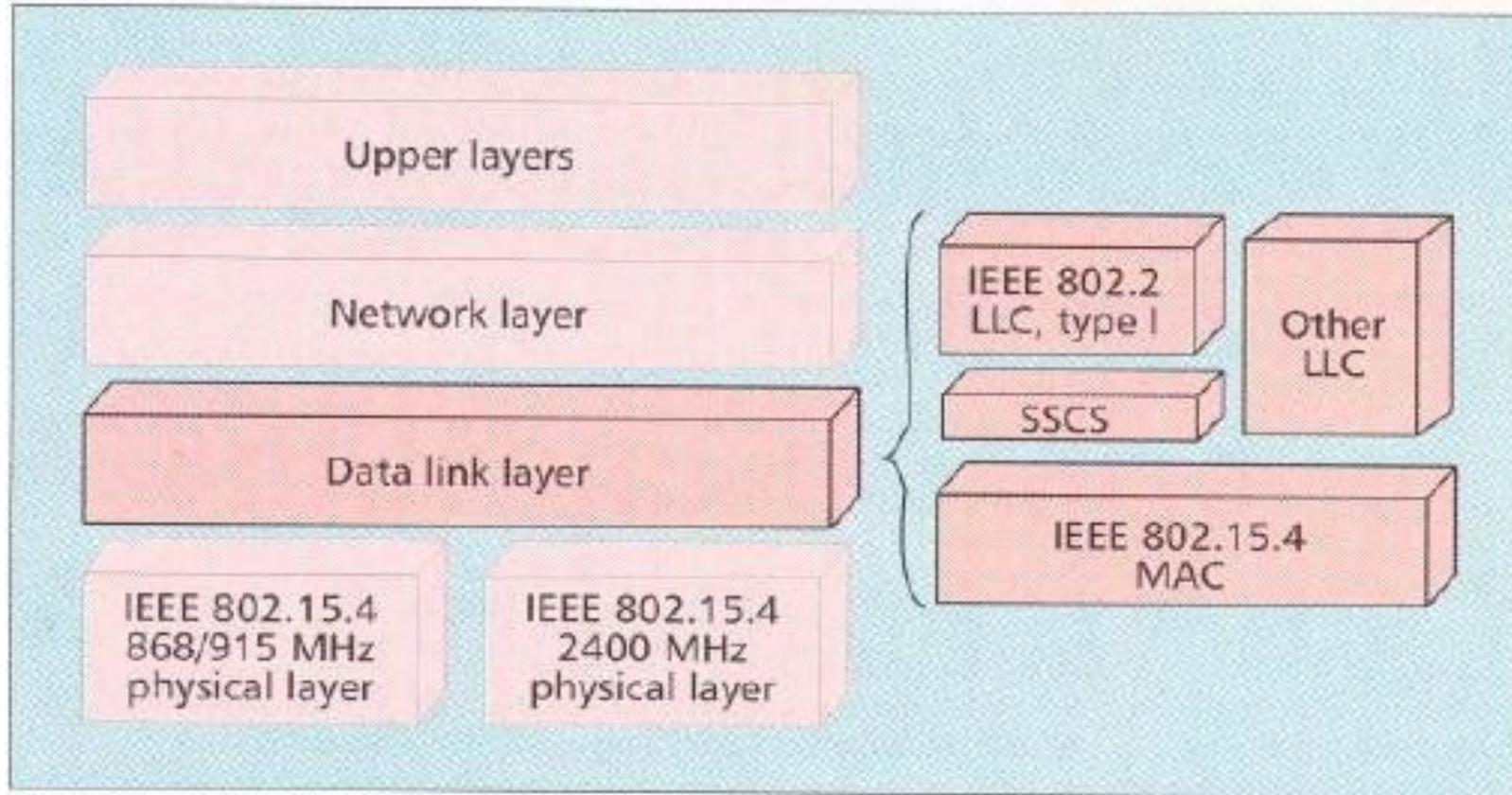
In order to achieve the low power and low cost goals established by IEEE 802.15.4 the following approaches are taken

- Reduce the amount of data transmitted
- Reduce the transceiver duty cycle and frequency of data transmissions
- Reduce the frame overhead
- Reduce complexity
- Reduce range
- Implement strict power management mechanisms (power-down and sleep)

IEEE 802.15.4

- IEEE 802.15.4 deals with only PHY layer and portion of Data link layer.
- The higher-layer protocols are left to industry and the individual applications.
- The Zigbee Alliance is an association of companies involved with building higher-layer standards based on IEEE 802.15.4. This includes network, security, and application protocols.

IEEE 802.15.4



IEEE 802.15.4 draft standard supports multiple network topologies including star and peer to peer topology.

IEEE 802.15.4

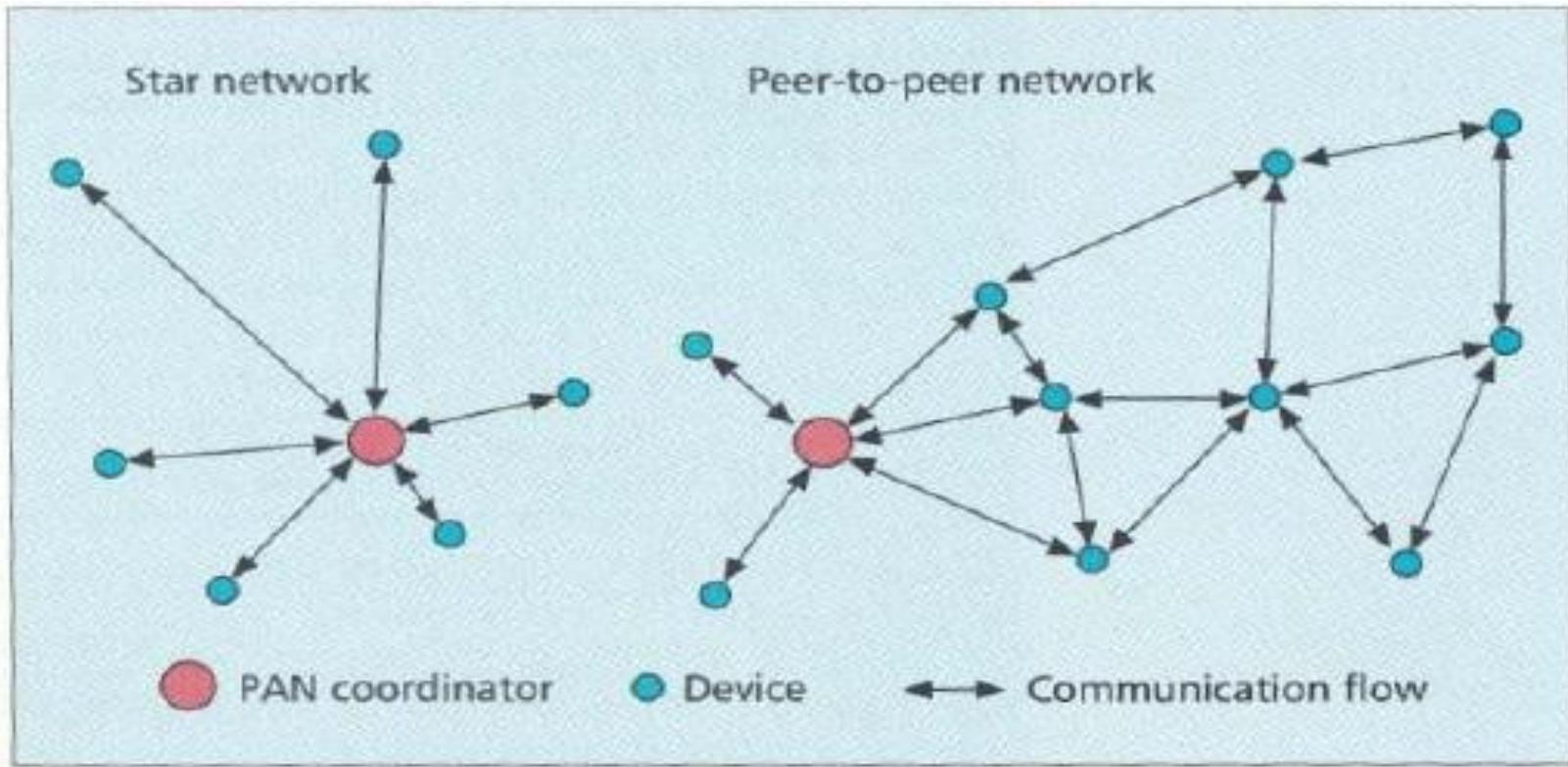


Figure 1. Star and peer-to-peer networks.

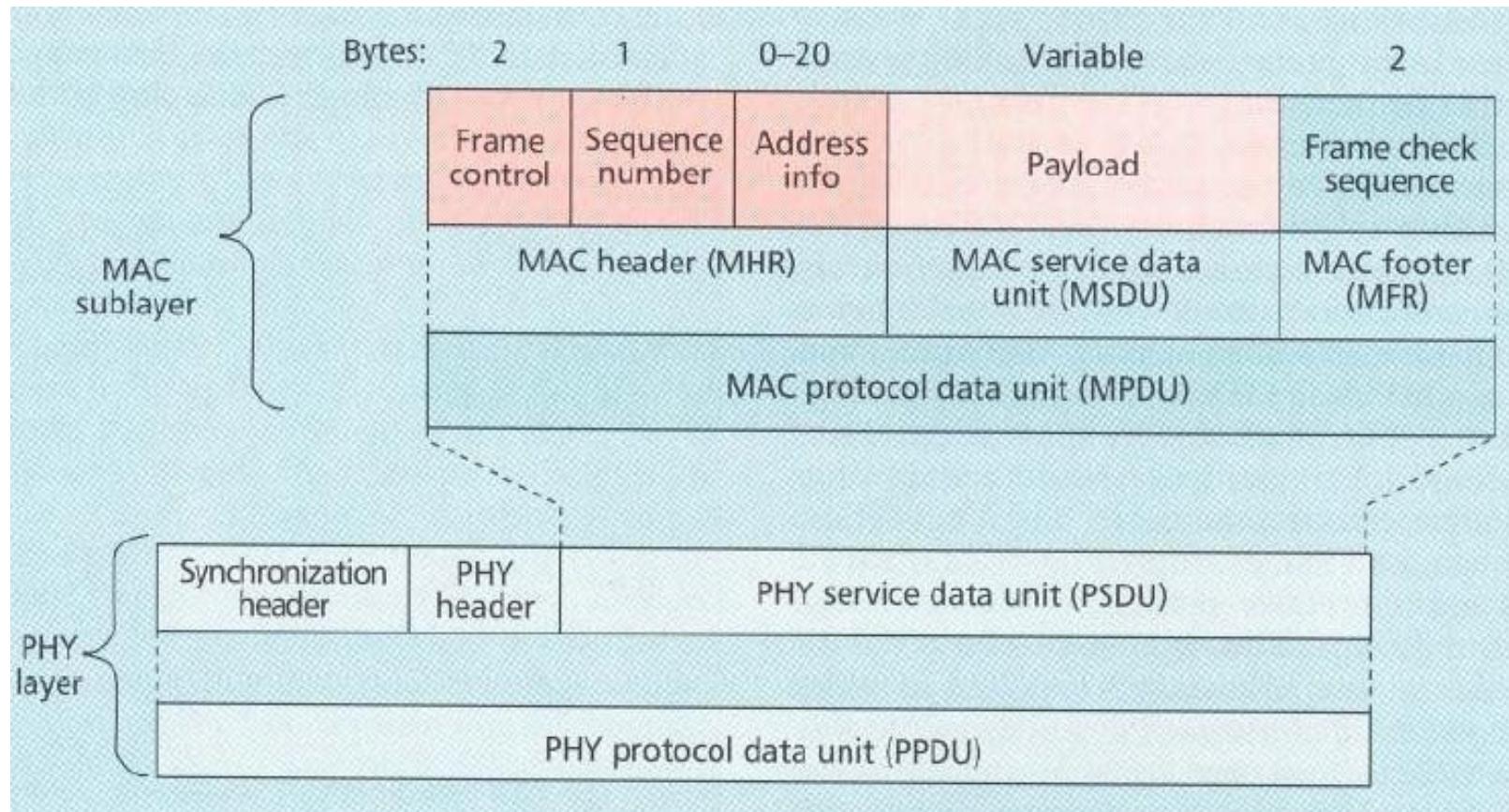
IEEE 802.15.4

- IEEE 802 splits DLL into MAC and LLC sublayers.
- LLC is standardized and is common in 802.3, 802.11, 802.15.1.
- Features of the IEEE 802.15.4 MAC are
 - Association and disassociation
 - acknowledged frame delivery
 - Channel access mechanism
 - Frame validation
 - Guaranteed time slot management

IEEE 802.15.4 ...MAC

- MAC provides data and management services to upper layers
- 802.15.4 MAC is of very low complexity, making it very suitable for its intended low-end applications, albeit at the cost of a smaller feature set than 802.15.1 (e.g., 802.15.4 does not support synchronous voice links).

IEEE 802.15.4 ...MAC



IEEE 802.15.4 ...MAC

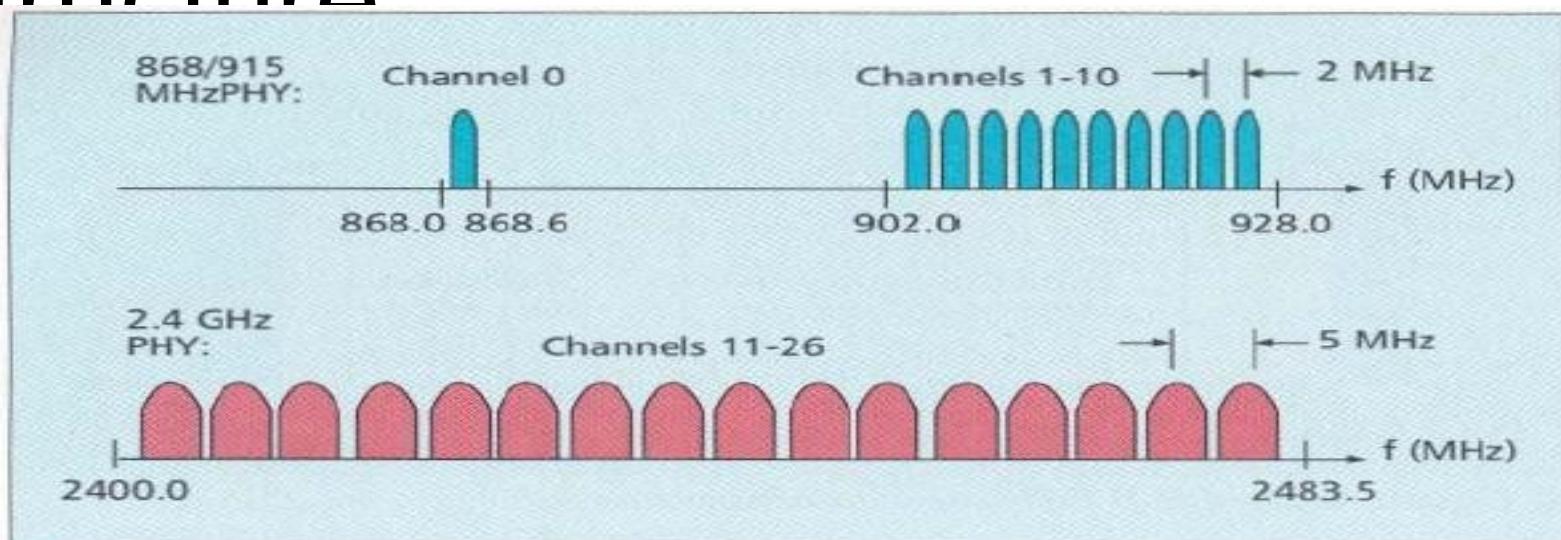
- Frame control field **indicates the type of MAC** frame being transmitted, specifies the format of the address field, and controls the acknowledgment.
- Multiple address types : **64 bit physical address** and short 8 bit network assigned address are provided.
- Address field size may vary **from 0 to 20 bytes**.
- Payload field is variable with condition size of **mac frame <= 127 bytes**.
- FCS is used for integrity check using 16 bit

IEEE 802.15.4 ...PHY

- This standard provides 2 PHY options with frequency band as fundamental difference.
- 2.4 GHz band has worldwide availability and provides a transmission rate of 250 kb/s.
- The 868/915 MHz PHY specifies operation in the 868 MHz band in Europe and 915 MHz ISM band in the United States and offer data rates 20 kb/s and 40 kb/s respectively.

IEEE 802.15.4 ...Channel

Structure



■ Figure 5. The IEEE 802.15.4 channel structure.

Channel number	Channel center frequency (MHz)
$k = 0$	868.3
$k = 1, 2, \dots, 10$	$906 + 2(k - 1)$
$k = 11, 12, \dots, 26$	$2405 + 5(k - 11)$

■ Table 2. IEEE 802.15.4 channel frequencies.

IEEE 802.15.4 ... Modulation

PHY	Frequency band	Data parameters			Spreading parameters	
		Bit rate (kb/s)	Symbol rate (kbaud)	Modulation	Chip rate (Mchips/s)	Modulation
868/915	868.0–868.6 MHz	20	20	BPSK	0.3	BPSK
MHz PHY	902.0–928.0 MHz	40	40	BPSK	0.6	BPSK
2.4 GHz PHY	2.4–2.4835 GHz	250	62.5	16-ary orthogonal	2.0	O-QPSK

IEEE 802.15.4 ...Practical SetUp

https://www.youtube.com/watch?v=y_FV-2hTYE

ModBus

https://www.youtube.com/watch?v=txi2p5_OjKU

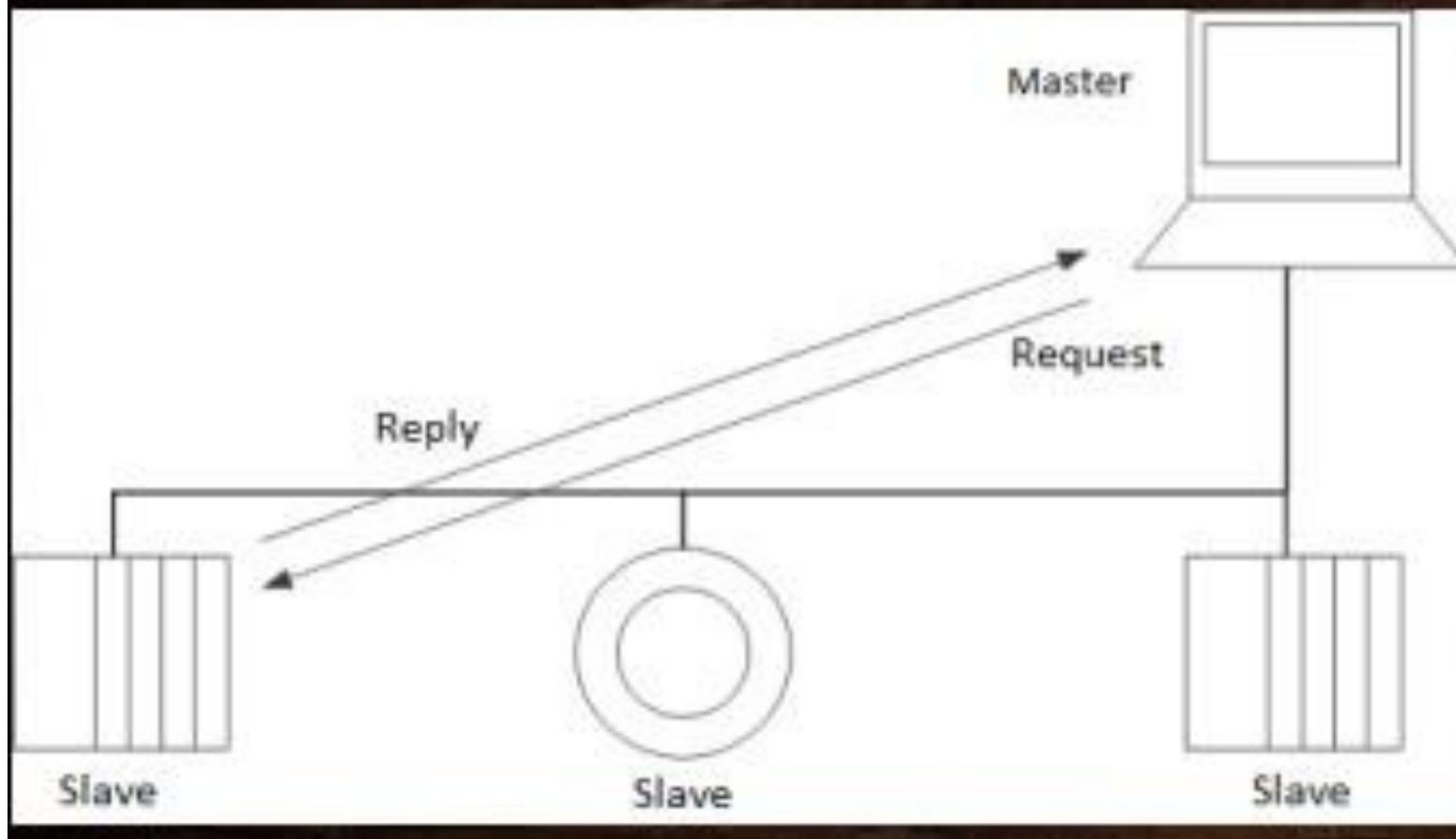
- Modbus is a **serial communications protocol** originally published by Modicon (now Schneider Electric)
- Used to establish **master-slave/client-server** communication between intelligent devices
- Openly published and royalty-free
- Modbus **enables communication between many (approximately 247) devices** connected to the same network

ModBus

- MODBUS devices communicate using a master-slave technique in which **only one device** (the master) **can initiate transactions** (called queries).
- The other devices (slaves) respond by supplying the requested data to the master
- A slave is any peripheral device (I/O transducer, valve, network or other measuring device), which processes information and sends its output to the master .
- Masters can address individual slaves, or can initiate a broadcast message to all slaves.

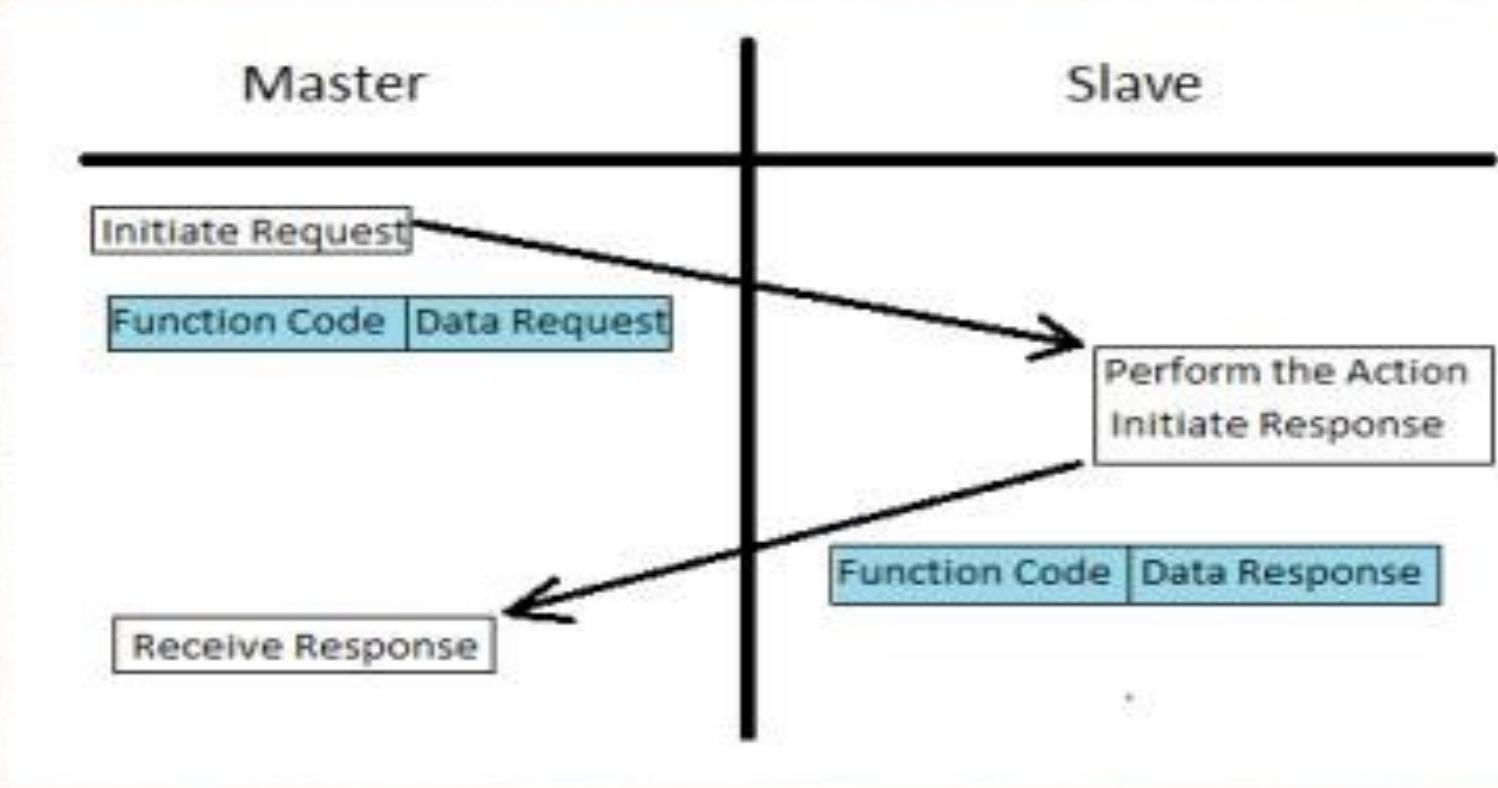
ModBus

Basic Modbus Network



ModBus

Basic Modbus Transaction



ModBus

- MODBUS Frames :
 - ADU ...Application Data Unit
 - PDU Protocol Data Unit

ModBus

- MODBUS Frames :
 - ADU ...Application Data Unit
 - PDU Protocol Data Unit
- The PDU frames : function Code+ data.
- The ADU frames : Add+FC+data+Error check .
- The FC -> action to perform and the data -> information to be used for this action.

ModBus Data TTypes

- Modbus transactions always perform a set of actions by reading or writing to a set of four data ,used by the Modbus application layer.

Primary Tables	Object Type	Type of
Discrete Input	Single bit	Read-Only
Coils	Single bit	Read-Write
Input Registers	16-bit word	Read-Only
Holding Registers	16-bit word	Read-Write

ModBus Accessing Data

- 16-bit Unsigned Registers And Single-bit Coils
 - Input Registers And Holding Registers
 - Input Coils And Status Coils
- 64 kb of space is allocated for registers and coils

Function Code	
1	Read Coil Status
2	Read Input Status
3	Read Holding Registers
4	Read Input Registers
5	Write Single Coil Status
6	Write Single Register
15	Multiple Coil Write
16	Multiple Register Write

ModBus Transmission modes

- ASCII ... Uses Longitude Redundancy Check
- Remote Terminal Unit ... Uses Cyclic Redundancy Check
 - In Modbus RTU, bytes are sent consecutively with a 3-1/2 character space between messages for a delimiter. This allows the software to know when a new message is starting.
 - Any delay between bytes will cause Modbus RTU to interpret it as the start of a new message.
 - Modbus ASCII marks the start of each message with a colon character ":" (hex 3A).
 - The end of each message is terminated with the carriage return and line feed characters (hex 0D and 0A)

ModBus Transmission modes

ASCII Mode-

When controllers are setup to communicate on a Modbus network using **ASCII (American Standard Code for Information Interchange)** mode, each 8-bit byte in a message is sent as two ASCII characters. The main advantage of this mode is that it allows time intervals of up to one second to occur between characters without causing an error.

The format for each byte in ASCII mode is:

Coding System: Hexadecimal, ASCII characters 0-9, A-F. One hexadecimal character contained in each ASCII character of the message.

Bits per Byte: 1 start bit

7 data bits, least significant bit sent first

1 bit for even/odd parity; no bit for no parity

1 stops bit if parity is used; 2 bits if no parity

Error Check Field: Longitudinal Redundancy Check (LRC)

ModBus Transmission modes

RTU Mode-

When controllers are setup to communicate on a Modbus network using **RTU (Remote Terminal Unit)** mode, each 8-bit byte in a message contains two 4-bit hexadecimal characters. The main advantage of this mode is that its greater character density allows better data throughput than ASCII for the same baud rate. Each message must be transmitted in a continuous stream.

The format for each byte in RTU mode is:

Coding System: 8-bit binary, hexadecimal 0-9, A-F. Two hexadecimal characters contained in each 8-bit field of the message.

Bits per Byte: 1 start bit

8 data bits, least significant bit sent first

1 bit for even/odd parity; no bit for no parity

1 stops bit if parity is used; 2 bits if no parity

Error Check Field: Cyclical Redundancy Check (CRC)

BACNet Protocol

<https://www.youtube.com/watch?v=oevGXrkxEos>

BACNet

- **Building Automation and Control Networks** developed by American Society of heating, Refrigerating and Air-Conditioning Engineers(ASHRAE)
- It is a data communication protocol designed for communication between building automated system components
- **This is an Object Oriented protocol**
- **Objects** : Physical device, temperature input(analog input) , A relay control(binary output), schedules
- **Services** : Used to perform read, write and I/O

BACNet

- **BacNet Objects** are evaluated and controlled by their properties
- Property Name, Value

Object Name	“Lighting Area”
Object Type	BINARY_OUTPUT
Present Value	Active
Status_Flags	Normal, In-Service
Out_Of_Service	False
Inactive_Text	“Off”
Active_Text	“On”

BACNet Services

<https://www.youtube.com/watch?v=a3TuEypguy8&t=22s>

- **BacNet** Services are formal requests that one BACNet device sends to another to ask it to do something
- Categories :
 - **Object Access** (Read, Write, Create, Delete)
 - **Alarm and Event** (Alarms and Changes of State)
 - **File Access** (Trend data, Data transfer)
 - **Remote Device Management** (Discover, Time Synchronization, Backup and Restore Database, Initialization, Who Is, I Am , Who-Has, I-Have)
 - **Virtual Terminal** (HMI via menus)
- **These follow a Client-Server model**

BACNet Protocol Stack

BACNET Application Layer				Application (7)
BACNET Network Layer				Network (3)
ISO 8802-2 (IEEE 802.2) Type 1	MS/TP	PTP	LonTalk	Data Link (2)
ISO 8802- 3 (IEEE 802.3) Ethernet	ATA/ANSI 878.1 ARCNET	EIA - 485	EIA - 232	Physical (1)

BACnet Protocol Stack

OSI Stack

BACNet Network Types i.e Physical and Data Link Layer

<https://www.youtube.com/watch?v=cM4KVD1119o&t=219s>

BACNet IP

- Used with existing ethernet, WAN

BACNet MS/TP

- Uses Twisted Pair EIA -485 upto 4000 feet

BACNet ISO 8802-3

- Limited to Single Infrastructure without IP routers

BACNet P2P

- Used only for dial-up telephone networks

KNx Protocol

KNX

- Abbreviation for KONNEX evolved from EHS (European Home Systems Protocol), EIB (European Installation Bus), BatiBUS
- Used for Building Automation
- **Operates on more than one physical layer**
e.g **twisted pair wiring, Ethernet, infrared**
- Every Unit hooked up to the KNX system is smart enough and does not rely on other parts to function
- **KNX devices are sensors, actuators, system devices.**

KNX

- KNX Devices have 3 modes :
 - A-mode(automatic)....Configure themselves
 - E-mode (easy)...Require training to install
 - S-mode(system mode)....must be programmed by specialists.
- KNX network can be formed with tree, line and star topologies
- This can Link upto 57,375 devices

KNX

- For Routing of messages KNX uses telegrams

Control 8 Bits	Source Address 16 Bits	Dest Address 17 bits	Routing Counter 3 Bits	Length 3 Bit	Data Upto 16 bytes	Parity 8 Bits
---------------------------	---------------------------------------	-------------------------------------	---------------------------------------	-------------------------	-----------------------------------	--------------------------

KNX Telegram

**Control – Decides Priority
or For Acknowledgment**

- High, Low, Alarm, System
- ACK, NAK, BUSY

**Source
Address**

- 4 bits- Area ID, 4 Bits- Line ID, 4 Bits- Device ID

Dest Addr

- Can be Physical or Logical – 17th Bit indicates PHY - 0 or LOG-1

**Routing
Counter**

- Defines Hop Count ...Limited to 6 Hops

Parity

- Used to Secure the Telegram

KNX

- Used for control of building management
 - Lighting
 - Blinds/Shutters
 - HVAC
 - Metering
 - Remote Control
 - Refrigerators, Washers, Dryers
 - Security Systems

KNX

- Advantages
 - Platform Independent
 - Low energy consumption
 - Open Standard

ZigBee Protocol

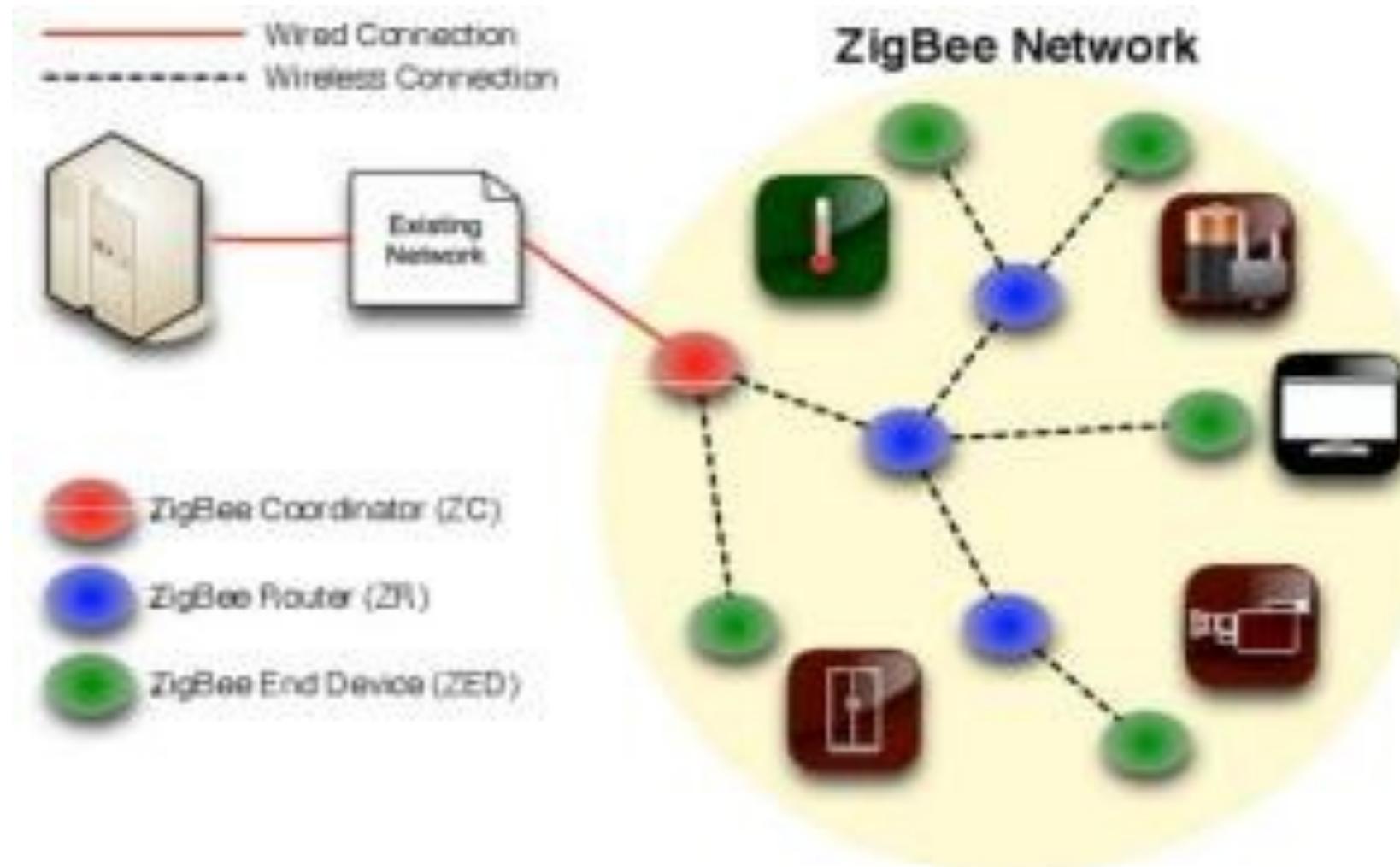
ZigBee Technology

- Built on IEEE 802.15.4 standard for Wide Personal Area Network(WPAN)
- This defines PHY and MAC layers to handle many devices at low-data rates.
- These operate at 868 MHz(Europe with 20Kbps), 902-928 MHz(US with 40 Kbps) and 2.4 GHz(Entire world with 250 Kbps)
- Low-cost and low-powered mesh network, which covers range of 10-100m.
- These can be extended with the help of routers

ZigBee Architecture

- This consists of Three Devices in the n/w layer
 - Zigbee coordinator
 - Router
 - End Device
- Zigbee coordinator - Every Network in Zigbee must have a coordinator(root) that handles and stores the information and also transmit and receive data operations.
- Router – These are intermediary devices that permit data to pass to and fro through them to other devices.
- End Devices – Have limited functionality to communicate with parent nodes

ZigBee Architecture



ZigBee Architecture

- This consists of various layers out of which PHY and MAC Layer are defined by 802.15.4
- Zigbee has own Network Layer and Application Layer.
 - Physical Layer – Does modulation, demodulation
 - MAC Layer – responsible for reliable transmission of data with CSMA/CA
 - Network – Routing, Network configurations, connections and disconnection management
 - Application support sub-layer : Interfaces with data managing services
 - Application framework : Provides two types of messaging service General messaging(GMS), Key-value pair(K_V pair).

ZigBee OPERATING Modes

- Non- Beacon
 - The coordinators and routers continuously monitor active state of incoming data hence more power is consumed.
- Beacon
 - When there is no data communication from end devices, the routers and coordinators enter into sleep state.
 - The coordinator wakes up periodically and transmits the beacons to routers

ZigBee Topologies

- Star...Here there is one coordinator responsible for initiating and managing devices.
- Mesh.... Several routers are connected
- Cluster- Tree

ZigBee N/W Frame Format

Octets : 2	2	2	1	1	Variable
Frame Control	Dest Add r	Sour ce Addr	Radius	Seq No	Frame Payload
Routing Fields					
Network Header					Network Payload

ZigBee N/W Frame

- **Frame Control** – Define various parameters like
 - Communication type – Unicast/MutiCast/Broadcast
 - Security – Enabled/Disabled
 - Route discovery – Enabled/Disabled
 - Source / Destination Addr Specified or not
- **Addr** : 64-bit / 16-bit
- **Radius** – defines maximum number of hops allowed for packet
- **Seq. No** – Packet Counter

ZigBee APS Frame Format

Octets : 1	0/1	0/1	0/2	0/1	Variable
Frame Control	Dest End Point	Cluster Identifier	Profile Identifier	Source Endpoint	Frame Payload
Addressing Fields					
APS Header					Network Payload

ZigBee APS Frame

- **Frame Control** – Define various parameters like :
 - **Frame type-** Data/Command/ACK
 - Security – Enabled/Disabled
 - Delivery mode – Unicast/Broadcast/Multicast
 - Source / Destination Addr Specified or not
- **Addr** : 8-bit source / dest addr
- **Cluster Identifier** – Used to identify the cluster that is used in binding operation of zigbee coordinator. This is present for data frames but not for command frames.
- **Profile Identifier** – Specifies the profile for which frame is intended. **Used only for Data and ACK frame.**

ZigBee Applications

- Home Automation
- Smart Metering
- Smart Grid Monitoring
- Industrial Automation

IoT Device Life Cycle

Boot Up

- Device is loading the firmware and starts to work as defined

Initialization

- Establish connection, Sync Data, Read configuration

Operation

- Device performs its designed task continuously

Update

- New Firmware arrives, device reboots, and loads new firmware

IoT Device Life Cycle

....BootUp

- **Firmware integrity check:** To ensure that firmware has not been modified or tampered by others, the best method is to implement an integrity check by embedded checksum or **secure password**.
- **Secure boot:** Encrypt firmware with PKI or **public/private certification** to secure the whole boot-up process.

IoT Device Life Cycle

....Initialization

• AAA protection

- Use **proper encryption to avoid user/device hijack.**
- Default account credentials appear in many IoT devices.
It's best to have an activation process which **requires end-user to change default password.**
- **Key/Certification protection:**
 - Use a **KMS (Key Management System) or CMS (Certification Management System)** to protect **encryption/decryption keys**, or store those keys in a **TPM (Trusted Platform Module).**

IoT Device Life Cycle

Initialization: Communication protection: The communication between device and device, device and the Internet, or device and user interface (through mobile apps or web apps) should be encrypted (HTTPs, AES 128, 256, and others).

• **Identity protection:** To prevent a fake identity within the communication group, it is necessary to make sure the **communicated object is certified**. A KMS or CMS can also play an important role here.

IoT Device Life Cycle

Operation

- AAA protection

- Remove all backdoor debug user accounts. From several studies, we have found out that many IoT devices keep those accounts for debugging purposes in the system and that increases the chances of penetration.
- During the operation stage, the IoT device may still associate with new devices, users, and clouds, for example; add new monitor sensors for a connected home, or **creating additional user account for**

IoT Device Life Cycle

Operation

- **Monitoring:**

- The device should implement knowledge to detect abnormal operations and, if such operations occur, provide a warning to the backend or end user.

- **Integrity check:** Run-time integrity checks can prevent the device from being compromised during operation.

Leveraging cloud technology to have two-way integrity checks will be the most effective way.

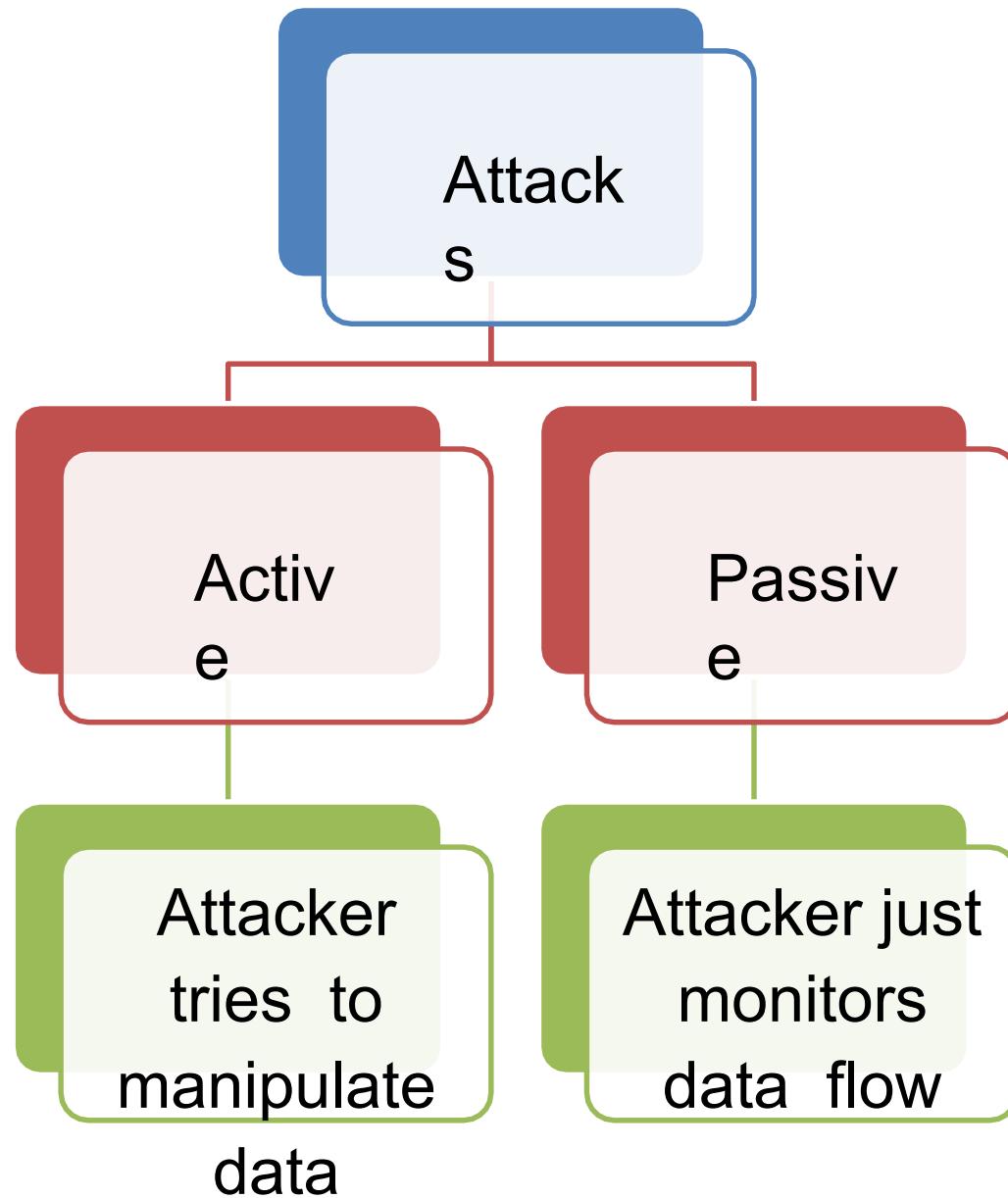
IoT Device Life CycleOperation

- **Risk management:** Use a method like virtual patching or a host IPS to reduce risk before Firmware Over-the-Air (FOTA) triggers

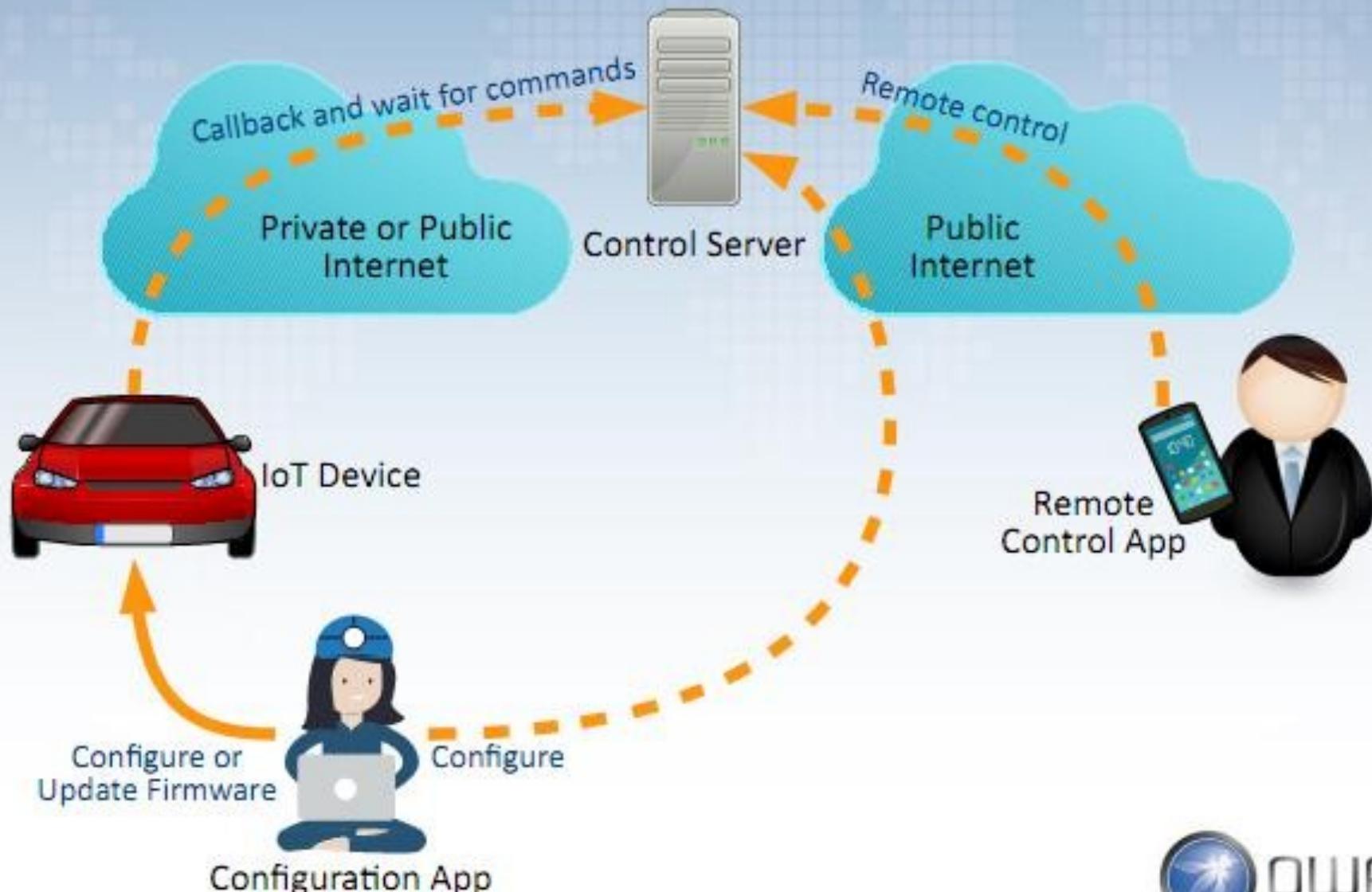
IoT Device Life CycleUpdation

- Secure FOTA (Firmware Over-the-Air):**

Before the FOTA trigger, the new firmware needs to be encrypted and checked to make sure the next lifecycle will be performing a secure boot up again.



Typical IoT Infrastructure



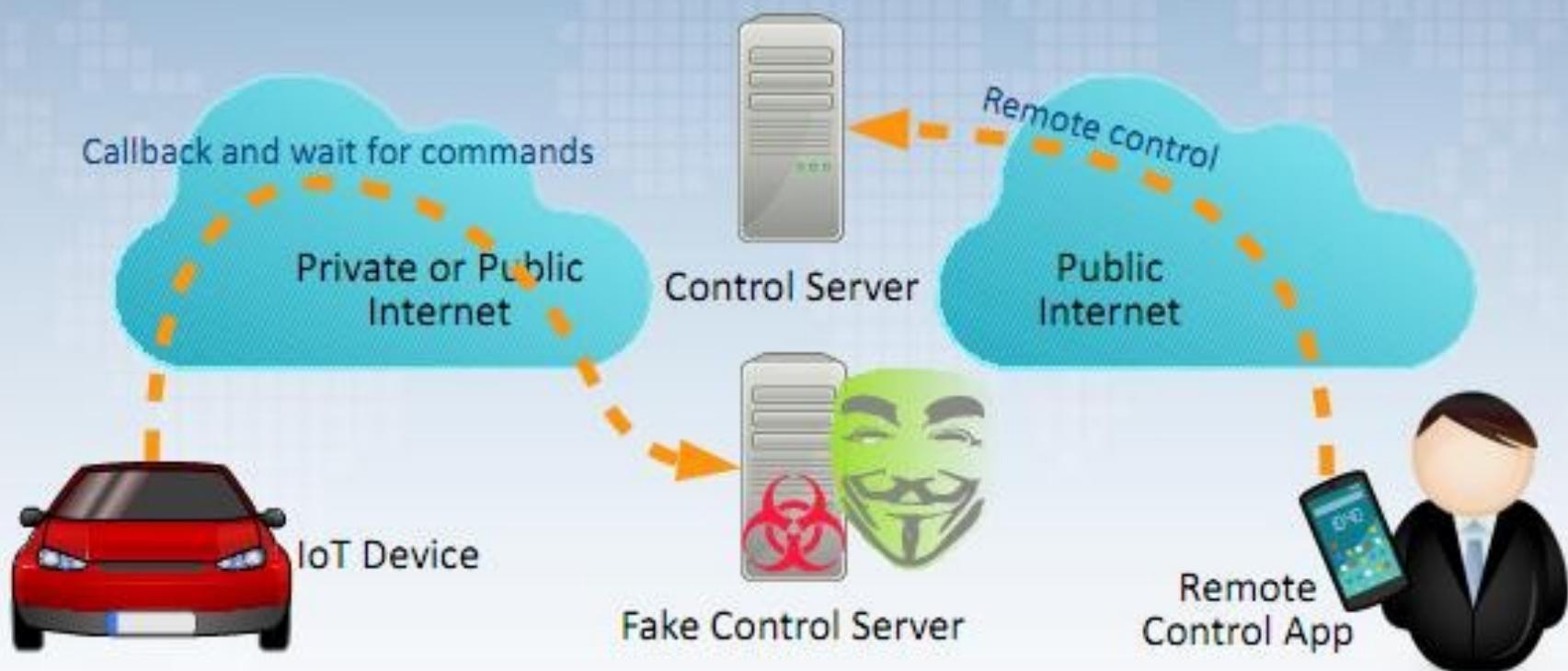
Problems of IoT Security

- Initial design was for private communication network then **moved to IP network** and later on the Internet
- **Firmware updates are hard or nearly impossible** after installations
- Started with basic security then found the security flaws and attached more complex security requirements later
- **Low security devices** from early design are still out there and used in compatible fall-back mode

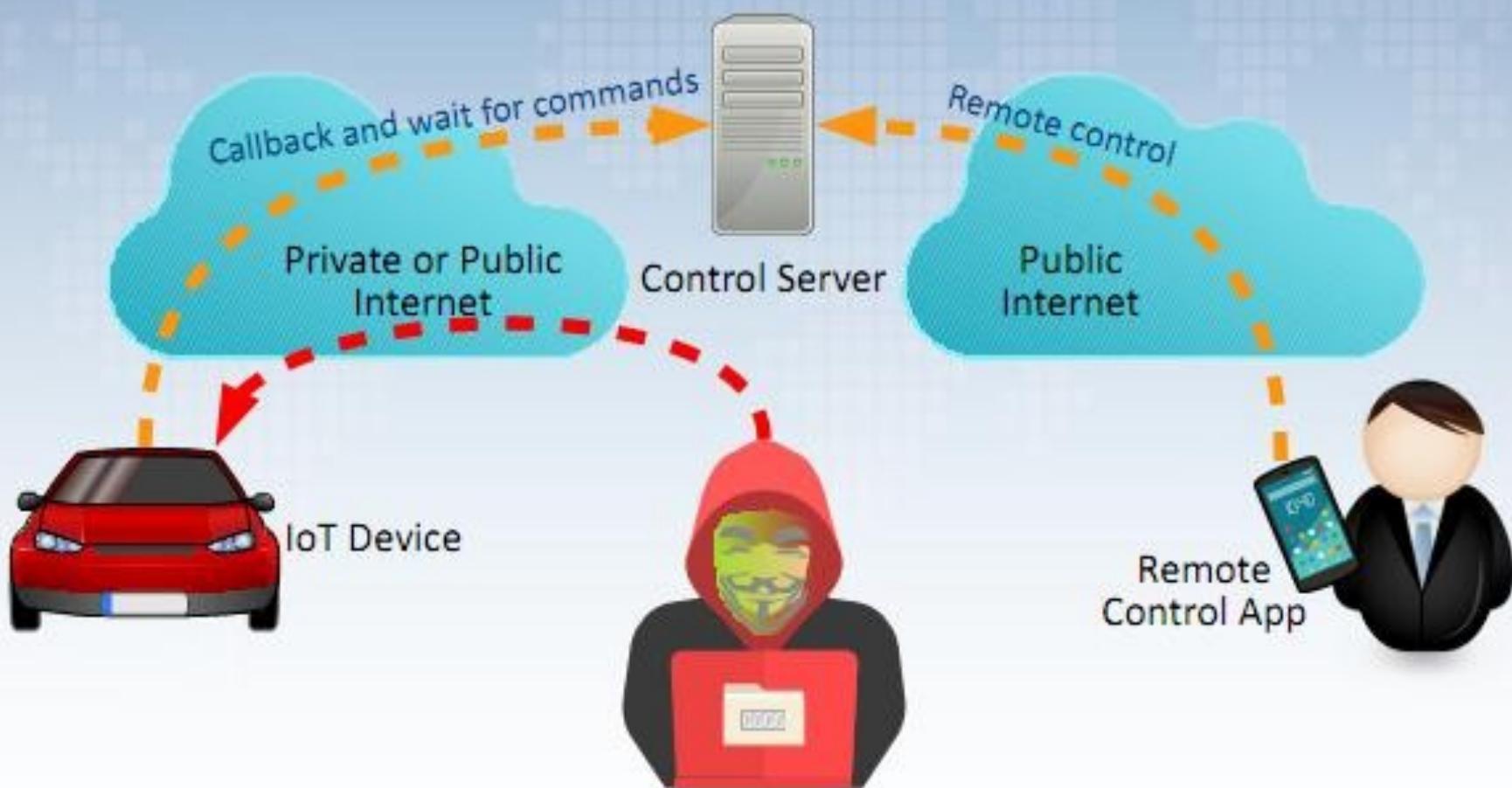
Problems of IoT Security

- Fake Control Server
- Attack on Device Ports
- Attack on Server Ports
- Steal Credential
- Inject Bad Configuration
- Sniff Data on Private Network

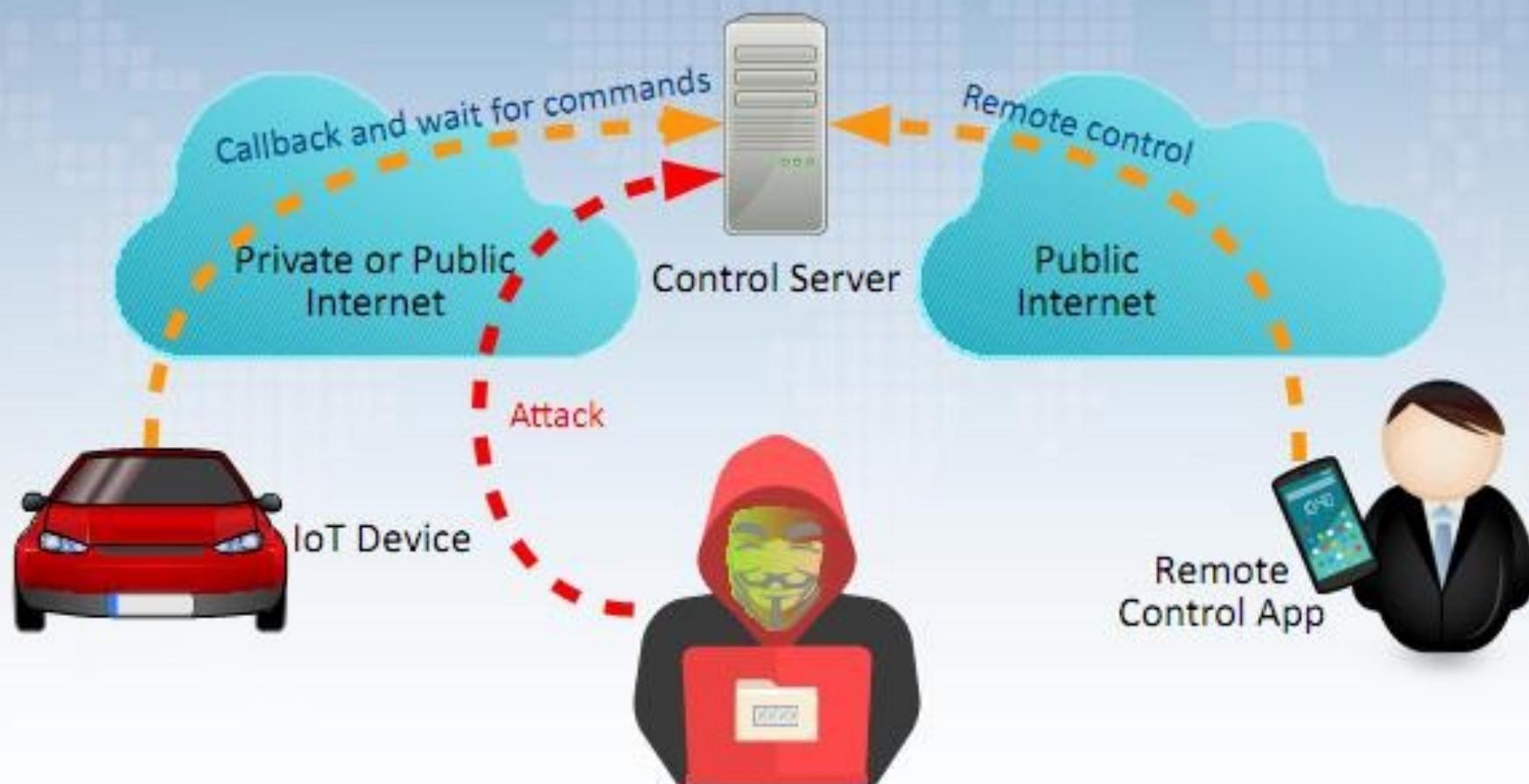
Typical Attack: Fake Control Server



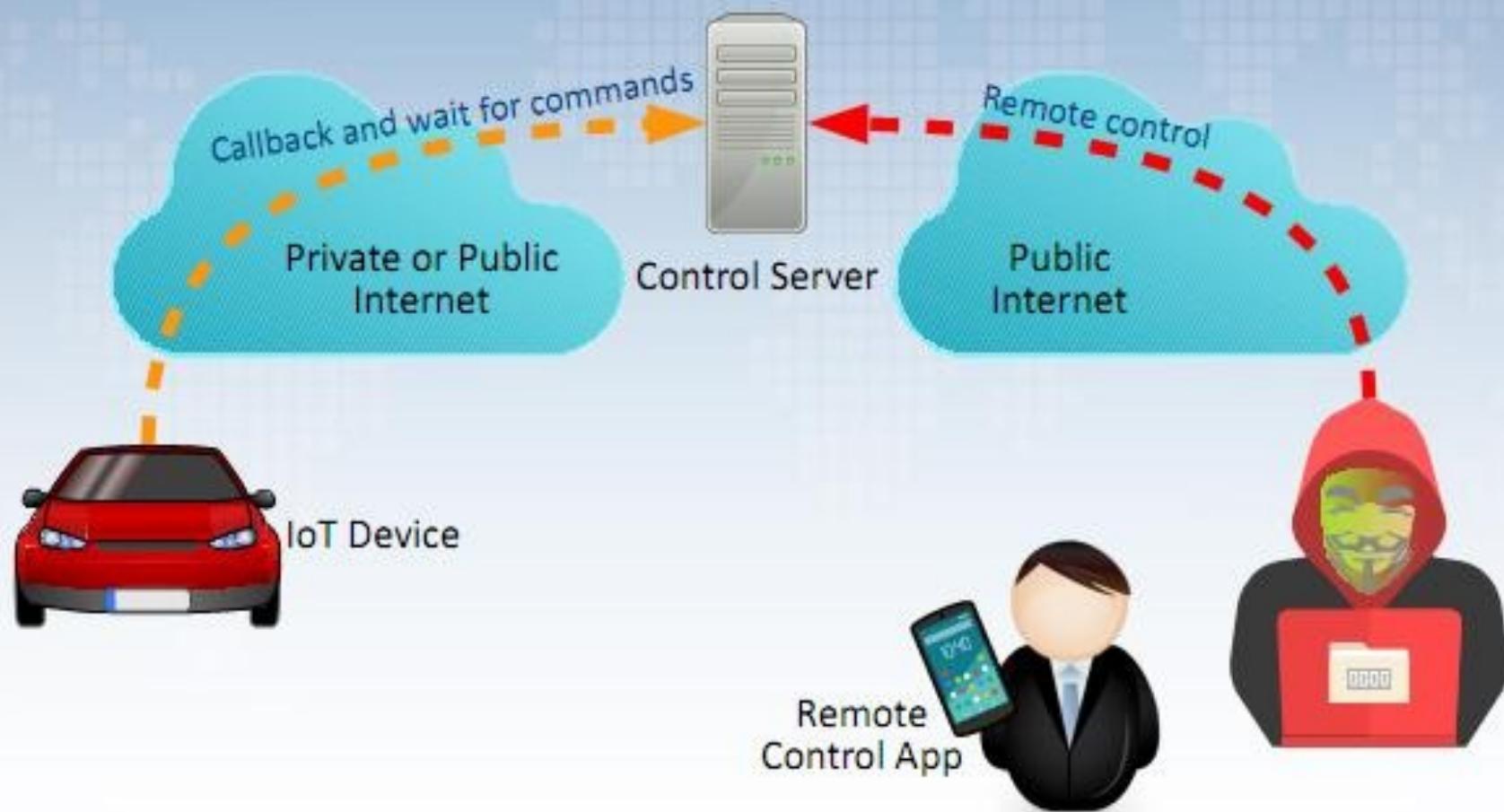
Typical Attack: Attack on Device Open Ports



Typical Attack: Attack on Server Open Ports



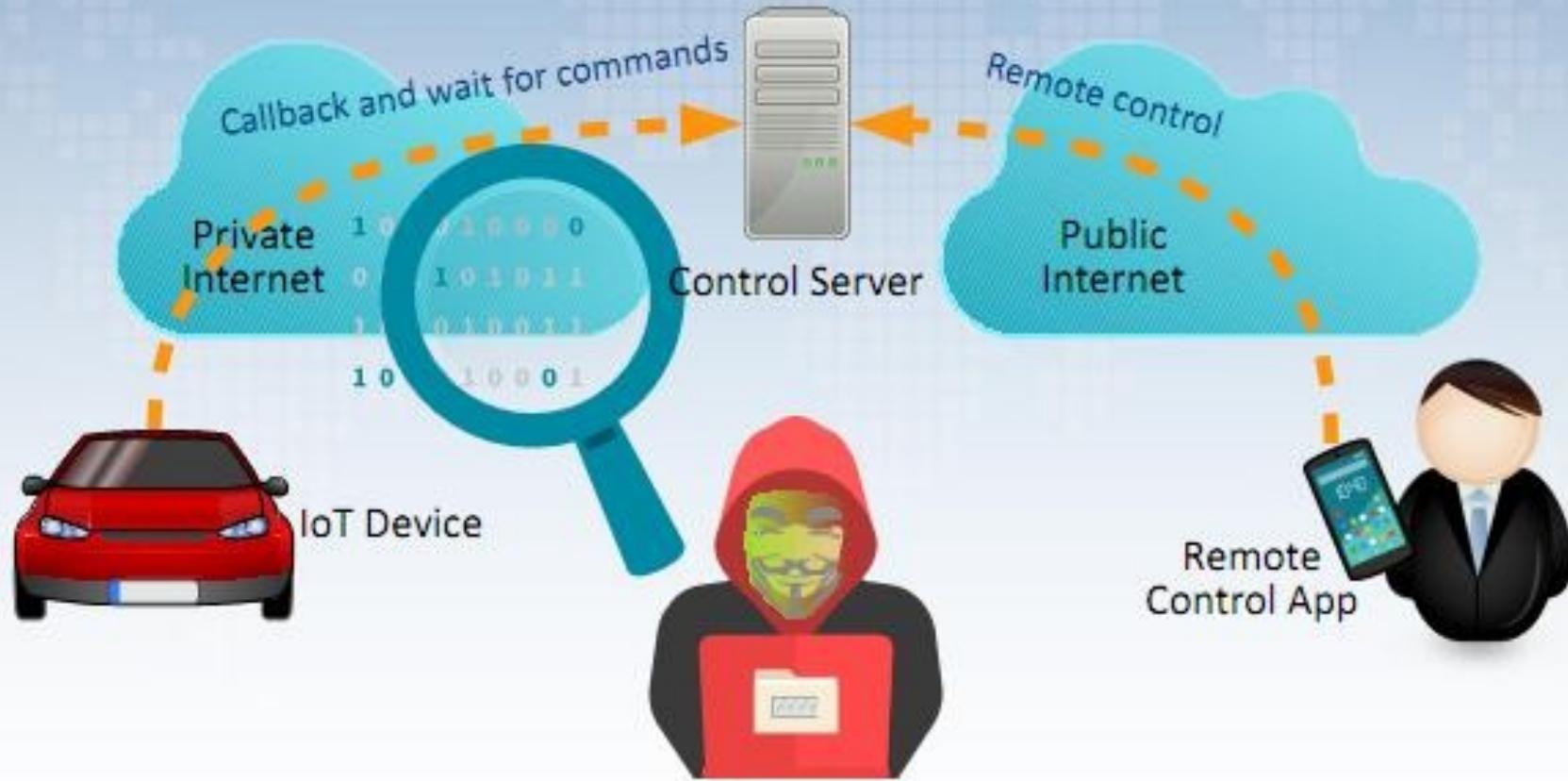
Typical Attack: Steal Credential



Typical Attack: Inject Bad Configuration or Firmware



Typical Attack: Sniff Data on Private Network



IoT Vulnerabilities

- I1 Insecure Web Interface
- I2 Insufficient Authentication/Authorization
- I3 Insecure Network Services
- I4 Lack of Transport Encryption/Integrity Verification
- I5 Privacy Concerns
- I6 Insecure Cloud Interface
- I7 Insecure Mobile Interface
- I8 Insufficient Security Configurability
- I9 Insecure Software/Firmware
- I10 Poor Physical Security

**WANT PC
MS**

IoT Vulnerabilities

1

Insecure Web Interface

covers IoT device administrative interfaces

Obstacles



Default usernames
and passwords



No account lockout

XSS, CSRF, SQLi
vulnerabilities

Solutions



Allow default usernames
and password to be changed



Enable account lockout



Conduct web application
assessments



Insecured Web Interface

- XSS - Cross- Site Scripting
- CSRF- Cross- Site Request Forgery
- SQLi – SQL Injection

IoT Vulnerabilities in IoT

Insufficient Authentication/Authorization

covers all device interfaces and services

2



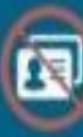
Obstacles



Weak passwords



Password recovery mechanisms
are insecure



No two-factor authentication
available

Solutions



Require strong, complex
passwords



Verify that password recovery
mechanisms are secure



Implement two-factor
authentication where possible

Two-Factor Authentication

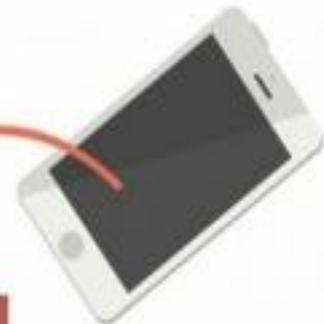
- It is a method of confirming users' claimed identities by using a combination of *two* different factors:
 - 1) something they know,
 - 2) something they have,
 - 3) something they are.
- A good example of two-factor authentication is the withdrawing of money from an ATM; only the correct combination of a bank card (something the user possesses) and a PIN (something the user knows) allows the transaction to be carried out.

IoT Vulnerabilities in IoT

3

Insecure Network Services

covers all network services including device, cloud, web and mobile



Obstacles



Unnecessary ports are open



Ports exposed to the internet via UPnP



Network services vulnerable to denial of service

Solutions

Minimize open network ports



Do not utilize UPnP



Review network services for vulnerabilities



Network Services

Vulnerability

- **Malware**
- **Social engineering attacks** that fool users into giving up personal information such as a username or password.
- **Outdated or unpatched software** that exposes the systems running the application and potentially the entire network.
- **Misconfigured firewalls / operating systems** that allow or have default policies enabled.

IoT Vulnerabilities in IoT

Obstacles

Sensitive information is passed in clear text

SSL/TLS is not available or not properly configured

Proprietary encryption protocols are used

Solutions

Encrypt communication between system components

Maintain SSL/TLS implementations

Do not use proprietary encryption solutions

Lack of Transport Encryption
covers all network services including
device, cloud, web and mobile

4



Transport Layer Vulnerability

- SSL (Secure Socket Layer) not available
- TLS (Transport Layer Security not available)
- Use Proprietary Encryption Protocols (Not Standard (like Kerberos, AES,DES, RSA))

IoT Vulnerabilities in IoT

5

Privacy Concerns

covers all components of IoT solution



Obstacles

- Too much personal information is collected
- Collected information is not properly protected
- End user is not given a choice to allow collection of certain types of data

Solutions

- Minimize data collection
- Anonymize collected data
- Give end users the ability to decide what data is collected

IoT Vulnerabilities in IoT



Insecure Cloud Interface
covers cloud APIs or cloud-based web interfaces

6

Obstacles

Interfaces are not reviewed for security vulnerabilities

Weak passwords are present

No two-factor authentication is present

Solutions



Security assessments of all cloud interfaces



Implement two-factor authentication



Require strong, complex passwords

IoT Vulnerabilities in IoT

7

Insecure Mobile Interface

covers mobile application interfaces



Weak passwords
are present

Obstacles



No two-factor authentication
implemented



No account lockout
mechanism



Implement account
lockout after failed
login attempts



Implement two-factor
authentication



Require strong,
complex passwords

Solutions

IoT Vulnerabilities in IoT

Insufficient Security Configurability covers the IoT device

8

Obstacles

Password security options are not available

Encryption options are not available

No option to enable security logging



Solutions

Make security logging available

Allow the selection of encryption options

Notify end users in regards to security alerts

IoT Vulnerabilities in IoT

9

Insecure Software/Firmware covers the IoT Device



Obstacles

- Update servers are not secured
- Device updates transmitted without encryption
- Device updates not signed

Solutions

- Sign updates
- Verify updates before install
- Secure update servers

IoT Vulnerabilities in IoT

Poor Physical Security covers the IoT device

10

Obstacles

Unnecessary external ports like USB ports

Access to operating systems through remove media

Inability to limit administrative capabilities

Solutions

Minimize external ports like USB ports

Properly protect operating system

Include ability to limit administrative capabilities



Security Challenges in IoT

- Wireless communication
- Physical insecurity
- Constrained devices Potentially sensitive data
- Lack of standards
- Heterogeneity: weakest link problem
- A systems, not software problem
- Classic web / internet threats
- Identity management & dynamism
- Inconvenience and cost

Attacks in Different Layers of IoT

Transport

Send
Wrong
Data

Incorrect
Control
Packets

Network

Routing
Loop

WormHol
e

Network
Partitionin

Denial of
Service

MAC

Spoofin
g

Man In
the
Middle

Eaves
Dropping

How to Secure IoT Devices

Download

...<https://www.youtube.com/watch?v=2Zc2PDXTjsI>

1. Secure Boot
2. Authentication
3. Protected Ports ..Physical Security
4. Secure Storage
5. Secure Connections

Key Elements in IoT Security

Identity Establishment

- Use Public Key Cryptography

Access Control

- Define boundary of data access for devices...Data Access is done by Authentication

Data and Message Security

- Use Data Encryption Standards

Non-Repudiation and Availability

- Rejecting the fact that one entity has sent or received the message, and make all resources available and updated

Security Model

- Represent Security Features to be followed by an IoT Application

Security Model for IoT

**Security ..Authentication,
Confidentiality, Integrity,**

Availability

Trust ...Repudiation

Privacy..Users Privacy, Laws,

Use Case and Misuse Cases in IoT Security

- Use Cases : Normal Behavior
- Misuse Cases : Malicious Behavior

Home Automation System

Use Case : Granting access to the owner of house via valid credentials

Misuse Case : Unauthorized Access

Misuse Cases in IoT Security

Manipulation of Credentials

- Changing the username and password

Unauthorized Data Transmission

- Leads to Expose Information

Denial of Service Attack

- Flooding of a particular request may deny permission to authorized user

Man in the middle Attack

- A Person may act to a system as if he is a owner or part of the system.

Q.1 What are different challenges of IOT?

>>

The Internet of Things (IoT) has affected the world in every field of life. It has created a lot of avenues for the people. The companies that have not adopted IoT in their business are working on it and will start using it in the future for their business growth. Despite all the growth and popularity, IoT is facing challenges to grow more.

Even with the immense usage and growth of IoT, it is still facing issues that need to be resolved. These issues are as follows:

1. Technological issues
2. Challenges in business
3. Society challenges

1.Tecnological Issues:

- Connectivity:

Nowadays IoT devices work based on client-server architecture, where every device is connected to a centralized cloud server. All the activities are taken place with this architecture.

Currently, there are only thousands of devices that are connected to a server, so connectivity is enough for these devices. But as you know that IoT is growing day by day. And it has been predicted that in coming years there will be billions of devices that will connect to the Internet.

So, here comes the connectivity challenge in the picture. IoT has no such system that can connect the massive number of devices on the Internet. So, there is a need to introduce a decentralized architecture to lessen the load from the servers. Blockchain technology can be a solution to the connectivity issue.

- Security:

IoT is facing serious security issues that prove fatal and alarming for the public and administration who have adopted the technology. The hackers have got access to the smart devices and are changing their instructions.

The future of IoT is in danger due to this issue. This is the big challenge that IoT companies need to overcome to get a bright future and to grow the technology more and more.

- Standards:

The companies are not planning with an organized approach. They do not agree upon a specific standard that makes the devices unreliable. The IoT companies fail to adopt a standard that is why they lack planning, executing, and maintaining IoT systems.

- Lack of intelligent analysis:

The algorithms are not producing accurate results. Sometimes it might be possible that the data given to the sensor is not accurate. So, if the data is not accurate, the results will be wrong. The result is an inaccurate analysis. So, it is a big challenge for IoT to get the proper analytical tools to overcome these inaccuracies.

2. Challenges in Business

Nowadays more people are coming to invest and start their business in IoT. This is because it is trending and changing the lives of the people. It also has a bright future so that it is a hot field for the people.

Businesses must satisfy all the rules and standards that they need. But unfortunately, they are not working well currently. So, the challenges must be addressed to resolve the issues.

- Industrial IoT:

It covers all the devices or sensors used in industries like electric meters, pipeline monitors, robots, and many other industrial devices that are connected to the Internet.

- Commercial IoT:

This category involves all the devices of the medical industry, tracking devices, and inventory controls.

- IoT for consumers:

It covers all the devices used by the consumers like mobile phones, smartwatches, home appliances that are connected like laptops, and different types of information systems.

3.Society Challenges

IoT is facing many issues regarding society that are proving to be a big challenge for it. The companies are working on them, but they still have the following issues:

- The customers change their demands frequently so IoT has to fulfill them
- The new devices grow gradually, so time is needed
- Inventing new things and integrating the previous ones with the new ones demands effort and money
- The users are growing every day. So IoT has to tackle all of them.
- These issues can disappoint the users and they might not go to buy these products
- The users have no complete knowledge of these devices. So, if the interface is complex, it will put a full stop to the growth of the product.

Q.2 What is M2M communication?

>>

Machine-to-machine (M2M) communications is used for automated data transmission and measurement between mechanical or electronic devices. The key components of an M2M system are: Field-deployed wireless devices with embedded sensors or RFID-Wireless communication networks with complementary wireline access includes, but is not limited to cellular communication, Wi-Fi, ZigBee, WiMAX, wireless LAN (WLAN), generic DSL (xDSL) and fiber to the x (FTTx).

The main purpose of machine-to-machine communication is to collect data and transmit it to a network. Another goal of M2M is to automatically perform actions that are triggered by sequences of events. Additionally, the art of machine learning can be used so that machines optimize their action sequences. This application for M2M technology is closely related to artificial intelligence and is the basis for the Internet of Things.

Characteristics of M2M

One characteristic of M2M communication is that its low energy use increases the efficiency of systems during data exchanges. The network operator is responsible for service packages – often including monitoring functions – so that users can keep track of important events. Data transfers can be delayed in the network if higher priority data is sent simultaneously. Alternatively, users can schedule data transfers using a timer, or small amounts of data can be transferred

continuously. In logistics, machines can even be programmed by location so that they automatically send out notifications or turn on when they are in a certain area.

Q.3 Explain different characteristics of IOT?

>>

The no-need-to-know in terms of the underlying details of infrastructure, applications interface with the infrastructure via the APIs. The flexibility and elasticity allows these systems to scale up and down at will utilizing the resources of all kinds CPU, storage, server capacity, load balancing, and databases. The “pay as much as used and needed” type of utility computing and the “always on!, anywhere and any place” type of network-based computing. The fundamental characteristics of the IoT are as follows:

Interconnectivity:

With regard to the IoT, anything can be interconnected with the global information and communication infrastructure.

Things-related services:

The IoT is capable of providing thing-related services within the constraints of things, such as privacy protection and semantic consistency between physical things and their associated virtual things. In order to provide thing-related services within the constraints of things, both the technologies in physical world and information world will change.

Heterogeneity:

The devices in the IoT are heterogeneous as based on different hardware platforms and networks. They can interact with other devices or service platforms through different networks

Dynamic changes:

The state of devices change dynamically, e.g., sleeping and waking up, connected and/or disconnected as well as the context of devices including location and speed. Moreover, the number of devices can change dynamically.

Enormous scale:

The number of devices that need to be managed and that communicate with each other will be at least an order of magnitude larger than the devices connected to the current Internet. Even more critical will be the management of the data generated and their interpretation for application purposes. This relates to semantics of data, as well as efficient data handling.

Safety:

As we gain benefits from the IoT, we must not forget about safety. As both the creators and recipients of the IoT, we must design for safety. This includes the safety of our personal data and the safety of our physical well-being. Securing the endpoints, the networks, and the data moving across all of it means creating a security paradigm that will scale.

Connectivity:

Connectivity enables network accessibility and compatibility. Accessibility is getting on a network while compatibility provides the common ability to consume and produce data.

Q.4. Short note on

1] Bluetooth:

>>

- An important short-range communications technology is of course Bluetooth, which has become very important in computing and many consumer product markets.
- It is expected to be key for wearable products in particular, again connecting to the IoT albeit probably via a smartphone in many cases.
- The new Bluetooth Low-Energy (BLE) – or Bluetooth Smart, as it is now branded – is a significant protocol for IoT applications. Importantly, while it offers similar range to Bluetooth it has been designed to offer significantly reduced power consumption.

- Standard: Bluetooth 4.2 core specification
- Frequency: 2.4GHz (ISM)
- Range: 50-150m(Smart/BLE)
- Data Rates: 1Mbps(Smart/BLE)

2] Zigbee:

>>

ZigBee, like Bluetooth, has a large installed base of operation, although perhaps traditionally more in industrial settings.

ZigBee PRO and ZigBee Remote Control (RF4CE), among other available ZigBee profiles, are based on the IEEE802.15.4 protocol, which is an industry-standard wireless networking technology operating at 2.4GHz targeting applications that require relatively infrequent data exchanges at low data-rates over a restricted area and within a 100m range such as in a home or building.

ZigBee/RF4CE has some significant advantages in complex systems offering low-power operation, high security, robustness and high scalability with high node counts and is well positioned to take advantage of wireless control and sensor networks in M2M and IoT applications.

ZigBee architecture has divided into three sections

- IEEE 802.15.4 which consists of MAC and physical layers
- ZigBee layers, which consist of the network layer, the ZigBee device object (ZDO), the application sublayer, and security management
- Manufacturer application: Manufacturers of ZigBee devices can use the ZigBee application profile or develop their own application profile.

The latest version of ZigBee is the recently launched 3.0, which is essentially the unification of the various ZigBee wireless standards into a single standard. An example product and kit for ZigBee development are TI's CC2538SF53RTQT ZigBee System-On-Chip IC and CC2538 ZigBee Development Kit.

Standard: ZigBee 3.0 based on IEEE802.15.4

Frequency: 2.4GHz

Range: 10-100m

3] RFID:

>>

Radio Frequency Identification (RFID) is a method that is used to track or identify an object by radio transmission uses over the web. Data digitally encoded in an RFID tag which might be read by the reader. This is device work as a tag or label during which data read from tags that are stored in the database through the reader as compared to traditional barcodes and QR codes. It is often read outside the road of sight either passive or active RFID.

UHF RFID (Ultra-High Frequency RFID): It is used on shipping pallets and some driver's licenses. Readers send signals in the 902-928 MHz band. Tags communicate at distances of several meters by changing the way they reflect the reader signals; the reader is able to pick up these reflections. This way of operating is called backscatter.

HF RFID (High-Frequency RFID): It operates at 13.56 MHz and is likely to be in your passport, credit cards, books, and noncontact payment systems. HF RFID has a short-range, typically a meter or less because the physical mechanism is based on induction rather than backscatter.

There are also other forms of RFID using other frequencies, such as LF RFID(Low-Frequency RFID), which was developed before HF RFID and used for animal tracking

There are two types of RFID :

1. Passive RFID –

In this device, RF tags are not attached by a power supply and passive RF tag stored their power. When it is emitted from active antennas and the RF tag are used specific frequency like 125-134MHZ as low frequency, 13.56MHZ as a high frequency and 856MHZ to 960MHZ as ultra-high frequency.

2. Active RFID –

In this device, RF tags are attached by a power supply that emits a signal and there is an antenna which receives the data.

Working Principle of RFID :

Generally, RFID uses radio waves to perform AIDC function. AIDC stands for Automatic Identification and Data Capture technology which performs object identification and collection and mapping of the data.

An antenna is an device which converts power into radio waves which are used for communication between reader and tag. RFID readers retrieve the information from RFID tag which detects the tag and reads or writes the data into the tag. It may include one processor, package, storage and transmitter and receiver unit.

Application of RFID :

- It utilized in tracking shipping containers, trucks and railroad, cars.
- It uses in Asset tracking.
- It utilized in credit-card shaped for access application.
- It uses in Personnel tracking.

4] Security challenges in IOT

>>

1. Software and firmware vulnerabilities

Ensuring the security of IoT systems is tricky, mostly because a lot of smart devices are resource-constrained and have limited computing power. Thus, they can't run powerful, resource-hungry security functions and are likely to have more vulnerabilities than non-IoT devices.

Many IoT systems have security vulnerabilities for the following reasons:

- Lack of computational capacity for efficient built-in security
- Poor access control in IoT systems
- Limited budget for properly testing and improving firmware security
- Lack of regular patches and updates due to limited budgets and technical limitations of IoT devices
- Users may not update their devices, thus restricting vulnerability patching
- With time, software updates might be unavailable for older devices
- Poor protection from physical attacks: an attacker can get close enough to add their chip or hack the device using radio waves

Malicious actors aim to leverage vulnerabilities they've found in a target IoT system to compromise its communications, install malware, and steal valuable data. For example, the use of vulnerable credentials like weak, recycled, and default passwords allowed hackers to

hack **Ring smart cameras**. They even managed to communicate with victims remotely using the camera's microphone and speakers.

2. Insecure communications

Most existing security mechanisms were initially designed for desktop computers and are difficult to implement on resource-constrained IoT devices. That's why traditional security measures aren't as efficient when it comes to protecting the communication of IoT devices.

One of the most dangerous threats caused by insecure communications is the possibility of a **man-in-the-middle (MitM) attack**. Hackers can easily perform MitM attacks to compromise an update procedure and take control of your device if it doesn't use secure encryption and authentication mechanisms. Attackers can even install malware or change a device's functionality. Even if your device doesn't fall victim to an MitM attack, the data it exchanges with other devices and systems can still be captured by cybercriminals if your device sends it in cleartext messages.

Connected devices are susceptible to attacks from other devices. For instance, if attackers gain access to just one device in a home network, they can easily compromise all other unisolated devices in it.

3. Data leaks from IoT systems

We've already established that by capturing unencrypted messages from your IoT system, hackers can get access to the data it processes. This might include even sensitive data like your location, bank account details, and health records. However, abusing poorly secured communications isn't the only way attackers can gather valuable data.

All data is transferred via and stored in the cloud, and cloud-hosted services can also experience external attacks. Thus, data leaks are possible from both devices themselves and the cloud environments they're connected to.

Third-party services in your IoT systems are another possible source of a data leak. For instance, **Ring smart doorbells** were found to be sending customer data to companies such as Facebook and Google without proper customer consent. This incident appeared because of third-party tracking services enabled in the Ring mobile app.

4. Cyberattacks

Apart from the malware and MITM attacks discussed above, IoT systems can also be susceptible to various cyberattacks. Here's a list of the most common types of attacks on IoT devices:

1. **Denial-of-service (DoS) attacks:** IoT devices have limited processing power, which makes them highly vulnerable to denial-of-service attacks. During a DoS attack, a device's ability to respond to legitimate requests is compromised due to a flood of fake traffic.
2. **Denial-of-sleep (DoSL) attacks:** Sensors connected to a wireless network should continuously monitor the environment, so they're often powered by batteries that don't require frequent charging. Battery power is preserved by keeping the device in sleep

mode most of the time. Sleep and awake modes are controlled according to the communication needs of different protocols, such as medium access control (MAC). Attackers may exploit vulnerabilities of the MAC protocol to carry out a DoSL attack. This type of attack drains battery power and thus disables the sensor.

3. **Device spoofing:** This attack is possible when a device has improperly implemented digital signatures and encryption. For instance, a poor public key infrastructure (PKI) may be exploited by hackers to “spoof” a network device and disrupt IoT deployments.
4. **Physical intrusion:** Though most attacks are performed remotely, physical intrusion of a device is also possible if it’s stolen. Attackers can tamper with device components to make them operate in an unintended way.
5. **Application-based attacks:** These types of attacks are possible when there are security vulnerabilities in device firmware or software used on embedded systems or weaknesses in cloud servers or backend applications.

5] Levels of IOT:

>>

Physical Devices and Controllers:

These are similar to the sensor nodes discussed earlier. The actual things in the Internet of Things.

Connectivity:

The nodes in sensor nodes discussed earlier can be related to this. The basic device that has some connectivity radios and contains the sensors. The Network devices also fall in the same segment.

Edge Computing:

The small level data processing at the gateway and sensor nodes is known as edge computing. Some small tasks and actuator handling can also be done here.

Data Accumulation:

The data that is sent over the internet via gateways by the sensor nodes are acquired and stored in a database on the Cloud.

Data Abstraction:

Normalization, filtering, expansion, aggregation of the data is done at the cloud mainly. The main aim is to get the required and significant data out of all the data that is collected. Some small operations can be done at the Edge also.

Application:

Analytics over the data and responding according to the data to control the actuators at the sensor nodes is one of the main applications of the IoT Architecture.

Collaboration and Processing:

This is the final level of the IoT Reference model. The human interaction and involvement in the IoT scenario are seen as the most neglected parts. Not only the device should be smart enough to perform certain tasks but they should also have some intuitive interactions with the human. The involvement of People and Business processes is an essential part of developing an interesting IoT application.

IOT Answers

1) What is vision of IOT

The Internet of Things (IoT) describes the network of physical objects—“things” that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet.

The end goal is to have plug-n-play smart objects that can be deployed in any environment with an interoperable interconnection backbone that allows them to blend with other smart objects around them. A vision where things (wearable, watch, alarm clock, home devices, surrounding objects with) become smart and behave alive through sensing, computing and communicating systems. A vision where embedded devices interact with remote objects or persons through connectivity, for examples, using Internet or Near Field Communication or other technologies.

2) Characteristics of IOT

The no-need-to-know in terms of the underlying details of infrastructure, applications interface with the infrastructure via the APIs. The flexibility and elasticity allows these systems to scale up and down at will utilizing the resources of all kinds CPU, storage, server capacity, load balancing, and databases. The “pay as much as used and needed” type of utility computing and the “always on!, anywhere and any place” type of network-based computing. The fundamental characteristics of the IoT are as follows:

Interconnectivity: With regard to the IoT, anything can be interconnected with the global information and communication infrastructure.

Things-related services: The IoT is capable of providing thing-related services within the constraints of things, such as privacy protection and semantic consistency between physical things and their associated virtual things. In order to provide thing-related services within the constraints of things, both the technologies in physical world and information world will change.

Heterogeneity: The devices in the IoT are heterogeneous as based on different hardware platforms and networks. They can interact with other devices or service platforms through different networks.

Dynamic changes: The state of devices change dynamically, e.g., sleeping and waking up, connected and/or disconnected as well as the context of devices including location and speed. Moreover, the number of devices can change dynamically.

Enormous scale: The number of devices that need to be managed and that communicate with each other will be at least an order of magnitude larger than the devices connected to the current Internet. Even more critical will be the management of the data generated and their interpretation for application purposes. This relates to semantics of data, as well as efficient data handling.

Safety: As we gain benefits from the IoT, we must not forget about safety. As both the creators and recipients of the IoT, we must design for safety. This includes the safety of our personal data and the safety of our physical well-being. Securing the endpoints, the networks, and the data moving across all of it means creating a security paradigm that will scale.

Connectivity: Connectivity enables network accessibility and compatibility. Accessibility is getting on a network while compatibility provides the common ability to consume and produce data.

3) RFID

RFID or Radio Frequency Identification is an automatic identification method that uses wireless non-contact radio frequency waves in which data is digitally encoded in RFID tags or smart labels which can be read by reader through radio waves. The transfer of data takes place between a reader and a movable thing that can be identified & track. RFID can be considered similar to barcodes as data read from tags are stored in database or you can say device which captures label's data, stores data in a database.

The major difference between barcodes or QR codes and RFID is that RFID tag data can be read outside line-of-sight whereas traditional barcodes can't. RFID doesn't require any physical contact between reader or scanner and tagged item. There is a microchip placed inside a label which is used to transfer data when label is exposed to radio waves. RFID tags are mainly used in industries for tracking progress of a product.

RFID System:

RFID System composed of RFID Reader & RFID Tags.

1. RFID Reader –

It is a device used to communicate with RFID Tag which consists of one or more antennas, used to emit radio waves & receive signals back, from RFID Tag. The RFID reader is also called as interrogator as it used to interrogate RFID Tag.

2. RFID Tags –

RFID Tags consists of 2 parts:

- Integrated Circuit :**

It is used for storing & processing data.

- Antenna :**

It is used for transmitting receiving signal.

- Active Tag :**

These have their own power supply and allows a read range of about 100 feet.

- Passive Tag :**

A reader inductively gives power to Passive Tags as they don't have their own power supply. Passive Tags are most widely used Tag and their read range is approximately 30 feet.

Working Principle of RFID:

Generally, RFID uses radio waves to perform AIDC function. AIDC stands for Automatic Identification and Data Capture technology which performs object identification and collection and mapping of the data. An antenna is a device which converts power into radio waves which are used for communication between reader and tag. RFID readers retrieve the information from RFID tag which detects the tag and reads or writes the data into the tag. It may include one processor, package, storage and transmitter and receiver unit.

Application of RFID:

- Document tracking.
- Controlling access to restricted areas
- Personnel tracking
- Inventory management
- Supply chain management
- Manufacturing
- Healthcare

4) Bluetooth

An important short-range communications technology is of course Bluetooth, which has become very important in computing and many consumer product markets.

- It is expected to be key for wearable products in particular, again connecting to the IoT albeit probably via a smartphone in many cases.
- The new Bluetooth Low-Energy (BLE) – or Bluetooth Smart, as it is now branded – is a significant protocol for IoT applications. Importantly, while it offers similar range to Bluetooth it has been designed to offer significantly reduced power consumption.
- Standard: Bluetooth 4.2 core specification
- Frequency: 2.4GHz (ISM)
- Range: 50-150m (Smart/BLE)
- Data Rates: 1Mbps (Smart/BLE)

5) Zigbee

ZigBee is a Personal Area Network task group with low rate task group 4. It is a technology of home networking. ZigBee is a technological standard created for controlling and sensing the network. As we know that ZigBee is the Personal Area network of task group 4 so it is based on IEEE 802.15.4 and is created by Zigbee Alliance.

ZigBee is a standard that addresses the need for very low-cost implementation of Low power devices with Low data rates for short-range wireless communications.

Types of ZigBee Devices:

- **Zigbee Coordinator Device:** It communicates with routers. This device is used for connecting the devices.
- **Zigbee Router:** It is used for passing the data between devices.
- **Zigbee End Device:** It is the device that is going to be controlled.

General Characteristics of Zigbee Standard:

- Low Power Consumption

- Low Data Rate (20- 250 kbps)
- Short-Range (75-100 meters)
- Support Small and Large Networks (up to 65000 devices (Theory); 240 devices (Practically))
- Low Cost of Products and Cheap Implementation (Open Source Protocol)
- Extremely low duty cycle.

Operating Frequency Bands (Only one channel will be selected for use in a network):

1. **Channel 0:** 868 MHz (Europe)
2. **Channel 1-10:** 915 MHz (the US and Australia)
3. **Channel 11-26:** 2.4 GHz (Across the World)

Zigbee Network Topologies:

- **Star Topology** (ZigBee Smart Energy): Consists of a coordinator and several end devices, end devices communicate only with the coordinator.
- **Mesh Topology** (Self Healing Process): Mesh topology consists of one coordinator, several routers, and end devices.
- **Tree Topology**: In this topology, the network consists of a central node which is a coordinator, several routers, and end devices. the function of the router is to extend the network coverage.

Architecture of Zigbee:

Zigbee architecture is a combination of 6 layers.

1. Application Layer
 2. Application Interface Layer
 3. Security Layer
 4. Network Layer
 5. Medium Access Control Layer
 6. Physical Layer
- **Physical layer:** The lowest two layers i.e the physical and the MAC (Medium Access Control) Layer are defined by the IEEE 802.15.4 specifications. The Physical layer is closest to the hardware and directly controls and communicates with the Zigbee radio. The physical layer translates the data packets in the over-the-air bits for transmission and vice-versa during the reception.
 - **Medium Access Control layer (MAC layer):** The layer is responsible for the interface between the physical and network layer. The MAC layer is

also responsible for providing PAN ID and also network discovery through beacon requests.

- **Network layer:** This layer acts as an interface between the MAC layer and the application layer. It is responsible for mesh networking.
- **Application layer:** The application layer in the Zigbee stack is the highest protocol layer and it consists of the application support sub-layer and Zigbee device object. It contains manufacturer-defined applications.

Zigbee Applications:

1. Home Automation
2. Medical Data Collection
3. Industrial Control Systems
4. meter reading system
5. light control system

6) Security Challenges in IOT

1. Lack of encryption –

Although encryption is a great way to prevent hackers from accessing data, it is also one of the leading IoT security challenges.

These drives like the storage and processing capabilities that would be found on a traditional computer.

The result is an increase in attacks where hackers can easily manipulate the algorithms that were designed for protection.

2. Insufficient testing and updating –

With the increase in the number of IoT(internet of things) devices, IoT manufacturers are more eager to produce and deliver their device as fast as they can without giving security too much of although.

Most of these devices and IoT products do not get enough testing and updates and are prone to hackers and other security issues.

3. Brute forcing and the risk of default passwords –

Weak credentials and login details leave nearly all IoT devices vulnerable to password hacking and brute force.

Any company that uses factory default credentials on their devices is placing both their business and its assets and the customer and their valuable information at risk of being susceptible to a brute force attack.

4. IoT Malware and ransomware –

Increases with increase in devices.

Ransomware uses encryption to effectively lock out users from various devices and platforms and still use a user's valuable data and info.

5. Cyberattacks-

Apart from the malware and MITM attacks discussed above, IoT systems can also be susceptible to various cyberattacks. Here's a list of the most common types of attacks on IoT devices:

1. Denial-of-service (DoS) attacks: IoT devices have limited processing power, which makes them highly vulnerable to denial-of-service attacks. During a DoS attack, a device's ability to respond to legitimate requests is compromised due to a flood of fake traffic.

2. Denial-of-sleep (DoSL) attacks: Sensors connected to a wireless network should continuously monitor the environment, so they're often powered by batteries that don't require frequent charging. Battery power is preserved by keeping the device in sleep mode most of the time. Sleep and awake modes are controlled according to the communication needs of different protocols, such as medium access control (MAC). Attackers may exploit vulnerabilities of the MAC protocol to carry out a DoSL attack. This type of attack drains battery power and thus disables the sensor.

3. Device spoofing: This attack is possible when a device has improperly implemented digital signatures and encryption. For instance, a poor public key infrastructure (PKI) may be exploited by hackers to "spoof" a network device and disrupt IoT deployments.

4. Physical intrusion: Though most attacks are performed remotely, physical intrusion of a device is also possible if it's stolen. Attackers can tamper with device components to make them operate in an unintended way.

5. Application-based attacks: These types of attacks are possible when there are security vulnerabilities in device firmware or software used on embedded systems or weaknesses in cloud servers or backend applications.

Example –

A hacker can hijack a computer camera and take pictures.

By using malware access points, the hackers can demand ransom to unlock the device and return the data.

IoT botnet aiming at cryptocurrency –

IoT botnet workers can manipulate data privacy, which could be massive risks for an open Crypto market. The exact value and creation of cryptocurrencies code face danger from mal-intentioned hackers.

The blockchain companies are trying to boost security. Blockchain technology itself is not particularly vulnerable, but the app development process is.

7) Levels of IOT:

Physical Devices and Controllers: These are similar to the sensor nodes discussed earlier. The actual things in the Internet of Things.

Connectivity: The nodes in sensor nodes discussed earlier can be related to this. The basic device that has some connectivity radios and contains the sensors. The Network devices also fall in the same segment.

Edge Computing: The small level data processing at the gateway and sensor nodes is known as edge computing. Some small tasks and actuator handling can also be done here.

Data Accumulation: The data that is sent over the internet via gateways by the sensor nodes are acquired and stored in a database on the Cloud.

Data Abstraction: Normalization, filtering, expansion, aggregation of the data is done at the cloud mainly. The main aim is to get the required and significant data out of all the data that is collected. Some small operations can be done at the Edge also.

Application: Analytics over the data and responding according to the data to control the actuators at the sensor nodes is one of the main applications of the IoT Architecture.

Collaboration and Processing: This is the final level of the IoT Reference model. The human interaction and involvement in the IoT scenario are seen as the most neglected parts. Not only the device should be smart enough to perform certain tasks but they should also have some intuitive interactions with the human. The involvement of People and Business processes is an essential part of developing an interesting IoT application.

8) Challenges in the world of IOT

The **Internet Of Things** has been facing many areas like Information Technology, Healthcare, Data Analytics and Agriculture. The main focus is on protecting privacy as it is the primary reason for other challenges including government participation. Integrated effort from the government, civil society and private sectors would play a vital role in protecting the following values given below in to prevent IoT from getting hampered:

Scalability:

Billions of internet-enabled devices get connected in a huge network, large volumes of data are needed to be processed. The system that stores, analyses the data from these IoT devices needs to be scalable. In present, the era of IoT evolution everyday objects are connected with each other via Internet. The raw data obtained from these devices need big data analytics and cloud storage for interpretation of useful data.

Interoperability:

Technological standards in most areas are still fragmented. These technologies need to be converged. Which would help us in establishing a common framework and the standard for the IoT devices. As the standardization process is still lacking, interoperability of IoT with legacy devices should be considered critical. This lack of interoperability is preventing us to move towards the vision of truly connected everyday interoperable smart objects.

Lack of government support:

Government and Regulatory bodies like FDA should come up and bring up regulations by setting up a standard committee for safety and security of devices and people.

Safety Of Patients:

Most Of IoT devices are left unattended, as they are connected with real-world objects. If used on patients as wearable devices, any technical error in security can be life-threatening for patient.

Security And Personal Privacy:

There has been no research in security vulnerabilities and its improvements. It should ensure Confidentiality, Integrity and Availability of personal data of patient.

Design Based Challenge:

With the development in technology design challenges are increasing at a faster rate. There have been issues regarding design like limited computation power, limited energy and limited memory which need to be sorted out.

9) What is M2M communication

In this the interaction or communication takes place between machines by automating data/programs. In this machine level instructions are required for communication. Here communication takes place without human interaction. The machines may be either connected through wires or by wireless connection. An M2M connection is a point-to-point connection between two network devices that helps in transmitting information using public networking technologies like Ethernet and cellular networks. IoT uses the basic concepts of M2M and expands by creating large “cloud” networks of devices that communicate with one another through cloud networking platforms. The main purpose of machine-to-machine communication is to collect data and transmit it to a network.

Advantages

This M2M can operate over cellular networks and is simple to manage. It can be used both indoors and outdoors and aids in the communication of smart objects without the need for human interaction. The M2M contact facility is used to address security and privacy problems in IoT networks. Large-scale data collection, processing, and security are all feasible.

Disadvantages

However, in M2M, use of cloud computing restricts versatility and creativity. Data security and ownership are major concerns here. The challenge of achieving interoperability between cloud/M2M IoT systems is daunting. M2M connectivity necessitates the existence of a reliable internet connection.

Examples:

- Smart Washing machine sends alerts to the owners' smart devices after completion of washing or drying of clothes.
- Smart meters tracks amount of energy used in household or in companies and automatically alert the owner.

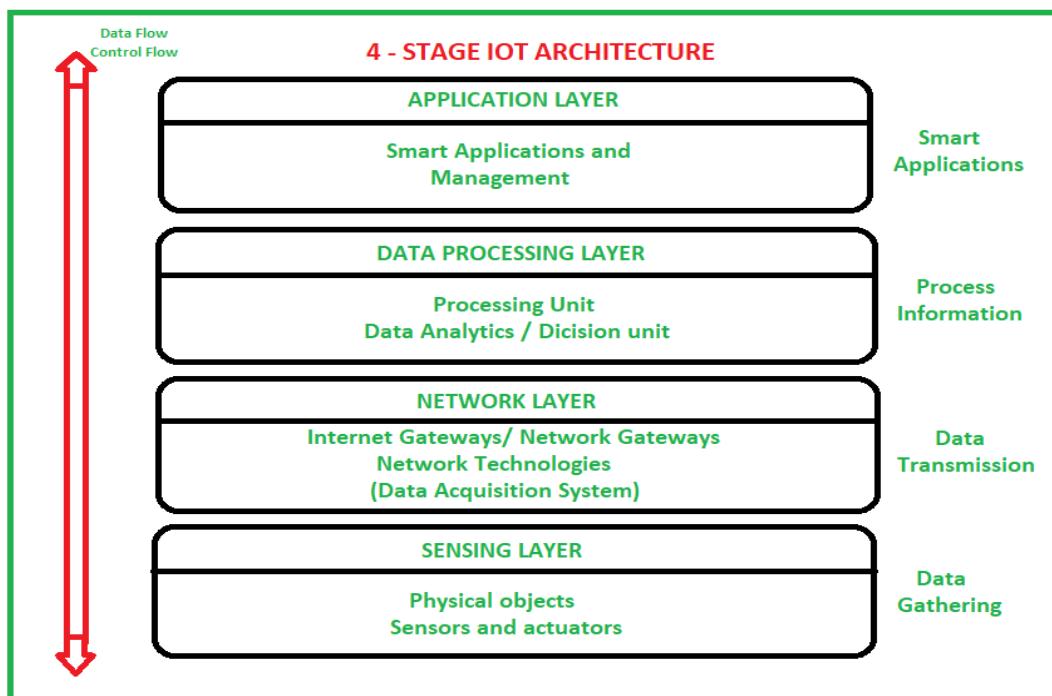
10) Need for security in IOT

There have already been lots of cases of IoT devices being hacked when criminals have searched for IoT security vulnerabilities and been successful. Some businesses have even had their industrial robots hacked as well as equipment connected to them. The reason is that hackers are able to alter control-loop parameters, tamper with production logic, and alter the robot's state, and much more.

A group of researchers decided to demonstrate how much damage a hacked robot can actually do. They found vulnerabilities in the robotic arm's system and were able to program the robot to cause millions of dollars' worth of damage to the products it was manufacturing. Cybercriminals will stop at nothing, even hacking medical equipment.

11) Architecture of IOT

[Internet of Things \(IoT\)](#) technology has a wide variety of applications and use of Internet of Things is growing so faster. Depending upon different application areas of Internet of Things, it works accordingly as per it has been designed/developed. But it has not a standard defined architecture of working which is strictly followed universally. The architecture of IoT depends upon its functionality and implementation in different sectors. Still, there is a basic process flow based on which IoT is built.



There are 4 layers present that can be divided as follows: Sensing Layer, Network Layer, Data processing Layer, and Application Layer. These are explained as following below.

1. Sensing Layer –

Sensors, actuators, devices are present in this Sensing layer. These Sensors or Actuators accept data(physical/environmental parameters), processes data and emits data over network.

2. Network Layer –

Internet/Network gateways, Data Acquisition System (DAS) are present in this layer. DAS performs data aggregation and conversion function (Collecting data and aggregating data then converting analog data of sensors to digital data etc). Advanced gateways which mainly opens up connection between Sensor networks and Internet also performs many basic gateway functionalities like malware protection, and filtering also some times decision making based on inputted data and data management services, etc.

3. Data processing Layer –

This is processing unit of IoT ecosystem. Here data is analyzed and pre-processed before sending it to data center from where data is accessed by software applications often termed as business applications where data is monitored and managed and further actions are also prepared. So here Edge IT or edge analytics comes into picture.

4. Application Layer –

This is last layer of 4 stages of IoT architecture. Data centers or cloud is management stage of data where data is managed and is used by end-user applications like agriculture, health care, aerospace, farming, defense, etc.

Unit II

Embedded IoT Platform Design Methodology

INTERNET OF THINGS

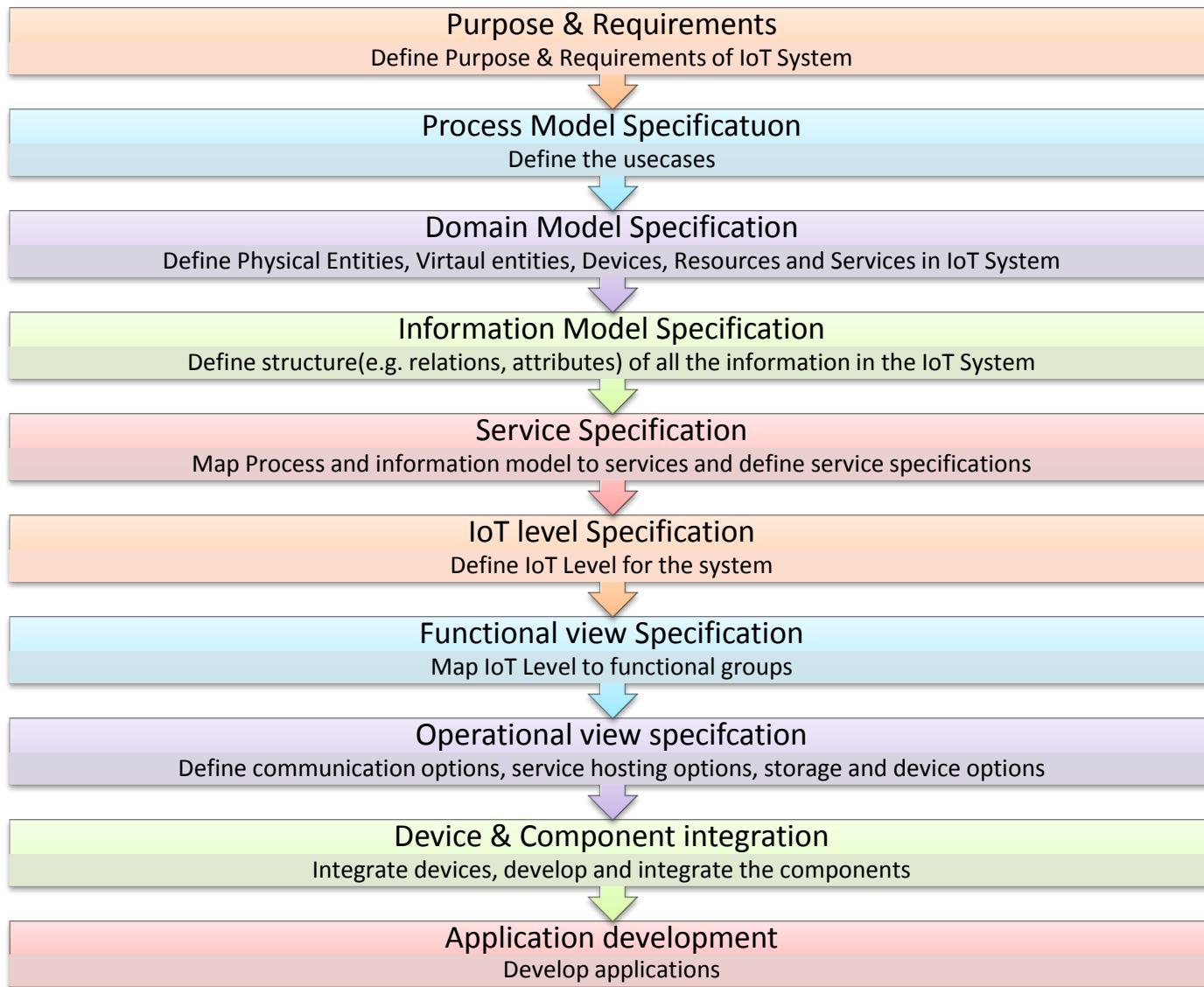
A Hands-On Approach



Outline

- Purpose and Requirement Specification
- Process Specification
- Domain model specification
- Information Model Specification
- Service specifications
- IoT Level Specifications
- Functional view specification
- Operational View Specification
- Device and Component integration
- Application development

Steps involved in IoT System Design Methodology



Advantages of Using Design methodology

- Reducing the design, testing and maintenance time
- Provide better interoperability
- Reduce the complexity

Step 1 : Purpose and Requirement Specification

Defines

- **System purpose**
- **behavior and**
- **Requirements** (such as **data collection** requirements, **data analysis** requirement, **system management** requirements, **data privacy and security** requirements, **User interfaces** requirements)

Step 1 : Purpose and Requirement Specification

- **Purpose** : An automated irrigation mechanism which turns the pumping motor ON and OFF on detecting the moisture content of the earth without the intervention of human
- **Behavior** : System should monitor the amount of soil moisture content in soil. In case the soil moisture of the soil deviates from the specified range, the watering system is turned ON/OFF. In case of dry soil, it will activate the irrigation system, pumping water for watering the plants.
- **System Management Requirements** : system should remotely provide monitoring and control functions

Step 1 : Purpose and Requirement Specification

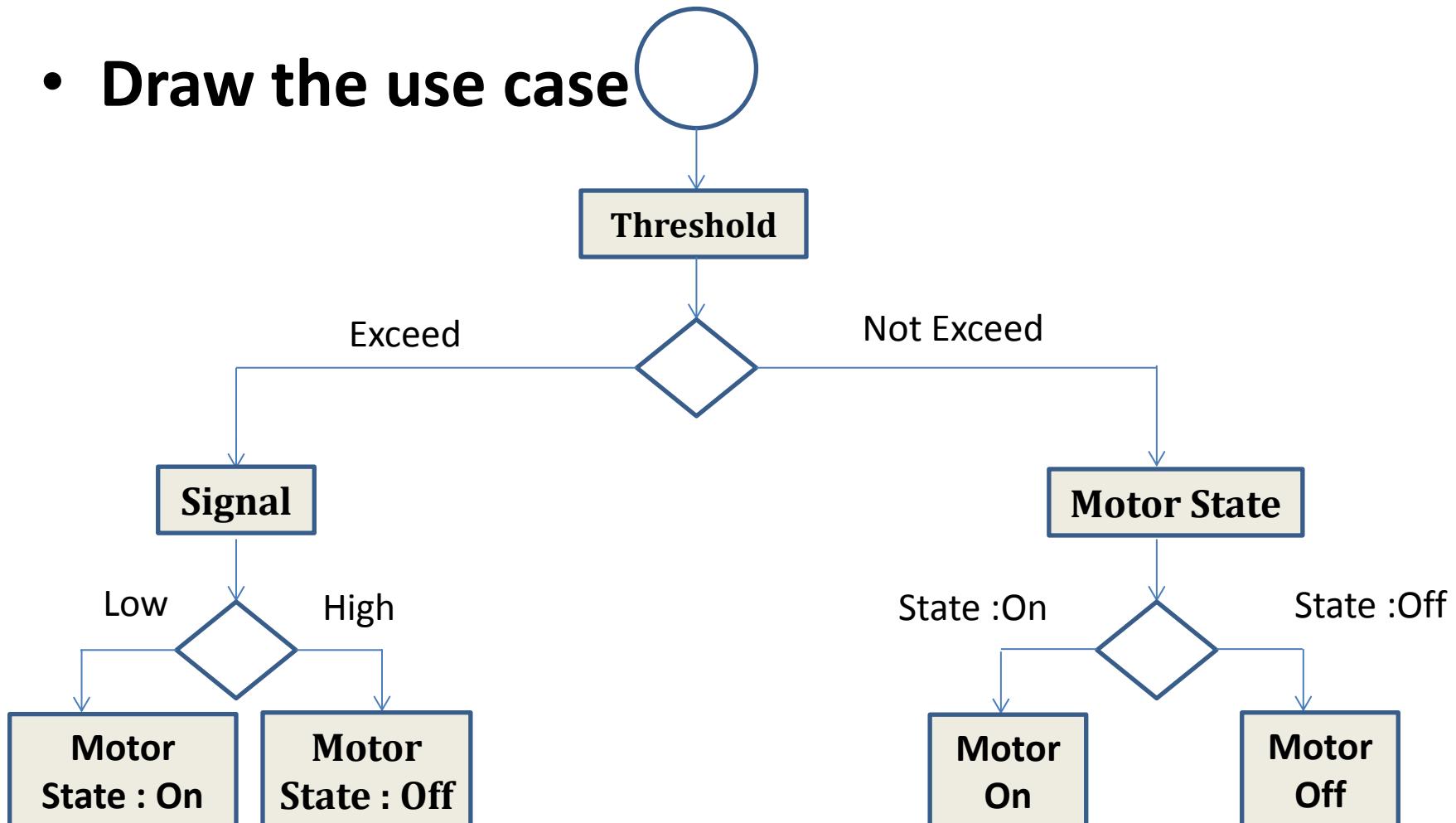
- **Data Analysis Requirements :** system should perform local analysis of data
- **Application Deployment Requirement :** Deployed locally on device, but acts remotely without manual intervention.
- **Security :** Authentication to Use the system must be available

Step 2 : Process Specification

- Define the process with the help of use cases
- The use cases are formally described based on Purpose & requirement specification
- In this use case :
 - **Circle** denotes a state or an attribute

Step 2 : Process Specification

- Draw the use case



Step 3 : Domain Model Specification

- **Describes the main concepts, entities and objects** in the domain of IoT system to be designed
- Entities , Objects and Concepts include the following : Physical entity, Virtual entity , Device, Resource, Service

Step 3 : Domain Model Specification

- **Physical Entity:**
 - Discreet identifiable entity in physical environment
 - For eg. Pump, motor, LCD
 - The IoT System provides the information about the physical entity (using sensors) or performs actuation upon the Physical entity(like switching a motor on etc.)
 - In smart irrigation example, there are three Physical entities involved :
 - **Soil (whose moisture content is to be monitored)**
 - **Motor (to be controlled)**
 - **Pump (To be controlled)**
- **Virtual Entity:**
 - Representation of physical entity in digital world
 - For each physical entity there is a virtual entity

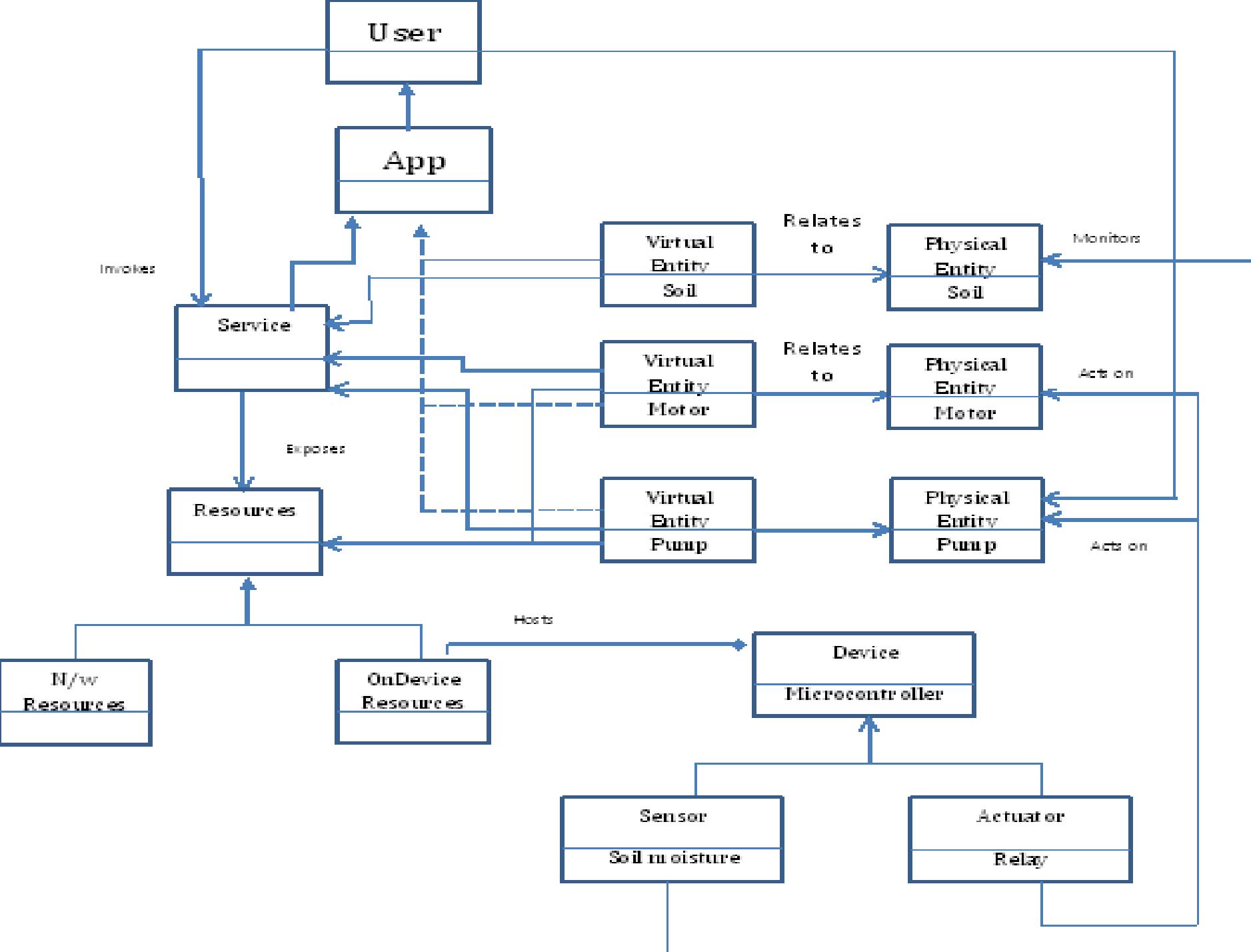
Step 3 : Domain Model Specification

- **Device:**
 - Medium for interactions between Physical and Virtual Entities.
 - Devices (Sensors) are used to gather information from the physical entities
 - Devices are used to identify Physical entities (Using Tags)
 - In Smart Irrigation System, device is soil moisture sensor and buzzer as well as the actuator (relay switch) attached to it.

Step 3 : Domain Model Specification

- In smart irrigation system there are three services :
 - A service that sets the signal to low/ high depending upon the threshold value
 - A service that sets the motor state on/off
 - A controller service that runs and monitors the threshold value of the moisture and switches the state of motor on/off depending upon it.

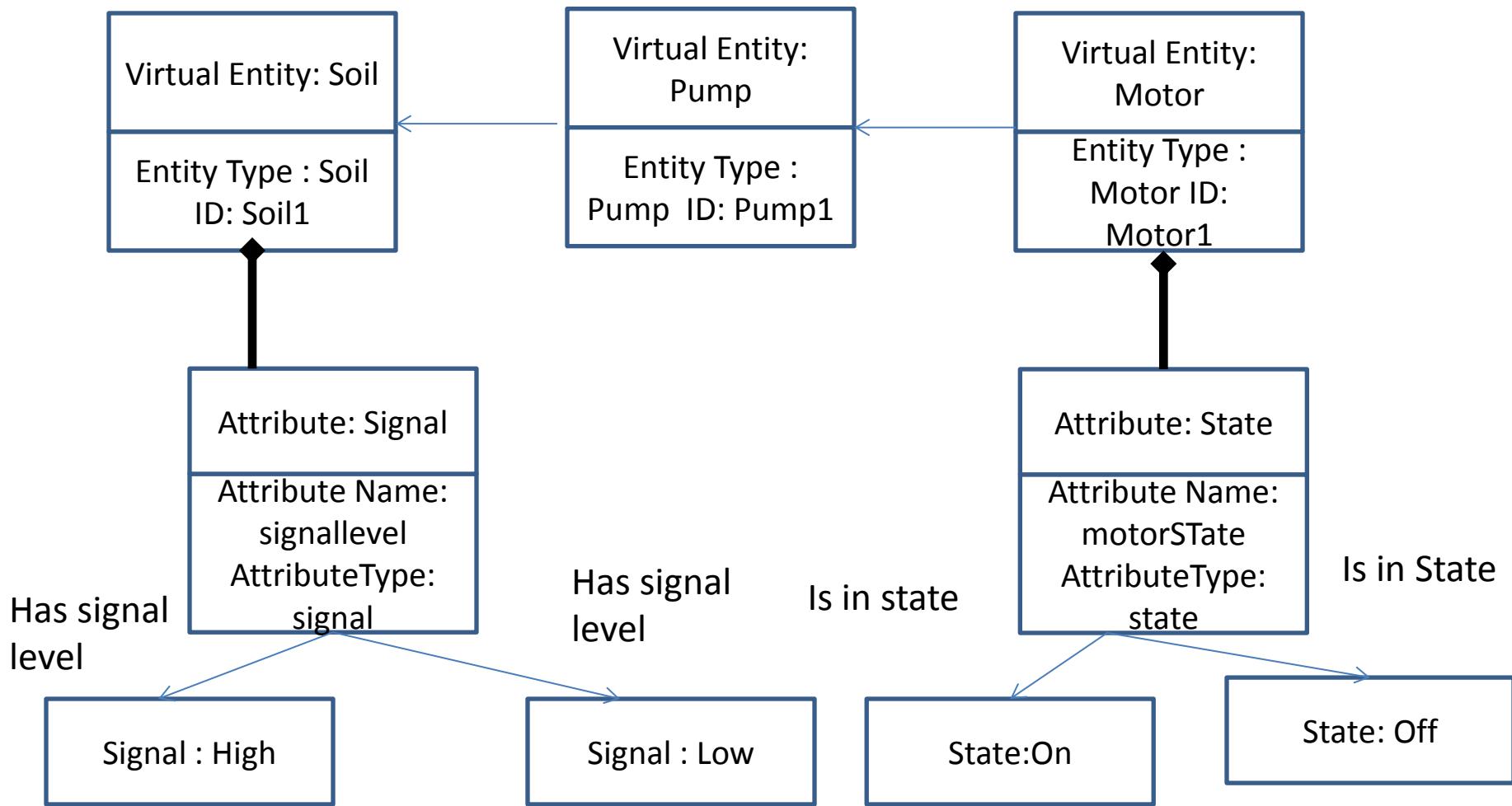
When threshold value is not crossed the controller retrieves the motor status from database and switches the motor on/off.



Step 4 : Information Model Specification

- Defines the structure of all the information in the IoT system (such as attributes, relations etc.)
- It does not describe the specifics of how the information is represented or stored.
- This adds more information to the Virtual entities by defining their attributes and relations
- I: e, **Draw Class diagram**

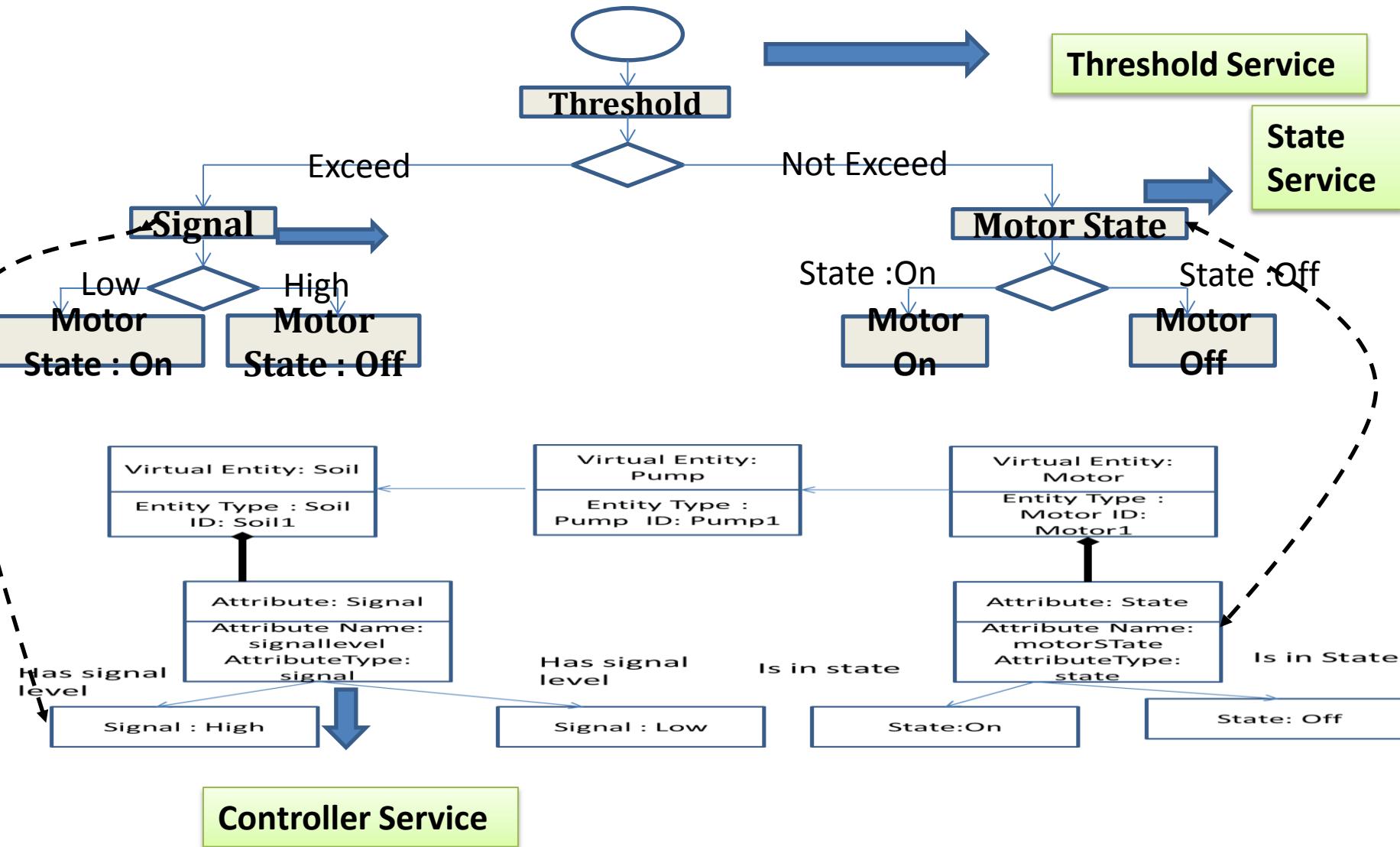
Step 4 : Information Model Specification



Step 5 : Service Specification

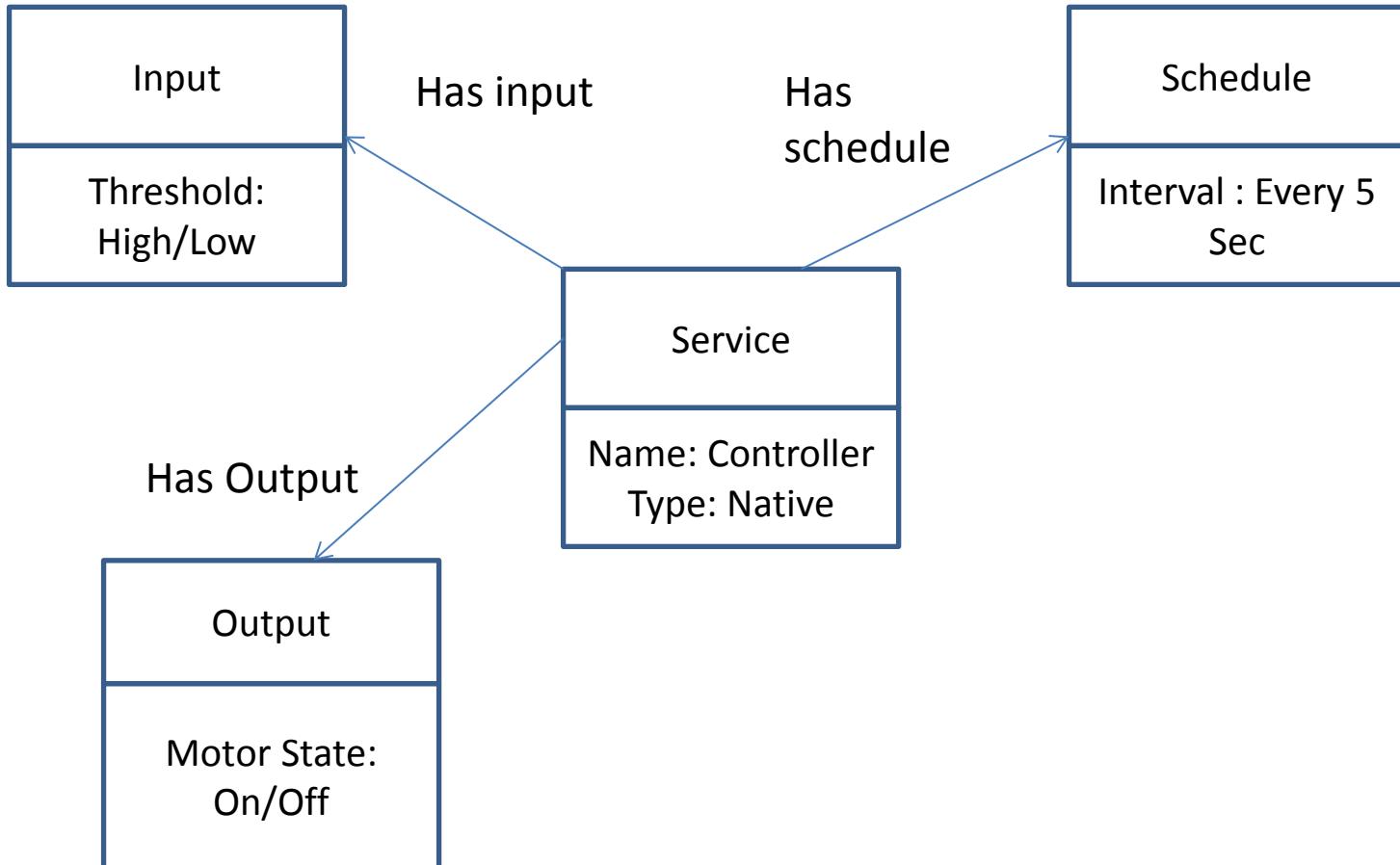
- Define the services in IoT System, service **types**, service **inputs/outputs**, service **endpoints**, service **schedules**, service **preconditions** and service **effects**
- **Services can be controller service, Threshold service, state service, for smart irrigation system**
- These services either change the state/attribute values or retrieve the current values.
- For eg.
 - Threshold service sets signal to high or low depending upon the soil moisture value.
 - State service sets the motor state : on or off
 - Controller service monitors the threshold value as well as the motor state and switches the motor on/off and updates the status in the database

Step 5 : Service Specification



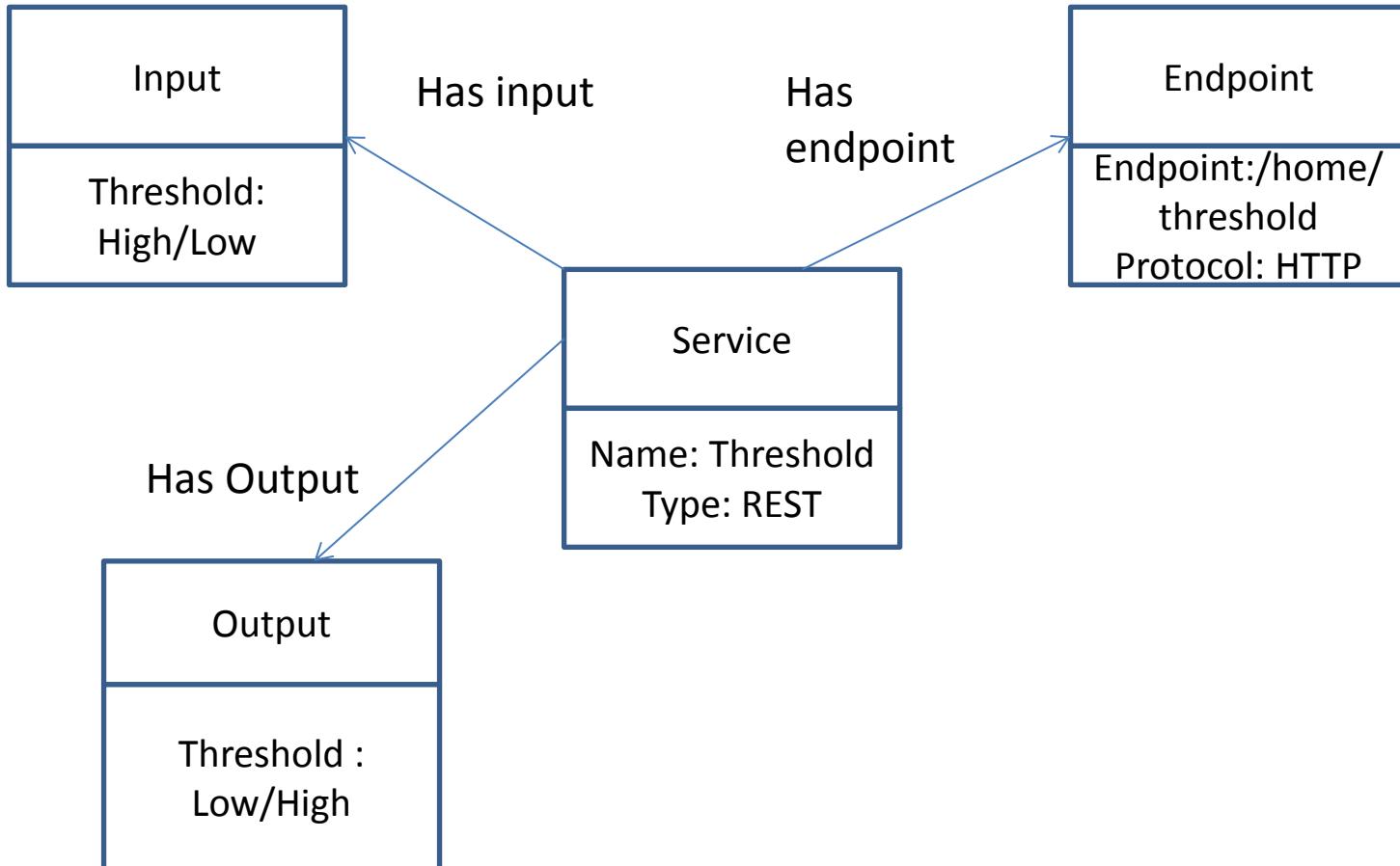
Step 5 : Service Specification

..>Controller Service

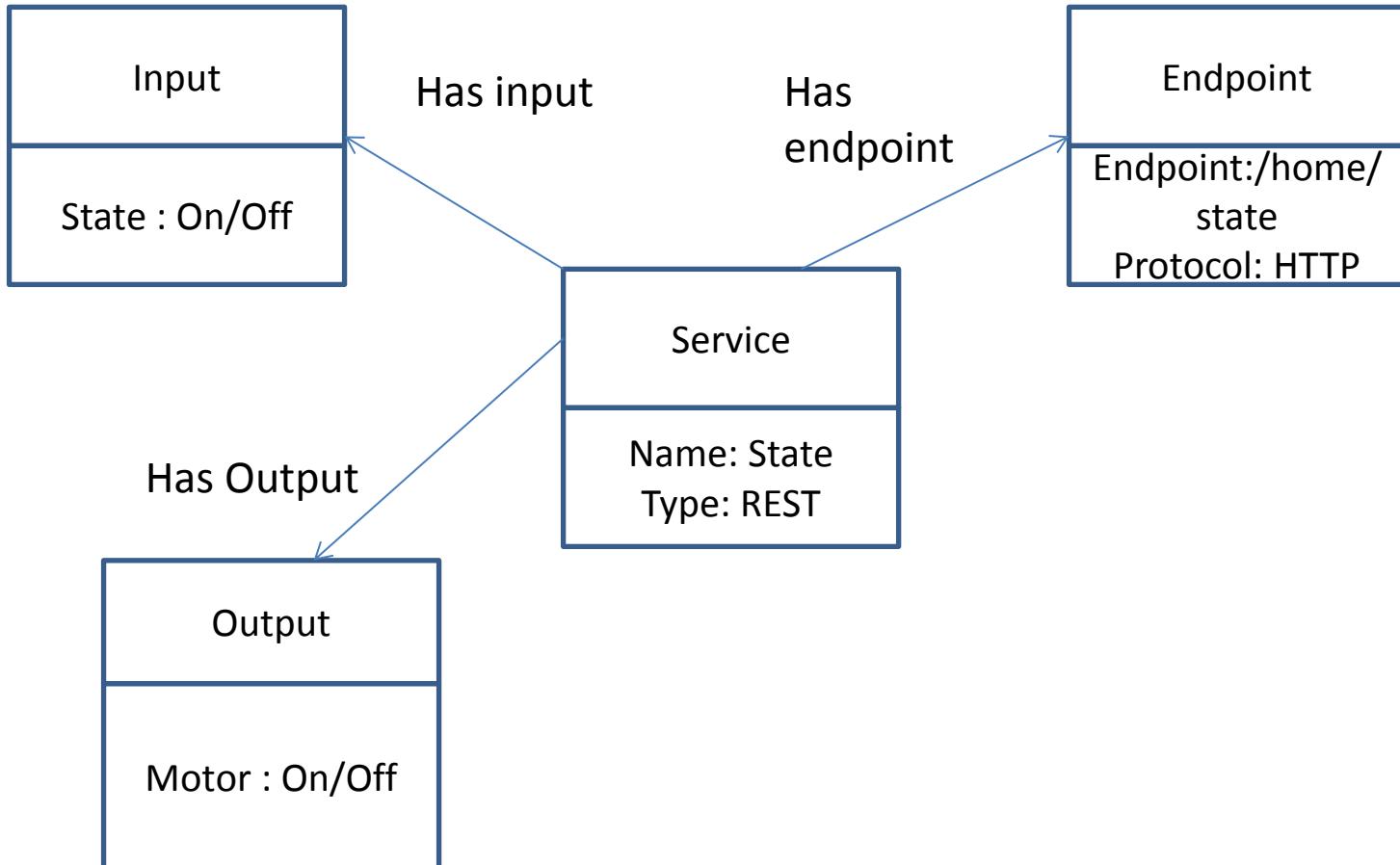


Step 5 : Service Specification

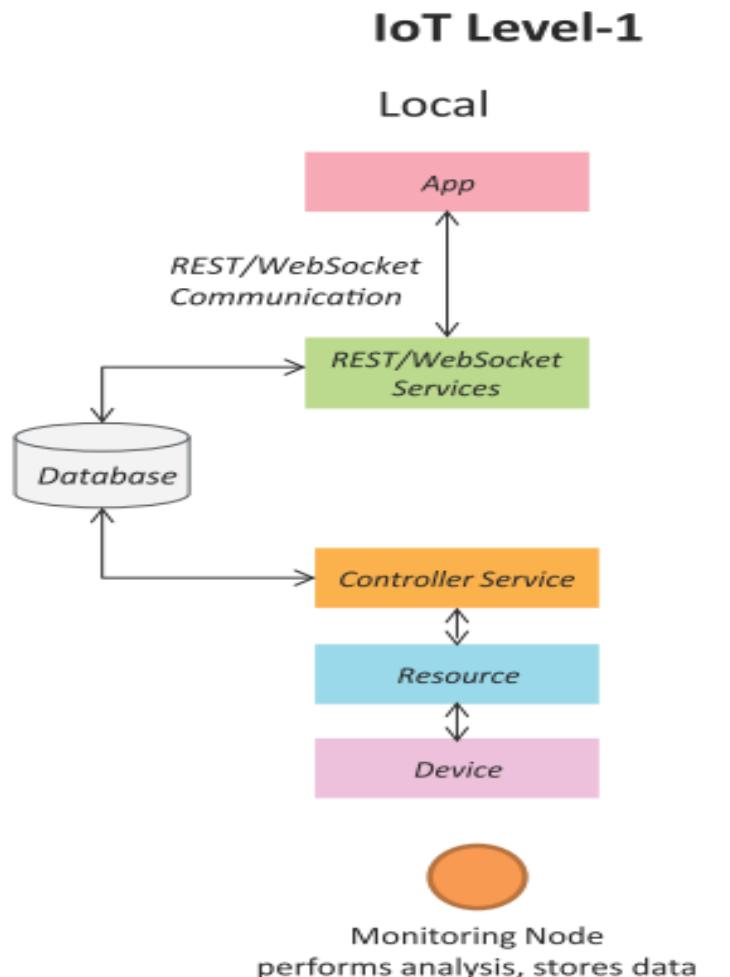
..>Threshold Service



Step 5 : Service Specification ..>State Service



Step 6 : IoT Level Specification



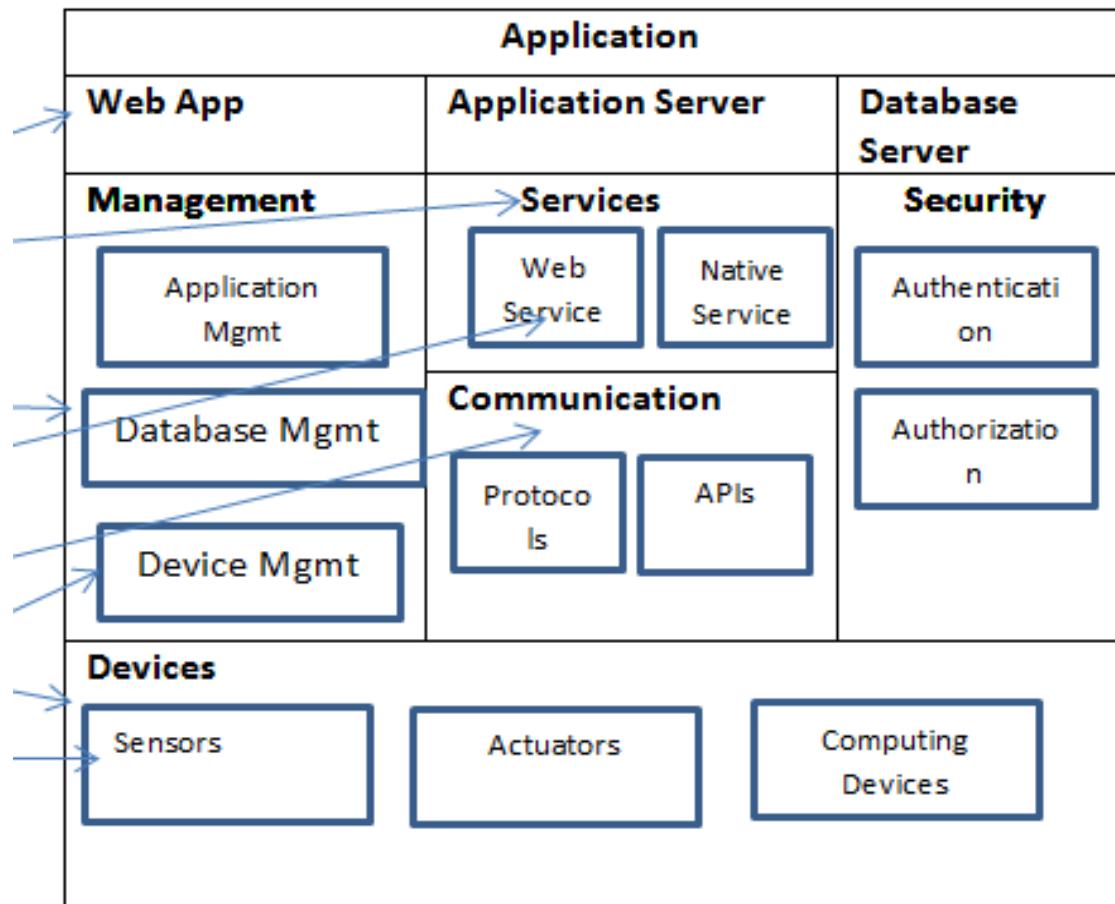
- Decide the deployment level of IoT System.
Here I am using Deployment Level 1.

Step 7 : Functional View Specification

- Define the functions of IoT System grouped into various functional groups.
- These functional groups provide functionalities for interacting with the concepts defined in Domain model specification.

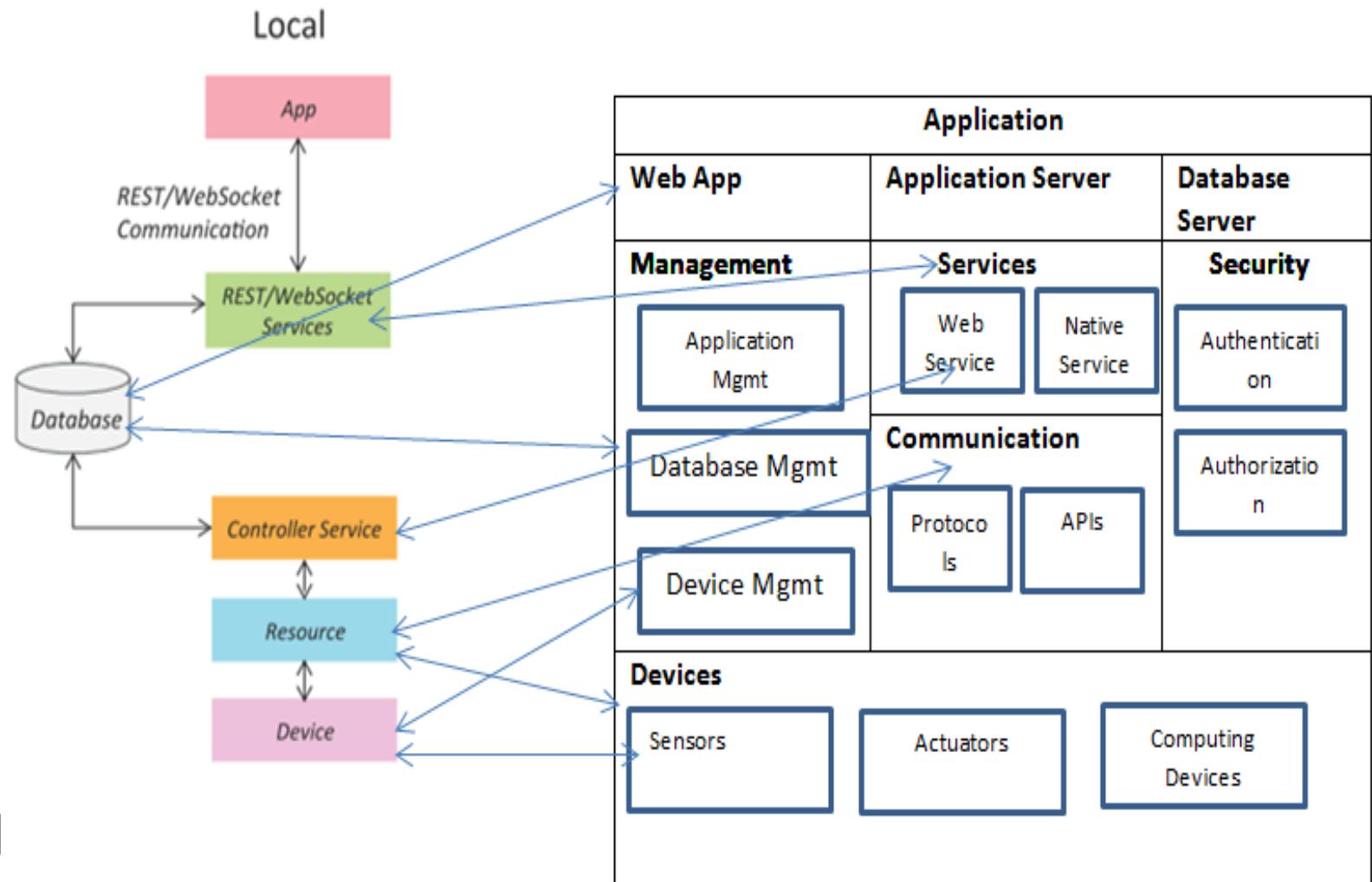
Step 7 : Functional View

Specification...Drawing Functional Groups



Step 7 : Functional View Specification

...Deployment level to Functional Group Mapping



Step 8 : Operational View Specification

- Define the Operations/options related to IoT System development
- Such as Device options, Storage options, Application hosting option

Step 8 : Operational View Specification of automated irrigation system

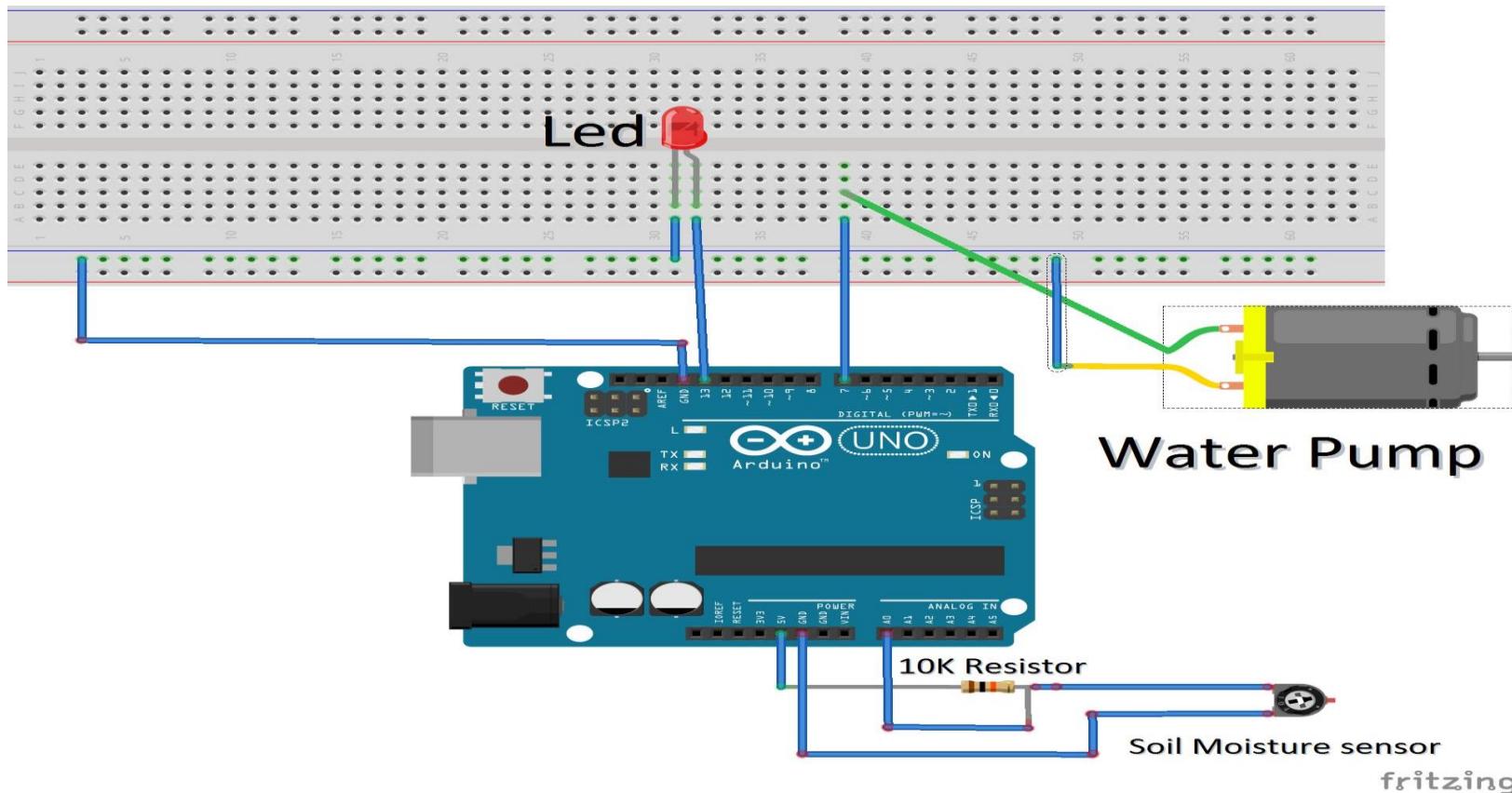
- Application
 - Web App : PhP WebApp
 - Application Server : Google App engine
 - Database Server : MySQL
- Services
 - Native : Controller Service
 - Web : REST
- Communication
 - Communication APIs : REST APIs
 - Communication Protocol :
 - Link Layer: 802.11
 - N/w : IPV6
 - Transport : TCP
 - Application : HTTP

Step 8 : Operational View Specification of automated irrigation system

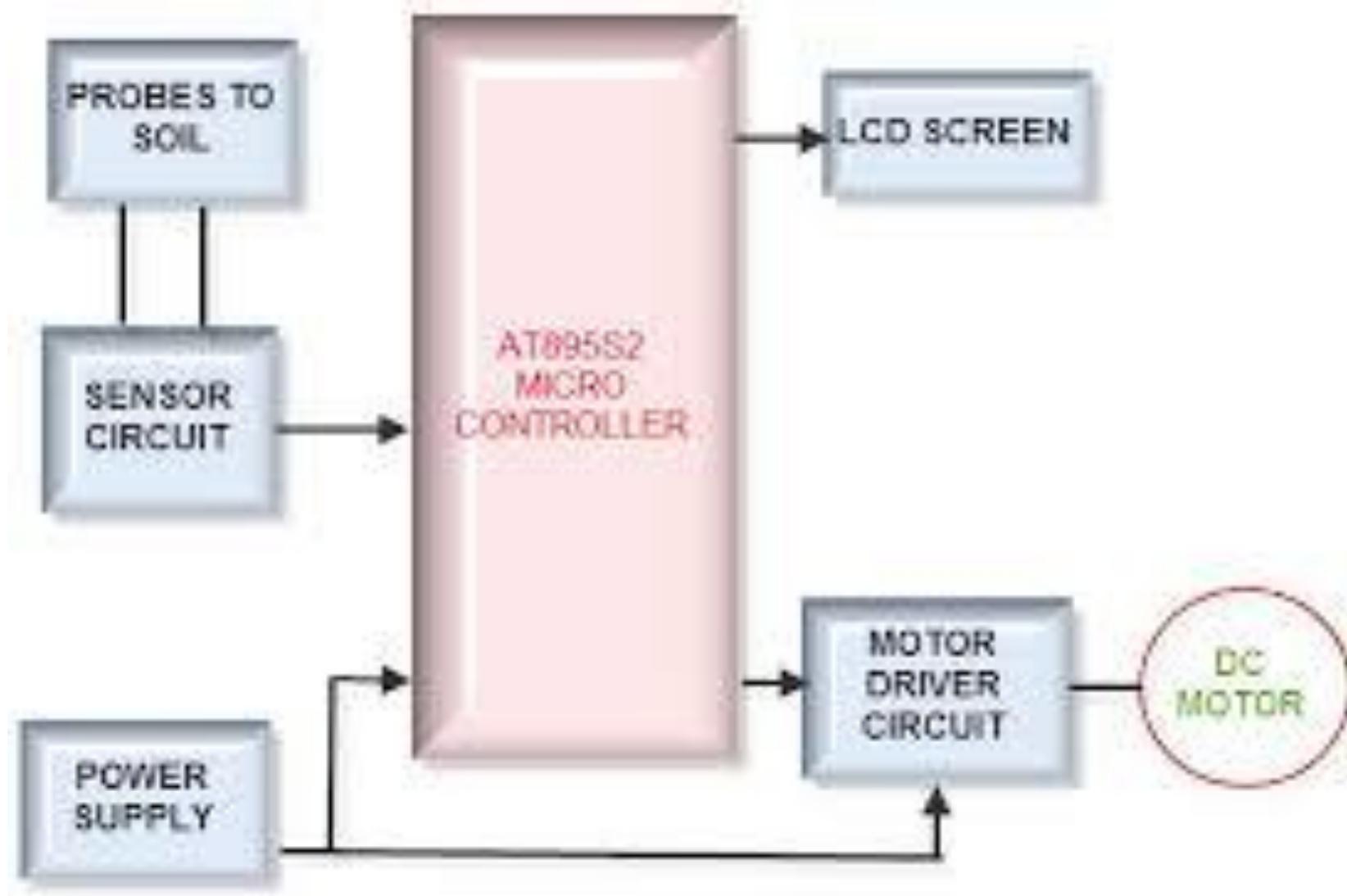
- Management
 - Device Management: Arduino device management
 - Application Management : PHP App Management
 - Database Management: MySQL Db Mgmt
- Security
 - Login Management

Step 9 : Device and Component Integration

- Integrates the devices and components and draw a schematic diagram showing the same

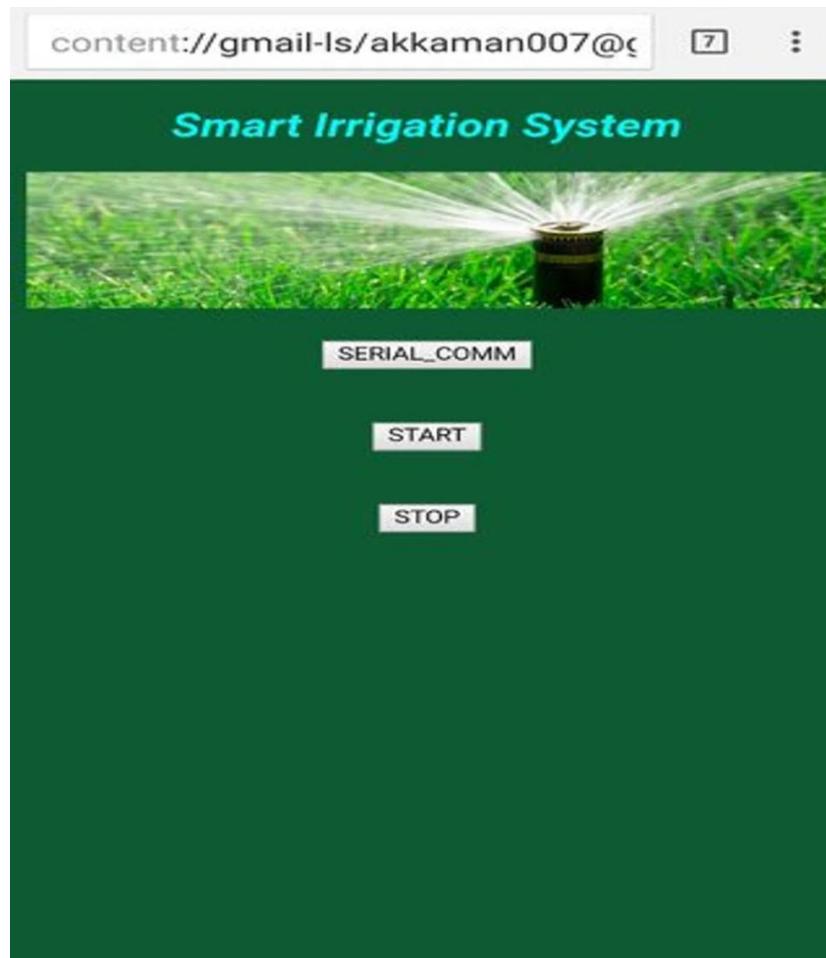


Device and Component Integration : Alternate Diagram



Step 10 : Application development

- GUI / Screenshot of IoT Application



IoT Health Monitoring Application(HMA)

Purpose and Requirement Specification

- Purpose of the Project:** The Health Monitoring System is basically used to monitor the Patient Body Temperature and Pulse by respective sensors. This information is captured and stored, so that the authorized personnel can view and analyze the Data remotely at any time.

IoT Health Monitoring Application(HMA)

Purpose and Requirement Specification

- **Behavior:** The Monitoring is done in real time to identify the state of the patient. This information can be used to analyze the state of a patient or to get sensitive data in order to be sequentially captured for medical diagnosis. The system Alarms in Emergency Situation and notifies the staff and relatives of the patient.

IoT Health Monitoring Application(HMA)

Purpose and Requirement Specification

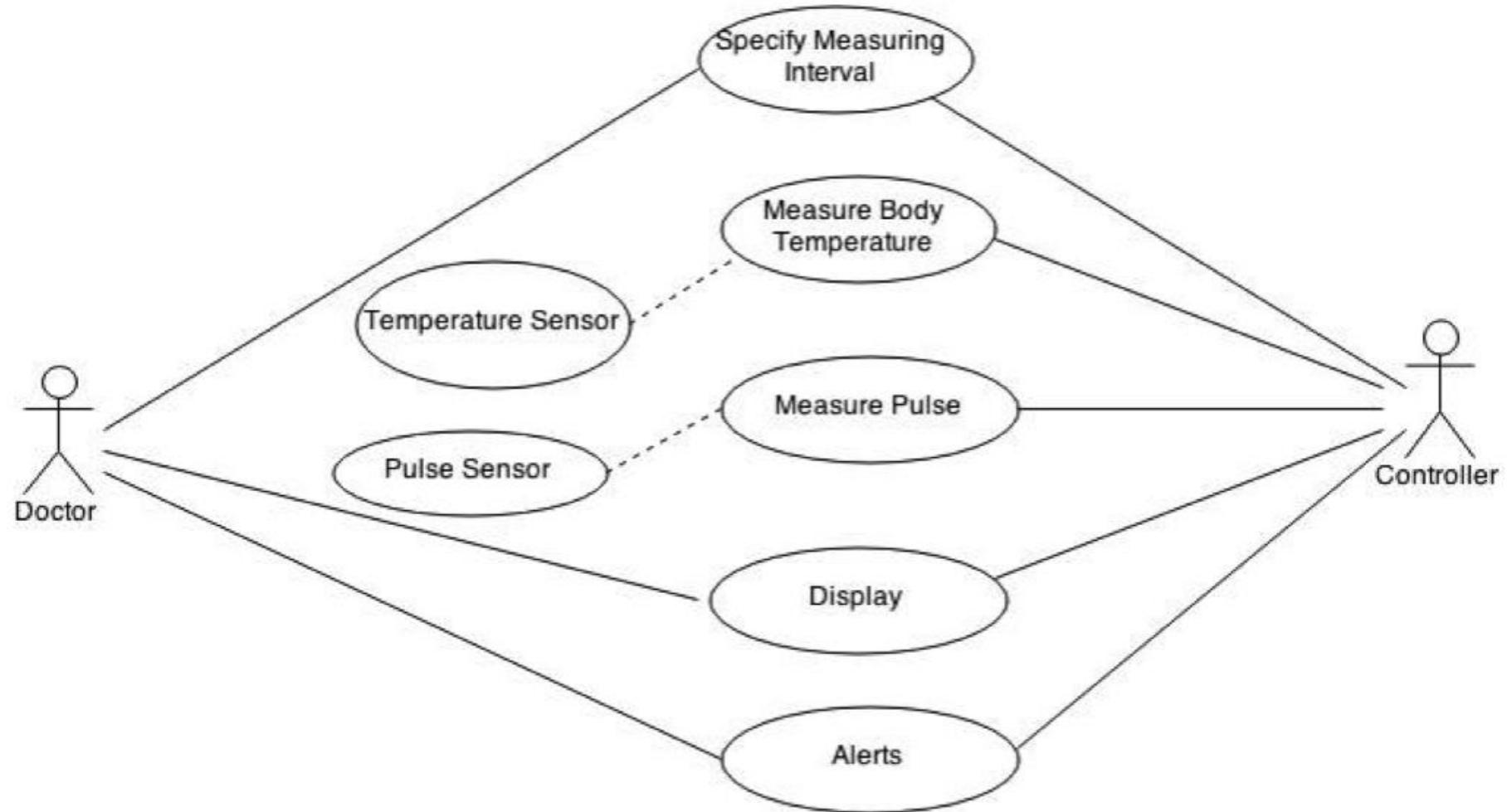
- **System Management Requirement:** The system provides remote monitoring and control functions
- **Data Analysis Requirement:** System allows the analysis of data and can be visualized in graphical format.
- **Application Deployment Requirement:** Application will be deployed locally on the device, but can be monitored remotely.
- **System Requirements:** Only Authorized users can access and control the Application

IoT Health Monitoring Application(HMA) Process Specification

- **The Use Case Diagram describes the use case's of the system and the actors involved.**
- **The Process diagram shows the steps involved in the process.**

The sensors read the information from the Human Body and stores it in Database, when the values go beyond the threshold limit it sends alerts.

IoT Health Monitoring Application(HMA) Process Specification

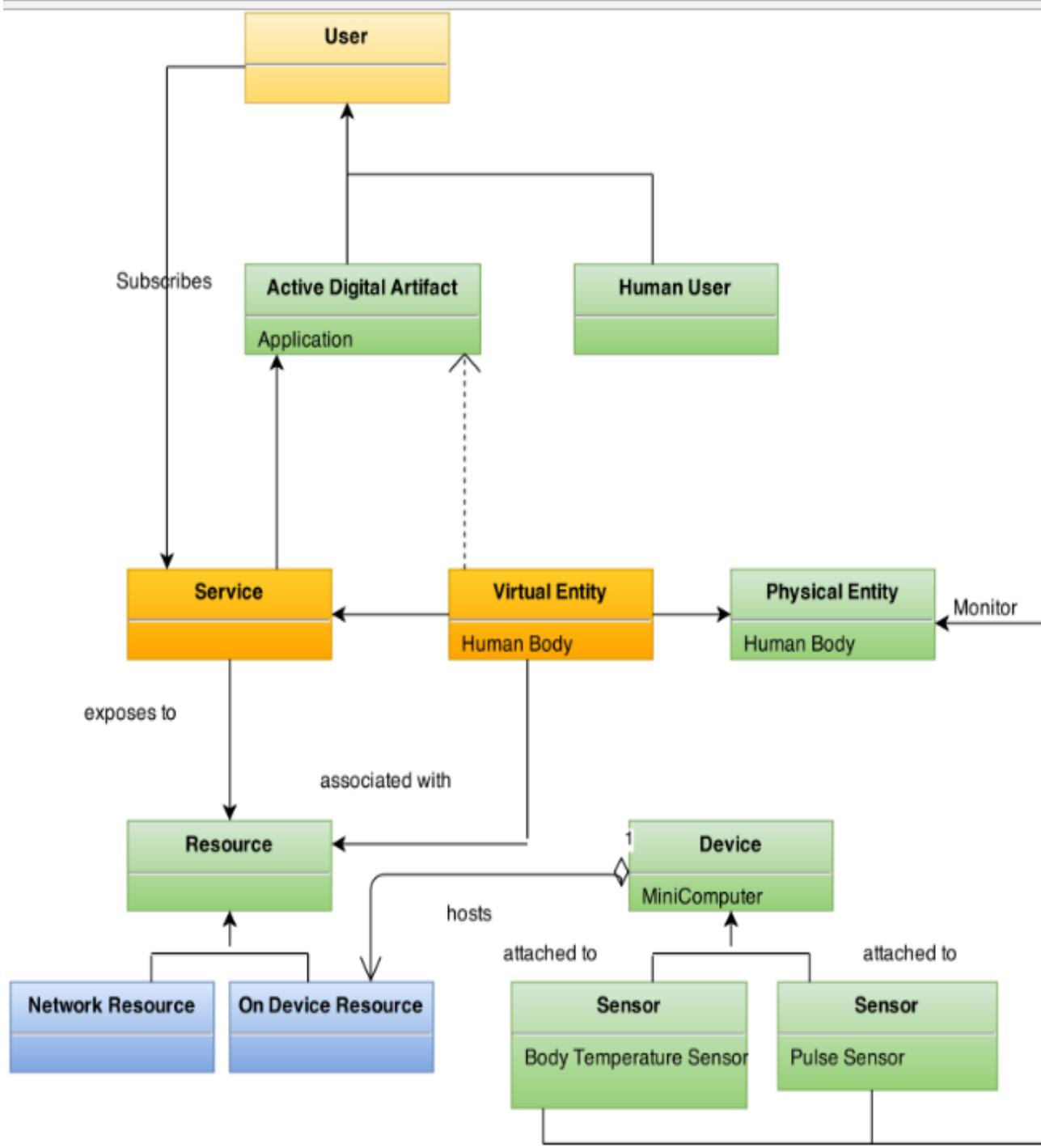


IoT Health Monitoring Application(HMA) Domain Specification

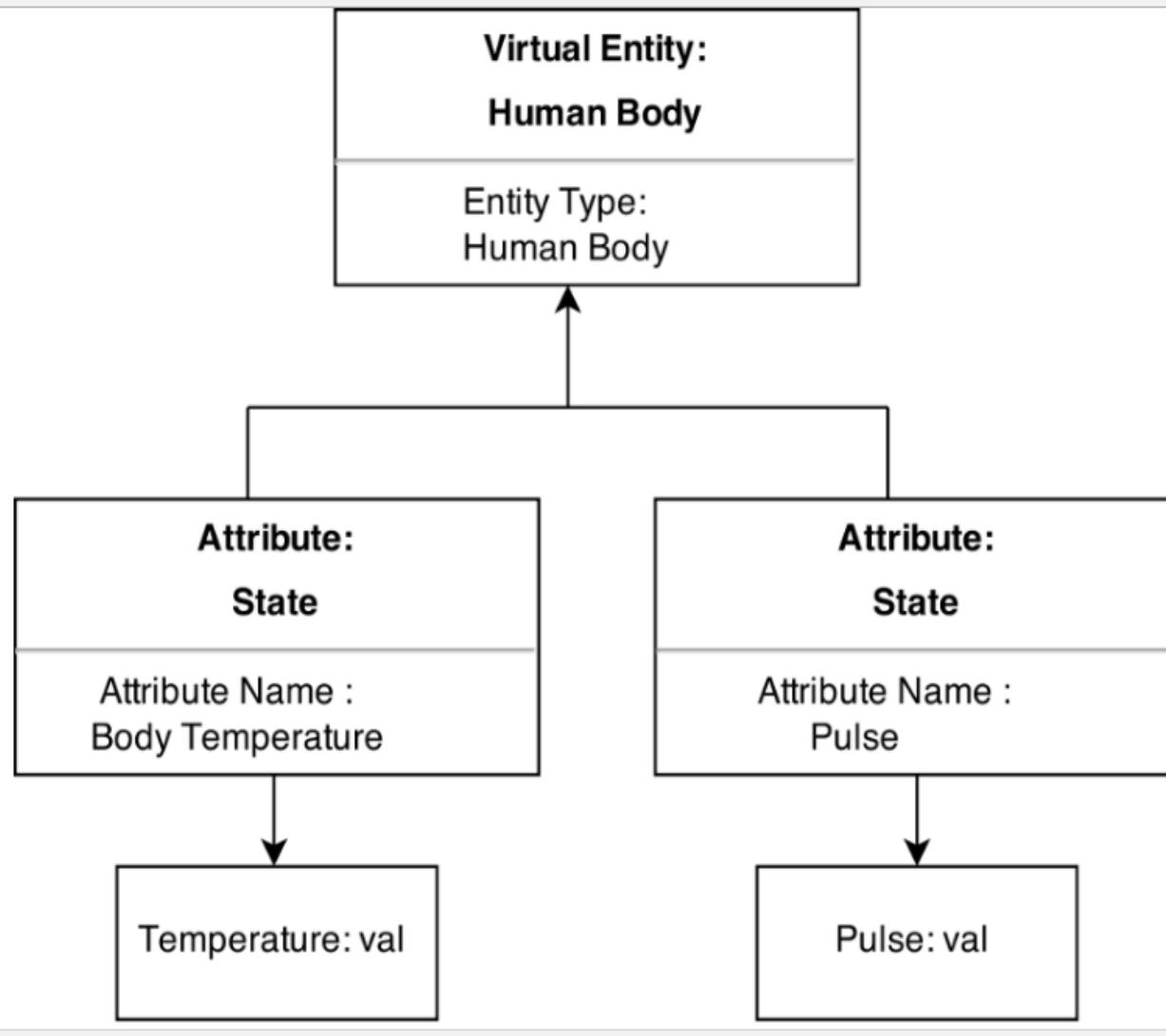
- **Physical Entity :** In HMA, human body is the physical entity where body temperature and pulse are monitored using respective sensors.
- **Virtual Entity :** It's a representation of Physical Entity in Digital World. For each physical entity there exist one virtual entity in Domain Model.
- **Device:** Provides medium for interactions between Physical and Virtual Entities. In HMA, device is a single board (Arduino) which has temperature and pulse sensor attached to it.

IoT Health Monitoring Application(HMA) Domain Specification

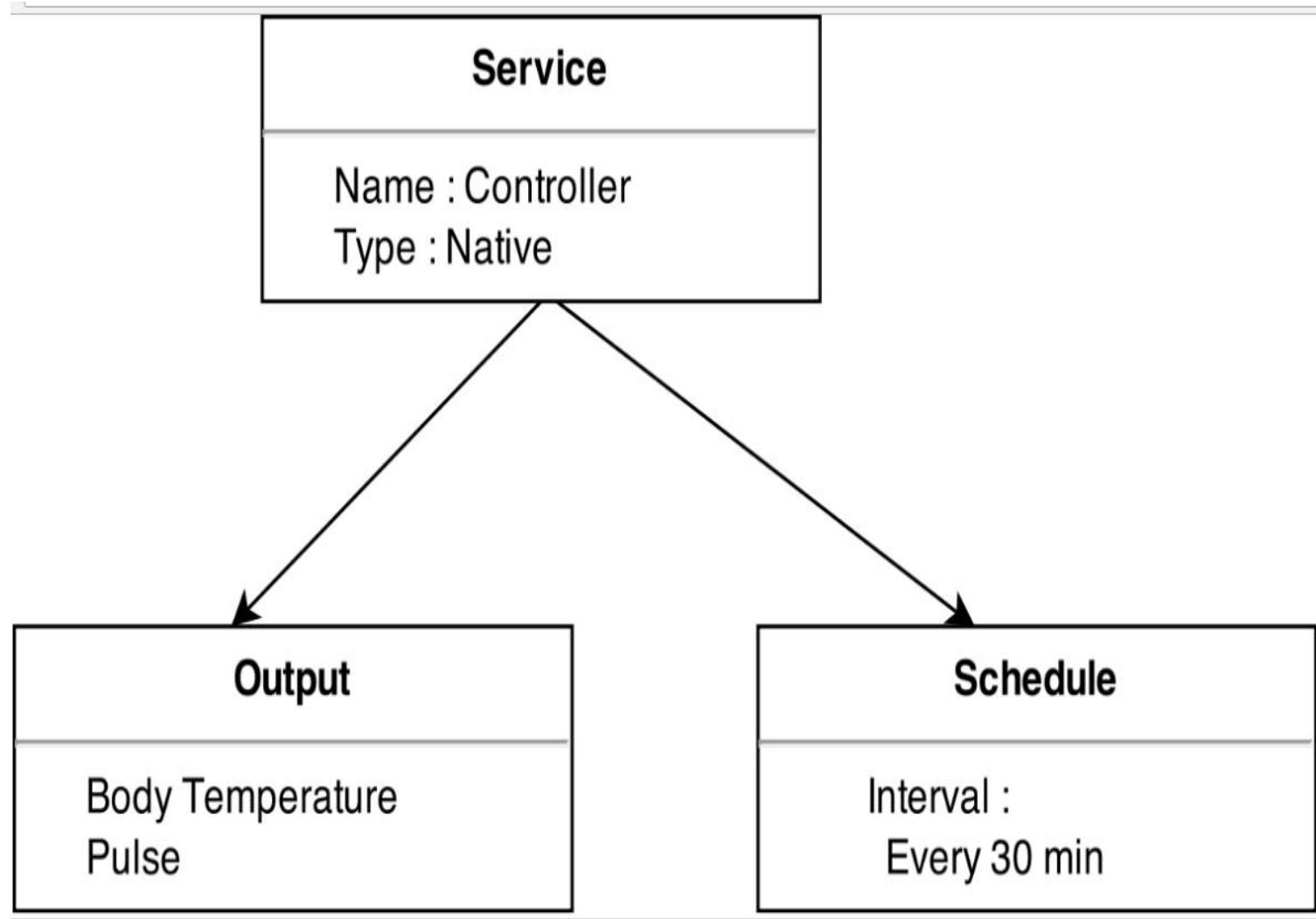
- **Resource** : Can be Software Components which can be either “on-device” or “network-resources”. On-Device resources are hosted on device and include software components that provide information about physical entity. Network Resources are software components on network such as Database.
- **Service** : In HMA, the services will be, service that retrieve current information, native service that runs on the device.



IoT Health Monitoring Application(HMA) Information Model Specification

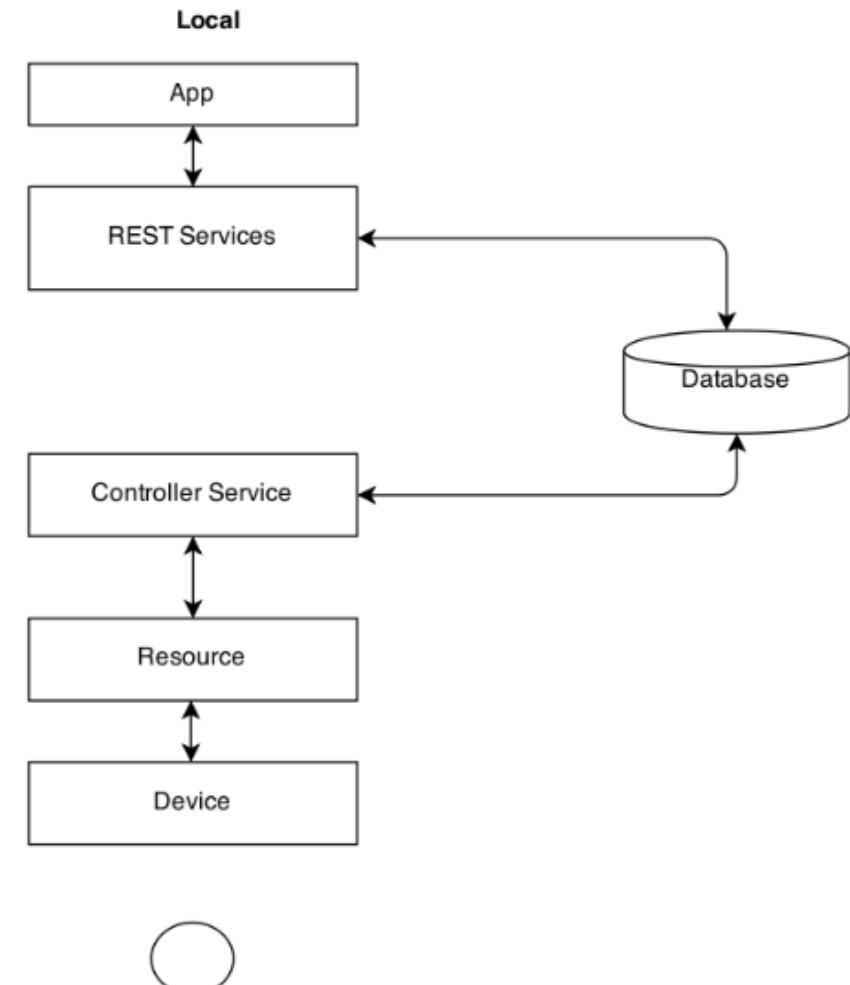


IoT Health Monitoring Application(HMA) Service Specification



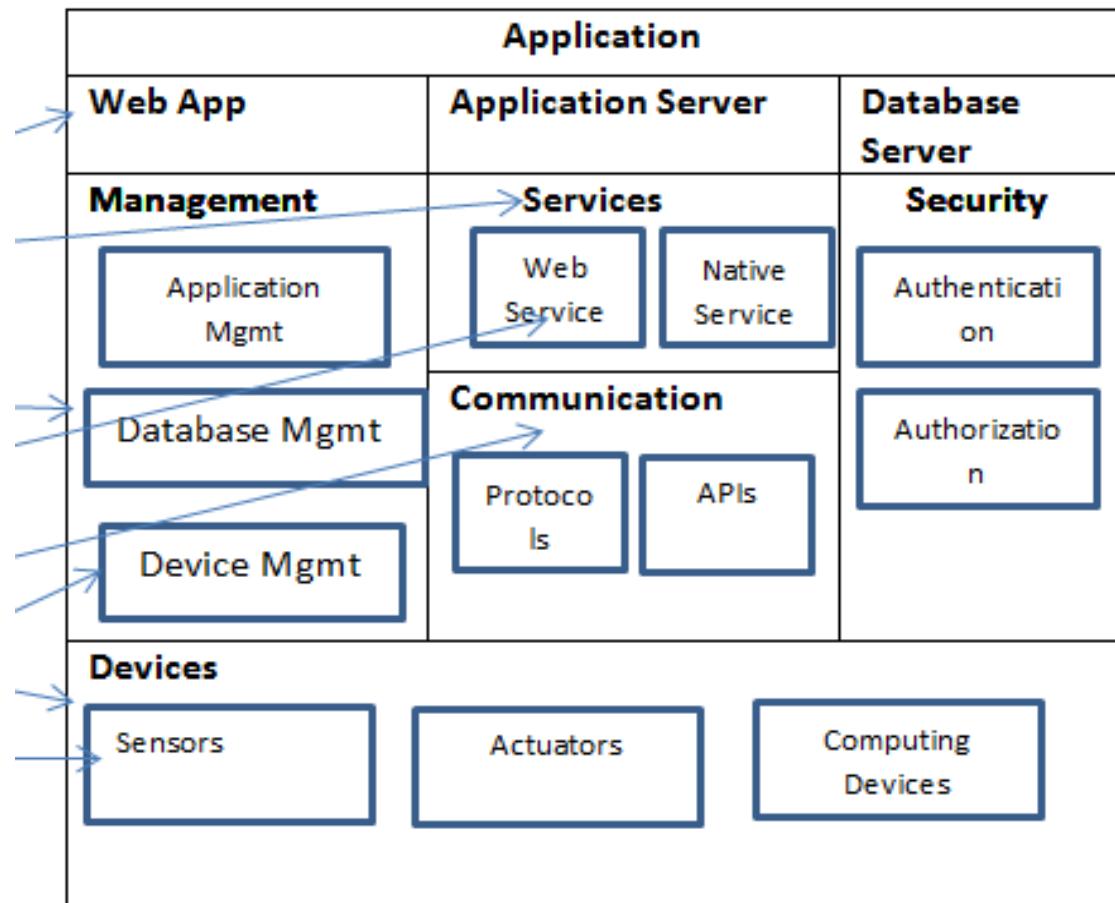
IoT Health Monitoring Application(HMA) IoT Level Specification

- The system has single device that performs sensing, stores data perform analysis and host the Application. Thus, IoT Level 1

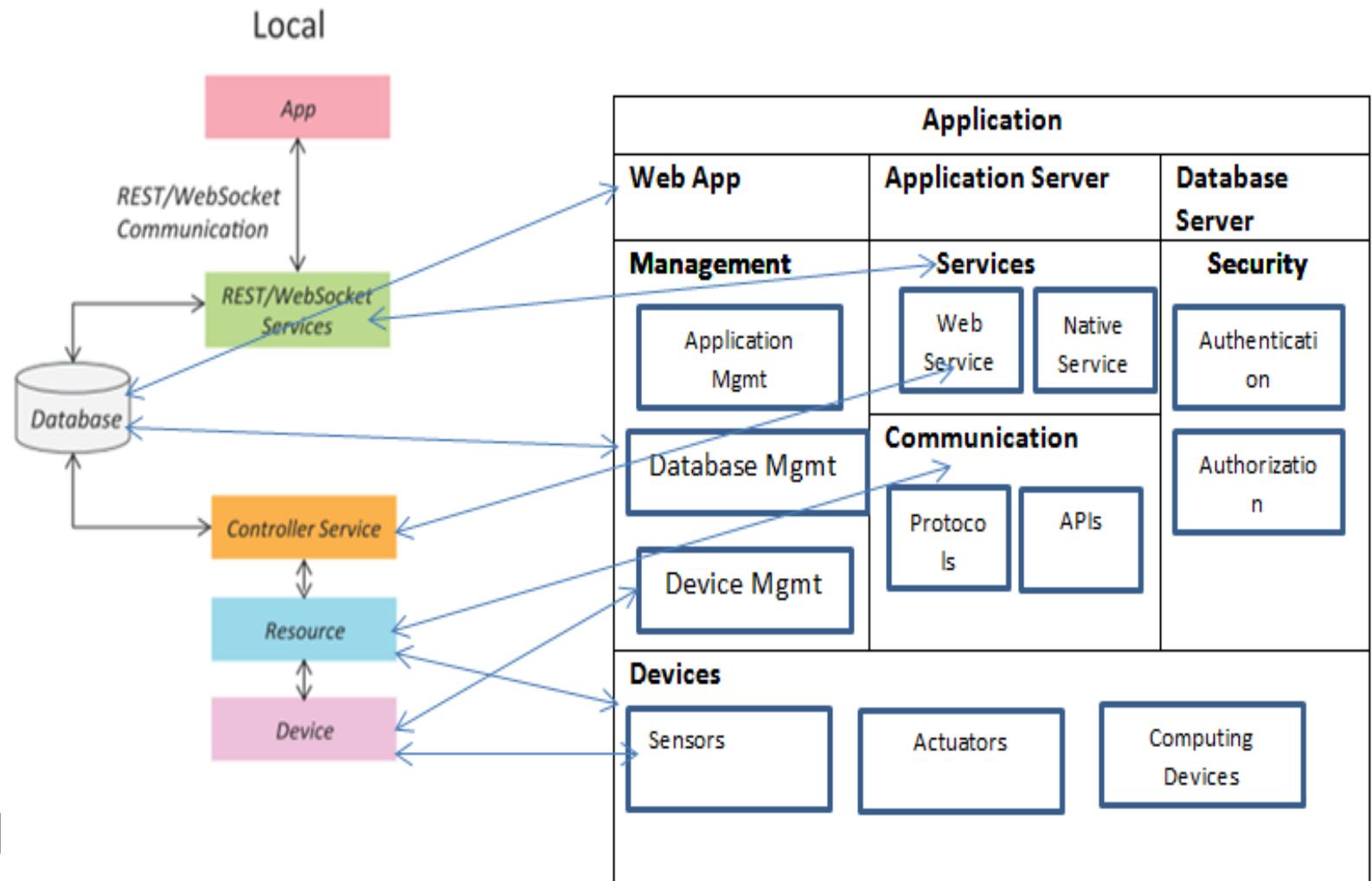


Monitoring Node
performs analysis, stores data

IoT Health Monitoring Application(HMA) Functional Specification



IoT Health Monitoring Application(HMA) Functional Specification



Step 8 : Operational View

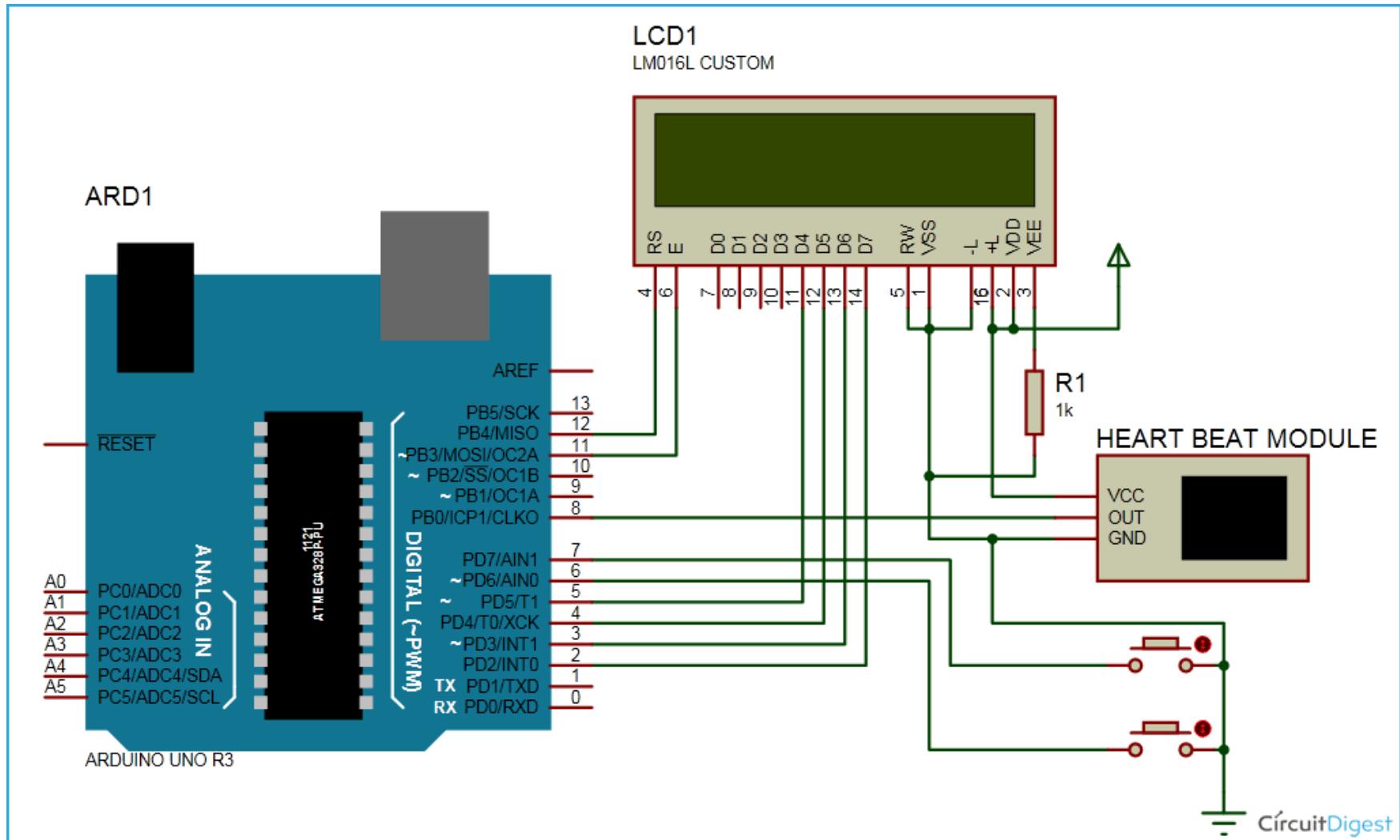
Specification of HMA

- Application
 - Web App : PhP WebApp
 - Application Server : Google App engine
 - Database Server : MySQL
- Services
 - Native : Controller Service
 - Web : REST
- Communication
 - Communication APIs : REST APIs
 - Communication Protocol :
 - Link Layer: 802.11
 - N/w : IPV6
 - Transport : TCP
 - Application : HTTP

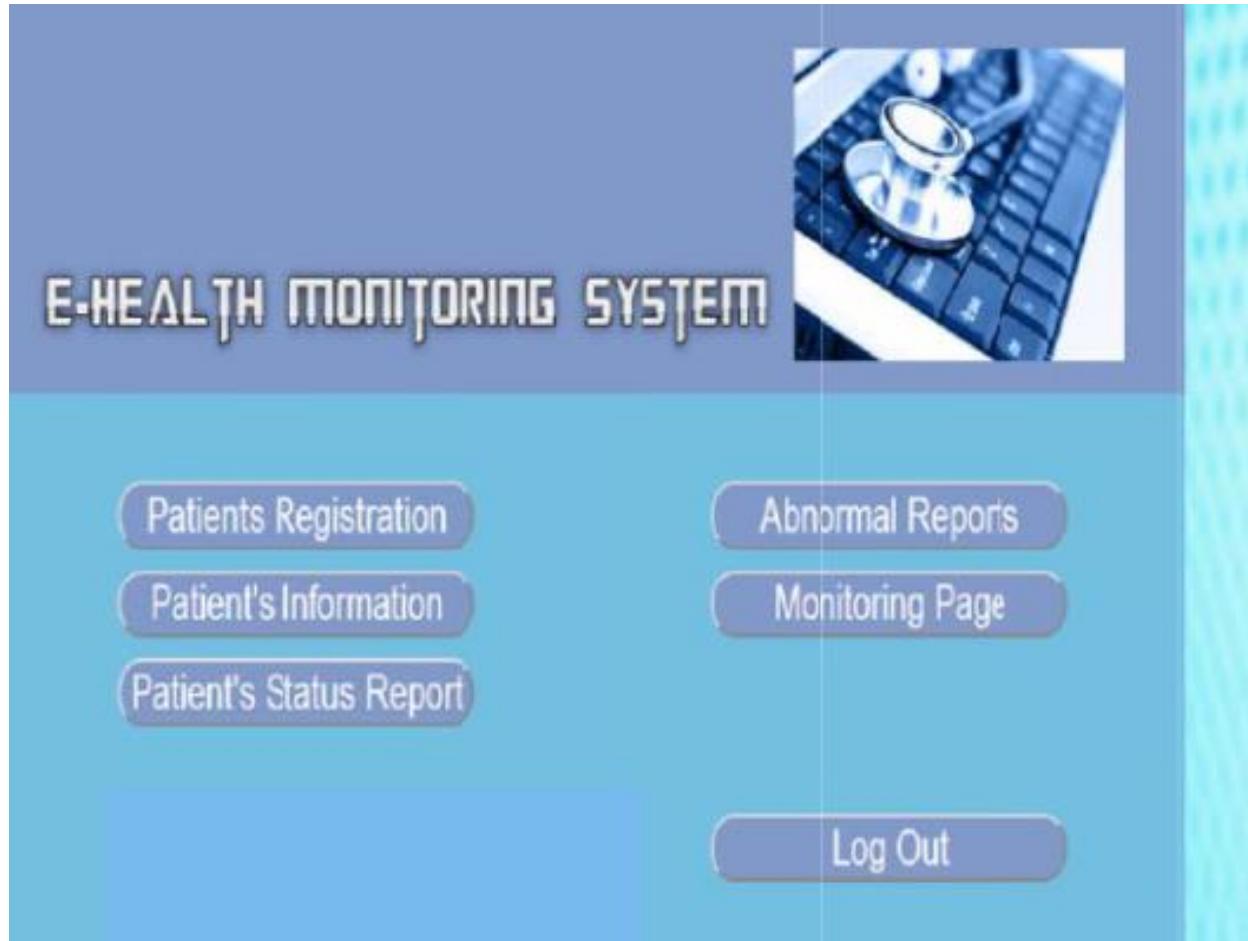
Step 8 : Operational View Specification of HMA

- Management
 - Device Management: Arduino device management
 - Application Management : PHP App Management
 - Database Management: MySQL Db Mgmt
- Security
 - Login Management

Step 9 : Device and Component Integration



Step 10 : Application Specification





Home Intrusion Detection System

HID

Purpose and Requirement Specification

- **Purpose of the Project:** The purpose of home intrusion detection system is to detect intrusions using sensors and raise alerts, if necessary.

HID

Purpose and Requirement Specification

- **Behavior:** In case of any intrusion, I intend to capture a picture of the intruder, mail the image to the respective end users and alert them. I would like to use an alarm which goes on in case of an intrusion.

HID

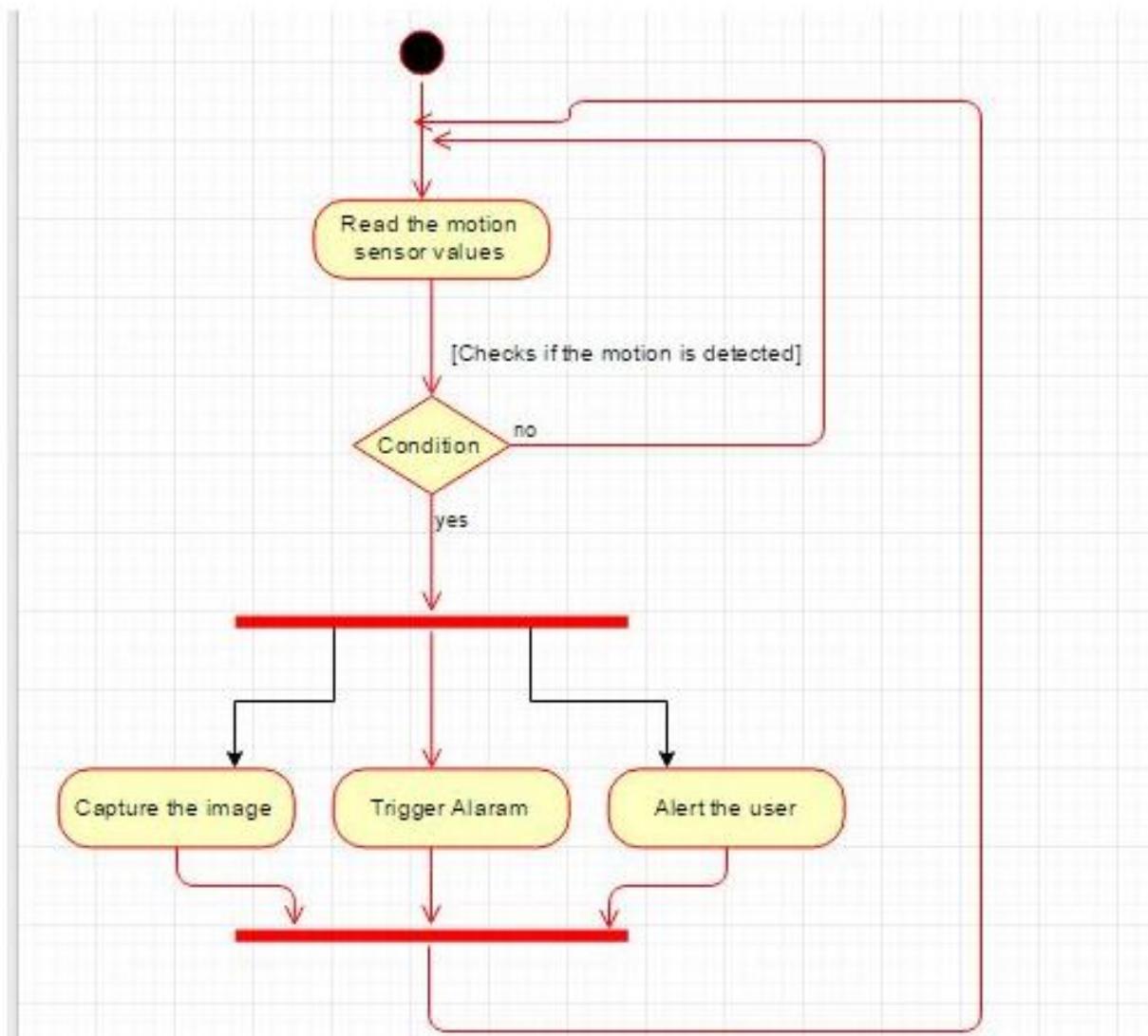
Purpose and Requirement Specification

- **Data Analysis Requirement:** The light dependent resistor data is analyzed locally. If the sensor reading falls below the threshold value, an image is captured and send as an alert to the user.
- **Application Deployment Requirement:** The application is deployed locally on the device and can be accessible from anywhere via node-red.
- **System Requirements:** Only Authorized users can access and control the Application

HID Process Specification

- **The Use Case Diagram describes the use case's of the system and the actors involved.**
- **The Process diagram shows the steps involved in the process.**
- With the help of Light dependent resistor and PIR motion sensor, I am going to detect the motions in the room.
- If a motion is detected, I intend to capture the image with the help of a webCam and store locally
- Now the alerts are sent to the user with the captured image.
- Also I am using the buzzer which turns on in case of any intrusion.

HID Process Specification

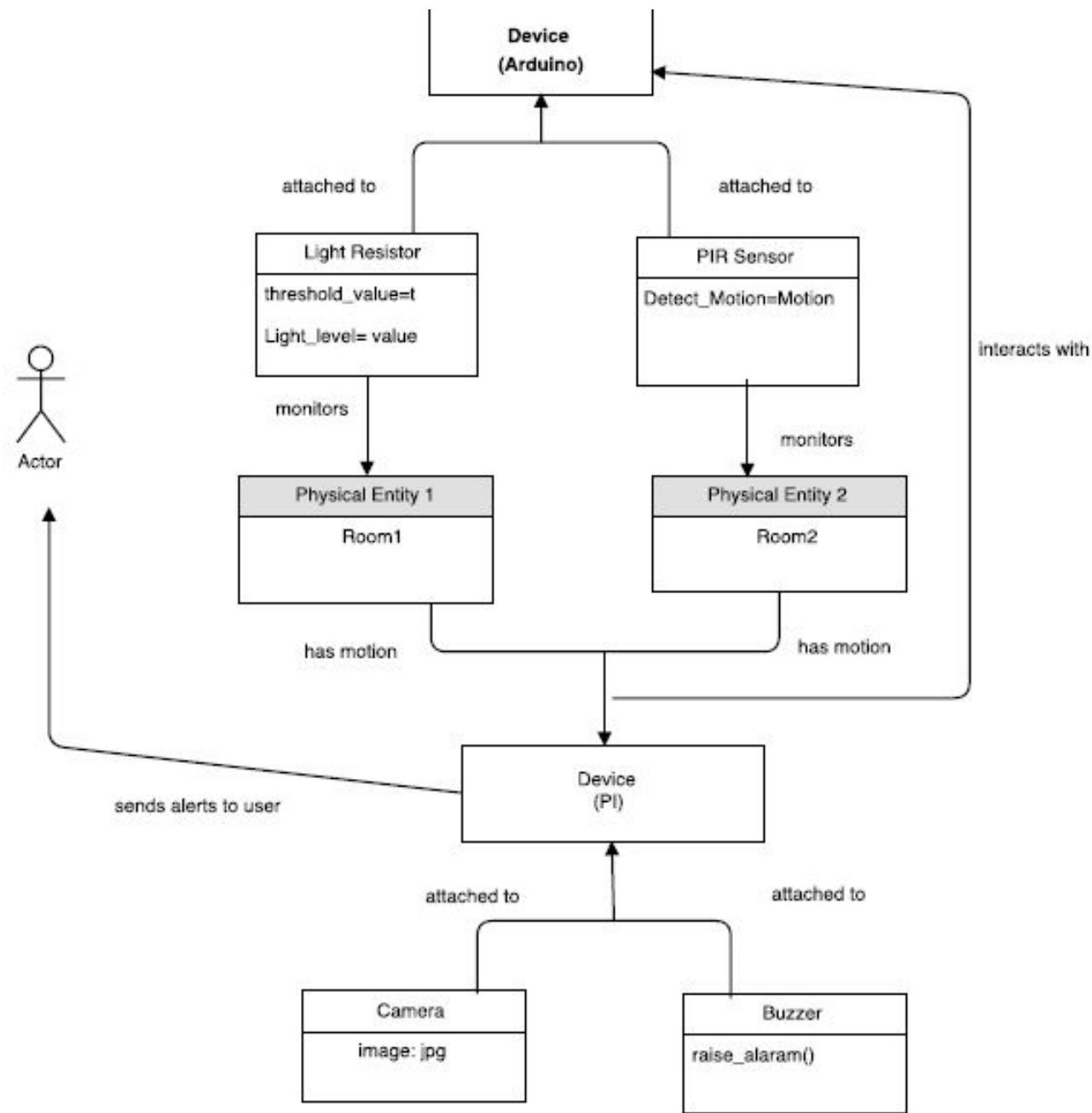


HID Process Specification

- From the above process diagram, we can see that the process starts at the circle. The PIR motion sensor which is shown in the rectangular box in the above figure, is placed in a room. It detects the motion in the room. If there is a motion then an image is captured and an alert will be sent to the user. And also it turns on the buzzer. This decision (yes/no) to raise an alert is shown in the diamond box in the above figure.

HID Domain Specification

- The domain model describes the main concepts, entities and objects in the domain. The domain includes physical entities **for rooms**. The **devices in my system** are **nodemcu** to which PIR and light dependent resistor is attached and **raspberry pi** to which **camera** and **buzzer** are attached to send alerts.



IoT level Specification

- implement my IOT system that is “**Home intrusion detection system**” in *level 5*. The IOT level 5 system has multiple nodes and one coordinator node. Coordinator node collects the data from the end nodes and sends to the cloud.

Home Intrusion detection System- IOT Level5

