UDHAYAS SPNC
PUBLICATION

Internet of Things

Author 1

Dr T R Nisha Dayana

Assistant Professor
Department of
Computer Science & IT
VISTAS

Author 2

Dr S.PRATHIBA

Assistant Professor,
Department of
Advanced Computing
and Analytics,
VISTAS.

Author 3

Ms. Selin Chandra C S

Assistant Professor
Department of
Computer Science & IT
VISTAS

**UDHAYAS
SPNC
PUBLICATION**

2155, Main Road ,Ayyapakkam, Chennai-77

spncpublication@gmail.com/
udhyasbooks@gmail.com
www.udhayas.in

Internet of Things

By :

Author1:
Dr. T R Nisha Dayana
Assistant Professor
Department of Computer Science & IT
VISTAS

Author2:
Dr S.PRATHIBA
Assistant Professor,
Department of Advanced Computing and Analytics,
VISTAS.

Author 3:
Ms. Selin Chandra C S
Assistant Professor
Department of Computer Science & IT
VISTAS

UDHAYAS
SPNC PUBLICATION
2155, 4TH MAIN ROAD,
AYYAPAKKAM, CHENNAI – 600 077.
Cell : 9003124064 / Websit: www.udhayas.in
email : spncpublication@gmail.com

Internet of Things

By

Author1:
Dr. T R Nisha Dayana
Assistant Professor
Department of Computer Science & IT
VISTAS

Author2:
Dr S.PRATHIBA
Assistant Professor,
Department of Advanced Computing and Analytics,
VISTAS.

Author 3:
Ms. Selin Chandra C S
Assistant Professor
Department of Computer Science & IT
VISTAS

Publication:
December 2025

Pages : 110

ISBN: 978-93-49030-01-5

Published By:

Price : Rs.300/-

UDHAYAS
SPNC PUBLICATION
2155, 4TH MAIN ROAD,
AYYAPAKKAM, CHENNAI – 600 077.
Cell : 9003124064 /
Website: www.udhayas.in
E**mail** : spncpublication@gmail.com

Printed At : Udhayas Print, Chennai.

INTERNET OF THINGS

INTERNET OF THINGS

UNIT I FUNDAMENTALS OF IOT 9

Introduction - Definition and Characteristics of IoT - Physical design - IoT Protocols - Logical design - IoT communication models, IoT Communication APIs - Enabling technologies - Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems, IoT Levels and Templates - Domain specific IoTs - IoT Architectural view.

UNIT II ELEMENTS OF IOT 9

IoT and M2M- difference between IoT and M2M - Software Defined Networks - Network Function Virtualization - IoT systems management – Needs - NETCONF, YANG - IoT design methodology.

UNIT III IOT PROTOCOLS 9

Sensors and actuators - Communication modules – Zigbee - LoRa - RFID - Wi-Fi - Power sources.

UNIT IV BUILDING IoT WITH CLOUD AND DATA ANALYTICS 9

IoT platforms – Arduino – Raspberry Pi - Cloud Computing in IoT - Cloud Connectivity - Big Data Analytics - Data Visualization

UNIT V CHALLENGES IN IOT AND CASE STUDIES 9

Security Concerns and Challenges - Real time applications of IoT – Home automation – Automatic lighting – Home intrusion detection – Cities – Smart parking – Environment – Weather monitoring system – Agriculture – Smart irrigation.

COURSE OUTCOMES:

Upon completion of the course, the student should be able to:

- Describe the characteristics, physical and logical designs, domains and architecture.
- Differentiate M2M and IoT, SDN and NFV design methodologies

TOTAL: 45 PERIODS

TEXT BOOKS

1. Arshdeep Bahga, Vijay Madisetti, "Internet of Things-A hands-on approach", Universities Press, 2015
2. Olivier Hersent, David Boswarthick, Omar Elloumi, “The Internet of Things: Key applications and Protocols”, Wiley Publications 2nd edition , 2013

REFERENCES BOOKS

1. Raj Kamal, “Internet of Things – Architecture and Design Principles”, Mc Graw Hill Education Pvt. Ltd., 2017
2. Internet of Things and Data Analytics, Hwaiyu Geng, P.E, Wiley Publications, 2017
3. Manoel Carlos Ramon, ―Intel® Galileo and Intel® Galileo Gen 2: API Features and Arduino Projects for Linux Programmers‖, Apress, 2014
4. Marco Schwartz, ―Internet of Things with the Arduino Yun‖, Packt Publishing, 2014
5. Adrian McEwen, Hakim Cassimally, “Designing the Internet of Things”, Wiley Publications,2012.

UNIT I **FUNDAMENTALS OF IOT**

Introduction - Definition and Characteristics of IoT - Physical design - IoT Protocols - Logical design - IoT communication models, IoT Communication APIs - Enabling technologies - Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems, IoT Levels and Templates - Domain specific IoTs - IoT Architectural view.

1.1 Introduction

- The Internet of Things represents the whole way from collecting data, processing it, taking an action corresponding to the signification of this data to storing everything in the cloud. All this is made possible by the internet
- The Internet of things has become a very widely spread concept in the last few years. The reason for this is mainly the need to computerize and control most of the surrounding objects and have access to data in real time.
- Example: Parking sensors, about phones which can check the weather and so on

1.1.1 Definition & Characteristics of IoT Definition:

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

Characteristics of IoT

i)**Dynamic & Self Adapting**:

IoT devices and systems may have the capability to dynamically adapt with the changing contexts and take actions based on their operating conditions, user‘s context or sensed environment.

Eg: The surveillance system comprising of a number of surveillance cameras. The surveillance camera can adapt modes based on whether it is day or night. The surveillance system is adapting itself based on context and changing conditions. ii)**Self Configuring:**

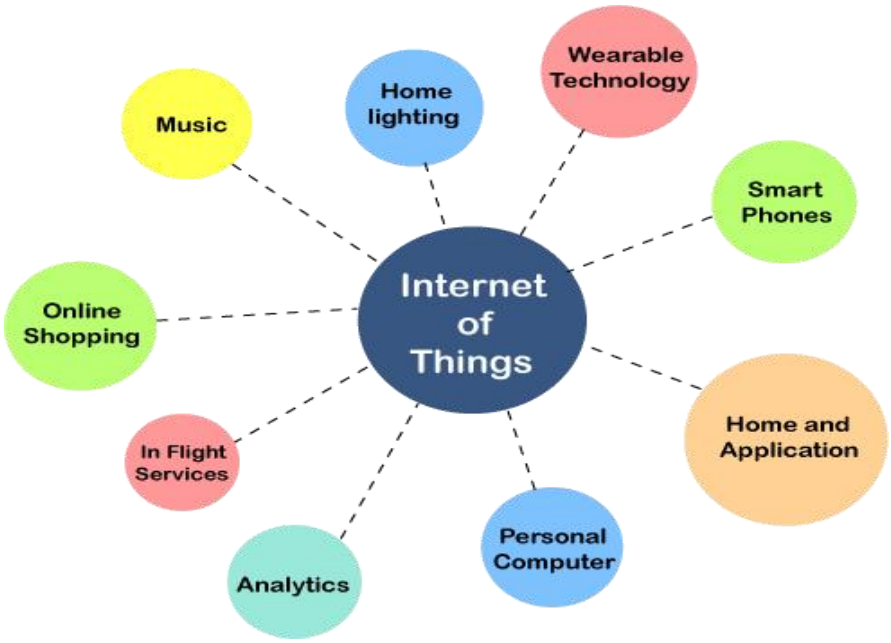
IOT devices have self configuring capability,allowing a large number of devices to work together to provide certain functionality. These devices have the ability configure themselves setup networking, and fetch latest software upgrades with minimal manual or user interaction.

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

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

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

Applications of IoT:



1) Home
2) Cities
3) Environment
4) Energy
5) Retail
6) Logistics
7) Agriculture
8) Industry
9) Health &LifeStyle

Physical Design of IoT :

The "Things" in IoT usually refers to IoT devices which have unique identities and can perform remote sensing, actuating and monitoring capabilities.

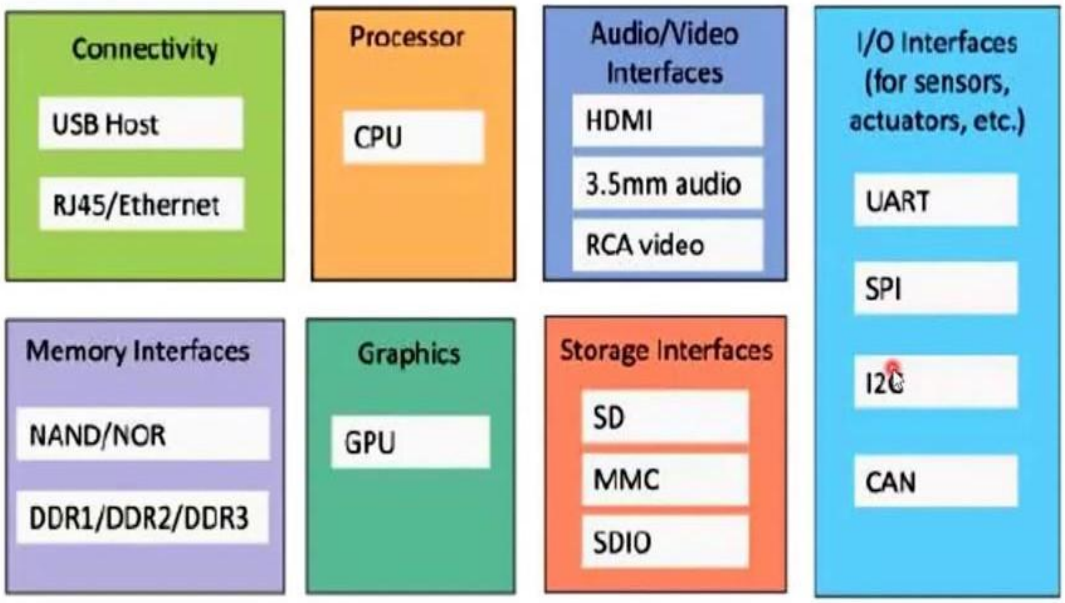
IoT devices can:

- Exchange data with other connected devices and applications (directly or indirectly), or
- Collect data from other devices and process the data locally or
- Send the data to centralized servers or cloud-based application back-ends for processing the data,
- Perform some tasks locally and other tasks within the IoT infrastructure, based on temporal and space constraints

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.

Generic Block Diagram of IoT Device



- HDMI: High definition multimedia Interface.
- 3.5mm: Audio Jack which headphone adapter.
- RCA: Radio corporation of America.
- UART: Universal Asynchronous Receiver Transmitter.
- SPI: Serial Peripheral Interface.
- I2C: Inter integrated circuit
- CAN: Controller Area Network used for Micro-controllers and devices to communicate.
- SD: Secure digital (memory card)
- MMC: multimedia card
- SDIO: Secure digital Input Output
- GPU: Graphics processing unit.
- DDR: Double data rate

IoT Protocols:

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

Protocols:

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

b) Network/Internet Layer:

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

Protocols:

- IPv4: Internet Protocol version4 is used to identify the devices on a n/w using a hierarchical addressing scheme. 32 bit address. Allows total of 2*32addresses.
- IPv6: Internet Protocol version6 uses 128 bit address scheme and allows 2*128 addresses.
- 6LOWPAN:(IPv6 over Low power Wireless Personal Area Network) operates in 2.4 GHz frequency range and data transfer 250 kb/s.

c) Transport Layer:

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

Protocols:

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

d) Application Layer:

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

Protocols:

- HTTP: Hyper Text Transfer Protocol that forms foundation of WWW. Follow requestresponse model

Stateless protocol.

- CoAP: Constrained Application Protocol for machine-to-machine(M2M) applications with constrained devices, constrained environment and constrained n/w. Uses client-server architecture.
- WebSocket: allows full duplex communication over a single socket connection.
- MQTT: Message Queue Telemetry Transport is light weight messaging protocol based on publish-subscribe model. Uses client server architecture. Well suited for constrained environment.
- XMPP: Extensible Message and Presence Protocol for real time communication and streaming XML data between network entities. Support client-server and server-server communication.
- DDS: Data Distribution Service is data centric middleware standards for device-to-device or machine-to-machine communication. Uses publish-subscribe model.
- AMQP: Advanced Message Queuing Protocol is open application layer protocol for business messaging. Supports both point-to-point and publish-subscribe model.

LOGICAL DESIGN of IoT

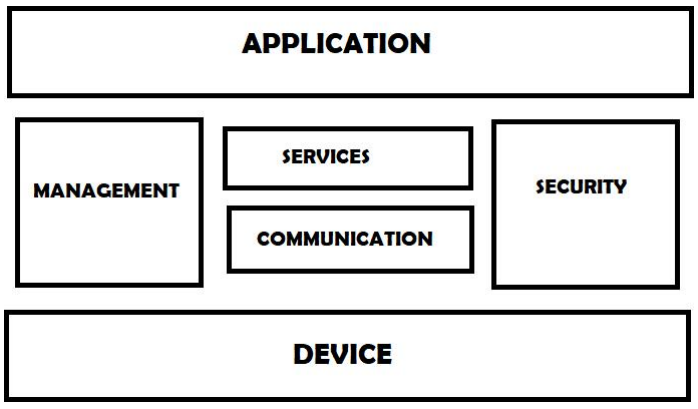
Refers to an abstract represent of entities and processes without going into the low level specifies of implementation.

1) IoT Functional Blocks
2) IoT Communication Models
3) IoT Comm. APIs

1) IoT Functional Blocks:

Provide the system the capabilities for identification, sensing, actuation, communication and management

- Device: An IoT system comprises of devices that provide sensing, actuation, monitoring and control functions.
- Communication: handles the communication for IoT system.
- Services: for device monitoring, device control services, data publishing services and services for device discovery.
- Management: Provides various functions to govern the IoT system.
- Security: Secures IoT system and priority functions such as authentication, authorization, message and context integrity and data security.
- Application: IoT application provide an interface that the users can use to control and monitor various aspects of IoT system.



2) **IoT Communication Models:**

A) Request-Response

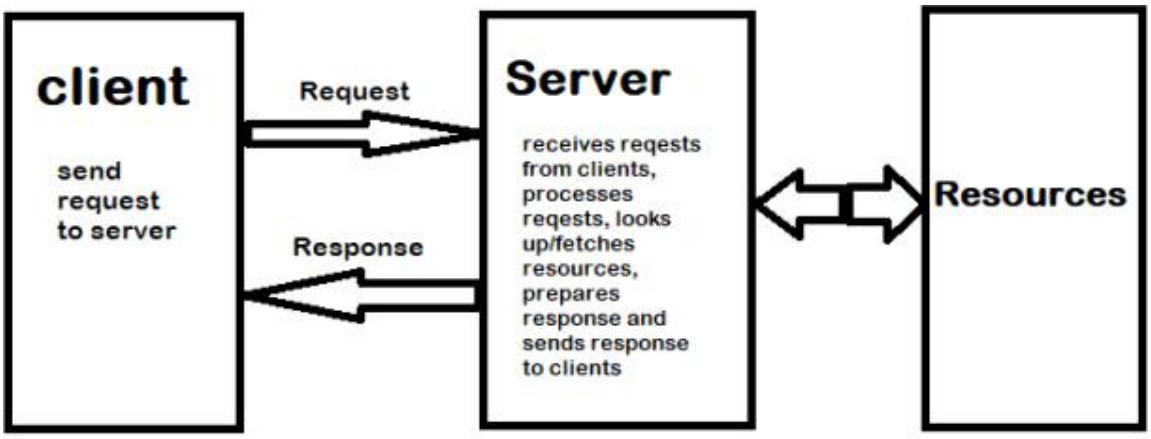
B) Publish-Subscribe

C)Push-Pull

D) Exclusive Pair

A) Request-Response

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.



Request-Response Communication Model

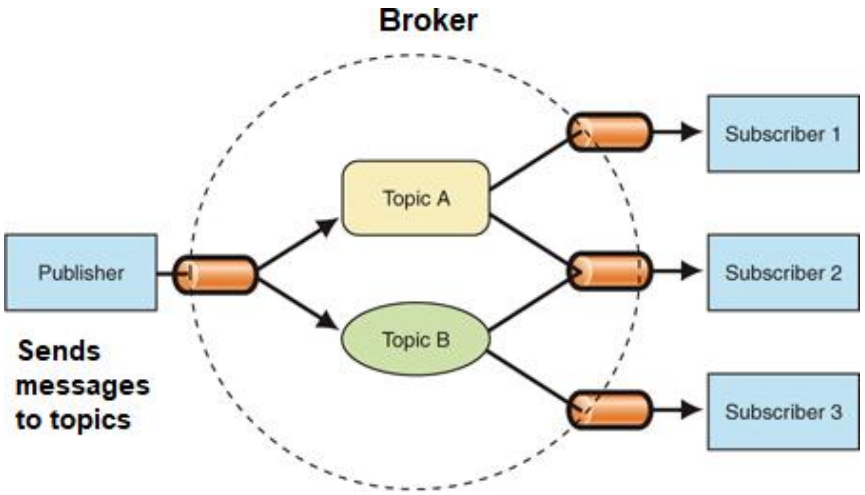
B) Publish-Subscribe communication model:

a. Publish-Subscribe is a communication model that involves publishers, brokers and consumers.

b. 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.

c. Consumers subscribe to the topics which are managed by the broker.

d. When the broker receives data for a topic from the publisher, it sends the data to all the subscribed consumers

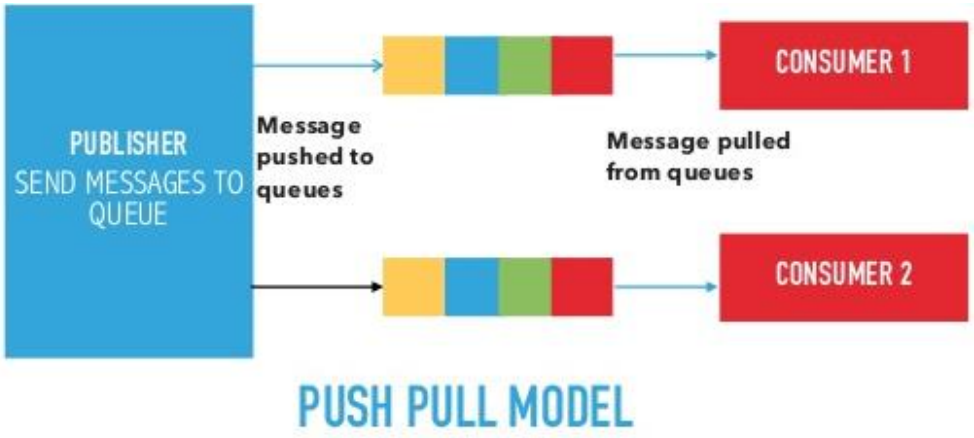


C) Push-Pull communication model:

a. 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.

b. Queues help in decoupling the messaging between the producers and consumers.

c. 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.



PUSH PULL MODEL

D) Exclusive Pair communication model:

a. Exclusive Pair is a bidirectional, fully duplex communication model that uses a persistent connection between the client and server.

b. Once the connection is setup it remains open until the client sends a request to close the connection.

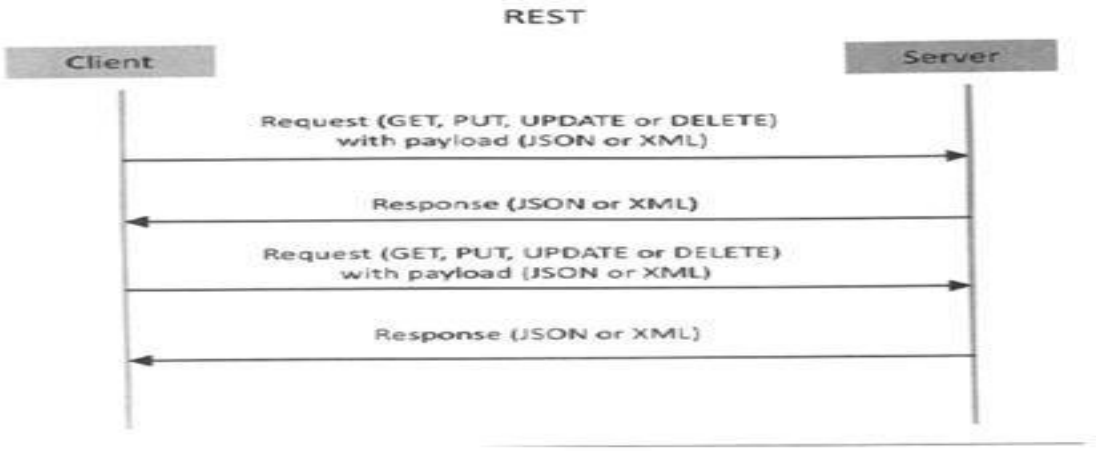
c. Client and server can send messages to each other after connection setup.

EXCLUSIVE PAIR COMMUNICATION MODEL

3) **IoT Communication APIs**: a) REST based communication APIs(Request-Response Based Model)
b) WebSocket based Communication APIs(Exclusive PairBasedModel)

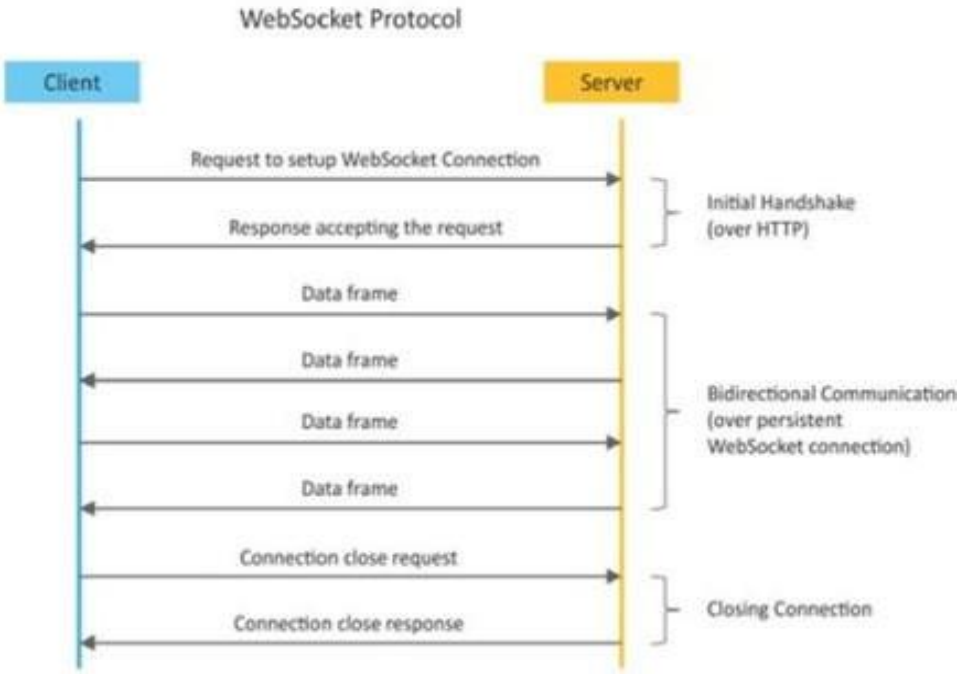
Request-Response model used by REST:

RESTful webservice is a collection of resources which are represented by URIs. RESTful web API has a base URI(e.g: http://example.com/api/tasks/). The clients and requests to these URIs using the methods defined by the HTTP protocol(e.g: GET, PUT, POST or DELETE). A RESTful web service can support various internet media types.



Request-response model used by REST

b) WebSocket Based Communication APIs: WebSocket APIs allow bi-directional, full duplex communication between clients and servers. WebSocket APIs follow the exclusive pair communication



model.

1.4 **IoT Enabling Technologies**

IoT is enabled by several technologies including Wireless Sensor Networks, Cloud Computing, Big Data

Analytics, Embedded Systems, Security Protocols and architectures, Communication Protocols, Web Services, Mobile internet and semantic search engines.

1.4.1 Wireless Sensor Networks

A wireless sensor network comprises of distributed devices with sensors which are used to monitor the environmental and physical conditions. A WSN consist of a number of end nodes and routers and a co-ordinator. The coordinator collects the data from all the nodes. Coordinator also acts as a gateway that connects the WSN to the internet.

WSNs used in IoT systems are described as follows:

- Weather Monitoring System: in which nodes collect temp, humidity and other data, which is aggregated and analyzed.
- Indoor air quality monitoring systems: to collect data on the indoor air quality and concentration of various gases.
- Soil Moisture Monitoring Systems: to monitor soil moisture at various locations.
- Surveillance Systems: use WSNs for collecting surveillance data(motion data detection).
- Smart Grids : use WSNs for monitoring grids at various points.
- Structural Health Monitoring Systems: Use WSNs to monitor the health of structures(building, bridges) by collecting vibrations from sensor nodes deployed at various points in the structure.

WSNs are enabled by wireless communication protocols such as IEEE 802.15.4. Zig Bee is one of the most popular wireless technologies used by WSNs .Zig Bee specifications are based on IEEE 802.15.4. Zig Bee operates 2.4 GHz frequency and offers data rates upto 250 KB/s and range from 10 to 100meters.

1.4.2 Cloud Computing

Cloud computing is a transformative computing paradigm that involves delivering applications and services over the internet. Cloud computing involves provisioning of computing, networking and storage resources on demand and providing these resources as metered services to the users, in a “pay as you go”. Cloud computing resources can be provisioned on-demand by the users, without requiring interactions with the cloud service provider. The process of provisioning resources is automated.

Cloud computing services are offered to users in different forms.

- ***Infrastructure-as-a-service(IaaS):***Provides users the ability to provision computing and storage resources. These resources are provided to the users as a virtual machine instances and virtual storage.
- ***Platform-as-a-Service(PaaS):*** Provides users the ability to develop and deploy application in cloud using the development tools, APIs, software libraries and services provided by the cloud service provider.
- ***Software-as-a-Service(SaaS):*** Provides the user a complete software application or the user interface to the application itself. The cloud service provider manages the underlying cloud infrastructure including servers, network, operating systems, storage, and application software.

1.4.3 Big data Analysis

Big data is defined as collections of data sets whose volume , velocity or variety is so large that it is difficult to store, manage, process and analyze the data using traditional databases and data processing tools.

Some examples of big data generated by IoT are Sensor data generated by IoT systems.

- Machine sensor data collected from sensors established in industrial and energy systems.
- Health and fitness data generated IoT devices.
- Data generated by IoT systems for location and tracking vehicles.
- Data generated by retail inventory monitoring systems.

The underlying characteristics of Big Data are

Volume: There is no fixed threshold for the volume of data for big data. Big data is used for massive scale data.

Velocity: Velocity is another important characteristics of Big Data and the primary reason for exponential growth of data.

Variety: Variety refers to the form of data. Big data comes in different forms such as structured or unstructured data including test data, image , audio, video and sensor data .

1.4.4 Communication Protocols:

Communication Protocols form the back-bone of IoT systems and enable network connectivity and coupling to applications.

- Allow devices to exchange data over network.
- Define the exchange formats, data encoding addressing schemes for device and routing of packets from source to destination.
- It includes sequence control, flow control and retransmission of lost packets.

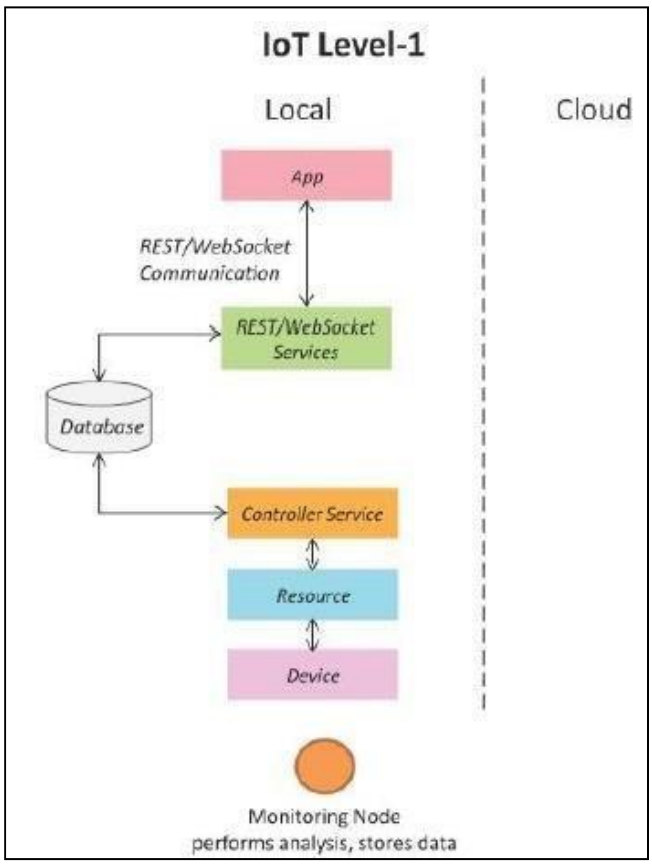
1.4.5 Embedded Systems:

Embedded Systemsis a computer system that has computer hardware and software embedded to perform specific tasks. Key components of embedded system include microprocessor or micro controller, memory (RAM, ROM, Cache), networking units (Ethernet Wi-Fi Adaptor), input/output units (Display, Keyboard, etc..,) and storage (Flash memory). Embedded System range from low cost miniaturized devices such as digital watches to devices such as digital cameras, POS terminals, vending machines, appliances etc.,

1.5 IOT Levels and Deployment Templates.

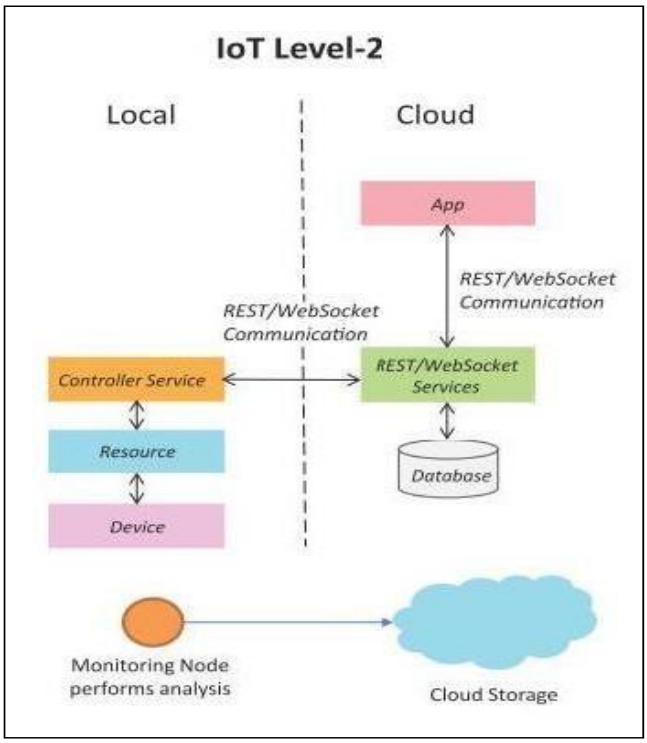
1.5.1 IoT Level-1

Level-1 IoT systems has a single node that performs sensing and/or actuation, stores data, performs analysis and host the application. Suitable for modeling low cost and low complexity solutions where the data involved is not big and analysis requirement are not computationally intensive. An e.g., of IoT Level1 isHomeautomation.The system consist of a single node that allows controlling the lights and appliances in a home the device used in this system interfaces with the lights and appliances using electronic rely switches. The status information of each light or appliances is maintained in a local database. REST services deployed locally allow retrieving and updating the state of each lighter appliance in the status database. The controller service continuously monitors the state of each light or appliance by retrieving the light from the database.



1.5.2 IoT Level 2

IoT Level2 has a single node that performs sensing and/or actuating and local analysis as shown in fig. Data is stored in cloud and application is usually cloud based. Level2 IoT systems are suitable for solutions where data are involved is big, however, the primary analysis requirement is not computationally intensive and can



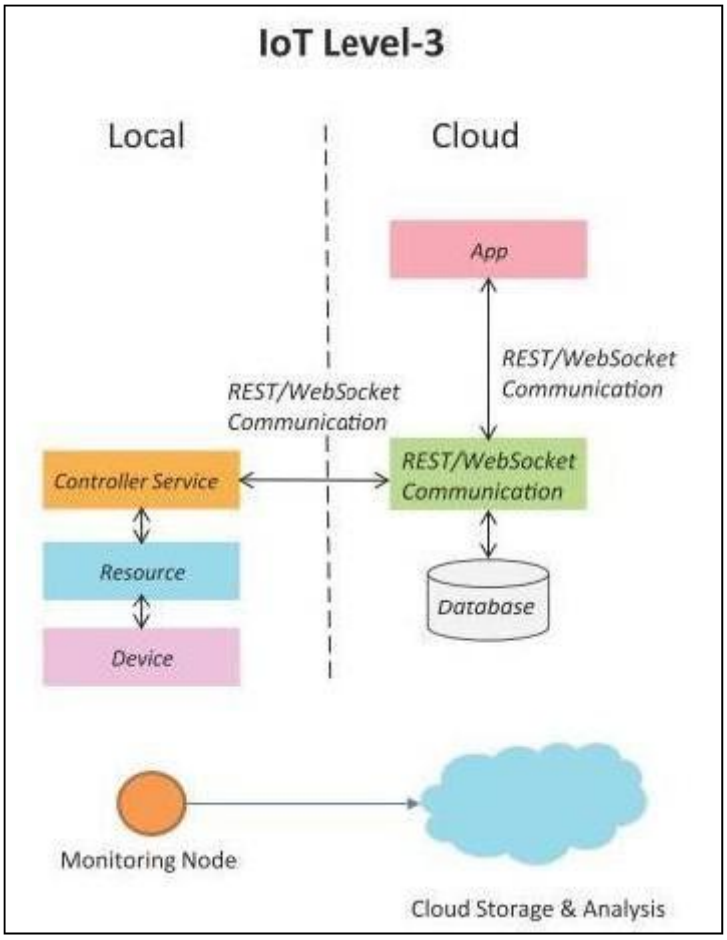
be done locally itself. An e,g., of Level2 IoT system for Smart Irrigation.

The system consists of a single node that monitors the soil moisture level and controls the irrigation system. The device used system collects soil moisture data from sensors. The controller service continuously monitors the moisture level. A cloud based REST web service is used for storing and retrieving moisture data which is stored in a cloud database. A cloud based application is used for visualizing the moisture level over a period of time which can help in making decision about irrigation schedule.

1.5.3 IoT Level 3

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

The system consists of a single node that monitors the vibration levels for the package being shipped . The device in this system uses accelerometer and gyroscope sensor for monitoring vibration levels. The controller serves in the sensor data to the cloud in a real time using a websocket service. The data is stored in the cloud and also visualizing the cloud based applications . The analysis components in the cloud can trigger alerts if the vibration level becomes greater than the threshold.



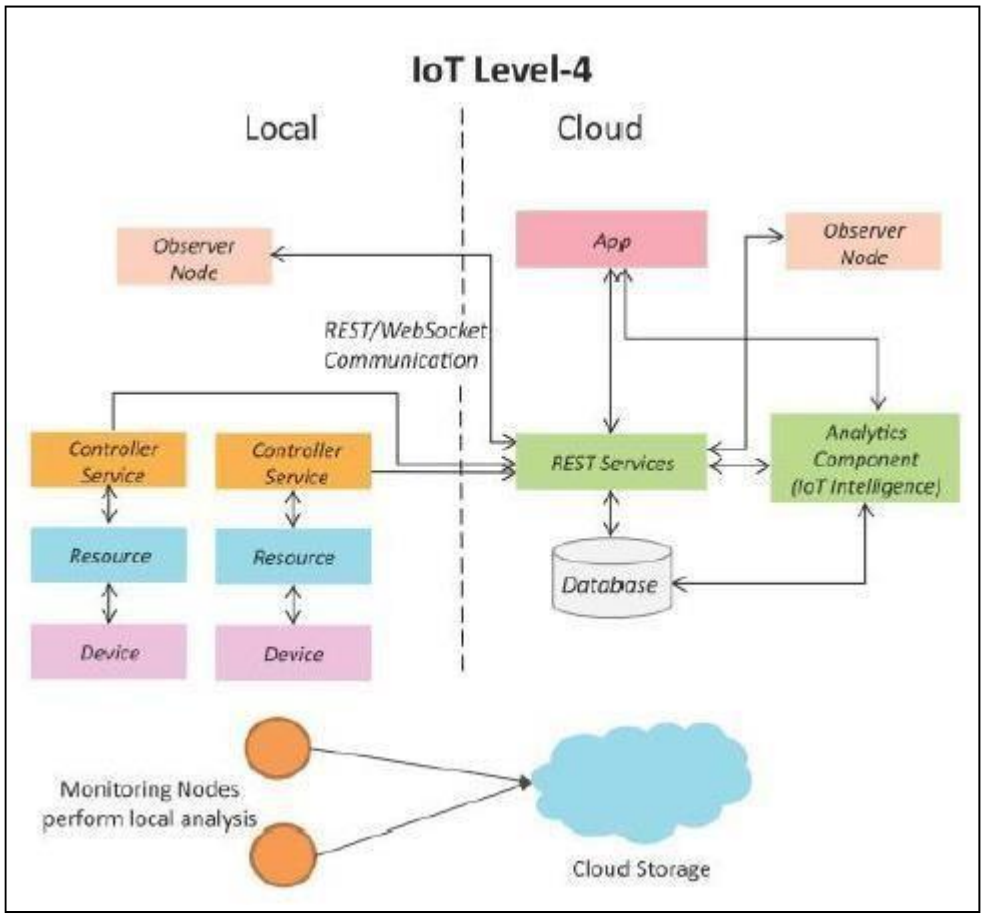
1.5.4 IoT Level 4

This System has multiple nodes that perform local analysis. Data is stored in the cloud and application is cloud based as shown in fig. Level4 contains local and cloud based observer nodes which can subscribe to and receive information collected in the cloud from IoT devices. Level 4 IoT systems are suitable for solutions where multiple nodes are required, the data involved in big and the analysis requirements are computationally intensive.

Example : IoT System for Noise Monitoring.

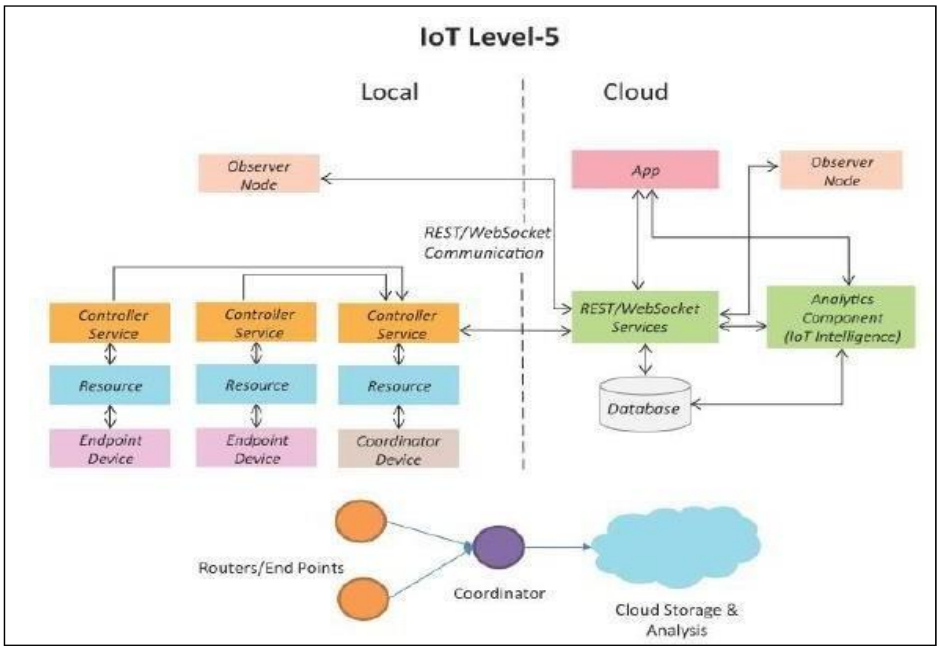
The system consists of multiple nodes placed in different locations for monitoring noise levels in an area. The nodes in this example are equipped with sound sensors. Nodes are independent of each other. Each

nodes runs its owner controller service that sends the data to the cloud . The data is stored in cloud database. The analysis of data collected from a number of nodes is done in the cloud. A cloud based application is used for visualizing the aggregated data.



1.5.5 IoT Level 5

System has multiple end nodes and one coordinator node as shown in fig. The end nodes that perform sensing and/or actuation. Coordinator node collects data from the end nodes and sends to the cloud. Data is stored and analyzed in the cloud and application is cloud based. Level5 IoT systems are suitable for solution based on wireless sensor network, in which data are high intensive.



Example :IoT system for Forest Fire Detection.

The system consists of multiple nodes placed in different locations for monitoring temperature, humidity and CO_2 levels in a forest. The end nodes in this example are equipped with various sensors such as temperature, humidity and CO_2. The coordinator node collects the data from the end nodes and act as a gateway that provides internet connectivity to the IoT system. The controller service on the coordinator device sends the collected data to the cloud. The data is stores in a cloud database. The analysis of data is done in the computing cloud to aggregate the data and make predictions. A cloud based applications is used for visualizing the data

1.5.6 IoT Level 6.

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

Example weather monitoring system

The system consists of multiple nodes placed in different locations for monitoring temperatures,humidity and pressure in an area.the end nodes are equipped with various sensors(such as temperature,humidity and pressure).the end nodes send the data to the cloud realtime using a websocketservice.the data is stored in a cloud database. The analysis of data is done in a cloud to aggregate a data and make predictions.acloud based application is used for visualizing the data.

DOMAIN SPECIFIC IoTs

1) Home Automation:

a) **Smart Lighting:** helps in saving energy by adapting the lighting to the ambient conditions and switching on/off or diming the light when needed.

b) **Smart Appliances:** make the management easier and also provide status information to the users remotely.

c) **Intrusion Detection:** use security cameras and sensors(PIR sensors and door sensors) to detect intrusion and raise alerts. Alerts can be in the form of SMS or email sent to the user.

d) **Smoke/Gas Detectors:** Smoke detectors are installed in homes and buildings to detect smoke that is typically an early sign of fire. Alerts raised by smoke detectors can be in the form of signals to a fire alarm system. Gas detectors can detect the presence of harmful gases such as CO, LPGetc.,

2) Cities:

a) **Smart Parking:** make the search for parking space easier and convenient for drivers. Smart parking are powered by IoT systems that detect the no. of empty parking slots and send information over internet to smart application backends.

b) **Smart Lighting:** for roads, parks and buildings can help in saving energy.

c) **Smart Roads:** Equipped with sensors can provide information on driving condition, travel time estimating and alert in case of poor driving conditions, traffic condition and accidents.

d) **Structural Health Monitoring:** uses a network of sensors to monitor the vibration levels in the structures such as bridges and buildings.

e) **Surveillance:** The video feeds from surveillance cameras can be aggregated in cloud based scalable storage solution.

f) **Emergency Response:** IoT systems for fire detection, gas and water leakage detection can help in generating alerts and minimizing their effects on the critical infrastructures.

3) Environment:

a) **Weather Monitoring:** Systems collect data from a no. of sensors attached and send the data to cloud based applications and storage back ends. The data collected in cloud can then be analyzed and visualized by cloud based applications.

b) **Air Pollution Monitoring:** System can monitor emission of harmful gases(CO_2, CO, NO, NO_2 etc.,) by factories and automobiles using gaseous and meteorological sensors. The collected data can be analyzed to make informed decisions on pollutions control approaches.

c) **Noise Pollution Monitoring:** Due to growing urban development, noise levels in cities have increased and even become alarmingly high in some cities. IoT based noise pollution monitoring systems use a no. of noise monitoring systems that are deployed at different places in a city. The data on noise levels from the station is collected on servers or in the cloud. The collected data is then aggregated to generate noise maps.

d) **Forest Fire Detection:** Forest fire can cause damage to natural resources, property and human life. Early detection of forest fire can help in minimizing damage.

e) **River Flood Detection:** River floods can cause damage to natural and human resources and human life. Early warnings of floods can be given by monitoring the water level and flow rate. IoT based river flood monitoring system uses a no. of sensor nodes that monitor the water level and flow rate sensors.

4) Energy:

a) **Smart Grids:** is a data communication network integrated with the electrical grids that collects and analyze data captured in near-real-time about power transmission, distribution and consumption. Smart grid technology provides predictive information and recommendations to utilities, their suppliers, and their customers on how best to manage power. By using IoT based sensing and measurement technologies, the health of equipment and integrity of the grid can be evaluated.

b) **Renewable Energy Systems:** IoT based systems integrated with the transformers at the point of interconnection measure the electrical variables and how much power is fed into the grid. For wind energy systems, closed-loop controls can be used to regulate the voltage at point of interconnection which coordinate wind turbine outputs and provides power support.

c) **Prognostics:** In systems such as power grids, real-time information is collected using specialized electrical sensors called Phasor Measurment Units(PMUs) at the substations. The information received from PMUs must be monitored in real-time for estimating the state of the system and for predicting failures.

5) Retail:

a) **Inventory Management:** IoT systems enable remote monitoring of inventory using data collected by RFIDreaders.
b) **Smart Payments:** Solutions such as contact-less payments powered by technologies such as Near Field Communication(NFC) and Bluetooth.

c) **Smart Vending Machines:** Sensors in a smart vending machines monitors its operations and send the data to cloud which can be used for predictive maintenance.

6) Logistics:

a) **Route generation & scheduling:** IoT based system backed by cloud can provide first response to the route generation queries and can be scaled upto serve a large transportation network.

b) **Fleet Tracking:** Use GPS to track locations of vehicles inreal-time.

c) **Shipment Monitoring:** IoT based shipment monitoring systems use sensors such as temp, humidity, to monitor the conditions and send data to cloud, where it can be analyzed to detect foods poilage.

d) **Remote Vehicle Diagnostics:** Systems use on-board IoT devices for collecting data on Vehicle operations(speed, RPMetc.,) and status of various vehicle subsystems.

7) Agriculture:

a) **Smart Irrigation:** to determine moisture amount in the soil.

b) **Green House Control:** to improve productivity.

8) Industry:

a) Machine diagnosis and prognosis

b) Indoor Air Quality Monitoring **9) Health and Lifestyle:**

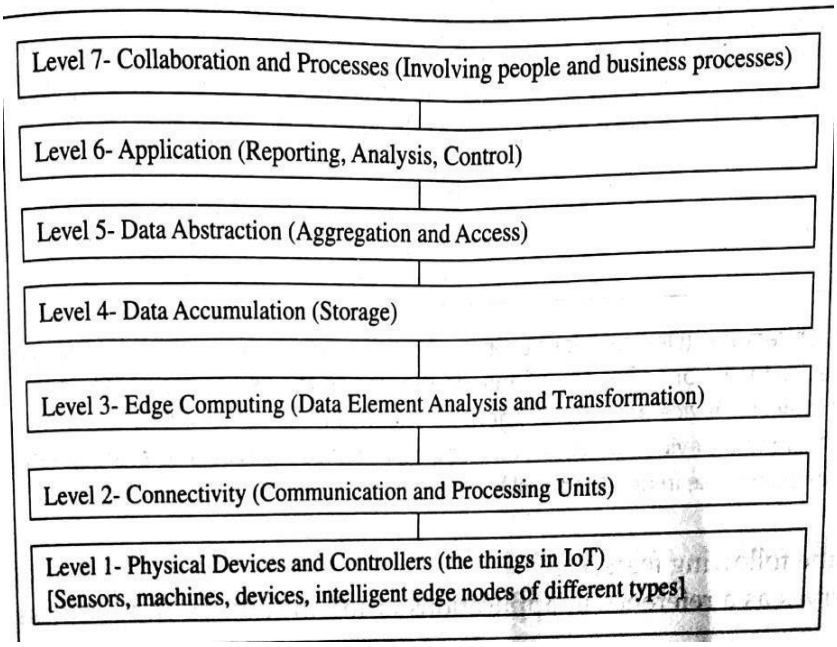
a) Health & Fitness Monitoring

b) Wearable Electronics

IoT Architectural View:

The IoT system is defined in different levels called as tiers. A model enables the conceptualisation of the framework.

A reference model can be used to depict the building blocks, successive interactions and integration.



The diagram below depicts the CISCO presentation of a reference model comprising of 7 levels and the functions of each level.

Features of the architecture:

- The architecture serves as a reference in the applications of IoT in services and business processes.
- A set of sensors which are smart, capture the data, perform necessary data element analysis and transformation as per device application framework and connect directly to a communication manager.
- The communication management subsystem consists of protocol handlers, message routers and access management.
- Data routes from gateway through the Internet and data centre to the application server or enterprise server which acquires that data.
- Organisation and analysis subsystems enable the services, business processes, enterprise integration and complex processes.

UNIT II ELEMENTS OF IOT

IoT and M2M- difference between IoT and M2M - Software Defined Networks - Network Function Virtualization - IoT systems management – Needs - NETCONF, YANG - IoT design methodology.

M2M Communication

- Machine-to-machine communication, or M2M, is exactly as it sounds: two machines "communicating," or exchanging data, without human interfacing or interaction.
- This includes serial connection, powerline connection (PLC), or wireless communications in the industrial Internet of Things (IoT).
- Switching over to wireless has made M2M communication much easier and enabled more applications to be connected. In general, when someone says M2M communication, they often are referring to cellular communication for embedded devices.
- Examples of M2M communication in this case would be vending machines sending out inventory information or ATM machines getting authorization to dispense cash. As businesses have realized the value of M2M, it has taken on a new name: The Internet of Things (IoT).
- IoT and M2M have similar promises: to fundamentally change the way the world operates. Just like IoT, M2M allows virtually any sensor to communicate, which opens up the possibility of systems monitoring themselves and automatically responding to changes in the environment, with a much reduced need for human involvement.
- M2M and IoT are almost synonymous—the exception is IoT (the newer term) typically refers to wireless communications, whereas M2M can refer to any two machines—wired or wireless—communicating with one another.

Traditionally, M2M focused on "industrial telematics," which is a fancy way of explaining data transfer for some commercial benefit. But many original uses of M2M still stand today, like smart meters. Wireless M2M has been dominated by cellular since it came out in the mid-2000's with 2G cell networks. Because of this, the cellular market has tried to brand M2M as an inherently cellular thing by offering M2M data plans. But cellular M2M is only one subsection of the market, and it shouldn't be thought of as a cellular-only area.

How M2M Works

As previously stated, machine-to-machine communication makes the Internet of Things possible. According to Forbes, M2M is among the fastest-growing types of connected device technologies in the market right now, largely because M2M technologies can connect millions of devices within a single network. The range of connected devices includes anything from vending machines to medical equipment to vehicles to buildings. Virtually anything that houses sensor or control technology can be connected to some sort of wireless network.

This sounds complex, but the driving thought behind the idea is quite simple. Essentially, M2M networks are very similar to LAN or WAN networks, but are exclusively used to allow machines, sensors, and controls, to communicate. These devices feed information they collect back to other devices in the network. This process allows a human (or an intelligent control unit) to assess what is going on across the whole network and issue appropriate instructions to member devices.

M2M Applications

The possibilities in the realm of M2M can be seen in four major use cases, which we've detailed below:

1. MANUFACTURING

Every manufacturing environment—whether it's food processing or general product manufacturing—relies on technology to ensure costs are managed properly and processes are executed efficiently. Automating manufacturing processes within such a fast-paced environment is expected to improve processes even more. In the manufacturing world, this could involve highly automated equipment maintenance and safety procedures.

For example, M2M tools allow business owners to be alerted on their smartphones when an important piece of equipment needs servicing, so they can address issues as quickly as they arise. Sophisticated networks of sensors connected to the Internet could even order replacement parts automatically.

2. HOME APPLIANCES

IoT already affects home appliance connectivity through platforms like Nest. However, M2M is expected to take home-based IoT to the next level. Manufacturers like LG and Samsung are already slowly unveiling smart home appliances to help ensure a higher quality of life for occupants.

For example, an M2M-capable washing machine could send alerts to the owners' smart devices once it finishes washing or drying, and a smart refrigerator could automatically order groceries from Amazon once its inventory is depleted. There are many more examples of home automation that can potentially improve quality of life for residents, including systems that allow members of the household to remotely control HVAC systems using their mobile devices. In situations where a homeowner decides to leave work early, he or she could contact the home heating system before leaving work to make sure the temperature at home will be comfortable upon arrival.

3. HEALTHCARE DEVICE MANAGEMENT

One of the biggest opportunities for M2M technology is in the realm of health care. With M2M technology, hospitals can automate processes to ensure the highest levels of treatment. Using devices that can react faster than a human healthcare professional in an emergency situation make this possible. For instance, when a patient's vital signs drop below normal, an M2M-connected life support device could automatically administer oxygen and additional care until a healthcare professional arrives on the scene. M2M also allows patients to be monitored in their own homes instead of in hospitals or care centers.

For example, devices that track a frail or elderly person's normal movement can detect when he or she has had a fall and alert a healthcare worker to the situation.

4. SMART UTILITY MANAGEMENT

In the new age of energy efficiency, automation will quickly become the new normal. As energy companies look for new ways to automate the metering process, M2M comes to the rescue, helping energy companies automatically gather energy consumption data, so they can accurately bill customers. Smart meters can track how much energy a household or business uses and automatically alert the energy company, which supplants sending out an employee to read the meter or requiring the customer to provide a reading. This is even more important as utilities move toward more dynamic pricing models, charging consumers more for energy usage during peak times. A few key analysts predict that soon, every object or device will need to be able to connect to the cloud. This is a bold but seemingly accurate statement. As more consumers, users, and business owners demand deeper connectivity, technology will need to be continually equipped to meet the needs and challenges of tomorrow. This will empower a wide range of highly automated processes, from equipment repairs and firmware upgrades to system diagnostics, data retrieval, and analysis. Information will be delivered to users, engineers, data scientists, and key decision-makers in real time, and it will eliminate the need for guesswork.

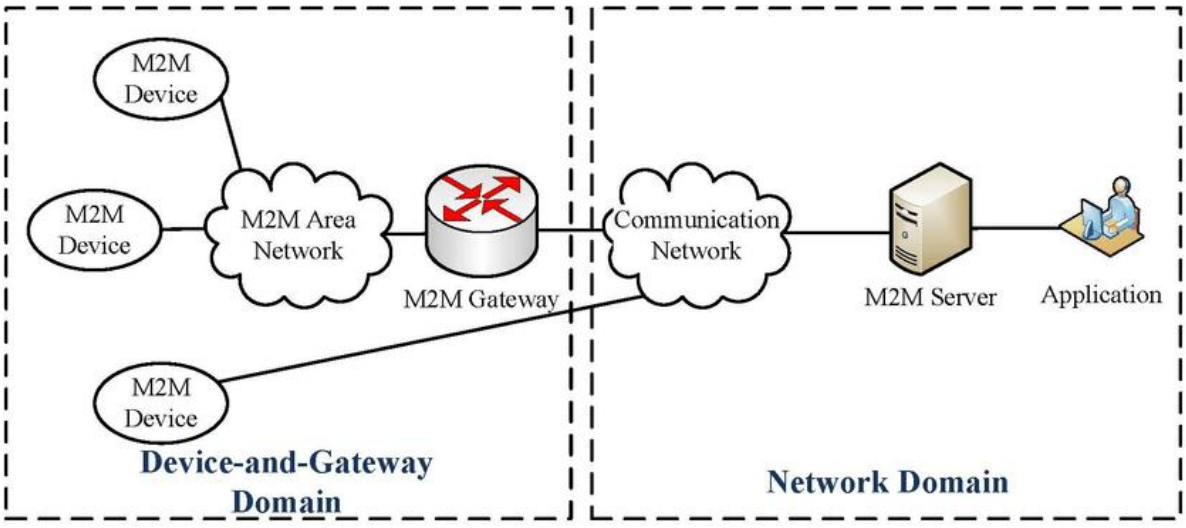
There are different M2M applications, environment monitoring, civil protection and public safety, supply chain management, energy and utility distribution as in smart grid, smart grid separately common. we have intelligent transportation systems, healthcare, automation of buildings, military applications, agriculture, home networks all these are different applications of M2M.

M2M features:

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

General Architecture of M2M Systems:

- M2M device connects to the network domain via direct connectivity or M2M gateway. In the first case, the M2M device connects to the network domain via the access network, which performs the procedures such as registration, authentication, authorization, management, and provisioning with the network domain. In the second case, the M2M device connects to the M2M gateway using the M2M area network.
- M2M area network provides connectivity between M2M devices and M2M gateways.
- M2M gateway acts as a proxy between M2M devices and the network domain. As an example, an M2M gateway can run an application that collects and treats various information (e.g., contextual parameters) from sensors and meters.
- M2M communication network provides connection between the M2M gateways/devices and the M2M servers. Usually it contains two parts: the access network and the Internet.
- M2M server works as a middleware layer to pass data through various application services.



Difference Between IoT and M2M:

M2M, or machine-to-machine, is a direct communication between devices using wired or wireless communication channels. M2M refers to the interaction of two or more devices/machines that are connected to each other. These devices capture data and share with other connected devices, creating an intelligent network of things or systems. Devices could be sensors, actuators, embedded systems or other connected elements.

M2M technology could be present in our homes, offices, shopping malls and other places. Controlling electrical appliances like bulbs and fans using RF or Bluetooth from your smartphone is a simple example of M2M applications at home. Here, the electrical appliance and your smartphone are the two machines interacting with each other.
The Internet of Things (IoT) is the network of physical devices embedded with sensors, software and electronics, enabling these devices to communicate with each other and exchange data over a computer network. The things in the IoT refer to hardware devices uniquely identifiable through a network platform within the Internet infrastructure.

M2M versus the IoT	
M2M	**IoT**
M2M is about direct communication between machines.	The IoT is about sensors automation and Internet platform.
It supports point-to-point communication.	It supports cloud communication.
Devices do not necessarily rely on an Internet connection.	Devices rely on an Internet connection.
M2M is mostly hardware-based technology.	The IoT is both hardware- and software-based technology.
Machines normally communicate with a single machine at a time.	Many users can access at one time over the Internet.
A device can be connected through mobile or other network.	Data delivery depends on the Internet protocol (IP) network.

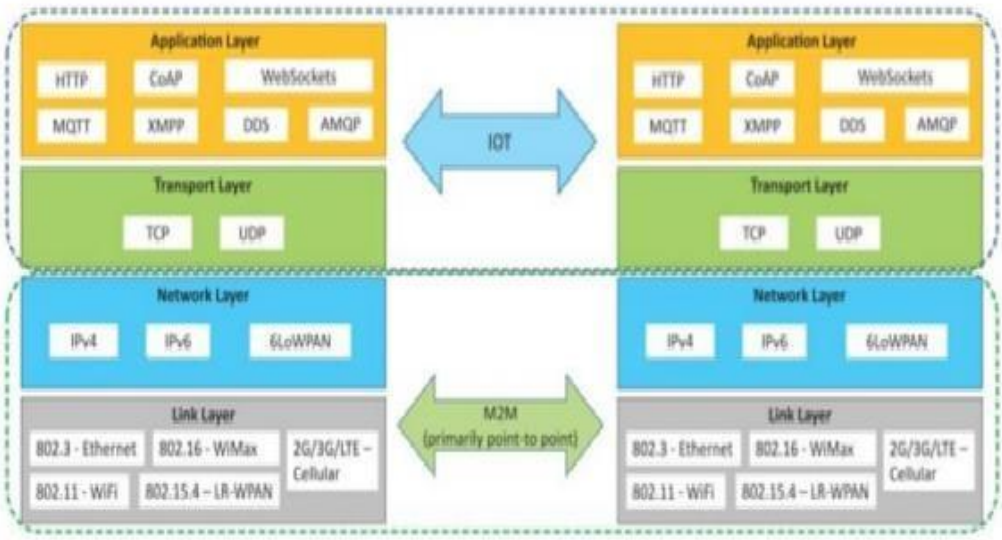
Some more differences like:

Communication Protocols:

• M2M and IoT can differ in how the communication between the machines ordevices happens.
• M2M uses either proprietary or non-IP based communication protocols for communication within the M2M area networks. IoT uses IP bases communication protococls.

Machines in M2M vs Things in IoT:

• The "Things" in IoT refers to physical objects that have unique identifiers and can sense and communicate with their external environment (and user applications) or their internal physical states.
• M2M systems, in contrast to IoT, typically have homogeneous machine types within an M2M area



network.

Hardware vs Software Emphasis:

• While the emphasis of M2M is more on hardware with embedded modules, the emphasis of IoT is more on software.

Data Collection & Analysis:

• M2M data is collected in point solutions and often in on-premises storage infrastructure.
• In contrast to M2M, the data in IoT is collected in the cloud (can be public, private or hybrid cloud).

Applications:

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

Software defined Networking(SDN)

SDN stands for Software Defined Network which is a networking architecture approach. It enables the control and management of the network using software applications. Through Software Defined Network (SDN) networking behavior of the entire network and its devices are programmed in a centrally controlled manner through software applications using open APIs.

To understand software-defined networks, we need to understand the various planes involved in networking.

1. Data Plane
2. Control Plane

Data plane:

All the activities involving as well as resulting from data packets sent by the end-user belong to this plane. This includes:

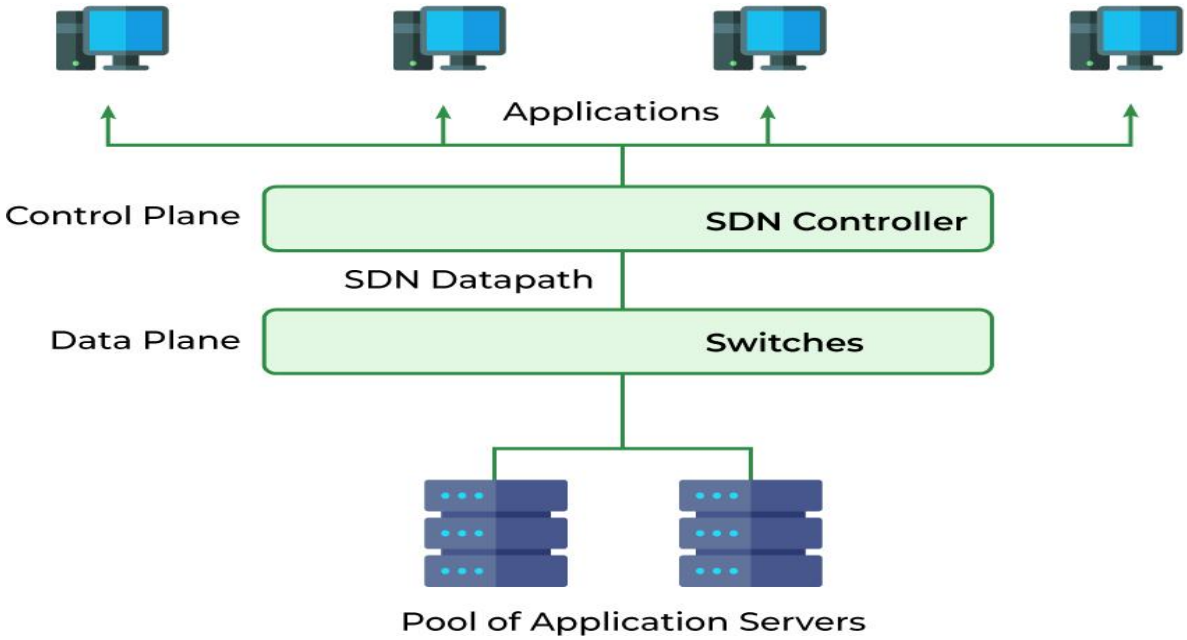
- Forwarding of packets.
- Segmentation and reassembly of data.
- Replication of packets for multicasting.

Control plane:

All activities necessary to perform data plane activities but do not involve end-user data packets belong to this plane. In other words, this is the brain of the network. The activities of the control plane include:

- Making routing tables.
- Setting packet handling policies.

Software Defined Networking (SDN)



Why SDN is Important?

- **Better Network Connectivity:** SDN provides very better network connectivity for sales, services, and internal communications. SDN also helps in faster data sharing.
- **Better Deployment of Applications:** Deployment of new applications, services, and many business models can be speed up using Software Defined Networking.
- **Better Security:** Software-defined network provides better visibility throughout the network. Operators can create separate zones for devices that require different levels of security. SDN networks give more freedom to operators.
- **Better Control with High Speed:** Software-defined networking provides better speed than other networking types by applying an open standard software-based controller.

In short, it can be said that- SDN acts as a "Bigger Umbrella or a HUB" where the rest of other networking technologies come and sit under that umbrella and get merged with another platform to bring out the best of the best outcome by decreasing the traffic rate and by increasing the efficiency of data flow.

Where is SDN Used?

- Enterprises use SDN, the most widely used method for application deployment, to deploy applications faster while lowering overall deployment and operating costs. SDN allows IT administrators to manage and provision network services from a single location.
- Cloud networking software-defined uses white-box systems. Cloud providers often use generic hardware so that the Cloud data center can be changed and the cost of CAPEX and OPEX saved.

Components of Software Defining Networking (SDN)

The three main components that make the SDN are:

1. **SDN Applications:** SDN Applications relay requests or networks through SDN Controller using API.
2. **SDN controller:** SDN Controller collects network information from hardware and sends this information to applications.
3. **SDN networking devices:** SDN Network devices help in forwarding and data processing tasks.

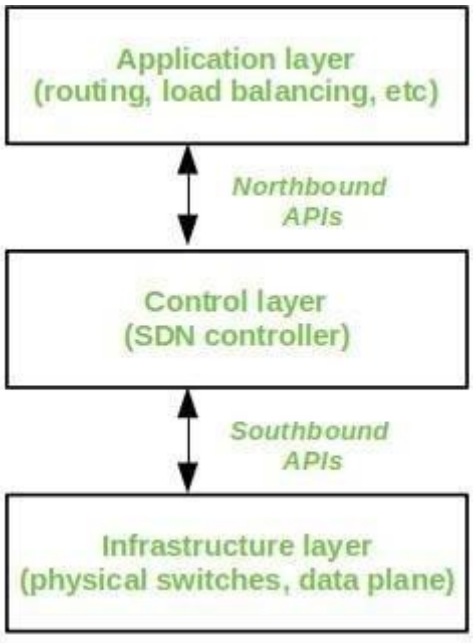
SDN Architecture

In a traditional network, each switch has its own data plane as well as the control plane. The control plane of various switches exchange topology information and hence construct a forwarding table that decides where an incoming data packet has to be forwarded via the data plane. Software-defined networking (SDN) is an approach via which we take the control plane away from the switch and assign it to a centralized unit called the SDN controller. Hence, a network administrator can shape traffic via a centralized console without having to touch the individual switches. The data plane still resides in the switch and when a packet enters a switch, its forwarding activity is decided based on the entries of flow tables, which are pre-assigned by the controller. A flow table consists of match fields (like input port number and packet header) and instructions. The packet is first matched against the match fields of the flow table entries. Then the instructions of the corresponding flow entry are executed. The instructions can be forwarding the packet via one or multiple ports, dropping the packet, or adding headers to the packet. If a packet doesn't find a corresponding match in the flow table, the switch queries the controller which sends a new flow entry to the switch. The switch forwards or drops the packet based on this flow entry.

A typical SDN architecture consists of three layers.

- **Application layer:** It contains the typical network applications like intrusion detection, firewall, and load balancing
- **Control layer:** It consists of the SDN controller which acts as the brain of the network. It also allows hardware abstraction to the applications written on top of it.
- **Infrastructure layer:** This consists of physical switches which form the data plane and carries out the actual movement of data packets.

The layers communicate via a set of interfaces called the north-bound APIs(between the application and control layer) and southbound APIs(between the control and infrastructure layer).



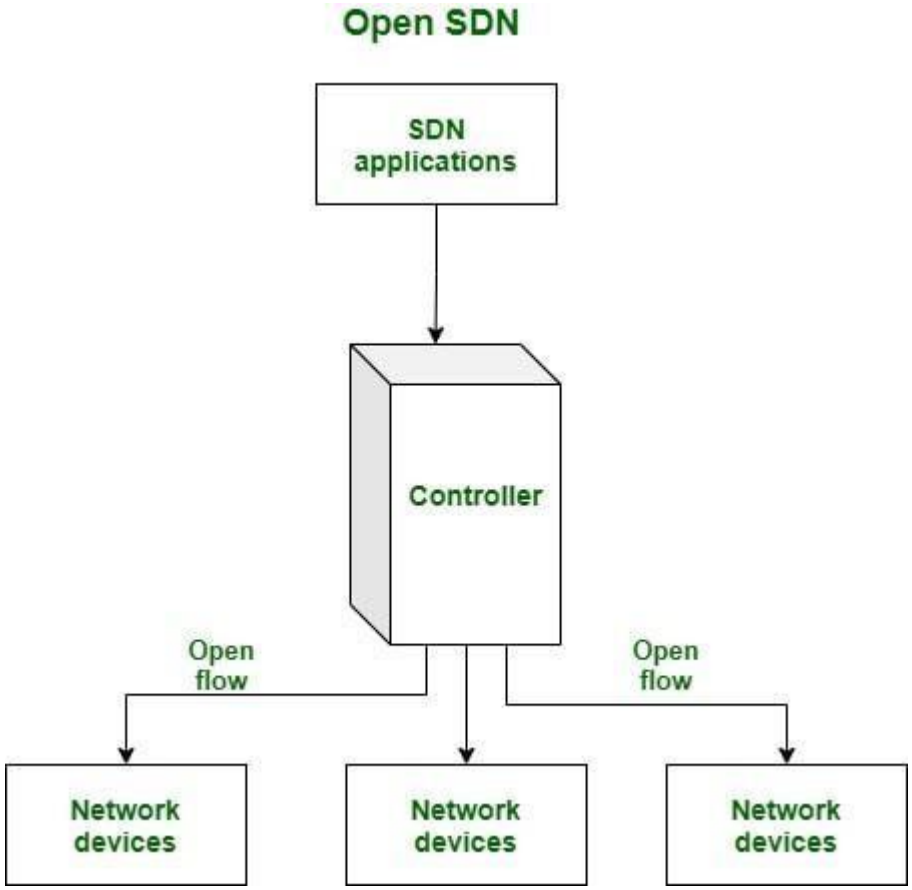
SDN Architecture

Different Models of SDN

There are several models, which are used in SDN:

1. Open SDN
2. SDN via APIs
3. SDN via Hypervisor-based Overlay Network
4. Hybrid SDN

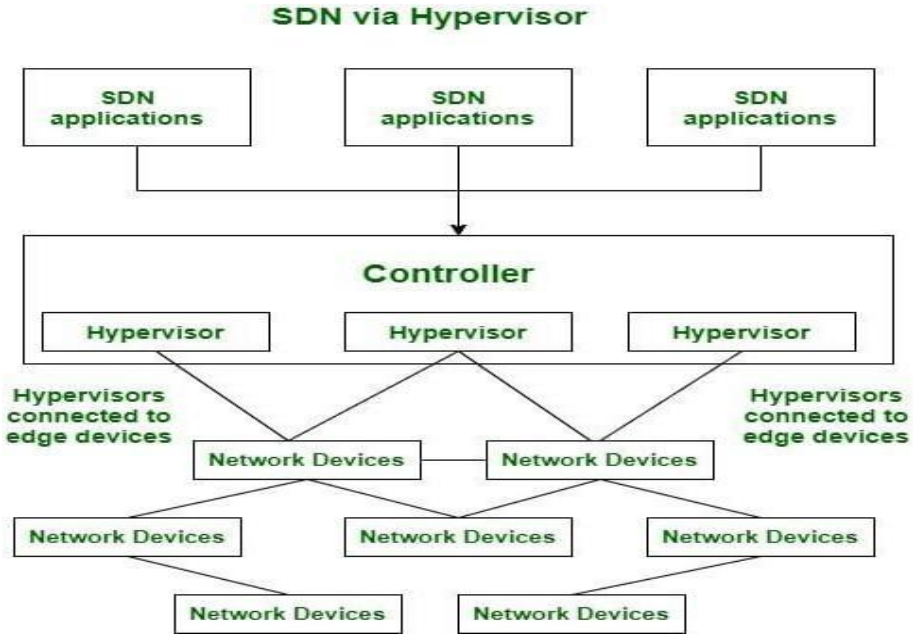
1. Open SDN: Open SDN is implemented using the OpenFlow switch. It is a straightforward implementation of SDN. In Open SDN, the controller communicates with the switches using south-bound API with the help of OpenFlow protocol.



Open SDN

2. SDN via APIs: In SDN via API, the functions in remote devices like switches are invoked using conventional methods like SNMP or CLI or through newer methods like Rest API. Here, the devices are provided with control points enabling the controller to manipulate the remote devices using APIs.

3. SDN via Hypervisor-based Overlay Network: In SDN via the hypervisor, the configuration of physical devices is unchanged. Instead, Hypervisor based overlay networks are created over the physical network. Only the devices at the edge of the physical network are connected to the virtualized networks, thereby concealing the information of other devices in the physical network.

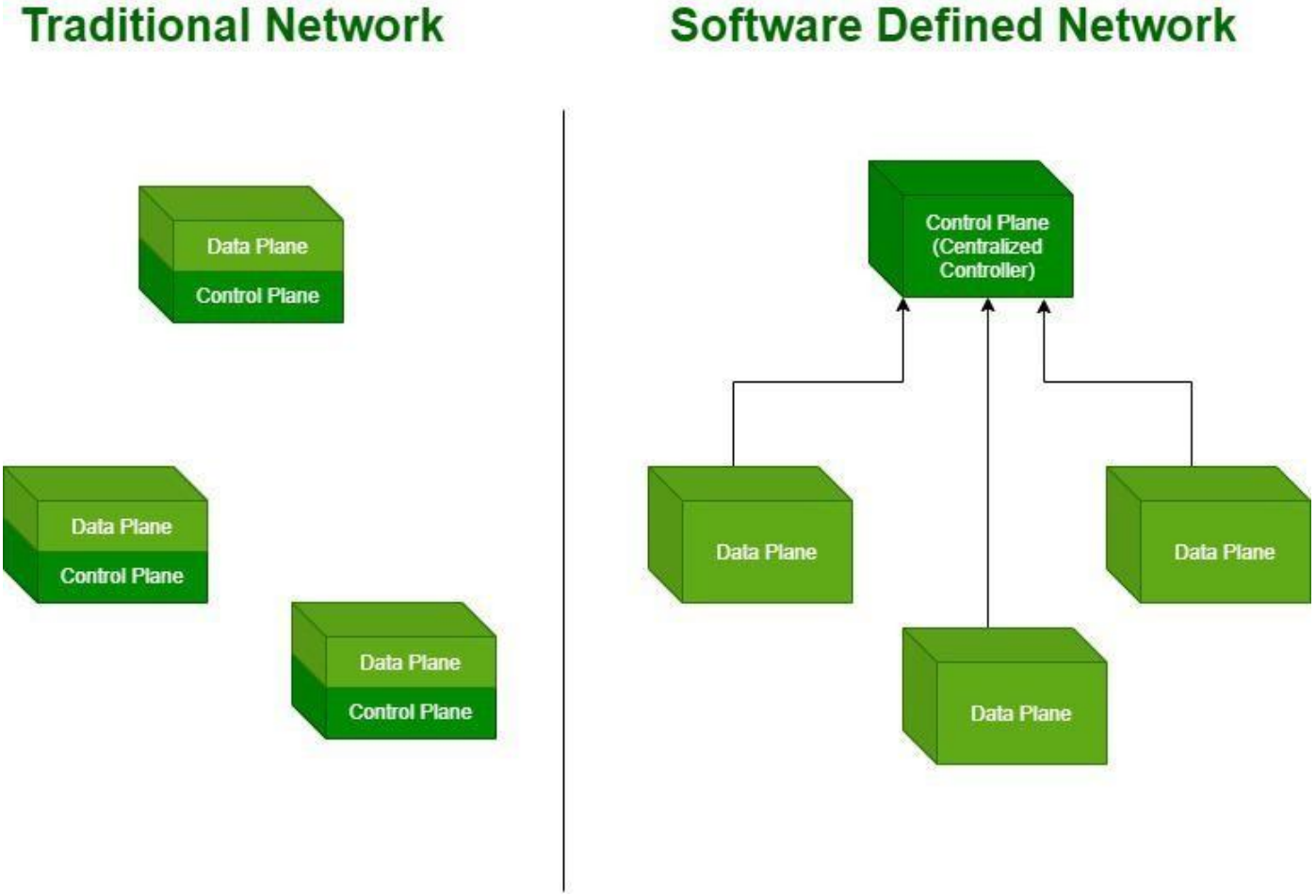


SDN via Hypervisor-based Overlay Network

4. Hybrid SDN: Hybrid Networking is a combination of Traditional Networking with software-defined networking in one network to support different types of functions on a network.
Difference between SDN and Traditional Networking

Software Defined Networking	**Traditional Networking**
Software Defined Network is a virtual networking approach.	A traditional network is the old conventional networking approach.
Software Defined Network is centralized control.	Traditional Network is distributed control.
This network is programmable.	This network is nonprogrammable.
Software Defined Network is the open interface.	A traditional network is a closed interface.
In Software Defined Network data plane and control, the plane is decoupled by software.	In a traditional network data plane and control plane are mounted on the same plane.

For more details you can refer differences between SDN and Traditional Networking article.



Difference between SDN and Traditional Networking

Advantages of SDN

- The network is programmable and hence can easily be modified via the controller rather than individual switches.
- Switch hardware becomes cheaper since each switch only needs a data plane.
- Hardware is abstracted, hence applications can be written on top of the controller independent of the switch vendor.
- Provides better security since the controller can monitor traffic and deploy security policies. For example, if the controller detects suspicious activity in network traffic, it can reroute or drop the packets.

Disadvantages of SDN

- The central dependency of the network means a single point of failure, i.e. if the controller gets corrupted, the entire network will be affected.
- The use of SDN on large scale is not properly defined and explored.

Network Functions Virtualization

- The term "Network Functions Virtualization" (NFV) refers to the use of virtual machines in place of physical network appliances. There is a requirement for a hypervisor to operate networking software and procedures like load balancing and routing by virtual computers. A network functions virtualization standard was first proposed at the OpenFlow World Congress in 2012 by the European Telecommunications Standards Institute (ETSI), a group of service providers that includes AT&T, China Mobile, BT Group, Deutsche Telekom, and many more.

Need of NFV:

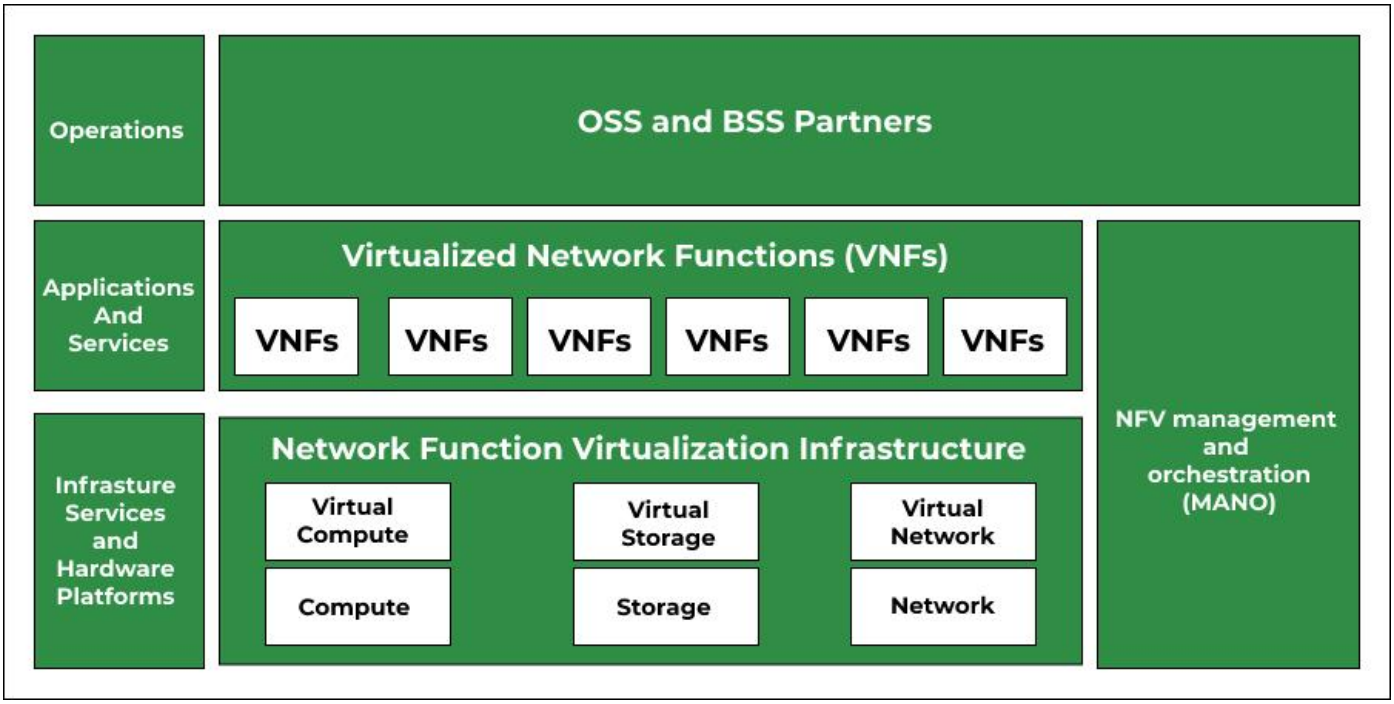
With the help of NFV, it becomes possible to separate communication services from specialized hardware like routers and firewalls. This eliminates the need for buying new hardware and network operations can offer new services on demand. With this, it is possible to deploy network components in a matter of hours as opposed to months as with conventional networking. Furthermore, the virtualized services can run on less expensive generic servers.

Advantages:

- Lower expenses as it follows Pay as you go which implies companies only pay for what they require.
- Less equipment as it works on virtual machines rather than actual machines which leads to fewer appliances, which lowers operating expenses as well.
- Scalability of network architecture is quite quick and simple using virtual functions in NFV. As a result, it does not call for the purchase of more hardware.

Working:

Usage of software by virtual machines enables to carry out the same networking tasks as conventional hardware. The software handles the task of load balancing, routing, and firewall security. Network engineers can automate the provisioning of the virtual network and program all of its various components using a hypervisor or software-defined networking controller.



Benefits of NFV:

- Many service providers believe that advantages outweigh the issues of NFV.
- Traditional hardware-based networks are time-consuming as these require network administrators to buy specialized hardware units, manually configure them, then join them to form a network. For this skilled or well-equipped worker is required.
- It costs less as it works under the management of a hypervisor, which is significantly less expensive than buying specialized hardware that serves the same purpose.
- Easy to configure and administer the network because of a virtualized network. As a result, network capabilities may be updated or added instantly.

Risks of NFV:
Security hazards do exist, though, and network functions virtualization security issues have shown to be a barrier to widespread adoption among telecom companies. The following are some dangers associated with implementing network function virtualization that service providers should take into account:

- **Physical security measures do not work:** Comparing virtualized network components to locked-down physical equipment in a data center enhances their susceptibility to new types of assaults.
- Malware is difficult to isolate and contain: Malware travels more easily among virtual components running on the same virtual computer than between hardware components that can be isolated or physically separated.
- **Network activity is less visible:** Because traditional traffic monitoring tools struggle to detect potentially malicious anomalies in network traffic going east-west between virtual machines, NFV necessitates more fine-grained security solutions.

NFV Architecture:

An individual proprietary hardware component, such as a router, switch, gateway, firewall, load balancer, or intrusion detection system, performs a specific networking function in a typical network architecture. A virtualized network substitutes software programs that operate on virtual machines for these pieces of hardware to carry out networking operations.

Three components make up an NFV architecture:

- **Centralized virtual network infrastructure:** The foundation of an NFV infrastructure can be either a platform for managing containers or a hypervisor that abstracts the resources for computation, storage, and networking.
- **Applications:** Software delivers many forms of network functionality by substituting for the hardware elements of a conventional network design (virtualized network functions).
- **Framework:** To manage the infrastructure and provide network functionality, a framework is required (commonly abbreviated as MANO, meaning Management, Automation, and Network Orchestration).

NeedforIoTSystems Management

Managingmultipledeviceswithinasinglesystem requiresadvancedmanagementcapabilitie.

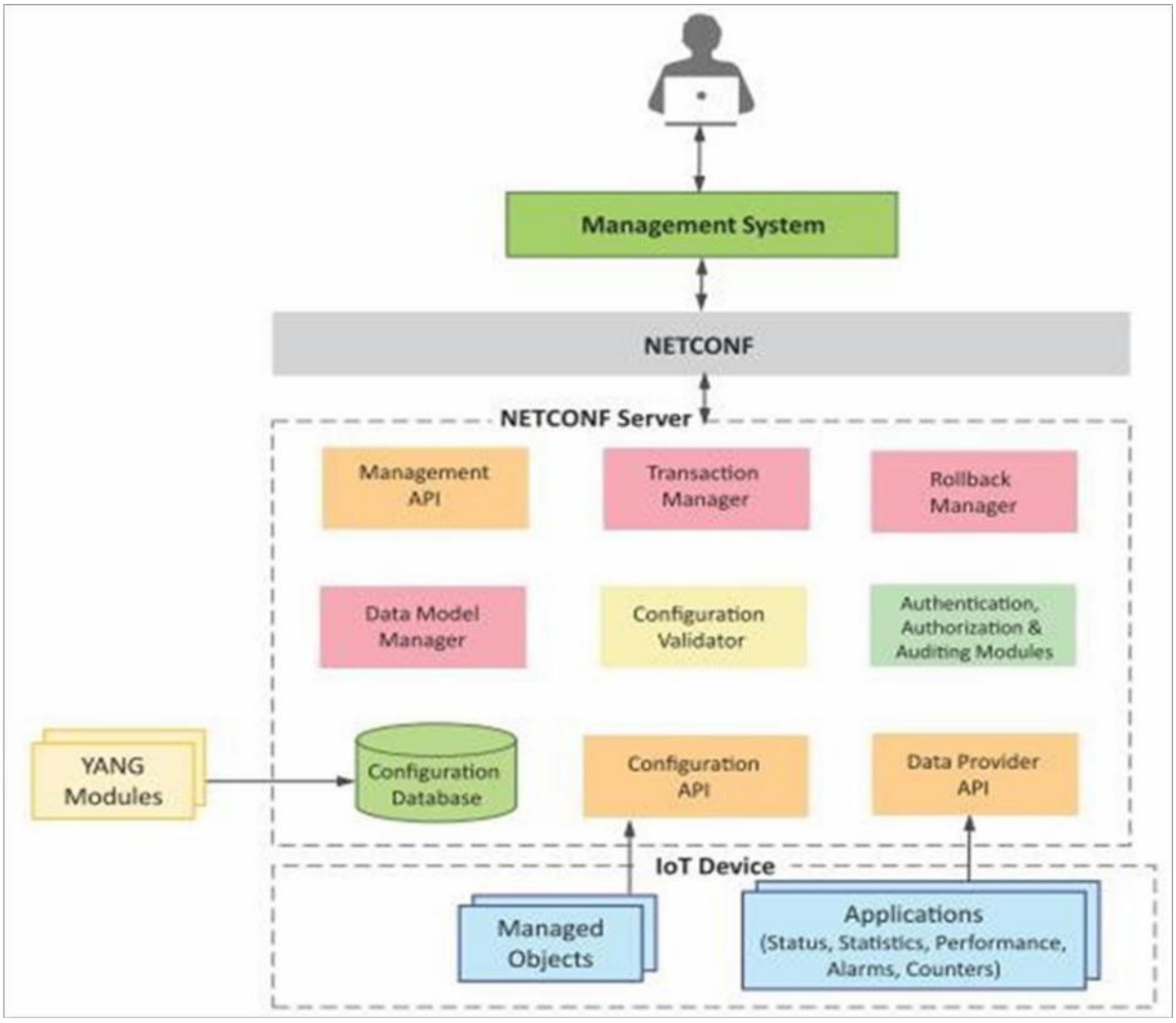
1) **AutomatingConfiguration:**IoTsystemmanagementcapabilitiescanhelpin automating the systemconfiguration.
2) **Monitoring Operational & Statistical Data :** Management systems can help in monitoring opeartional and statistical data of a system. This data can be used for fault diagnosis orprognosis.
3) **Improved Reliability:** A management system that allows validating the system configurationsbeforetheyareputinto effect canhelp inimprovingthe systemreliability.
4) **System Wide Configurations :** For IoT systems that consists of multiple devices or nodes, ensuring system wide configuration can be critical for the correct functioning of thesystem.
5) **Multiple System Configurations :** For some systems it may be desirable to have multiplevalidconfigurationswhichareappliedatdifferenttimesorin certainconditions.
6) **Retrieving&ReusingConfigurations:** Managementsystemswhichhavethecapability of retrieving configurations from devices can help in reusing the configurations for other devices of the sametype.

IoTSystemsManagementwithNETCONF-YANG

YANGisadatamodelinglanguageusedtomodelconfigurationandstatedatamanupulated by the NETCONF protocol.

ThegenericapproachofIoTdevicemanagement with NETCONF-YANG.
Rolesof various componentsare:

1) ManagementSystem
2) ManagementAPI
3) TransactionManager
4) RollbackManager
5) Data ModelManager
6) ConfigurationValidator
7) ConfigurationDatabase
8) ConfigurationAPI
9) Data ProviderAPI



1) **Management System :** The operator uses a management system to send NETCONF messages to configure the IoT device and receives state information and notifications from the device as NETCONFmessages.

2) **ManagementAPI :**allowsmanagementapplicationtostartNETCONFsessions.

3) **Transaction Manager:** executes all the NETCONF transactions and ensures that ACID properties hold true for thetrasactions.

4) **Rollback Manager :** is responsible for generating all the transactions necessary to rollback a current configuration to its original state.

5) **DataModelManager:** KeepstrackofalltheYANGdatamodelsandthecorresponding managed objects. Also keeps track oftheapplications which providedataforeach partof a datamodel.

6) **Configuration Validator :** checks if the resulting configuration after applying a transaction would be a validconfiguration.

7) **ConfigurationDatabase:**containsbothconfigurationandoperastionaldata.

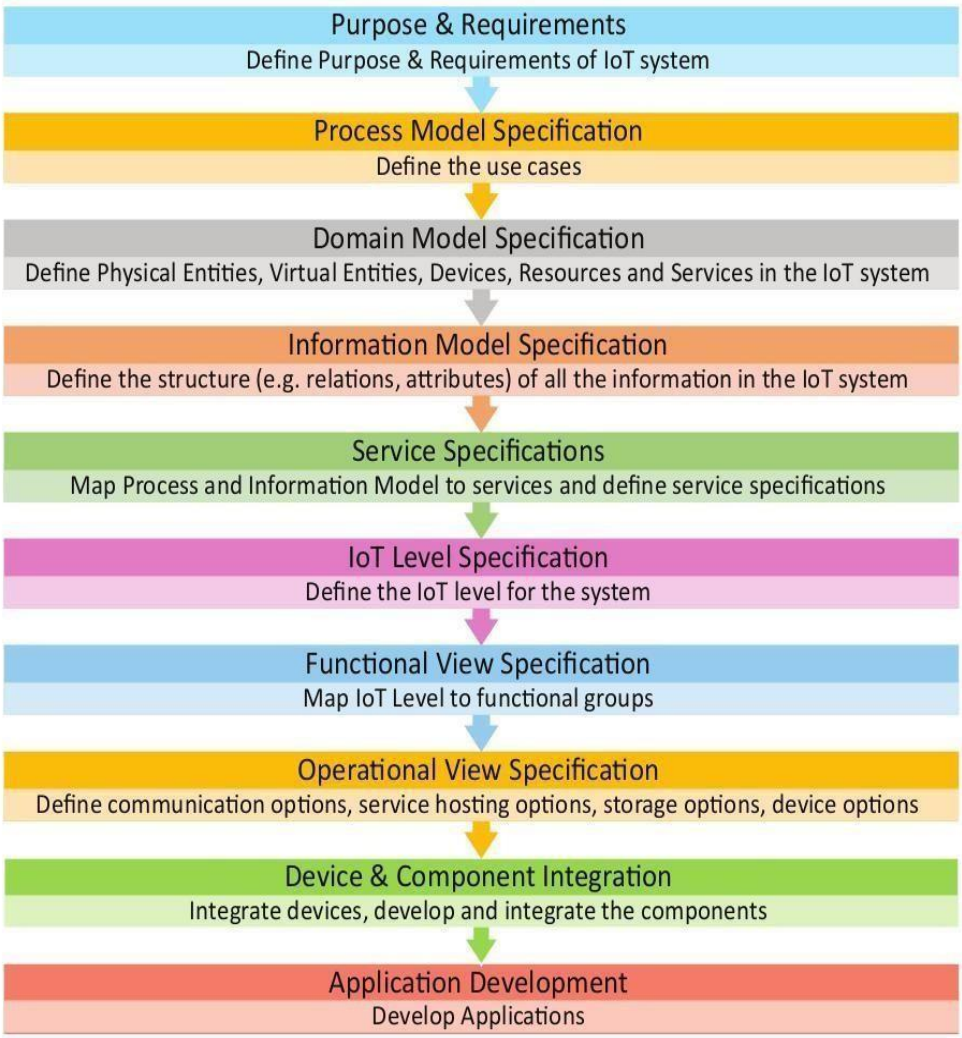
8) **Configuration API :** Using the configuration API the application on the IoT device can be read configuration data from the configuration datastore and write opeartional data to the opearationaldatastore.

9) **DataProviderAPI:** Applicationsonthe IoTdevicecanregisterforcallbacksforvarious events using the Data Provider API. Through the Data Provider API, the applications can report statistics and opeartionaldata.

StepsforIoTdeviceManagementwithNETCONF-YANG

1) Create a YANG model of the system that defines the configuration and state data of the system.
2) CompletetheYANGmodelwiththe_Inctool'whichcomeswithLibnetconf.
3) Fillinthe IoTdevicemangementcodeintheTransAPImodule.
4) BuildthecallbacksC filetogeneratethelibraryfile.
5) Load the YANG module and the TransAPImodule into the Netopeer server using Netopeermanagertool.
6) The operator can now connect from the management system to the Netopeer server using the NetopeerCLI.
7) Operator can issue NETCONF commands from the Netopeer CLI. Command can be issued to changew the configuration dsta, get operational dat or execute an RPC on the IoTdevice.

IoT Design Methodology – Steps



Step 1: Purpose & Requirements Specification:

The first step in IoT system design methodology is to define the purpose and requirements of the system. In this step, the system purpose, behavior and requirements (such as data collection requirements, data analysis requirements, system management requirements, data privacy and security requirements, user interface requirements, ...) are captured.

Step 2: Process Specification:

The second step in the IoT design methodology is to define the process specification. In this step, the use cases of the IoT system are formally described based on and derived from the purpose and requirement specifications.

Step 3: Domain Model Specification:

The third step in the IoT design methodology is to define the Domain Model. The domain model describes the main concepts, entities and objects in the domain of IoT system to be designed. Domain model defines the attributes of the objects and relationships between objects. Domain model provides an abstract representation of the concepts, objects and entities in the IoT domain, independent of any specific technology or platform. With the domain model, the IoT system designers can get an understanding of the IoT domain for which the system is to be designed.

Step 4: Information Model Specification:

The fourth step in the IoT design methodology is to define the Information Model. Information Model defines the structure of all the information in the IoT system, for example, attributes of Virtual Entities, relations, etc. Information model does not describe the specifics of how the information is represented or stored. To define the information model, we first list the Virtual Entities defined in the Domain Model. Information model adds more details to the Virtual Entities by defining their attributes and relations.

Step 5: Service Specifications:

The fifth step in the IoT design methodology is to define the service specifications. Service specifications define the services in the IoT system, service types, service inputs/output, service endpoints, service schedules, service preconditions and service effects.

Step 6: IoT Level Specification:

The sixth step in the IoT design methodology is to define the IoT level for the system.

Step 7: Functional View Specification:

The seventh step in the IoT design methodology is to define the Functional View. The Functional View (FV) defines the functions of the IoT systems grouped into various Functional Groups (FGs). Each Functional Group either provides functionalities for interacting with instances of concepts defined in the Domain Model or provides information related to these concepts.

Step 8: Operational View Specification:

The eighth step in the IoT design methodology is to define the Operational View Specifications. In this step, various options pertaining to the IoT system deployment and operation are defined, such as, service hosting options, storage options, device options, application hosting options, etc

Step 9: Device & Component Integration:

The ninth step in the IoT design methodology is the integration of the devices and components.

Step 10: Application Development:

The final step in the IoT design methodology is to develop the IoT application.

UNIT III IOT PROTOCOLS

Sensors and actuators - Communication modules – Zigbee - LoRa - RFID - Wi-Fi - Power sources.

Sensors:

- Generally speaking, a sensor is a device that is able to detect changes in an environment. By itself, a sensor is useless, but when we use it in an electronic system, it plays a key role. A sensor is able to measure a physical phenomenon (like temperature, pressure, and so on) and transform it into an electric signal. These three features should be at the base of a good sensor:
- It should be sensitive to the phenomenon that it measures
- It should not be sensitive to other physical phenomena
- It should not modify the measured phenomenon during the measurement process
- There is a wide range of sensors we can exploit to measure almost all the physical properties around us. A few common sensors that are widely adopted in everyday life include thermometers, pressure sensors, light sensors, accelerometers, gyroscopes, motion sensors, gas sensors and many more.

A sensor can be described using several properties, the most important being:

- **Range**: The maximum and minimum values of the phenomenon that the sensor can measure.
- **Sensitivity**: The minimum change of the measured parameter that causes a detectable change in output signal.
- **Resolution**: The minimum change in the phenomenon that the sensor can detect.

Sensor Classification:

Sensors can be grouped using several criteria:

Passive or Active: Passive sensors do not require an external power source to monitor an environment, while Active sensors require such a source in order to work. A passive sensor is one which just 'listens' to what is happening.

Examples include:

- A light sensor which detects if a light is shining on it.
- An infra-red sensor which detects the temperature of an object.

An active sensor is one which transmits a signal into the environment and then measures the response that comes back.
One example is an ultrasonic system:

- A pulse of ultrasound is emitted.
- If an object is in the way, the pulse is reflected back.
- The sensor detects it.
- The time taken between emission and detection gives an indication of the distance of the object.

Another classification is based on the method used to detect and measure the property (mechanical, chemical, etc.).

Analog and Digital: Analog sensors produce an analog, or continuous, signal while digital sensors produce a discrete signal.

There are different types of sensors that produce continuous analog output signal and these sensors are analog sensors. This continuous output signal produced by the analog sensors is proportional to the measurand. Generally, There are various types of analog sensors; practical examples of various types of analog sensors are as follows: accelerometers, pressure sensors, light sensors, sound sensors, temperature sensors, and so on.
Unlike analog sensor, Digital Sensor produce discrete values (0 and 1's). Discrete values often called digital or binary signals in digital communication.
Electronic sensors or electrochemical sensors in which data conversion and data transmission take place digitally are digital sensors. These digital sensors are replacing analog sensors as they are capable of overcoming the drawbacks of analog sensors. The digital sensor consists of majorly three components such as senor, cable, and transmitter. But, In digital sensors, the signal measured directly converted into digital signal output inside the digital sensor itself. So, this digital signal transmitted through cable digitally. There are different types of digital sensors that overcome the disadvantages of analog sensors.
Then, scaler sensors basically measure scalar variables which can measure only the changes in the magnitude whereas, the vector senses not only the magnitude, but also the direction. So, scalar sensor example would be temperature sensor is an example of scalar sensor because you know irrespective of which orientation you put, the sensor temperature sensor or in which direction you are taking it, it is going to give you the magnitude value. Only the changes in the magnitude of the temperature, on the contrary we have the vector sensor. For example, the camera sensor or the accelerometer sensor whose values are dependent on the orientation on the direction and so on direction in which the sensor is being put and the weight is measuring. Scalar sensors measure only the magnitude physical quantities, such as temperature colour, pressure, strain etcetera. These are scalar quantities and measurement of the change of magnitude is sufficient to convey the information.
On the other hand, vector sensors produce output signal of the voltage which is generally proportional to the magnitude as well as the direction and orientation of the quantity that is being measured. So, physical quantities such as the sound, image, velocity, acceleration orientation, these are all vector quantities and their measurement is not just dependent on the magnitude, but also on the direction. So, for example, accelerometer sensor, they give outputs in three dimensions x, y and z coordinate axis.

Some of the types of sensors:

1) Temperature Sensors

- Temperature sensors measure the amount of heat energy in a source, allowing them to detect temperature changes and convert these changes to data. Machinery used in manufacturing often requires environmental and device temperatures to be at specific levels. Similarly, within agriculture, soil temperature is a key factor for crop growth.

2) Humidity Sensors

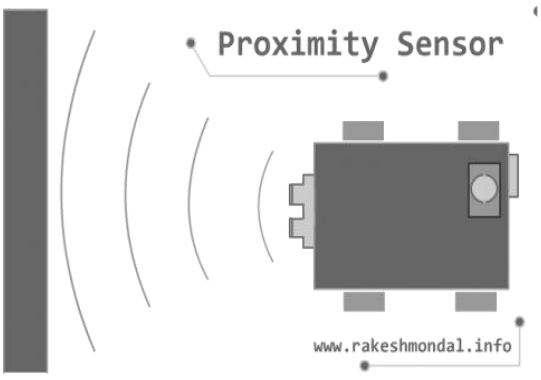
- These types of sensors measure the amount of water vapor in the atmosphere of air or other gases. Humidity sensors are commonly found in heating, vents and air conditioning (HVAC) systems in both industrial and residential domains. They can be found in many other areas including hospitals, and meteorology stations to report and predict weather.

3). Pressure Sensors

- A pressure sensor senses changes in gases and liquids. When the pressure changes, the sensor detects these changes, and communicates them to connected systems. Common use cases include leak testing which can be a result of decay. Pressure sensors are also useful in the manufacturing of water systems as it is easy to detect fluctuations or drops in pressure.

5. Proximity Sensors

- Proximity sensors are used for non-contact detection of objects near the sensor. These types of sensors often emit electromagnetic fields or beams of radiation such as infrared. Proximity sensors have some interesting use cases. In retail, a proximity sensor can detect the motion between a customer and a product in which he or she is interested. The user can be notified of any discounts or special offers of products located near the sensor. Proximity sensors are also used in the parking lots of malls, stadiums and airports to indicate parking availability. They can also be used on the assembly lines of chemical, food and many other types of industries.



6. Level Sensors

- Level sensors are used to detect the level of substances including liquids, powders and granular materials. Many industries including oil manufacturing, water treatment and beverage and food manufacturing factories use level sensors. Waste management systems provide a common use case as level sensors can detect the level of waste in a garbage can or dumpster.

7. Accelerometers

- Accelerometers detect an object's acceleration i.e. the rate of change of the object's velocity with respect to time. Accelerometers can also detect changes to gravity. Use cases for accelerometers include smart pedometers and monitoring driving fleets. They can also be used as anti-theft protection alerting the system if an object that should be stationary is moved.

8. Gyroscope

- Gyroscope sensors measure the angular rate or velocity, often defined as a measurement of speed and rotation around an axis. Use cases include automotive, such as car navigation and electronic stability control (anti-skid) systems. Additional use cases include motion sensing for video games, and camera-shake detection systems.

9. Gas Sensors

- These types of sensors monitor and detect changes in air quality, including the presence of toxic, combustible or hazardous gasses. Industries using gas sensors include mining, oil and gas, chemical research and manufacturing. A common consumer use case is the familiar carbon dioxide detectors used in many homes**.**

10. Infrared Sensors

- These types of sensors sense characteristics in their surroundings by either emitting or detecting infrared radiation. They can also measure the heat emitted by objects. Infrared sensors are used in a variety of different IoT projects including healthcare as they simplify the monitoring of blood flow and blood pressure.
- Televisions use infrared sensors to interpret the signals sent from a remote control. Another interesting application is that of art historians using infrared sensors to see hidden layers in paintings to help determine whether a work of art is original or fake or has been altered by a restoration process.

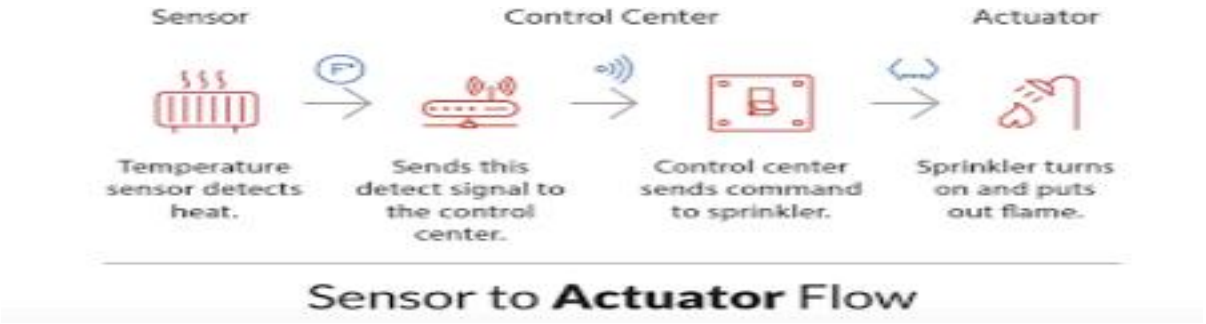
11. Optical Sensors

Optical sensors convert rays of light into electrical signals. There are many applications and use cases for optical sensors. In the auto industry, vehicles use optical sensors to recognize signs, obstacles, and other

things that a driver would notice when driving or parking. Optical sensors play a big role in the development of driverless cars. Optical sensors are very common in smart phones. For example, ambient light sensors can extend battery life. Optical sensors are also used in the biomedical field including breath analysis and heart-rate monitors**.**

Actuators:

- An IoT device is made up of a Physical object ("thing") + Controller ("brain") + Sensors + Actuators + Networks (Internet). An actuator is a machine component or system that moves or controls the mechanism or the system. Sensors in the device sense the environment, then control signals are generated for the actuators according to the actions needed to perform.
- A servo motor is an example of an actuator. They are linear or rotatory actuators, can move to a given specified angular or linear position. We can use servo motors for IoT applications and make the motor rotate to 90 degrees, 180 degrees, etc., as per our need.
- The following diagram shows what actuators do; the controller directs the actuator based on the sensor data to do the work.

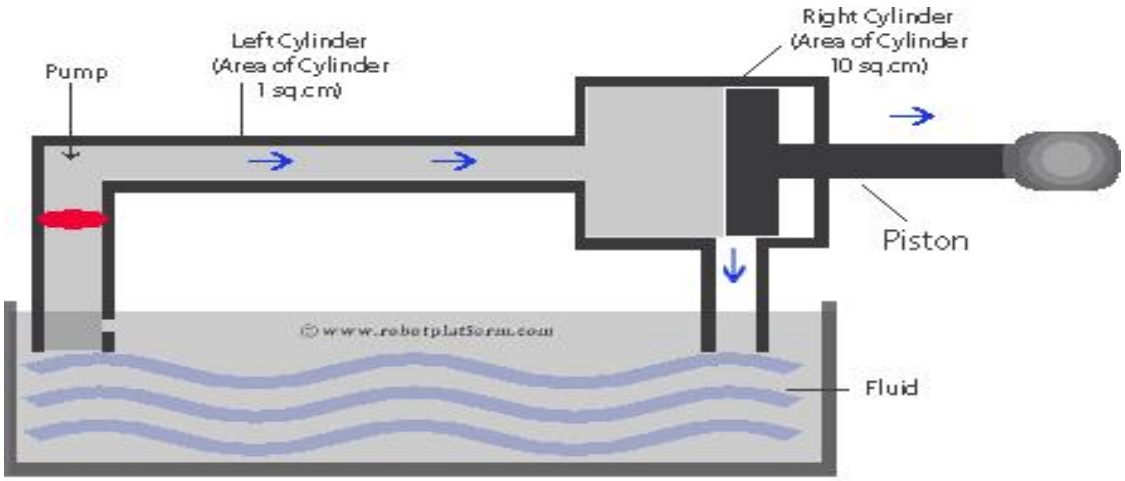


- The control system acts upon an environment through the actuator. It requires a source of energy and a control signal. When it receives a control signal, it converts the source of energy to a mechanical operation. On this basis, on which form of energy it uses, it has different types given below.

Types of Actuators:

Hydraulic Actuators –

A hydraulic actuator uses hydraulic power to perform a mechanical operation. They are actuated by a cylinder or fluid motor. The mechanical motion is converted to rotary, linear, or oscillatory motion, according to the need of the IoT device. Example- construction equipment uses hydraulic actuators because hydraulic actuators can generate a large amount of force. So, this name suggests, these hydraulic actuators consist of a cylinder or fluid motor that uses hydraulic power to facilitate mechanical operation. The mechanical motion is converted to linear rotary or oscillatory motion. Basically when some fluid passes through, then you know that motion is converted to some linear motion or some oscillatory motion or rotary motion and since liquids are nearly impossible to compress, most of the hydraulic actuators basically exert considerable force which is the reason why liquid based actuators are typically used and these are quite popular because of this particular reason.

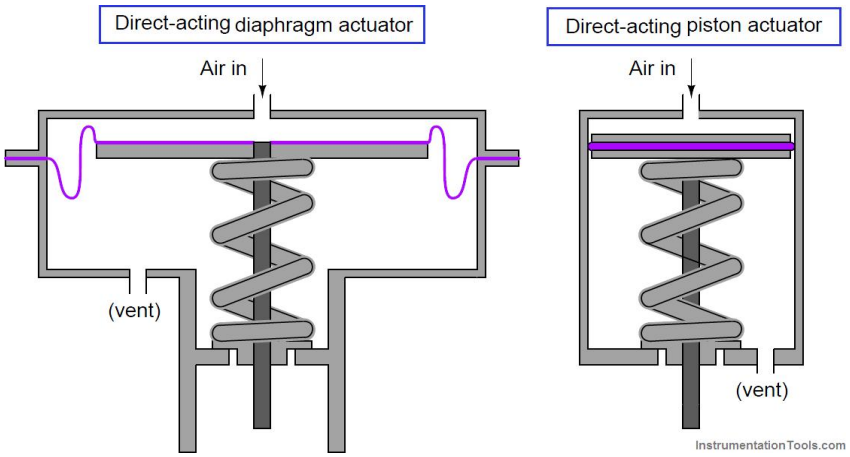


Hydraulic Actuator

Pneumatic Actuators –

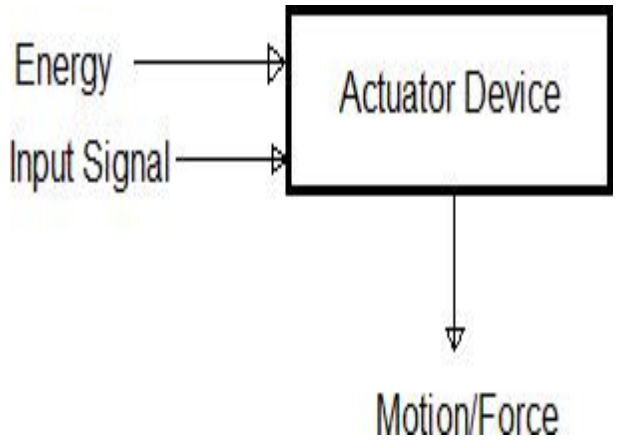
A pneumatic actuator uses energy formed by vacuum or compressed air at high pressure to convert into either linear or rotary motion. Example- Used in robotics, use sensors that work like human fingers by using compressed air. Pneumatic actuator, pneumatic means air based. A pneumatic actuator basically converts the energy formed by vacuum or compressed air at high pressure into either linear or rotatory motion. Pneumatic actuators basically exert a lot of force and for example, the pneumatic brakes can be very responsive to small changes in pressure that are applied by the driver.

Pneumatic brakes are quite common in different devices like trucks etc. They use pneumatic brakes. So, hydraulic brakes are more common in cars, in trucks pneumatic brakes are quite common. The advantage of pneumatic brakes, is that they are very responsive to small changes.



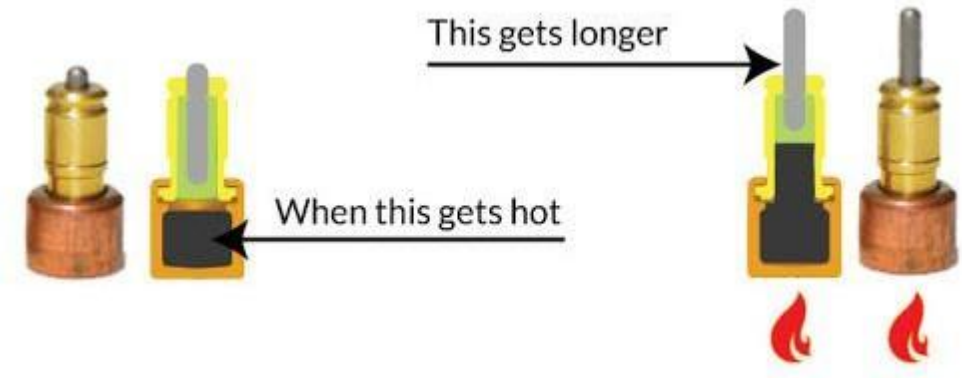
Electrical Actuators –

An electric actuator uses electrical energy, is usually actuated by a motor that converts electrical energy into mechanical torque. An example of an electric actuator is a solenoid based electric bell. An electric actuator is generally powered by a motor that converts electrical energy into mechanical torque. So, this electrical energy is used to actuate the equipment, such as the solenoid valve which control the flow of water in pipes in response to electrical signals.



Thermal /Magnetic Actuators –

- Actuators are simply devices used to transform energy into motion. A thermal actuator is a type of non-electric motor made of components such as a piston and a thermal sensitive material capable of producing linear motion in response to temperature changes.



- Magnetic Actuators: Magnetic Actuators use magnetic effects to generate forces which impact on the motion of a part in the actuator.

Mechanical Actuators –

- A mechanical actuator executes movement by converting rotary motion into linear motion. It involves pulleys, chains, gears, rails, and other devices to operate.

Communication modules:

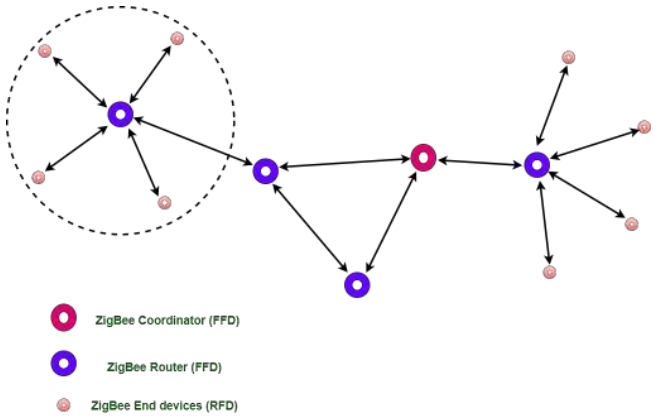
Zigbee Architecture

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)
- Network Join Time (~ 30 msec)
- 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.
- 3 frequency bands with 27 channels.

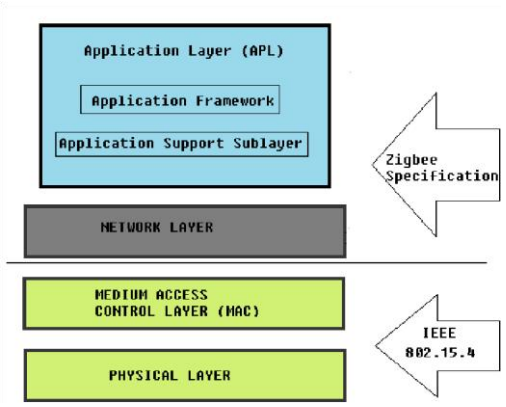
Operating Frequency Bands (Only one channel will be selected for use in a network):

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

1. Application Layer
2. Network Layer
3. Medium Access Control Layer
4. 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.

Channel Access:

1. **Contention Based Method** (Carrier-Sense Multiple Access With Collision Avoidance Mechanism)
2. **Contention Free Method** (Coordinator dedicates a specific time slot to each device (Guaranteed Time Slot (GTS)))

Zigbee Applications:

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

LoRa and LoRaWAN

The LoRaWAN protocol is a Low Power Wide Area Networking (LPWAN) communication protocol that functions on LoRa. The LoRaWAN specification is open so anyone can set up and operate a LoRa network.

LoRa is a wireless audio frequency technology that operates in a license-free radio frequency spectrum. LoRa is a physical layer protocol that uses spread spectrum modulation and supports long-range communication at the cost of a narrow bandwidth. It uses a narrow band waveform with a central frequency to send data, which makes it robust to interference.

Characteristics of LoRaWAN technology

- Long range communication up to 10 miles in line of sight.
- Long battery duration of up to 10 years. For enhanced battery life, you can operate your devices in class A or class B mode, which requires increased downlink latency.
- Low cost for devices and maintenance.
- License-free radio spectrum but region-specific regulations apply.
- Low power but has a limited payload size of 51 bytes to 241 bytes depending on the data rate. The data rate can be 0,3 Kbit/s – 27 Kbit/s data rate with a 222 maximal payload size.

Advantages:

8. Long Range: LoRaWAN can provide long-range communication, spanning several kilometers in urban areas and even greater distances in rural environments. This long-range capability is a significant advantage for applications that require wide-area coverage.

9. Low Power Consumption: IoT devices using LoRaWAN can operate on batteries for an extended period, often several years, before needing a battery replacement or recharge. This low power consumption is crucial for remote and battery-powered devices.

10. Scalability: LoRaWAN networks are highly scalable, allowing for the addition of a large number of devices to a single network without significant infrastructure changes.

11. Cost-Efficiency: Due to its low power requirements and long-range capabilities, LoRaWAN can be a cost-effective solution for many IoT applications. It reduces the need for frequent battery replacements and complex power infrastructure.

12. License-Free Spectrum: LoRaWAN operates in unlicensed ISM radio bands, reducing regulatory and licensing requirements. This simplifies deployment and lowers operational costs.

13. Wide Adoption: LoRaWAN has gained widespread adoption and support from various companies and organizations, creating a thriving ecosystem of devices, gateways, and network providers.

14. Security Features: LoRaWAN includes security features such as encryption and device authentication to protect data transmitted between devices and the network.

8. Use Cases: LoRaWAN is suitable for a wide range of IoT use cases, including smart agriculture, smart cities, industrial IoT, asset tracking, and environmental monitoring, among others.

Disadvantages:

15. Low Data Rate: LoRaWAN is designed for low-data-rate applications. If you need to transmit large amounts of data quickly, it may not be the best choice.

16. Limited Bandwidth: LoRaWAN networks have limited available bandwidth, which can lead to network congestion in areas with a high density of devices.

17. Latency: LoRaWAN is optimized for low power and long range, which can result in higher latency compared to other wireless technologies. This may not be suitable for applications requiring real-time data transmission.

18. Interference: In crowded radio frequency environments, interference from other devices operating in the same frequency bands can affect LoRaWAN communication quality.

19. Not Suitable for High Mobility: LoRaWAN is designed for stationary or slowly moving devices. It may not be suitable for applications that require high mobility, such as asset tracking on fast-moving vehicles.

20. Initial Deployment Complexity: Setting up a LoRaWAN network can be more complex than other wireless technologies, as it requires the installation of gateways and configuration of network servers.

21. Dependence on Gateways: LoRaWAN devices rely on gateways to relay data to the network. If gateways are unavailable or experience issues, communication can be disrupted.

8. Limited Use Cases: While LoRaWAN is versatile, it may not be the best choice for all IoT applications, especially those that require high bandwidth, low latency, or high mobility.

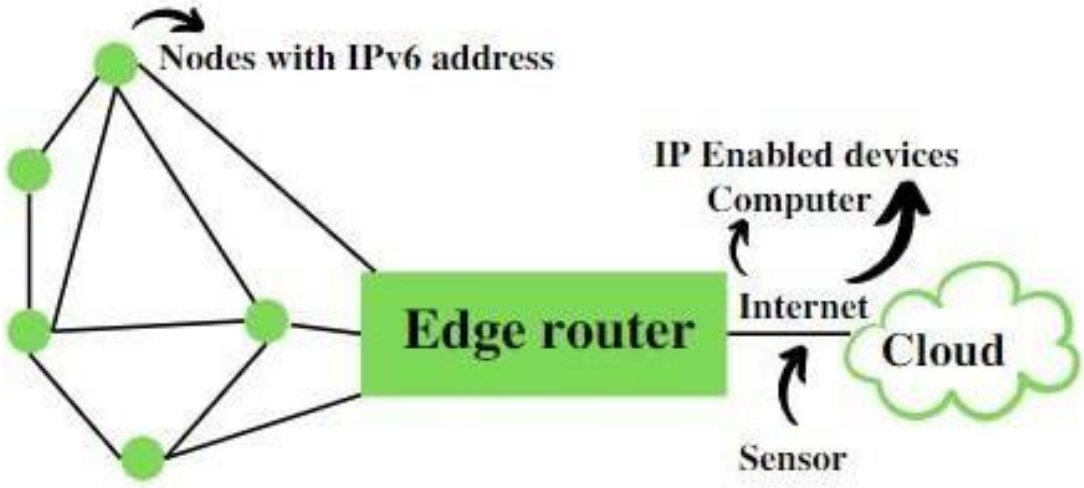
6LowPAN

6LoWPAN is an IPv6 protocol, and It's extended from is IPv6 over Low Power Personal Area Network. As the name itself explains the meaning of this protocol is that this protocol works on Wireless Personal Area Network i.e., WPAN.

WPAN is a Personal Area Network (PAN) where the interconnected devices are centered around a person's workspace and connected through a wireless medium. You can read more about WPAN at WPAN. 6LoWPAN allows communication using the IPv6 protocol. IPv6 is Internet Protocol Version 6 is a network layer protocol that allows communication to take place over the network. It is faster and more reliable and provides a large number of addresses.

6LoWPAN initially came into existence to overcome the conventional methodologies that were adapted to transmit information. But still, it is not so efficient as it only allows for the smaller devices with very limited processing ability to establish communication using one of the Internet Protocols, i.e., IPv6. It has very low cost, short-range, low memory usage, and low bit rate.

It comprises an Edge Router and Sensor Nodes. Even the smallest of the IoT devices can now be part of the network, and the information can be transmitted to the outside world as well. For example, LED Streetlights.



- It is a technology that makes the individual nodes IP enabled.
- 6LoWPAN can interact with 802.15.4 devices and also other types of devices on an IP Network. For example, Wi-Fi.
- It uses AES 128 link layer security, which AES is a block cipher having key size of 128/192/256 bits and encrypts data in blocks of 128 bits each. This is defined in IEEE 802.15.4 and provides link authentication and encryption.

Basic Requirements of 6LoWPAN:

1. The device should be having sleep mode in order to support the battery saving.
2. Minimal memory requirement.
3. Routing overhead should be lowered.

Features of 6LoWPAN:

1. It is used with IEEE 802.15,.4 in the 2.4 GHz band.
2. Outdoor range: ~200 m (maximum)
3. Data rate: 200kbps (maximum)
4. Maximum number of nodes: ~100

Advantages of 6LoWPAN:

1. 6LoWPAN is a mesh network that is robust, scalable, and can heal on its own.
2. It delivers low-cost and secure communication in IoT devices.
3. It uses IPv6 protocol and so it can be directly routed to cloud platforms.
4. It offers one-to-many and many-to-one routing.
5. In the network, leaf nodes can be in sleep mode for a longer duration of time.

Disadvantages of 6LoWPAN:

1. It is comparatively less secure than Zigbee.
2. It has lesser immunity to interference than that Wi-Fi and Bluetooth.
3. Without the mesh topology, it supports a short range.

Applications of 6LoWPAN:

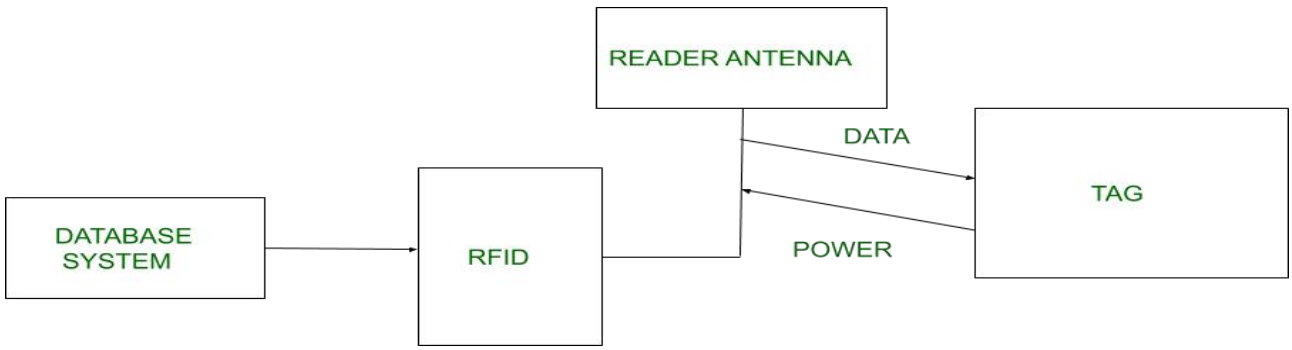
1. It is a wireless sensor network.
2. It is used in home-automation,
3. It is used in smart agricultural techniques, and industrial monitoring.

Security and Interoperability with 6LoWPAN:

- **Security**: 6LoWPAN security is ensured by the AES algorithm, which is a link layer security, and the transport layer security mechanisms are included as well.
- **Interoperability**: 6LoWPAN is able to operate with other wireless devices as well which makes it interoperable in a network.

Radio Frequency Identification (RFID)

Radio Frequency Identification (RFID) is a form of wireless communication that incorporates the use of electromagnetic or electrostatic coupling in the radio frequency portion of the electromagnetic spectrum to uniquely identify an object, animal or person. It uses radio frequency to search ,identify, track and communicate with items and people. it 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 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.



Kinds of RFID :

There are many kinds of RFID, each with different properties, but perhaps the most fascinating aspect of RFID technology is that most RFID tags have neither an electric plug nor a battery. Instead, all of the energy needed to operate them is supplied in the form of radio waves by RFID readers. This technology is called passive RFID to distinguish it from the(less common) active RFID in which there is a power source on the tag.

UHF RHID (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 –**

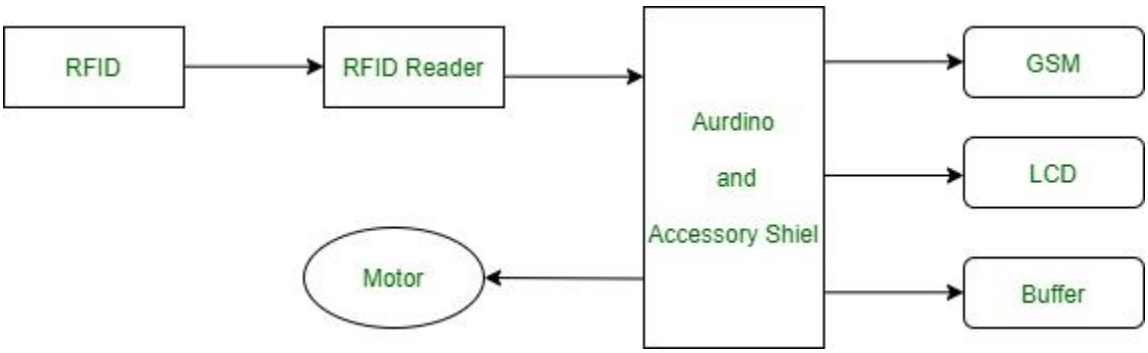
   Passive RFID tags does not have their own power source. It uses power from the reader. In this device, RF tags are not attached by a power supply and passive RF tag stored their power. When it is emitted from active antennas and the RF tag are used specific frequency like 125-134KHZ as low frequency, 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. means, active tag uses a power source like battery. It has it's own power source, does not require power from source/reader.

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.



Working of RFID System :

Every RFID system consists of three components: a scanning antenna, a transceiver and a transponder. When the scanning antenna and transceiver are combined, they are referred to as an RFID reader or interrogator. There are two types of RFID readers — fixed readers and mobile readers. The RFID reader is a network-connected device that can be portable or permanently attached. It uses radio waves to transmit signals that activate the tag. Once activated, the tag sends a wave back to the antenna, where it is translated into data.

The transponder is in the RFID tag itself. The read range for RFID tags varies based on factors including the type of tag, type of reader, RFID frequency and interference in the surrounding environment or from other RFID tags and readers. Tags that have a stronger power source also have a longer read range.

Features of RFID :

- An RFID tag consists of two-part which is an microcircuit and an antenna.
- This tag is covered by protective material which acts as a shield against the outer environment effect.
- This tag may active or passive in which we mainly and widely used passive RFID.

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.
- Controlling access to restricted areas.
- It uses ID badging.
- Supply chain management.
- Counterfeit prevention (e.g., in the pharmaceutical industry).

Advantages of RFID :

- It provides data access and real-time information without taking to much time.
- RFID tags follow the instruction and store a large amount of information.
- The RFID system is non-line of sight nature of the technology.
- It improves the Efficiency, traceability of production.
- In RFID hundred of tags read in a short time.

Disadvantages of RFID :

- It takes longer to program RFID Devices.
- RFID intercepted easily even it is Encrypted.
- In an RFID system, there are two or three layers of ordinary household foil to dam the radio wave.

- There is privacy concern about RFID devices anybody can access information about anything.
- Active RFID can costlier due to battery.

Wi-Fi

- **Wi-Fi**, a brand name given by the Wi-Fi Alliance (formerly Wireless Ethernet Compatibility Alliance), is a generic term that refers to the communication standard for the wireless network which works as a Local Area Network to operate without using the cable and any types of wiring.
- It is known as **WLAN**. The communication standard is **IEEE 802.11**. Wi-Fi works using Physical Data Link Layer.
- Nowadays in all mobile computing devices such as laptops, mobile phones, also digital cameras, smart TVs has the support of Wi-Fi. The Wi-Fi connection is established from the access point or base station to the client connection or any client-to-client connection within a specific range, the range depends on the router which provides the radio frequency through Wi-Fi. These frequencies operate on 2 types of bandwidth at present, 2.4 GHz and 5 GHz.

All the modern laptops and mobiles are capable of using both bandwidths, it depends on the Wi-Fi adapter which is inside the device to catch the Wi-Fi signal. 2.4 GHz is the default bandwidth supported by all the devices. 2.4 GHz can cover a big range of areas to spread the Wi-Fi signal but the frequency is low, so in simple words, the speed of the internet is less and 5 GHz bandwidth is for a lower range of area but the frequency is high so the speed is very high.

Let's say, if there is an internet connection of 60 MB/s bandwidth, then for 2.4 GHz bandwidth, it provides approx 30 to 45 MB/s of bandwidth connection and for 5 GHz bandwidth, it provides approx 50 to 57 MB/s bandwidth.

History:

The concept of Wi-Fi is very old but its implementation is not so old. At first **ALOHA System** is a wireless network system that is used to connect Hawaii island via a network in the year 1971. Where the protocol is used for this was ALOHA protocol and the network used packet transfer. Later it's converted to IEEE 802.11 protocol.
Then in 1985, the Federal Communications Commission (FCC) released a new network for general uses which works on 900 Mhz, 2.4 GHz, and 5.8 GHz bandwidth. This is known as the *ISM band*. Also, IBM introduced a *Token Ring LAN* network for connecting several computers, it can transfer data at 4 Mb/s speed. Then in 1988, a wireless cashier system was invented based on the Token Ring LAN network known as *waveLAN*, it operates at 900MHz or 2.4 GHz band and offers speeds of 1 to 2 Mbps. Then it was converted to *IEEE 802.11LAN/MAN* standards in 1989. ?Then in 1990, IEEE 802.11 Working Group for Wireless LANs is established by **Vic Hayes**, who was known as the **"Father of WiFi"**.
Then in 1994, *Dr. Alex Hills* introduced a research project on the wireless network, which provided coverage of the network to 7 buildings wirelessly.
Then in 1996 *Commonwealth Scientific and Industrial Research Organization (CSIRO)* introduced a wireless network based on the same protocol 802.11, later it was known as IEEE 802.11a standards.
Then after all this in 1997 the first version of Wi-Fi is released officially which is 802.11 and it can support a maximum of 2 Mb/s link speed. Then in 1999, the link speed is increased to 11 Mb/s over the 2.4 GHz frequency band, this version is known as *802.11b*
Then after a month, the IEEE 802.11a standard is approved officially, which provides up to 54 Mb/s link speed over the 5 GHz band, but the signal range is weaker than the 2.4 GHz band.

Then in 2003, the speed is increased in a new version, known as *802.11g*. The speed offers up to 54 to 108 Mb/s over 2.4 GHz.
After this two more versions were introduced that are, *802.11i and 802.11e*. In 802.11i, the security mechanism was increased and in 802.11e, Voice over Wireless LAN and multimedia streaming are involved.
Then in 2009, 802.11n is developed, which supports both 2.4 GHz and 5 GHz radiofrequency. And these are used simultaneously by dual-band routers and can reach maximum speeds of 600 Mbps.

Then in 2014, a new version was introduced that offers a potential speed of 1733 Mb/s in the 5 GHz band. This version is known as *802.11ac*. Till now this is the latest version of Wi-Fi.

Applications of Wi-Fi :

Wi-Fi has many applications, it is used in all the sectors where a computer or any digital media is used, also for entertaining Wi-Fi is used. Some of the applications are mentioned below –

- Accessing Internet: Using Wi-Fi we can access the internet in any Wi-Fi-capable device wirelessly.
- We can stream or cast audio or video wirelessly on any device using Wi-Fi for our entertainment.
- We can share files, data, etc between two or more computers or mobile phones using Wi-Fi, and the speed of the data transfer rate is also very high. Also, we can print any document using a Wi-Fi printer, this is very much used nowadays.
- We can use Wi-Fi as **HOTSPOTS** also, it points Wireless Internet access for a particular range of area. Using Hotspot the owner of the main network connection can offer temporary network access to Wi-Fi-capable devices so that the users can use the network without knowing anything about the main network connection. Wi-Fi adapters are mainly spreading radio signals using the owner network connection to provide a hotspot.
- Using Wi-Fi or WLAN we can construct simple wireless connections from one point to another, known as Point to point networks. This can be useful to connect two locations that are difficult to reach by wire, such as two buildings of corporate business.
- One more important application is **VoWi-Fi**, which is known as **voice-over Wi-Fi**. Some years ago telecom companies are introduced VoLTE (Voice over Long-Term Evolution). Nowadays they are introduced to VoWi-Fi, by which we can call anyone by using our home Wi-Fi network, only one thing is that the mobile needs to connect with the Wi-Fi. Then the voice is transferred using the Wi-Fi network instead of using the mobile SIM network, so the call quality is very good. Many mobile phones are already getting the support of VoWi-Fi.
- Wi-Fi in offices: In an office, all the computers are interconnected using Wi-Fi. For Wi-Fi, there are no wiring complexities. Also, the speed of the network is good. For Wi-Fi, a project can be presented to all the members at a time in the form of an excel sheet, ppt, etc. For Wi-Fi, there is no network loss as in cable due to cable break.
- Also using W-Fi a whole city can provide network connectivity by deploying routers at a specific area to access the internet. Already schools, colleges, and universities are providing networks using Wi-Fi because of its flexibility.
- Wi-Fi is used as a *positioning system* also, by which we can detect the positions of Wi-Fi hotspots to identify a device location.

Types of Wi-Fi:

Wi-Fi has several types of standards, which are discussed earlier, here just the name of the standards are defined,

Standards	Year of Release	Description
Wi-Fi-1 (802.11b)	**1999**	**This version has a link speed from 2Mb/s to 11 Mb/s over a 2.4 GHz frequency band**
Wi-Fi-2 (802.11a)	**1999**	**After a month of release previous version, 802.11a was released and it provide up to 54 Mb/s link speed over 5 Ghz band**
Wi-Fi-3 (802.11g)	**2003**	**In this version the speed was increased up to 54 to 108 Mb/s over 2.4 GHz**

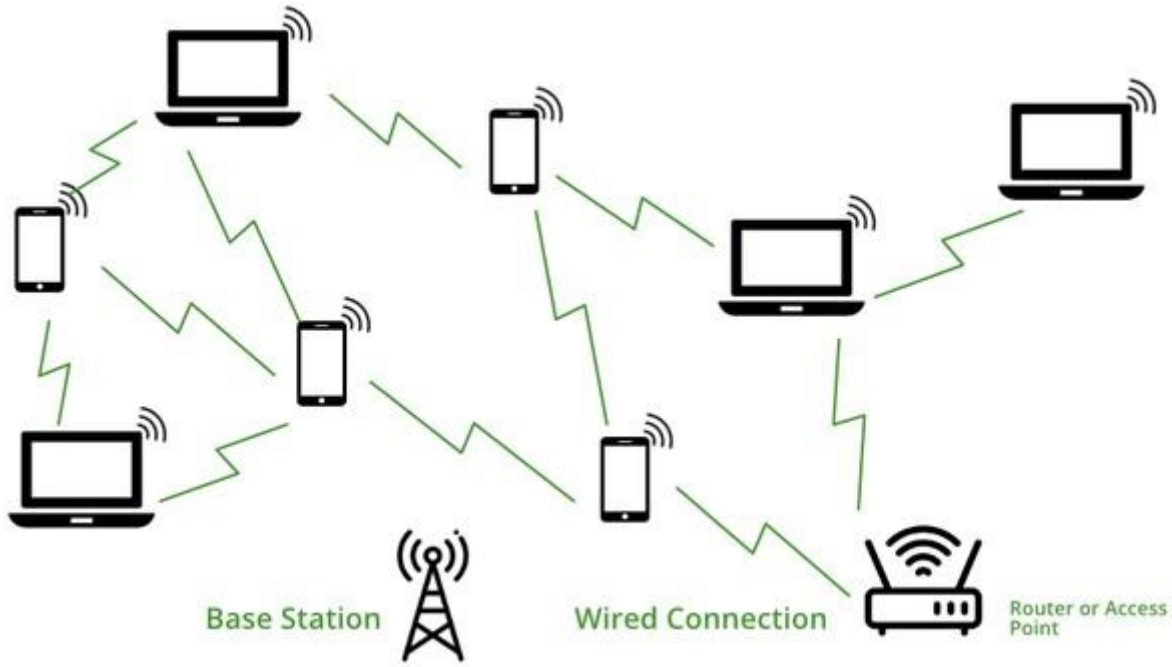
802.11i	**2004**	**This is the same as 802.11g but only the security mechanism was increased in this version**
802.11e	**2004**	**This is also the same as 802.11g, only Voice over Wireless LAN and multimedia streaming are involved**
Wi-Fi-4 (802.11n)	**2009**	**This version supports both 2.4 GHz and 5 GHz radio frequency and it offers up to 72 to 600 Mb/s speed**
Wi-Fi-5 (802.11ac)	**2014**	**It supports a speed of 1733 Mb/s in the 5 GHz band**

A new version will release in 2020 named *802.11ax* developed by **Huawei**, which can support, a maximum of 3.5 Gb/s. it will know **Wi-Fi 6**.

How does Wi-Fi work?

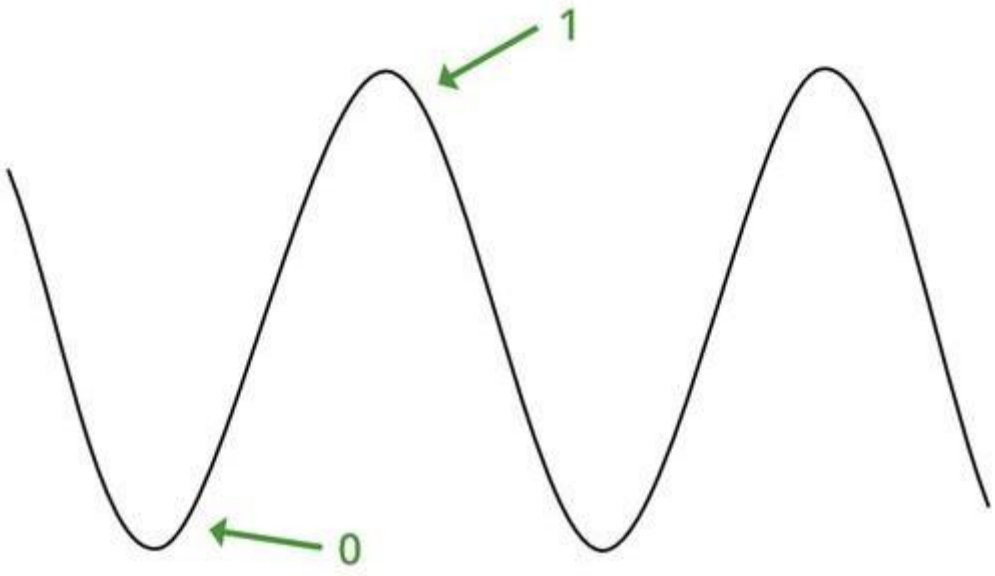
Wi-Fi is a wireless technology for networking, so it uses Electromagnetic waves to transmit networks. We know that there are many divisions of Electromagnetic waves according to their frequency such as X-ray, Gamma-ray, radio wave, microwave, etc, in Wi-Fi, the radio frequency is used. For transmitting Wi-Fi signal there is three medium,

- **Base station network or an Ethernet(802.3) connection:** It is the main host network from where the network connection is provided to the router.
- **Access point or router:** it is a bridge between a wired network and a wireless network. It accepts a wired Ethernet connection and converts the wired connection to a wireless connection and spreads the connection as a radio wave.
- **Accessing devices:** It is our mobile, computer, etc from where we use the Wi-Fi and surfing internet.



Working of Wi-Fi

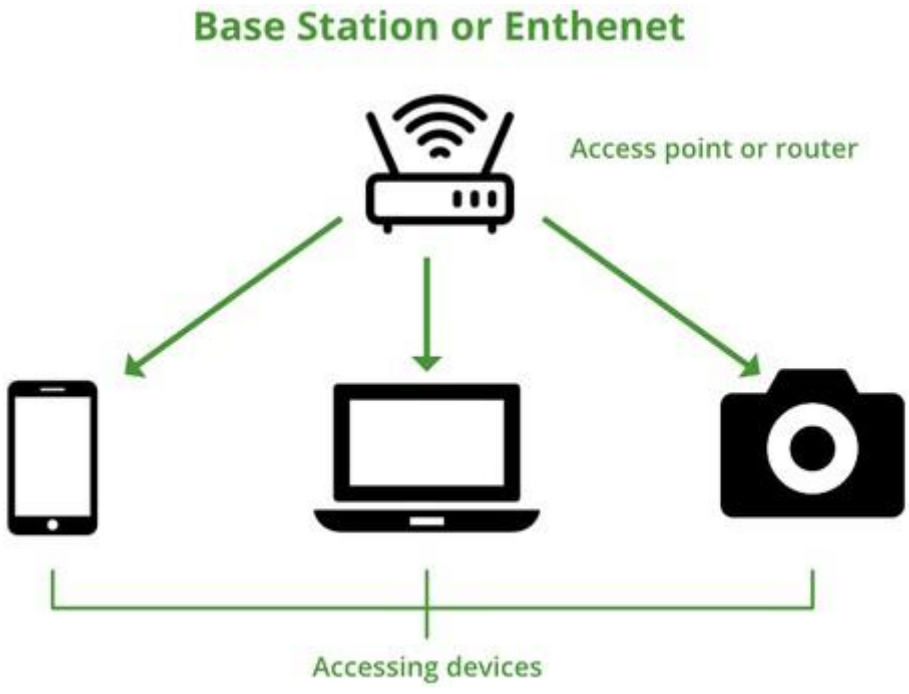
All the electronics devices read data in binary form, also router or our devices, here routers provide radio waves and those waves are receive by our devices and read the waves in binary form. We all know how a wave looks like, the upper pick of the wave is known as 1 and the lower pick of the wave is known as 0 in binary. Like below:



*Data transmission*Some more terminologies

- **SSID (Service Set Identifier)**: It is a 32 character name that identifies the Wi-Fi network and differentiates one Wi-Fi from another Wi-Fi. All the devices are attempting to connect a particular SSID. Simply, SSID is the name of the wireless network.
- **WPA-PSK (Wi-Fi Protected Access- Pre-Shared Key):** It is a program developed by the Wi-Fi Alliance Authority to secure wireless networks with the use of Pre-Shared Key(PSK) authentication. WPA has 3 types, such as WPA. WPA2, WPA3. It is a way of encrypting the Wi-Fi signal to protect from unwanted users.
- Wi-Fi uses **Ad-Hoc** networks to transmit. It is a point-to-point network without any interface.

How signals are reached to our devices?



Base Station

Advantages of Wi-Fi

- It is a flexible network connection, no wiring complexities. Can be accessed from anywhere in the Wi-Fi range.
- It does not require regulatory approval for individual users.
- It is salable, can be expanded by using Wi-Fi Extenders.
- It can be set up in an easy and fast way. Just need to configure the SSID and Password.
- Security in a high in Wi-Fi network, its uses **WPA** encryption to encrypt radio signals.
- It is also lower in cost.
- It also can provide Hotspots.
- it supports roaming also.

Disadvantages of Wi-Fi

- Power consumption is high while using Wi-Fi in any device which has a battery, such as mobile, laptops, etc.
- Many times there may be some security problems happening even it has encryption. Such as many times has known devices become unknown to the router, Wi-Fi can be hacked also.
- Speed is slower than a direct cable connection.
- It has lower radiation like cell phones, so it can harm humans.
- Wi-Fi signals may be affected by climatic conditions like thunderstorms.
- Unauthorized access to Wi-Fi can happen because it does not have a firewall.
- To use Wi-Fi we need a router, which needs a power source, so at the time of power cut, we cannot access the internet.

Power Source

- Power sources in IoT (Internet of Things) devices are a critical consideration as they directly impact the device's functionality, longevity, and deployment options.

Here are some important notes about power sources in IoT:

1. Battery Power:

- **Advantages:**
  - Portability: Battery-powered IoT devices are highly portable and can be placed virtually anywhere without the need for a power outlet.
  - Low Maintenance: Batteries can provide power for extended periods (months to years) without the need for frequent maintenance.

- **Disadvantages:**
  - Limited Lifetime: Batteries have a finite lifespan, and their replacement or recharging can be costly and impractical for certain deployments.
  - Size and Weight: Batteries can add bulk and weight to IoT devices, which may not be suitable for small or lightweight applications.
  - Environmental Impact: Battery disposal and the environmental impact of disposable batteries are concerns.

2. Solar Power:

- **Advantages:**

  - Renewable Energy: Solar panels harness energy from the sun, providing a renewable and eco-friendly power source.

- Extended Lifespan: Solar-powered IoT devices can operate for extended periods without the need for battery replacement.

- Disadvantages:

- Sunlight Dependency: Solar power is dependent on sunlight, which can be limited in certain geographic locations or during cloudy days.
- Initial Costs: Solar panel installation can have high upfront costs, although it can lead to long-term savings.

3. Energy Harvesting:

- Advantages:

- Energy from the Environment: Energy harvesting technologies, such as vibration, thermal, or kinetic energy, allow IoT devices to capture energy from their environment.
- Continuous Operation: When implemented effectively, energy harvesting can enable continuous device operation without the need for battery replacement.

- Disadvantages:

- Variable Energy Availability: The availability of environmental energy sources can vary, making it challenging to ensure consistent device operation.
- Energy Storage: Energy harvested must be stored efficiently for later use, which may require specialized components.

4. Wired Power:

- Advantages:

- Reliable and Stable: Wired power sources, such as AC or DC power outlets, provide a stable and reliable source of energy.
- High Power Capacity: Wired connections can support high-power IoT devices and applications.

- Disadvantages:

- Limited Mobility: Devices relying on wired power sources are typically fixed and cannot be easily moved.
- Installation Complexity: Installing wired power connections may be labor-intensive and costly, especially in remote or outdoor locations.

5. Hybrid Power:

- Advantages:

- Combining Sources: Hybrid power systems can combine multiple power sources, such as batteries and solar panels, to provide redundancy and extended operation.
- Flexibility: Hybrid systems can adapt to changing environmental conditions and energy availability.

- Disadvantages:

- Complexity: Designing and managing hybrid power systems can be complex and may require specialized knowledge.

6. Ultra-Low Power Consumption:

- Reducing power consumption through efficient hardware design and software optimization is crucial for extending the lifespan of battery-powered IoT devices**.**

7. Energy-Efficient Communication Protocols:

- Choosing energy-efficient communication protocols like LoRaWAN or MQTT-SN can minimize the power required for data transmission.

8. Energy Monitoring and Management:

- Implementing energy monitoring and management features in IoT devices can help optimize power usage and extend battery life.

UNIT IV BUILDING IoT WITH CLOUD AND DATA ANALYTICS

IoT platforms – Arduino – Raspberry Pi - Cloud Computing in IoT - Cloud Connectivity - Big Data Analytics - Data Visualization

Iot Platform

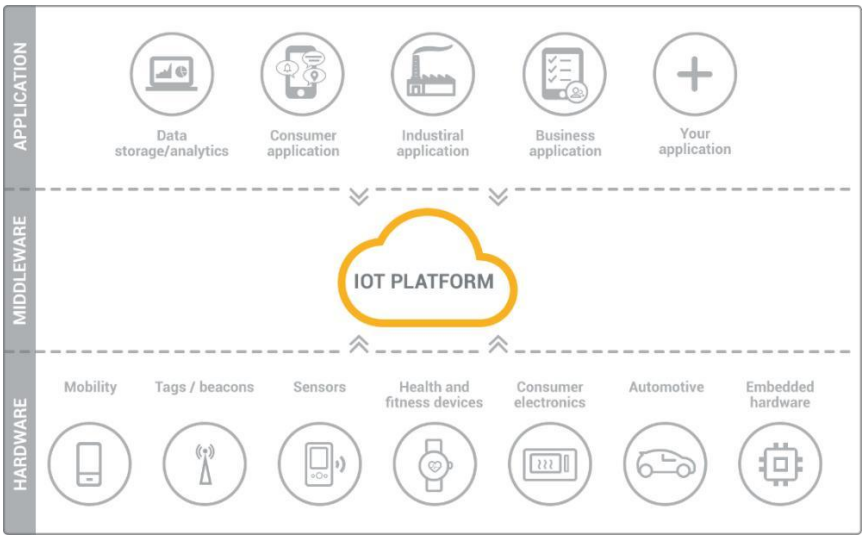
An IoT platform is a multi-layer technology that enables straightforward provisioning, management, and automation of connected devices within the Internet of Things universe. It basically connects your hardware, however diverse, to the cloud by using flexible connectivity options, enterprise-grade security mechanisms, and broad data processing powers. For developers, an IoT platform provides a set of ready-to-use features that greatly speed up development of applications for connected devices as well as take care of scalability and cross- device compatibility.

Thus, an IoT platform can be wearing different hats depending on how you look at it. It is commonly referred to as middleware when we talk about how it connects remote devices to user applications (or other devices) and manages all the interactions between the hardware and the application layers. It is also known as a cloud enablement platform or IoT enablement platform to pinpoint its major business value, that is empowering standard devices with cloud-based applications and services. Finally, under the name of the IoT application enablement platform, it shifts the focus to being a key tool for IoT developers.

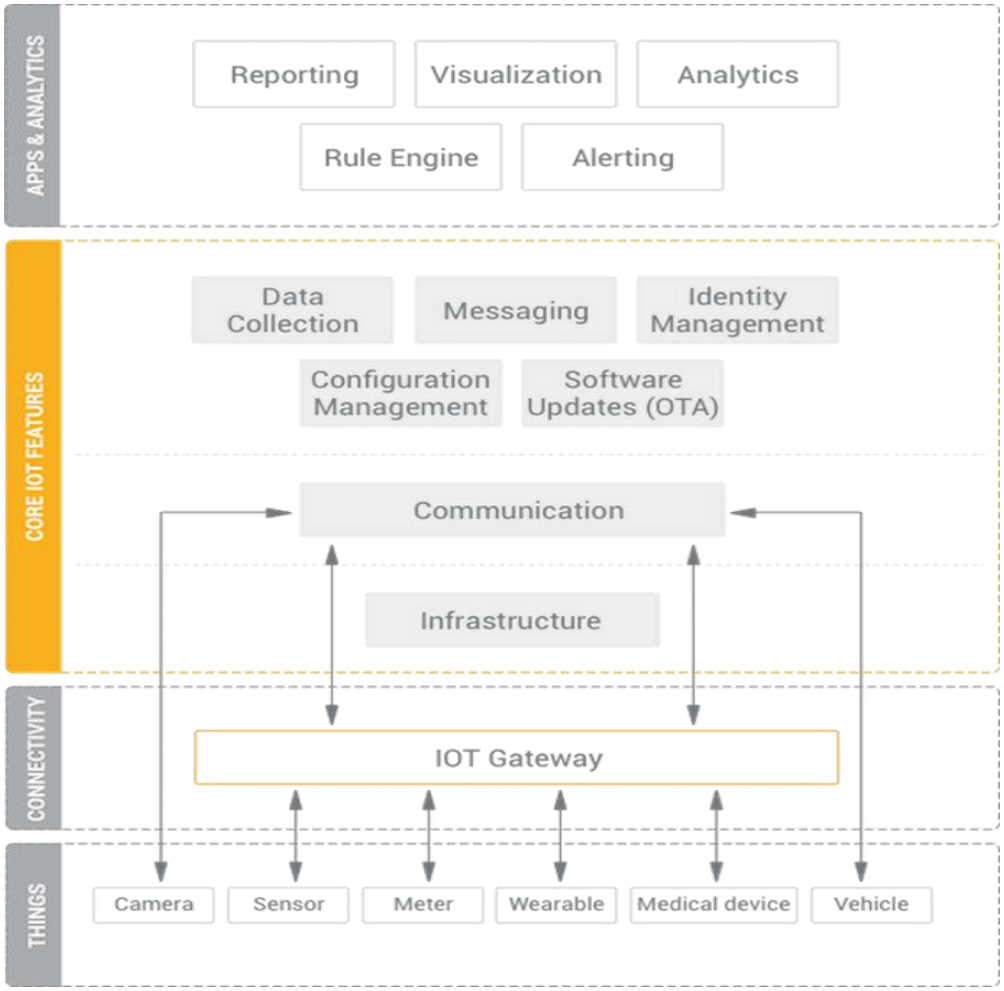
IoT platform as the middleware :

IoT platforms originated in the form of IoT middleware, which purpose was to function as a mediator between the hardware and application layers. Its primary tasks included data collection from the devices over different protocols and network topologies, remote device configuration and control, device management, and over-the-air firmware updates.

To be used in real-life heterogeneous IoT ecosystems, IoT middleware is expected to support integration with almost any connected device and blend in with third-party applications used by the device. This independence from underlying hardware and overhanging software allows a single IoT platform to manage any kind of connected device in the same straightforward way.



Modern IoT platforms go further and introduce a variety of valuable features into the hardware and application layers as well. They provide components for frontend and analytics, on-device data processing, and cloud-based deployment. Some of them can handle end-to-end IoT solution implementation from the ground up.



IoT platform technology stack

In the four typical layers of the IoT stack, which are things, connectivity, core IoT features, and applications & analytics, a top-of-the-range IoT platform should provide you with the majority of IoT functionality needed for developing your connected devices and smart things.

Your devices connect to the platform, which sits in the cloud or in your on-premises data center, either directly or by using an IoT gateway. A gateway comes useful whenever your endpoints aren't capable of direct cloud communication or, for example, you need some computing power on edge. You can also use an IoT gateway to convert protocols, for example, when your endpoints are in LoRaWan network but you need them to communicate with the cloud over MQTT.

An IoT platform itself can be decomposed into several layers. At the bottom there is the infrastructure level, which is something that enables the functioning of the platform. You can find here components for container management, internal platform messaging, orchestration of IoT solution clusters, and others.

The communication layer enables messaging for the devices; in other words, this is where devices connect to the cloud to perform different operations.

The following layer represents core IoT features provided by the platform. Among the essential ones are data collection, device management, configuration management, messaging, and OTA software updates.

Sitting on top of core IoT features, there is another layer, which is less related to data exchange

between devices but rather to processing of this data in the platform. There is reporting, which allows you to generate custom reports. There is visualization for data representation in user applications. Then, there are a rule engine, analytics, and alerting for notifying you about any anomalies detected in your IoT solution.

Importantly, the best IoT platforms allow you to add your own industry-specific components and third-party applications. Without such flexibility adapting an IoT platform for a particular business scenario could bear significant extra cost and delay the solution delivery indefinitely.

Advanced IoT platforms

There are some other important criteria that differentiate IoT platforms between each other, such as scalability, customizability, ease of use, code control, integration with 3rd party software, deployment options, and the data security level.

- **Scalable (cloud native)** – advanced IoT platforms ensure elastic scalability across any number of endpoints that the client may require. This capability is taken for granted for public cloud deployments but it should be specifically put to the test in case of an on-premises deployment, including the platform's load balancing capabilities for maximized performance of the server cluster.
- **Customizable** – a crucial factor for the speed of delivery. It closely relates to flexibility of integration APIs, louse coupling of the platform's components, and source code transparency. For small-scale, undemanding IoT solutions good APIs may be enough to fly, while feature-rich, rapidly evolving IoT ecosystems usually require developers to have a greater degree of control over the entire system, its source code, integration interfaces, deployment options, data schemas, connectivity and security mechanisms, etc.
- **Secure** – data security involves encryption, comprehensive identity management, and flexible deployment. End-to-end data flow encryption, including data at rest, device authentication, user access rights management, and private cloud infrastructure for sensitive data – this is the basics of how to avoid potentially compromising breaches in your IoT solution.

Cutting across these aspects, there are two different paradigms of IoT solution cluster deployment offered by IoT platform providers: a public cloud IoT PaaS and a self-hosted private IoT cloud.

IoT cloud enablement

An IoT cloud is a pinnacle of the IoT platforms evolution. Sometimes these two terms are used interchangeably, in which case the system at hand is typically an IoT platform-as-aservice (PaaS). This type of solution allows you to rent cloud infrastructure and an IoT platform all from a single technology provider. Also, there might be ready-to-use IoT solutions (IoT cloud services) offered by the provider, built and hosted on its infrastructure. However, one important capability of a modern IoT platform consists in a private IoT cloud enablement. As opposed to public PaaS solutions located at a provider's cloud, a private IoT cloud can be hosted on any cloud infrastructure, including a private data center. This type of deployment offers much greater control over the new features development, customization, and third-party integrations. It is also advocated for stringent data security and performance requirements.

Arduino is a prototype platform (open-source) based on an easy-to-use hardware and software. It consists of a circuit board, which can be programed (referred to as a microcontroller) and a ready-made software called Arduino IDE (Integrated Development Environment), which is used to write and upload the computer code to the physical board.

The key features are −

- Arduino boards are able to read analog or digital input signals from different sensors and turn it into an output such as activating a motor, turning LED on/off, connect to the cloud and many other actions.
- You can control your board functions by sending a set of instructions to the microcontroller on the board via Arduino IDE (referred to as uploading software).
- Unlike most previous programmable circuit boards, Arduino does not need an extra piece of hardware (called a programmer) in order to load a new code onto the board. You can simply use a USB cable.
- Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program.
- Finally, Arduino provides a standard form factor that breaks the functions of the micro-controller into a more accessible package.

Board Types

Various kinds of Arduino boards are available depending on different microcontrollers used. However, all Arduino boards have one thing in common: they are programed through the Arduino IDE.

The differences are based on the number of inputs and outputs (the number of sensors, LEDs, and buttons you can use on a single board), speed, operating voltage, form factor etc. Some boards are designed to be embedded and have no programming interface (hardware), which you would need to buy separately. Some can run directly from a 3.7V battery, others need at least 5V.

Here is a list of different Arduino boards available.

Arduino boards based on ATMEGA328 microcontroller

Board Name	Operating Volt	Clock Speed	Digital i/o	Analog Inputs	PWM	UART	Programming Interface
Arduino Uno R3	5V	16MHz	14	6	6	1	USB via ATMega16U2
Arduino Uno R SMD	5V	16MHz	14	6	6	1	USB via ATMega16U2
Red Board	5V	16MHz	14	6	6	1	USB via FTDI

Arduino Pro 3.3v/8 MHz	3.3V	8MHz	14	6	6	1	FTDICompatible Header
Arduino Pr 5V/16MHz	5V	16MHz	14	6	6	1	FTDICompatible Header
Arduino mini 05	5V	16MHz	14	8	6	1	FTDICompatible Header
Arduino Pro mini 3.3v/8mhz	3.3V	8MHz	14	8	6	1	FTDICompatible Header
Arduino Pro mini 5v/16mhz	5V	16MHz	14	8	6	1	FTDI-Compatible Header
Arduino Ethernet	5V	16MHz	14	6	6	1	FTDICompatible Header
Arduino Fio	3.3V	8MHz	14	8	6	1	FTDICompatible Header
LilyPad Arduino 328 main board	3.3V	8MHz	14	6	6	1	FTDICompatible Header

LilyPad Arduino simple board	3.3V	8MHz	9	4	5	0	FTDICompatible Header

Arduino boards based on ATMEGA32u4 microcontroller

Board Name	Operating Volt	Clock Speed	Digital i/o	Analog Inputs	PWM	UART	Programming Interface
Arduino Leonardo	5V	16MHz	20	12	7	1	Native USB
Pro micro 5V/16MHz	5V	16MHz	14	6	6	1	Native USB
Pro micro 3.3V/8MHz	5V	16MHz	14	6	6	1	Native USB
LilyPad Arduino USB	3.3V	8MHz	14	6	6	1	Native USB

Arduino boards based on ATMEGA2560 microcontroller

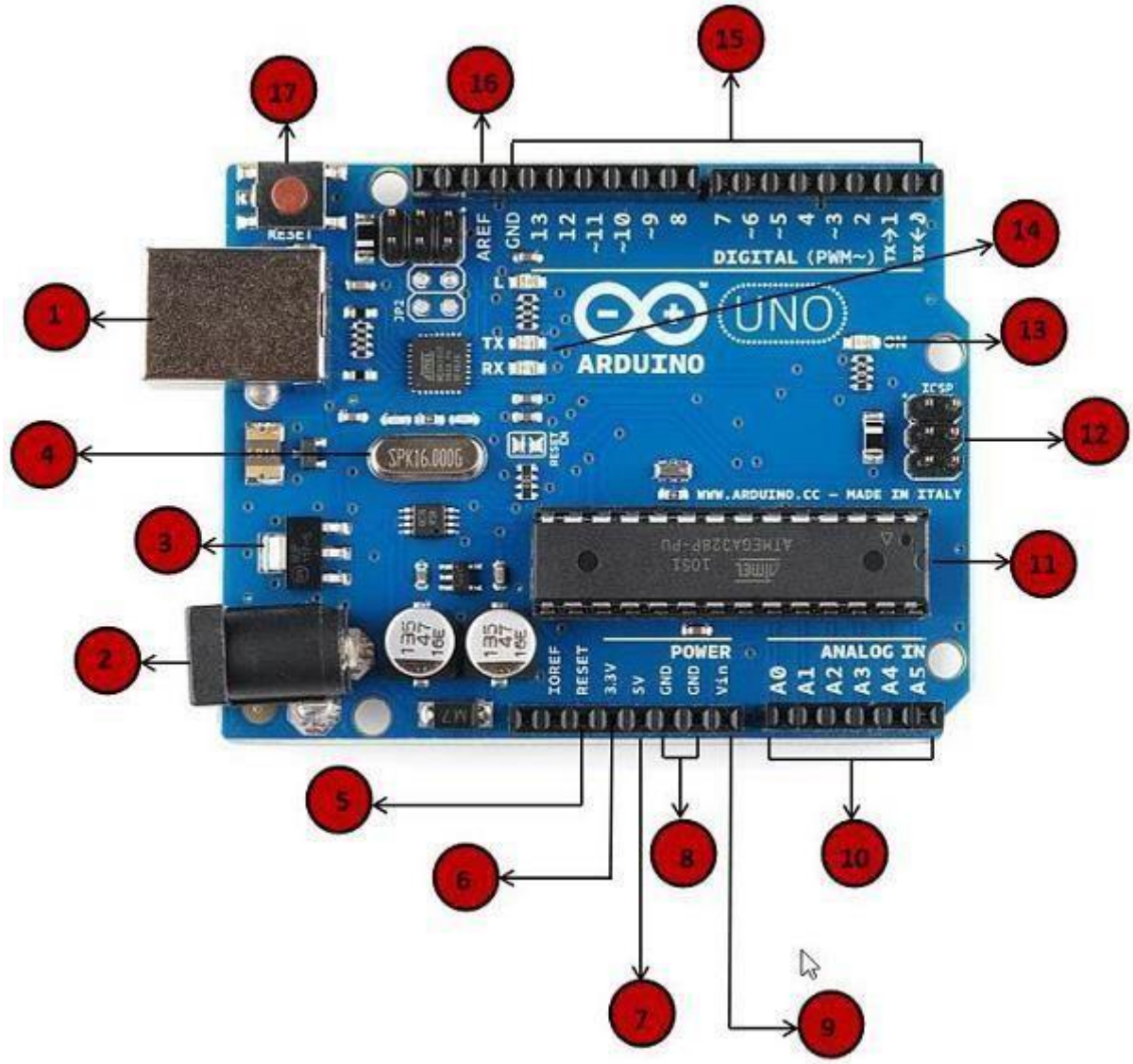
Board Name	Operating Volt	Clock Speed	Digital i/o	Analog Inputs	PWM	UART	Programming Interface
Arduino Meg 2560 R3	5V	16MHz	54	16	14	4	USB via ATMega16U2B
Mega Pro 3.3V	3.3V	8MHz	54	16	14	4	FTDICompatible Header

Mega Pro 5V	5V	16MHz	54	16	14	4	FTDICompatible Header
Mega Pro Mini 3.3V	3.3V	8MHz	54	16	14	4	FTDICompatible Header

Arduino boards based on AT91SAM3X8E microcontroller

Board Name	Operating Volt	Clock Speed	Digital i/o	Analog Inputs	PWM	UART	Programming Interface
Arduino Meg 2560 R3	3.3V	84MHz	54	12	12	4	USB native

In this chapter, we will learn about the different components on the Arduino board. We will study the Arduino UNO board because it is the most popular board in the Arduino board family. In addition, it is the best board to get started with electronics and coding. Some boards look a bit different from the one given below, but most Arduinos have majority of these components in common.



1	**Power USB** Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection (1).
2	**Power (Barrel Jack)** Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack (2).
3	**Voltage Regulator** The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.

Crystal Oscillator

The crystal oscillator helps Arduino in dealing with time issues. How does Arduino calculate time? The answer is, by using the crystal oscillator. The number printed on top of the Arduino crystal is 16.000H9H. It tells us that the frequency is 16,000,000 Hertz or 16 MHz.

Arduino Reset

You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET (5).

Pins (3.3, 5, GND, Vin)

- 3.3V (6) − Supply 3.3 output volt
- 5V (7) − Supply 5 output volt
- Most of the components used with Arduino board works fine with 3.3 volt and 5 volt.
- GND (8)(Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- Vin (9) − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

Analog pins

The Arduino UNO board has six analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor.

11

Main microcontroller

Each Arduino board has its own microcontroller (11). You can assume it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different from board to board. The microcontrollers are usually of the ATMEL Company. You must know what IC your board has before loading up a new program from the Arduino IDE. This information is available on the top of the IC. For more details about the IC construction and functions, you can refer to the data sheet.

12

ICSP pin

Mostly, ICSP (12) is an AVR, a tiny programming header for the Arduino consisting of MOSI, MISO, SCK, RESET, VCC, and GND. It is often referred to as an SPI (Serial Peripheral Interface), which could be considered as an "expansion" of the output. Actually, you are slaving the output device to the master of the SPI bus.

13

Power LED indicator

This LED should light up when you plug your Arduino into a power source to indicate that your board is powered up correctly. If this light does not turn on, then there is something wrong with the connection.

14

TX and RX LEDs

On your board, you will find two labels: TX (transmit) and RX (receive). They appear in two places on the Arduino UNO board. First, at the digital pins 0 and 1, to indicate the pins responsible for serial communication. Second, the TX and RX led (13). The TX led flashes with different speed while sending the serial data. The speed of flashing depends on the baud rate used by the board.
RX flashes during the receiving process.

15

Digital I/O

The Arduino UNO board has 14 digital I/O pins (15) (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled “~” can be used to generate PWM.

16	**AREF** AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

After learning about the main parts of the Arduino UNO board, we are ready to learn how to set up the Arduino IDE. Once we learn this, we will be ready to upload our program on the Arduino board.

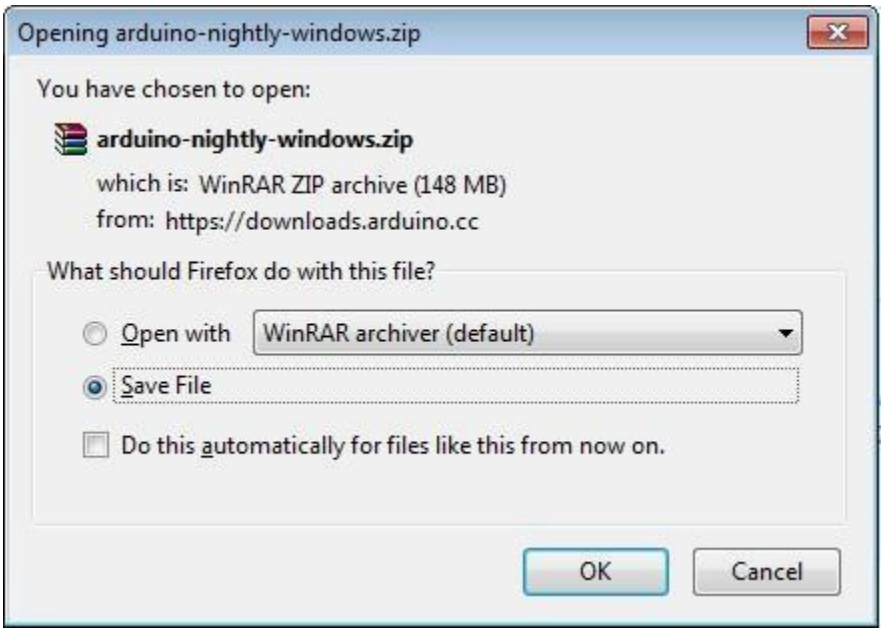
In this section, we will learn in easy steps, how to set up the Arduino IDE on our computer and prepare the board to receive the program via USB cable.

Step 1 − First you must have your Arduino board (you can choose your favorite board) and a USB cable. In case you use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable (A plug to B plug), the kind you would connect to a USB printer as shown in the following image.

In case you use Arduino Nano, you will need an A to Mini-B cable instead as shown in the following image.

Step 2 − Download Arduino IDE Software.

You can get different versions of Arduino IDE from theDownload pageon the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.



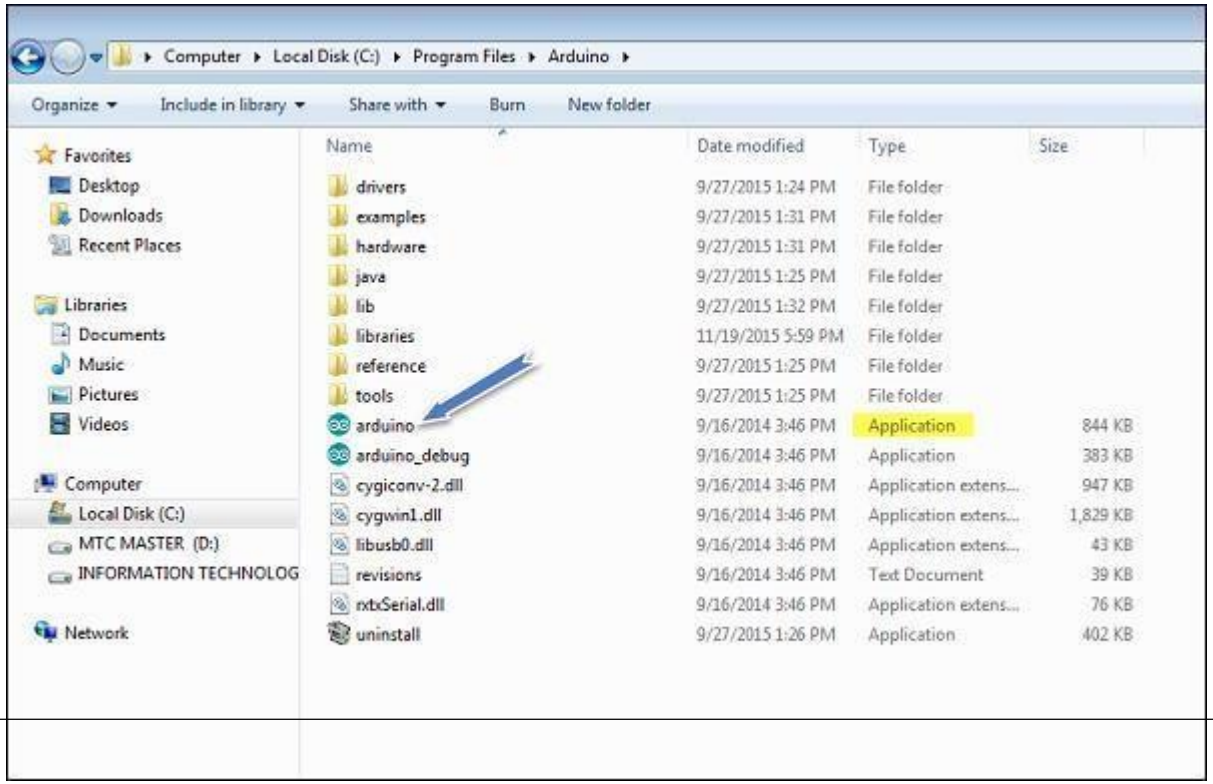
Step 3 − Power up your board.

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port.

Connect the Arduino board to your computer using the USB cable. The green power LED (labeled PWR) should glow.

Step 4 − Launch Arduino IDE.

After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Double-click the icon to



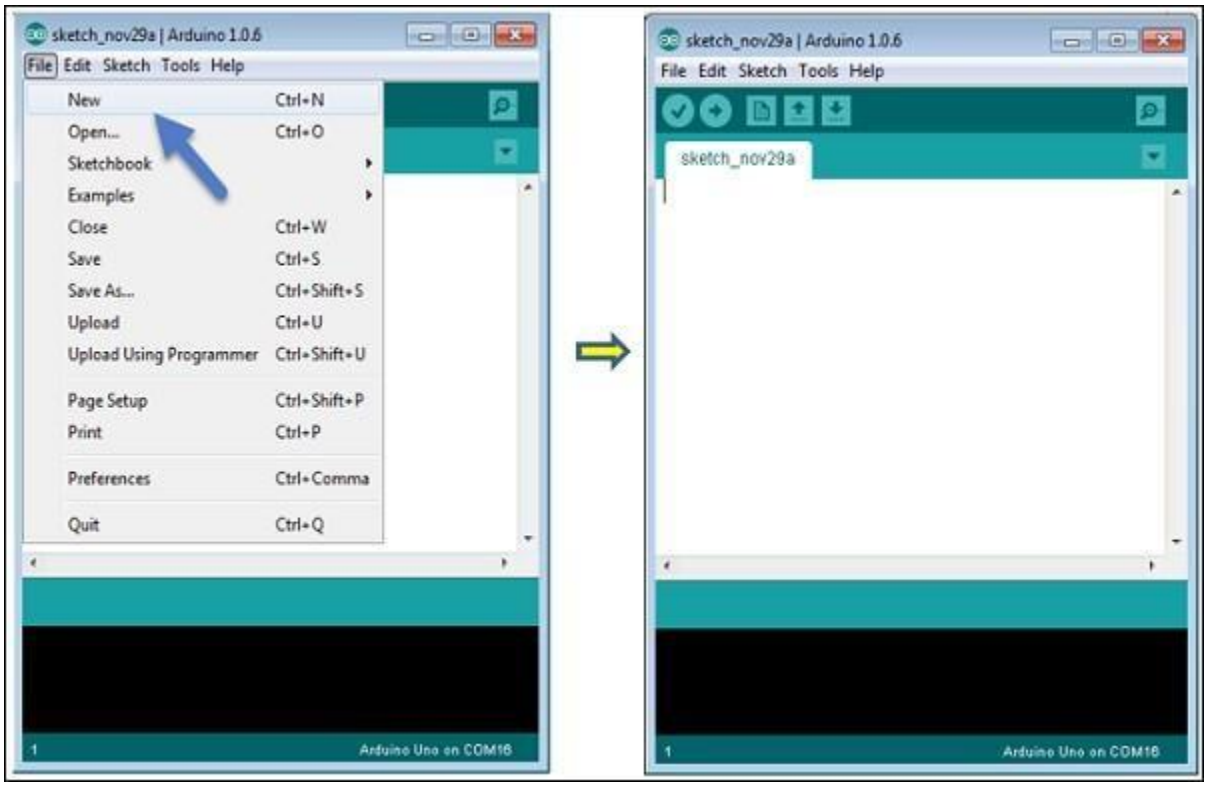
start the IDE.

Step 5 − Open your first project.

Once the software starts, you have two options −

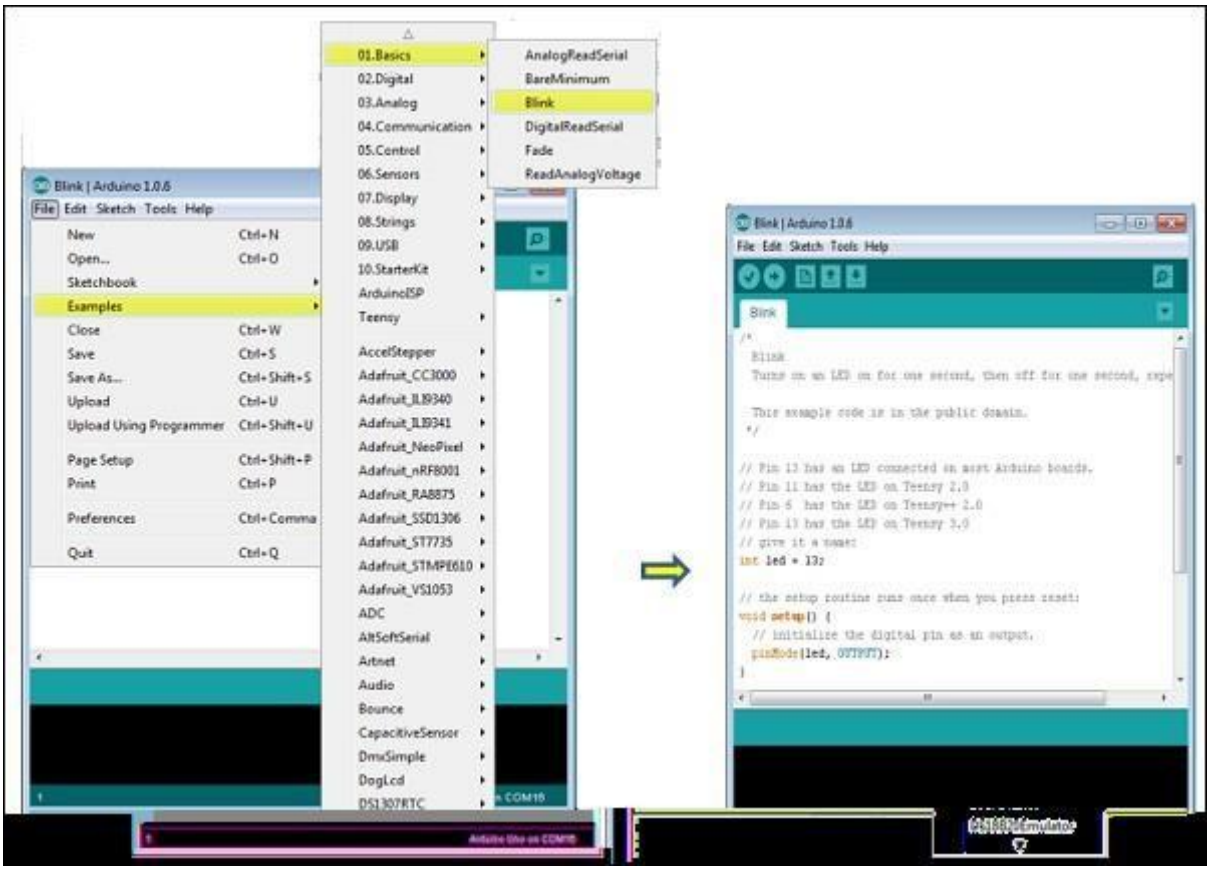
- Create a new project.
- Open an existing project example.

To create a new project, select File → **New**.



To open an existing project example, select File → Example → Basics → Blink.

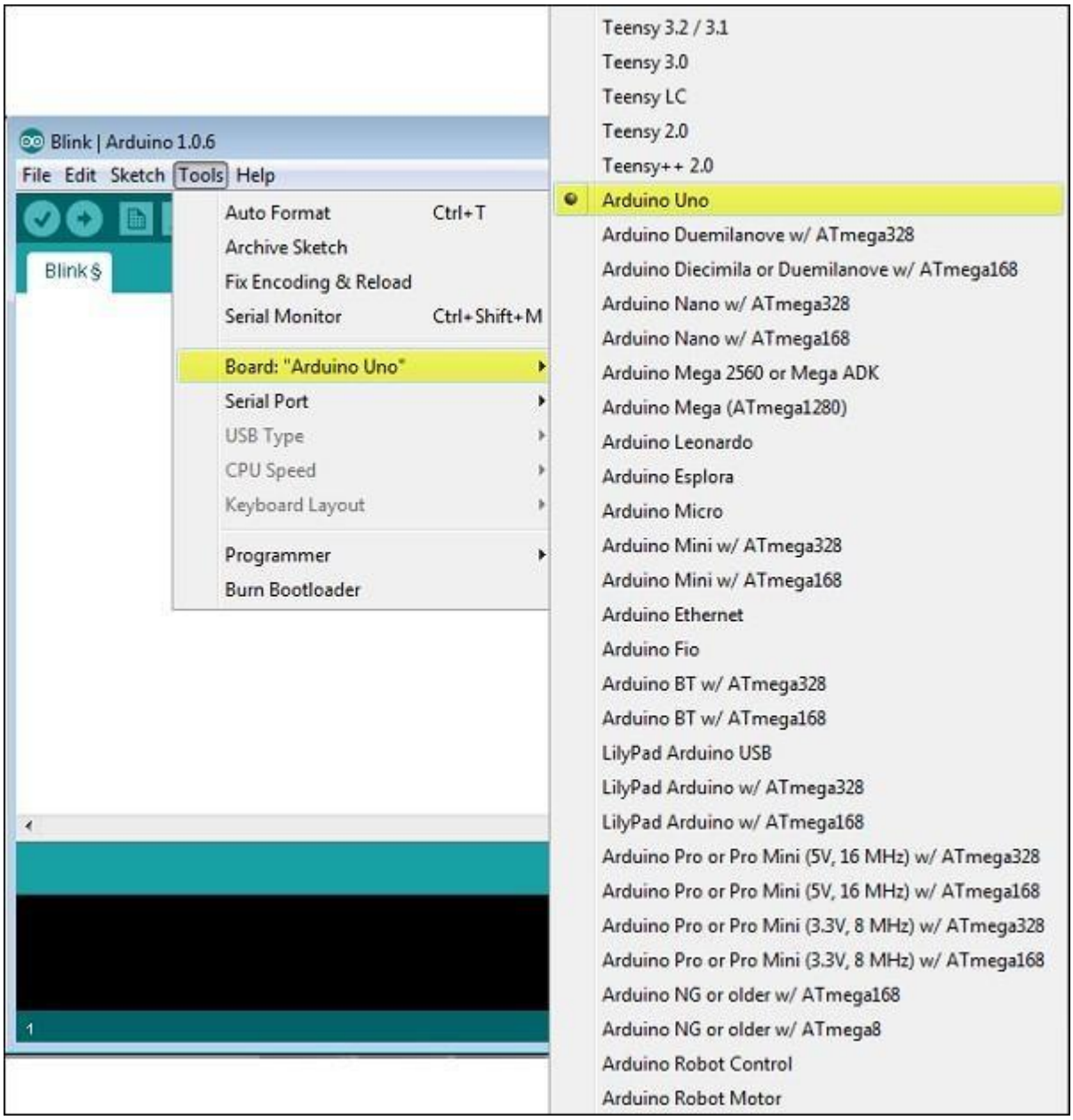
Here, we are selecting just one of the examples with the name **Blink**. It turns the LED on and off with some time delay. You can select any other example from the list.



Step 6 − Select your Arduino board.

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer.

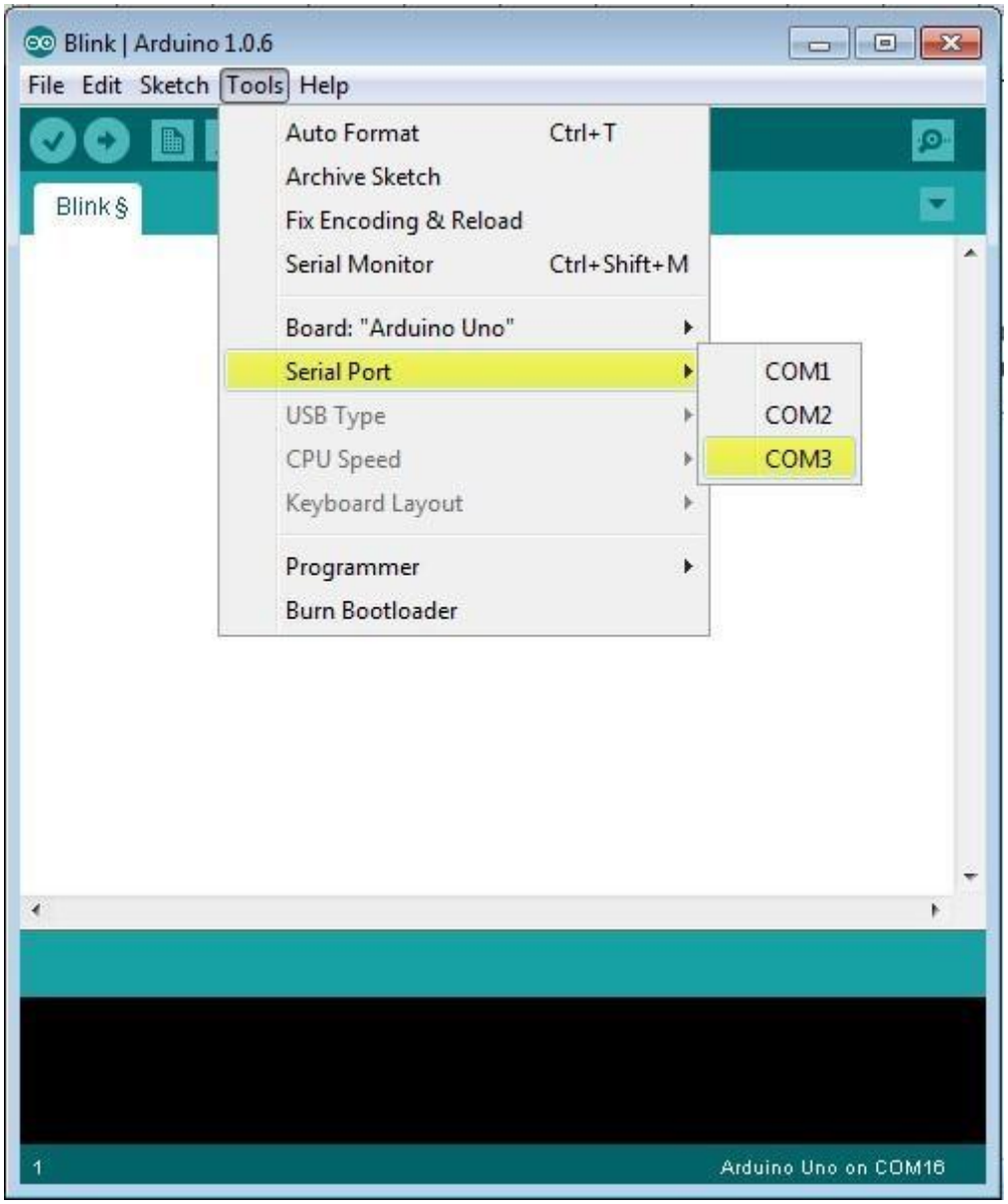
Go to Tools → Board and select your board.



Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

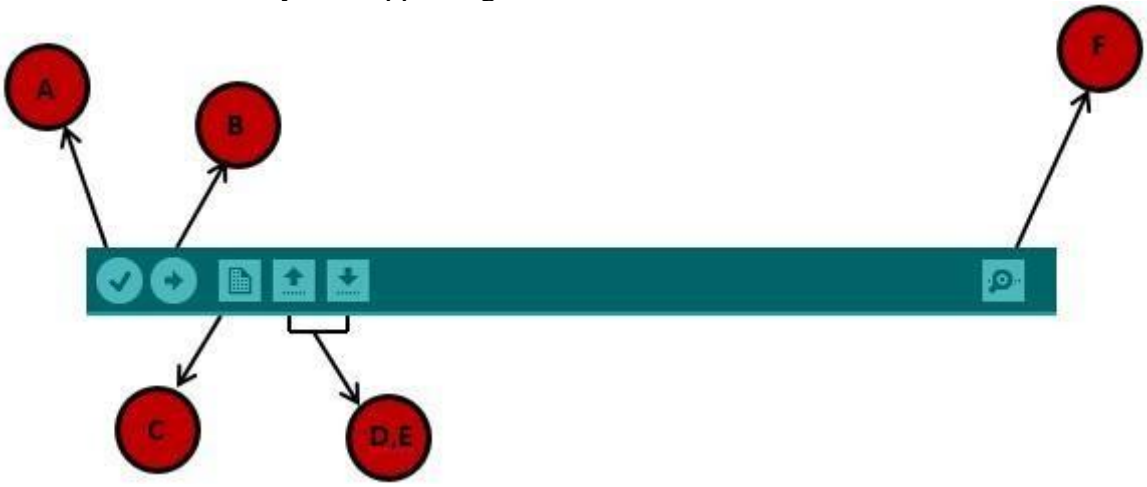
Step 7 − Select your serial port.

Select the serial device of the Arduino board. Go to **Tools → Serial Port** menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.



Step 8 − Upload the program to your board.

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.



A − Used to check if there is any compilation error.

B − Used to upload a program to the Arduino board.

C − Shortcut used to create a new sketch.

D − Used to directly open one of the example sketch.

E − Used to save your sketch.

F − Serial monitor used to receive serial data from the board and send the serial data to the board.

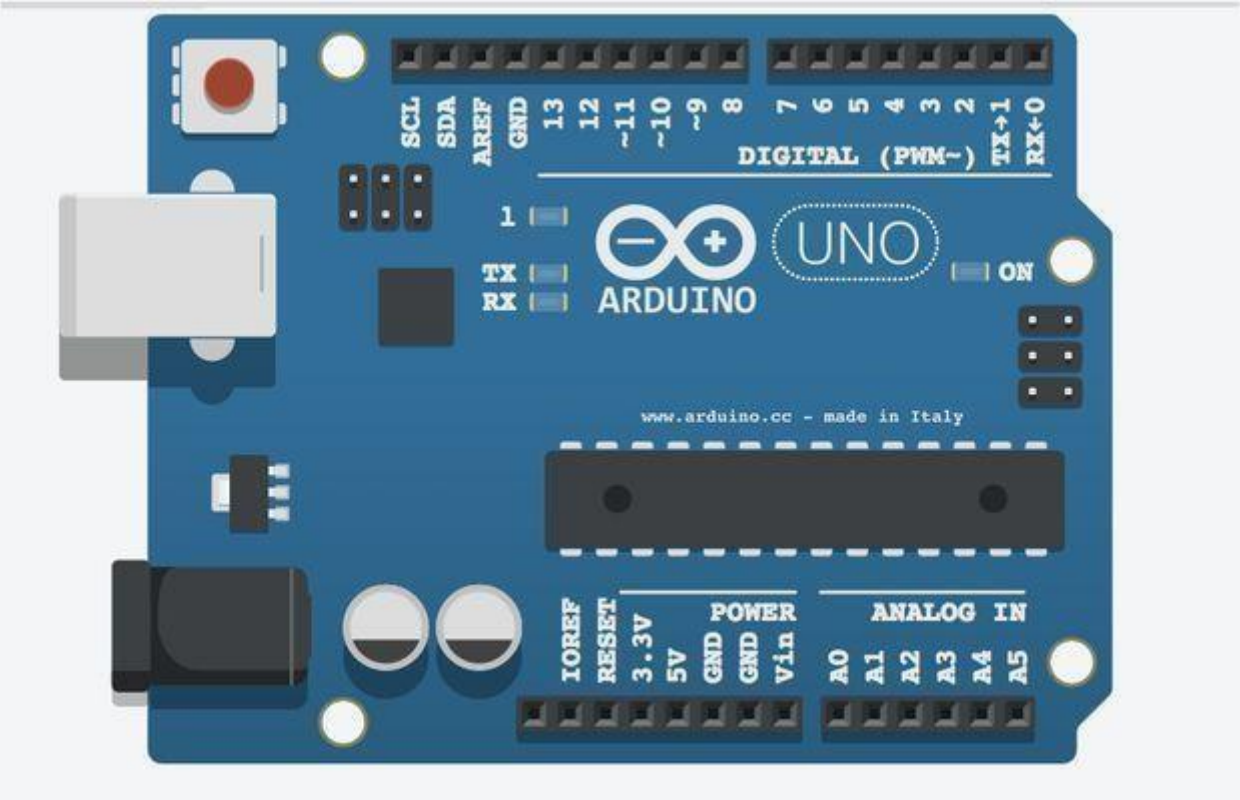
Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

IoT Platforms Overview: Arduino, Raspberry Pi

The IoT concepts imply a creation of network of various devices interacting with each other and with their environment. Interoperability and connectivity wouldn't be possible without hardware platforms that help developers solve issues such as building autonomous interactive objects or completing common infrastructure related tasks.

Let's go through the most popular IoT platforms and see how they work and benefit IoT software developers.

Arduino



The Arduino platform was created back in 2005 by the Arduino company and allows for open source prototyping and flexible software development and back-end deployment while providing significant ease of use to developers, even those with very little experience building IoT solutions.

Arduino is sensible to literally every environment by receiving source data from different external sensors and is capable to interact with other control elements over various devices, engines and drives. Arduino has a built-in micro controller that operates on the Arduino software.

Projects based on this platform can be both standalone and collaborative, i.e. realized with use of external tools and plugins. The integrated development environment (IDE) is composed of the open source code and works equally good with Mac, Linux and Windows OS. Based on a *processing* programming language, the Arduino platform seems to be created for new users and for experiments. The processing language is dedicated to visualizing and building interactive apps using animation and Java Virtual Machine (JVM) platform.

Let's note that this programming language was developed for the purpose of learning basic computer programming in a visual context. It is an absolutely free project available to every interested person. Normally, all the apps are programmed in C/C++, and are wrapped with *avr-gcc* (WinAVR in OS Windows).
Arduino offers analogue-to-digital input with a possibility of connecting light, temperature or sound sensor modules. Such sensors as *SPI* or *I2C* may also be used to cover up to 99% of these apps' market.

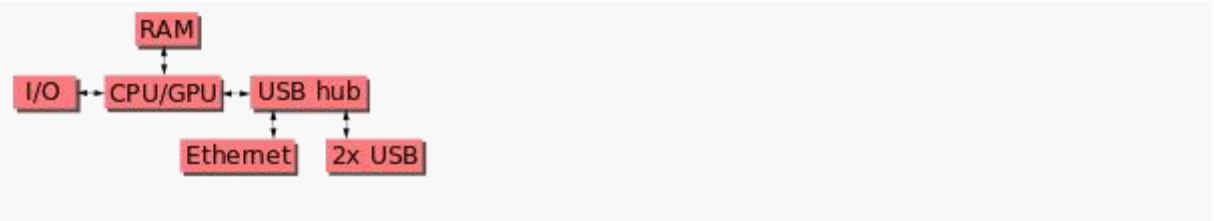
Arduino is a microcontroller (generally it is the 8-bit ATmega microcontroller), but not a mini-computer, which makes Arduino somehow limited in its features for advanced users. Arduino provides an excellent interactivity with external devices and offers a wide range of user manuals, project samples as well as a large community of users to learn from / share knowledge with.

Raspberry Pi

Raspberry Pi(/paɪ/) is a series of smallsingle-board computersdeveloped in theUnited Kingdomby theRaspberry Pi Foundationin association withBroadcom. Early on, the Raspberry Pi project leaned towards the promotion of teaching basiccomputer sciencein schools and indeveloping countries. Later, the original model became far more popular than anticipated,selling outside itstarget marketfor uses such asrobotics. It is now widely used in many areas, such as for weather monitoring,because of its low cost, modularity, and open design.

After the release of the second board type, the Raspberry Pi Foundation set up a new entity, named Raspberry Pi Trading, and installedEben UptonasCEO, with the responsibility of developing technology.The Foundation was rededicated as an educational charity for promoting the teaching of basic computer science in schools and developing countries.The Raspberry Pi is one of the best-sellingBritish computers.

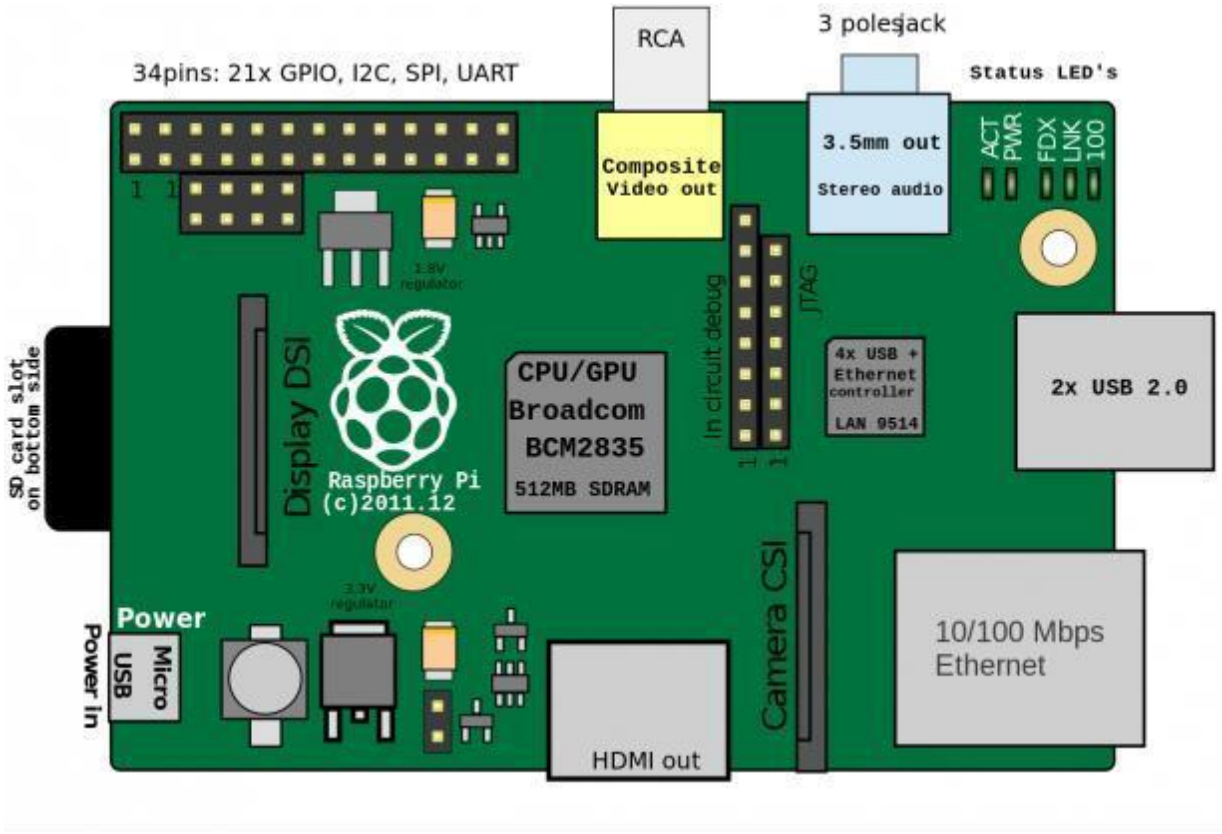
The Raspberry Pi hardware has evolved through several versions that feature variations in the type of the central processing unit, amount ofmemorycapacity, networking support, and peripheral-device support.



This block diagram describes Model B and B+; Model A, A+, and the Pi Zero are similar, but lack theEthernetandUSBhub components. The Ethernet adapter is internally connected to an additional USB port. In Model A, A+, and the Pi Zero, the USB port is connected directly to thesystem on a chip(SoC). On the Pi 1 Model B+ and later models the USB/Ethernet chip contains a five-port USB hub, of which four ports are available, while the Pi 1 Model B only provides two. On the Pi Zero, the USB port is also connected directly to the SoC, but it uses amicro USB(OTG) port. Unlike all other Pi models, the 40 pin GPIO connector is omitted on the Pi Zero, with solderable through-holes only in the pin locations. The Pi Zero WH remedies this.

Processor speed ranges from 700 MHz to 1.4 GHz for the Pi 3 Model B+ or 1.5 GHz for the Pi 4; on-board memory ranges from 256MiBto 1GiBrandom-access memory(RAM), with up to 8 GiB available on the Pi 4.SecureDigital(SD) cards in MicroSDHC form factor (SDHC on early

models) are used to store the operating system and program memory. The boards have one to fiveUSBports. For video output,HDMIandcomposite videoare supported, with a standard 3.5 mmtip-ring-sleevejack for audio output. Lower-level output is provided by a number ofGPIOpins, which support common protocols likeI²C. The B-models have an8P8CEthernetport and the Pi 3, Pi 4 and Pi Zero W have on-boardWi-Fi802.11nandBluetooth.



Raspberry Pi is a mono-board computing platform that's as tiny as a credit card. Initially it was developed for computer science education with later on progress to wider functions.

Since the inception of Raspberry, the company sold out more than 8 million items. Raspberry Pi 3 is the latest version and it is the first 64-bit computing board that also comes with built-in Wi-Fi and Bluetooth functions. According to Raspberry Pi Foundation CEO Eben Upton, "*it's been a year in the making*". The Pi3 version is replaced with a quad-core 64-bit
1.2 GHz ARM Cortex A53 chip, 1GB of RAM, VideoCore IV graphics, Bluetooth 4.1 and 802.11n Wi-Fi. The developers claim the new architecture delivers an average 50% performance improvement over the Pi 2.

Another peculiarity of Raspberry Pi is the *GPIO* (General Purpose Input-Output), which is a low-level interface of self-operated control by input-output ports. Raspberry has it as a 40pin connector.

Raspberry Pi uses Linux as its default operating system (OS). It's also fully Android compatible. Using the system on Windows OS is enabled through any virtualization system like *XenDesktop.* If you want to develop an application for Raspberry Pi on your computer, it is necessary to download a specific toolset comprised of ARM-compiler and some libraries complied down to ARM-target platform like *glibc.*

Cloud Computing in IOT

One component that improves the success of the Internet of Things is Cloud Computing. Cloud computing enables users to perform computing tasks using services provided over the Internet. The use of the Internet of Things in conjunction with cloud technologies has become a kind of catalyst: the Internet of Things and cloud computing are now related to each other. These are true technologies of the future that will bring many benefits.

Due to the rapid growth of technology, the problem of storing, processing, and accessing large amounts of data has arisen. Great innovation relates to the mutual use of the Internet of Things and cloud technologies. In combination, it will be possible to use powerful processing of sensory data streams and new monitoring services. As an example, sensor data can be uploaded and saved using cloud computing for later use as intelligent monitoring and activation using other devices. The goal is to transform data into insights and thus drive cost-effective and productive action.

Benefits And Functions of IoT Cloud:

There are many benefits of combining these services –

1. IoT Cloud Computing provides many connectivity options, implying large network access. People use a wide range of devices to gain access to cloud computing resources: mobile devices, tablets, laptops. This is convenient for users but creates the problem of the need for network access points.
2. Developers can use IoT cloud computing on-demand. In other words, it is a web service accessed without special permission or any help. The only requirement is Internet access.
3. Based on the request, users can scale the service according to their needs. Fast and flexible means you can expand storage space, edit software settings, and work with the number of users. Due to this characteristic, it is possible to provide deep computing power and storage.
4. Cloud Computing implies the pooling of resources. It influences increased collaboration and builds close connections between users.
5. As the number of IoT devices and automation in use grows, security concerns emerge. Cloud solutions provide companies with reliable authentication and encryption protocols.
6. Finally, IoT cloud computing is convenient because you get exactly as much from the service as you pay. This means that costs vary depending on use: the provider measures your usage statistics. A growing network of objects with IP addresses is needed to connect to the Internet and exchange data between the components of the network.

It is important to note that cloud architecture must be well-designed since reliability, security, economy, and performance optimization depends upon it. Using well-designed CI/CD pipelines, structured services, and sandboxed environments results in a secure environment and agile development.

Comparison of Internet of Things and Cloud Computing:

Cloud is a centralized system helping to transfer and deliver data and files to data centers over the Internet. A variety of data and programs are easy to access from a centralized cloud system.

The Internet of Things refers to devices connected to the Internet. In the IoT, data is stored in real-time, as well as historical data. The IoT can analyze and instruct devices to make effective decisions, as well as track how certain actions function.

Cloud computing encompasses the delivery of data to datacenters over the Internet. IBM divides cloud computing into six different categories:

1. **Platform as a Service (PaaS) –**

   The cloud contains everything you need to build and deliver cloud applications so there is no need to maintain and buy equipment, software, etc.

2. **Software as a Service (SaaS) –**

   In this case, applications run in the cloud and other companies operate devices that connect to users' computers through a web browser.

3. **Infrastructure as a Service (IaaS) –**

   IaaS is an option providing companies with storage, servers, networks and hubs processing data for each use.

4. **Public cloud –**

   Companies manage spaces and provide users with quick access through the public network.

5. **Private cloud –**

   The same as a public cloud, but only one person has access here, which can be an organization, an individual company, or a user.

6. **Hybrid cloud –**

   Based on a private cloud, but provides access to a public cloud.

Now, the Internet of Things refers to connecting devices to the Internet. Everyday devices such as cars and household appliances may have an Internet connection, and with the advancement of the Internet of Things, more and more devices will join this list.

Pairing with edge computing:

Data processing at the network edge or edge computing is used with IoT solutions and enables faster processing and response times. To get a better understanding of how this works, consider a large factory with many implemented IoT sensors. In this situation, it makes sense, before sending data to the cloud for processing, to aggregate it close to the border to prevent cloud overload by reducing direct connections. Data centers with this approach make data processing much faster. Yet, an approach that is only based on the edge will never provide a complete view of business operations. If there is no cloud solution, then the factory only controls each unit individually. Also, it has no way of imagining how these units work in relation to each other. This is why only the combination of the edge and the cloud will enable businesses to benefit from IoT developments.

The Role of Cloud Computing on the Internet of Things:

Cloud computing works to improve the efficiency of daily tasks in conjunction with the Internet of Things. Cloud computing is about providing a path for data to reach its destination while the Internet of Things generates a huge amount of data.
According to Amazon Web Services, there are four benefits of cloud computing:

1. No need to pre-guess infrastructure capacity needs
2. Saves money, because you only need to pay for those resources that you use, the larger the scale, the more savings
3. In a few minutes, platforms can be deployed around the world
4. Flexibility and speed in providing resources to developers

Cloud connectivity in IoT

How communication works

Cloud connectivity, a fundamental part of the majority of the Internet of Things (IoT) projects, is intertwined with expertise in embedded systems and software integration. That's why MCU suppliers are working closely with cloud service providers to develop integrated hardware and software solutions that enable IoT developers to establish an edge-to-cloud connection using out-of-the-box solutions quickly and efficiently.

These collaborations take out most of the complexities in cloud-connected IoT deployments and thus significantly lower the barrier to entry for IoT solutions deployment. IoT developers, for instance, can connect development boards supplied by MCU vendors to cloud services like Google's Cloud IoT Core and Amazon's AWS IoT Core with a single click.

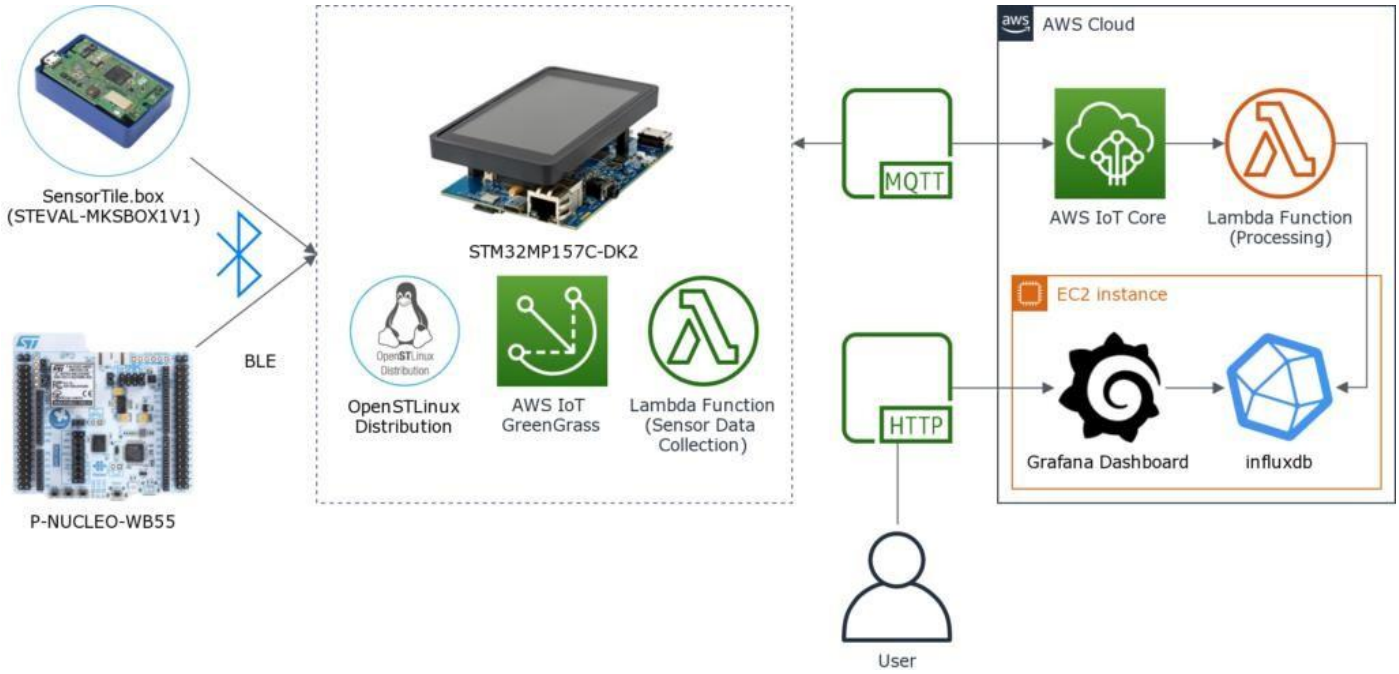


Figure 1: A view of an IoT node-to-cloud connection showing the hardware and software building blocks employed for an end-to-end IoT connection. (Image: STMicroelectronics)

Anatomy of cloud connection

But how does an IoT-to-cloud connection actually work? For a start, like every communication cannel, the link between an IoT device and the cloud service is established via wired or wireless communication networks such as Ethernet and Wi-Fi. Next, there are two common transport- and application-layer protocols that help facilitate communication between an IoT device and a cloud service.

At the transport-layer level, the device-to-cloud communication usually takes place either via Transmission Control Protocol (TCP) or User Datagram Protocol (UDP). Here, it's important to note that though TCP takes more network overhead, it's favored in IoT applications for its reliability. UDP, on the other hand, is more suitable for applications like video streaming that can afford some data packet loss.

Next, at the application-layer level, Hyper Text Transfer Protocol (HTTP) is the common standard to send connection requests and return responses for TCP-based communications. Message Queuing Telemetry Transport (MQTT) is another application-layer protocol; it's lightweight with a small code footprint and is becoming popular in resource-constrained IoT devices.

The role of APIs

It's a two-way data communication over the Internet between a device and a remote service. An IoT device establishes a connection with the Internet Protocol (IP) network and is subsequently hooked to the cloud. Here, support from MCU vendors ranges from the hardware level to the API stacks to facilitate IoT-to-cloud development.

The APIs are well defined, and they include open-source client libraries, example codes, and protocol stacks. Then there are third-party tools that make these APIs easier to use and deploy. So, all developers

need to learn is how to use IoT device interfaces effectively; embedded designers who know how to use or leverage IoT device APIs are now in high demand.

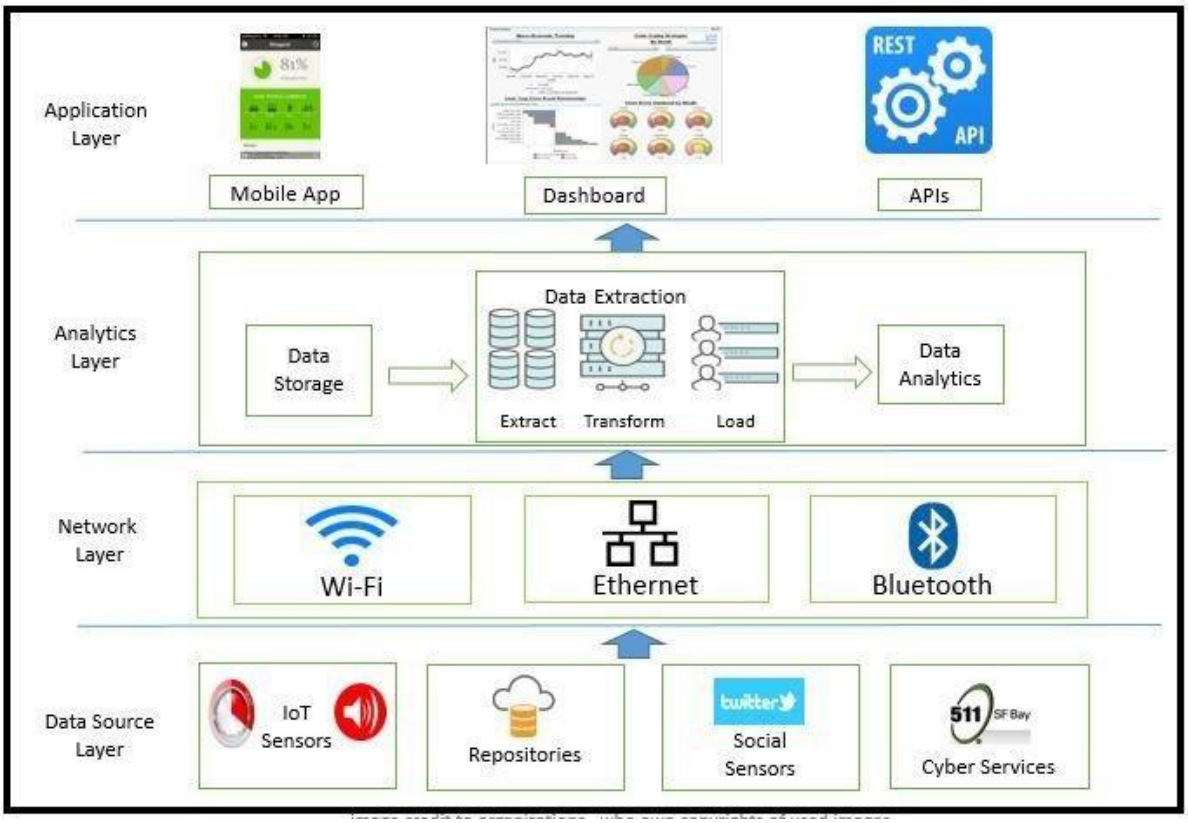
Figure 2: A step-by-step display of how an MCU-based IoT device can be connected to a cloud service like AWS IoT Core. (Image: CNX Software)

Take the example of how MCU suppliers are integrating software platforms like Amazon FreeRTOS into their microcontroller offerings **(Figure 2)**. Amazon FreeRTOS is an open-source real-time operating system (RTOS) for microcontrollers that includes kernel and software libraries to connect small, low-power MCUs to Amazon's cloud service AWS IoT Core.

The FreeRTOS-enabled microcontrollers can directly connect to a cloud service like AWS IoT Core, or they can connect to a local edge device such as AWS Greengrass and sustain the communications even if the IoT device loses connection to the cloud.

BIG DATA ANALYTICS

Data Analytics is the science (and art!) of applying statistical techniques to large data sets to obtain actionable insights for making smart decisions. It is the process to uncover hidden patterns, unknown correlations, trends and any other useful business information.



Analysing data

IoT data needs to be analysed in order to make it useful, but manually processing the flood of data produced by IoT devices is not practical. So, most IoT solutions rely on automated analytics. Analytics tools are applied to the telemetry data to generate descriptive reports, to present data through dashboards and data visualizations, and to trigger alerts and actions.

There are many different open-source analytics frameworks or IoT platforms that can be used to provide IoT data processing and analytics to your IoT solutions. Analytics can be performed in real-time as the data is received or through batch processing of historical data. Analytics approaches include distributed analytics, real-time analytics, edge analytics and machine learning.

Distributed analytics

Distributed analytics is necessary in IoT systems to analyse data at scale, particularly when dealing with historical data that is too vast to be stored or processed by a single node. Data can be spread across multiple databases; for example, device data might be bucketed into databases for each device per time period, such as hourly, daily, or monthly, like the IBM Watson IoT Historian Service that connects to Cloudant NoSQL database that stores the IoT data. Analytics may involve aggregating results which are distributed across multiple geographical locations. You'll want to adopt a storage driver or analytics framework that bridges distributed storage and compute infrastructure to allow seamless querying across distributed databases. Of particular note for processing of distributed data are the ecosystem of frameworks arising from the Hadoop community. Apache Hadoop is a batch processing framework that uses a MapReduce engine to process distributed data. Hadoop is very mature and was one of the first open-source frameworks to take off for big data analytics. There's also Apache Spark, which was started later with an intention to improve on some of the weak points of Hadoop. Hadoop and Spark are ideal for historical IoT data analytics for batch-processing where time sensitivity is not an issue, such as performing analysis over a complete set of data and producing a result at a later time.

Real-time analytics

Analytics for high-volume IoT data streams is often performed in real-time, particularly if the stream includes time-sensitive data, where batch processing of data would produce results too late to be useful or

any other application where latency is a concern.
Real-time analytics are also ideal for time series data, because unlike batch processing, real-time analytics tools usually support controlling the window of time analysis, and calculating rolling metrics, for example, to track hourly averages over time rather than calculating a single average across an entire dataset.
Frameworks that are designed for real-time stream analytics include Apache Storm and Apache Samza (usually used with Kafka and Hadoop YARN). Hybrid engines that can be used for either stream or batch analytics include Apache Apex, Apache Spark, and Apache Flink. Apache Kafka acts as an ingestion layer that can sit over the top of an engine like Spark, Storm or Hadoop. For a guide to selecting between these open source frameworks, read Choosing the right platform for high-performance, cost-effective stream processing applications.

Edge analytics

IoT analytics is not usually applied to raw device data. The data is pre-processed to filter out duplicates or to re-order, aggregate or normalize the data prior to analysis. This processing typically occurs at the point of acquisition, on the IoT devices themselves or on gateway devices that aggregate the data, to determine which data needs to be sent upstream.

Analytics applied at the edges of the network, as close as possible to the devices generating the data is known as edge analytics. Linux Foundation's Edge X Foundry, an open source IoT edge computing framework, also supports edge analytics.

Edge analytics is low-latency and reduces bandwidth requirements because not as much data needs to be transmitted from the device. However, constrained devices have limited processing capacity, so most IoT solutions use a hybrid approach involving edge analytics and upstream analytics.

Machine learning

Using traditional mathematical statistical models for analytics provides value as they can be used to track goals, create reports and insights, predict trends, and create simulations that are used to predict and optimize for specific outcomes. For example, you can predict the outcome of applying a specific action, predict the time to failure for a given piece of equipment, or optimize the configuration of an IoT system in terms of cost or performance.

However, the value of statistical analytics models diminishes when applied to dynamic data that contains many variables that change over time, when you don't know what factors to look for, or what variables to change to achieve a desired outcome like reducing cost or improving efficiency. In these cases, instead of using a statistical model, machine learning algorithms that learn from the data can be applied.

Machine learning can be applied to historic or real-time data. Machine learning techniques can be used to identify patterns, identify key variables and relationships between them to automatically create and refine analytics models, and then use those model for simulations or to produce decisions. Machine learning approaches have the advantage over static statistical analytics models that as new data comes in, the models can be improved over time, which leads to improved results. The state of the art Machine learning techniques mostly come in the domain of Deep learning using neural networks (Convolutional Neural Networks, Long Short Term Memory networks). Emerging and promising areas of research include Active learning, Multi modal and Multi-Task learning, and Transformer-based language models. That being said, the traditional machine learning methods like Regression, Support Vector Machines, and Decision trees can still prove to be effective in a lot of applications.

Data Visualization

Data visualization is the graphical representation of data to uncover patterns, relationships, and insights that might not be easily discernible in raw data. It involves transforming complex datasets into visual visuals, such as charts, graphs, maps, and infographics, to make data more understandable and accessible to a wide audience.

In the context of IoT, data visualization plays a crucial role in translating the massive amounts of data generated by IoT devices into meaningful and actionable insights. It helps in uncovering hidden patterns,

trends, and anomalies within the data, leading to informed decision-making and improved business outcomes.

Data visualization allows users to quickly grasp complex concepts and understand the significance of the data. Instead of poring over spreadsheets or rows of numbers, data visualizations present information in a visually appealing and intuitive manner, enabling easier interpretation and analysis.

By representing data visually, various relationships and patterns can be identified. For example, line charts can show trends over time, scatter plots can reveal correlations between variables, and maps can display geographic distribution.

Data visualization also offers the opportunity to explore data interactively. Users can utilize interactive elements, such as filters, drill-downs, and zoom features, to delve deeper into the data and uncover more detailed insights. This interactivity enhances the user experience and allows for dynamic exploration of the data.

Furthermore, data visualization promotes data storytelling by presenting information in a compelling narrative format. It helps to communicate complex ideas and data-driven insights effectively to a broad audience, including non-technical stakeholders. By presenting data in a visually engaging way, data visualizations facilitate better understanding, engagement, and decision-making.

Importance of Data Visualization in IoT

Data visualization plays a critical role in the realm of IoT, where vast amounts of data are being generated from interconnected devices. Let's explore the key reasons why data visualization is essential in the IoT landscape:1. **Understanding complex data:** IoT devices generate massive volumes of data that can be challenging to comprehend in its raw form. Through data visualization, complex data sets can be simplified and presented visually, allowing users to quickly grasp patterns, trends, and outliers. This understanding is vital for making data-driven decisions and identifying opportunities for improvement.

2. **Identifying actionable insights:** Data visualizations enable the identification of actionable insights from IoT data. By representing data visually, patterns and relationships become more apparent, enabling organizations to extract valuable insights. These insights can drive operational efficiency, optimize processes, and uncover potential areas for innovation.

3. **Real-time monitoring:** IoT devices generate data in real-time, which requires near-instantaneous analysis. Data visualizations provide real-time monitoring dashboards, enabling organizations to track key performance indicators (KPIs), identify anomalies, and respond promptly to critical events. Real-time data visualization empowers organizations to make informed decisions and take timely actions.

4. **Improved collaboration:** Data visualizations are easily understood by diverse audiences, including non-technical stakeholders. By presenting data visually, organizations can foster collaboration between technical and non-technical teams. This interdisciplinary collaboration facilitates a shared understanding of IoT data, leading to more effective communication, problem-solving, and decision-making.

5. **Enhanced data exploration:** Data visualization tools offer interactive features that empower users to explore and analyze IoT data in depth. These features include filters, drill-down capabilities, and dynamic visual representations. With interactive data visualizations, users can uncover hidden insights, perform root cause analysis, and gain a deeper understanding of IoT data.

6. **Improved decision-making:** Data visualization simplifies complex information, empowering decision-makers to quickly comprehend the implications of IoT data. Visual representations enable stakeholders to grasp the big picture, evaluate multiple data points simultaneously, and make informed decisions. This leads to more effective problem-solving, reduced response time, and better outcomes.

7. **Data-driven storytelling:** Data visualization enables organizations to tell compelling stories with their IoT data. By presenting data in a visually engaging manner, organizations can captivate audiences and communicate data-driven insights effectively. With data-driven storytelling, organizations can gain buy-in from stakeholders, inspire action, and drive positive business outcomes.

Benefits of Data Visualization in IoT

Data visualization in the context of IoT offers numerous benefits that help organizations harness the power of their data. Let's explore some of the key advantages of using data visualization in the IoT landscape:

1. **Improved data comprehension:** Data visualization simplifies complex data by presenting it in a visual format that is easy to understand. By representing data using charts, graphs, and other visual elements, users can quickly grasp patterns, trends, and relationships within IoT data. This improved comprehension enhances decision-making and drives actionable insights.

2. **Enhanced decision-making:** Data visualization empowers decision-makers to make informed and data-driven decisions. By visualizing IoT data, decision-makers can quickly identify trends, outliers, and insights that are difficult to discern in tabular or textual formats. Visual representations facilitate quicker decision-making, leading to better business outcomes.

3. **Identification of anomalies and issues:** Data visualization enables the identification of anomalies and issues within IoT data. By spotting unusual patterns or outliers, organizations can proactively address problems or take necessary actions. Detecting anomalies in real-time helps prevent system failures, optimize processes, and improve overall operational efficiency.

4. **Optimization of IoT systems:** Data visualization provides valuable insights into the performance and efficiency of IoT systems. By visualizing key metrics and performance indicators, organizations can identify bottlenecks, inefficiencies, and areas for improvement. This helps optimize IoT systems, enhance device performance, and ensure reliable and efficient operations.

5. **Enhanced data exploration:** Data visualization tools offer interactive features that facilitate in-depth exploration of IoT data. Users can easily filter, drill down, and manipulate visualizations to gain deeper insights. This interactive data exploration helps uncover hidden patterns, correlations, and dependencies within IoT data, enabling predictive analysis and data-driven decision-making.

6. **Improved communication and collaboration:** Data visualizations simplify the communication of complex IoT data to diverse audiences, including non-technical stakeholders. Visual representations make it easier to convey insights, trends, and key findings, fostering collaboration and shared understanding. By communicating data visually, organizations can align teams, secure buy-in from stakeholders, and drive effective decision-making.

7. **Real-time monitoring and alerts:** Data visualization allows for real-time monitoring of IoT data. Interactive dashboards provide up-to-date insights and alerts on key performance indicators and metrics. Real-time monitoring enables organizations to respond promptly to critical events, mitigate risks, and ensure optimal operational performance.

8. **Storytelling with data:** Data visualization enables impactful storytelling with IoT data. By presenting data visually, organizations can create compelling narratives that engage and inspire audiences. Visual storytelling helps convey the significance and implications of IoT data, driving action and enabling stakeholders to understand the value of data-driven insights.

Common Tools and Techniques for Data Visualization in IoT

Data visualization in the IoT landscape is facilitated by a variety of tools and techniques specifically designed to handle the complexities of IoT data. Let's explore some of the commonly used tools and techniques for data visualization in IoT:

1. **Business Intelligence (BI) Platforms:** BI platforms such as Tableau, Power BI, and QlikView provide powerful data visualization capabilities. These platforms enable users to connect and visualize IoT data from various sources, create interactive dashboards, and generate insightful reports. They offer a range of chart types, advanced data manipulation options, and real-time monitoring features.

2. **Custom-built Visualizations:** Organizations often develop custom-built visualizations tailored to their specific IoT data and use cases. These visualizations can be created using programming languages such as D3.js, Python's Matplotlib, or JavaScript frameworks like Highcharts. Custom visualizations provide flexibility and allow for unique representations of IoT data.

3. **Real-time Monitoring and Dashboarding:** Real-time monitoring is crucial in IoT data visualization. Tools such as Grafana, Kibana, and Splunk enable users to create real-time dashboards that display key metrics, alerts, and IoT data streams. These tools offer interactive visualizations, anomaly detection, and the ability to drill down into real-time data for in-depth analysis.

4. **Geospatial Visualization:** Geospatial visualization is often used in IoT applications that involve geographic data. Platforms like ArcGIS, Google Maps API, and Mapbox provide mapping capabilities to visualize IoT data in a spatial context. Geospatial visualizations help analyze location-based data, identify spatial patterns, and make location-specific decisions.

5. **Time-Series Analysis and Visualization:** Time-series analysis is crucial in IoT, where data is collected over time. Tools like Grafana, Python's Matplotlib, and Excel offer specialized features for time-series analysis and visualizations. These tools enable users to uncover trends, seasonality, and anomalies in time-series IoT data and present them in intuitive visual formats.

6. **Interactive Data Exploration:** Tools like Plotly, d3.js, and ggplot2 allow for interactive data exploration and manipulation. These tools provide users with the ability to filter, sort, and drill down into IoT data to gain deeper insights. Interactive visualizations enhance the exploration process and support data-driven decision-making.

7. **Data Streaming Visualization:** For real-time data streaming from IoT devices, tools like Apache Kafka and Apache Flink are often used. These tools allow for the data ingestion, processing, and visualization of high-velocity IoT data streams. They provide real-time visualizations and analytics for streaming data, enabling organizations to monitor and analyze IoT data as it flows.

8. **Machine Learning and AI Visualization:** Machine learning and AI techniques are employed in IoT data analysis. Tools like TensorFlow, RapidMiner, and KNIME provide visualization capabilities specifically designed for machine learning models and algorithms. These tools enable users to visualize model outputs, analyze predictions, and evaluate the performance of AI-based IoT applications.

Best Practices for Data Visualization in IoT

Effective data visualization in the IoT landscape requires adherence to certain best practices to ensure clarity, accuracy, and actionable insights. Let's explore some of the key best practices for data visualization in IoT:

1. **Understand the Audience:** Start by understanding the target audience and their specific needs and goals. Consider the level of technical expertise, domain knowledge, and the key insights they are seeking from the IoT data. Customize the visualizations accordingly to ensure they resonate with the audience and effectively convey the intended message.

2. **Keep it Simple and Clear:** Simplify complex data by using a minimalist approach. Avoid cluttered and overloaded visuals that can confuse or overwhelm the viewers. Choose clean and straightforward visual elements, removing any unnecessary embellishments. Focus on clarity and ensure that the key message is communicated clearly.

3. **Choose the Right Visualization Techniques:** Select appropriate visualization techniques that best represent the IoT data and insights you want to convey. Consider the data type, relationships, and objectives. Whether it's line charts, bar graphs, maps, or scatter plots, choose visuals that effectively communicate the message and facilitate easy interpretation.

4. **Provide Context:** Contextualize the IoT data by providing relevant background information, explanations, and labels. Include axes labels, titles, and legends to help viewers understand and interpret the visualizations accurately. Adding context ensures that the viewers can grasp the significance of the data and make informed decisions based on it.

5. **Use Color and Contrast Thoughtfully:** Color is a powerful tool in data visualization, but it should be used thoughtfully. Choose a color palette that enhances the visual appeal without compromising clarity. Use color to highlight key data points or patterns, and ensure sufficient contrast to distinguish between different elements in the visualization.

6. **Ensure Data Accuracy and Integrity:** Validate the accuracy and integrity of the IoT data before visualizing it. Check for outliers, missing data, and any data inconsistencies that could affect the reliability and validity of the visualizations. Incorrect or misleading visualizations can lead to poor decision-making, so ensure the data is trustworthy.

7. **Make it Interactive:** Incorporate interactive elements in the visualizations to enable viewers to explore the IoT data further. Interactive features like filters, brushable charts, and tooltips allow users to analyze specific areas of interest, drill down into details, and extract more insights. Interactivity enhances engagement and facilitates deeper exploration of IoT data.

8. **Consider Device Limitations:** Keep in mind the limitations of the devices through which the visualizations will be accessed. Ensure that the visualizations are responsive and optimized for different screen sizes and resolutions. Consider the bandwidth and processing power of the devices to ensure a smooth and seamless user experience.

9. **Iterative Design and Testing:** Take an iterative approach to design and testing of the visualizations. Seek feedback from end users and stakeholders to fine-tune the visualizations and ensure they effectively meet their needs. Test the visualizations on various devices and platforms to identify and resolve any usability or compatibility issues.

10. **Document and Maintain Consistency:** Document the design decisions, guidelines, and standards for data visualization in IoT. Maintain consistency in style, colors, fonts, and layouts across different visualizations to reinforce the brand identity and create a cohesive experience. Consistency improves user familiarity and allows for easier understanding and interpretation of the visualizations.

Real-world Examples of Data Visualization in IoT

Data visualization plays a crucial role in various real-world applications of IoT, enabling organizations to gain valuable insights from complex data and make data-driven decisions. Let’s explore some noteworthy examples of data visualization in IoT:

1. **Smart Cities:** In smart cities, IoT devices gather data on traffic patterns, air quality, energy consumption, and more. Data visualization tools help city officials analyze and visualize this data to make informed decisions. For example, interactive maps can display real-time traffic congestion, enabling officials to optimize traffic flow and reduce congestion.

2. **Manufacturing and Industrial IoT:** In manufacturing, IoT devices collect data on the production line, equipment performance, and product quality. Data visualizations provide real-time monitoring of manufacturing processes, highlighting areas for improvement and identifying bottlenecks. Interactive dashboards allow operators to adjust parameters to optimize production efficiency.

3. **Agriculture and Precision Farming:** In precision farming, IoT sensors collect data on soil moisture, temperature, and crop health. Data visualization tools help farmers understand this data visually, enabling them to make informed decisions on irrigation, fertilization, and disease control. Visualizations can display crop health maps, showing areas that require attention.

4. **Healthcare and Telemedicine:** In healthcare, IoT devices and wearables monitor patient vitals and collect health data. Data visualization tools enable healthcare professionals to monitor patients remotely, visualize trends in health parameters, and detect anomalies. This visual information assists in making timely decisions and providing personalized care.

5. **Energy Management:** IoT devices monitor energy usage in smart buildings, homes, and grids. Data visualizations help users understand energy consumption patterns, identify areas of high energy usage, and optimize energy efficiency. Interactive dashboards can display real-time energy consumption and provide recommendations for reducing energy waste.

6. **Retail Analytics:** In the retail industry, IoT devices track customer movement, footfall, and buying behavior. Data visualizations enable retailers to understand customer behavior, optimize store layouts, and personalize shopping experiences. Heat maps and flow diagrams visualize foot traffic patterns, informing strategic decisions on product placement and store design.

7. **Transportation and Logistics:** In the transportation sector, IoT sensors gather data on vehicle performance, fuel consumption, and route optimization. Data visualization tools help fleet managers monitor vehicle conditions, track routes, and identify areas for improvement. Real-time visualizations can display live tracking of vehicles, enabling efficient routing and minimizing delays.

8. **Environmental Monitoring:** IoT sensors collect data on air quality, water quality, and weather conditions. Data visualization tools transform this data into interactive maps, graphs, and heat maps, providing a comprehensive view of environmental factors. Visualizations enable researchers and policymakers to monitor and respond to environmental changes effectively.

UNIT V CHALLENGES IN IOT AND CASE STUDIES

Security Concerns and Challenges - Real time applications of IoT – Home automation – Automatic lighting – Home intrusion detection – Cities – Smart parking – Environment – Weather monitoring system – Agriculture – Smart irrigation.

Security Concerns and Challenges

IoT security is based on a cybersecurity strategy to protect IoT devices and the vulnerable networks they connect to from cyber attacks. IoT devices have no built-in security. IoT security is needed to help prevent data breaches because IoT devices transfer data over the internet unencrypted and operate undetected by standard cybersecurity systems.

Along with the meaning of IoT Security, it is important to understand the many challenges facing enterprises when dealing with IoT security issues. IoT devices were not built with security in mind. The ongoing proliferation and diversity of IoT devices and communications channels increases the potential for your organization to be exposed to cyber threats.

Unfortunately, there is no way to install security software on most IoT devices. IoT devices may even ship with malware on them that infects the network when they connect. This is why network security is a priority for IoT security.

Many network security solutions do not have the ability to detect connected IoT devices or show which devices are communicating on the network.

The following sections explore these and other big IoT security challenges including:

- Weak authentication and authorization
- Lack of encryption
- Vulnerabilities in firmware and software
- Insecure communications
- Difficulty in patching and updating devices

Weak Authentication and Authorization

IoT devices often rely on weak authentication and authorization practices, which makes them vulnerable to threats. For example, many devices use default passwords making it easier for hackers to gain access to IoT devices and the networks they use for communication. In addition, rogue IoT devices (i.e., undetected) that are connected to the network can be used to steal data or launch attacks.

Lack of Encryption

The overwhelming majority of IoT device network traffic is unencrypted making confidential and personal data vulnerable to a malware attack such as ransomware or other form of data breach or theft. This includes IoT devices used for medical imaging and patient monitoring, as well as security cameras and printers.

Vulnerabilities in Firmware and Software

The short development cycles and low price points of IoT devices limit the budget for developing and testing secure firmware. Without this built-in IoT security, IoT devices are vulnerable to the most rudimentary forms of attack. From firmware to software and third-party apps–millions of devices are affected by vulnerabilities in standard components.

Plus, network environments can be compromised by vulnerable web apps and software for IoT devices. Whether it is a new threat or old malware, without IoT security, all types of vulnerabilities make IoT devices good targets for savvy bad actors to stage cyberattacks.

Insecure Communications Protocols and Channels

IoT devices are often connected to the same network as other devices, which means that an attack on one device can spread to others. Lack of network segmentation and oversight of the ways IoT devices communicate makes them easier to intercept. For example, not long ago the automotive industry's adoption of Bluetooth technology in IoT devices resulted in a wave of data breaches that made the news. As well, protocols like HTTP (Hypertext Transfer Protocol) and API–are all channels that IoT devices rely on and cyber criminals exploit.

For example, in 2022, millions of Buetooth digital locks in smart cars could be remotely unlocked by hackers exploiting a vulnerability in Bluetooth technology. As well, protocols like HTTP (Hypertext Transfer Protocol) and API-are channels that IoT devices rely on and cyber criminals can exploit.

Difficulty in Patching and Updating Devices

IoT manufacturers don't focus on building IoT security into their devices to make hardware tamper proof and secure. Many IoT devices are not designed to receive regular IoT security updates, which makes them vulnerable to attacks. Without built-in IoT security it's difficult to ensure secure upgrades, provide firmware updates and patches, and perform dynamic testing. Therefore, the onus is on the organization to protect its IoT devices and network environment from cyber threats.

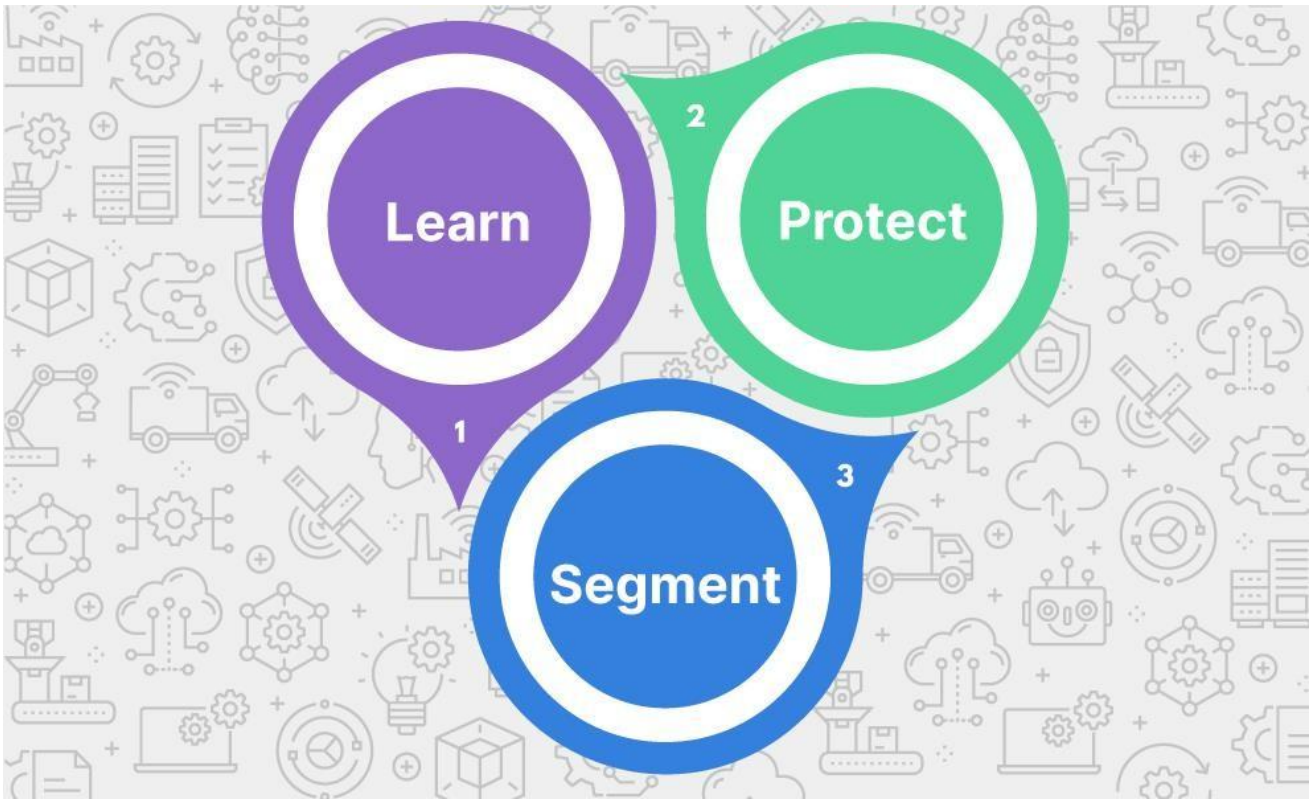
IoT Security Challenges

The IoT attack surface expands every day as more and more devices come online–from our smartwatches and smart TVs, to our smart homes and smart cars, to the ever-growing industry IoT. In addition to consumer goods, IoT sensors are widely used in healthcare, manufacturing, and supply chain operations, as well as for green agriculture, the economy, and national defense.

Burgeoning IoT spans virtually any device or sensor that connects to the internet-from a large container on an ocean barge to a small Tile Tracker for your phone. To underscore, the IEEE IoT technology forecast of connected devices is expected to increase by about 300% from 8.7 billion devices in 2020 to more than 25 billion IoT devices in 2030.

Given the expanded attack surface for security risks to availability, integrity and confidentiality, IoT security is critical for organizations to protect their network environments from IoT device-borne threats.

How to address IoT Security Requirements?



IoT and security requirements can only be accomplished with an integrated solution that delivers visibility, segmentation, and protection throughout the entire network infrastructure, such as a holistic security fabric approach.

Your IoT security must contain the following key abilities:

- **Learn:** With complete network visibility, security solutions can authenticate and classify IoT devices to build a risk profile and assign them to IoT device groups.
- **Segment:** Once the enterprise understands its IoT attack surface, IoT devices can be segmented into policy-driven groups based on their risk profiles.
- **Protect:** The policy-driven IoT groups and internal network segmentation enable monitoring, inspection, and policy enforcement based on the activity at various points within the infrastructure.

Understanding IoT Security Requirements

IoT security requirements support an IoT security strategy that is specific to the business, industry, and network environment. There is a broad swath of protection to be considered in addition to the rigor of practicing administrative oversight, conducting regular patches and updates, enforcing use of strong passwords, and focusing on Wi-Fi security.

Monitoring network and device behavior to detect deviations is a best practice to detect malware from an IoT device vulnerability. Another best practice is network segmentation of IoT devices whereby they connect to a separate network to isolate vulnerable devices and threats to prevent malware from spreading across the enterprise. Applying zero-trust network access provides an additional layer of security.

With the limited configuration capabilities of many IoT devices, instead of trying to secure the IoT firmware and software, you can protect your IoT environment with security solutions that provide multiple layers of protection including endpoint encryption.

As the IoT and the cloud converge, consider securing the technologies with another layer of cloud-based security solutions that also add processing capabilities to devices at the edge.

There are many different protocols used by IoT devices from internet protocols and network protocols to Bluetooth and other communications protocols. Understanding the protocols your devices use can help reduce security risks.

Industries that rely on GPS for critical operations should monitor their GPS connected devices for potential security issues such as fake or jammed GPS signals.

Conducting A Risk Assessment for IoT Devices and Systems

Attackers prey on negligence. They take advantage of organizations that do not oversee IoT devices that are connected to the corporate network. These devices can include anything from rogue devices to overlooked routers with outdated firmware. Understanding the risk of each device that is connected to your network and monitoring individual behavior is critical to prevent cyber attacks.

Also essential to IoT security is maintaining a full inventory of networked devices on the corporate network. Finding a solution that can discover–in minutes–all the IoT connections within your network should be a top priority.

Implementing Strong Authentication and Authorization Mechanism

Authentication is one of the most crucial security measures for an engineer to consider in an IoT deployment. IT administrators can determine which IoT authentication and authorization type, such as one-way, two-way, or three-way, will serve the organization best based on the mechanism's latency and data requirements.

As mentioned above (e.g., default passwords), most IoT devices come with poor authentication. When deploying IoT devices, similar to websites and web apps, one of the best methods for IT

admins to secure IoT devices is to use digital certificates. IoT device certificates are integral to an IoT security strategy.

Ensuring Adequate Encryption and Secure Communications

The primary purpose of encryption is to protect the confidentiality of digital data stored on computer systems or transmitted over the internet or any other computer network. IoT encryption is a key player in securing many different types of IoT devices. By encrypting data communications from IoT devices, an organization stands to gain confidentiality of contents, authentication of origin, data integrity, and awareness of the sender.

Encryption is an effective way to secure data, but the cryptographic keys must be carefully managed to ensure data remains protected, yet accessible when needed. While IoT devices often are not targets themselves, without built-in security, they serve as attractive conduits for the distribution of malware that could result in a data breach.

Data encryption is not a substitute for other information protection controls, such as physical access, authentication and authorization, or network access controls. Data encryption is a method to reduce risk as is the practice of using secure communications protocols and channels for sensitive data.

Although IoT devices are easy to deploy, their communication protocols must have the processing power, range, and reliability to run on existing internet infrastructure as called out in the criteria for IoT implementation (Wi-Fi 802.11 a/b/g/n/ac, etc.).

It is important to consider power consumption when designing an IoT network. Low power wireless networks are best. For this reason, communication protocols created for IoT application requirements generally fall into two groups:

- Low Power Wide Area Networking (LPWAN)
- Wireless Personal Area Networking (WPAN)

Securing Firmware and Software Updates with Patches

Like other digital devices, IoT devices must be patched and updated to prevent threats from taking advantage of vulnerabilities in software and firmware. Installing updates and patching vulnerabilities is essential to IoT security as well as operational technology (OT). When devices cannot be patched or taken offline to prevent exploitation, administrators can deploy intrusion prevention systems (IPS).

Partnering with IoT Security Experts for Effective Risk Management

Managing IoT security on your network could be overwhelming without the help of IoT detection services and tools that discover IoT devices, block malicious traffic, and enable virtual patching. Detection is based on a local (installed) library of IoT devices that is regularly expanded and updated for the latest threats and vulnerabilities. Along with an IPS and network access control, detection services are integral to an IT security strategy for effective risk management.

Which IoT Device Types Are Most Susceptible to Security Risks?

Cyber attacks are used to exploit unprotected IoT devices with tactics such as network scanning, remote code execution, and command injection. The healthcare industry has the highest share of IoT security issues from internet connected devices used for medical imaging systems, patient monitoring systems, and medical device gateways. Another high-risk sector includes commonly used IoT devices such as security cameras and printers. Consumer electronics, IP phones, and energy management devices are also at higher risk.

Examining the Top Industries Vulnerable to IoT Security Breaches

Many industries have adopted IoT at the risk of higher exposure to cyber threats from vulnerabilities in IoT devices. Some industries are more vulnerable than others due to the sensitive nature of their data (e.g., medical records, autonomous vehicles, or intellectual property).

These include large organizations with complex networks, digital factories and plants that rely on industrial operational technology (OT), and healthcare organizations that use medical IoT for patient care such as networked scanners, monitoring tools, wearable devices, and other internet connected systems.

IoT devices are not built to meet the business and regulatory requirements of critical industries. If developers integrated security into IoT devices and software, it would go a long way to help protect sensitive data and prevent exploitation when those devices go online.

IoT Applications:

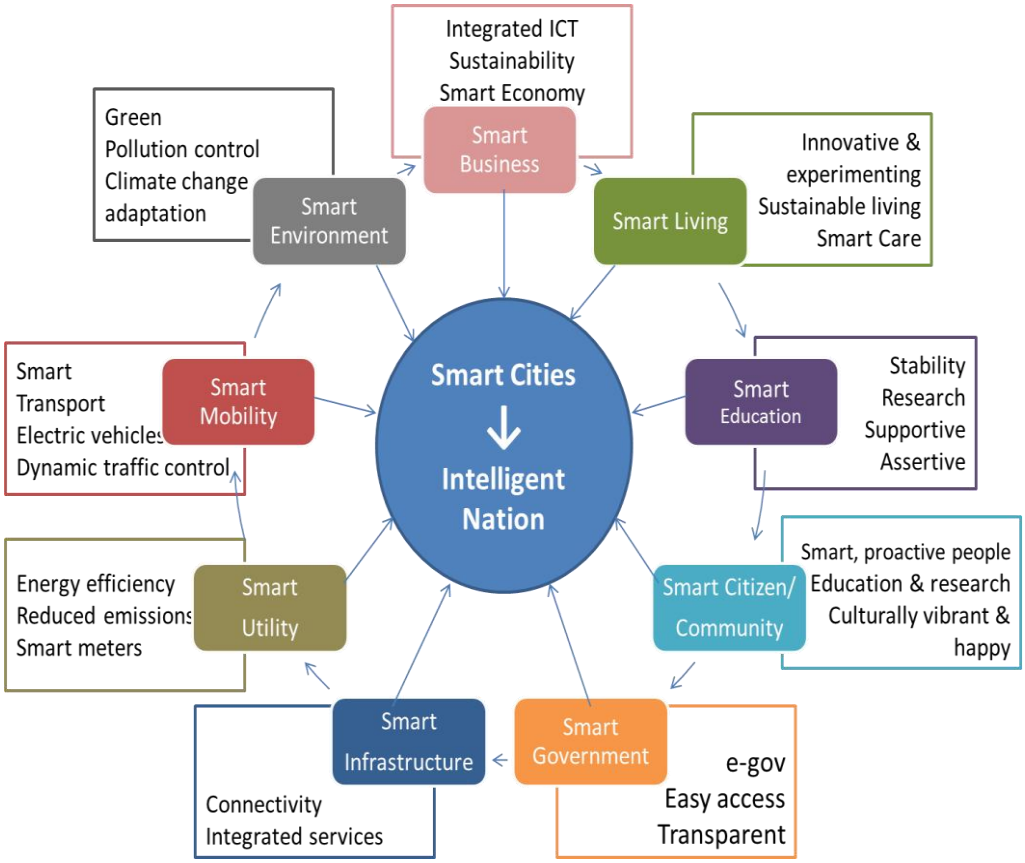
IoT applications promise to bring immense value into our lives. With newer wireless networks, superior sensors and revolutionary computing capabilities, the **Internet of Things** could be the next frontier in the race for its share of the wallet. IoT applications are expected to equip billions of everyday objects with connectivity and intelligence. It is already being deployed extensively, few applications of IoT:

- Wearables
- Smart Home Applications
- Smart Buildings
- Smart Infrastructure
- Securities
- Health Care
- Smart Cities
- Agriculture
- Industrial Automation

Smart Cities:

The urbanization process has greatly improved people's standard of living, providing water supplies and sewerage systems, residential and office buildings, education and health services and convenient transportation. The concentration of educated people in cities helps to improve the industrial structure and promote production efficiency. However, urbanization also creates new challenges and problems. As a representative developing country, the economic advantages of Indian cities are being offset by the perennial urban curses of overcrowding, air and water pollution, environmental degradation, contagious diseases and crime; the urban issues of reducing air pollution and providing clean water, safe neighborhoods and efficient infrastructure desperately need to be addressed.

All these challenges and problems force citizens, governments and stakeholders to pay attention to the environment and sustainable development of cities, and to try to find a set of technical solutions to reduce these urban problems. The Information and Communication Technology (ICT) revolution has offered people the opportunity to reduce the scale of and/or solve urbanization issues. During the past 10 years, city systems have become more digital and information-based, and there has been a fundamental change in the living environment of citizens and the governing mode of cities. The economy, culture, transport, entertainment and all other aspects of cities have become closely combined with ICT, and the Internet has become a major part of citizens' daily lives. The abundant accomplishments of digitizing a city's information not only introduce daily convenience to the population, but also establish an infrastructure and conglomeration of data as a basis for further evolution of modern cities. Over the last 10 years, innovative information technologies such as cloud computing, 'big data', data vitalization, the 'Internet of Things' and mobile computing have become widely adopted in a variety of different areas. Cloud computing enables, developers to provide internet services without the need for a large capital outlay on hardware for deployment or the staff to operate it. The amount of information published and processed both on- and offline has given rise to an information explosion, and a new field dedicated to dealing with it—big data—which has spawned the need for new, more scalable, techniques to derive answers from huge sets of data. The emergence of the Internet of Things makes it possible to access remote sensor data and to control the physical world from a distance, meaning that cities can effectively sense and manage essential elements such as the water supply, building operations, and road and transport networks. Data vitalization proposes a new paradigm for large-scale dataset analysis and offers ubiquitous data support for top-level applications for smart cities. With the help of mobile computing, users can access and process information anywhere, and anytime, on all aspects of life. The urbanization, growth and associated problems of modern cities, coupled with the rapid development of new ICT, has enabled us to first envisage the 'smart cities' concept, and now to begin to build smart cities, which is seen as the future form for cities. Figure 1 shows how a smart city is formed. Smart city includes smart business, smart living, smart education, smart community, smart government, smart infrastructure, smart utility, smart mobility and smart environment.



The new Internet of Things (IoT) applications are enabling Smart City initiatives worldwide. It provides the ability to remotely monitor, manage and control devices, and to create new insights and actionable information from massive streams of real-time data. The main features of a smart city include a high degree of information technology integration and a comprehensive application of information resources. The essential components of urban development for a smart city should include smart technology, smart industry, smart services, smart management and smart life. The Internet of Things is about installing sensors (RFID, IR, GPS, laser scanners, etc.) for everything, and connecting them to the internet through specific protocols for information exchange and communications, in order to achieve intelligent recognition, location, tracking, monitoring and management. With the technical support from IoT, smart city need to have three features of being instrumented, interconnected and intelligent. Only then a Smart City can be formed by integrating all these intelligent features at its advanced stage of IOT development. The explosive growth of Smart City and Internet of Things applications creates many scientific and engineering challenges that call for ingenious research efforts from both academia and industry, especially for the development of efficient, scalable, and reliable Smart City based on IoT. New protocols, architectures, and services are in dire needs to respond for these challenges. The goal of the special issue is to bring together scholars, professors, researchers, engineers and administrators resorting to the state-of-the-art technologies and ideas to significantly improve the field of Smart City based on IoT.

So, when we talked about smart cities; what is it. So, in addition to the regular infrastructure that is there in any city for example, the urban infrastructure consisting of office buildings residential areas hospitals schools transportation police and so on you also need something in addition to make the cities smart. So, what is this in addition let us talk about. So, smart means what smart means that it is in terms of the services that are given to the respective stake holders of these cities. So, citizens are able to do things in a better manner in an improved manner then usual and how is that made possible that is made possible with the help of nothing, but the ICT technologies information and communication technologies which also includes electronics embedded electronics different other advanced topologies in electrical in a electrical sciences and so on. So, computers electronics put together can make these cities smart.So, definitely will have to take help of sensors ,sensor networks sensor networks then actuators then the different other communication technologies RFID, NFC, ZWAVE and so and so forth.

Modern urban spaces are hotbeds of new ideas and world-shaking innovations. As for urban adoption of connected tech: all things considered, it really makes practical sense. Densely populated areas stand to gain the most from improved surroundings, and depending on the city, they might already come equipped with the fundamental IT infrastructures, which makes the further adaption easier. Meanwhile, the IoT might also offer some solutions to ease the huge burden that the urban explosion has meant for the existing infrastructures.

Connected City

A common definition for a smart city is using ICT to make a city (administration, education, transportation, etc.) more intelligent and efficient. The definitions and concepts of smart cities are still emerging, and there is currently no clear and consistent definition of a smart city among the different stakeholders. In order to implement and assess smart cities in practice, a deeper understanding of the 'smart city' still needs to be defined. Many countries and cities have launched their own smart city projects to resolve urbanization issues and challenges. The USA was one of the first countries to launch a smart city project with a high compliment of smarter planet notions from President Barack Obama. In particular, for developing countries, the speed of urbanization is considerably faster and, as a consequence, the infrastructure problems faced are much greater. In 2014, India declared an intention to build more than 100 smart cities, with high-technology communication capabilities, throughout the country. ICT plays an important role in smart city construction. Top-level architecture research plays a considerable role in guiding technology development in every domain of a smart city and improving research into resource configuration. Now let's extrapolate the potential use cases to an entire city in which we have many objects that are capable of capturing information and interacting with other objects. The street lamp can now not only communicate with the devices that are closest but with other objects that are connected to the Internet and process this information to make decisions, for example, about the intensity of the light that is the most appropriate. The objects can also send information about what is happening in their environment or process different information. If the information from the street lamp is processed alongside with information from a nearby traffic light, we can start talking about the IoT use cases in the smart city environment.

When it comes to smart cities and the management of public space, the scope of possibilities, that IoT offers, is infinite. In other words, the IoT comes with considerable possibilities and room for manoeuvre within the field of smart cities. It is one of the aspects that we will touch in the Master's in Global Smart City Manager. IoT is a technology that is already there, that has been developed for a long time, but whose implantation in the public space will prevail in the years to come.

And depending on the way we approach our smart city project or the implementation of this technology in public space, smart city projects will be developed in one way, or another, they will be able to achieve common objectives in one way or another.

Possible IoT Use Cases for Smart Cities

- **Smart parking**

An IoT solution will permit monitoring the availability of parking spots in the city. With the GPS data from drivers' smartphones (or road-surface sensors in the ground), smart parking solutions let the user know when the closest parking spot becomes free to find a parking spot faster and easier instead of blindly driving around.

- **Smart roads and smart traffic congestion management**

Different IoT solutions will permit to monitor vehicle and pedestrian levels to optimize driving and walking routes. The use of different types of sensors, as well as GPS data from drivers' smartphones will help to determine the number, location and the speed of vehicles. Thanks to a cloud management platform which connects various traffic lights, the city will be able to monitor green light timings and automatically alter the lights based on the current traffic situation to prevent congestion. Better control of traffic congestion will also help to improve air quality**.**

- **Smart public transport**

With the help of IoT sensors, we can obtain data to learn about the patterns of how citizens use public transport. Smart public transport solutions can combine multiple sources, such as ticket sales and traffic information. The users could also use an app to contact the authorities in case they spot incidents or suspicious activities.

- **Smart street lighting**

IoT-based smart cities allow better maintenance and control of street lamps. Equipping streetlights with sensors and connecting them to a cloud management solution makes them more straightforward and cost-effective. With this system, the city can adapt the lighting schedule to the lighting zone and weather conditions.

- **Smart waste management**

Waste-collecting is another service that could be optimized with an IoT-enabled solution by tracking waste levels, as well as providing route optimization and operational analytics.

Advantages of a Smart City

Smart cities can be described as cities capable of using information technology to create efficiencies and create sustainability, and improve the quality of life of it's residents. A smart city is basically a living entity, capable of extraordinary adaptations that we once thought were not possible. This post will be discussing smart cities, including what makes a smart city, it's benefits, it's effects on the environment, and what negative effects, if any it might have on it's citizens and the world as a whole.

The benefits of smart cities

- **Efficient distribution of resources**

Smart cities have an overall better organization and infrastructure. All the sectors are involved in a complex interplay that simplifies everyday life for people who live and work in the city. The cameras at the bus stops can identify how many people are waiting to board; the sensors on the approaching bus know how many people ride the bus at any given point in time, and how many people are currently on the bus. The combination of the information from the bus stop and the bus then leads to the city's response. There can then be redistribution of people and buses if it appears that the current course of events will not be efficient.

- **Seamless communication**

Communications between the various systems and sensors in a smart city is very important. In fact, without them the smart city cannot efficiently redistribute resources and make citizens' lives better. However, smart cities bring about a different, equally efficient communication–the communication between the citizens and the government of the particular city.

In prior times, policies and programs were made based on what the government perceived to be needed by the city. This often led to massive oversights and the omission of key policies altogether. In a smart city, the

policy makers have all they could ever need to make informed decisions. The information gathered all across the city provide an invaluable line of communication between the needs of the city, and the people who can address those needs.

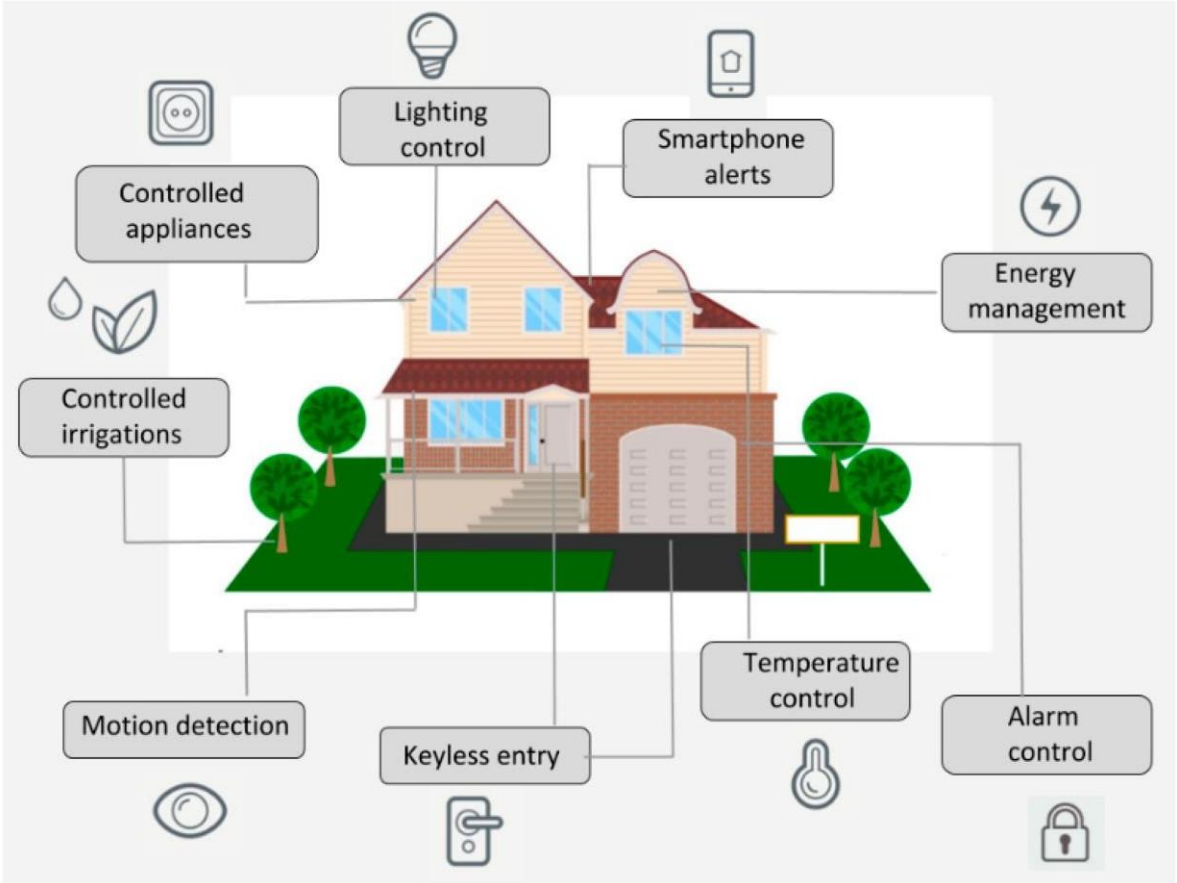
- **Speed of implementation**

Still on governments and policies, every country with a democracy can testify to the fact that it takes quite a while for policies, or any sort of new development to get implemented. This is partly due to bureaucracy and the multiple levels of government, and also partly due to the human factor. Smart cities overcome these problems very easily. Because the points that need improvements have already been identified, the implementation becomes easier. All the automation, analytics, and sensors contribute to making it easier for most of the changes to be implemented remotely, creating a seamless flow of change from conception to execution.

Smart Home:

The Internet of Things (IoT) is a system that allows devices to be connected and remotely monitored across the Internet. In the last years, the IoT concept has had a strong evolution, being currently used in various domains such as smart homes, telemedicine, industrial environments, etc. Wireless sensor network technologies integrated into the IoT enable a global interconnection of smart devices with advanced functionalities. A wireless home automation network, composed of sensors and actuators that share resources and are interconnected to each other, is the key technology to making intelligent homes.

A “smart home” is a part of the IoT paradigm and aims to integrate home automation. Allowing objects and devices in a home to be connected to the Internet enables users to remotely monitor and control them. These include light switches that can be turned on and off by using a smartphone or by voice command, thermostats that will adjust the indoor temperatures and generate reports about energy usage, or smart irrigation systems that will start at a specific time of a day, on a custom monthly schedule, and thus will control water waste. Smart home solutions have become very popular in the last years. Figure 1 shows an example of a smart home that uses different IoT-connected utilities.



One of the greatest advantages of home automation systems is their easy management and control using different devices, including smartphones, laptops and desktops, tablets, smart watches, or voice

assistants. Home automation systems offer a series of benefits; they add safety through appliance and lighting control, secure the home through automated door locks, increase awareness through security cameras, increase convenience through temperature adjustment, save precious time, give control, and save money.

Several home automation systems involved with IoT have been proposed by academic researchers in the literature in the last decade. In wireless-based home automation systems, different technologies have been used, each of them with their pros and cons. For example, Bluetooth-based automation is low cost, fast, and easy to be installed, but it is limited to short distances. GSM and ZigBee are widely used wireless technologies as well. GSM provides long-range communication at the cost of a mobile plan of the service provider that operates in the area. Zigbee is a wireless mesh network standard that is designed to be low-cost and with low power consumption, targeted at battery-powered devices in wireless control and monitoring applications. However, it has a low data speed, low transmission, as well as low network stability, and has a high maintenance cost. The advantages of Wi-Fi technology over ZigBee or Z-Wave are related to price, complexity (meaning simplicity), and accessibility. First, Wi-Fi-enabled smart devices are usually cheap. In addition, it is easier to find do-it-yourself devices that use Wi-Fi, resulting a less expensive option. Second, Wi-Fi is already a necessity and it is in most homes, so it is easier to buy devices that are already Wi-Fi-enabled. Finally, Wi-Fi is characterized by simplicity, meaning that a user must connect only a minimal number of devices for a home automation setup. Since it is very common, the investment on extra hardware is avoided; a user only needs the basic setup for a home automation system. However, Wi-Fi is not designed to create mesh networks, it consumes ten times more energy than similar devices using ZigBee

, Z-Wave, or Bluetooth for example, and many Wi-Fi routers can only allow up to thirty devices connected at once. As compared to Ethernet, Wi-Fi brings several advantages, including the easy connection and access of multiple devices, the expandability (adding new devices without the hassle of additional wiring), lower cost, or single access point requirement.

The basic architecture enables measuring home conditions, process instrumented data, utilizing microcontroller-enabled sensors for measuring home conditions and actuators for monitoring home embedded devices.The popularity and penetration of the smart home concept is growing in a good pace, as it became part of the modernization and reduction of cost trends. This is achieved by embedding the capability to maintain a centralized event log, execute machine learning processes to provide main cost elements, saving recommendations and other useful reports.

Smart home services

- **Measuring home conditions**

A typical smart home is equipped with a set of sensors for measuring home conditions, such as: temperature, humidity, light and proximity. Each sensor is dedicated to capture one or more measurement. Temperature and humidity may be measured by one sensor, other sensors calculate the light ratio for a given area and the distance from it to each object exposed to it. All sensors allow storing the data and visualizing it so that the user can view it anywhere and anytime. To do so, it includes a signal processer, a communication interface and a host on a cloud infrastructure.

- **Managing home appliances**

Creates the cloud service for managing home appliances which will be hosted on a cloud infrastructure. The managing service allows the user, controlling the outputs of smart actuators associated with home appliances, such as such as lamps and fans. Smart actuators are devices, such as valves and switches, which perform actions such as turning things on or off or adjusting an operational system. Actuators provides a variety of functionalities, such as on/off valve service, positioning to percentage open, modulating to control changes on flow conditions, emergency shutdown (ESD). To activate an actuator, a digital write command is issued to the actuator.

- **Controlling home access**

Home access technologies are commonly used for public access doors. A common system uses a database with the identification attributes of authorized people. When a person is approaching the access control system, the person’s identification attributes are collected instantly and compared to the database. If it matches the database data, the access is allowed, otherwise, the access is denied. For a wide distributed

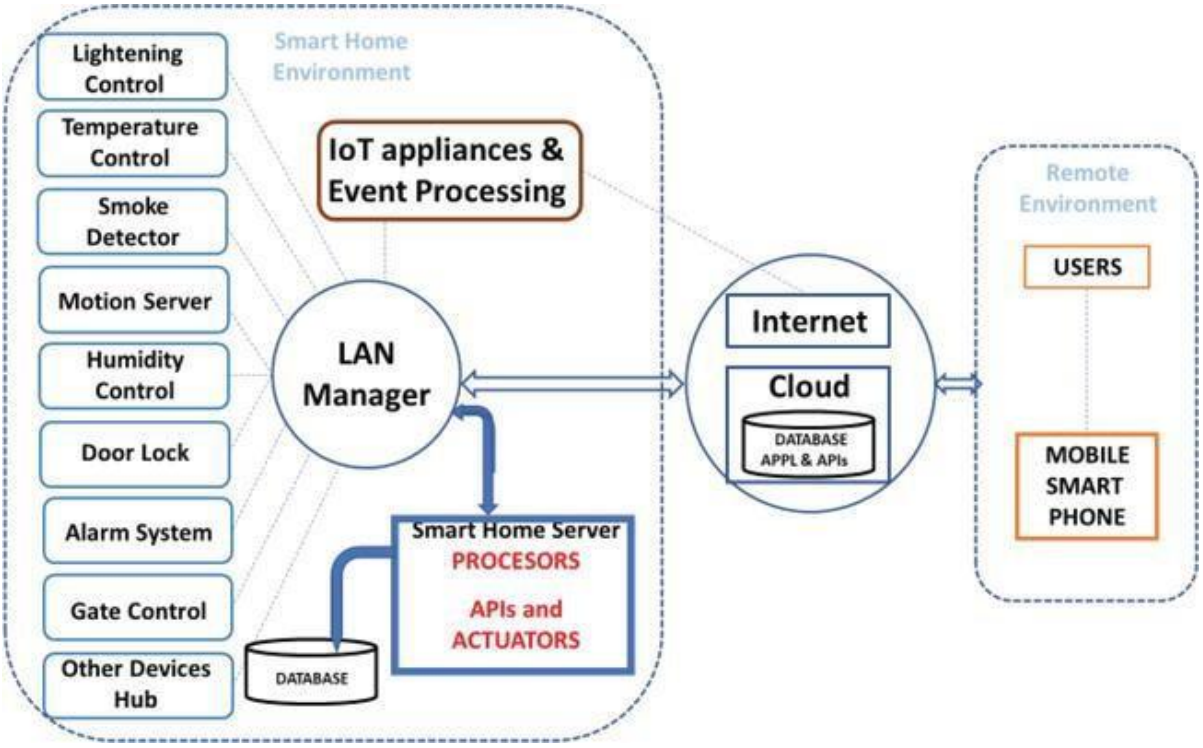
institute, we may employ cloud services for centrally collecting persons' data and processing it. Some use magnetic or proximity identification cards, other use face recognition systems, finger print and RFID.

In an example implementation, an RFID card and an RFID reader have been used. Every authorized person has an RFID card. The person scanned the card via RFID reader located near the door. The scanned ID has been sent via the internet to the cloud system. The system posted the ID to the controlling service which compares the scanned ID against the authorized IDs in the database.

The main components

To enable all of the above described activities and data management, the system is composed of the following components, as described in Figure 1.

- Sensors to collect internal and external home data and measure home conditions. These sensors are connected to the home itself and to the attached-to-home devices. These sensors are not internet of things sensors, which are attached to home appliances. The sensors' data is collected and continually transferred via the l
- ocal network, to the smart home server.
- Processors for performing local and integrated actions. It may also be connected to the cloud for applications requiring extended resources. The sensors' data is then processed by the local server processes.
- A collection of software components wrapped as APIs, allowing external applications execute it, given it follows the pre-defined parameters format. Such an API can process sensors data or manage necessary actions.
- Actuators to provision and execute commands in the server or other control devices. It translates the required activity to the command syntax; the device can execute. During processing the received sensors' data, the task checks if any rule became true. In such case the system may launch a command to the proper device processor.
- Database to store the processed data collected from the sensors [and cloud services]. It will also be used for data analysis, data presentation and visualization. The processed data is saved in the attached database for future use.



- Cloud computing and its contribution to IoT and smart home: Cloud computing is a shared pool of computing resources ready to provide a variety of computing services in different levels, from basic infrastructure to most sophisticated application services, easily allocated and released with minimal

efforts or service provider interaction. In practice, it manages computing, storage, and communication resources that are shared by multiple users in a virtualized and isolated environment. IoT and smart home can benefit from the wide resources and functionalities of cloud to compensate its limitation in storage, processing, communication, support in pick demand, backup and recovery. For example, cloud can support IoT service management and fulfillment and execute complementary applications using the data produced by it. Smart home can be condensed and focus just on the basic and critical functions and so minimize the local home resources and rely on the cloud capabilities and resources. Smart home and IoT will focus on data collection, basic processing, and transmission to the cloud for further processing. To cope with security challenges, cloud may be private for highly secured data and public for the rest.

IoT challenges for Smart City and Smart Home:

- **Infrastructure**

Smart Cities utilize sensor technology to gather and analyze information in an effort to improve the quality of life for residents. Sensors collect data on everything from rush hour stats to crime rates to overall air quality. Complicated and costly infrastructure is involved in installing and maintaining these sensors. How will they be powered? Will it involve hard-wiring, solar energy, or battery operation? Or, in case of power failure, perhaps a combination of all three? Funding for new infrastructure projects is limited and approval processes can take years. Installing new sensors and other improvements cause temporary – though still frustrating – problems for people living in these cities.

- **Security and Hackers**

As IoT and sensor technology use expands, so does the threat level to security. This begs the question…is technology really considered "smart" if hackers can break into it and shut down an entire city? Recent discussion involving cyber-terror threats to vulnerable and outdated power grids has everyone a bit more concerned and skeptical about technology and security. Smart Cities are investing more money and resources into security, while tech companies are creating solutions with new built-in mechanisms to protect against hacking and cyber-crimes.

- **Privacy Concerns**

In any major city, there's a balance between quality of life and invasion of privacy. While everyone wants to enjoy a more convenient, peaceful, and healthy environment, nobody wants to feel like they are constantly being monitored by "Big Brother."

Cameras installed on every street corner may help deter crime, but they can also install fear and paranoia in law-abiding citizens. Another valid concern is the amount of data being collected from all the smart sensors residents come into contact with each day.

- **Educating & Engaging the Community**

For a Smart City to truly exist and thrive, it needs "smart" citizens who are engaged and actively taking advantage of new technologies. With any new city-wide tech project, part of the implementation process must involve educating the community on its benefits. This can be done through a series of in-person town hall-style meetings and email campaigns with voter registration, as well as an online education platform that keeps citizens engaged and up-to-date. When a community feels like it's playing a part in the overall decisions that affect daily life, and is being communicated to in a clear and thoughtful manner, it's more apt to use the technology and encourage others to use it as well. This is key to a Smart City's success.

Connected vehicles:

Connected vehicle technology can change our transportation system as we know it by enabling safe, interoperable networked wireless communications among vehicles, the infrastructure, and passengers' personal communications devices. Connected vehicle technology will enable cars, trucks, buses, and other vehicles to "talk" to each other with in-vehicle or aftermarket devices that continuously share important safety and mobility information. Connected vehicles can also use wireless communication to "talk" to traffic signals, work zones, toll booths, school zones, and other types of infrastructure.

Different communications technologies (satellite, cellular, dedicated short range communications) may be utilized depending on the performance requirements of the connected vehicle applications. Cars, trucks, buses, and other vehicles can "talk" to each other with in-vehicle or aftermarket devices that continuously

share important safety and mobility information. Connected vehicles can also use wireless communication to "talk" to traffic signals, work zones, toll booths, school zones, and other types of infrastructure. The vehicle information communicated does not identify the driver or vehicle, and technical controls have been put in place to help prevent vehicle tracking and tampering with the system. The vision for connected vehicle technologies is to transform surface transportation systems to create a future where:

• Highway crashes and their tragic consequences are significantly reduced

• Traffic managers have data to accurately assess transportation system performance and actively manage the system in real time, for optimal performance

• Travelers have continual access to accurate travel time information about mode choice and route options, and the potential environmental impacts of their choices

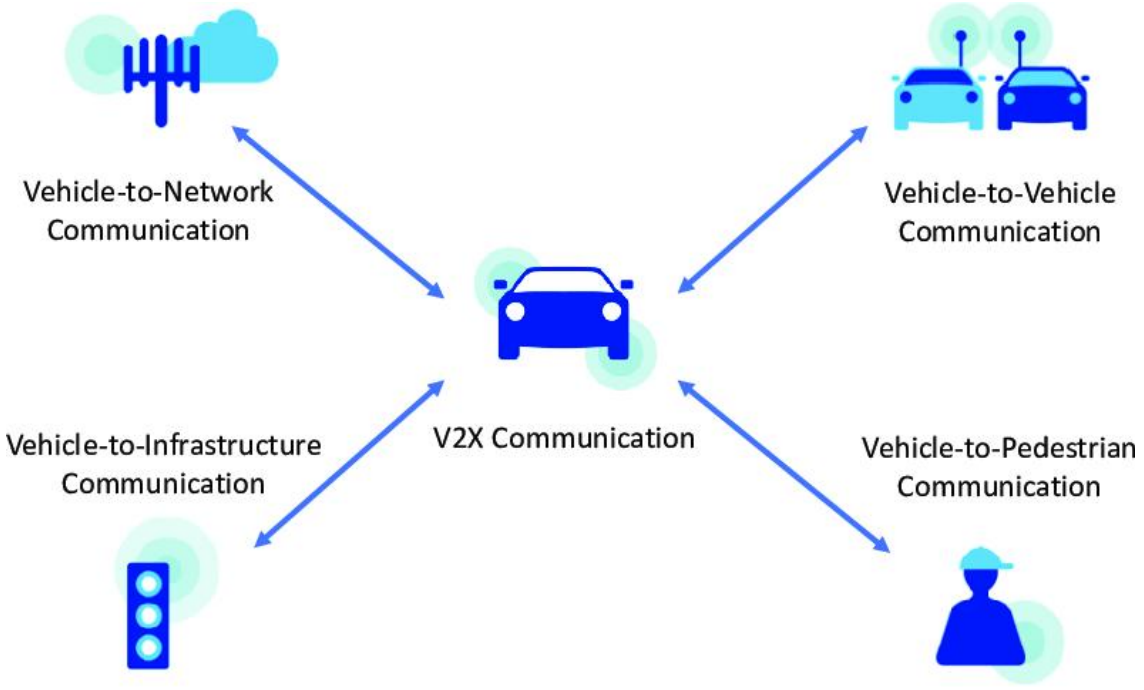
• Vehicles can talk to traffic signals to eliminate unnecessary stops and help drivers operate vehicles for optimal fuel efficiency.

Challenges:

- Security
- Privacy
- Scalability
- Reliability
- Quality of service
- Lack of Global Standards

What is Vehicle to Everything (V2X)?

Vehicle to Everything (V2X) is a vehicular communication system that supports the transfer of information from a vehicle to moving parts of the traffic system that may affect the vehicle. The main purpose of V2X technology is to improve road safety, energy savings, and traffic efficiency on the roads.



How Vehicle to Everything (V2X) Works

In a V2X communication system, the information travels from the vehicle sensors and other sources through high-bandwidth, high-reliability links, allowing it to communicate with other cars, infrastructure such as parking spaces and traffic lights, and smartphone-tossing pedestrians. By sharing information, such as speed, with other entities around the vehicle, the technology improves the driver's awareness of potential dangers and helps reduce the severity of injuries, road accident fatalities, and collision with other vehicles. The technology also enhances traffic efficiency by warning drivers of upcoming traffic, suggesting alternative routes to avoid traffic and identifying available parking spaces.

Components of V2X Technology

The key components of V2X technology include V2V (vehicle-to-vehicle) and V2I (vehicle-to-infrastructure). V2V allows vehicles to communicate with other vehicles on the road, while V2I allows vehicles to communicate with external entities, such as traffic lights, parking spaces, cyclists, and pedestrians. The technologies help improve road safety, reduce fuel consumption, and enhance the experience between drivers and other road users, such as cyclists and pedestrians. When V2X systems are integrated into traditional vehicles, drivers can receive important information about the weather patterns, nearby accidents, road conditions, road works warning, emergency vehicle approaching, and activities of other drivers using the same road. Autonomous vehicles equipped with V2X systems may provide more information to the existing navigation system of the vehicle. The systems also make it possible for autonomous vehicles to scan the surrounding environment and make immediate decisions based on the information received.

Smart Grid:

What is the Smart Grid?

The "grid" is the electrical network serving every resident, business and infrastructure service in a city. The "smart grid" is the next generation of those energy systems, which have been updated with communications technology and connectivity to drive smarter resource use.
The technologies that make today's IoT-enabled energy grid "smart" include wireless devices such as sensors, radio modules, gateways and routers. These devices provide the sophisticated connectivity and communications that empower consumers to make better energy usage decisions, allow cities to save electricity and expense, and enables power authorities to more quickly restore power after a blackout.
The Smart Grid is critical to building a secure, clean, and more efficient future, according to the International Energy Agency (IEA). The Smart Grid is part of an IoT framework, which can be used to remotely monitor and manage everything from lighting, traffic signs, traffic congestion, parking spaces, road warnings, and early detection of things like power influxes as the result of earthquakes and extreme weather. The Smart Grid does this through a network of transmission lines, smart meters, distribution automation, substations, transformers, sensors, software and more that are distributed to businesses and homes across the city.
Smart Grid technologies all contribute to efficient IoT energy management solutions that are currently lacking in the existing framework. What makes the IoT Smart Grid better is two-way communication between connected devices and hardware that can sense and respond to user demands. These technologies mean that a Smart Grid is more resilient and less costly than the current power infrastructure.

The main advantages identified in this document are:

- **Energy savings through reducing consumption**

One of the advantages of smart grids is that they can tell us the consumption at an energy meter at any time, so users are better informed of their real consumption. Moreover, with better consumption monitoring, contracted power can be adjusted to meet the real need of each consumer. These two factors result in users reducing their consumption and tailoring their contracted power to their real needs.

- **Better customer service and more accurate bills**

Another key advantage offered by tele-management systems is that bills are more accurate. They always reflect the real consumption of each month instead of estimates, reducing the cost of the old system of manual energy meter readings. In addition to being able to access information about the installation remotely, problems become easier to diagnose and solutions can therefore be implemented faster, improving customer service. Now a days customers have to notify companies for them to take action. But with remote management the system itself automatically reports all incidents to the electric company so it can respond faster to users.

- **Reduced balancing cost**

Smart Grids can collect much more data than the manual energy meter reading system. This permits the use of data analysis techniques and the preparation of highly realistic consumption forecasts as many more variables are taken into account. Utilities can then better tailor their production to consumption (balances) and reduce energy surpluses.

- **Reduction of carbon emissions**

All the benefits above involve reducing consumption, which entails a reduction in CO2 emissions. We can thus say that Smart Grids lead to a more sustainable future. All this will directly contribute to the future integration of electric vehicle charging systems on the mains. The deployment of renewable energy systems is also made easier as utilities gain greater control of their grids.

- **Smart Grid Enables Renewable Energy Generation**

Traditional energy grids are designed to transmit electricity from a large, centralized power station to a wide network of homes and businesses in the area. At this stage, the electric grid is not designed to accept inputs from homes and businesses that are generating power via solar panels or windmills. A smart grid is designed to accept power from renewable resources. Crucially, the smart grid in conjunction with wirelessly enabled smart meters can keep track of how much energy a net-positive establishment is generating and reimburse them accordingly. The smart grid also allows for monitoring of solar panels and equipment as well. We mentioned earlier that a smart grid can mitigate the effects of a disaster such as a terrorist attack or natural disaster on a power station, a feat that's possible due to decentralized energy generation. Under the traditional model, a small number of power plants powered a city. This left these services vulnerable to threats that would result in widespread blackouts and energy shortages. With a decentralized model, even if the centralized power plant is taken offline, multiple alternative sources, including wind and solar, can supplant the resources in the grid. This decentralized system is much harder to take offline and can provide a robustness that's not possible when one plant is powering an entire city.

Smart Grid for the Future

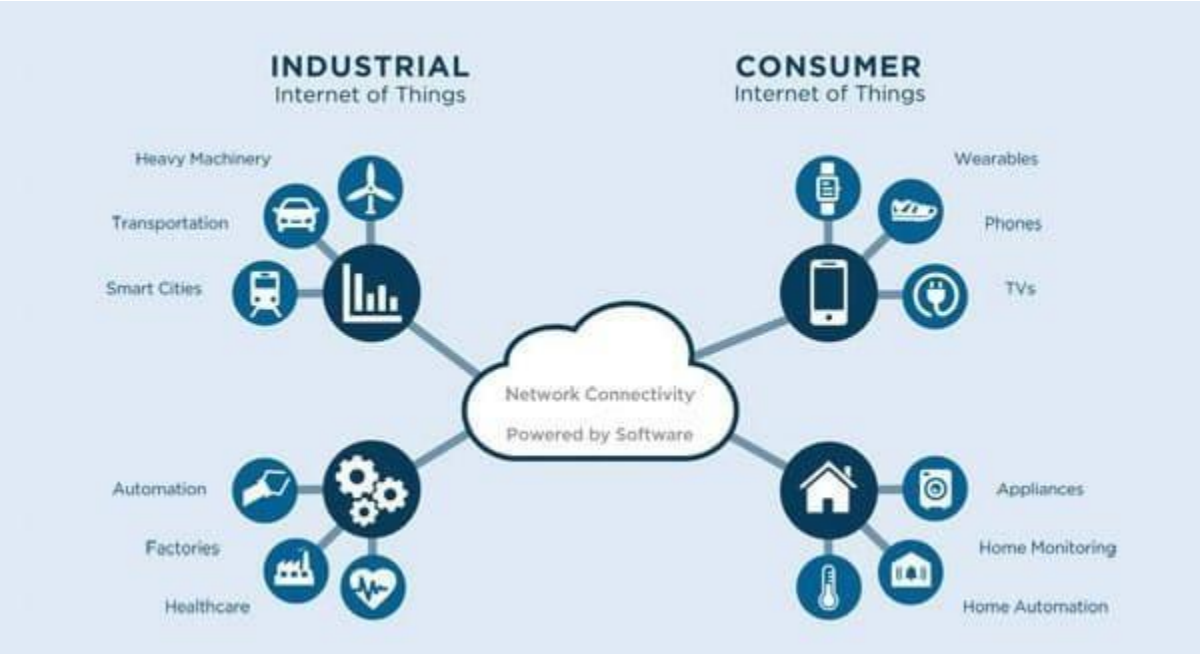
Smart grid technology can be expressed in a single sentence: a new electric grid with two-way communication. For the first time, businesses and consumers can get real time billing information while utility companies can better meet the needs of their customers as they react to demand spikes and fix or manage blackouts and other challenges. Smart grid is resilient, efficient and green which is good for the consumer, the utility company and the environment. Wireless technology will replace thousands of miles of cable that would have been needed to advance the smart grid to where it is today.

Challenges:

- High Investment
- Cyber attacks

Industrial IoT:

The Industrial Internet of Things (IIoT), which is considered as the main future IoT-application area, is defined by the Industrial Internet Consortium as machines, computers and people enabling intelligent industrial operations using advanced data analytics for transformational business outcomes" ("Industrial Internet Consortium," 2017). Generally, IioT is one basis of Industry 4.0 and the digital transformation. The IioT is the connection between IT (information technology) and OT (operational technology). IioT is the most important segment in IoT, much more than consumer applications. The Industrial Internet of Things is related to the Industry 4.0: all IoT applications in Industry 4.0 are forms of IioT but not all IioT use cases are about the industries which are categorized as Industry 4.0. Typical use cases of the Industrial Internet of Things include intelligent machine applications, industrial control applications, factory floor use cases, condition monitoring, use cases in agriculture or smart grid applications. It is important to know that the IioT is not just about saving costs and optimizing efficiency though. Companies also have the possibility to realize important transformations and can find new opportunities, e.g., entirely new business models in Industry 4.0.



According to TechTarget, IioT can be formally defined as "the use of smart sensors and actuators to enhance manufacturing and industrial processes. Also known as the industrial internet or Industry 4.0, IioT leverages the power of smart machines and real-time analytics to take advantage of the data that dumb machines have produced in industrial settings for years."
Industrial IoT capabilities require widespread digitization of manufacturing operations. Organizations must include four primary pillars to be considered a fully IioT-enabled operation:

- Smart machines equipped with sensors and software that can track and log data.
- Robust cloud computer systems that can store and process the data.
- Advanced data analytics systems that make sense of and leverage data collected from systems, informing manufacturing improvements and operations.
- Valued employees, who put these insights to work and ensure proper manufacturing function.

Benefits of IioT

These are 5 of the biggest benefits of adopting a fully connected IIoT manufacturing operation.

Increase efficiency

The biggest benefit of IIoT is that it gives manufacturers the ability to automate, and therefore optimize their operating efficiency. Robotics and automated machinery can work more efficiently and accurately, boosting productivity and helping manufacturers streamline their functions.

Additionally, physical machinery can be connected to software via sensors that monitor performance on a constant basis. This enables manufacturers to have better insights into the operational performance of individual pieces of equipment as well as entire fleets.
IIoT-enabled data systems empower manufacturers to improve operating efficiencies by:

- Bypassing manual tasks and functions and implementing automated, digital ones
- Making data-driven decisions regarding all manufacturing functions
- Monitoring performance from anywhere – on the manufacturing floor or from thousands of miles away

Reduce Errors

Industrial IoT empowers manufacturers to digitize nearly every part of their business. By reducing manual process and entries, manufacturers are able to reduce the biggest risk associated with manual labor – human error.
This goes beyond just operational and manufacturing errors. IIoT solutions also can reduce the risk of cyber and data breaches caused by human error. A Cyber Security Trend report cited people as the biggest cause of cyber security breaches, with human error being the culprit 37% of the time. AI and machine learning-enabled programs and machinery can do much of the required computing themselves, eliminating the potential for someone to make a simple mistake, and put the manufacturer's data at risk.

Predictive Maintenance

Nothing negatively impacts a manufacturing operation more than machine downtime. When maintenance in the manufacturing world is reactive rather than proactive, manufacturers are stuck trying to identify what the issue is, how it can be repaired, and what it will cost. With predictive maintenance powered by industrial IoT solutions, all of those issues are alleviated.
When machinery performance and function is monitored consistently, manufacturers can create a baseline. This baseline and the corresponding data empowers companies with the information they need to see any issue before it occurs. They can then schedule maintenance prior to downtime, which benefits them in that they:

- Have the parts required for the job
- Know the cost of the project beforehand, and can budget for it
- Move production to another area of the facility, so the product quotas are unaffected
- Ensure that machinery is operating at maximum efficiency

Improve Safety

All of the data and sensors required of a fully functioning IIoT manufacturing operation are also helping to bolster workplace safety. "Smart manufacturing" is turning into "smart security" when all of the IIoT sensors work together to monitor workplace and employee safety.
Integrated safety systems are protecting workers on the floor, on the line, and in distribution. If an accident occurs, everyone in the facility can be alerted, operations can cease, and company leadership can intervene and make sure the accident and incident is resolved. This incident can also generate valuable data that can help prevent a repeat occurrence in the future.
A newer option some manufacturers are utilizing is the use of wearable technology among their employees. Wearables have been part of IoT since its infancy, and it are just now being utilized in industrial IoT operations.
Wearables help leadership keep tabs on things like employee posture and the surrounding noise levels, and they can then improve work conditions and potentially improve performance. They can also alert employees when they aren't following proper workplace safety procedures, so they can correct their actions and stay safe on the job.

Reduce Costs

Knowledge is power, and the knowledge provided to manufacturers via IIoT solutions is giving them the tools they need to reduce costs and generate more revenue. Data-driven insights into operations, production, marketing, sales, and more can steer businesses in a profitable direction.

All of the aforementioned benefits of IIoT – predictive maintenance, fewer errors, improved quality control, and maximized efficiencies – will all boost profits for a manufacturer. Industrial IoT also offers arguably the most valuable tool for leaders of a manufacturing company – insights from anywhere, anytime.

Remote monitoring of manufacturing operations is now possible 365 days a year, 24/7, from anywhere in the world. This 360-degree view into the entire manufacturing process, and the follow-up service provided to customers in their buying journey, is an invaluable asset.

UDHAYAS SPNC
PUBLICATION

Internet of Things

Author 1

Dr T R Nisha Dayana., M.Sc., M.Phil, Ph.D.,
Assistant Professor
Department of Computer Science & IT,
VISTAS, Chennai.

Author 2

Dr S.PRATHIBA M.sc, M.Phil, Ph.D.,
Assistant Professor,
Department of Advanced Computing and Analytics,
VISTAS.

Author 3

Ms. Selin Chandra C S., MCA, M.Phil., (PhD).,
Assistant Professor
Department of Computer Science & IT
VISTAS

INTERNET OF THINGS

ISBN : 978-93-49030-01-5

BOOK PRICE : ₹300
PUBLISHED : DECEMBER 2025

UDHAYAS
SPNC PUBLICATION
2155, Main Road , Ayyapakkam,
Chennai-77

spncpublication@gmail.com/
udhyasbooks@gmail.com
www.udhayas.in