Cloud Computing
Module 1

Subject : Cloud Computing for the Students of IEM

Benefits of Cloud Computing

(1) Ability to get rid of most or all hardware and software

- ✔ No need to have own server, cables, network switches, backup generators, redundant routers, and so on.
- ✔ Cloud provider can manage all of these for a monthly fee.
- ✔ Reducing expenses is essential in any business model and cloud platform can benefit their consumers in this factor.

Benefits of Cloud Computing

(2) Centralized data security

Data backups are centralized in the cloud providers' data centers. Consumers do not need to take any onsite or offsite backup.

Consumers can take advantage of cloud security technologies such as data encryption and two-factor authentication for greater privacy.

Benefits of Cloud Computing

(3) Higher performance and availability

- ✔ Cloud computing increases input/output operations per second (IOPS).
- ✔ Cloud services also offer high availability with no downtime because they're distributed across multiple cloud facilities.
- ✔ Cloud providers are responsible for updating cloud systems and fixing bugs and security issues in cloud software, which is transparent to end users.

Benefits of Cloud Computing

(4) Quick application deployment

- ✔ Unpredictable business needs often require cloud computing resources on short notice.
- ✔ You can improve your cloud application development by quickly deploying cloud applications because they are readily available without the need to procure additional hardware or wait for IT staff to set up servers.

Benefits of Cloud Computing

(5) Instant business insights

Cloud-based platforms provide a unique opportunity to access data as soon as it's collected.

This facilitates better decision-making as well as insight into what the future may hold for your organization based on predictions from historical data.

Benefits of Cloud Computing

(6) Business continuity

In the event of disaster or unforeseen circumstances

Cloud is an effective solution

Benefits of Cloud Computing

(7) Price-performance and cost savings

- ✔ Comparatively lower initial investment is required to implement a cloud strategy.
- ✔ Organizations save substantial amounts in the long run as no need of maintaining expensive hardware or data centers.
- ✔ Since there are no upfront costs to use cloud-based systems, new businesses can test them.

Benefits of Cloud Computing

(8) Virtualized computing

Cloud computing is perfect for virtualized computer environments because cloud resources can be allocated instantly to support significant increases in demand so you never experience downtime again.

With cloud computing, your business can expand its capabilities almost effortlessly to meet growing demands without increasing staff or capital expenditures.

Benefits of Cloud Computing

(9) Cloud computing is greener

Cloud computing is a greener technology than traditional IT solutions. By moving to the cloud, businesses can reduce their energy consumption and carbon footprint by up to 90%.

Rather than having in-house servers and software, businesses can use cloud-based services to access the same applications and data from any computer or device with an internet connection.

This eliminates the need for businesses to purchase and maintain their own IT infrastructure.

Cloud Cube Model

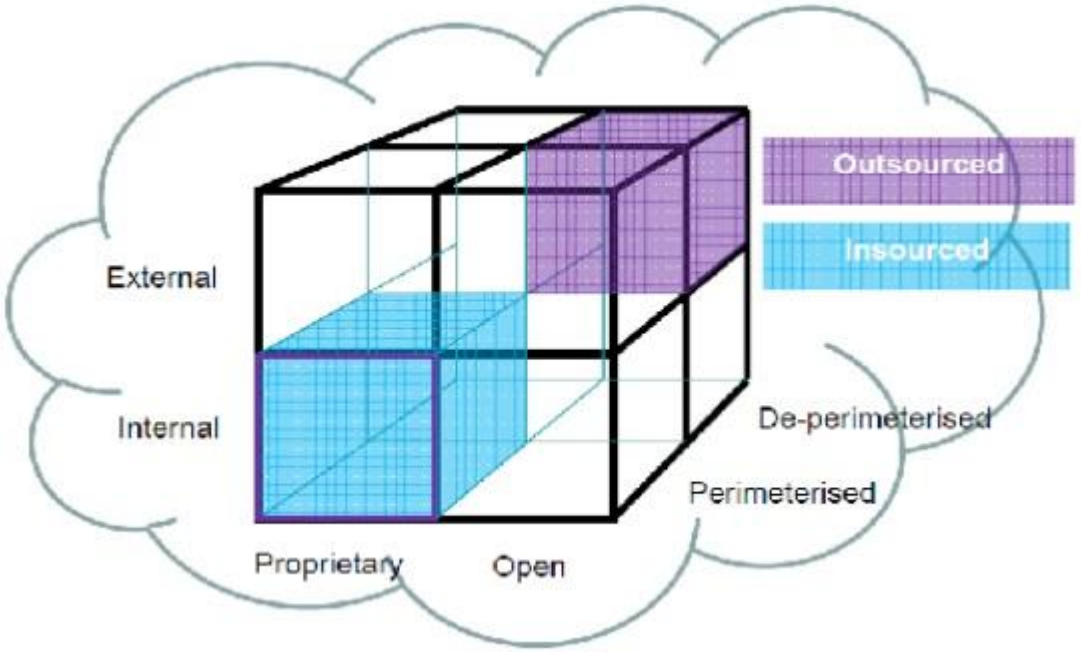
Cloud Cube Model, designed and developed by Jericho forum. Which helps to categorize the cloud network based on the four-dimensional factor: Internal/External, Proprietary/Open, De-Perimeterized/Perimeterized, and Insourced/Outsourced.

The main goal of cloud cube model is to provide the security to the cloud network and protect it.
In cloud computing security plays an important part for different cloud users.

Cloud cube model also enables secure collaboration of cloud formations

It is helpful for different types of organizations and businesses.

Cloud Cube Model



Dimensions of Cloud Cube Model

(1) Internal/External:

The most basic cloud form is the **external and internal cloud form.**
The external or internal dimension defines the physical location of the data. It acknowledges us whether the data exists inside or outside of your organization's boundary.

Here, the data which is stored using a **private cloud** deployment will be considered internal and data outside the cloud will be considered external.

Cloud Cube Model

Dimensions of Cloud Cube Model

(2) Proprietary/Open

The second type of cloud formation is **proprietary/open**. The proprietary or open dimension states about the state of ownership of the **cloud technology** and interfaces. It also tells the degree of interoperability, while enabling data transportability between the system and other cloud forms.

The **proprietary dimension** means, that the organization providing the **service is securing** and protecting the data under their ownership.

The **open dimension** is using a technology in which there are more suppliers. Moreover, the user is not constrained in being able to share the data and collaborate with selected partners using the open technology.

(3) Perimeterized/de-perimeterized:

The Perimeterised and De-perimeterized dimension tells us whether you are operating inside your traditional it mindset or outside it.

Perimeterized dimension means, continuing to operate within the traditional IT boundary.

With the help of VPN and operation of the virtual server in your own IP domain, the user can extend the organizations perimeter into external Cloud Computing domain. This means that the user is making use of the own services to control access.

Cond.

Perimeterized/de-perimeterized:

In De-perimeterized dimension, the data will be encapsulated with metadata and mechanisms, which will further help to protect the data and limit the inappropriate usage.

(4) Insourced/Outsourced

The **Insourced/Outsourced dimensions** have two states.

In the *outsourced dimension* the services provided by the third party.

In the *insourced dimension* the services provided by the own staff under the control.

Cloud Cube Model

How to Secure Data in the Cloud Cube Model?
There are some steps and points to keep in mind before securing your data in a cloud cube model:

Step 1
The **classification of the data**, the customer should know what rules must be applied to protect it.

Step 2
It should be **ensured**, the data exist only in specific trust levels.

Step 3
It should check that what **regulatory compliance and restrictions** are applicable. For example, the data should stay in a particular boundary and whether it has to stay in the safe harbor or not.

How to Secure Data in the Cloud Cube Model?

After the data is classified and is ready to put in the required zone, the assigned person is in a position to decide the following factors:-

The data and processes, which are to be moved in the cloud.
At what level the user wants to operate in the cloud. It can be infrastructure, platform, software, or **platform as a service**.
The cloud formations, which are mostly compatible as per the requirement.
The level of **operation in the cloud** can be different as per the requirement.
Below is the chart which shows the Cloud layers, where the cloud operates.

Cloud Computing and IoT syllabus:
IoT part: Introduction to IoT – IoT definition – Characteristics – IoT Complete Architectural Stack – IoT enabling Technologies, IoT Challenges, M2M and IoT, Smart applications of IoT, Protocols in communication, Key elements of protocol, Different layers of the IoT protocol stack, Different components of IOT. Basics of networking, Protocols and standards in communication, ISO-OSI model.

Cloud computing part: Defining Cloud Computing Paradigm. Cloud Types – NIST model, Cloud Cube model, Deployment models (Public, Private, Hybrid and Community Clouds), Service - Platform as a Service, Software as a Service with examples of services/service providers, models - Infrastructure as a Service, Cloud Reference model, Characteristics of Cloud Computing – a shift in paradigm Benefits and advantages of Cloud Computing. A brief introduction on Composability, Infrastructure, Platforms, Virtual Appliances,

Communication Protocols, Applications, Connecting to the Cloud by Clients.

Networking: Networking refers to the linking of computers and communication network devices (also referred to as **hosts**), which interconnect through a network (Internet or Intranet) and are separated by unique device identifiers (Internet protocol, IP addresses and media access control, MAC addresses). These hosts may be connected by a single path or through multiple paths for sending and receiving data. The data transferred between the hosts may be text, images, or videos, which are typically in the form of binary bit streams.

Network Types: i) Connection types: Point-to-point: Point-to-point connections are used to establish direct connections between two hosts. Day-to-day systems such as a remote controlfor an air conditioner or television is a point to point connection, where the connection has the whole channel dedicated to it only. These networks were designed to work over duplex links and are functional for both synchronous as well as asynchronous systems. Regarding computer networks, point to point connections find usage for specific purposes such as in optical networks.

Point-to-multipoint: In a point-to-multipoint connection, more than two hosts share the same link. This type of configuration is similar to the one-to-many connection type. Point-to-multipoint connections find popular use in wireless networks and IP telephony. The channel is shared between the various hosts, either spatially or temporally. One common scheme of spatial sharing of the channel is frequency division multiple access (FDMA). Temporal sharing of channels include approaches such as time division multiple access (TDMA).

ii) Physical topology: physical manner in which communication paths between the hosts are connected.

1. star: In a star topology, every host has a point-to-point link to a central controller or hub. The hosts cannot communicate with one another directly; they can only do so through the central hub. The hub acts as the network traffic exchange. For large-scale systems, the hub, essentially, has to be a powerful server to handle all the simultaneous traffic flowing through it. However, as there are fewer links (only one link per host), this topology is cheaper and easier to set up. The main advantages of the star topology are easy installation and the ease of fault identification within the network. If the central hub remains uncompromised, link failures between a host and the hub do not have a big effect on the network, except for the host that is affected. However, the main disadvantage of this topology is the danger of a single point of failure. If the hub fails, the whole network fails.
2. Mesh: In a mesh topology, every host is connected to every other host using a dedicated link (in a point-to-point manner). This implies that for n hosts in a mesh, there are a total of **n(n − 1)/2 dedicated** full duplex links between the hosts. This massive number of links makes the mesh topology expensive. However, it offers certain specific advantages over other topologies. The first significant advantage is the **robustness and resilience** of the system. Even if a link is down or broken, the network is still fully functional as there remain other pathways for the traffic to flow through. The second advantage is the **security and privacy of the traffic** as the data is only seen by the intended recipients and not by all members of the network. The third advantage is the **reduced data load on a single host,** as every host in this network takes care of its traffic load. However, owing to the complexities in forming physical connections between devices and the cost of establishing these links, mesh networks are used very selectively, such as in backbone networks.

3. Bus: A bus topology follows the point-to-multipoint connection. A **backbone cable or bus** serves as the primary traffic pathway between the hosts. The hosts are connected to the main bus employing **drop lines or taps.** The main advantage of this topology is the ease of installation. However, there is a restriction on the length of the bus and the number of hosts that can be simultaneously connected to the bus due to signal loss over the extended bus. The bus topology has a simple cabling procedure in which a single bus (backbone cable) can be used for an organization. Multiple drop lines and taps can be used to connect various hosts to the bus, making installation very easy and cheap. However, the main drawback of this topology is the **difficulty in fault localization** within the network.
4. **Ring:** A ring topology works on the principle of a point-to-point connection. Here, each host is configured to have a dedicated point-to-point connection with its two immediate neighboring hosts on either side of it through repeaters at each host. The repetition of this system forms a ring. The **repeaters** at each host capture the incoming signal intended for other hosts, regenerates the bit stream, and passes it onto the next repeater. Fault identification and set up of the ring topology is quite simple and straightforward. However, the main disadvantage of this system is the high probability of a single point of failure. If even one repeater fails, the whole network goes down.

iii) Network reachability: Computer networks are divided into four broad categories based on network reachability: personal area networks, local area networks, wide area networks, and metropolitan area networks.

(i) Personal Area Networks (PAN): PANs, as the name suggests, are mostly restricted to individual usage. A good example of PANs may be connected wireless headphones, wireless speakers, laptops, smartphones, wireless keyboards, wireless mouse, and printers within a house. Generally, PANs are wireless networks, which make use of low-range and low-power technologies such as **Bluetooth**. The reachability of PANs lies in the range of a few centimeters to a few meters.

(ii) Local Area Networks (LAN): A LAN is a collection of hosts linked to a single network through wired or wireless connections. However, LANs are restricted to buildings, organizations, or campuses. Typically, a few leased lines connected to the Internet provide web access to the whole organization or a campus; the lines are further redistributed to multiple hosts within the LAN enabling hosts. The hosts are much more in number than the actual direct lines to the Internet to access the web from within the organization. This also allows the organization to define various access control policies for web access within its hierarchy. Typically, the present-day data access rates within the LANs range from **100 Mbps to 1000 Mbps**, with very high fault-tolerance levels. Commonly used network components in a LAN are **servers, hubs, routers**, switches, terminals, and computers.

(iii) Metropolitan Area Networks (MAN): The reachability of a MAN lies between that of a LAN and a WAN. Typically, MANs connect various organizations or buildings within a given geographic location or city. An excellent example of a MAN is an Internet service provider (ISP) supplying Internet connectivity to various organizations within a city. As MANs are costly, they may not be owned by individuals or even single organizations. Typical networking devices/ components in MANs are modems and cables. MANs tend to have **moderate fault tolerance levels**.

(iv) Wide Area Networks (WAN): WANs typically connect diverse geographic locations. However, they are **restricted within the boundaries of a state or country**. The data rate of WANs is in the order of a fraction of LAN's data rate. Typically, WANs connecting two LANs or MANs may use public switched telephone networks **(PSTNs)** or **satellite-based links**. Due to the long transmission ranges, WANs tend to have more errors and noise during transmission and are very costly to maintain. The **fault tolerance of WANs are generally low**.

Networking: Networking refers to the linking of computers and communication network devices (also referred to as **hosts**), which interconnect through a network (Internet or Intranet) and are separated by unique device identifiers (Internet protocol, IP addresses and media access control, MAC addresses). These hosts may be connected by a single path or through multiple paths for sending and receiving data. The data transferred between the hosts may be text, images, or videos, which are typically in the form of binary bit streams.

Network Types: i) Connection types: Point-to-point: Point-to-point connections are used to establish direct connections between two hosts. Day-to-day systems such as a remote controlfor an air conditioner or television is a point to point connection, where the connection has the whole channel dedicated to it only. These networks were designed to work over duplex links and are functional for both synchronous as well as asynchronous systems. Regarding computer networks, point to point connections find usage for specific purposes such as in optical networks.

Point-to-multipoint: In a point-to-multipoint connection, more than two hosts share the same link. This type of configuration is similar to the one-to-many connection type. Point-to-multipoint connections find popular use in wireless networks and IP telephony. The channel is shared between the various hosts, either spatially or temporally. One common scheme of spatial sharing of the channel is frequency division multiple access (FDMA). Temporal sharing of channels include approaches such as time division multiple access (TDMA).

ii) Physical topology: physical manner in which communication paths between the hosts are connected.

1. star: In a star topology, every host has a point-to-point link to a central controller or hub. The hosts cannot communicate with one another directly; they can only do so through the central hub. The hub acts as the network traffic exchange. For large-scale systems, the hub, essentially, has to be a powerful server to handle all the simultaneous traffic flowing through it. However, as there are fewer links (only one link per host), this topology is cheaper and easier to set up. The main advantages of the star topology are easy installation and the ease of fault identification within the network. If the central hub remains uncompromised, link failures between a host and the hub do not have a big effect on the network, except for the host that is affected. However, the main disadvantage of this topology is the danger of a single point of failure. If the hub fails, the whole network fails.
2. Mesh: In a mesh topology, every host is connected to every other host using a dedicated link (in a point-to-point manner). This implies that for n hosts in a mesh, there are a total of **n(n − 1)/2 dedicated** full duplex links between the hosts. This massive number of links makes the mesh topology expensive. However, it offers certain specific advantages over other topologies. The first significant advantage is the **robustness and resilience** of the system. Even if a link is down or broken, the network is still fully functional as there remain other pathways for the traffic to flow through. The second advantage is the **security and privacy of the traffic** as the data is only seen by the intended recipients and not by all members of the network. The third advantage is the **reduced data load on a single host,** as every host in this network takes care of its traffic load. However, owing to the complexities in forming physical connections between devices and the cost of establishing these links, mesh networks are used very selectively, such as in backbone networks.

3. Bus: A bus topology follows the point-to-multipoint connection. A **backbone cable or bus** serves as the primary traffic pathway between the hosts. The hosts are connected to the main bus employing **drop lines or taps.** The main advantage of this topology is the ease of installation. However, there is a restriction on the length of the bus and the number of hosts that can be simultaneously connected to the bus due to signal loss over the extended bus. The bus topology has a simple cabling procedure in which a single bus (backbone cable) can be used for an organization. Multiple drop lines and taps can be used to connect various hosts to the bus, making installation very easy and cheap. However, the main drawback of this topology is the **difficulty in fault localization** within the network.
4. **Ring:** A ring topology works on the principle of a point-to-point connection. Here, each host is configured to have a dedicated point-to-point connection with its two immediate neighboring hosts on either side of it through repeaters at each host. The repetition of this system forms a ring. The **repeaters** at each host capture the incoming signal intended for other hosts, regenerates the bit stream, and passes it onto the next repeater. Fault identification and set up of the ring topology is quite simple and straightforward. However, the main disadvantage of this system is the high probability of a single point of failure. If even one repeater fails, the whole network goes down.

iii) Network reachability: Computer networks are divided into four broad categories based on network reachability: personal area networks, local area networks, wide area networks, and metropolitan area networks.

(i) Personal Area Networks (PAN): PANs, as the name suggests, are mostly restricted to individual usage. A good example of PANs may be connected wireless headphones, wireless speakers, laptops, smartphones, wireless keyboards, wireless mouse, and printers within a house. Generally, PANs are wireless networks, which make use of low-range and low-power technologies such as **Bluetooth**. The reachability of PANs lies in the range of a few centimeters to a few meters.

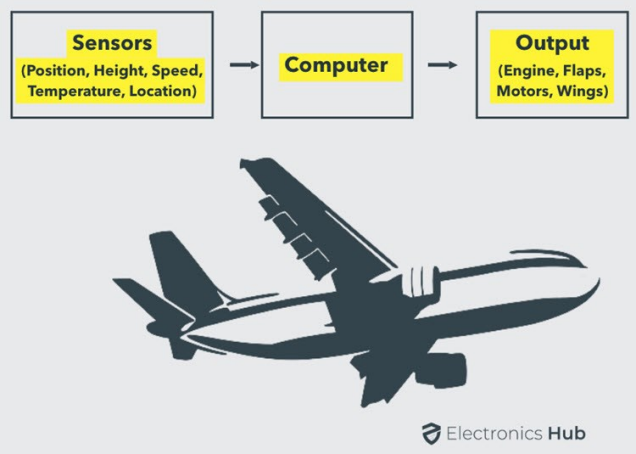
(ii) Local Area Networks (LAN): A LAN is a collection of hosts linked to a single network through wired or wireless connections. However, LANs are restricted to buildings, organizations, or campuses. Typically, a few leased lines connected to the Internet provide web access to the whole organization or a campus; the lines are further redistributed to multiple hosts within the LAN enabling hosts. The hosts are much more in number than the actual direct lines to the Internet to access the web from within the organization. This also allows the organization to define various access control policies for web access within its hierarchy. Typically, the present-day data access rates within the LANs range from **100 Mbps to 1000 Mbps**, with very high fault-tolerance levels. Commonly used network components in a LAN are **servers, hubs, routers**, switches, terminals, and computers.

(iii) Metropolitan Area Networks (MAN): The reachability of a MAN lies between that of a LAN and a WAN. Typically, MANs connect various organizations or buildings within a given geographic location or city. An excellent example of a MAN is an Internet service provider (ISP) supplying Internet connectivity to various organizations within a city. As MANs are costly, they may not be owned by individuals or even single organizations. Typical networking devices/ components in MANs are modems and cables. MANs tend to have **moderate fault tolerance levels**.

(iv) Wide Area Networks (WAN): WANs typically connect diverse geographic locations. However, they are **restricted within the boundaries of a state or country**. The data rate of WANs is in the order of a fraction of LAN's data rate. Typically, WANs connecting two LANs or MANs may use public switched telephone networks **(PSTNs)** or **satellite-based links**. Due to the long transmission ranges, WANs tend to have more errors and noise during transmission and are very costly to maintain. The **fault tolerance of WANs are generally low**.

Real Time Application of Sensors

The example we are talking about here is the Autopilot System in aircrafts. Almost all civilian and military aircrafts have the feature of Automatic Flight Control system or sometimes called as Autopilot.



An Automatic Flight Control System consists of several sensors for various tasks like speed control, height monitoring, position tracking, status of doors, obstacle detection, fuel level, maneuvering and many more. A Computer takes data from all these sensors and processes them by comparing them with pre-designed values.

The computer then provides control signals to different parts like engines, flaps, rudders, motors etc. that help in a smooth flight. The combination of Sensors, Computers and Mechanics makes it possible to run the plane in Autopilot Mode.

All the parameters i.e., the Sensors (which give inputs to the Computers), the Computers (the brains of the system) and the mechanics (the outputs of the system like engines and motors) are equally important in building a successful automated system.

This is an extremely simplified version of Flight Control System. In fact, there are hundreds of individual control systems which preform unique tasks for a safe and smooth journey.

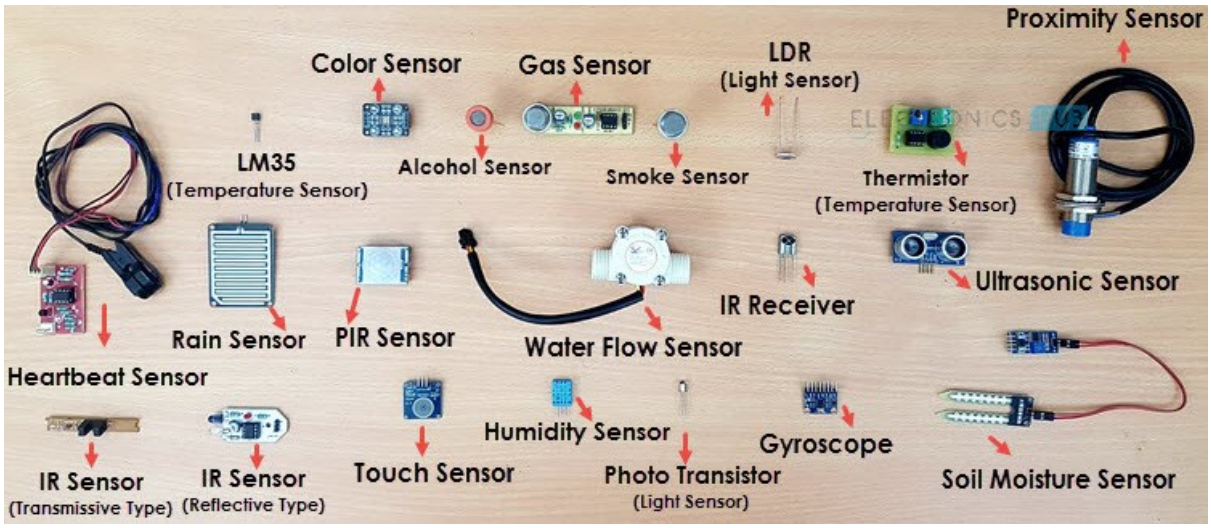
But in this tutorial, we will be concentrating on the Sensors part of a system and look at different concepts associated with Sensors (like types, characteristics, classification etc.).

What is a Sensor?

There are numerous definitions as to what a sensor is but I would like to define a Sensor as an input device which provides an output (signal) with respect to a specific physical quantity (input).

The term “input device” in the definition of a Sensor means that it is part of a bigger system which provides input to a main control system (like a Processor or a Microcontroller).

Another unique definition of a Sensor is as follows: It is a device that converts signals from one energy domain to electrical domain. The definition of the Sensor can be better understood if we take an example in to consideration.



The simplest example of a sensor is an LDR or a Light Dependent Resistor. It is a device, whose resistance varies according to intensity of light it is subjected to. When the light falling on an LDR is more, its resistance becomes very less and when the light is less, well, the resistance of the LDR becomes very high.

We can connect this LDR in a voltage divider (along with other resistor) and check the voltage drop across the LDR. This voltage can be calibrated to the amount of light falling on the LDR. Hence, a Light Sensor.

Now that we have seen what a sensor is, we will proceed further with the classification of Sensors.

Classification of Sensors

There are several classifications of sensors made by different authors and experts. Some are very simple and some are very complex. The following classification of sensors may already be used by an expert in the subject but this is a very simple classification of sensors.

In the first classification of the sensors, they are divided in to Active and Passive. Active Sensors are those which require an external excitation signal or a power signal.

Passive Sensors, on the other hand, do not require any external power signal and directly generates output response.

The other type of classification is based on the means of detection used in the sensor. Some of the means of detection are Electric, Biological, Chemical, Radioactive etc.

The next classification is based on conversion phenomenon i.e., the input and the output. Some of the common conversion phenomena are Photoelectric, Thermoelectric, Electrochemical, Electromagnetic, Thermooptic, etc.

The final classification of the sensors are Analog and Digital Sensors. Analog Sensors produce an analog output i.e., a continuous output signal (usually voltage but sometimes other quantities like Resistance etc.) with respect to the quantity being measured.

Digital Sensors, in contrast to Analog Sensors, work with discrete or digital data. The data in digital sensors, which is used for conversion and transmission, is digital in nature.

Different Types of Sensors

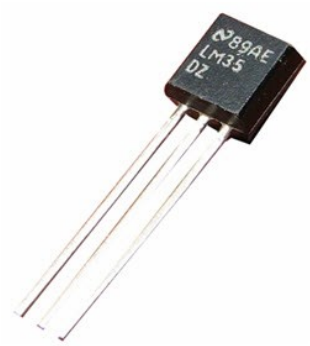
The following is a list of different types of sensors that are commonly used in various applications. All these sensors are used for measuring one of the physical properties like Temperature, Resistance, Capacitance, Conduction, Heat Transfer etc.

1. Temperature Sensor
2. Proximity Sensor
3. Accelerometer
4. IR Sensor (Infrared Sensor)
5. Pressure Sensor
6. Light Sensor
7. Ultrasonic Sensor
8. Smoke, Gas and Alcohol Sensor
9. Touch Sensor
10. Color Sensor
11. Humidity Sensor
12. Position Sensor
13. Magnetic Sensor (Hall Effect Sensor)
14. Microphone (Sound Sensor)
15. Tilt Sensor
16. Flow and Level Sensor
17. PIR Sensor
18. Touch Sensor
19. Strain and Weight Sensor

We will see about few of the above-mentioned sensors in brief. More information about the sensors will be added subsequently. A list of projects using the above sensors is given at the end of the page.

Temperature Sensor

One of the most common and most popular sensors is the Temperature Sensor. A Temperature Sensor, as the name suggests, senses the temperature i.e., it measures the changes in the temperature.



LM35 - Temperature Sensor IC

10KΩ NTC Thermistor

There are different types of Temperature Sensors like Temperature Sensor ICs (like LM35, DS18B20), Thermistors, Thermocouples, RTD (Resistive Temperature Devices), etc.

Temperature Sensors can be analog or digital. In an Analog Temperature Sensor, the changes in the Temperature correspond to change in its physical property like resistance or voltage. LM35 is a classic Analog Temperature Sensor.

Coming to the Digital Temperature Sensor, the output is a discrete digital value (usually, some numerical data after converting analog value to digital value). DS18B20 is a simple Digital Temperature Sensor.

Temperature Sensors are used everywhere like computers, mobile phones, automobiles, air conditioning systems, industries etc.

Proximity Sensors

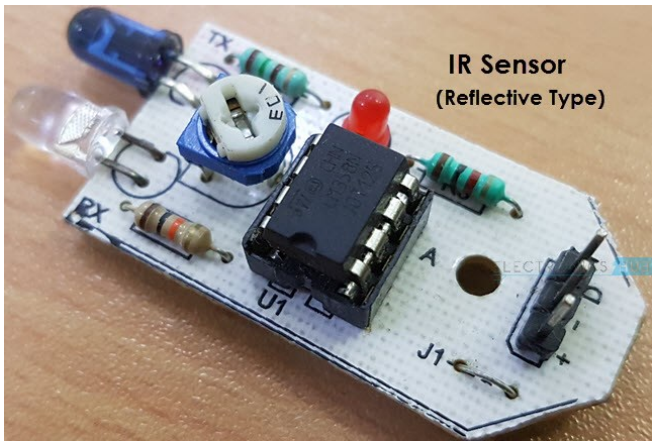
A Proximity Sensor is a non-contact type sensor that detects the presence of an object. Proximity Sensors can be implemented using different

techniques like Optical (like Infrared or Laser), Sound (Ultrasonic), Magnetic (Hall Effect), Capacitive, etc.

Some of the applications of Proximity Sensors are Mobile Phones, Cars (Parking Sensors), industries (object alignment), Ground Proximity in Aircrafts, etc.

Infrared Sensor (IR Sensor)

IR Sensors or Infrared Sensor are light based sensor that are used in various applications like Proximity and Object Detection. IR Sensors are used as proximity sensors in almost all mobile phones.



There are two types of Infrared or IR Sensors: Transmissive Type and Reflective Type. In Transmissive Type IR Sensor, the IR Transmitter (usually an IR LED) and the IR Detector (usually a Photo Diode) are positioned

facing each other so that when an object passes between them, the sensor detects the object.

The other type of IR Sensor is a Reflective Type IR Sensor. In this, the transmitter and the detector are positioned adjacent to each other facing the object. When an object comes in front of the sensor, the infrared light from the IR Transmitter is reflected from the object and is detected by the IR Receiver and thus the sensor detects the object.

Different applications where IR Sensor is implemented are Mobile Phones, Robots, Industrial assembly, automobiles etc.

Ultrasonic Sensor

An Ultrasonic Sensor is a non-contact type device that can be used to measure distance as well as velocity of an object. An Ultrasonic Sensor works based on the properties of the sound waves with frequency greater than that of the human audible range.

Ultrasonic Sensor

Using the time of flight of the sound wave, an Ultrasonic Sensor can measure the distance of the object (similar to SONAR). The Doppler Shift property of the sound wave is used to measure the velocity of an object.

Light Sensor

Sometimes also known as Photo Sensors, Light Sensors are one of the important sensors. A simple Light Sensor available today is the Light

Dependent Resistor or LDR. The property of LDR is that its resistance is inversely proportional to the intensity of the ambient light i.e., when the intensity of light increases, its resistance decreases and vise-versa.

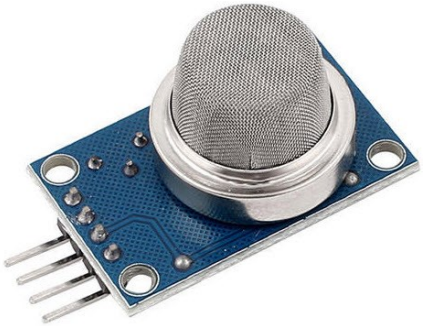
By using LDR is a circuit, we can calibrate the changes in its resistance to measure the intensity of Light. There are two other Light Sensors (or Photo Sensors) which are often used in complex electronic system design. They are Photo Diode and Photo Transistor. All these are Analog Sensors.

There are also Digital Light Sensors like BH1750, TSL2561, etc., which can calculate intensity of light and provide a digital equivalent value.

Smoke and Gas Sensors

One of the very useful sensors in safety related applications are Smoke and Gas Sensors. Almost all offices and industries are equipped with several smoke detectors, which detect any smoke (due to fire) and sound an alarm.

Gas Sensors are more common in laboratories, large scale kitchens and industries. They can detect different gases like LPG, Propane, Butane, Methane (CH4), etc.



Now-a-days, smoke sensors (which often can detect smoke as well gas) are also installed in most homes as a safety measure.

The “MQ” series of sensors are a bunch of cheap sensors for detecting CO, CO2, CH4, Alcohol, Propane, Butane, LPG etc. You can use these sensors to build your own Smoke Sensor Application.

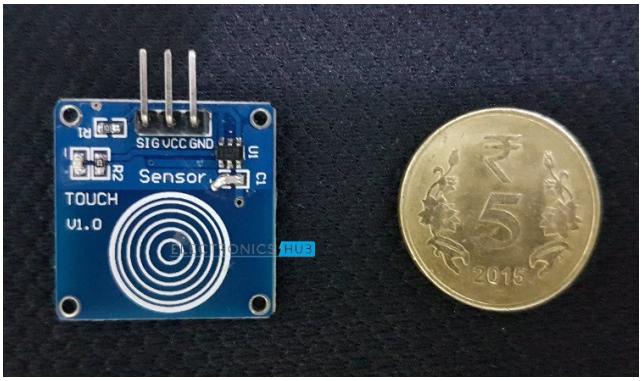
Alcohol Sensor

As the name suggests, an Alcohol Sensor detects alcohol. Usually, alcohol sensors are used in breathalyzer devices, which determine whether a person is drunk or not. Law enforcement personnel uses breathalyzers to catch drunk-and-drive culprits.

Touch Sensor

We do not give much importance to touch sensors but they became an integral part of our life. Whether you know or not, all touch screen devices (Mobile Phones, Tablets, Laptops, etc.) have touch sensors in them. Another common application of touch sensor is trackpads in our laptops.

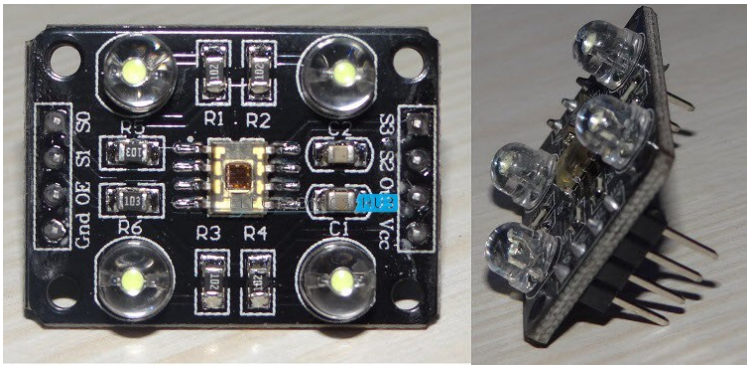
Touch Sensors, as the name suggests, detect touch of a finger or a stylus. Often touch sensors are classified into Resistive and Capacitive type. Almost all modern touch sensors are of Capacitive Types as they are more accurate and have better signal to noise ratio.



If you want to build an application with Touch Sensor, then there are low-cost modules available and using those touch sensors, you can build TOUCH DIMMER SWITCH CIRCUIT USING ARDUINO.

Color Sensor

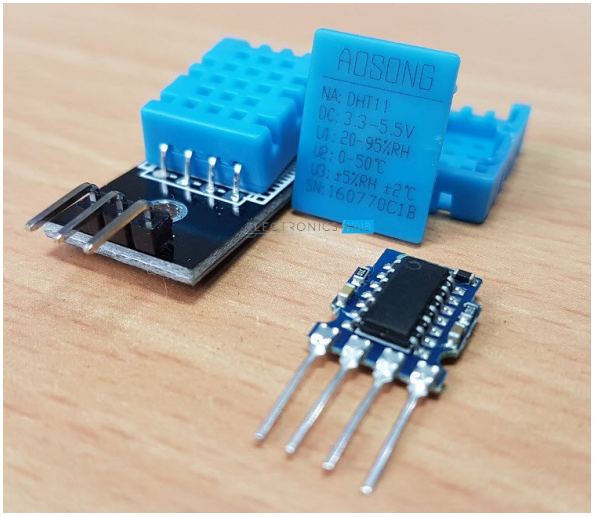
A Color Sensor is an useful device in building color sensing applications in the field of image processing, color identification, industrial object tracking etc. The TCS3200 is a simple Color Sensor, which can detect any color and output a square wave proportional to the wavelength of the detected color.



Humidity Sensor

If you see Weather Monitoring Systems, they often provide temperature as well as humidity data. So, measuring humidity is an important task in many applications and Humidity Sensors help us in achieving this.

Often all humidity sensors measure relative humidity (a ratio of water content in air to maximum potential of air to hold water). Since relative humidity is dependent on temperature of air, almost all Humidity Sensors can also measure Temperature.



Humidity Sensors are classified into Capacitive Type, Resistive Type and Thermal Conductive Type. DHT11 and DHT22 are two of the frequently used Humidity Sensors in DIY Community (the former is a resistive type while the latter is capacitive type).

Cloud Computing
Module 1

Subject : Cloud Computing for the Students of IEM

Challenges of using a cloud service provider

While there are many advantages to working with a cloud provider, there are some considerations to keep in mind.

The key things for organizations to consider:

Complex contracts: You will need to negotiate contracts and SLAs with each cloud provider you use. Multiple providers and vendors can lead to complex SLA relationships with different parameters and guaranteed service expectations.

Vendor lock-in: Some cloud providers do not integrate well with competitor products and services. Becoming too dependent on a single provider can make it difficult to migrate data and workloads to another technology stack without incurring high costs, incompatibilities, or even legal constraints.

Challenges of using a cloud service provider

Security responsibility: Generally a shared responsibility model is followed.
It means cloud security is implemented by both the cloud provider and the customer.
Poor understanding of a provider's responsibilities and consumer's own can lead to substantial security risks or security breaches.

Along with these specific concerns when working with cloud service providers, your organization will also need to have a solid cloud migration strategy in place and be ready to tackle new complexity when it comes to monitoring and managing cloud environments.

Composability

- ✔ Composability is the art of building systems in a modular and flexible way. It emphasizes creating components that are not only reusable but can seamlessly fit together, forming a cohesive and adaptable architecture.

- ✔ Composability is a systems design approach made to increase agility and accelerate application development by reusing existing assets and reassembling them in unique ways to satisfy specific user requirements.

Composability

Composable systems break down larger projects into smaller, more modular “components.”
Each component addresses a specific problem or use case.

Individual components can then be selected and assembled in various combinations to create new experiences without building from scratch.

In the context of application development, modular composability refers to the rapid assembly of new digital experiences by drawing from a library of pre-existing components that can be arranged—or “composed”—to address a specific use case.

Composability

The key elements are:
Modularity
Autonomy
Discoverability

Composability

Modularity

To put it simply, modularity is about doing one thing well. It involves bundling a specific set of services into a single component that is dedicated to achieving a specific purpose.

Autonomy

To qualify as a truly composable system, the individual components must be autonomous—meaning, they are entirely self-contained, and are not dependent on other parts of the system. Also, it should be possible to update one part of the system without affecting any other parts of the system.

Composability

Discoverability

An important element in a composable system approach is that individual components can be reused over and over again to assemble new experiences.

For that to occur, the individual components must be easily discoverable by other teams.

Building a reusable component that lacks discoverability negates the value of composable systems design.

Virtual Appliance (VA)

- ✔ A software equivalent of a hardware device.
- ✔ It contains an operating system (OS) and a customized application to perform a fixed set of functions.
- ✔ When a software appliance is installed on a virtual machine (VM), it creates a virtual appliance, which is nothing but a VM image file.

Virtual Appliance (VA)

VAs usually come in the Open Virtualization Format (OVF).

It is vendor-independent, the appliance can be easily packaged and distributed in a single-file format.

OVF is also beneficial for customers because it allows them to deploy, manage and update complex solutions easily.

Virtual Appliance (VA)

Virtual appliances play a major role in SaaS service model.
The delivery of software remotely through a user's web browser.

VAs are also useful for quickly provisioning OSes and applications in the platform as a service (PaaS) model.

Virtual Appliance (VA)

Two types of virtual appliances:

Closed virtual appliance, which is always packaged, distributed, maintained, updated and managed as a unit

Open virtual appliance, which allows to customers to make modifications.

Virtual Appliance (VA)

How a virtual appliance is deployed

- ✔ A VA can be deployed as a VM or a subset of a VM running atop virtualization technology.
- ✔ It is possible to package, maintain and manage multiple VMs as a single unit.
- ✔ Deploying an application as a VA can eliminate problems with installation and configuration, such as software or driver compatibility issues.
- ✔ Users can simply download a single file and run the application.
- ✔ Resources required for maintenance are also reduced.

Virtual Appliance (VA)

Virtual appliance Use Cases

- ✔ Virtual Appliance have proved useful for deploying network applications.
- ✔ It is helpful in grid computing too.
- ✔ In SaaS delivery model.

Virtual Appliance (VA)

Benefits of virtual appliances

Reduced costs for developers, vendors and customers. For developers and appliance vendors, virtual appliances help lower development and distribution costs.

It is achieved by reducing the need for hardware testing and decreasing the number of platforms that need to be supported.

VAs are a cheaper alternative to hardware appliances because they don't have to manage inventory or support hardware components.

It can be distribute online.

VA reduces the cost of owning, operating and managing the software.

Virtual Appliance (VA)

Benefits of virtual appliances

Easier IT management

- ✔ In VA, users have to manage a single solution instead of multiple applications, OS and server hardware.
- ✔ Moreover, they can get support from a single vendor for all components in the VA.
- ✔ So, simplifies IT management, administration and maintenance.

Virtual Appliance (VA)

Benefits of virtual appliances

Accelerated time-to-market and time-to-value

- ✔ A VA reduces the time required for product evaluation, configuration, packaging and deployment, accelerating time-to-value for customers.
- ✔ It also shortens the sales cycles for vendors, accelerating their time-to-market.
- ✔ Further, vendors can expand customer reach by targeting potential customers they would not be able to target with hardware appliances.

Virtual Appliance (VA)

Benefits of virtual appliances

Enhanced security with isolation.

- ✔ Virtual appliances run in an isolated environment, with different appliances shielded from each other.
- ✔ With such an arrangement, if the security of any VA is compromised, other VAs will not be affected and can continue functioning.

Communication Protocols in Cloud Computing

A **Gossip protocol** or epidemic protocol is a procedure or process of computer peer-to-peer communication that is based on the way epidemics spread. Some distributed systems use peer-to-peer gossip to ensure that data is disseminated to all members of a group.

MTP (Media Transfer Protocol) - The Media Transfer Protocol, introduced by Microsoft, is a protocol designed for intelligent storage devices like phones and digital audio players.

It is based on, and fully compatible with, the Picture Transfer Protocol (PTP). MTP allows the synchronization of files between portable devices and a personal computer (PC).

Communication Protocols in Cloud Computing

- ✔ CEE (Coverage Enhanced Ethernet Protocol) - Converged enhanced Ethernet is a term used to refer to the IEEE 802.1 standard version
- ✔ Next generation Ethernet
- ✔ A standardized packet lossless technology in input/output consolidation fiber channel over Ethernet networks.

Communication Protocols in Cloud Computing

SRP (State Routing Protocol) - In link-state routing protocols, each router possesses information about the complete network topology.

Each router then independently calculates the best next hop from it for every possible destination in the network using local information of the topology. The collection of best-next-hops forms the routing table.

This contrasts with distance-vector routing protocols, which work by having each node share its routing table with its neighbours.

In a link-state protocol, the only information passed between the nodes is the information used to construct the connectivity maps.

Examples of link-state routing protocols:

- Open Shortest Path First (OSPF)
- Intermediate System to Intermediate System (IS-IS)

Communication Protocols in Cloud Computing

SSHP (Secure Shell Protocol) - SSH transport layer protocol
The Secure Shell (SSH) is a protocol for secure remote login and other secure network services over an insecure network.

SSH transport layer protocol, which typically runs on top of TCP/IP.

IGMP (Internet Group Management Protocol) - IGMP is a communication protocol used by hosts and adjacent routers for multicasting communication with IP networks and uses the resources efficiently to transmit the message/data packets.

Multicast communication can have single or multiple senders and receivers and thus, IGMP can be used in streaming videos, gaming or web conferencing tools.

Connecting to the Cloud by Clients

1. Direct Connect:

- ✔ Direct Connect services enable dedicated network connections with cloud service providers like AWS and Azure.
- ✔ Compared to VPNs, direct connections offer significantly higher bandwidth.
- ✔ Setting up direct connect requires router installations at each colocation, and many service providers support international locations, making it a viable option for enterprises operating globally.

Connecting to the Cloud by Clients

2. VPN (Virtual Private Network):

- ✔ VPN is a simple and affordable way to connect with cloud applications.
- ✔ Cloud service providers offer native VPN or VPN appliances through their network service control panels.
- ✔ VPN supports various devices, including OS-powered VPN services and solutions, VPN concentrators, and more.
- ✔ Notably, VPN often doesn't have minimum commitment requirements for data transfers, making it cost-effective.

Connecting to the Cloud by Clients

3. VPN Gateway or Software VPN:

- ✔ Software VPN gateways provide another popular method to connect with cloud applications.
- ✔ These gateways typically offer resizable cloud computing capacity for easy scalability.
- ✔ Additionally, you can leverage marketplace or software solutions on these connections.
- ✔ VPN gateways ensure consistent connections to co-locations, establishing a backbone for global operations and connecting to virtual private clouds in multiple locations.

Connecting to the Cloud by Clients

4. MPLS VPN (Multiprotocol Label Switching)

- ✔ MPLS VPN utilizes a mechanism that directs and carries data between network nodes.
- ✔ It enables enterprises to build virtual links across vast distances and supports multiple network protocols.
- ✔ MPLS VPN is protocol-independent, highly scalable, and eliminates the need for specific data link layer technologies like frame relay, ATM, Ethernet, or SONET.
- ✔ It is particularly suitable for organizations utilizing various network protocols.

Connecting to the Cloud by Clients

5. Telco Managed Services

Telecommunication companies now offer managed services based on direct connect.

These services bundle network connectivity to your cloud provider's network.

The advantage of telco-managed services is the flexibility to choose connectivity options, such as MPLS VPN and Internet VPN, for your preferred cloud service provider.

Cloud Computing

Subject : Cloud Computing for the Students of IEM

Defining Cloud Computing Paradigm

Over the years different computing paradigms have been developed and used. In fact different computing paradigms have existed before the cloud computing paradigm. Let us take a look at all the computing paradigms below.

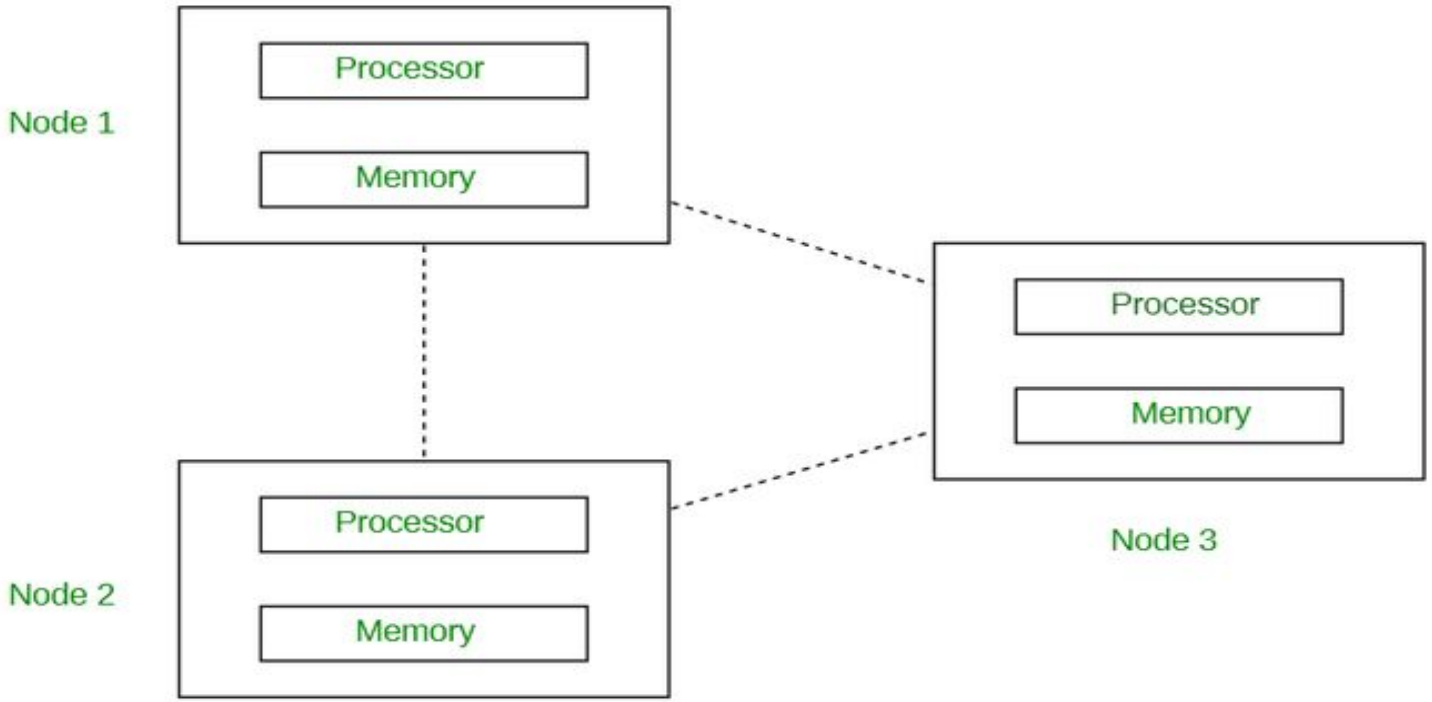
Distributed Computing :

Distributed computing is defined as a type of computing where multiple computer systems work on a single problem.
Here all the computer systems are linked together and the problem is divided into sub-problems where each part is solved by different computer systems.

The goal of distributed computing is to increase the performance and efficiency of the system and ensure fault tolerance.

In the below diagram, each processor has its own local memory and all the processors communicate with each other over a network.

Defining Cloud Computing Paradigm



Parallel Computing :

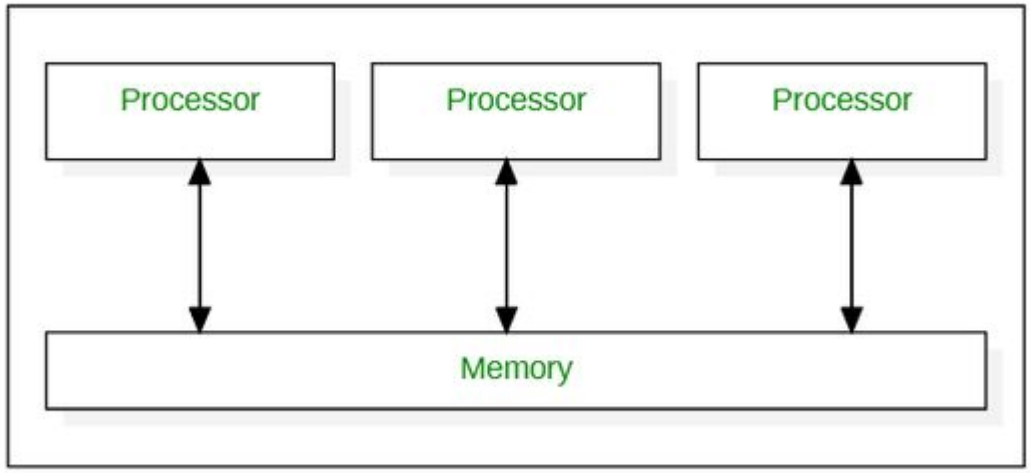
Parallel computing is defined as a type of computing where multiple computer systems are used simultaneously.

Here a problem is broken into sub-problems and then further broken down into instructions.

These instructions from each sub-problem are executed concurrently on different processors.

Defining Cloud Computing Paradigm

Parallel Computing :



Cluster Computing :

A cluster is a group of independent computers that work together to perform the tasks given.

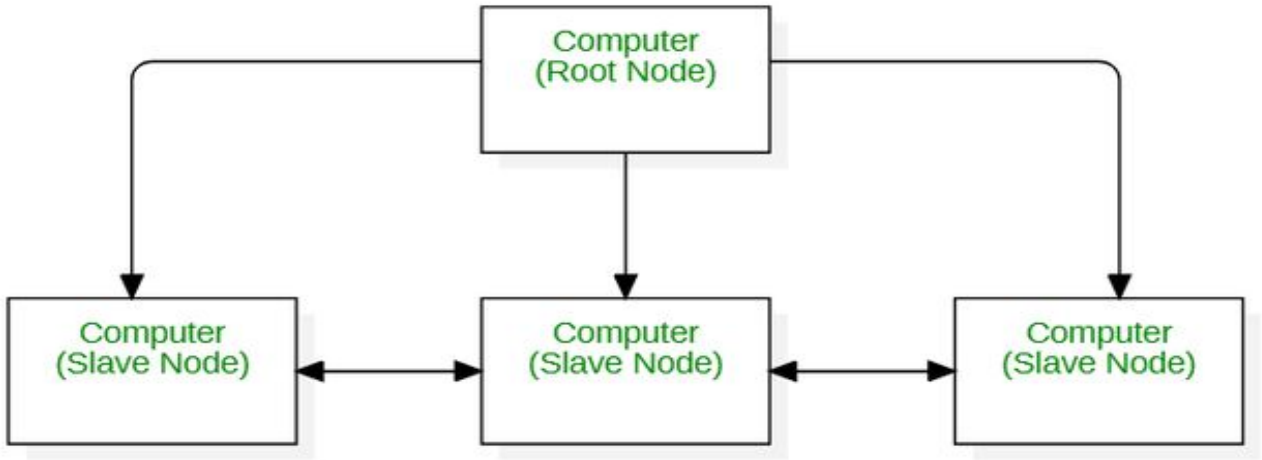
Cluster computing is defined as a type of computing that consists of two or more independent computers, referred to as nodes, that work together to execute tasks as a single machine.

The goal of cluster computing is to increase the performance, scalability and simplicity of the system.

As you can see in the below diagram, all the nodes, (irrespective of whether they are a parent node or child node), act as a single entity to perform the tasks.

Defining Cloud Computing Paradigm

Cluster Computing :



Cloud Computing :

Cloud is defined as the usage of someone else's server to host, process or store data.

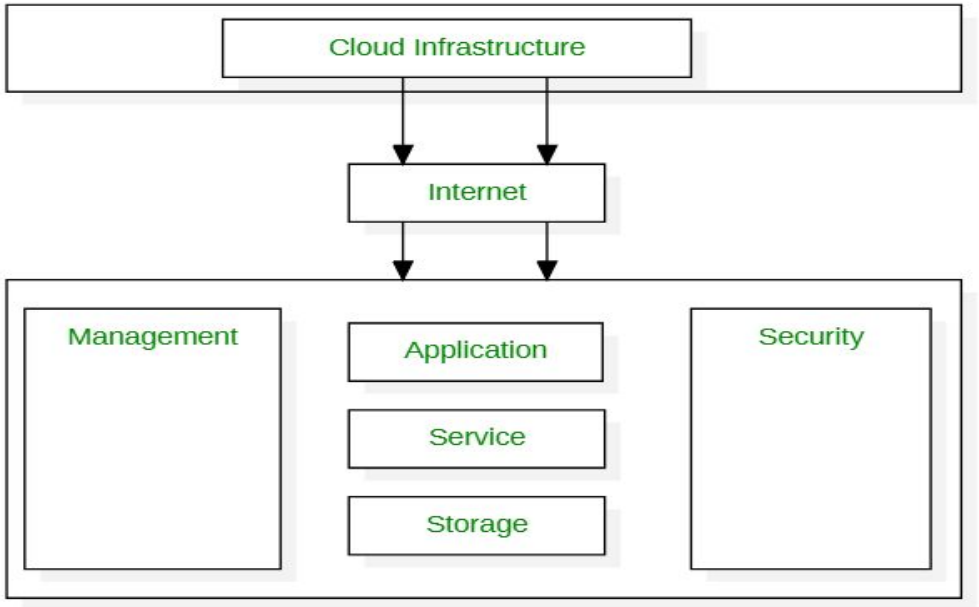
Cloud computing is defined as the type of computing where it is the delivery of on-demand computing services over the internet on a pay-as-you-go basis.

It is widely distributed, network-based and used for storage.

There type of cloud are public, private, hybrid and community and some cloud providers are Google cloud, AWS, Microsoft Azure and IBM cloud.

Defining Cloud Computing Paradigm

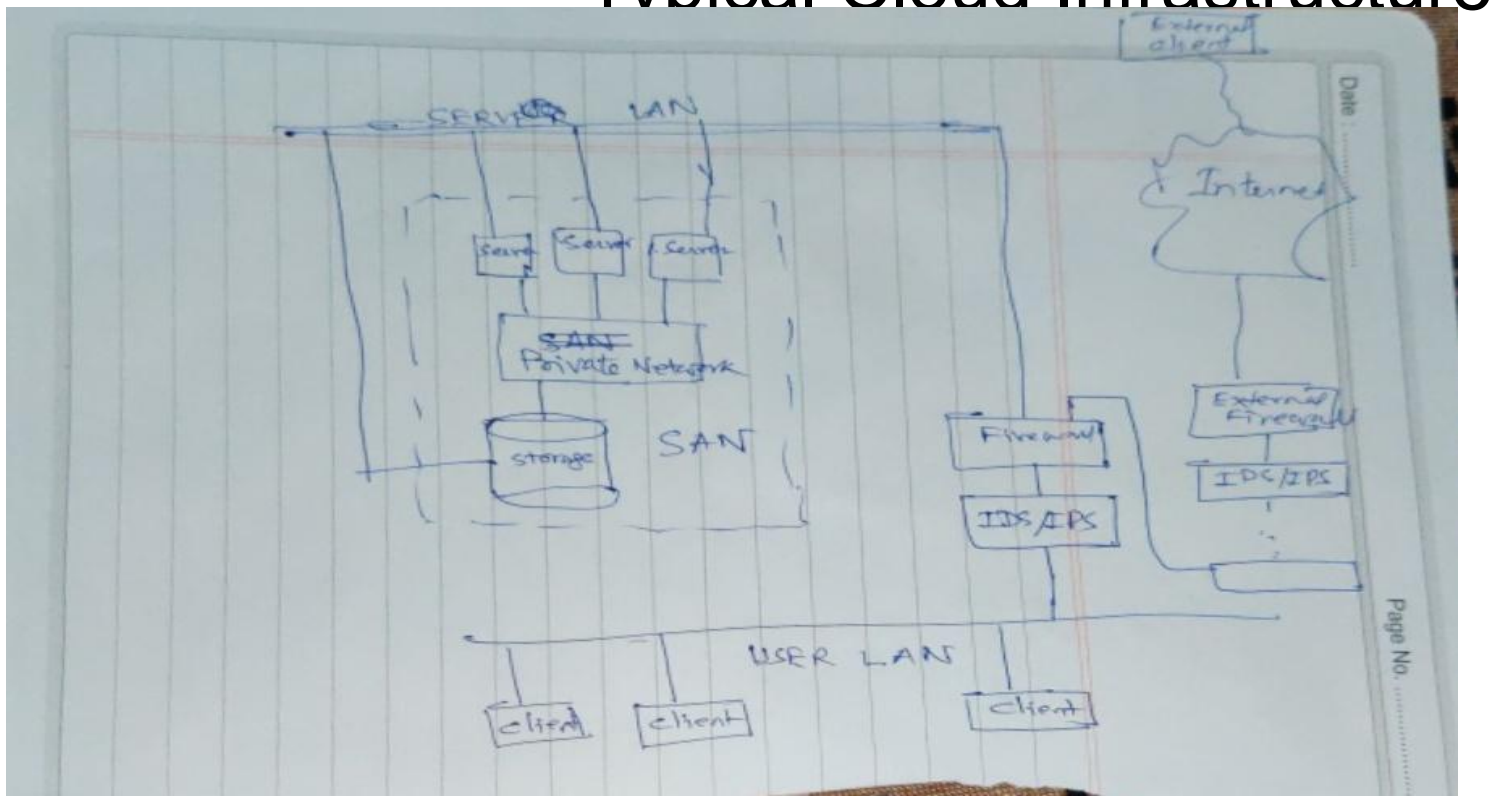
Cloud Computing :



Defining Cloud Computing Paradigm

Cloud Computing :

Typical Cloud Infrastructures



Cloud Computing :

Definitions of Cloud Computing

"A cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and the consumers."

"Cloud computing refers to both the applications delivered as services over the Internet, and the hardware and system software in the datacenters that provide those services."

Cloud Computing :

Features of Cloud Computing

- ✔ It is massively scalable,
- ✔ It can be encapsulated as an abstract entity that delivers different levels of services to customers outside the Cloud,
- ✔ It is driven by economies of scale,
- ✔ The services can be dynamically configured through virtualization
- ✔ Delivered on demand.

Defining Cloud Computing Paradigm

There are three main factors contributing to the surge of interest in Cloud Computing:

- ✔ Rapid decrease in hardware cost and increase in computing power and storage capacity,
- ✔ Advent of multi-core architectures and modern supercomputers consisting of hundreds of thousands of cores.
- ✔ The exponentially growing data size in scientific instrumentation/simulation and Internet publishing and archiving,
- ✔ The wide-spread adoption of Services Computing and Web 2.0 applications.

Cloud Types – NIST Model

There was a special publication of National Institute of Standards and Technology (NIST) - 800-145 in September 2011 .

As per NIST definitions,

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Cloud Types – NIST Model

As per this, cloud model is composed of

- ✔ Five essential characteristics,
- ✔ Three service models,
- ✔ Four deployment models.

Cloud Types – NIST Model

A. Essential Characteristics

(1) On-demand self-service.
A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.

(2) Broad network access.
Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).

Cloud Types – NIST Model

A. Essential Characteristics

(3) Resource pooling

The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.

There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter).

Examples of resources include storage, processing, memory, and network bandwidth.

Cloud Types – NIST Model

A. Essential Characteristics

(4) Rapid elasticity.

Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.

(5) Measured service.

Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts).

Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

Cloud Types – NIST Model

B. Service Models

(1) Software as a Service (SaaS).

The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user specific application configuration settings.

Cloud Types – NIST Model

B. Service Models

(2) Platform as a Service (PaaS)

The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider.

The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment

Cloud Types – NIST Model

B. Service Models

(3) Infrastructure as a Service (IaaS)

The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications.

The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).

Cloud Types – NIST Model

C. Deployment Models

1) Private cloud.

The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units).

Can be owned, managed, and operated by the organization, a third party, or some combination of them

Can exist as on premises.

Cloud Types – NIST Model

C. Deployment Models

(2) Community cloud

The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations).

Owned, managed, and operated by one or more of the organizations in the community
a third party

Can exist as on or off premises.

Cloud Types – NIST Model

C. Deployment Models

(3) Public cloud

The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them.

It exists on the premises of the cloud provider.

Cloud Types – NIST Model

C. Deployment Models

(4) Hybrid cloud

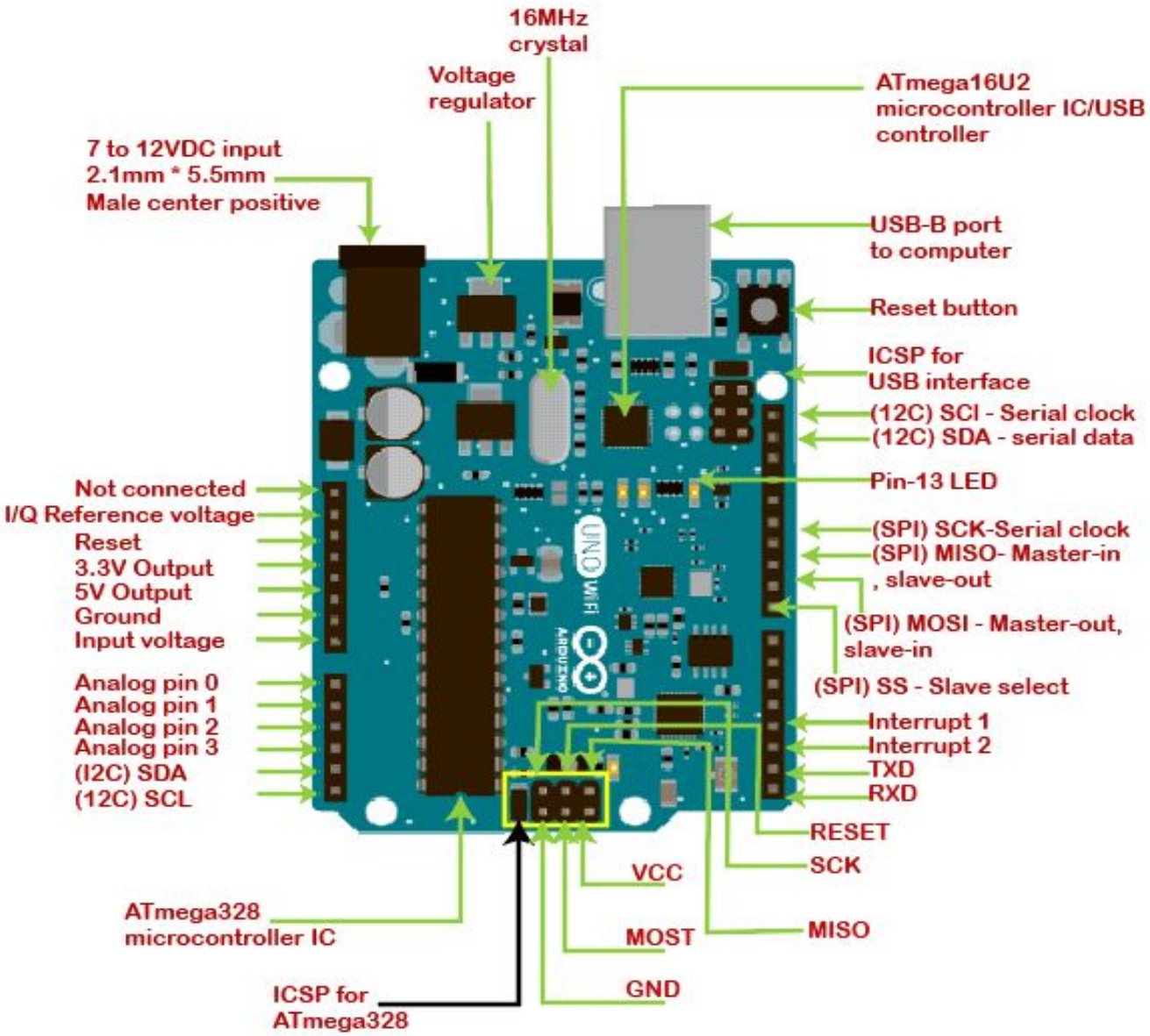
The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability.

Examples - Cloud bursting for load balancing between clouds

Familiarization with Arduino Uno Boards

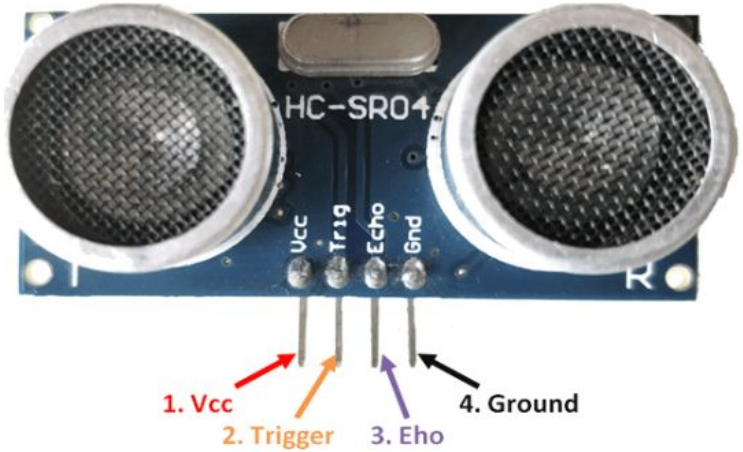
Arduino is a software as well as hardware platform that helps in making electronic projects. It is an open source platform and has a variety of controllers and microprocessors.

Pin configurations:



Sensors:

HCSR04 Ultrasonic Sensor Module Distance Measuring Sensor for Arduino is a well-equipped sensor for object detection and distance measuring



Cloud Computing
Module 1

Subject : Cloud Computing for the Students of IEM

Benefits of Cloud Computing

(1) Ability to get rid of most or all hardware and software

- ✔ No need to have own server, cables, network switches, backup generators, redundant routers, and so on.
- ✔ Cloud provider can manage all of these for a monthly fee.
- ✔ Reducing expenses is essential in any business model and cloud platform can benefit their consumers in this factor.

Benefits of Cloud Computing

(2) Centralized data security

Data backups are centralized in the cloud providers' data centers. Consumers do not need to take any onsite or offsite backup.

Consumers can take advantage of cloud security technologies such as data encryption and two-factor authentication for greater privacy.

Benefits of Cloud Computing

(3) Higher performance and availability

- ✔ Cloud computing increases input/output operations per second (IOPS).
- ✔ Cloud services also offer high availability with no downtime because they're distributed across multiple cloud facilities.
- ✔ Cloud providers are responsible for updating cloud systems and fixing bugs and security issues in cloud software, which is transparent to end users.

Benefits of Cloud Computing

(4) Quick application deployment

- ✔ Unpredictable business needs often require cloud computing resources on short notice.
- ✔ You can improve your cloud application development by quickly deploying cloud applications because they are readily available without the need to procure additional hardware or wait for IT staff to set up servers.

Benefits of Cloud Computing

(5) Instant business insights

Cloud-based platforms provide a unique opportunity to access data as soon as it's collected.

This facilitates better decision-making as well as insight into what the future may hold for your organization based on predictions from historical data.

Benefits of Cloud Computing

(6) Business continuity

In the event of disaster or unforeseen circumstances

Cloud is an effective solution

Benefits of Cloud Computing

(7) Price-performance and cost savings

- ✔ Comparatively lower initial investment is required to implement a cloud strategy.
- ✔ Organizations save substantial amounts in the long run as no need of maintaining expensive hardware or data centers.
- ✔ Since there are no upfront costs to use cloud-based systems, new businesses can test them.

Benefits of Cloud Computing

(8) Virtualized computing

Cloud computing is perfect for virtualized computer environments because cloud resources can be allocated instantly to support significant increases in demand so you never experience downtime again.

With cloud computing, your business can expand its capabilities almost effortlessly to meet growing demands without increasing staff or capital expenditures.

Benefits of Cloud Computing

(9) Cloud computing is greener

Cloud computing is a greener technology than traditional IT solutions. By moving to the cloud, businesses can reduce their energy consumption and carbon footprint by up to 90%.

Rather than having in-house servers and software, businesses can use cloud-based services to access the same applications and data from any computer or device with an internet connection.

This eliminates the need for businesses to purchase and maintain their own IT infrastructure.

Cloud Cube Model

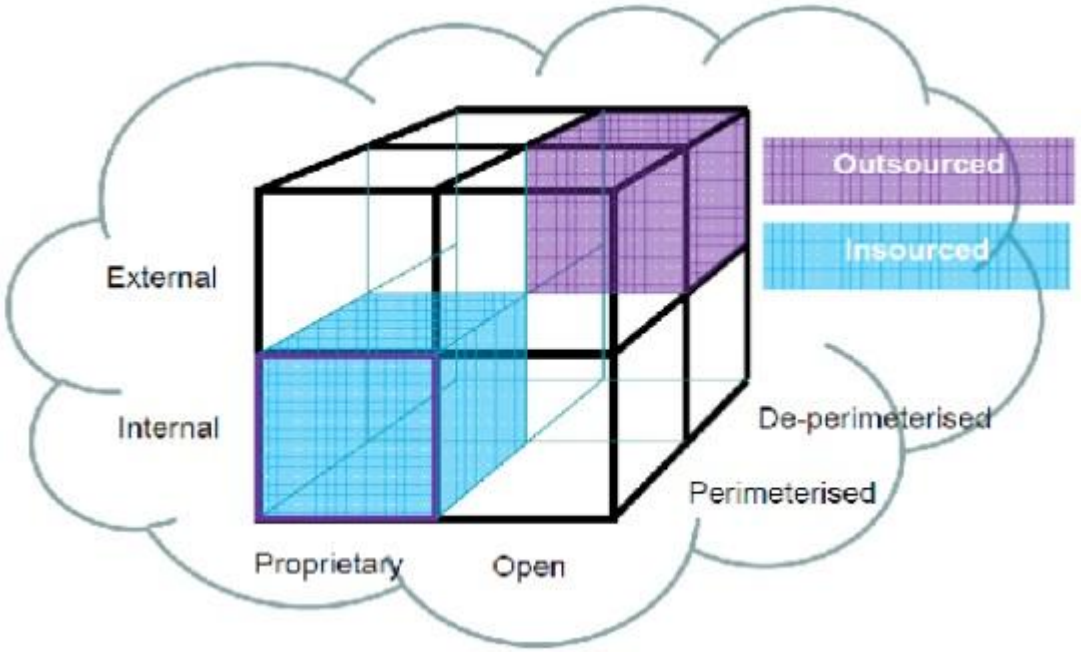
Cloud Cube Model, designed and developed by Jericho forum. Which helps to categorize the cloud network based on the four-dimensional factor: Internal/External, Proprietary/Open, De-Perimeterized/Perimeterized, and Insourced/Outsourced.

The main goal of cloud cube model is to provide the security to the cloud network and protect it.
In cloud computing security plays an important part for different cloud users.

Cloud cube model also enables secure collaboration of cloud formations

It is helpful for different types of organizations and businesses.

Cloud Cube Model



Dimensions of Cloud Cube Model

(1) Internal/External:

The most basic cloud form is the **external and internal cloud form.**
The external or internal dimension defines the physical location of the data. It acknowledges us whether the data exists inside or outside of your organization's boundary.

Here, the data which is stored using a **private cloud** deployment will be considered internal and data outside the cloud will be considered external.

Cloud Cube Model

Dimensions of Cloud Cube Model

(2) Proprietary/Open

The second type of cloud formation is **proprietary/open**. The proprietary or open dimension states about the state of ownership of the **cloud technology** and interfaces. It also tells the degree of interoperability, while enabling data transportability between the system and other cloud forms.

The **proprietary dimension** means, that the organization providing the **service is securing** and protecting the data under their ownership.

The **open dimension** is using a technology in which there are more suppliers. Moreover, the user is not constrained in being able to share the data and collaborate with selected partners using the open technology.

(3) Perimeterized/de-perimeterized:

The Perimeterised and De-perimeterized dimension tells us whether you are operating inside your traditional it mindset or outside it.

Perimeterized dimension means, continuing to operate within the traditional IT boundary.

With the help of VPN and operation of the virtual server in your own IP domain, the user can extend the organizations perimeter into external Cloud Computing domain. This means that the user is making use of the own services to control access.

Cond.

Perimeterized/de-perimeterized:

In De-perimeterized dimension, the data will be encapsulated with metadata and mechanisms, which will further help to protect the data and limit the inappropriate usage.

(4) Insourced/Outsourced

The **Insourced/Outsourced dimensions** have two states.

In the *outsourced dimension* the services provided by the third party.

In the *insourced dimension* the services provided by the own staff under the control.

Cloud Cube Model

How to Secure Data in the Cloud Cube Model?
There are some steps and points to keep in mind before securing your data in a cloud cube model:

Step 1
The **classification of the data**, the customer should know what rules must be applied to protect it.

Step 2
It should be **ensured**, the data exist only in specific trust levels.

Step 3
It should check that what **regulatory compliance and restrictions** are applicable. For example, the data should stay in a particular boundary and whether it has to stay in the safe harbor or not.

How to Secure Data in the Cloud Cube Model?

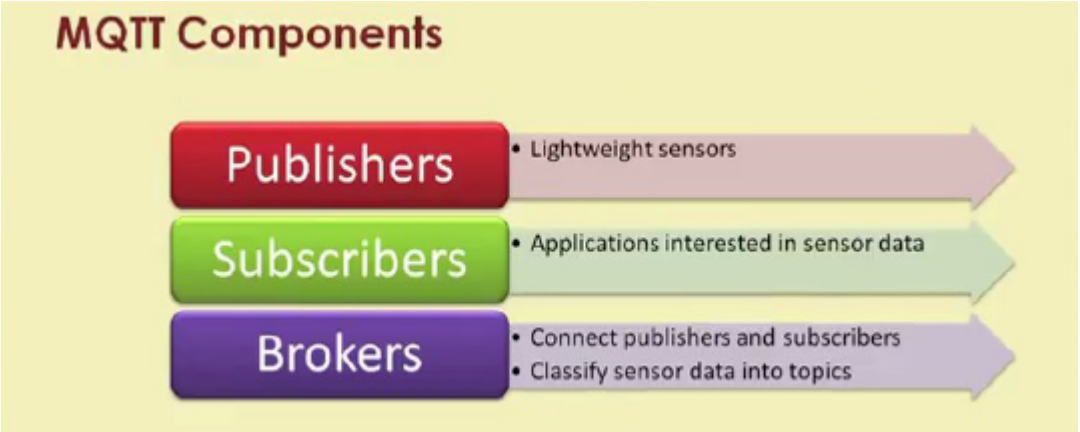
After the data is classified and is ready to put in the required zone, the assigned person is in a position to decide the following factors:-

The data and processes, which are to be moved in the cloud.
At what level the user wants to operate in the cloud. It can be infrastructure, platform, software, or **platform as a service**.
The cloud formations, which are mostly compatible as per the requirement.
The level of **operation in the cloud** can be different as per the requirement.
Below is the chart which shows the Cloud layers, where the cloud operates.

MQTT: Message queue telemetry transport or MQTT is a simple, lightweight publish–subscribe protocol, designed mainly for messaging in constrained devices and networks. It provides a one-to-many distribution of messages and is payload content agnostic. MQTT works reliably and flawlessly over high latency and limited bandwidth of unreliable networks without the need for significant device resources and device power.

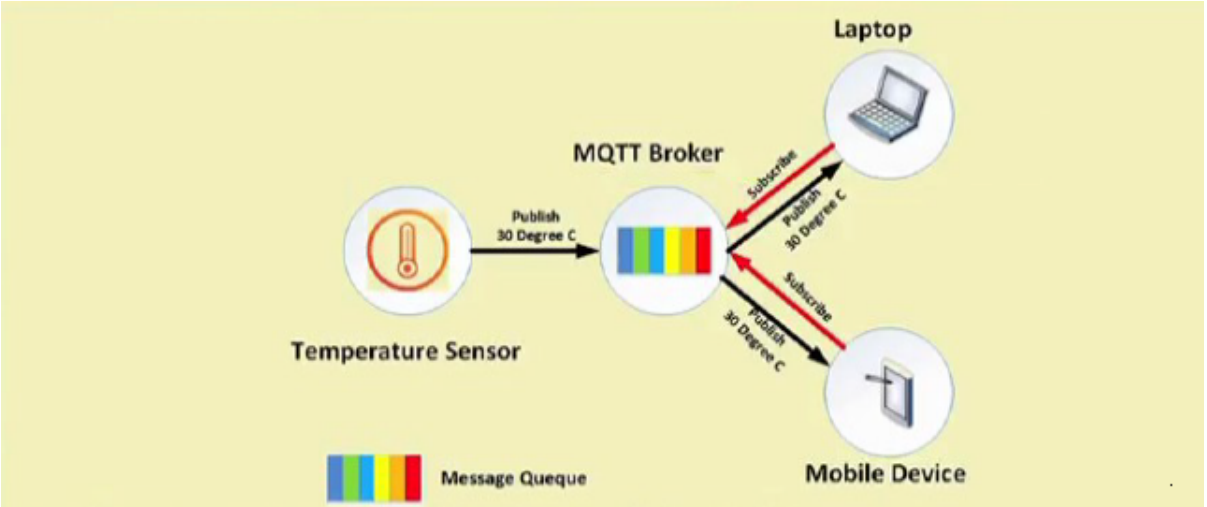
- ✓ **Message Queue Telemetry Transport.**
- ✓ ISO standard (ISO/IEC PRF 20922).
- ✓ It is a publish-subscribe-based lightweight messaging protocol for use in conjunction with the TCP/IP protocol.
- ✓ MQTT was introduced by IBM in 1999 and standardized by OASIS in 2013.
- ✓ Designed to provide connectivity (mostly embedded) between applications and middle-wares on one side and networks and communications on the other side.

- ✓ A message broker controls the publish-subscribe messaging pattern.
- ✓ A topic to which a client is subscribed is updated in the form of messages and distributed by the message broker.
- ✓ Designed for:
  - Remote connections
  - Limited bandwidth
  - Small-code footprint



MQTT's control message sizes can range between 2 bytes to 256 megabytes of data, with a fixed header size of 2 bytes. This enables the MQTT to reduce network traffic significantly. The connection credentials in MQTT are unencrypted and often sent as plain text. The responsibility of protecting the connection lies with the underlying TCP layer. The MQTT protocol provides support for 14 different message types, which range from connect/disconnect operations to acknowledgments of data.

The following are the standard MQTT message types:
(i) CONNECT: Publisher/subscriber request to connect to the broker.
(ii) CONNACK: Acknowledgment after successful connection between publisher/ subscriber and broker.
(iii) PUBLISH: Message published by a publisher to a broker or a broker to a subscriber.
(iv) PUBACK: Acknowledgment of the successful publishing operation.
(v) PUBREC: Assured delivery component message upon successfully receiving publish.
(vi) PUBREL: Assured delivery component message upon successfully receiving publish release signal.
(vii) PUBCOMP: Assured delivery component message upon successfully receiving publish completion.
(viii) SUBSCRIBE: Subscription request to a broker from a subscriber.
(ix) SUBACK: Acknowledgment of successful subscribe operation.
(x) UNSUBSCRIBE: Request for unsubscribing from a topic.
(xi) UNSUBACK: Acknowledgment of successful unsubscribe operation.
(xii) PINGREQ: Ping request message.
(xiii) PINGRESP: Ping response message.
(xiv) DISCONNECT: Message for publisher/subscriber disconnecting from the broker.



Source: "MQTT 101 – How to Get Started with the lightweight IoT Protocol", HiveMQ (Online)

Communication

- ✓ The protocol uses a **publish/subscribe** architecture (HTTP uses a request/response paradigm).
- ✓ Publish/subscribe is **event-driven** and enables messages to be pushed to clients.
- ✓ The central **communication point is the MQTT broker**, which is in charge of dispatching all messages between the senders and the rightful receivers.
- ✓ Each client that publishes a message to the broker, includes a **topic** into the message. The **topic is the routing information for the broker.**

- ✓ Each client that wants to receive messages subscribes to a certain topic and the broker delivers all messages with the matching topic to the client.
- ✓ Therefore the clients don't have to know each other. They only communicate over the topic.
- ✓ This architecture enables highly scalable solutions without dependencies between the data producers and the data consumers.

MQTT Topics

- ✓ A topic is a **simple string** that can have more hierarchy levels, which are separated by a slash.
- ✓ A sample topic for sending temperature data of the living room could be *house/living-room/temperature*.
- ✓ On one hand the client (e.g. mobile device) can subscribe to the exact topic or on the other hand, it can use a **wildcard**.

- ✓ The subscription to *house/+/temperature* would result in all messages sent to the previously mentioned topic *house/living-room/temperature,* as well as any topic with an arbitrary value in the place of living room, such as *house/kitchen/temperature*.
- ✓ The plus sign is a **single level wild card** and only allows arbitrary values for one hierarchy.
- ✓ If more than one level needs to be subscribed, such as, the entire sub-tree, there is also a **multilevel wildcard** (#).
- ✓ It allows to subscribe to all underlying hierarchy levels.
- ✓ For example *house/#* is subscribing to all topics beginning with *house*.

Applications

- ✓ **Facebook Messenger** uses MQTT for online chat.
- ✓ **Amazon Web Services** use Amazon IoT with MQTT.
- ✓ **Microsoft Azure** IoT Hub uses MQTT as its main protocol for telemetry messages.
- ✓ The **EVRYTHNG IoT platform** uses MQTT as an M2M protocol for millions of connected products.
- ✓ **Adafruit** launched a free MQTT cloud service for IoT experimenters called Adafruit IO.

MQTT-SN

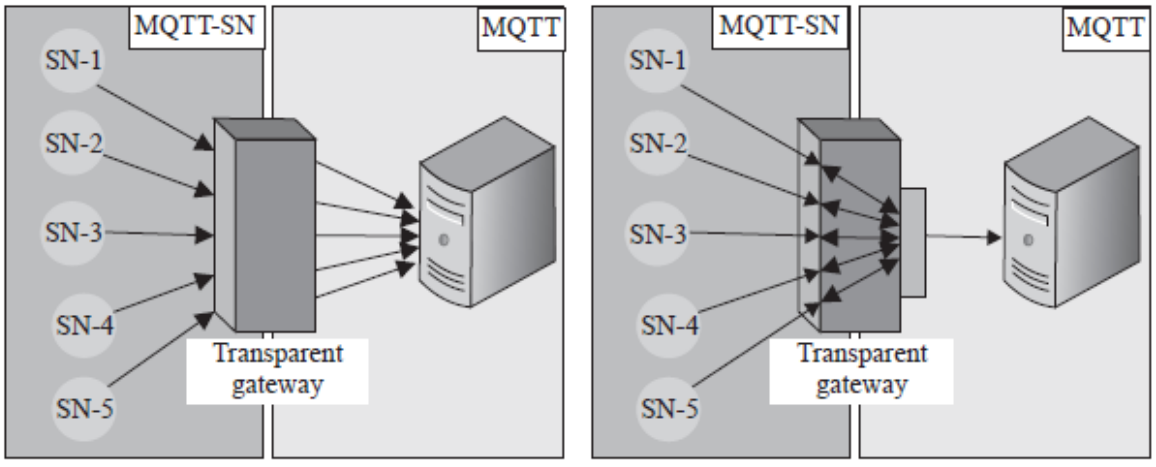
The primary MQTT protocols heavily inspire **MQTT for sensor networks or MQTTSN**; however, the MQTT-SN is robust enough to handle the requirements and challenges of wireless communications networks in sensor networks. Typical features of MQTT-SN include

low bandwidth usage, ability to operate under high link failure conditions; it is suitable for low-power, low-cost constrained nodes and networks.

The major differences between the original MQTT and MQTT-SN include the following:

i) The CONNECT message types are broken into three messages in which two are optional and are tasked with the communication of the testament message and testament topic to the broker.
ii) The topic name in the PUBLISH messages are replaced by topic identifiers, which are only 2 bytes long. This reduces the traffic generated from the protocol and enables the protocol to operate over bandwidth-constrained networks.
iii) A separate mechanism is present for topic name registration with the broker in MQTT-SN. After a topic identifier is generated for the topic name, the identifier is informed to the publisher/subscribers. This mechanism also supports the reverse pathway.
iv) In special cases in MQTT-SN, pre-defined topic identifiers are present that need no registration mechanism. The mapping of topic names and identifiers are known in advance to the broker as well as the publishers/subscribers.
v) The presence of a special discovery process is used to link the publisher/subscriber to the operational broker's network address in the absence of a preconfigured broker address.
vi) The subscriptions to a topic, Will topic, and Will message are persistent in MQTTSN. The publishers/subscribers can modify their Will messages during a session.
vii) Sleeping publishers/subscribers are supported by a keep-alive procedure, which is offline, and which helps buffer the messages intended for them in the broker until they wake up. This feature of MQTT-SN is not present in regular MQTT.

Figure shows the two gateway types in MQTT-SN:
1) transparent gateway and
2) aggregating gateway.
The MQTT-SN converts/translates MQTT and MQTTSN traffic by acting as a bridge between these two network types. The transparent gateway creates as many connections to the MQTT broker as there are MQTT-SN nodes within its operational purview; whereas the aggregating gateway creates a single connection to the MQTT broker, irrespective of the number of MQTT-SN nodes under it.



(a) Transparent gateway (b) Aggregating gateway

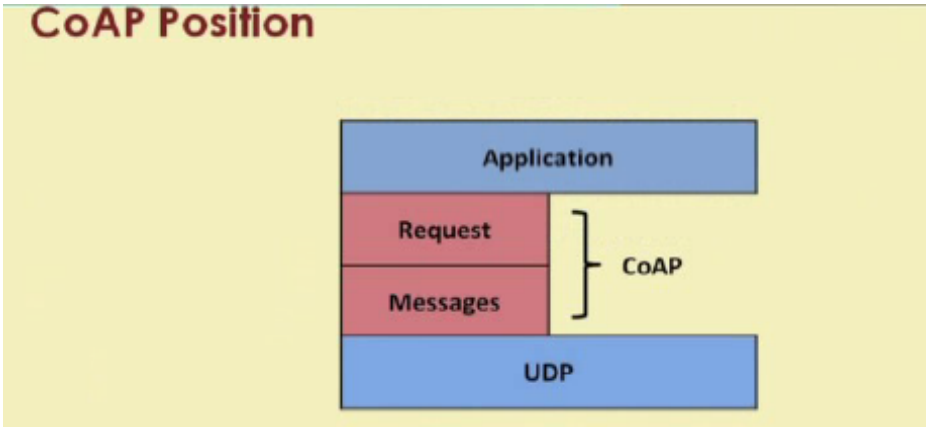
COAP:
The constrained application protocol, or CoAP as it is more popularly known, is designed for use as a web transfer protocol in constrained devices and networks, which are typically low power and lossy. The constrained devices typically have minimal RAM and an 8-bit processor at most. CoAP can efficiently work on such devices, even when these devices are connected to highly lossy networks with high packet loss, high error rates, and bandwidth in the range of kilobits.

Introduction

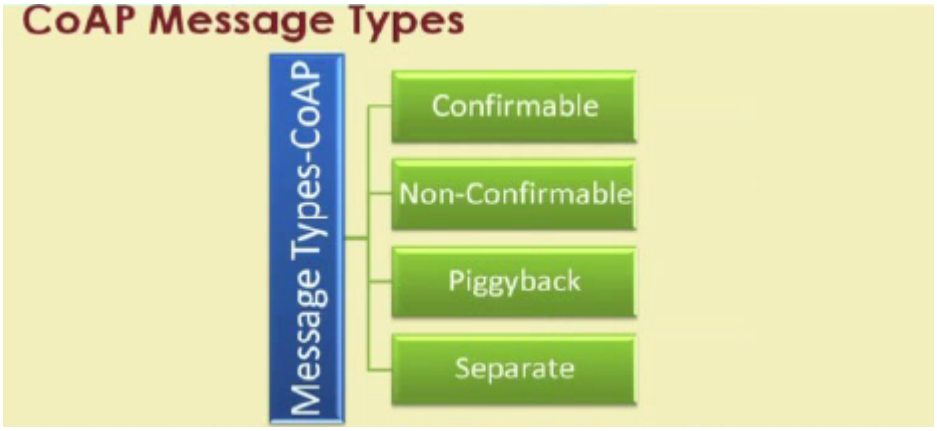
- ✓ CoAP – **Constrained Application Protocol**.
- ✓ **Web transfer protocol** for use with constrained nodes and networks.
- ✓ **Designed for Machine to Machine** (M2M) applications such as smart energy and building automation.
- ✓ Based on **Request-Response model** between end-points
- ✓ Client-Server interaction is **asynchronous over a datagram oriented transport protocol** such as UDP

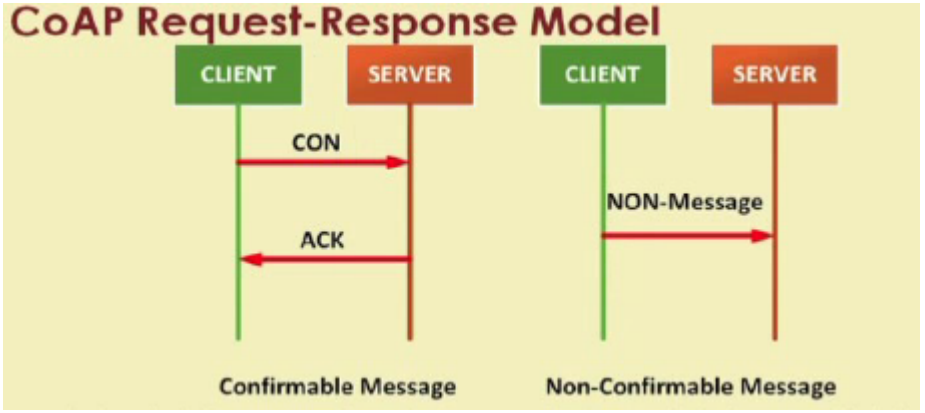
- ✓ The Constrained Application Protocol (CoAP) is a session layer protocol designed by IETF Constrained RESTful Environment (CoRE) working group to provide lightweight RESTful (HTTP) interface.
- ✓ Representational State Transfer (REST) is the standard interface between HTTP client and servers.
- ✓ Lightweight applications such as those in IoT, could result in significant overhead and power consumption by REST.
- ✓ CoAP is designed to enable low-power sensors to use RESTful services while meeting their power constraints.

- ✓ Built over UDP, instead of TCP (which is commonly used with HTTP) and has a light mechanism to provide reliability.
- ✓ CoAP architecture is divided into two main sub-layers:
  - Messaging
  - Request/response.
- ✓ The messaging sub-layer is responsible for reliability and duplication of messages, while the request/response sub-layer is responsible for communication.
- ✓ CoAP has four messaging modes:
  - Confirmable
  - Non-confirmable
  - Piggyback
  - Separate

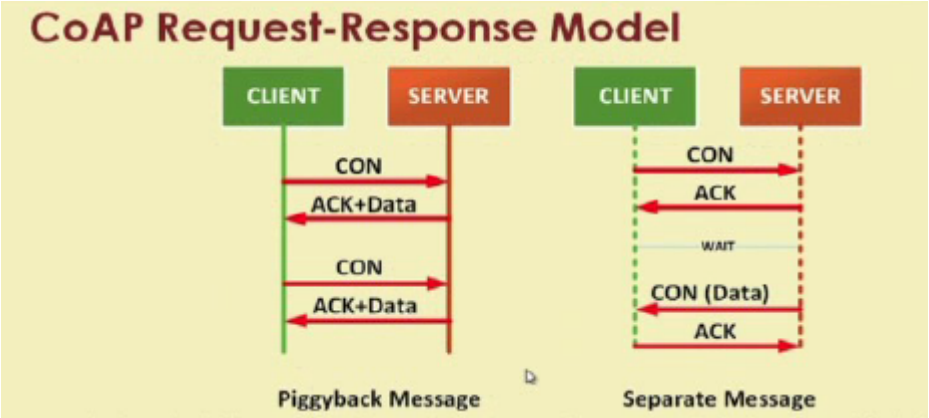


CoAP is a protocol of the session layer.





- ✓ Confirmable and non-confirmable modes represent the reliable and unreliable transmissions, respectively, while the other modes are used for request/response.
- ✓ Piggyback is used for client/server direct communication where the server sends its response directly after receiving the message, i.e., within the acknowledgment message.
- ✓ On the other hand, the separate mode is used when the server response comes in a message separate from the acknowledgment, and may take some time to be sent by the server.
- ✓ Similar to HTTP, CoAP utilizes GET, PUT, PUSH, DELETE messages requests to retrieve, create, update, and delete, respectively



Features

- ✓ Reduced overheads and parsing complexity.
- ✓ URL and content-type support.
- ✓ Support for the discovery of resources provided by known CoAP services.
- ✓ Simple subscription for a resource, and resulting push notifications.
- ✓ Simple caching based on maximum message age.

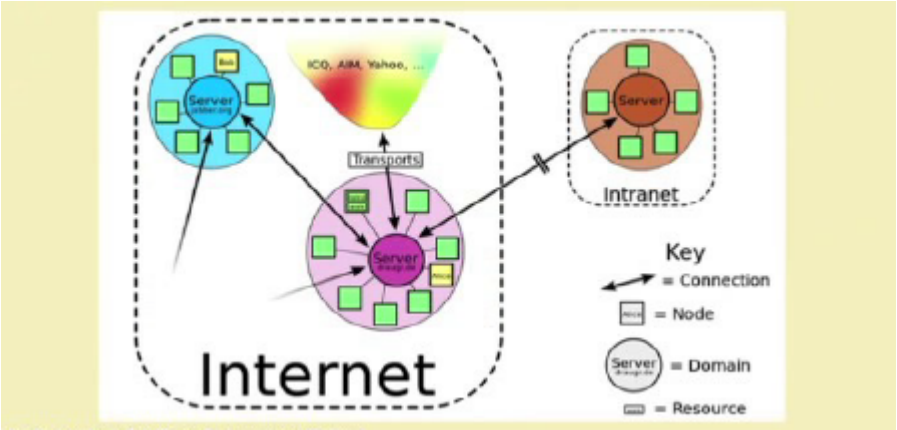
XMPP :

XMPP is **Extensible Messaging and Presence Protocol.** It is a message oriented middleware that is based on XML, whereas XML is particularly used for unstructured data. XMPP is useful for real time exchange of structured data and it is an open standard protocol. XMPP uses a client server architecture, it uses a decentralized model meaning that there is no server that is involved in the message transfer and it provides facilities for discovery of messages which are residing locally or globally across the network and the availability information of these services.

- ✓ XMPP uses a **client-server architecture**.
- ✓ As the model is **decentralized**, no central server is required.
- ✓ XMPP provides for the **discovery of services** residing locally or across a network, and the **availability information** of these services.
- ✓ Well-suited for cloud computing where virtual machines, networks, and firewalls would otherwise present obstacles to alternative service discovery and presence-based solutions.
- ✓ Open means to support machine-to-machine or peer-to-peer communications across a diverse set of networks.

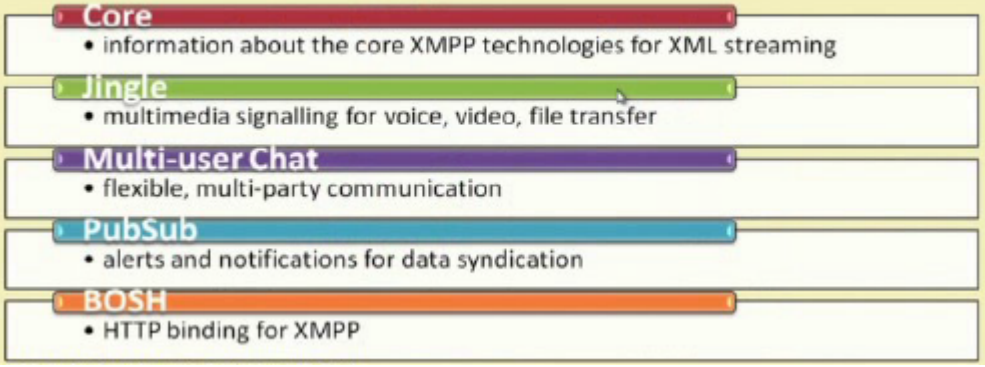
Highlights

- ✓ Decentralization – No central server; anyone can run their own XMPP server.
- ✓ Open standards – No royalties or granted permissions are required to implement these specifications
- ✓ Security – Authentication, encryption, etc.
- ✓ Flexibility – Supports interoperability



This particular figure what we see is with the help of XMPP, not only it is possible to communicate with other servers like in the case of the traditional internet, but also with other messaging platform such as ICQ, AIM, Yahoo and so on. Additionally it is also possible to communicate with other intranets.

Core XMPP Technologies



Weaknesses

- ✓ Does not support QoS.
- ✓ Text based communications induces higher network overheads.
- ✓ Binary data must be first encoded to base64 before transmission.

Applications

- ✓ Publish-subscribe systems
- ✓ Signaling for VoIP
- ✓ Video
- ✓ File transfer
- ✓ Gaming
- ✓ Internet of Things applications
  - Smart grid
  - Social networking services

Different Protocols in IoT

1. Infrastructure Protocols

1. **Internet protocol version 4 (IPv4)**
2. **Internet protocol version 6 (IPv6)**
3. LOADng
4. **RPL (Routing Protocol for Low-Power and Lossy Networks)**
5. 6LoWPAN
6. QUIC
7. Micro internet protocol (uIP)
8. Nano internet protocol (nanoIP)
9. Content-centric networking (CCN)

2. Discovery Protocols

1. Physical web
2. Multicast DNS (mDNS)
3. Universal plug and play (UPnP)

3. Data Protocols

1. **MQTT**
2. **MQTT-SN**
3. **CoAP**
4. AMQP
5. **XMPP**
6. SOAP
7. REST
8. WebSocket

4. Identification Protocols

1. EPC
2. uCode
3. URIs

5.Device Management

1. **TR-069**
2. **OMA-DM**

6. Semantic Protocols

1. JSON-LD
2. Web thing model

1. Infrastructure Protocols:

IPv4: The IPv4 header packet shown in Figure has 13 distinct fields, the functions of which are given as follows.

• **VER**: It is 4 bits long and represents the version of IP. It is 4 bits (binary: 0100).

• **HLEN**: It is 4 bits long and denotes the length of the IPv4 packet header.

• **ToS**: It is 8 bits long. The first six most significant bits represent the differentiated services code point (DSCP) to be provided to this packet (by the routers). Explicit congestion notification (ECN), which gives information about the congestion witnessed in the network, is handled by the last 2 bits.

• **TOTAL LENGTH**: It is 16 bits long and identifies the length of the entire IPv4 packet, including the header and the payload.
• **IDENTIFIER**: It is 16 bits long and used to identify the original packets in case of packet fragmentation along the network.
• **FLAGS**: It is a 3-bit field with the most significant bit always set to 0. FLAGS indicates whether a packet can be fragmented or not in case the packet is too big for the network resources.
• **FRAGMENT OFFSET**: It identifies the exact offset or fragment position of the original IP packet and is 13 bits long.
• **TTL**: It is 8 bits long and prevents a packet from looping infinitely in the network. As it completes a link, its value is decremented by one.
• **PROTOCOL**: It is 8 bits long. This field identifies the protocol of the packet as user datagram protocol, UDP (17), transmission control protocol, TCP (6), or Internet control message protocol, ICMP (1). The identification is made at the network layer of the destination host.
• **HEADER CHECKSUM**: It is 16 bits long and used for identifying whether a packet is error-free or not.
• **SOURCE ADDRESS**: It indicates the origin address of the packet and is 32 bits long.
• **DESTINATION ADDRESS**: It indicates the destination address of the packet and is 32 bits long.
• **OPTIONS and PADDING**: It is an optional field, which may carry values for security, time stamps, route records, and others.

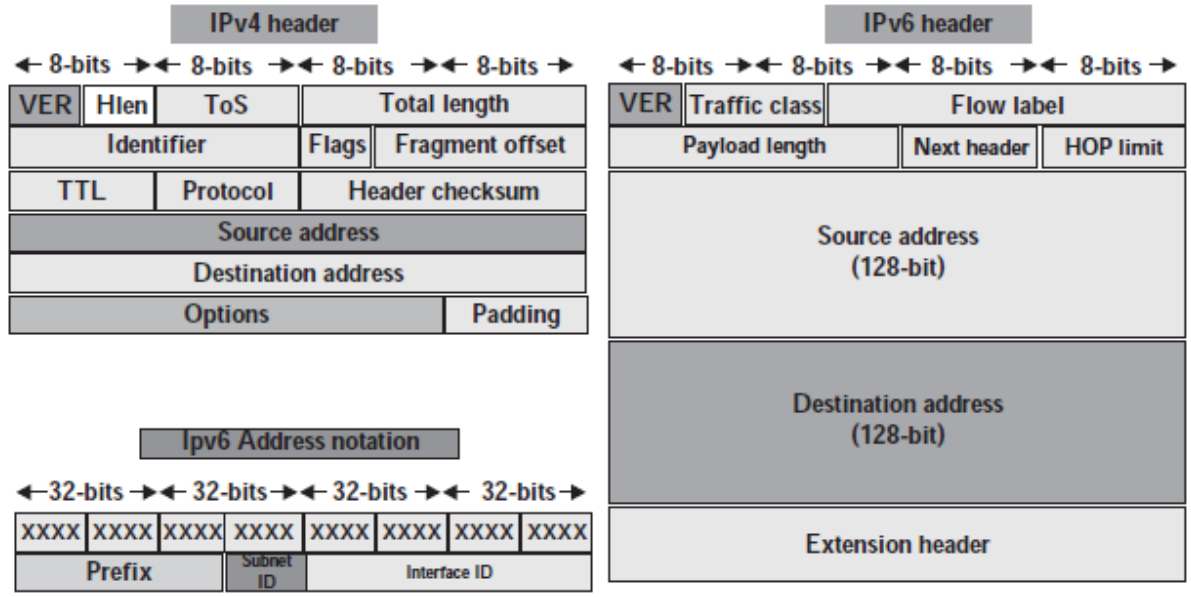
IPV6:

The IPv6 header packet shown in Figure has eight distinct fields, the functions of which are given as follows.
• **VER**: It is 4 bits long and represents the version of IP. It is 6 (binary: 0110).
• **TRAFFIC CLASS**: It is 8 bits long. The first six most significant bits represent the type of service to be provided to this packet (by the routers); explicit congestion notification (ECN) is handled by the last 2 bits.
• **FLOW LABEL**: It is 20 bits long and designed for streaming media or real time data. The FLOW LABEL allows for information flow ordering; it also avoids packet resequencing.
• **PAYLOAD LENGTH**: It is 16 bits long and provides a router with information about a packet's payload length or the amount of data contained in the packet's payload.
• **NEXT HEADER**: It is 8 bits long and informs the router about the type of extension header the packet is carrying. Some of the extension headers and their corresponding values are as follows: Hop-by-hop options header (0), routing header (43), fragment header (44), destination options header (60), authentication header (51), and encapsulating security payload header (50).
In case an extension header is absent, it represents the upper layer protocol data units (PDUs).
• **HOP LIMIT**: It is 8 bits long and prevents a packet from looping infinitely in the network. As it completes a link, the limit's value is decremented by one.
• **SOURCE ADDRESS**: It is 128 bits long and indicates the origin address of the packet.
• **DESTINATION ADDRESS**: It is 128 bits long and indicates the destination address of the packet.

The Internet Protocol Version 6 or IPv6 is a resultant of the developments on and beyond IPv4 due to fast depleting address ranges in IPv4. The IPv4 was not designed to handle the

needs of the future internet systems. The needs of massive scalability and limited resources gave rise to IPv6, which was developed by the IETF (Internet Engineering Task Force. IPv6 works on the OSI layer 3 (network layer). However, in contrast **to IPv4 (which is 32 bits long** and offers around 4,294,967,296 addresses), **IPv6** has a massive logical address range (which is **128 bits long**). Additional features in IPv6 include auto-configuration features, end-to-end connectivity, inbuilt security measures (IPSec), provision for faster routing, support for mobility, and many others.
These features not only make IPv6 practical for use in IoT but also makes it attractive for a majority of the present-day and upcoming IoT-based deployments. IPV6 it cannot be made to support IPv4 applications directly.

Figure shows the differences between IPv4 and IPv6 packet structures.



Important features of IPv6 are as follows:

(i) **Larger Addressing Range:** IPv6 has roughly four times more addressable bits than IPv4.

(ii) **Simplified Header Structure:** IPv6 header format is quite simple than IPv4. IPv6 header's increased size is mainly due to the increased number of bits needed for addressing purposes.

(iii) **End-to-End Connectivity:** Addressing allows packets from a source node to directly reach the destination node without the need for network address translations using IPv6.

(iv) **Auto-configuration:** The configuration of addresses is automatically done in IPv6.

(v) **Faster Packet Forwarding:** Routing decisions by a router are taken much faster by checking only the first few fields of the header.

(vi) **Inbuilt Security:** IPv6 supports inbuilt security mechanisms. IPv6 has security as an optional feature.

(vii) **Anycast Support :** Multiple networking interfaces are assigned the same IPv6 addresses globally; these addresses are known as anycast addresses. This mechanism enables routers to send packets to the nearest available destination during routing.

(viii) **Mobility Support**: The mobility support of IPv6 allows for mobile nodes to retain their IP addresses and remain connected, even while changing geographic areas of operation.

(ix) **Enhanced Priority Support**: The priority support system in IPv6 is entirely simplified as compared to IPv4. The use of traffic classes and flow labels determine the most efficient routing paths of packets for the routers.

(x) **Extensibility of Headers**: The options part of an IPv6 header can be extended by adding more information to it; it is not limited in size. Some applications may require quite a large options field, which may be comparable to the size of the packet itself.

RPL

RPL stands for routing protocol for low-power and lossy networks (LLN) and is designed for IPv6 routing. It follows a distance vector based routing mechanism. The protocol aims to achieve a **destination-oriented directed acyclic graph** (DODAG). The network DODAG is formed based on an objective function and a set of network metrics. The DODAG built by RPL is a **logical routing topology** which is built over a physical network. The logical topology is built using specific criteria set by network administrators. The most optimum path (best path) is calculated from the objective function, a set of metrics, and constraints. The **metrics in RPL** may be **expected transmission values (ETX), path latencies**, and others. Similarly, the **constraints** in RPL include encryption of links, **the presence of battery-operated nodes,** and others. In general, the metrics are either minimized or maximized, whereas the constraints need to be minimized. The **objective function** dictates **the rules for the formation of the DODAG.** Interestingly, in RPL, a single node in the mesh network may have multiple objective functions. The primary reason for this is attributed to the presence of different network traffic path quality requirements that need separate addressal within the same mesh network. Using RPL, a node within a network can simultaneously join more than one RPL instance (graphs). This enables RPL to support QoS-aware and constraint-based routing. An RPL node can also simultaneously take on multiple network roles: leaf node, router, and others. Figure below shows the RPL mechanism with different intra-mesh addressing arising due to different requirements of network and objective functions. The RPL border router, which is also the RPL root (in the illustrated figure), handles the intra-mesh addressing.

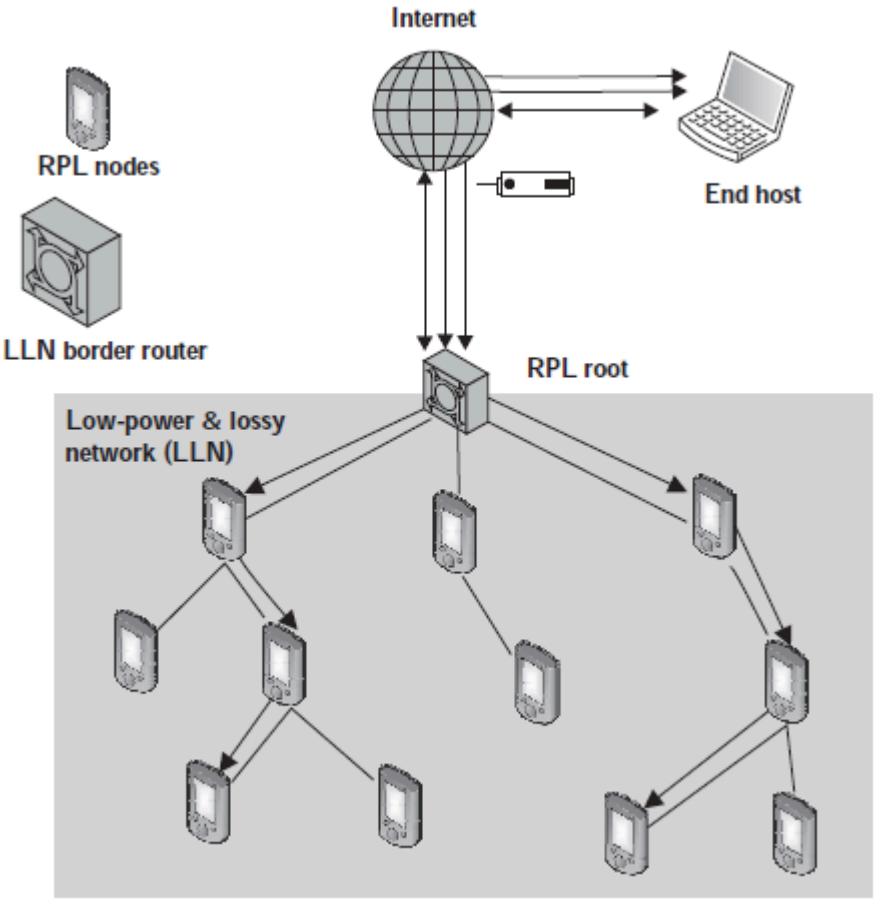


Figure: RPL information ow mechanism with different intra-mesh addressing and paths

RPL Instances

There are two instances associated with RPL: global and local. Global RPL instances are characterized by coordinated behavior and the possibility of the presence of more than one DODAG; they have a long lifetime. Local RPL instances are characterized by single DODAGs. The local RPL DODAG's root is associated directly with the DODAG-ID. The RPL instance ID is collaboratively and unilaterally allocated; it is divided between global and local RPL instances. Even the RPL control and data messages are tagged with their corresponding RPL instances using RPL instance IDs to avoid any ambiguity in operations.

Networking: Networking refers to the linking of computers and communication network devices (also referred to as **hosts**), which interconnect through a network (Internet or Intranet) and are separated by unique device identifiers (Internet protocol, IP addresses and media access control, MAC addresses). These hosts may be connected by a single path or through multiple paths for sending and receiving data. The data transferred between the hosts may be text, images, or videos, which are typically in the form of binary bit streams.

Network Types: i) Connection types: Point-to-point: Point-to-point connections are used to establish direct connections between two hosts. Day-to-day systems such as a remote controlfor an air conditioner or television is a point to point connection, where the connection has the whole channel dedicated to it only. These networks were designed to work over duplex links and are functional for both synchronous as well as asynchronous systems. Regarding computer networks, point to point connections find usage for specific purposes such as in optical networks.

Point-to-multipoint: In a point-to-multipoint connection, more than two hosts share the same link. This type of configuration is similar to the one-to-many connection type. Point-to-multipoint connections find popular use in wireless networks and IP telephony. The channel is shared between the various hosts, either spatially or temporally. One common scheme of spatial sharing of the channel is frequency division multiple access (FDMA). Temporal sharing of channels include approaches such as time division multiple access (TDMA).

ii) Physical topology: physical manner in which communication paths between the hosts are connected.

1. star: In a star topology, every host has a point-to-point link to a central controller or hub. The hosts cannot communicate with one another directly; they can only do so through the central hub. The hub acts as the network traffic exchange. For large-scale systems, the hub, essentially, has to be a powerful server to handle all the simultaneous traffic flowing through it. However, as there are fewer links (only one link per host), this topology is cheaper and easier to set up. The main advantages of the star topology are easy installation and the ease of fault identification within the network. If the central hub remains uncompromised, link failures between a host and the hub do not have a big effect on the network, except for the host that is affected. However, the main disadvantage of this topology is the danger of a single point of failure. If the hub fails, the whole network fails.
2. Mesh: In a mesh topology, every host is connected to every other host using a dedicated link (in a point-to-point manner). This implies that for n hosts in a mesh, there are a total of **n(n − 1)/2 dedicated** full duplex links between the hosts. This massive number of links makes the mesh topology expensive. However, it offers certain specific advantages over other topologies. The first significant advantage is the **robustness and resilience** of the system. Even if a link is down or broken, the network is still fully functional as there remain other pathways for the traffic to flow through. The second advantage is the **security and privacy of the traffic** as the data is only seen by the intended recipients and not by all members of the network. The third advantage is the **reduced data load on a single host,** as every host in this network takes care of its traffic load. However, owing to the complexities in forming physical connections between devices and the cost of establishing these links, mesh networks are used very selectively, such as in backbone networks.

3. Bus: A bus topology follows the point-to-multipoint connection. A **backbone cable or bus** serves as the primary traffic pathway between the hosts. The hosts are connected to the main bus employing **drop lines or taps.** The main advantage of this topology is the ease of installation. However, there is a restriction on the length of the bus and the number of hosts that can be simultaneously connected to the bus due to signal loss over the extended bus. The bus topology has a simple cabling procedure in which a single bus (backbone cable) can be used for an organization. Multiple drop lines and taps can be used to connect various hosts to the bus, making installation very easy and cheap. However, the main drawback of this topology is the **difficulty in fault localization** within the network.
4. **Ring:** A ring topology works on the principle of a point-to-point connection. Here, each host is configured to have a dedicated point-to-point connection with its two immediate neighboring hosts on either side of it through repeaters at each host. The repetition of this system forms a ring. The **repeaters** at each host capture the incoming signal intended for other hosts, regenerates the bit stream, and passes it onto the next repeater. Fault identification and set up of the ring topology is quite simple and straightforward. However, the main disadvantage of this system is the high probability of a single point of failure. If even one repeater fails, the whole network goes down.

iii) Network reachability: Computer networks are divided into four broad categories based on network reachability: personal area networks, local area networks, wide area networks, and metropolitan area networks.

(i) Personal Area Networks (PAN): PANs, as the name suggests, are mostly restricted to individual usage. A good example of PANs may be connected wireless headphones, wireless speakers, laptops, smartphones, wireless keyboards, wireless mouse, and printers within a house. Generally, PANs are wireless networks, which make use of low-range and low-power technologies such as **Bluetooth**. The reachability of PANs lies in the range of a few centimeters to a few meters.

(ii) Local Area Networks (LAN): A LAN is a collection of hosts linked to a single network through wired or wireless connections. However, LANs are restricted to buildings, organizations, or campuses. Typically, a few leased lines connected to the Internet provide web access to the whole organization or a campus; the lines are further redistributed to multiple hosts within the LAN enabling hosts. The hosts are much more in number than the actual direct lines to the Internet to access the web from within the organization. This also allows the organization to define various access control policies for web access within its hierarchy. Typically, the present-day data access rates within the LANs range from **100 Mbps to 1000 Mbps**, with very high fault-tolerance levels. Commonly used network components in a LAN are **servers, hubs, routers**, switches, terminals, and computers.

(iii) Metropolitan Area Networks (MAN): The reachability of a MAN lies between that of a LAN and a WAN. Typically, MANs connect various organizations or buildings within a given geographic location or city. An excellent example of a MAN is an Internet service provider (ISP) supplying Internet connectivity to various organizations within a city. As MANs are costly, they may not be owned by individuals or even single organizations. Typical networking devices/ components in MANs are modems and cables. MANs tend to have **moderate fault tolerance levels**.

(iv) Wide Area Networks (WAN): WANs typically connect diverse geographic locations. However, they are **restricted within the boundaries of a state or country**. The data rate of WANs is in the order of a fraction of LAN's data rate. Typically, WANs connecting two LANs or MANs may use public switched telephone networks **(PSTNs)** or **satellite-based links**. Due to the long transmission ranges, WANs tend to have more errors and noise during transmission and are very costly to maintain. The **fault tolerance of WANs are generally low**.

Protocols: A protocol is a set of rules that govern data communications. A protocol defines what is communicated, how it is communicated, and when it is communicated. The key elements of a protocol are syntax, semantics, and timing.

i) Syntax. The term syntax refers to the structure or format of the data, meaning the order in which they are presented. For example, a simple protocol might expect the first 8 bits of data to be the address of the sender, the second 8 bits to be the address of the receiver, and the rest of the stream to be the message itself.

ii) Semantics. The word semantics refers to the meaning of each section of bits. How is a particular pattern to be interpreted, and what action is to be taken based on that interpretation? For example, does an address identify the route to be taken or the final destination of the message?

iii) Timing. The term timing refers to two characteristics: when data should be sent and how fast they can be sent. For example, if a sender produces data at 100 Mbps but the receiver can process data at only 1 Mbps, the transmission will overload the receiver and some data will be lost.

Standards: Standards provide guidelines to manufacturers, vendors, government agencies, and other service providers to ensure the kind of interconnectivity necessary in today's marketplace and in international communications.

Standards Organizations: Standards are developed through the cooperation of standards creation committees, forums, and government regulatory agencies.

International Organization for Standardization (ISO): The ISO is a multinational body whose membership is drawn mainly from the standards creation committees of various governments throughout the world. The ISO is active in developing cooperation in the realms of scientific, technological, and economic activity.

International Telecommunication Union-Telecommunication Standards Sector (ITU-T):

By the early 1970s, a number of countries were defining national standards for telecommunications, but there was still little international compatibility. The United Nations responded by forming, as part of its International Telecommunication Union (ITU), a committee, the Consultative Committee for International Telegraphy and Telephony (CCITT). This committee was devoted to the research and establishment of standards for telecommunications in general and for phone and data systems in particular. On March 1, 1993, the name of this committee was changed to the International Telecommunication Union- Telecommunication Standards Sector (ITU-T).

American National Standards Institute (ANSI): Despite its name, the American National Standards Institute is a completely private, nonprofit corporation not affiliated with the U.S. federal government. However, all ANSI activities are undertaken with the welfare of the United States and its citizens occupying primary importance.

Institute of Electrical and Electronics Engineers (IEEE): The Institute of Electrical and Electronics Engineers is the largest professional engineering society in the world. International in scope, it aims to advance theory, creativity, and product quality in the fields of electrical engineering, electronics, and radio as well as in all related branches of engineering. As one of its goals, the IEEE oversees the development and adoption of international standards for computing and communications.

Electronic Industries Association (EIA): Aligned with ANSI, the Electronic Industries Association is a nonprofit organization devoted to the promotion of electronics manufacturing concerns. Its activities include public awareness education and lobbying efforts in addition to standards development. In the field of information technology, the EIA has made significant contributions by defining physical connection interfaces and electronic signaling specifications for data communication.

THE OSI MODEL: An ISO standard that covers all aspects of network communications is the Open Systems Interconnection model**. An open system** is a set of protocols that allows any two different systems to communicate regardless of their underlying architecture. The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software. The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust, and interoperable. The OSI model is a layered framework for the design of network systems that allows communication between all types of computer systems.

Physical Layer

The physical layer coordinates the functions required to carry a bit stream over a physical medium. It deals with the mechanical and electrical specifications of the interface and transmission medium. It also defines the procedures and functions that physical devices and interfaces have to perform for transmission to Occur.

The physical layer is also concerned with the following:

i) **Physical characteristics of interfaces and medium.** The physical layer defines the characteristics of the interface between the devices and the transmission medium. It also defines the type of transmission medium.

ii) **Representation of bits**. The physical layer data consists of a stream of bits (sequence of Os or 1s) with no interpretation. To be transmitted, bits must be encoded into signals-electrical or optical. The physical layer defines the type of encoding (how Os and Is are changed to signals).

iii) Data rate. The transmission rate-the number of bits sent each second-is also defined by the physical layer. In other words, the physical layer defines the duration of a bit, which is how long it lasts.

iv) Synchronization of bits. The sender and receiver not only must use the same bit rate but also must be synchronized at the bit level. In other words, the sender and the receiver clocks must be synchronized.

v) Line configuration. The physical layer is concerned with the connection of devices to the media. In a point-to-point configuration, two devices are connected through a dedicated link. In a multipoint configuration, a link is shared among several devices.

vi) Physical topology. The physical topology defines how devices are connected to make a network. Devices can be connected by using a mesh topology (every device is connected to every other device), a star topology (devices are connected through a central device), a ring topology (each device is connected to the next, forming a ring), a bus topology (every device is on a common link), or a hybrid topology (this is a combination of two or more topologies).

vii) Transmission mode. The physical layer also defines the direction of transmission between two devices: simplex, half-duplex, or full-duplex. In simplex mode, only one device can send; the other can only receive. The simplex mode is a one-way communication. In the half-duplex mode, two devices can send and receive, but not at the same time. In a full-duplex (or simply duplex) mode, two devices can send and receive at the same time

Data Link Layer

The data link layer transforms the physical layer, a raw transmission facility, to a reliable link. It makes the physical layer appear error-free to the upper layer (network layer).

Other responsibilities of the data link layer include the following:

i)**Framing.** The data link layer divides the stream of bits received from the network layer into manageable data units called frames.

ii) **Physical addressing.** If frames are to be distributed to different systems on the network, the data link layer adds a header to the frame to define the sender and/or receiver of the frame. If the frame is intended for a system outside the sender's network, the receiver address is the address of the device that connects the network to the next one.
iii) Flow control. If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.
iv)Error control. The data link layer adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged or lost frames. It also uses a mechanism to recognize duplicate frames. Error control is normally achieved through a trailer added to the end of the frame.
v)Access control. When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time.

Network Layer:
The network layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links). Whereas the data link layer oversees the delivery of the packet between two systems on the same network (links), the network layer ensures that each packet gets from its point of origin to its final destination.
If two systems are connected to the same link, there is usually no need for a network layer. However, if the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery.

Other responsibilities of the network layer include the following:
i)Logical addressing. The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems. The network layer adds a header to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver.
ii) **Routing.** When independent networks or links are connected to create intemetworks (network of networks) or a large network, the connecting devices (called routers or switches) route or switch the packets to their final destination. One of the functions of the network layer is to provide this mechanism.

Transport Layer:
The transport layer is responsible for process-to-process delivery of the entire message. A process is an application program running on a host. Whereas the network layer oversees source-to-destination delivery of individual packets, it does not recognize any relationship between those packets. It treats each one independently, as though each piece belonged to a separate message, whether or not it does. The transport layer, on the other hand, ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level.

Other responsibilities of the transport layer include the following:
i) Service-point addressing. Computers often run several programs at the same time. For this reason, source-to-destination delivery means delivery not only from one computer to the next but also from a specific process (running program) on one computer to a specific process (running program) on the other. The transport layer header must therefore include a type of address called a service-point address (or port address). The network layer gets each packet to the correct computer; the transport layer gets the entire message to the correct process on that computer.
ii) Segmentation and reassembly. A message is divided into transmittable segments, with each segment containing a sequence number. These numbers enable the transport layer to reassemble

the message correctly upon arriving at the destination and to identify and replace packets that were lost in transmission.

iii) Connection control. The transport layer can be either connectionless or connection- oriented. A connectionless transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine. A connection-oriented transport layer makes a connection with the transport layer at the destination machine first before delivering the packets. After all the data are transferred, the connection is terminated.

iii) Flow control. Like the data link layer, the transport layer is responsible for flow control. However, flow control at this layer is performed end to end rather than across a single link.

iv) Error control. Like the data link layer, the transport layer is responsible for error control. However, error control at this layer is performed process-to-process rather than across a single link. The sending transport layer makes sure that the entire message arrives at the receiving transport layer without error (damage, loss, or duplication). Error correction is usually achieved through retransmission.

Session Layer:

The services provided by the first three layers (physical, data link, and network) are not sufficient for some processes. The session layer is the network dialog controller. It establishes, maintains, and synchronizes the interaction among communicating systems.

Specific responsibilities of the session layer include the following:

i) Dialog control. The session layer allows two systems to enter into a dialog. It allows the communication between two processes to take place in either half-duplex (one way at a time) or full-duplex (two ways at a time) mode.

ii) Synchronization. The session layer allows a process to add checkpoints, or synchronization points, to a stream of data. For example, if a system is sending a file of 2000 pages, it is advisable to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged independently. In this case, if a crash happens during the transmission of page 523, the only pages that need to be resent after system recovery are pages 501 to 523. Pages previous to 501 need not be resent.

Presentation Layer:

The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems

Specific responsibilities of the presentation layer include the following:

i) Translation. The processes (running programs) in two systems are usually exchanging information in the form of character strings, numbers, and so on. The information must be changed to bit streams before being transmitted. Because different computers use different encoding systems, the presentation layer is responsible for interoperability between these different encoding methods. The presentation layer at the sender changes the information from its sender-dependent format into a common format. The presentation layer at the receiving machine changes the common format into its receiver-dependent format.

ii)Encryption. To carry sensitive information, a system must be able to ensure privacy. Encryption means that the sender transforms the original information to another form and sends the resulting message out over the network. Decryption reverses the original process to transform the message back to its original form.

ii) Compression. Data compression reduces the number of bits contained in the information. Data compression becomes particularly important in the transmission of multimedia such as text, audio, and video.