UNIT 1

INTRODUCTION OF IOT

IoTcomprisesthingsthathaveuniqueidentitiesandareconnected tointernet.By2020therewill be a total of 50 billion devices /things connected to internet. IoT is not limited to just connecting things to the internet but also allow things to communicate and exchange data.

Definition:

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

Characteristics:

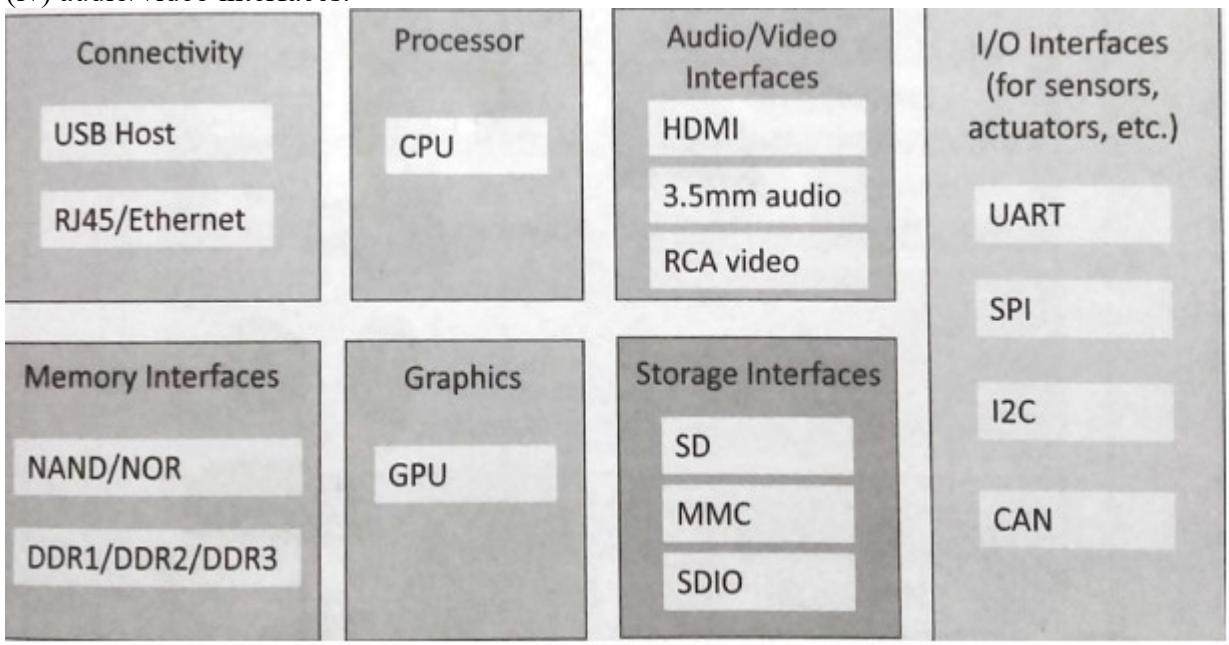
1) **Dynamic & Self Adapting**: IoT devices and systems may have the capability to dynamically adapt with the changing contexts and take actions based on their operating conditions, user_s context or sensed environment.
   Eg:thesurveillancesystemisadaptingitselfbasedoncontextandchangingconditions.
2) **Self-Configuring:**allowingalargenumberofdevicestoworktogethertoprovidecertain functionality.
3) **Inter Operable Communication Protocols:** support a number of interoperable communication protocols and can communicate with other devices and also with infrastructure.
4) **UniqueIdentity:**EachIoTdevicehasauniqueidentityandauniqueidentifier(IP address).
5) **IntegratedintoInformationNetwork:**thatallowthemtocommunicateandexchange data with other devices and systems.

ApplicationsofIoT:

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

PHYSICAL DESIGN OF IOT

Things in IoT: The thing in IoT refers to IoT devices which have unique identities and perform remote sensing, actuating and monitoring capabilities. IoT devices can exchange data with other connected devices applications. It collects data from other devices and process data either locally or remotely. An IoT device may consist of several interfaces for communication to other devices both wired and wireless. These includes (i) I/O interfaces for sensors, (ii) Interfaces for internet connectivity (iii) memory and storage interfaces and (iv) audio/video interfaces.

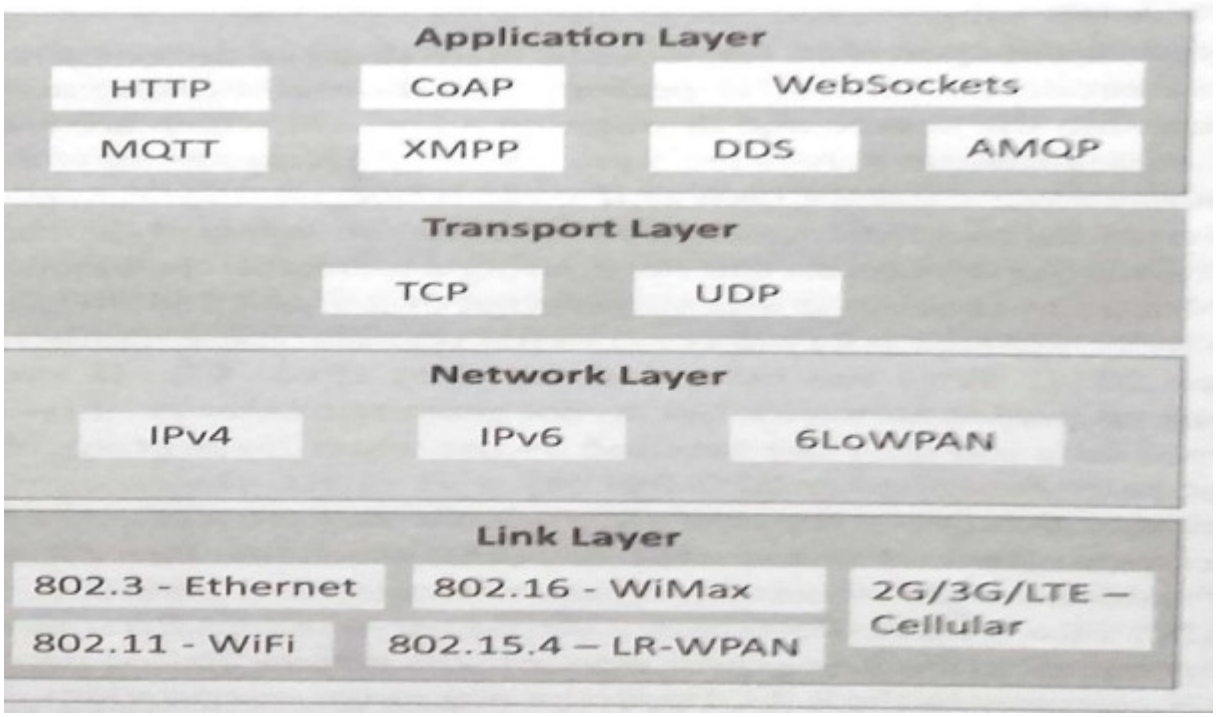


IoT Protocols:

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

Protocols:

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



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

1. IPv4: Internet Protocol version4 is used to identify the devices on a n/w using a hierarchical addressing scheme. 32 bit address. Allows total of 2**32addresses.
2. IPv6: Internet Protocol version6 uses 128 bit address scheme and allows 2**128 addresses.
3. 6LOWPAN:(IPv6overLowpowerWirelessPersonalAreaNetwork)operates in 2.4 GHz frequency range and data transfer 250 kb/s.

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

Protocols:

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

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

Protocols:

1.HTTP: Hyper Text Transfer Protocol that forms foundation of WWW. Follow request response model Stateless protocol.

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

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

4.MQTT: Message Queue Telemetry Transport is light weight messaging protocol based on publish-subscribe model. Uses client server architecture. Well suited for constrained environment.
5XMPP: Extensible Message and Presence Protocol for real time communication and streaming XML data

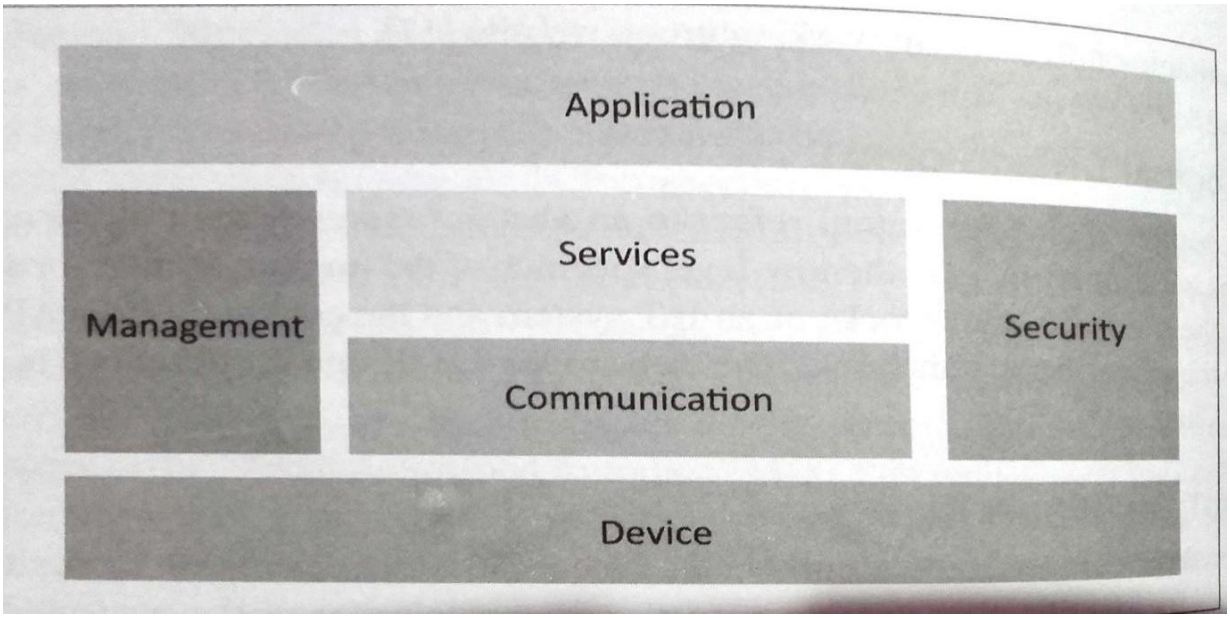
between network entities. Support client-server and server-server communication.
6.DDS: Data Distribution Service is data centric middleware standards for device-to-device or machine-to-machine communication. Uses publish-subscribe model.
7. AMQP: Advanced Message Queuing Protocol is open application layer protocol for business messaging. Supports both point-to-point and publish-subscribe model

LOGICAL DESIGN OF IOT

Referstoanabstractrepresentofentitiesandprocesseswithout goingintothelowlevel specifies of implementation

1) IoTFunctionalBlocks2)IoTCommunicationModels3)IoTComm.APIs

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

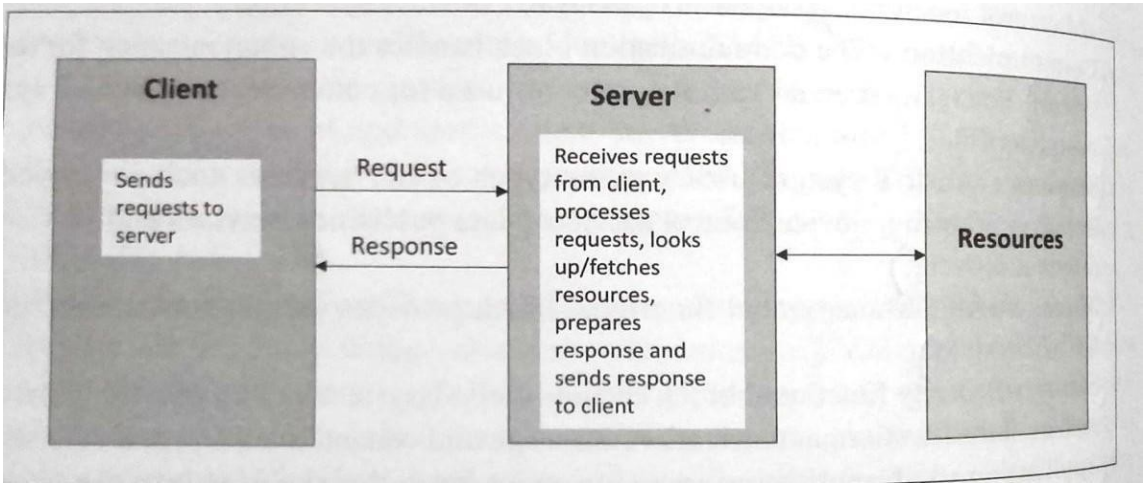


a. **Device:**AnIoTsystemcomprisesofdevicesthatprovidesensing,actuation,monitoring and control functions.
b. **Communication:**handlesthecommunicationforIoT system.
c. **Services:**fordevicemonitoring,devicecontrolservices,datapublishingservicesand services for device discovery.
d. **Management:**ProvidesvariousfunctionstogoverntheIoTsystem.
e. **Security:**Secures IoTsystemandpriorityfunctionssuchasauthentication, authorization,messageandcontextintegrityanddata security.
f. **Application:**IoTapplicationprovideaninterfacethattheuserscanusetocontroland monitor various aspects of IoT system.

2) IoTCommunicationModels:

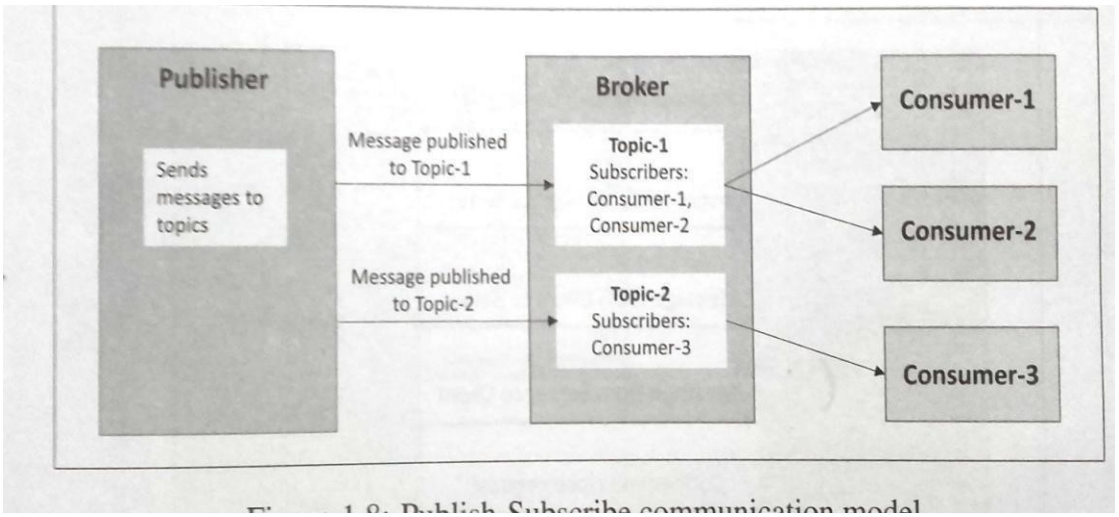
1) Request-Response2)Publish-Subscribe3)Push-Pull4)Exclusive-Pair

1) Request-ResponseModel:



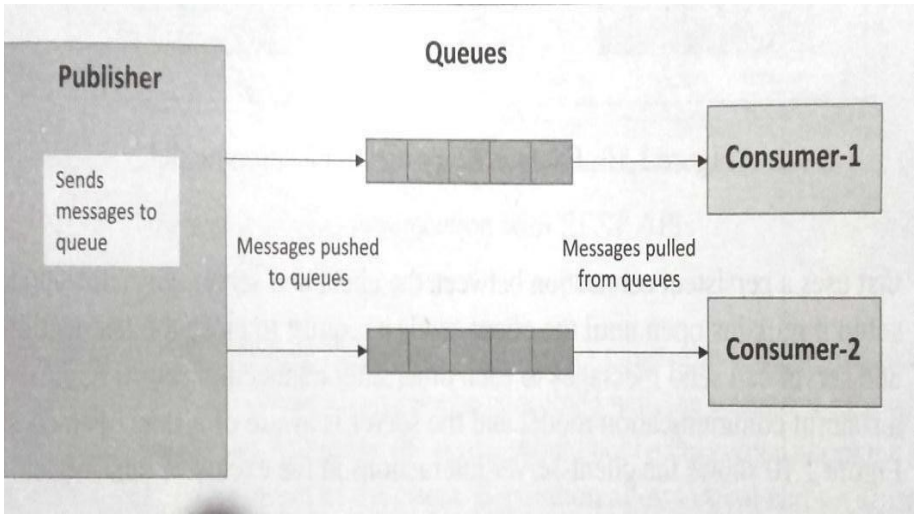
Inwhichtheclientsendsrequesttotheserverandtheserverrepliestorequests.Isa stateless communication model and each request-response pair is independent of others.

2) Publish-SubscribeModel:

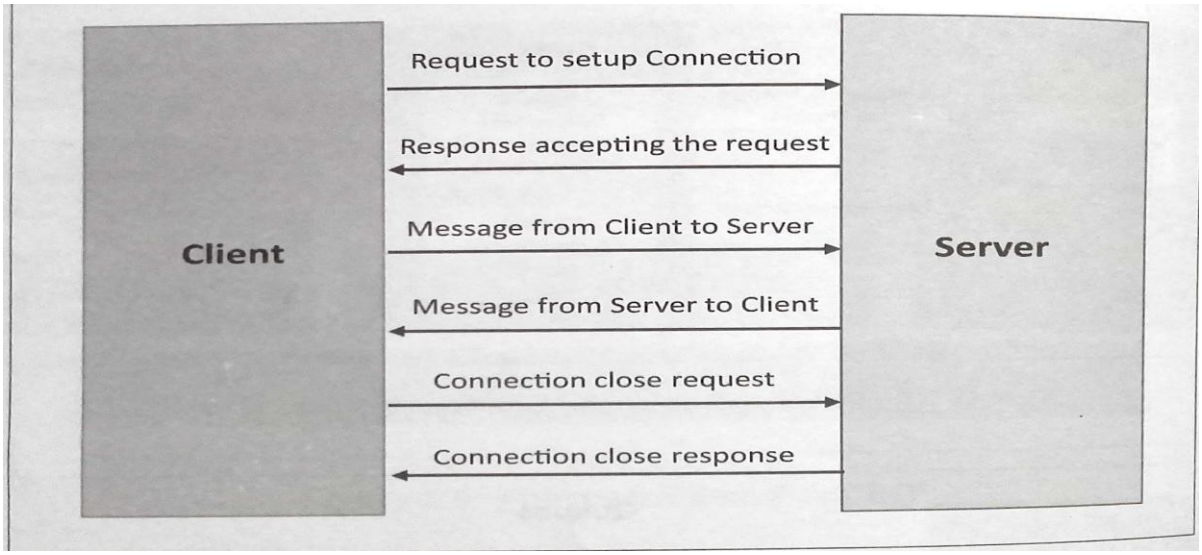


Involves publishers, brokers and consumers. Publishers are source of data. Publishers send data to the topics which are managed by the broker. Publishers are not aware of the consumers. Consumers subscribe to the topics which are managed by the broker. When the broker receives data for a topic from the publisher, it sends the data to all the subscribed consumers.

3) **Push-Pull Model:** in which data producers push data to queues and consumers pull data from the queues. Producers do not need to aware of the consumers. Queues help in decoupling the message between the producers and consumers.



4) **Exclusive Pair:** is bi-directional, fully duplex communication model that uses apersistent connection between the client and server. Once connection is set up it remains open until the client send a request to close the connection. Is a stateful communication model and server is aware of all the open connections.

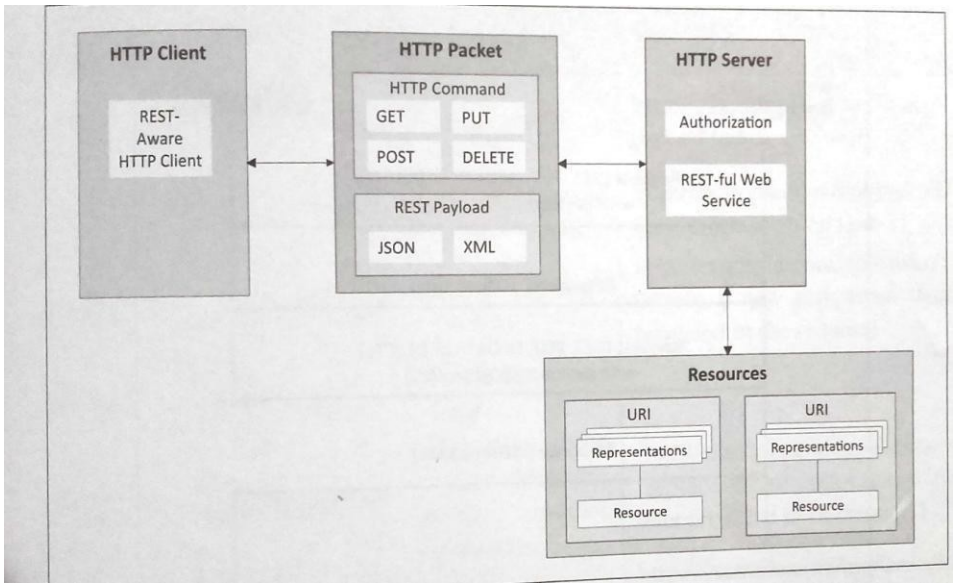


3)IoTCommunicationAPIs:

a) RESTbasedcommunicationAPIs(Request-ResponseBasedModel)
b) WebSocketbasedCommunicationAPIs(ExclusivePair BasedModel)

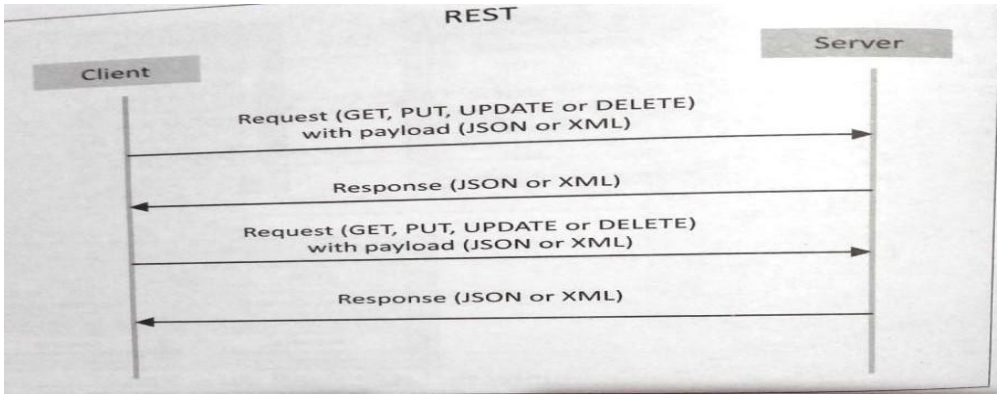
a) REST based communication APIs: Representational State Transfer(REST) is a set of architectural principles by which we can design web services and web APIs that focus on a system_s resources and have resource states are addressed and transferred.

TheRESTarchitecturalconstraints: Fig.showscommunicationbetweenclientserverwith REST APIs.



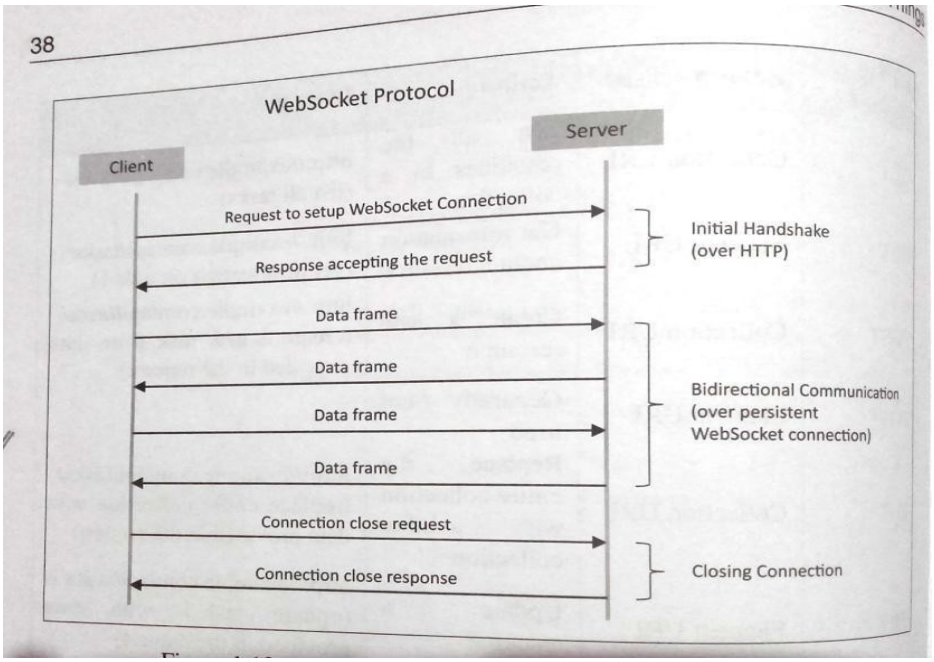
Client-Server: The principle behind client-server constraint is the separation of concerns. Separation allows client and server to be independently developed and updated.
Stateless: Each request from client to server must contain all the info. Necessary to understand the request, and cannot take advantage of any stored context on the server.
Cache-able: Cache constraint requires that the data within a response to a request be implicitly or explicitly labeled as cache-able or non-cacheable. If a response is cache-able, then a client cache is given the right to reuse that response data for later, equivalent requests.
Layered System: constraints the behavior of components such that each component cannot see beyond the immediate layer with which they are interacting.
User Interface: constraint requires that the method of communication between a client and a server must be uniform.
Code on Demand: Servers can provide executable code or scripts for clients to execute in their context. This constraint is the only one that is optional.

Request-ResponsemodelusedbyREST:



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

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



IoTEnablingTechnologies

IoT is enabled by several technologies including Wireless Sensor Networks, Cloud Computing, Big Data Analytics, Embedded Systems, Security Protocols and architectures, Communication Protocols, Web Services, Mobile internet and semantic search engines.

1) **Wireless Sensor Network(WSN):** Comprises of distributed devices with sensors which areusedtomonitortheenvironmentalandphysicalconditions.ZigBeeisoneofthemost popular wireless technologies used byWSNs.
   WSNsusedinIoTsystemsaredescribedasfollows:
   - WeatherMonitoringSystem:inwhichnodescollecttemp,humidityandother data, which is aggregated and analyzed.
   - Indoorairqualitymonitoringsystems:tocollectdataontheindoorairqualityand concentration of various gases.
   - SoilMoistureMonitoringSystems:tomonitorsoilmoistureatvariouslocations.
   - SurveillanceSystems:useWSNsforcollectingsurveillancedata(motiondata detection).
   - SmartGrids:useWSNsformonitoringgridsatvarious points.

   - Structural Health Monitoring Systems: Use WSNs to monitor the health of structures(building, bridges) by collecting vibrations from sensor nodes deployed at various points in the structure.

2) **CloudComputing:**Servicesareofferedtousersindifferentforms.
   - Infrastructure-as-a-service(IaaS):providesusers the abilityto provision computing and storage resources. These resources are provided to the users as a virtual machine instances and virtual storage.
   - Platform-as-a-Service(PaaS): provides users the ability to develop and deploy application in cloud using the development tools, APIs, software libraries and services provided by the cloud service provider.
   - Software-as-a-Service(SaaS):providestheuseracompletesoftwareapplicationor the user interface to the application itself.

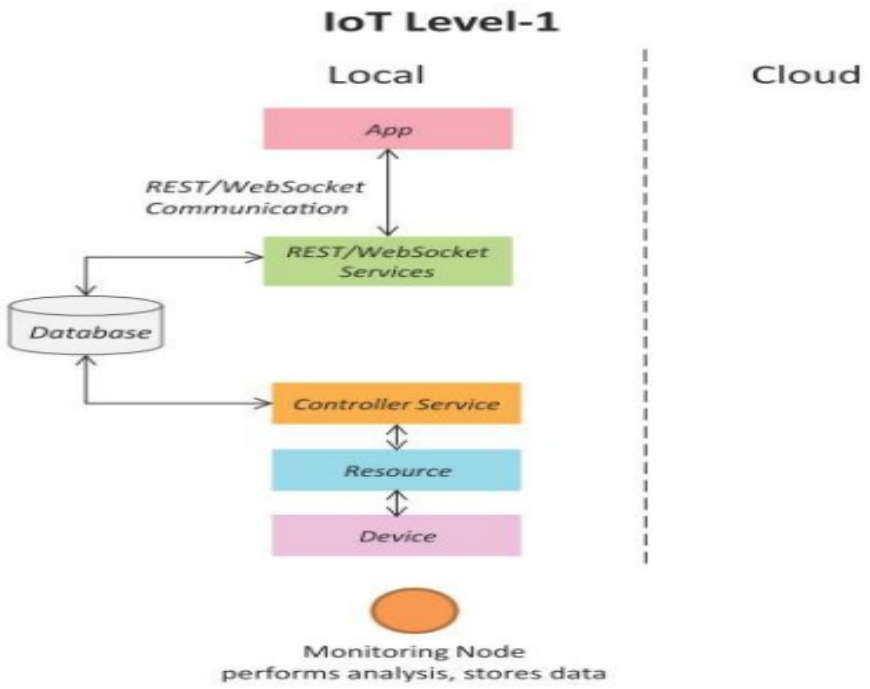
3) **BigDataAnalytics:**Someexamplesofbigdatageneratedby IoT are
   - SensordatageneratedbyIoTsystems.
   - Machinesensordatacollectedfromsensorsestablishedinindustria landenergy systems.
   - HealthandfitnessdatageneratedIoTdevices.
   - DatageneratedbyIoTsystemsforlocationandtrackingvehicles.
   - Data generated byretail inventorymonitoring systems.

4) **CommunicationProtocols:**formtheback-boneofIoTsystemsandenablenetwork connectivity and coupling to applications.
   - Allowdevicestoexchangedataover network.
   - Definetheexchangeformats,dataencodingaddressingschemesfo rdeviceand routing of packets from source to destination.
   - Itincludessequencecontrol,flowcontrolandretransmissionoflostpackets.

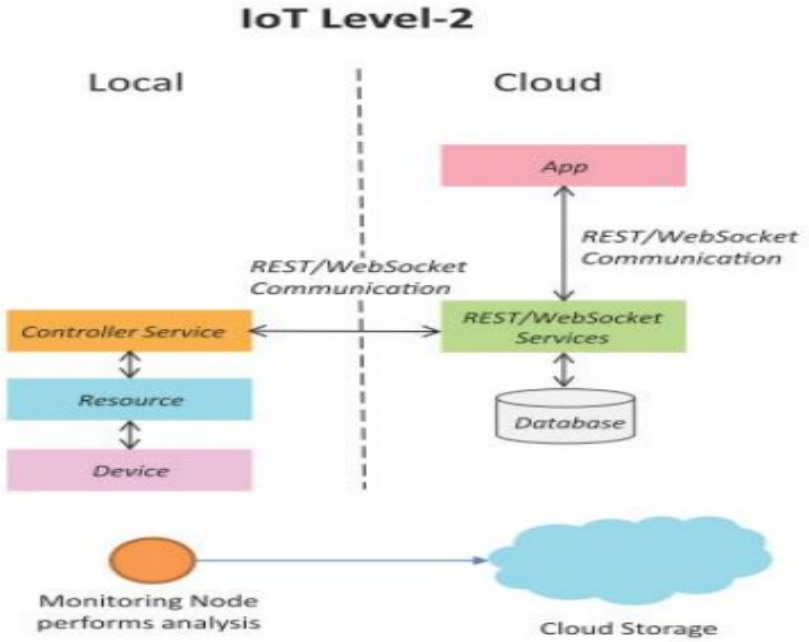
5) **Embedded Systems:** is a computer system that has computer hardware and software embedded to perform specific tasks. Embedded System range from low cost miniaturized devices such as digital watches to devices such as digital cameras, POS terminals, vending machines, appliances etc.,

IOT DEPLOYMENT LEVELS

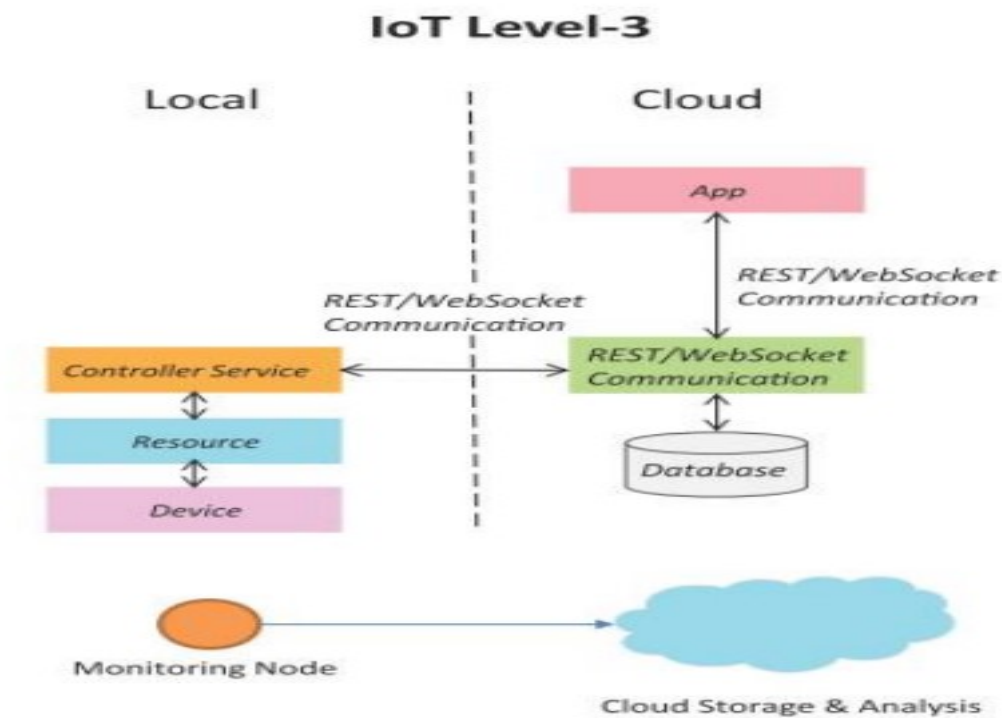
1) IoT Level1: System has a single node that performs sensing and/or actuation, stores data, Performs analysis and host the application as shown in fig. Suitable for modeling low cost and low complexity solutions where the data involved is not big and analysis requirement are not computationally intensive.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



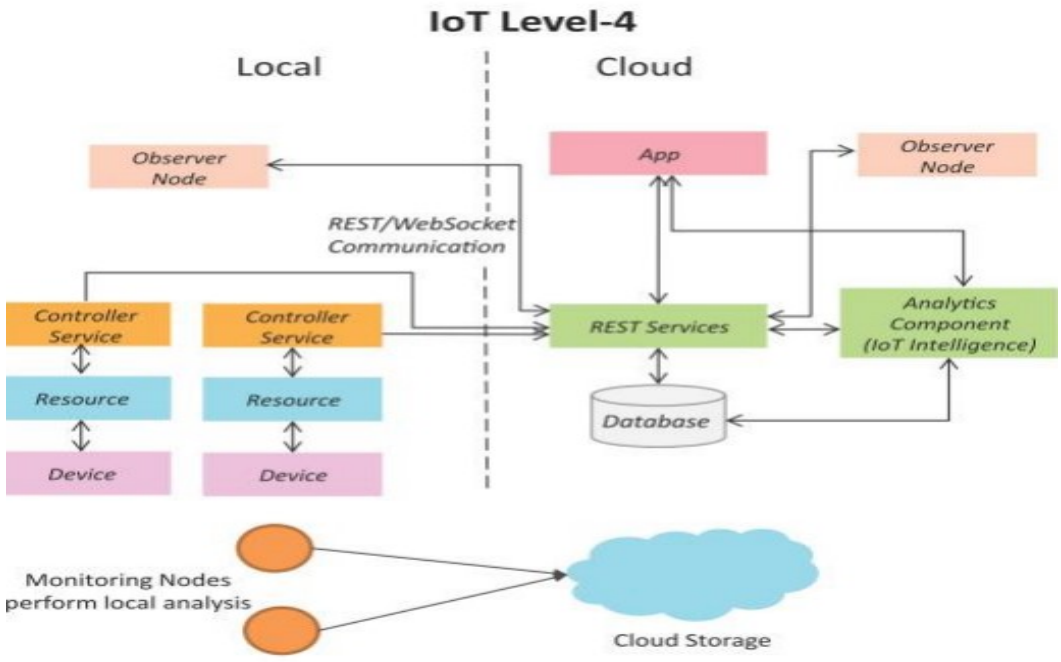
2) IoT Level2: has a single node that performs sensing and/or actuating and local analysis as shown in fig. Data is stored in cloud and application is usually cloud based. Level2 IoT systems are suitable for solutions where data are involved is big, however, the primary analysis requirement is not computationally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



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

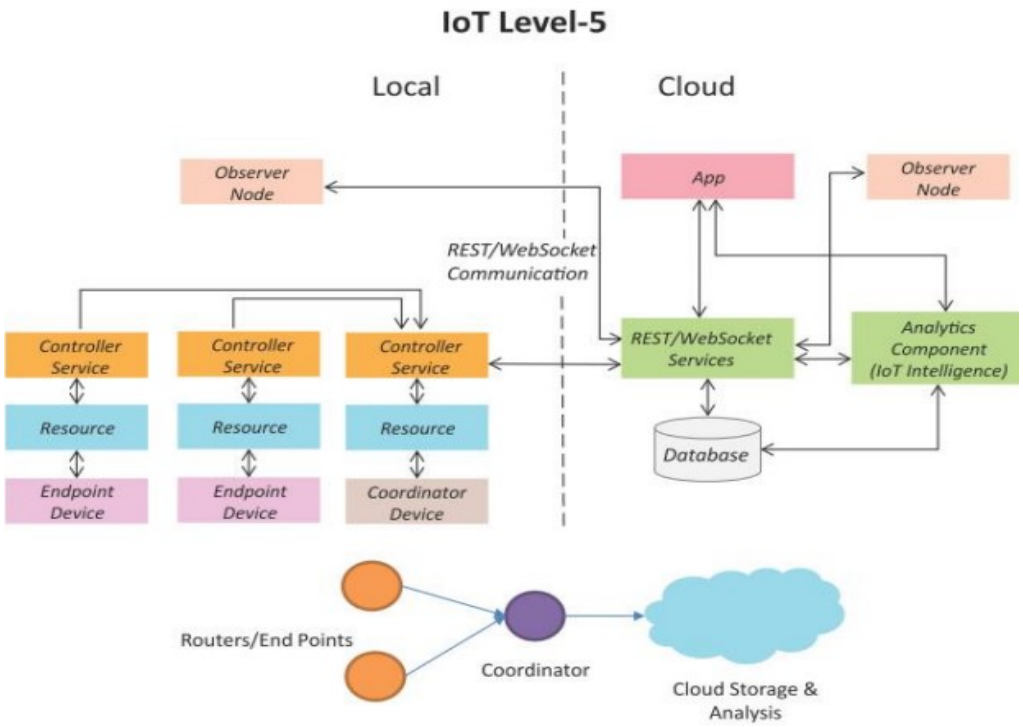


4) IoT Level4: System has multiple nodes that perform local analysis. Data is stored in the cloud and application is cloud based as shown in fig. Level4 contains local and cloud based observer nodes which can subscribe to and receive information collected in the cloud from IoT devices. An example of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



5) IoT Level5: System has multiple end nodes and one coordinator node as shown in fig. The end nodes that perform sensing and/or actuation. Coordinator node collects data from the end nodes and sends to the cloud.Dataisstoredandanalyzedinthecloudand application is cloud based. Level5

IoT systems are suitable for solution based on wireless sensor network, in which data involved is big and analysis requirements arecomputationally intensive. An example of Level5 system for Forest Fire Detection



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