UNIT-1

The Internet of Things : An over Oview of IOT, IOT Tech, behind IOTs Sources of IOT, M2M Communication, Examples of IOTs, Design Principles For Connected Devices.

Objectives :-

* Over view of IOT
* IOT Technology
* Sources of IOTs
* M2M Communication
* IOT Realtime Examples
* principles for Connected Devices.

1.1 Definition of IOT :-

It is a Concept which enables Communication between internetworking devices and applications, where by physical objects or 'things' Communicate through Internet.

The Concept of IOT began with things classified as identity Communication devices. RFID (Radio Frequency Identification devices) is an example of identity Communication devices. These are tagged to these devices for

their identification in future & can be tracked, Controlled and monitored using remote Computer.

Def 2:

The Internet is a vast global network of Connected servers, Computers, tablets & mobile that is governed by standard protocals for Connected system. It enables sending, receiving or Communication of information, Connectivity with remote Servers, Cloud and analytics platform.

Def 3:

The IOT is a Computing Concept that describes the idea ~~for~~ of every day physical objects being Connected to the internet and being able to identify themselves to other devices. The term is closely identified with RFID as the method of Communication include other sensor technologies, Wireless tech or QR (Quick Response) Codes.

What is QR Code:-

A machine-readable code Consisting of an array of black and white squares, typically used for storing URL's (or) other information for reading by the Camera on a smartphone.

Thing in English has number of uses and meanings. In a dictionary, thing is a word used to refer to a physical object, an action or idea, a situation or activity, in case when one does not wish to be precise.

IOT means a network of physical things (objects) sending, receiving or Communication information using the internet and network just as the Computers, tablets & mobiles, and thus enabling the monitoring, Coordinating or Controlling process accors the internet.

IOT Vision:-

IOT is a vision where things (wearable watches, alarm clocks, home devices, surrounding objects) become 'smart' & function like living entities by sending, Computing & Communicating through Embedded devices which interact with remote obj.

Example:-

Streetlights in a city can be made to fun. like living entities through sensing & Computing using tiny embedded devices that Communicate & interact

with a central Control-and-Command station through internet.

Assume that each light in a group of 32-streetlight comprises a sensing, Computing & Communication circuit. Each group Connects to a group-Controller through bluetooth (or) ZigBee. Each Controller Connects to Central Command-and-Control station through Internet.

The station receives info about each ligh in each group in a city. The info. received related to functioning of 32-lights, the faulty lights or absence of traffic in group vicinity and about the ambient conditions, whether cloudy, dark or normal daylight.

The station remotely programs the group controllers, which automatically take an appropriate action per the conditions of trafic & light levels.

So each group in the city is controlled by the 'Internet of Streetlights'.

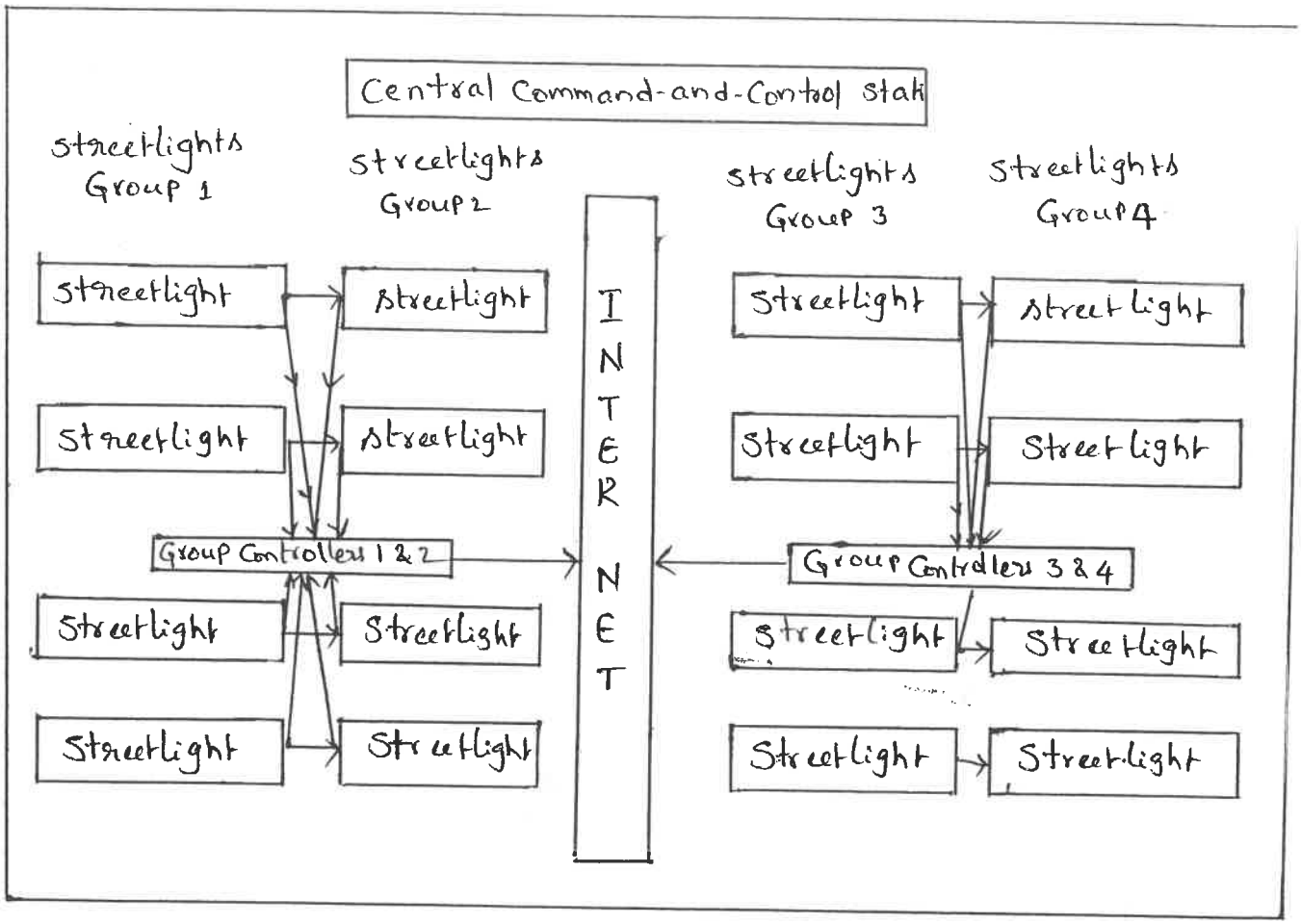


Fig:- Use of IOT Concept for streetlights in a city.

SMART & HYPERCONNECTED DEVICES :

HyperConnectivity means use of multiple systems and ~~Connected~~ devices to remain Constantly Connected to social networks and streams of information. Smart devices are devices with Computing and Communication capabilities that Can Constantly Connect to networks.

For example, a city network of streetlights which Constantly Connects to Controlling station shown in above

Figure

An RFID is an example of HYPERConnected devices. An RFID or smart label is a tagged to all Consignments. This many Consignments sent from a place can be Constantly tracked.

The below digram shows general framework for IOT using smart & hyperConnected devices, edge Computing & applications.

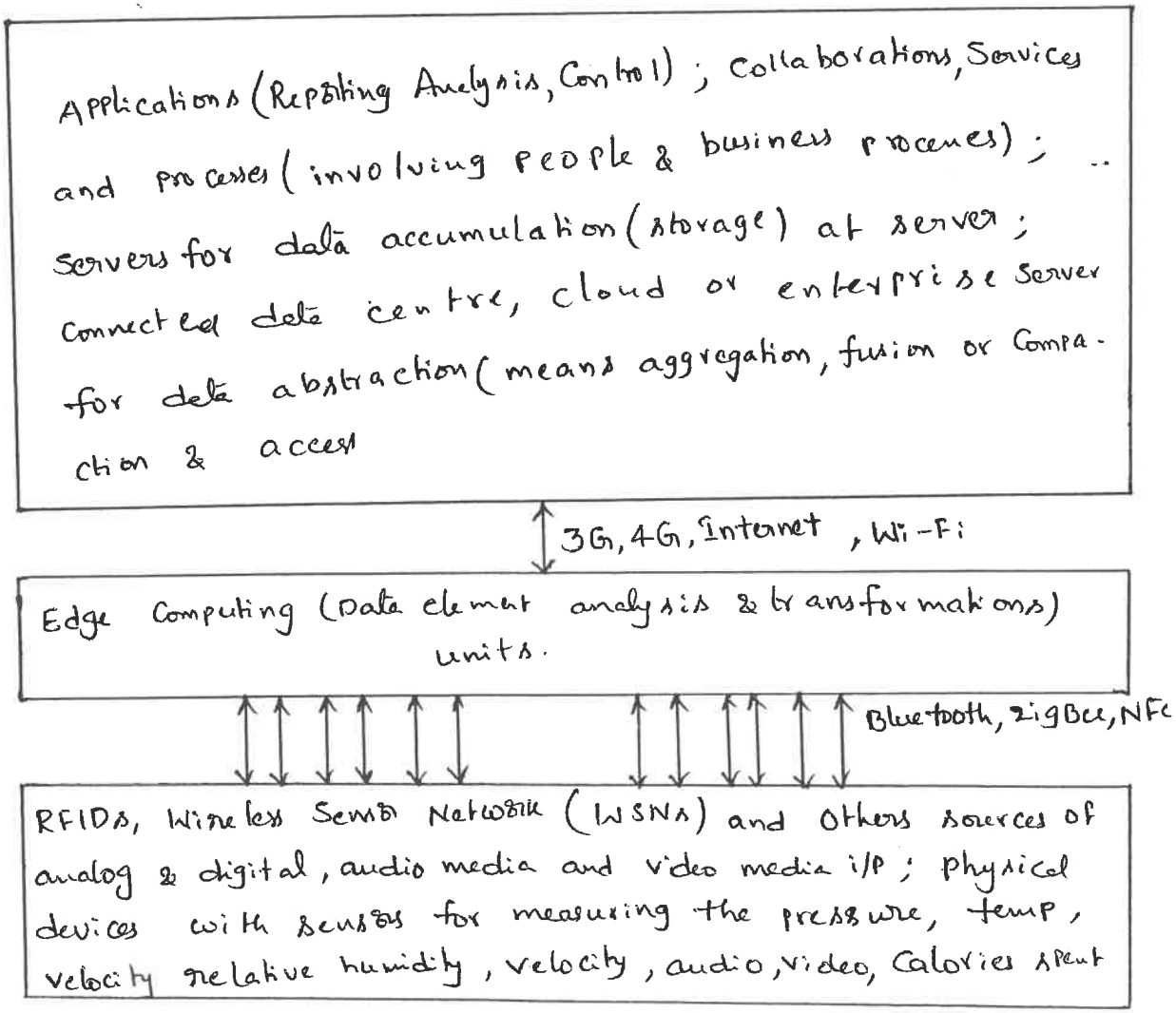


Fig: A general Frame work for IoT using smart and hyper connected devices, edge Computing and applications.

1.2 IoT Conceptual Framework :-

Ex 1 :

A Single Object Communicating with a central Server for Acquiring data . The following equation describes a simple Conceptual framework of IOT .

> Physical Object + Controller , Sensor and Actuators + Internet = IOT . ⟶ (1)

The above equation Conceptually describes the internet of Objects as Consisting of object, sensor and actuators, and internet for Connectivity to a web Service & mobile service providers .

IOT Consists of an internet work of devices & physical objects where in a no. of objects can gather the data at remote locations & Communicate to managing, acquiring, organising & analysing data in processes & Services

Ex 2 :

It showed the no. of streetlights Communicating data to group Controller which Connects Central Server using Internet .

General Framework Consists of no. of devices Communicating data to a data centre (or) Cloud Server . IoT

Framework of IoT used in no. of applications as well as an enterprise & business processes, more Complex than the one represented by Equation 1.

The below equation Conceptually represents the Actions and Communication of data at sensive level in IoT Consisting of interneworked devices & objects.

> Gather + Enrich + Stream + Manage + Acquire + Organise and Analyse = IoT with Connectivity to data center, enterprise or cloud Server. → (2)

⇒ The above equation is an IoT Conceptual Frame work for enterprise process and Services based on IoT Architecture.

The below equation is an alternative Conceptual representation for Complex system. It is an IBM IoT Conceptual Framework. The below equ. shows the action and Communication of data.

> Gather + Consolidate + Connect + Assemble + Manage and Analyse = IoT with Connectivity to Cloud Services
> ↳ (3)

It represent Conceptual Framework for IoT using cloud-platform based on processes & Services.

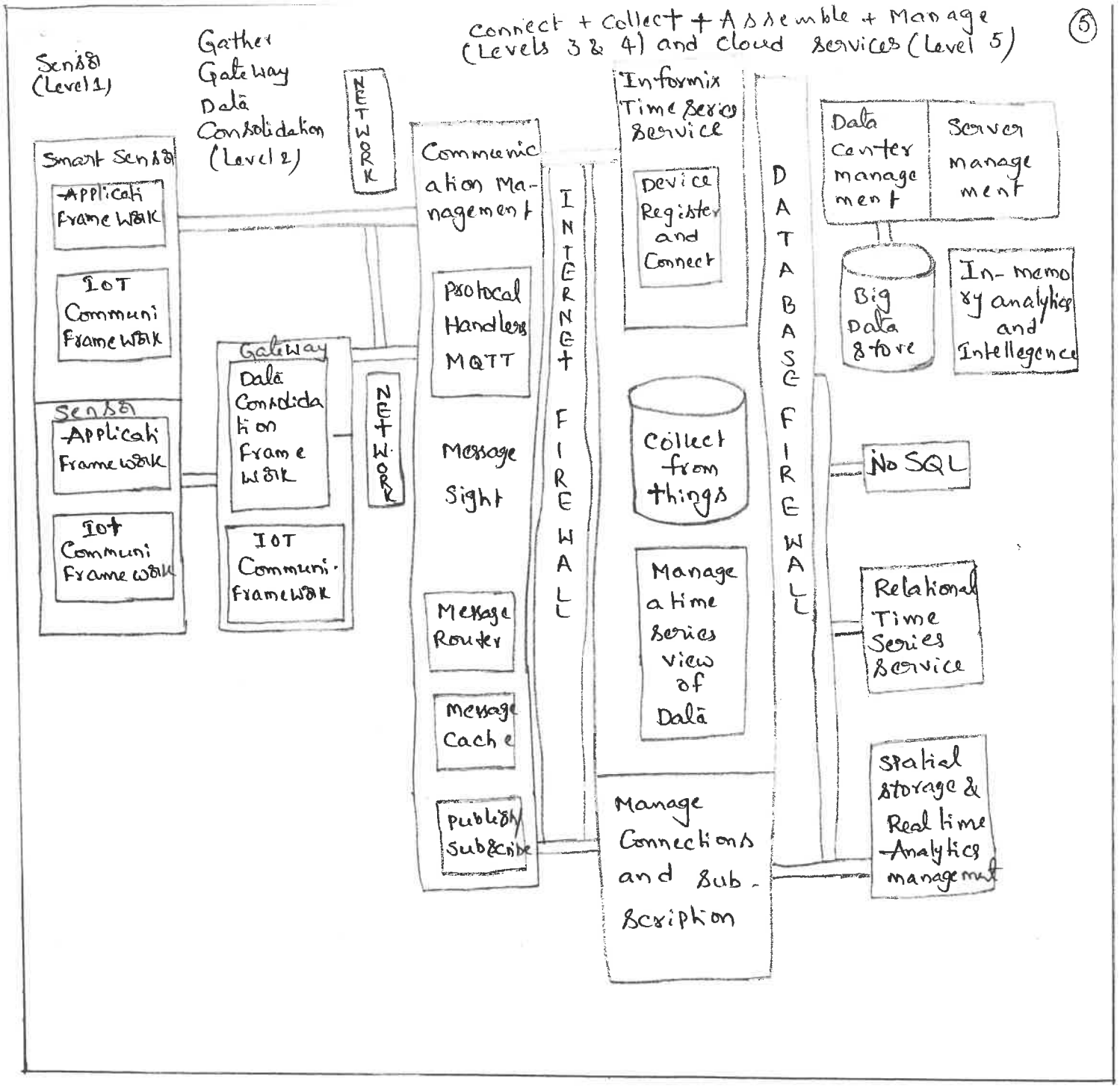


Fig: IBM Iot Conceptual Frame Work.

IoT ARCHITECTURAL VIEW:-

An IoT system has multiple levels (Like equations 1.1 to 1.3). These levels are known tiers. A model enables Conceptualisation of a framework. A reference model can be used to depict building blocks, successive interactions and integration. An example is CISCO's presentation of a reference model Comprising seven levels.

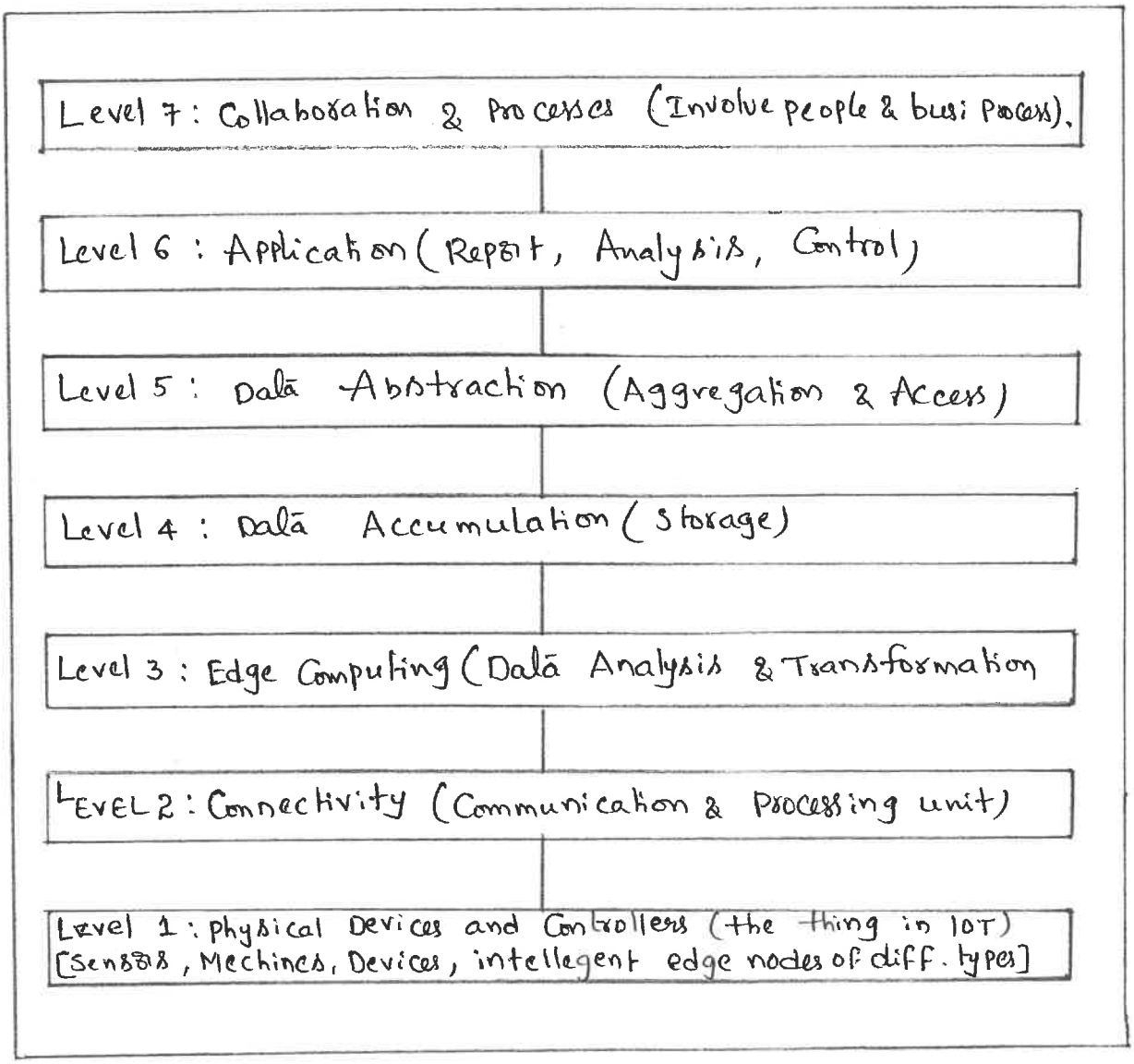


Fig:- An IoT Reference Model Suggested by CISCO.

Architecture Features :-

* The architecture serves as a reference in applications of IOT in services & business processes.
* A set of sensors which are smart, capture the data, perform necessary data element analysis & transformation as per device application frame work and connect directly to a Communication Manager.
* A set of sensor Circuits is Connected to a gateway possessing separate data capturing, gathering, Computing & Comm. Capabilities.
* Communication subsystem has functionalities for device identity database, device identity management & Access management.
* Communication - mangmt sub system Consists of protocol handlers, message routers & message cache.
* Data routers from the gateway through internet & data centre to application server (or) Enterprise server which acquires data.
* Organisation and analysis sub system enable services, business processes, enterprize integration & Complex processes.

A number of models (Cisco, purdue and other model) have been proposed at SWG (Sub working Group) Teleconfonence of Dec 2014.

1.3 Technology Behind IoT :-

The following entities provide a diverse tech, envisonment and core example of technologies, which core involved in IoT.

- Hardware (Arduino Rasberry Pi, Intel Galileo, Intel Edison, ARM mBed, etc)
- IDE for developing device software, firmware & API's.
- protocals [RPL, CoAP, RESTful HTTP, MQTT, XMPP (extensible Messaging & presense protocol])]
- Communication (2G, 3G, 4G, Wifi, ZigBee, Bluetooth)
- Network Backbone (IPV4, IPV6, UDP & GLOWPAN)
- Software (RIOT OS, Eclipse IoT, Contiki OS)
- Internetwork cloud platforms/ Data Centers. (Sense, Thingo ThingWorx, Nimbits, TCS CUP).

- Machine learning algorithm & software. An example of machine learning software is GROK from Numenta Inc. that uses machine intelligence to analyse the streaming data from clouds & un cover anamalies, and GROA perform high level of automation for analysing streaming data.

The following 5 entities can be considered for the 5 levels behind an IOT system.

1. Device platform consisting of device h/w & s/w using micro controller, and s/w for device API's & web applications.
2. Connecting and networking enabling interneworking of devices and physical objects called things & enabling the internet connectivity to remote servers.
3. Server and web programing enabling web application and web services.
4. Cloud platform enabling storage, computing prototype & product development platform.
5. Online transactions processing, online analytical processing

data analytics, predictive analytics & knowledge discovery enabling wider applications of IoT.

1.3.1:

Server-end-Technology:-

IoT servers are applications servers, enterprise servers, cloud servers, data centres & D.B.

Server offer the following software Components:-

- online
- Device identification, identity Management
- Data Accuring, Aggregation, integration, organising and analysing
- use of web applications, services.

1.3.2:

Major Components of IoT:-

Physical object:- Embedded software into Hardware.

Hardware: Consists of micro Controller, firmware, sensors etc

Communication model: software Consists of device API's and device interface for Communication over the n/w & Communication Circuit.

4). Software :- It Actions on (software) messages, info. and Commands which the device receive & then output to actuators.

Sensors and Circuits :-

Sensors :-

Sensors are electronic devices that sense the physical Environments.

Sensors are 2 types :

The first type gives analog inputs to Control units, example are thermistor, pressure gauge & Hall sensor.

Second type gives Digital input to Control unit. example are touch sensor, proximity sensor, metal sensor.

Control unit :-

Most Commonly used Control unit in IoT Consists of Micro Controller unit (MCU). MCU integrated Chip (or) Core in a VLSI or SoC.

Ex: ATMega 32u4.

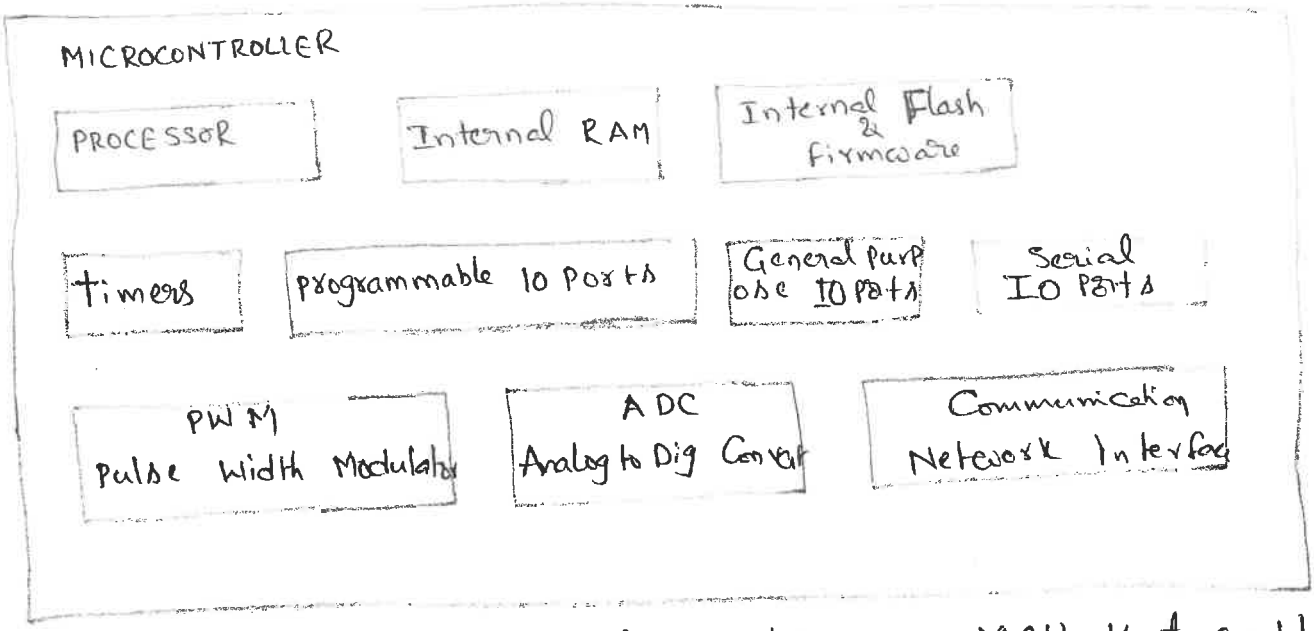


Fig: Various functional units in MCU that enable is an IOT device.

Communication Module:-

It Consist of protocol handlers, message queue & message cache. A device message-queue inserts the messages in the queue and delete the messages from the queue in FIFO Manner. A device message-Cache stores the received messages.

Representational state Transfer (REST) Archi. style can be used for HTTP Access by GET, POST, PUT and delete methods for resources and building Web services.

SOFTWARE :-

It Consists of 2 Components - S/w at the IOT devices.

Middleware :

OpenIoT is an open source middleware. It enables communication with sensors Clouds as well as cloud-based 'sensing as a service'. IoT sys is a Middleware which enables provisioning of Communication stack for smart device using IPV6, oBIX.

Operating System :-

Examples of OSs are RIOT, Raspbian, Alljoyn, spark & Contiki.

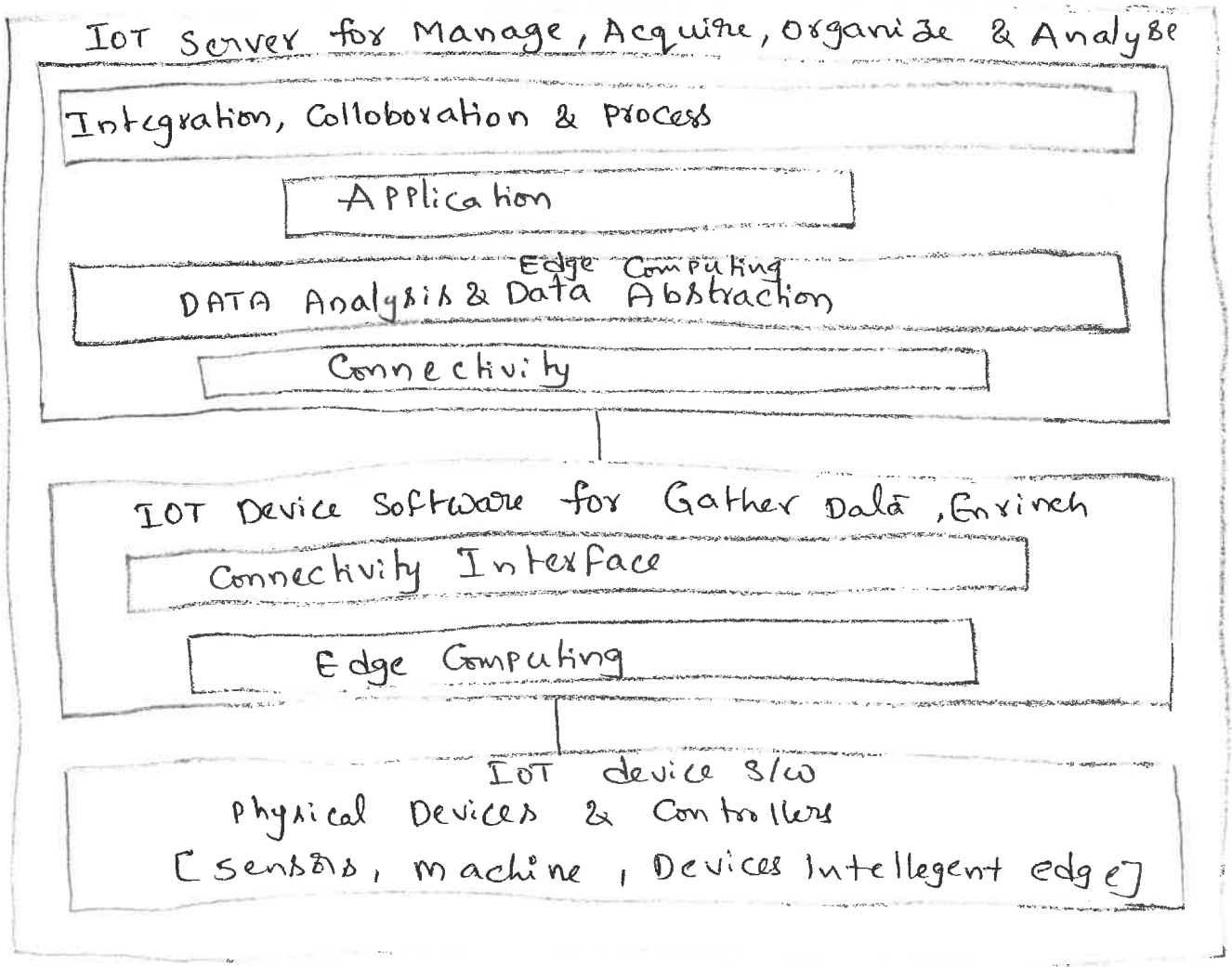


FIG: IoT S/w Components for device h/w.

1.3.3 : Development Tools & Open - Source Frame work for IoT Implementation;

Eclipse IoT provides open - source Imple. of standards such as MQTT CoAP, OMA-DM . and tools for working with Lua service.

Arduino development tools provide a set of software that includes an IDE. & (that includes an IDE) Arduino Prog. language for Hardware Specification.

Kinoma software plat form - Kinoma Create Kinoma studio development environment & Kinoma Plat form Runtime .

1.3.4 : APIs & Device Interfacing Components :-

Connectivity interface Consists of Communication API, device interfaces and processing units.

1.4 SOURCES OF IOT :-

Examples of hardware sources for IoT prototype development are Arduino Yun, Microduino, Beagle Board & RasWik.

Hardware prototype needs an IDE for development device software, firmware & API's.

1.4.1 : Popular IoT Development Boards :-

Arduino Yun :

This board uses micro Controller ATmega32u4 that supports Arduino and includes Wi-fi, Ethernet, USB port, Micro-SD card slots & 3-reset buttons. It also Contain with Atheros AR9331 that runs Linux.

Intel Galileo :-

It is a line of Arduino - Certified development boards. It is based on x86 Architecture.

Intel Edison :

It is a Computer module. It creates creation of prototype & development of Prototyping Projects & rapidly produces IoT & wearable Computing devices.

1.4.2 : Role of RFID & IoT Applications :-

Earlier IoT system were Internet-Connected RFID based system. RFID enables tracking and inventory control, identification in supply chain systems, Access to buildings and road tolls or secured store centre entries, & devices such as RFID-based temparature sensors.

1.4.3 : Wireless Sensor Networks (WSNs) :-

Sensors can be networked using wireless techn. & can co-operatively monitor physical (or) environmental conditions. Sensors acquire data from remote locations, which may not be easily accessible. Each wireless sensor also has communication abilities for which it uses a radio-frequency transceiver.

WSN Definition :-

Wireless sensor network (WSN) is defined as a network in which each sensor node connects wirelessly and capabilities of computations for data compactions.

1.5 M2M Communication :-

Machine-to-Machine (M2M) refers to the process of Communication of physical object (or) device at machine with others of the same type, Mostly for monitoring but also for Control purposes. Each Machine in an M2M system embeds a smart device. The device sense the data of machine & perform Computations & Communications functions.

1.5.1 : M2M IOT :

IOT Tech in industry involves the integration of Complex physical machinery M2M Communication with network sensors & uses Analytics, machine learning.

It closely related to IOT When smart devices or machines Collect data which transmitted via Internet to other devices or machines located remotely.

The difference between M2M & IoT is that M2M must deploy device to device and carry out coordination between, monitoring, controlling of devices & communicate with out the usage of Internet. But IoT deploys Internet, Server, internet protocal & Server (or) cloud end applications, services (or) processes.

M2M has many applications in fields such as industrial Automation, logistics, smart grid, smart cities, health & defence.

1.5.2 M2M Architecture :-

M2M Architecture consists of 3 domains.

1. M2M device domain
2. M2M network domain
3. M2M application domain.

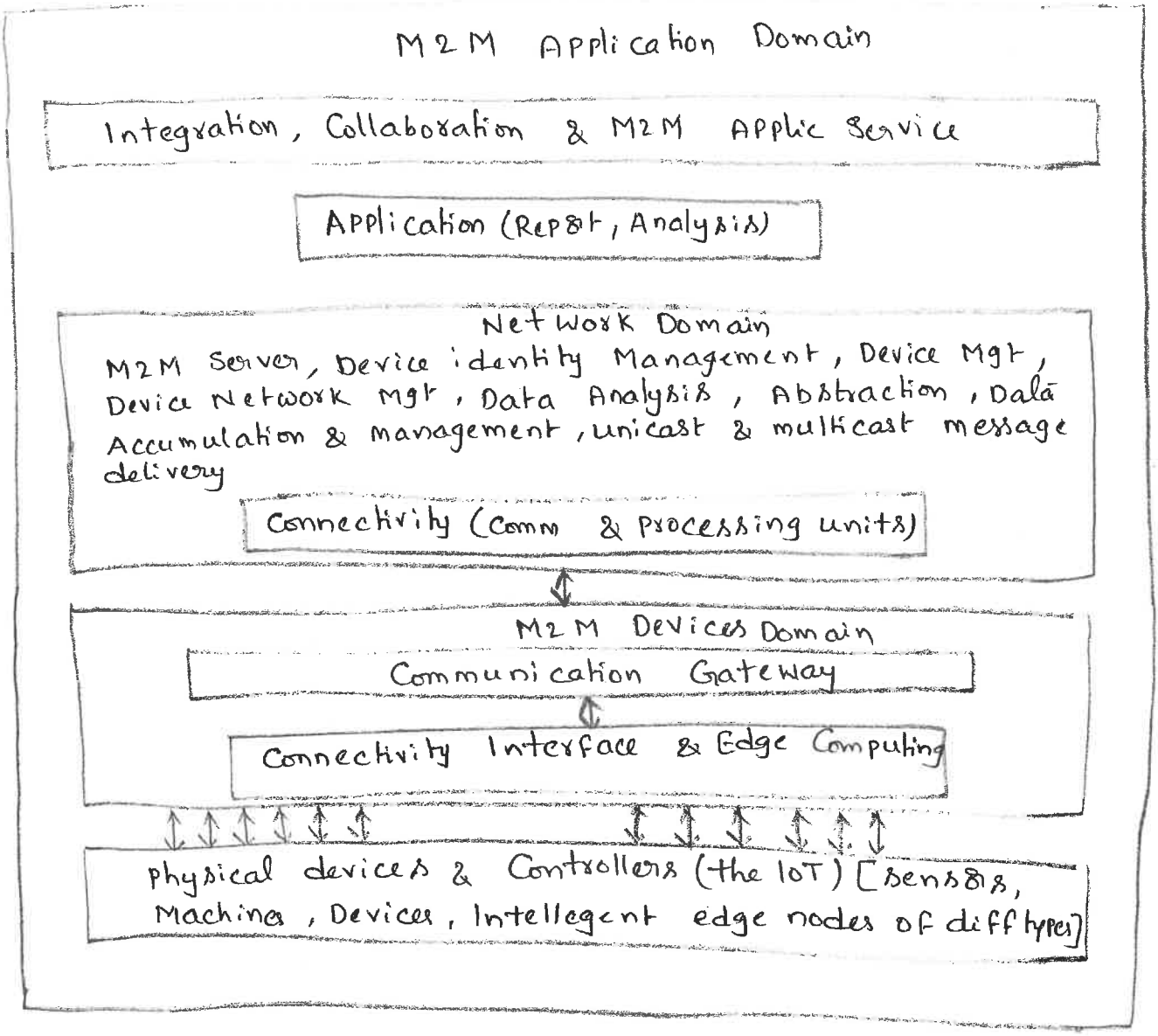


Fig : Three Domain of M2M Architecture.

1.5.3 : Software & Development tools :-

Examples of M2M S/W & development tools :

- Mango : It is an open source M2M based software. It supports multiple platforms, multiple protocals, Data bases, meta Points etc.

- Mainspring: Mainspring from M2MLabs is a development tool, & source framework for development M2M application. It enables:

★ Flexible modeling of devices

★ Communication between devices & applic

★ validation & Namalisation of data

★ Long-term data storage & retrival functions.

★ programming in java

★ Usage of no SQL Database

- Device Hive: It is an M2M Comm. framework.

It is an m2m platform & integration tool. it enables devices to Connect IoT. It includes web-based mgnt s/w that creates security-rules-based e-networks & monitoring devices.

- Open M2M protocols, Tools & Frame Work:-

These are the open protocols, tools & framework for M2M:

★ XMPP, MQTT - OASIS Standards & group & OMA LWM2M - OMA standard group for protocal.

* Various projects of Eclipse M2M industry working groups are koneki, Eclipse SCADA for open standards for Comm. protocols.
* ITU-T Focus Group M2M global standardisation initiative for a Common M2M Service Layer.

1.6 : EXAMPLES OF IOT :-

Examples of IoT usages are Wearable devices such as watches, fitness trackers, sleep monitoring & heart monitors etc.

1.6.1 : Wearable Smart Watch :-

Following are examples of wearable smart watches.

Samsung Galaxy Gear S :

- 2-inch Curved display
- Ability to make phone calls
- Wifi & Blue tooth Connectivity options.
- GPS enabled.
- S Health app measures heart rate & UV monitors.

Apple watch:

Apple i smart watch has Apps like Nike + Running to track morning or evening runs & health & fitness.

- Track walks
- Measure heart rate
- make payment using payment Wallet
- enable listing songs.
- Enable chat with family
- update e-mail
- Find a taxi
- update news
- navigate for long car trips.

1.6.2: SMART HOME:-

Sensors & Actuators manage a smart home with internet Connection. Wired & Wireless sensors are interconnected / incorporated into security sensors, cameras, thermostats, smart plugs, lights & entertainment systems.

Do-it-yourself (DIY) sensors & Actuators, include smart plug, motion detector, door/window detector, smoke detector, energy meter interface (electric, gas, water), remote control, smart relay, surveillance camera, wireless Hi-Fi speakers. etc.

A connected home has the following applications deployed in smart home:

- Mobile, tablets, IP-TV, VOIP telephony, Video-Conferencing, video-on-demand, video-surveillance, Wifi & internet.
- Home security: Access Control & Security alerts.
- Lighting Control
- Home health care
- Fire detection & leak detection.
- Energy effeciency.
- Solar panel monitoring.
- Automated meter reading.

Home Automation Software :-

Intel-based intelligent gateway enables creation of home automation system offered by service providers for telephony, mobile, cable, broadband & security.

1.6.3 : SMART CITIES :

The IoT Concept extends to IoE for developing smart cities. A Four layer archi framework developed at CISCO for a city is as follows.

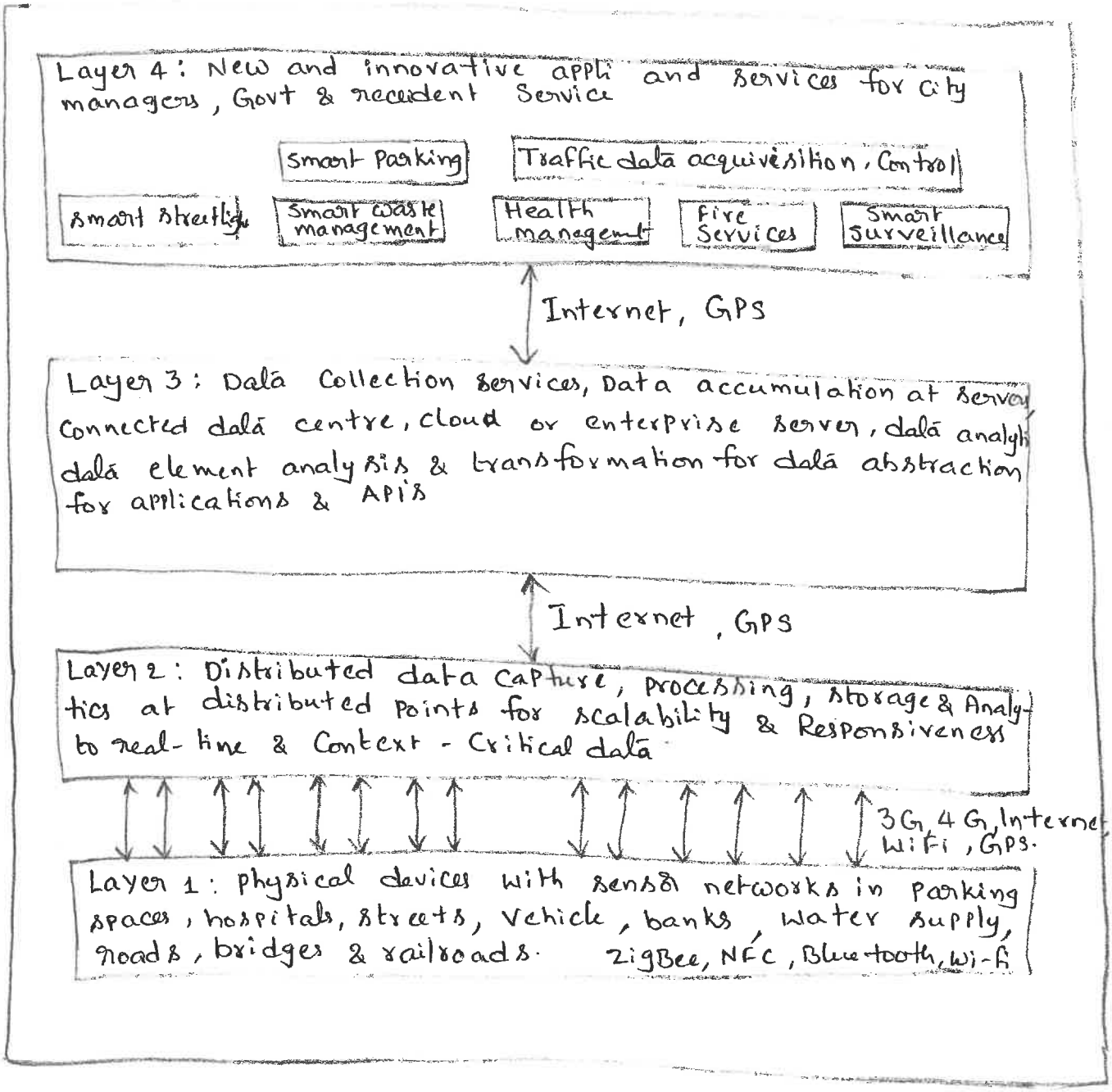


Fig: Four layer archi framework developed CISCO for a city.

IoT Technology :-

IoT primarly exploits standard protocols & networking Technologies. However the major enabling technologies and protocols of IoT are RFID, NFC, Low-energy Bluetooth, low-energy wireless, low-energy radio protocols, LTE-A and Wifi-direct. These technologies support the specific networking functionality needed in an IoT system in contrast to a standard uniform network of common systems.

NFC and RFID :-

RFID (Radio-frequency identification) and NFC (near-field Communication) provide simple, low energy and versatile options for identify and access tokens, Connection bootstrapping and payments.

- RFID Technology employs 2-way radio transmitter-receivers to identify and track tags associated with objects.

- NFC Consists of Communication protocols for electronic devices, typically a mobile device and a standard device.

Low - Energy Bluetooth :-

This technology supports the low - power, long - use need of IoT function while exploiting a standard technology with native support across systems.

Low - Energy Wireless :-

This technology replaces the most power hungry aspect of an IoT system. Though sensors and other elements can power down over long periods, Communication links (i.e, Wireless) must remain in listening mode. Low - energy wireless not only reduces Consumption, but also extends the life of device.

Radio protocols :-

ZigBee, Z-Wave, and Thread are radio protocols for Creating low - rate private area networks. These tech - nologies are low - power, but offer high throughput unlike many similar options. This increases the power of small local device networks without typical cost.

LTE - A :

LTE - A or LTE Advanced, delivers an important

upgrade to LTE Technology increasing not only its coverage, but also reducing its latency and raising its throughput. It gives IoT a tremendous power through expanding its range, with its most significant applications being vehicle, UAV, and similar Communication.

WiFi - Direct :-

Wifi - Direct eleminates the need for an access point. It allow P2P (Peer - to - Peer) Connections with the speed of wifi, but low latency. It eleminates an element of a network that often bogs it down, and it does not Compromise on speed.

The internet of Things will demand an extensive range of new Technologies.

The top most emerging IoT technologies are:

IoT Security :-

Security technologies will be required to protect IoT devices and plat-forms from both information attacks & physical tampering, to encrypt their Communications and to address new challanges such as impersonating "things" or denial-of-sleep attacks that drain batteries.

IoT Analytics :-

IoT business models will exploit the info. Gllected by "things" in many ways, which will be demand new analytics tools and algorithms. The data volumes increses over the next five years, needs of IoT may diverage fiurther from traditional Analytics.

IoT Device Manager :-

Long-lived nontrivial "things" will be require management & monitoring, including device monitoring, firmware & software updates, diagnostics, crash analysis & reporting, physical management & security management.

Low-Power, Short range IoT Network :-

It will dominate wireless IoT Connectivity through 2025, for outnumbering Connections using Wide-area IoT Networks.

Low-Power, WAN :-

Traditional cellular networks don't deliver a good Combination of technical features & operational Cost for those IoT applications that need wide area Coverage Combined with relatively low bandwidth, good battery life, low hardware & Operating Cost & high Connectivity.

IoT processors :-

The processors and architectures used by IoT device define many of their capabilities, such as whether they are capable of strong security & encryption, power consumption, whether they are sophisticated enough support an O.S, updatable firmware & embedded device management.

IoT Operating System :-

Traditional O.S such as windows & iOS were not designed for IoT applications. They consume too much power, need fast processors, and in some cases, lack features such as guaranteed real-time response. They also have too large a memory foot print for small devices and may not support the chips that IoT developers use. Consequently a wide range of IoT specific O.S has been developed to suit many different hardware foot prints & feature needs.

Event Stream processing :

Some IoT Applications will generate extreamly high data rates that must be analyzed in real time. Systems generate tens of thousands of events per second are common, & millions of events per second TO address

such requirements, distributed stream Computing platforms have emerged that can process very high data rate data streams & perform tasks such as real time analytics.

Design principles for Connected Devices :-

IoT or M2M device data refers to the data meant for Communication to an application, service (or) process. Data also refers to data received by a device for its monitoring.

Data stack denotes the data received after the actions at various - in - between Layers. Layers in open systems interConnection (OSI) model are application, presentation, Session, Transport, Network, Data Link & physical.

Actions at the data - adaption or other layers can be related to data privacy, data security, data Consolidation, aggregation, Compaction and fusion. An action can be a gateway action - Using one protocal for reception & another one for transmission.

The Actions at Layer can be adding additional header after appropriate data formatting.

Actions besides appending a header can be of appending the additional bits at the various in - between units.

Layer refers to a stage during a set of Actions at which the actions is taken as per a specified protocol, and then result passes to next layer until set of Actions Completed. A layer may Consist of various sub layers.

Physical Layer refers to a layer at transmitting-node or at receiving node for data bits. The transfer uses physical systems & refers to wireless (or) Wired transmission. It is a lowest layer.

Application Layer refers to a layer for transmitting or receiving the data bits of an application. Data Data bits route across network & transfer taks place as follows: application data from the application layer transfers after passing through several in-between Layers to the physical layer, and from there it transmits to receiving-end physical layer.

Level refers to a stage from the lowest to highest.

Domain refers a set of software, Layers or levels having specific applications and capabilities.

Gateway refers to software for Connecting 2 applic-
[illegible] at the sender & other receiver (ALC)

[application layer gateway]. A gateway may be of different types. A Communication gateway at device and gateway domain has capabilities as protocal-Conversion during Communication between 2 ends.

IPV6 (or) IPV4 for network layer.

Header means a set of octets Containing info about the data being sent. Header packs the data of layer before transmission to next Layer during Communication between 2 endpoints.

Packet means packaged data-stack which routes over network. Packet size limit is according to protocal.

IPV4 Packet Size limit is 2^{16} B

PDU (Protocal Data unit) is a unit of data which is specified in a protocal of a given layer which transfers from one layer to another.

MTU (Max Transmission Unit) is largest Size frame or packet or segment specified in octets (1 Octet = 1 byte = 8 bits) that can be sent in a Packet (or) frame based network such as internet.

Star network denotes number of nodes interacting with coordinator.

Mesh network denotes no. of nodes that interconnect each other.

End point device or node denotes the one that provide Connectivity to coordinator or router.

Coordinator denotes that one can connects a no. of end points as well as routers.

Master refers to one who initiates the pairing with the devices in star topology.

Routers refers to a device or node capable of storing paths to each destination.

process means a software component, which process the input and generate output.

UNIT - 1

SHORT ANSWER QUESTIONS :-

1). Definition of IoT ?

2). Discuss two real time examples of IoT ?

3). Explain Behind IoT Technology ?

4). What is M2M Communication ?

5). What is the vision of IoT ?

6). Explain Conceptual Framework of IoT ?

7). What are the major Components of IoT ?

8). Explain different Platforms & Integration tools in IoT .

9). Define RFID in IoT .

10). What is WSNs & Explain.

11). Explain Softwares & Development tools in IoT ?

12). Give a short notes on Design principles of IoT.

13). Explain any 3 IoT technologies .

14). What are the differences between IoT & M2M.

15). What is Communication module in IoT.

ESSAY QUESTIONS :-

1). Explain in detailed about IoT Technologies.

2). Discuss behind IoTs & Sources of IoTs.

3). Briefly discuss the Concept of M2M Communi-
cation

4). Draw & Explain in detailed about IoT Archi-
tectural view.

5). Explain in detailed about IoT Conceptual Frame
work with Neat Sketch.

6). Write any two realtime examples of IoT with
neat sketch.

UNIT-2

Business Models for Business Processes in the IoT, IoT/M2M systems LAYERS AND designs standardizations, Modified OSI stack for the IoT/M2M systems, ETSI M2M domains and High level capabilities, Communication Technologies, Data Enrinchment and Consolidation and Device Management Gateway Ease of designing and affordability.

Objectives :-

* IoT Business Process.
* IoT/M2M Systems Layers & Design standardization.
* Modified OSI Stack for IoT/M2M
* ETSI M2M domains
* Communication Technologies
* Data Enrichment & Consoliadation
* Device Management Gateway.

IoT / M2M SYSTEM, LAYERS & DESIGN STANDARDISATION :-

A no. of international organisational have taken action for IoT design standardisation.

These are the following examples :-

Internet Engineering Task Force (IETF), an international body initiated actions for addressing & working on recommendations for engineering specifications for IoT.

International TeleCommunication Union for TeleCommunication (ITU-T) suggested a reference model for IoT domain.

European TeleCommunication standards Institute (ETSI) initialed the development of a set of standards for the networks, & device & gateway domains for the Communication between machines (M2M).

Open Geospecial Consortium (OGC), an International Industry Consortium, has also suggusted open standards for sensors discovery, Capabilities, Quality & other Aspects with suppat to geographical

MODIFIED OSI MODEL for IOT / M2M SYS:-

OSI protocals means a family of info. exchange standard development Jointly by ISO and ITU-T. The Seven-layer OSI model is a standard model. It gives the basic outline for designing a Communication network.

Various models for data interchanges Consider the layers specified by OSI model, and modify it for simplicity acording to requirement.

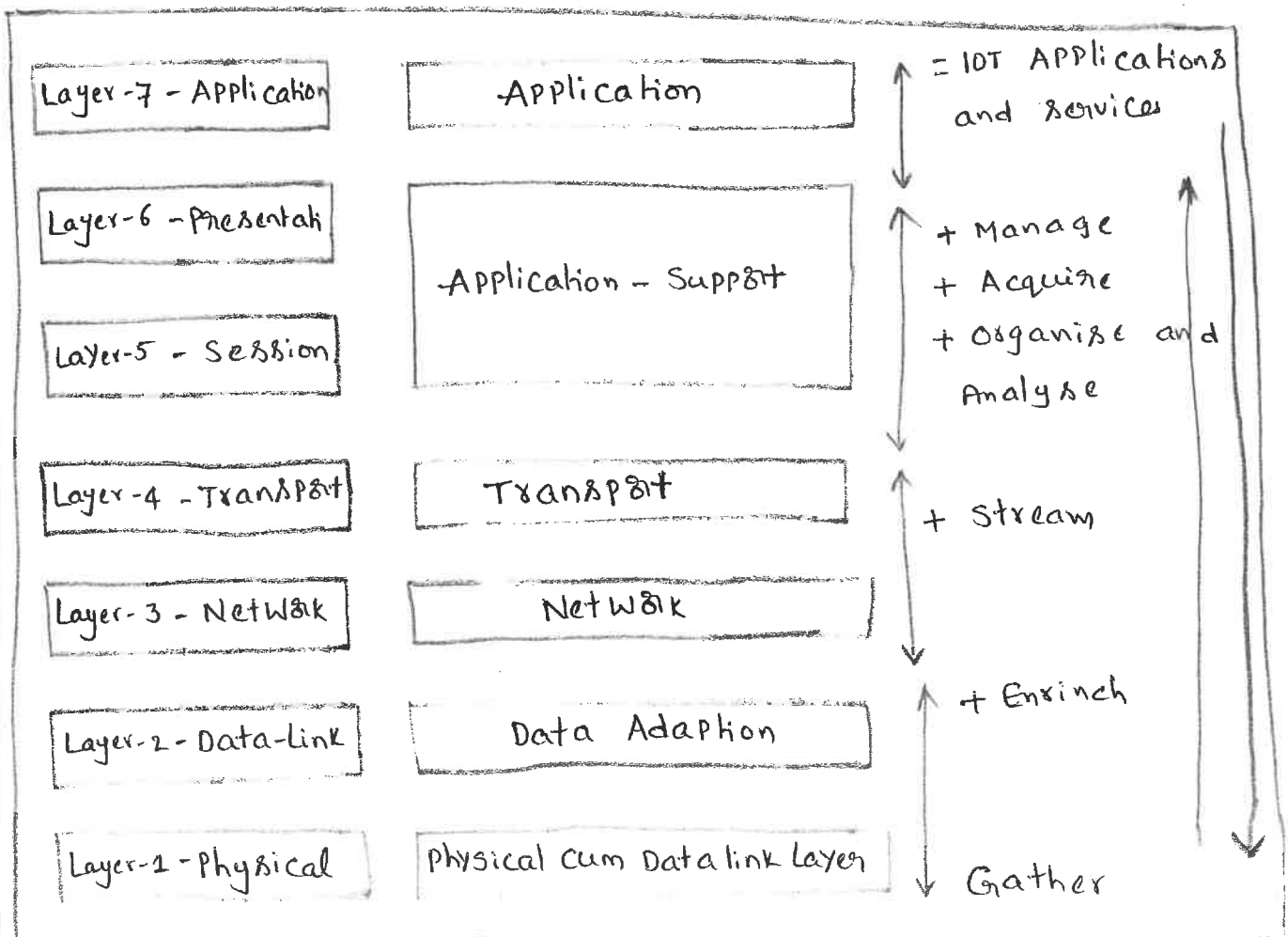


Fig: Seven-Layers generalised OSI model & IETF six layer mod

The above diagram shows a classical 7-Layer OSI model & the modifications in the model proposed by IETE. Data Communication from device end to application end. Each layer processes the received data and creates a new data stack which transfers it to the next layer. The processing takes place in between layers. Device end also receives data from an application after processing at the in-between layers.

Gather + Enrinch + Stream + (Manage + Acquire + Organise + Analyse) = IoT Applications and Services.

→ new Applications & Services present at Application Layer 6. A modifications to this is that the application-support layer 5 uses protocols, such as CoAP.

→ IoT Applic. and Services Commonly used them for network Communication.

Modifications are also at the data-link layer 2 & Physical layer 1 (L1). The new layers are data adaption (new L2) and physical cum data-link (new L1). The adaption layer include a Gateway.

ITU-T Reference Model :-

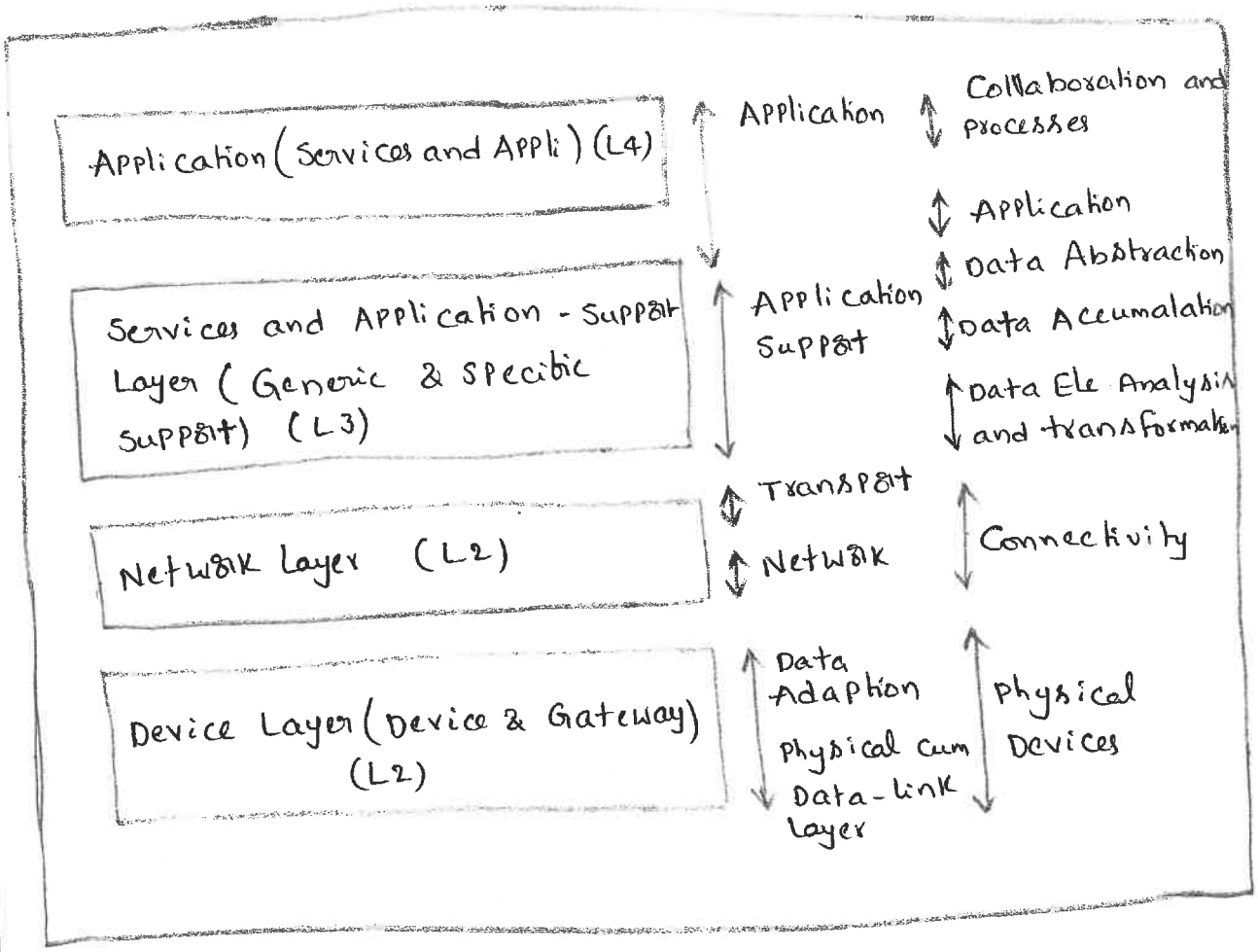


Fig: ITU-T reference model RM1, its Correspondence with Six layers of modified OSI & Comparision with 7-Levels suggested in CISCO IOT Reference model RM2.

- Lowest layer, L1 is the device layer and has device and gateway capabilities.
- Next layer, L2 has transport & network capabilities
- Next layer, L3, is the services and application-support layer. support layer has 2 types of capabilities.

generic and Specific Services.

- TOP Layer, L4 is for applications and services.

COMMUNICATION TECHNOLOGIES:-

Physical cum data-link layer in the model Consists of a local area network / Personal area network. A local network of IoT or M2M device deploys one of one of 2 types of technologies.

1. Wireless Communication Technologies.
2. Wired Communication Technologies.

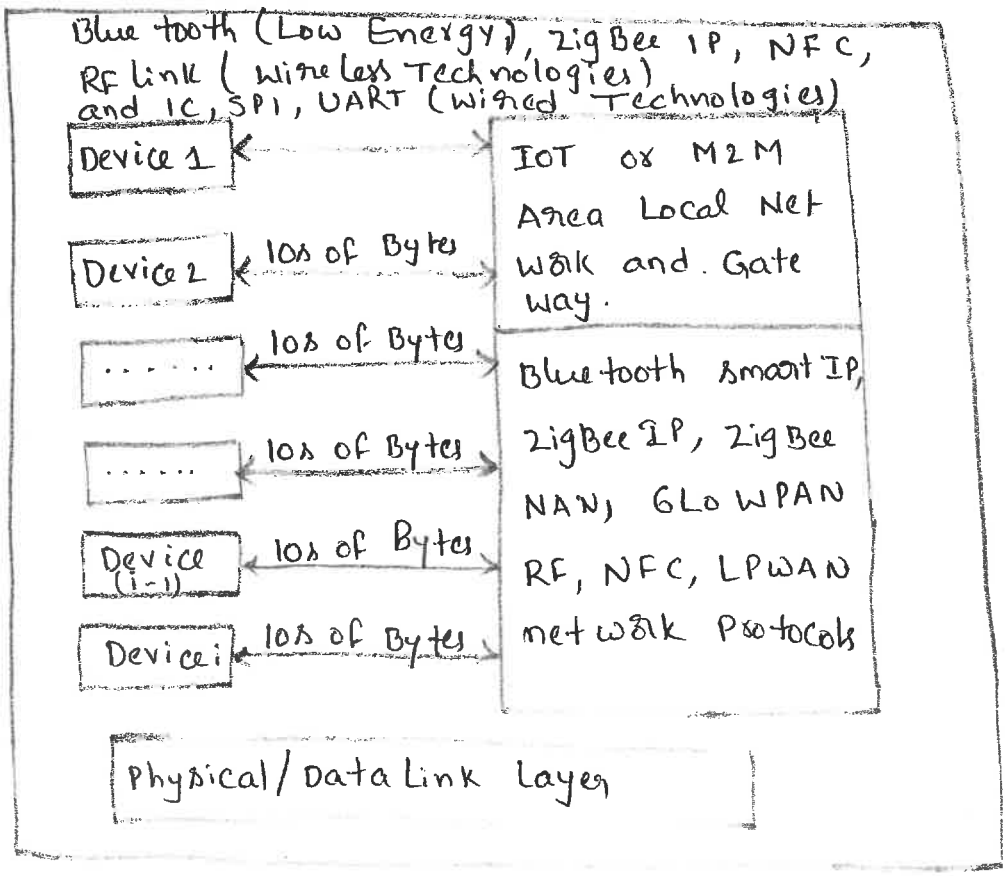


Fig: Connected devices 1st to ith Connected to local n/w & gate

The below fig shows Connected devices (1st to ith) Connectivity using different tech. for Comm of data from and to devices to local network Connectivity by/to a gateway.

Wireless Technologies :-

→ Physical cum Data link layer uses Wired (or) Wireless Comm. tech. Example of Wireless Comm. technologies are

1). NFC (Near-Field Communication)
2). RFID (Radio Frequency Communication)
3). BT (Bluetooth BR/EDR and Bluetooth Low energy)
4). ZigBee IP / ZigBee SE 2.0
5). Wi-Fi .
6). RF Transceivers and RF Modules.
7). Wireless USB.

Wired Communication Technology :-

Wired Communi. Can be Serial asynchronous Communication (or) synchronous Serial Communi. Example Wired Communication technologies are.

1). UART (Universal Asynchronous Transmitter) / USART Communication.
2). Serial Peripheral interface
3). I2C Bus
4). Wired USB
5). Ethernet .

Wireless Technology :- (or) Wireless Communi. Tech :-

NFC [Near field Communication] :-

It is an Enhancement of ISO/EC 14443 standard for Contact-less proximity-card.

NFC is an short distance (20 cm) Wireless Communication technology. It enables data exchange between cards in proximity and other devices. Example of applications of NFC are proximity-card reader /RFID /IOT /M2M mobile devices, mobile payment wallet.

NFC devices transmit and receive data at the same instance & the setup time is 0.1 s. The device or its reader can generate RF fields for the nearby passive devices such as Passive RFID. NFC device can check RF field and detect collision of transmitted signals.

Range of functioning is with in 10 to 20 cm. The device can also Communicate with Bluetooth and Wifi devices in order to extend the distance from 10 cm to 30 cm.

Three modes of Communication are:

1). Peer-to-Peer:

Both devices use active devices in which RF fields alternately generate when Communicating

2). Card - emulation Mode :-

Communication with out interruption for the read and write as required in a smart card and smart card reader. FeliCa and Mifare standards are Protocols for reading & writing data.

3). Reader mode :-

Using NFC the device reads passive RFID Device. The RF field is generated by an active NFC device.

RFID (Radio Frequency Identification) :-

RFID is an automatic identification method. RFIDs uses the internet. RFID usage is, therefore, in remote storage & retrieval of data is done at the RFID Tags. An RFID device functions as a tag, which may be placed on an object.

IOT applications of RFID are in business processes, such as parcels tracking and inventory Control, sales log-ins and supply-chain management.

Bluetooth BR/EDR and Bluetooth Low Energy :-

Bluetooth devices follow IEEE 802.15.1 standard protocol for L1. BT devices form a WPAN device network. 2 types of modes for the devices are Bluetooth low Energy (BT LE 1 mbps).

A latest version is bluetooth V4.2. BT LE is also called Bluetooth smart. Bluetooth V4.2 provides the LE data packet length extension, link layer privacy & Secure Connection. BT LE range is 150 m to 10 mW power output, data transfer rate is 1 Mbps. & setup time is less than 6s.

Bluetooth V5, released June 2016, has increased broadcast capacity by 800%, quadrupled the range & double speed.

A Device may have single BT LE or dual mode BT BR/EDR. It's features:

- Auto synchronous between mobile & Other devices when both use BT.
- Radio range depending on class of radio; class 1 (or) class 2 or radio's: 100 m, 10 m, or 1 m

- Support to NFC pairing for low latency in pairing the BT devices.
- TWO modes - dual or single mode devices are used for IoT / M2M devices.
- IPV6 Connection option for BT Smart with IPSP (internet protocal support profile)
- Smaller packets in LE mode.
- Operation in secured as well as unsecured modes
- AES - CCM 128 authenticated encryption algo for Confidentiality & Authentication
- Connection of IoT / M2M / mobile devices using BT EDR device to Internet with 24 mbps wi-fi 802.11 adaption layer.

ZigBee / ZigBee SE 2.0 :-

It follow the IEEE 802.15.4 standard protocal L1 (Physical cum data-link layer). ZigBee devices form a WPAN devices network.

ZigBee end-point devices form a WPAN of embedded sensors actuators, appliances, controllers (or) medical data systems which Connect to the internet for IoT Applications, services & business process.

ZigBee Neighbourhood Area Network (NAN) is a version for smart grid. ZigBee Smart energy version 2.0 has energy management and energy effeciency capabilities using IP Network.

The features of ZigBee IP are;

- L1 layer PDU = 127 B
- Used for low-power, short-range WPAN.
- The device can function in six modes - end point, ZigBee-ZigBee devices routers, ZigBee network Co-ordinator, ZigBee-IP coordinator, ZigBee-IP router & IP host.
- ZigBee IP enhancement provisions the IPV6 Connectivity A ZigBee IP device is a Reduce Function Device(RFD)
- The ZigBee router uses reactive and proactive protocals for routing mode, which enable applications in big-scale automation & remote Control.
- Self Configuration & self-healing dynamic Pairing mesh network, supports both multicast & unicast options.
- Multicast forwarding to support multicast Domain Name System (mDNS) Based Service Discovery.

- Support to development of discovery mechanism with full application confirmation.
- Support to pairing of coordinator with end-point devices & routers in star topology.
- Provide bigger network using multiple star topology & inter-PAN communication.
- Support to sensor nodes & sensors network integration.
- Low latency (<10 ms) link layer connection.
- Range is 10-200m, data transfer rate is 250 kbps, low power.
- Include RFD in ZigBee SE 2.0.

ZigBee NAN is for devices which are used for smart-metering, distribution automation devices & smart grid communication profile. NAN enables a utility's last-mile at HAN (HOME AREA NETWORK), outdoor Access network. smart meters to WAN gateways.

→ The below diagram shows ZigBee End point, Co-ordinator, Router, ZigBee IP routers modes forming star, mesh and IP network of ZigBee sensors, end devices & ZigBee routers device which interconnects Internet IPV4, IPV6

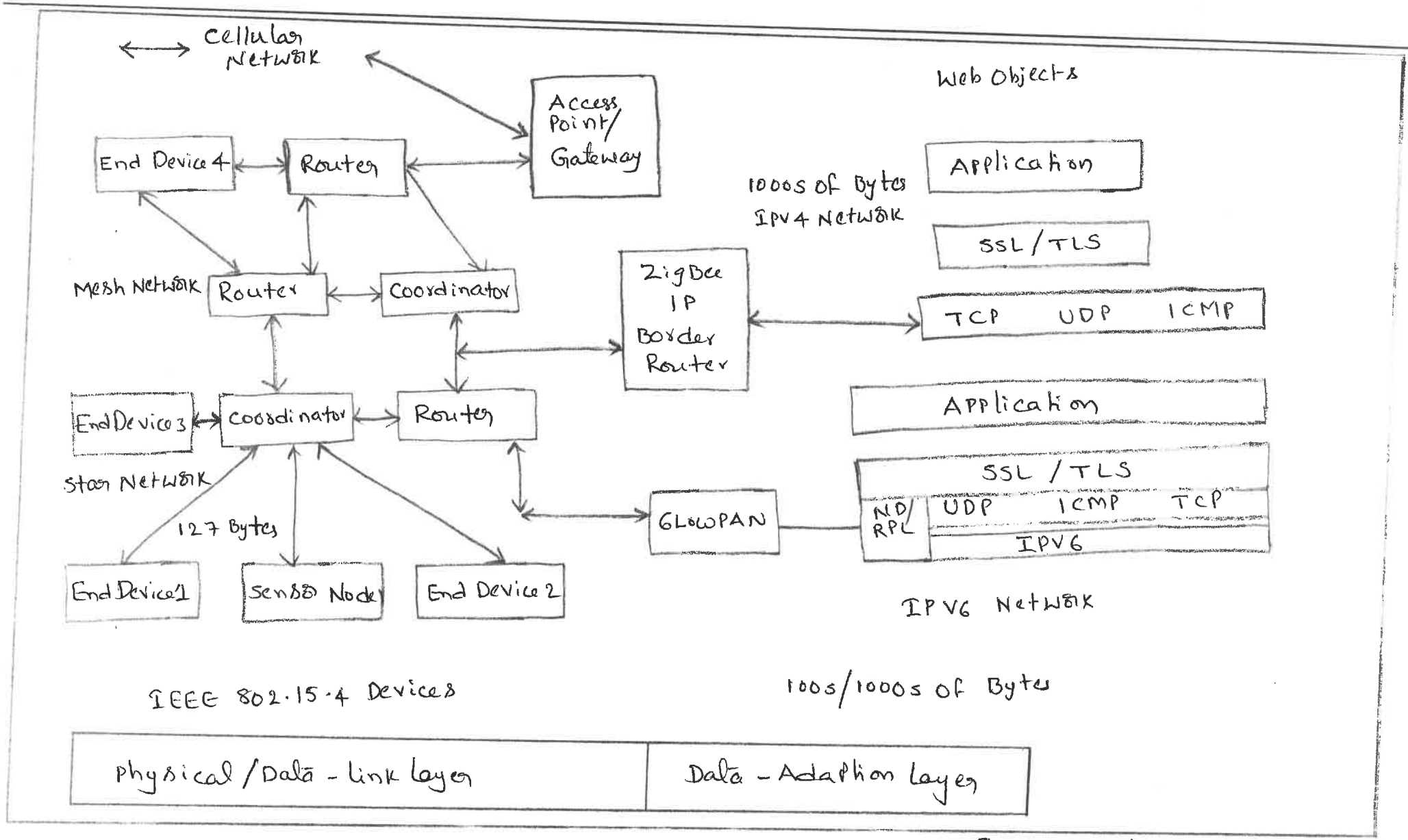


Fig : ZigBee end Point, Coordinator, router, ZigBee nodes forming star, mesh and IP networks of ZigBee sensors, end devices and zigBee Router devices which inter Connected to network/internet IPV4, IPV6 & Cellular networks.

The figure shows :

- Three end devices, two routers, one sensor node connected to coordinator ZigBee devices forming a star network.
- one end device, 2 routers & one coordinator forming a Mesh network.
- Mesh routers Connects to an AP/ gateway, which in turn Connects to a cellular network.
- Coordinator of mesh network Connect ZigBee IP border router, which enables local zigBee network Connectivity to internet.
- A router in star network Connects to 6LoWPAN, which Connects an IEEE 802.15.4 device network to IPV6.
- 1000s bytes Communicate between the network layer & IoT web objects.
- 127 B Communication between adaption layer IEEE 802.15. 4 devices at single data transfer.
- IETF ND (Neighbour Discovery), ROLL (Routing over Low power Loss Network), RPL routing, IPV6 / IPV4 network, TCP/UDP/ICMP transport, SSL/TLS security Layer Protocols for Comm. between web objects & ZigBee devices.

WiFi :

Wi-Fi is an interface technology that uses IEEE 802.11 protocal and enables WLANs (Wireless Local AreaNetwork). It Connect enterprises, universities & offices through home AP/public hotspots. WiFi Connects distributed WLAN networks using internet.

Automobiles, instruments, home networking, sensors, actuators, industrial device nodes, Computers, tablets, mobiles, printers & many devices have WiFi interface.

WiFi is very popular. Wifi interface Connect within themselves or to an AP or wireless router using WiFi PCMCIA or PCI card through the following:

- Base station or AP
- A WLAN transceiver or BS Can Connect one or more wireless devices to the internet.
- Peer-to-Peer with out access point : client devices with in Independent Base Service Set (IBSS) network can Communicate directly with each other. It enables fast & easy setting of 802.11 network.
- Peer to multipoint nodes with Basic Service Set(BSS) using one in-between AP point or distributed BSS Connect through multiple AP's.
- Connectivity range of each BSS depends on range of wireless bridges.

- Each BSS is a Service Set Identifier (SSID).

The below Digram shows 3 WLAN networks (BSSs) for sensor device nodes, mobiles, tablets, laptops, Computers and internet Connectivity of WLAN networks with IPV4 networks.

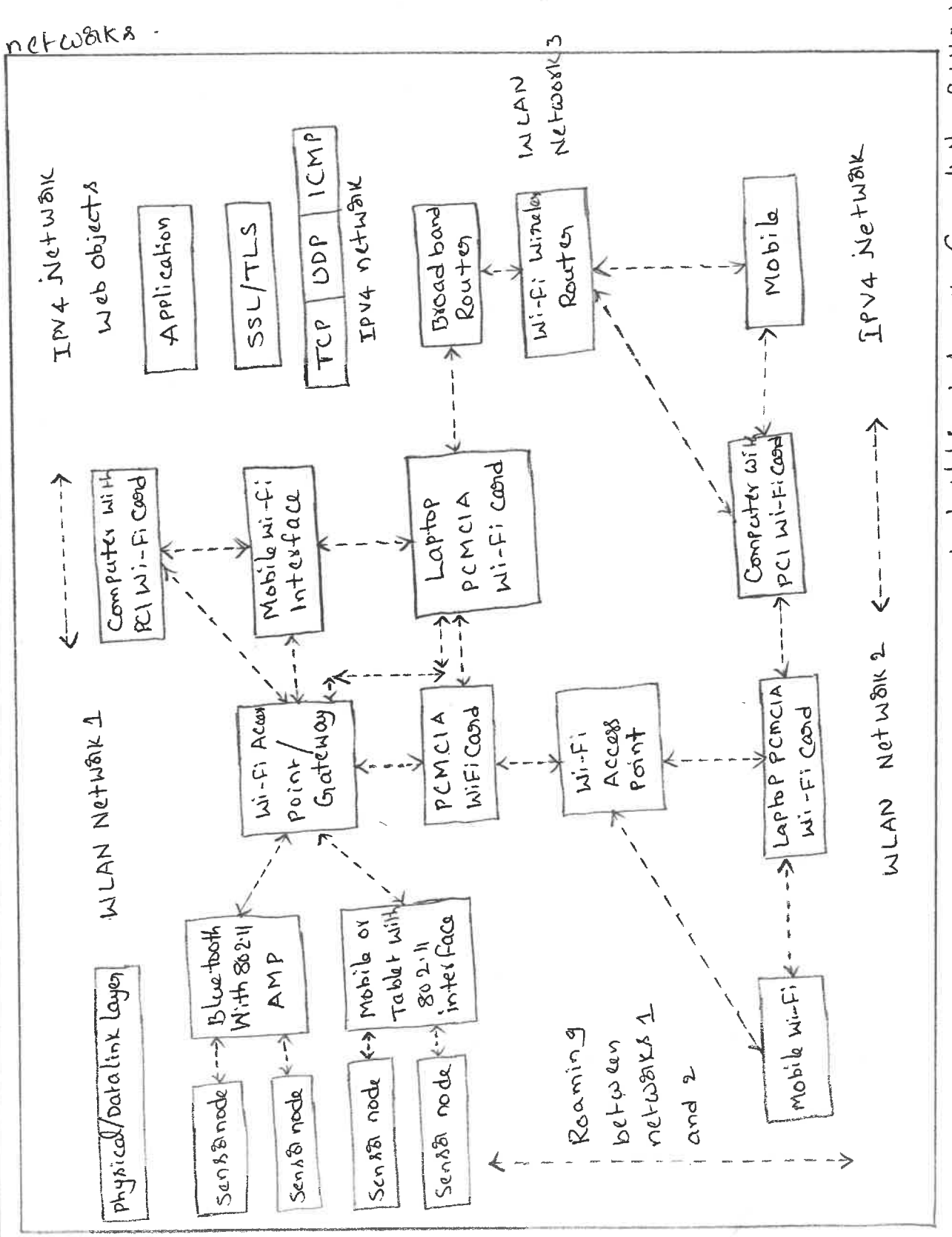


Fig: 3 WLAN N/W for sensor device nodes, mobile, tablets, Internet Connectivity of WLAN with IPV4 networks.

- Sensor nodes Connected to BT with Wi-Fi adaption, 802.11 interfaces in a WLAN. network 1
- Tablets, Wi-Fi, Computers also Connect in WLAN1 through AP.
- AP1 Connects to a broadband router 1 & to the IPV4 network 2.
- WLAN1 & WLAN2 function as BSS.
- WLAN 2 also Connect/ Consists of AP2, Wi-Fi router & other Wi-Fi enabled interface.
- Wi-Fi router Connects to multiple Wi-Fi nodes.
- Broadband routers 1 & 2 Connect using Wires, to IPV4 networks & Web objects.
- Interfaces use 2.4 GHz or 5 GHz antenna.
- Offers mobility and roaming.
- Have easy installation simplicity & flexibility
- Coverage range is 30 m to 125 m.
- used in a room having limited - Coverage 802.11 a which coexists with b, coexists with b&g
- uses the 802.11 b in wider Coverage range.
- Uses 802.11 g for high data rates up to 54 mbps & 802.11 n for very high 600 Mbps using multipl antennas.
- provide dynamic environment for network expen- dability & Scalability.

Wireless protected Access (WPA) & WEP Security Sublayer

RF Transceivers and RF Modules :-

RF transmitters, receivers & transceivers are simplest RF circuit. A transceiver transmits the RF from one end & receives the RF from other end, but internally additional circuit, which seperate the signals from both ends.

RF Technology Consists of following elements :-

* RF interface / physical layer, RF signals transmit between the nodes or endpoints
* RF network Architecture includes the overall system architecture, backhaul, server and bidirectional end-devices with radio duty cycling in the application. It means managing the active intervals, transmitting & receiving schedule & time intervals of action.

GPRS / GSM Cellular Networks - Mobile internet :-

An IoT / M2M Communication gateway can access a Wireless Wide Area Network (WWAN). The network access may use a GPRS network.

A mobile phone provisions for a USB Wired Port, BT and Wi-Fi Connectivity. Wireless Connectivity for Internet uses data Connectivity using GSM, GPRS, LTE mobile service

Wireless USB :-

It is an extension of USB 2.0 & it operates at ultra-wide band (UWB) 5.1 GHz to 10.6 GHz frequencies. It is for short range personal area Network (PAR) High speed 480 Mbps 3 m or 110 mbps 10m Channel).

Wired Communication Technology :-

It can be Serial asynchronous Communication (for example UART interface) or synchronous Serial Communication (Ex: SPI interface). Communication can be over a bus when a no. of systems Connect through a Common set of interConnection.

Wired Communication can be done using Ethernet IEEE 802.2 bus specifications - A MAC sublayer data frame may be according to Ethernet protocol. Wired Communication can also use a USB port, a micro USB.

UART / USART Serial Communication :-

A universal Asynchronous Transmitter (UART) enables Serial Communication of 8-bits serially with a start bit at the start of a byte on Serial Transmitter Data (TxD) output line.

Asynchronous refers to all bytes in a frame transmit, which can result in variation in time interval spacing. This is because clock information of transmitter does not transmit along with the data. The receiver clock also doesn't synchronize with data. Further, successive set of bytes may wait after transmission till an acknowledgement is received from receiving end.

An Universal Synchronous Asynchronous Transmitter (USART) enables serial Communication in synchronous as well as asynchronous.

Synchronous means all bytes in a frame transmit with equal time saping.

Serial pheripheral Interface:-

It is one of the widely used serial synchronous Comm methods. Source of serial synch. output (or) input called master. A receiver of serial synchronous input or output called a slave, when along with Serial data it also receives the synchronizing clock info from master. 4 sets of signals, viz, SCLK, MISO, MOSI & SS are used in 4 wires. when SS is active the device functions as a slave.

Master Input Slave Output (MISO) and Master Output Slave Input (MOSI) are serial synchronous I/O bits at master & slave & los are per synchronising clock.

I2C Bus :-

A no. of device integrated circuits for sensors, actuators, flash memory & touch screens need to data exchanges in a no. of processes. I.C's mutually network through a common synchronous serial bus called inter-integrated circuit (I2C).

The I2C was originally developed at phillips semi Conductors. There are 3 I2C bus standards:

1). Industrial 100 Kbps I2C,

2). 100 Kbps SM I2C

3). 400 Kbps I2C.

I2C bus has 2 lines that carry signals

* One is for the clock
* one is for the bidirectional data.

Wired USB :-

Universal Serial Bus is for fast serial transmission & reception between the hosts, the embedded system & distributed serial devices.

Ex: keyboard, printer, Scanner.

USB is a bus between the host system and number of interconnected peripheral devices. Maximum 127 devices can connect with a host. USB provides a fast (up to 12 mbps) & low speed (up to 1.5 mbps) serial transmission & reception between host & serial devices.

USB three standards are USB 1.1 (1.5 & 12 mbps), 3.0 (micro size) 5 Gbps and 3.1, 2.0 (mini size connector) 480 mb

Features of a USB are:

USB data format and transfer serial signals & non Return to zero (NRZ) & Clock is encoded by inserting synchronous code (SYNC) field befor each packet.

Data ~~types~~ transfer is of 4 types:

- ↳ Controlled data transfer
- ↳ Bulk data transfer
- ↳ Interrupt data transfer
- ↳ isosynchronous transfer.

USB is a polled bus. Polling mode functions as: A host controller regularly polls the presense of a device as

scheduled by software. It sends a token packet. Token consists of field type, direction, USB device address & device end-to-end point number. The device does handshacking through a handshake packet, indicating successful.

A USB support 3 types of pipes - i.e streams, default control, message.

Ethernet :-

It standard is IEEE 802.2 protocal for LAN, WSK stations & device LANs. Each frame at a LAN consists of header. Ethernet enables the service of local device nodes, computers.

Features :-

- Uses passive broadcast medium & Wired Connection based
- Formatting of frame is according to IEEE 802.2.
- Uses a 48-bit MAC address assigned distinctly to each computer
- ARP (Address Resolution protocal) resolves 32 bit IP address. Reverse Address Resolution protocal (RARP) resolves 48-bit destination host media address into 32-bit.

- Uses Wired bus topology, and transmission speeds are 10 mbps, 100 mbps (unshielded, shielded wires), 1 Gbps (high-quality cable), 4 Gbps (in twisted pair model & 10 Gbps.

- Uses wired bus topology.
- Uses MAC-based on CSMA/CD. The CSMA/CD mode is half-duplex (wired model which means transmit (Tx) & Receive (Rx) signals can be sent on same wire.
- Uses transmission data stack into frame at MAC layer, & each frame includes header.

Communication Technologies:-

Communication Technologies are NFC,
BTLE,
ZigBee,
WLAN protocols.

Differences between NFC, BT LE, ZigBee & WLAN Protocols.

Property	NFC	BT LE	ZigBee IP	WLAN 802.11
IEEE Protocal		802.15.1	802.15.4	802.11z
Physical Layer	848,424, 212,106 Kbps	2.4 GHz(LE-DSSS)	2.4 GHz, 868MHz and 433 MHz DSSS MAC layer CSMA/CA	2.4 GHz TWO PHY layers MAC Layer CSMA/CD
Data transfer rate	106 Kbps	1 Mbps	250 Kbps(2.4GHz, 40kbps 915 MHz, 20kbps 868.3MHz	11 mbps/54 mbps
Form Factor and Range	10-20 cm	Small	Small 10m to 200 m	Bigger
Power Dissipation	very low	Lower than ZigBee, much lower than WLAN 802.11	2mW Router and 0.1mW for end-device much lower than WLAN 802.11	Much Higher than ZigBee
Set up/ Connection/ Disconnection Intervals	0.1s	3s Connection Time <3 ms	20 ms Connection time <10 ms	
Security	-	AES-CCM-128	AES-CCM-128	WEP
Application	Payment-Wallet, short distance communic	WPAN, IOT/M2M devices, Widely present in mobile and tablets and, need addition circuit in sensors, actuators, Controllers & IOT device	WPAN, Wider presence in sensors, actuators, Controllers, automobile & medical electronic and IOT devices Connectivity using IPv6, 6LoWPAN, ROLL, RPL & TLSv1.2	WLAN and WWAN network tablet, desktops, mobile, devices with PCMCIA interface, home networking, Easy IPv4 Connectivity.
Broadcast/ Multicast/ Unicast	Unicast	Unicast	unicast/multicast	unicast
Network	point to point between active and passive devices	star topology, Peer-to-Peer Piconet expended by inter-piconets data transactions and synchronisation.	Low power, mesh or peer-to-peer star networks using end devices, Coordinators, router, ZigBee IP border router.	LAN Topology IBSS, BSS and distributed BSSs for WWLAN widely used for internet Connectivity of mobiles, tablets, desk-

Data Enrichment, Data Consolidation and Device Management AT Gateway:-

A gateway at a data-adaption layer has several functions. These are data privacy, data security, data Enrichment, data Consolidation, transformation and device Management.

The below figure shows IoT or M2M gateway Consisting of data enrichment, Consolidation & Device Management & Communication framework.

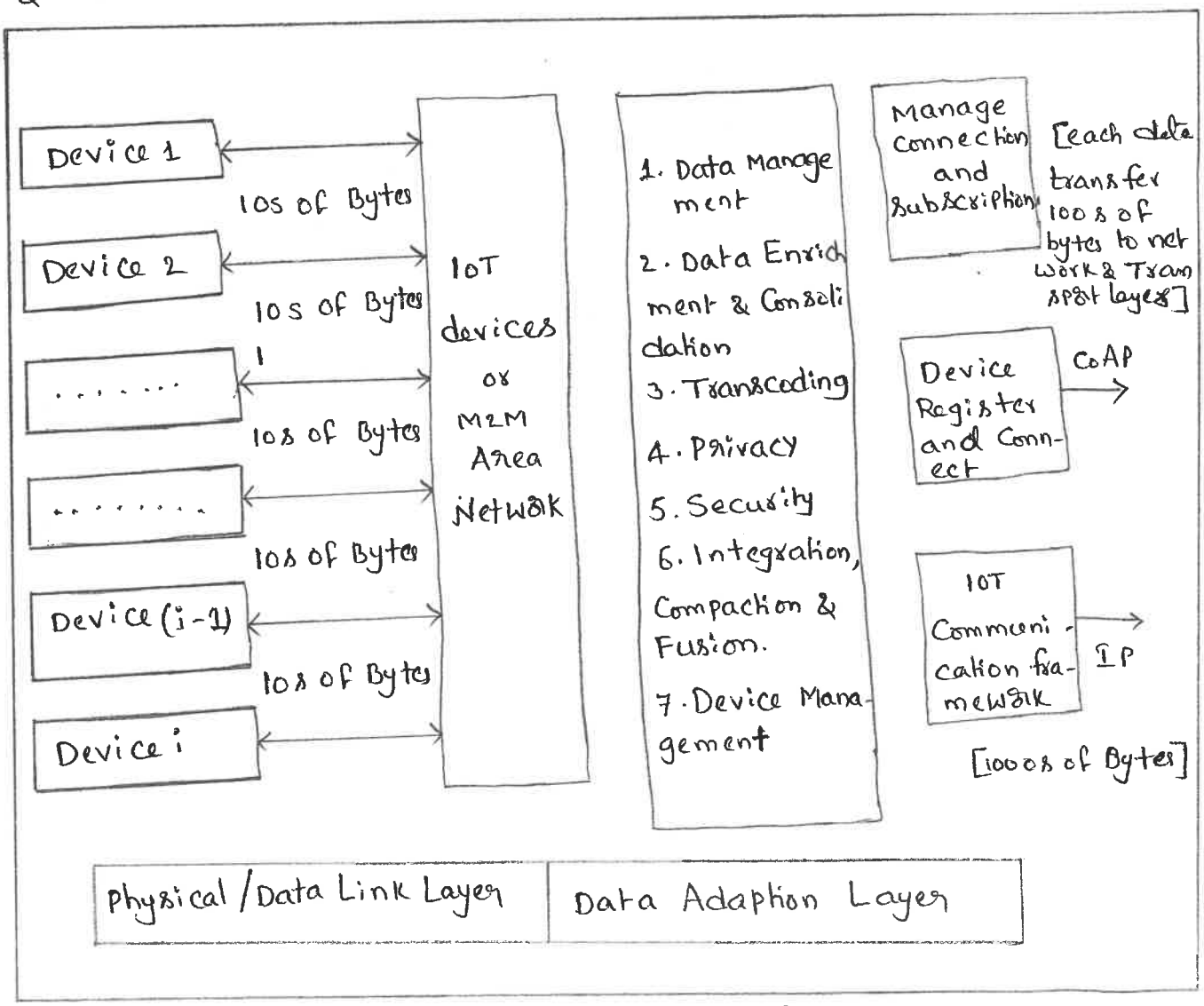


Fig: IoT or M2M gateway Consisting of data enrichment & Consolidation, device Mgt and Comm. Frame Works at adaption Layer

The gateway includes two functions viz. Data Management & Consolidation & Connected device Management.

The following subsections describe frame work for data enrinchment & Consolidation.

Data Management & Consolidation Gateway :-

Gate way includes one or more of the following functions : i.e transcoding & data management.

These are the data management & Consolidation function

- ✱ TransCoding.
- ✱ privacy, security
- ✱ Integration
- ✱ Compaction & fusion.

TransCoding :-

It means data adaption, Conversion & change of protocol, format using software. The gateway renders the web response & messages in formats & representations required & Acceptable an IoT device.

Fa example, use of transcoding enables the message request characters to be in ASCII format at the device

and Unicode at the server. It also Enables the use of XML format.

Transcoding involving formats, data and Code Conversion from one end to another when the multimedia data is transfered from server to mobile TV, Internet TV, VoIP phone or smart phones as the client devices. It also involves Compression & DeCompression.

Privacy :-

Data such as patient medical data, data for supplying goods in a Company from and to different locations, and changes in inventories may need privacy & Protection from Conscious transfer to untrustworthy destination using Internet.

privacy is an aspect of data management & must be remembered while designing of application.

Following are Components of privacy model:

- Device and applications identity-management.
- Authentication.
- Authorisation
- Trust
- Reputation.

A suitable encryption of identification of data source enforces privacy.

Secure Data Access :-

Access to data needs to be secure. The design ensures the authentication of a request for data & Authorization for accessing a response.

End-to-End security is another Aspect. While implies using a security protocol at each layer.

Data Gathering and Enrichment :-

IOT / M2M involves actions such as data-gathering (acquisition), validation, storage, processing, reminiscene (retention) and analysis.

Data gathering refers to data acquisition from the devices / devices network.

Four Modes of Gathering Data :

1. Polling refers data sought from device by addressing the device.
2. Event based gathering refers to data sought from the device on an event.
3. Scheduled interval refers to the data sought from a device at select intervals.

4. Continous monitoring refers to the data sought from a device continuously.

Data Enrichment refers to adding value, security & usability of data.

Data Dissemination :-

Consider the 3 steps for data enrinchment before the data dissemination to network as aggregation, Compaction & fusion.

Aggregation refers to process of joining together present & previously received data frames after removing redundent data.

Compaction means refering/making information shat with out changing the meaning.

Fusion means formatting the information received in parts through various data frames and several types of data; removing redundancy in received data.

Energy Dissipation in Data Dissemination :-

Energy Consumption for data dissemination is an important Consideration in many devices in WPAN and Wireless sensor nodes (WSN). This is due to limitted battery life. Energy is Consumed when performing Computation &

transformations. Higher data rate, the greater will be the energy Consumed. Higher is RF used, greater will be energy used.

Energy effecient Computations can be done by using concepts of data aggregation, Compaction & fusion. Lesser the data bytes Communication, greater the acquisition intervals, and lower data rate.

Data Source & Data Destination :-

ID : Each device & each device resource is assigned an ID for specifying the data of source. & separate ID for destination.

Address : Header field add the destination address.

Data Characteristics, Formats & Structures :-

Data characteristics can be temporal data, real time data, spatial data, real-wold data, proprietary data and big data.

Data receives from devices, formal before transmission on to internet.

Structures implies the ways for arranging the data bytes in sequences with size limit

Device Management Gateway :-

Device Management means provisioning for device ID or address which is the distinct from other resources, device activating, Configuring, registering, de registering, attaching & deattaching.

Device management means accepting subscription for its resources. Device fault management means Course of actions to be followed in case of fault develops in the device.

Open - Mobile Alliance (OMA) - DM and several standards are used for device management. OMA - DM suggests use of DM server which interacts with devices through gateway in IOT / M2M System. DM Server assigning device ID, activating, Configuring to device Services.

DM Server Communicates to gateway in case of low Power loss environment.

Gateway functions for DM are:

- Does forwarding fun when DM Server & device Can interact with out reformatting.
- Does protocal Conversion when the device and DM server use Distinct protocals.

EASE OF DESIGNING AND AFFORADABILITY:-

Design for Connected devices for IoT Applications, services and Business processes Considers the ease in designing the device physical, data-link adaption & gateway layer.

It means availability of SDKs (Software Development Kits), prototype development boards with smart sensors, actuators, Controllers and IoT devices which are low in cost and hardware which embeds and preferably Open source software Components & protocols. Hardware includes the device should embed minimum number or Components and use ready solutions for ease in designing local devices.

Designing also Considers ease as well as affordances for example, RFID or Card. The card has embedded micro-Controller, memory, OS, NFC interfaces, access point-based device activation.

A Wireless sensor uses, for example a mobile terminal which is a low Cost device with open-source OS & software.

Devices of smart homes and cities use ZigBee IP or BT LE 4.2 due to their affordability, ease of designing.

A design may add to the complexity, Connected devices may add complexity in the form ensuring data transfer to trusted destinations using encryption tools.

ETSI M2M Domains and High-level Capabilities:-

A domain specifies the functional areas. High-level architecture means architecture for functional and structural view. The diagram shows ETSI M2M domains & architecture & high level capabilities of each domain. It also shows that archi correspondence with six-layers modified OSI model as well as 4 layers of ITU-T reference model.

ETSI network domain has six capabilities & functions:

1. M2M application
2. M2M service capabilities
3. M2M management functions
4. Network management functions
5. CORE network
6. Access network.

The ETSI device & gateway domain following fun:

- Gateway between M2M area network., and CORE and access network, processing M2M service.
- M2M Service capabilities & applications
- M2M area network
- M2M devices.

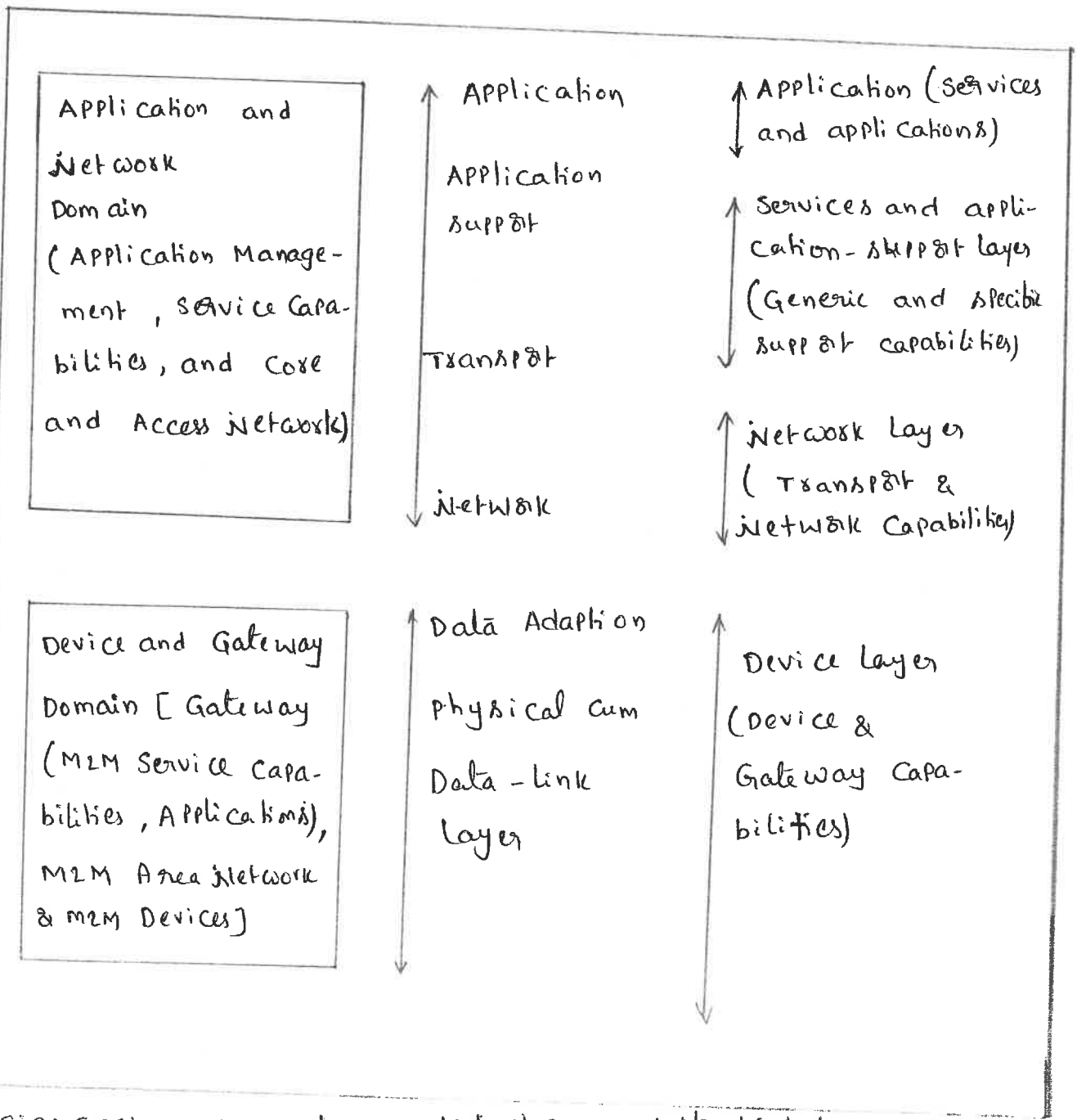


Fig: ETSI M2M Domain architecture and its high level capabilities, and its correspondences with six layers of modified OSI & 4 ITU-T

UNIT - II

SHORT ANSWER QUESTIONS :-

1). What is Data Enrichment ?

2). What is Data Consolidation ?

3). What is device management ?

4). What is Affordability ?

5). Define types of Wired Communication Technology.

6). Define types of Wireless Communication Technology.

7). Define ZigBee IP .

8). What is RFID.

9). What is NFC

10). Give any 3 differences between Wired & Wireless Communication Technology.

11). Give a short notes on ETSI M2M Domains.

12). Write a short notes on ITU-T reference model.

13). Write a short notes on Wi-Fi , RF-Transceiver, UART Serial Communication .

14). Define GPRS/ GSM Cellular networks .

15). Write any 3/4 differences between BTLE & WLAN protocols.

ESSAY QUESTIONS :-

1). Explain in detailed about IoT / M2M Layers & Design standardization. with neat Sketch.

2). Briefly discuss the Concept of ETSI M2M Domains and high level capabilities.

3). Explain in detailed about Communication Technologies (wired & Wireless)

4). Explain in detailed about the Concepts of Data Enrichment, Data Consolidation & Device management at gateway with neat sketch.

5). Discuss in detailed about Ease of designing and Affordability.

UNIT - 3.

Design principles for the Web Connectivity for Connected Devices, Web Communication protocols for Connected devices, Message Communication protocols for Connected Devices, Web Connectivity for Connected - Devices.

OBJECTIVES :

→ Design principles for Web Connectivity.

→ Connected Devices for Web.

→ Web Communication protocols

→ Connected Devices for Web protocols

→ Message Communication protocols for Connected Devices

→ Connected Devices for Web Connectivity.

Design Principles for Web Connectivity :-

Introduction :-

An IoT / M2M device network gateway needs Connectivity to Web servers. A Communication gateway enables Web Connectivity, while IoT / M2M specific protocols & methods enable Web Connectivity for a Connected devices network. A server enables IoT device data accumulation (& storage).

Application or App refers to a software for applications such as creating and sending an SMS, measuring and sending the measured data, receiving a message from a specific sender etc.

Application programming Interface (API) refers to a software Component, which receives message from one end.

For example, Consider an API for a Parcel tracking application. The API displays the fields for user inputs required for tracking, accepts the user i/p's, ~~required for tracking, accepts~~ displays a 'wait message' to the user and sends i/p parameters to an application at a server. The application in turn displays response from application (or) server.

A web service refers to a servicing software which uses web protocols, web objects (or) web sockets.

Object refers to a collection of resources; For example collection of data & methods to operate on that data.

object model is defined for usage of objects for values, messages, data or resource transfer & creation of one or more object-instances.

Communication gateway is one that functions as communication protocol translator for provisioning communi-capabilities.

Client refers to software object which makes request for data, message, resources.

Server defined as a software which sends a response on a request. The server also sends message.

web object is one that retrieves a resource from the web object at other end using a web protocol.

Broker denotes an object, which arranges the communication between 2 ends.

proxy refers to an application which receives a response from the server for usage of a client or application

and which also receives a response from the client for response retrived.

Communication protocol defines the rules & Conventions for Communication between networked devices & between devices. The protocol includes a mechanisms for devices or systems to identify make Connections with each other.

Web protocol is a protocol that defines the rules & conventions for Communication between the web server & web clients. It is a protocol for web Connectivity of web objects, clients, servers & intermediate servers or firewalls.

Firewall is one that protects ther servers from unauthentic resources.

A header Consists of set of words. The words Contain the information and parameters about the processing at Communication layer.

A state refers to an aspect related to some one (or) something at a particular time. State in reference to data interchange between web objects.

Resource denotes something that can be read, written (or) executed. A path specification is also a resource.

Each resource instance has a resource. The resource is a atomic operation/information which usage (or) usable during Computations.

Path denotes a navigation between 2 ends when accessing resource.

Universal Resource Identifier is generally used for saved resources, such as Contacts or address book.

Universal Resource Locator is generally used for retrieving a resource by client.

Datagram refers to a limited size data (2^{16} bytes). It is used for stateless Connectionless transfer from web objects.

REST (Representational State transfer) is a software architecture referring to ways of defining the identifiers for resources.

REST also refers usage of resource defined types when transfering objects between 2 ends.

MIME refers to the type of files that are widely used on the internet by web objects, applications.

Hyperlink refers to an specification of URL for a resource path.

Hypertext means text embedded with hyperlinks.

Web Communication Protocols for Connected Devices:-

Data of Connected devices routers over the Web into 2 types of Communication environments.

The environments are:

★ Constrained RESTful Environment (CORE):

IOT or M2M Devices Communicate between themselves in a LAN. A device typically sends or receives 10s of bytes. The data gathered after enriching & Consolidation from a no. of devices Consists of 100s of bytes. A gateway in the Communication framework enables the data of networked devices that Communicate over the internet using REST software.

→ Devices have the Constraint in the sense that their data is limited in size Compared to when data interchange between web clients & servers using HTTP, TCP & IP.

→ Another Constraint is for data routing when Routing Over network of low power and (data) Loss (RLL). RoLL network is a wireless network with low power transceiver.

★ Unicontstrained Environment :-

Web applications use HTTP and RESTful HTTP for Web client & web server Communication. A web object consists of 1000s of bytes. Data sockets over IP networks for Internet. Web applications & IP use IP & TCP protocols.

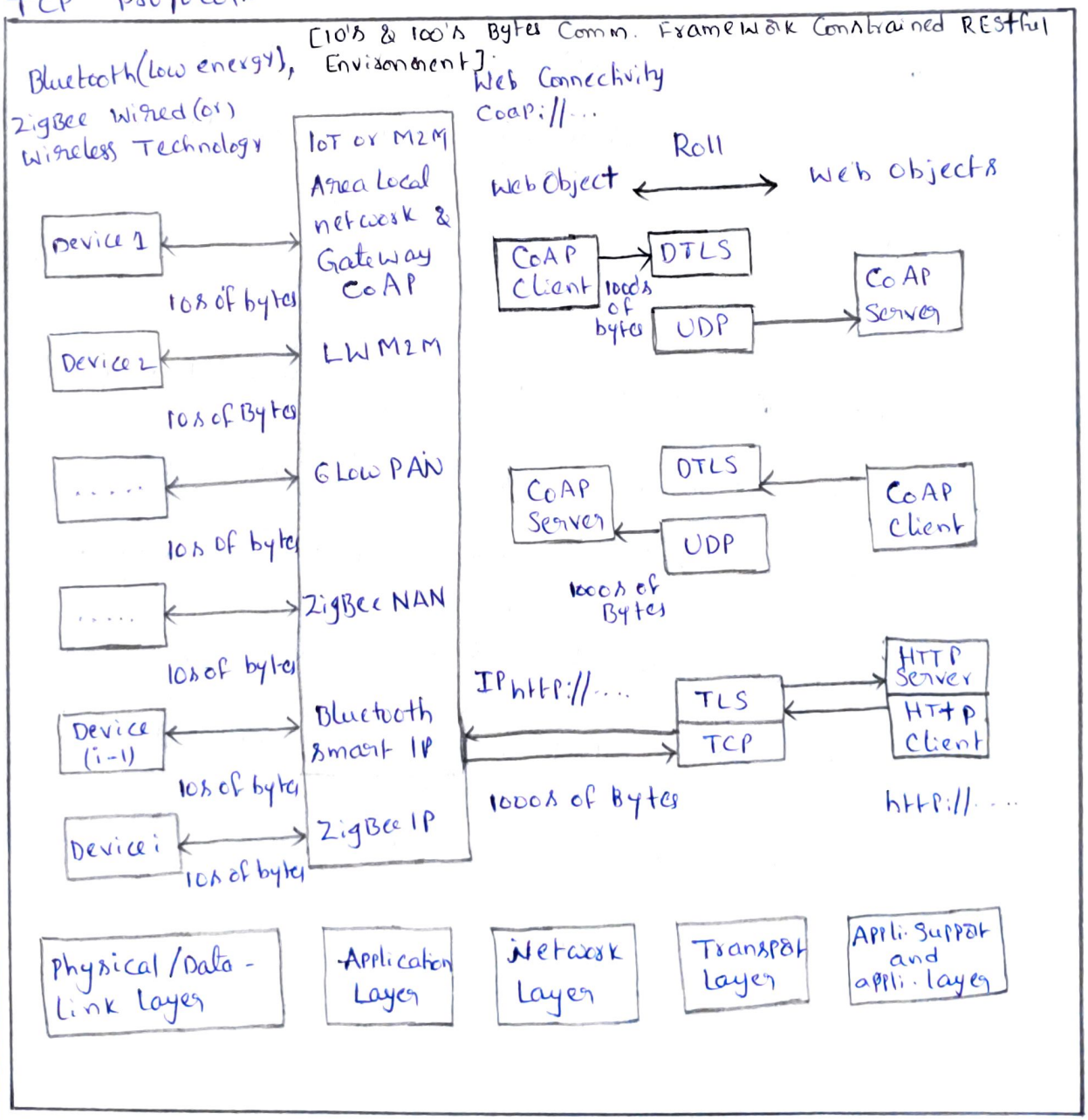


Fig: IoT or M2M Device local network Connectivity & Web Connectivity in Constrained & unConstrained RESTful HTTP environments using Communication protocols.

The figure shows devices local area network Connectivity, Web Connectivity in Constrained & unConstrained RESTful environments & protocols for Communications.

Assume that a web objects refers to a web client or web server. The web objects Communicates a request of client or a respons of server.

Fig shows the following;

- Assume i- devices (1, 2, ..., ith) Connected devices network, and local network having Connectivity between the devices at physical /data link and adaption layer. (LHS)
- Communication between web objects (RHS).
- IETF CoRE specifications, which include CoAP and UDP.
- Web objects protocols for sending a request (or) response
- Transport layer protocals used are Datagram Transport Layer Security (DTLS) and UDP. Data between web objects route using ROLL network specifications of IETF.
- 100s of bytes Communicate between the IoT web objects.
- 1000s of bytes of Communicate between HTTP Web objects using certain protocals for sending request (or) response

Constrained Application protocol :-

IETF recommends Constrained Application Protocol (CoAP) which is for CoRE using ROLL Data network.

CoAP features are :

- An IETF defined application - support Layer protocol.
- CoAP web - objects Communicate using request/response interaction model.
- It uses object - model for the resource & each object can have single or multiple instances.
- Each resource can have single or multiple instances.
- An object or resource use CoAP, DTLS & UDP protocols.
- Supports the resource directory and resource - discovery functions.
- The resources identifiers use URIs follow coap ://
- Has small message header, 4 bytes are for ver (version), T (message Type), TKL (Token Length), Code (request method or response code), message ID 16-bit identifier, token.
- CoRE Communication is asynchronous Communication over ROLL.
- Use of REST : The access by a CoAP object or its resource is thus using :

→ the URI

→ a subset of MIME Types

→ A subset of response Codes which are used for an HTTP object or resource.

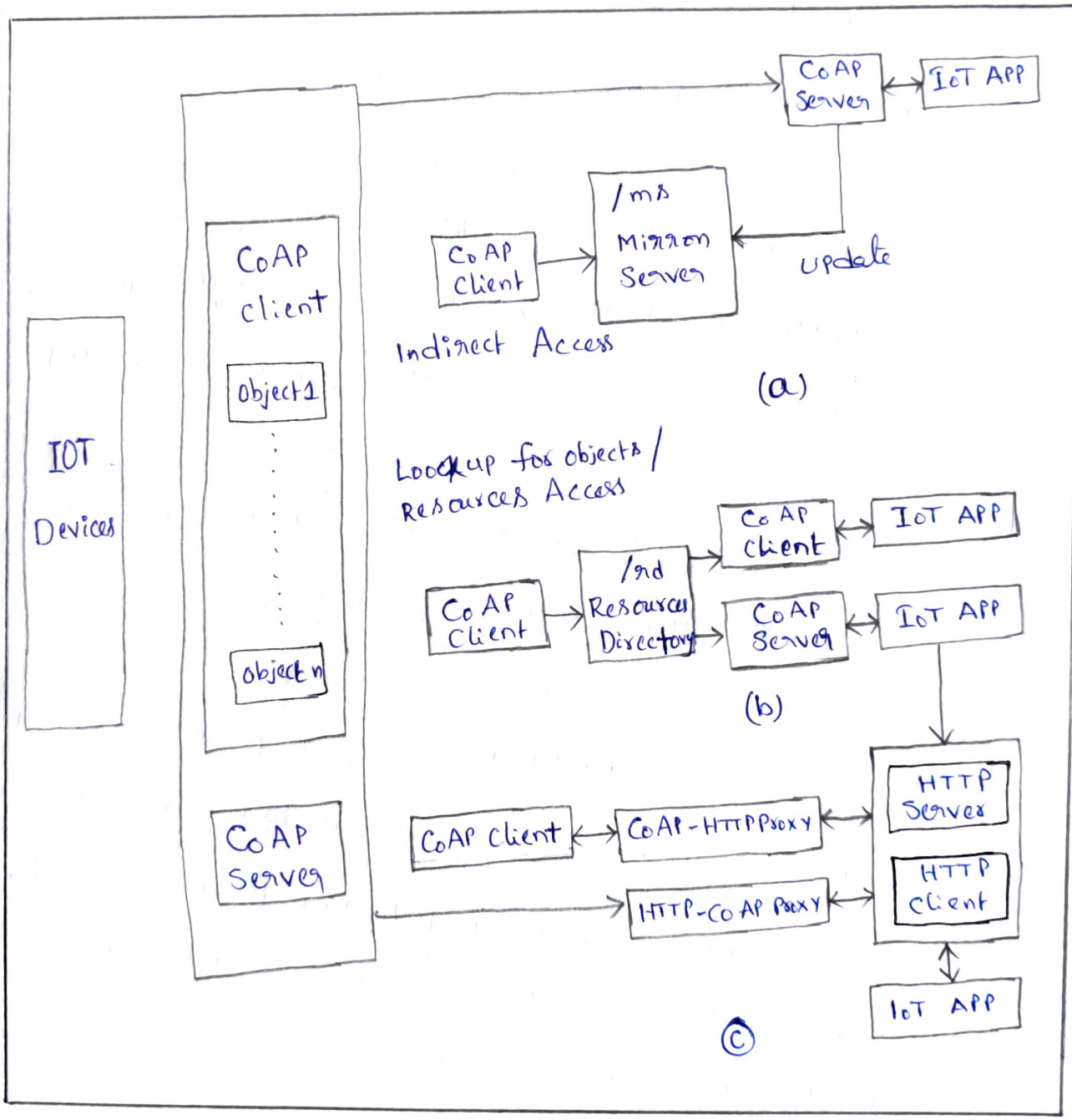


Fig: IoT or M2M devices local network Connectivity & Web Connectivity in Constrained

a) Direct & indirect access of CoAP client objects to CoAP Server.

b). CoAP client access for lookup of object or resource using a resource directory

c). CoAP client & Server access using Proxies.

CoAP Client Web Connectivity:

A proxy is an intermediate server, which accepts a request from client & sends the response to the client using protocol. It also passes request to the server and accepts a response from server using the same or an other protocol. HTTP - CoAP proxy accepts requests from HTTP client using HTTP protocol and sends the request to the server using CoAP protocol. CoAP-HTTA proxy accepts request from CoAP client using CoAP protocal & send request to HTTP protocol.

~~Fig 3.2 (c) show~~

The above figure (C) shows CoAP clients & server access using proxies.

Transport Layer Security (TLS) earlier known as (SSL) Secure Socket Layer is the protocol used for securing the TCP-based Internet data interchanges. DTLS is the TLS for datagram. The features of DTLS are:

- DTLS provisions for 3 types of security services — integrity, authentication and Confidentiality.
- DTLS protocol derives from TLS protocol and binds UDP for secured datagram transport.

- DTLS is well suited for securing applications.
- A part of DTLS is OpenSSL repository open ssl-0.9.8 security based on PSK, RPK and certificate.

Secured use of a key for client Authentication:-

PSK stands for pre-shared key and is a method of securing using a key to authenticate a client. The key contains up to 133 characters. PSK method generates a unique encryption key for each client. A PSK is a symmetric key with out forward secrecy. symmetric key means both end 1 and 2 use the same key, K12 for encryption and decryption.

Private key refers to a key agreed for usage of data encryption between a pair of sender & Receiver. The key is kept private between the two. Sender & receiver can be objects, applications, web services.

RPK stands for Random Pair-wise keys, which also stands for Raw Public key. [meaning only the private (or) public & other end using RPK uses (K2 & KP)] are asymmeric

Public key refers to a key, KP which an intermediate server or trusted entity.

X.509 certificate is a certificate with a chain of trusted based on an authorised certificate-Authority (CA) & public key Infrastructure (PKI). The sender submits a document once to the CA for digital signature, & CA issues a certificate of verification of document.

Light Weight M2M Communication protocol:-

(LWM2M) Light Weight M2M Comm. protocol is an application Layer protocol specified by Open Mobile Aliance (OMA) for transfer of service data. It finds application in M2M. It enables functionalities for device management in sensor networks. Communication protocol 'light weight' means that does nit depend on call to system resources.

Light weight presently means data transfer format between client and server are binary ~~data~~ and has Tag Length value (TLV) or java Script Object Notation (JSON) batches of object arrays & transfer upto 100s of bytes.

The protocol enables Communication between LWM2M client at IOT device & an LWM2M server at M2M

application & Service capability layer.

The below figure shows M2M devices LAN Connectivity. It shows Constrained devices network Connectivity with M2M application & Services using LWM2M OMA standard specification of LWM2M.

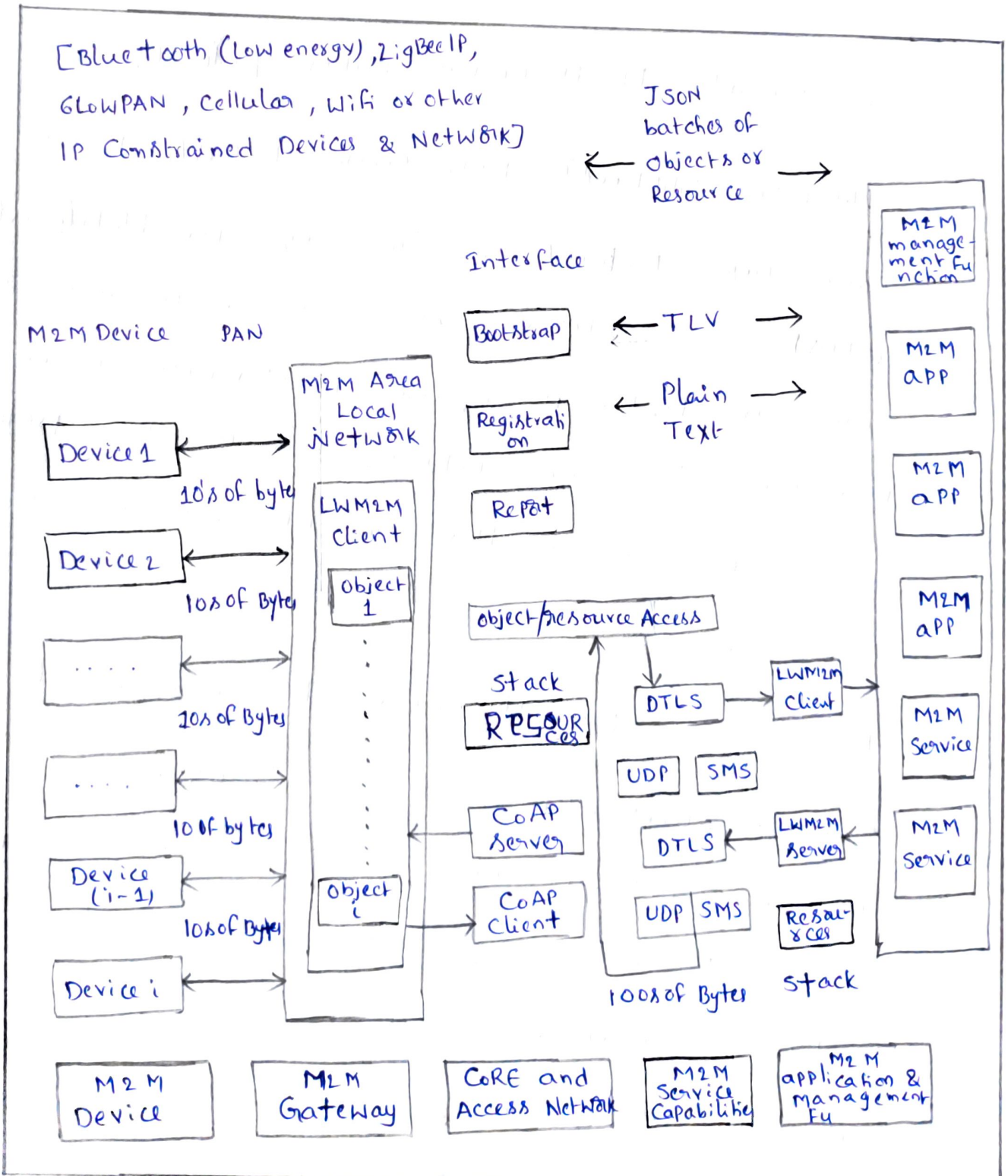


Fig: M2M Devices LAN Connectivity, and Constrained devices network Connectivity with M2M applications using LWM2M OMA Standard.

Assume i number of M2M devices.

→ Local M2M Constrained devices use.

Ex: Bluetooth, CoRE, ROLL, NFC etc, ZigBee

→ 10s of bytes Communicate between a device & the PAN.

→ Communication between LWM2M objects (RHS).

LWM2M Client refers to object instances as per OMA standard. LWM2M protocol.

→ CoRE network, for example, 3GPP.

→ Communication from an object instance using interface functions.

→ use of the CoAP, DTLS, and UDP protocols by object (or) resource.

→ 100s of bytes Communication/ Communicate between objects at the client or server for plaintext or JSON.

LWM2M specifications & features:-

→ An object or resource use CoAP, DTLS & UDP or SMS protocols for sending Request or Response.

→ use of plain text for a resource. (or) JSON using data transfer.

→ An object or its resources accessing using an URI.

→ An interface functions are for - bootstrapping;

→ Use of object model for resources and each object can have single or multiple instances.

→ Each resource can have single or multiple instances.

→ OMA or other standard specifying organisation defines the LWM2M objects for usage in M2M Communication.

→ Organisations can register the other LWM2M objects & resources with OMA.

→ M2M management functions can be M2M service Bootstrap function (MSBF) for Credentials of device & gateways & M2M authentication server (M2MAS / MAS).

JSON Format :-

Features of a JSON are:

- JSON is an open - standard format used primarily to transmit data between server and web application. As an alternate to XML.
- Originally derived from a Java Script Scripting Language, JSON is now a language - independ data

format, the coding of which can be in java or C or another programming languages used for parsing & generating JSON data.

Problem :-

Give a JSOM object example in JavaScript Using object definitions.

Solution :-

Following are the codes as per OMA specifications for LWM2M objects. The code is at :

```
{
 "0" :
        {
          "id" :0,
          "name": "LWM2M"
          "instancetype" : "Multiple"
          "Mandatory" : true,
          "description" : "this is LWM2M Description"
          "resourcedefs": {
          "id" :0,
          "instancetype": "single"
          "type" : "String",
```

"description" : "Unique identifies of LMW2M"

"1" : {},

{

"id" : 1,

"name" : "Boot strapServer",

"Operation" : "-",

.

.

.

}

"9" : {

"id" : 9,

"name" : "LWM2M"

"operation" : "-"

"type" : "integer"

"range" : " ",

"units" : " ",

"description" : "MISSON"

},

Ex : Describes the Codes used for data transfer in Java using JSON.

Give a JSON object example in Java

Solution:- JSON Object is a subclass of Java.util.HashMap.

Assume sensor name is SensTemp1, temp is int, and no. of sensors is a double. is_newSensorAdded() is a method which returns Boolean data type.

Following is the code in Java:

```
// import org.json.simple.JSONObject;
    JSONObject obj = new JSONObject();
        obj.put("name", "sensetemp");
        obj.put("tempa", new Integer(100));
        obj.put("no.of sensors", new Double(100.2));
        obj.put("is_newsensor", new Boolean(true));
    StringWriter out = new StringWriter();
        obj.writeJSONString(out);
    String jsonText = out.toString();
    System.out.print(jsonText);
```

Result : {"Name": "SensTemp1": "Temp": 26"
numberofsensors: "10000, "newSensAdded": null, }

Tag Length value format :-

In TLV format the first 2 bytes are used to identify the parameter, the 1st & 4th bytes indicate

the length of actual data.

Example code for data transfer in TLV format:

Problem:-

Give an example of TLV format messages.

Solution:-

Assume that an application service uses CoAP for which code is "200". Let engine parameter rpm is assigned ID = 125 and velocity assigned ID = 126. Format in TLV can be as follows:

</engine> ; EngObj ; id = "20" ; ct = "200",

</engine/rpm> ; EngObj ; id = "125",

</engine/velocity> ; EngObj ; id = "126".

MIME TYPE:-

Generic-type files are application/octet-stream; however, MIME-Type files are used in applications & Services.

Features of MIME:-

→ An internet standard which is for destination of Contents of different files.

- An internet standard which extends the SMTP format of email to support the text.
- Supports messages with multiple parts.
- Initially designed for mail.
- List of MIME type file is exhaustive.

MESSAGE COMMUNICATION PROTOCALS FOR CONNCTED DEVICES :-

A device / node / end-point / client / server sends and Receives messages. A Communication module includes a protocol handler, message queue and message cache. The protocol handler functions during transmission and reception of messages according the Communication protocols for these Actions. Message queue forms to keep a message untill it is transmitted towards its destination. Message check keeps in incoming message untill it is saved into module. A device message queue inserts (writes) the message into the queue and deletes (reads) the message.

The following section explained used by Message Communication protocols.

Terminology:-

Request/Response (client/server):-

A request/response message exchange refers to an object (client) requesting for resources and an object (server) sending the response(s). The objects, for example, HTTP Objects may use the REST Functions.

When sending a request, a protocol adds header words. Each header has fields. Each field interprets the request at the received objects.

Publish/Subscribe (pubsub):-

Publish/Subscribe message exchanges differ from request/response. A service publishes the messages.

Ex:

Weather information service publishes the messages of weather reports for potential receivers. A group Controller publishes the message.

A Publish / subscribe messaging protocol provisions for publication of message & reception on subscription (put & Get methods) by registered or authenticated devices.

Publication may be for measured values, for state information or resources of one or more types.

Subscription is for a resource-type. A separate subscription is required for each resource-type..

Resource Directory :

Resource Directory (RD) maintains information & values for each resource type. A resource of a resource type access from RD using URI for resource.

Resource Discovery :-

This service may advertise (publish) at regular intervals, the availability of the resources or types of the resources available & their states.

Registration / Registration update :-

Registration means a receiver register with a Service, such as an RD Receiver. When one or more end points or devices or nodes registers, then that devices gets

the access to resources and received published message.

Security Considerations may require authentication of both ends before registration. A separate registration is required for each end points.

"Registration update" means updating one or more end points (or) devices or nodes.

Pull (Subscribe / Notify) Data :-

Pull means pulling a resource, value, message or data of resource-type by registering and subscribing. pull may be using GET (or) initiating OBSERVE. The server maintains state information for a resource & notifies on change of state.

Polling & Observing :-

polling means finding from where new messages would be available or whether new messages are available or updates are available (or) whether information needs to be refreshed. When message store at data-base server then polling can be done by a client.

A state may refer to a connection or disconnection, sleep, awake, created, alive, old values persisting or update with new values (GET + OBSERVE). Observing means looking for change.

Push (public/subscribe) Data :-

Push means a service that pushes the message. Push is effient Compared to polling, particularly when notifying or sending alerts. This is because there can be many instances when polling returns no data. pull is efficient Compared to Polling (or) PUSH.

Message Cache :-

Cache means storing when available and can be used later on when required. Messages cache is useful in an environment of short or prolonged disConnections of a service. A message accessed one or more times.

Message Queue :-

It means storing the message in a sequence from devices or end points, so that when Connection changes then message can be forwarded. forwarding is done using FIFO.

Information / Query :-

The method is that an object (client) requests information using a query while another end-objects (Server) responds by replying to the query. The responding application processes the query.

Communication protocols :-

These are the protocols used in message Communication.

- ★ CoAP - SMS and CoAP - MQ
- ★ MQTT protocol

★ CoAP - SMS and CoAP - MQ :-

M2M or IOT devices uses SMS quite frequently. SMS is identified as the transport protocol for transmission of small data (upto 160 characters). It is used for Communicating with GSM/GPRS mobile service.

M2M or IOT devices uses message queuing quite frequently due to ROLL Envisonment and Constrained devices or Connection - breaks for long period.

CoAP - SMS :

It is a protocol when CoAP object uses IP as well as cellular networks and uses SMS. It is alternative to UDP - DLTS over Roll for CoAP object message & when using cellular Communication.

SMS is used instead of UDP + DTLS by a CoAP client or server. A CoAP client Comm. to a mobile terminal (MT) end point over the General Packet Radio Service (GPRS), High Speed Packet Access (HSPA) or Long terminal Evolution (LTE) network using CoAP - SMS protocols.

Following is IETF recommended terminology for use:

- SMS-C : SMS service centre
- SMS-SP SMS service provider.
- CIMD : Computer interface to message distribution.
- MS : Mobile Station at cellular network functioning as CoAP client or Server.
- MOS : Machine or IoT device as CoAP Client.
- MT : Machine or IoT device as CoAP Server.
- SMPP : Short message Peer-to-Peer for CoAP Data.

CoAP - SMS :-

It is a protocol when COAP object uses IP as well as cellular networks and uses SMS. It is alternative to UDP - DLTS over ROLL for CoAP object message & when using cellular Communication.

SMS is used instead of UDP + DTLS by a CoAP client or server. A CoAP client Comm. to a mobile terminal (MT) end point over the General Packet Radio Service (GPRS), High Speed Packet Access (HSPA) or Long terminal Evolution (LTE) network using CoAP - SMS protocols.

Following is IETF recommended terminology for use:

- SMS - C : SMS service centre
- SMS - SP SMS service provider.
- CIMD : Computer interface to message distribution.
- MS : Mobile Station at cellular network functioning as CoAP client or Server.
- MOS : Machine or IoT device as CoAP Client.
- MT : Machine or IoT device as CoAP Server.
- SMPP : Short message Peer-to-Peer for CoAP Data.

- SS7 : Signaling Service Protocol.

The CoAP - SMS features :-

★ An URI used as coap + sms :// in place of coap :// .
For example, URI may be coap + sms :// telNum/carLoc ationObject /latitude for latitude of location of a car after location of object measures location parameters using the GPS.
URI used when sending the SMS to specified telephone number - telNum.

★ A CoAP message encodes with alphabets for SMS Communication. An SMS Consists of 160 characters in 7-bit encoding.
Max. length for CoAP message is thus 140 B (= 160 x 7-bits / 8-bits) when an SMS-C support 8-bit encoding.
A CoAP message is thus 70 (= 160 x 7-bits / 16 bits) when SMS-C supports 16-bit encoding and multilingual alphabets.

★ CoAP end points have to work with a subscriber Identify Module (SIM) card for SMS in Cellular networks. The points are addressed by Mobile Station ISDN (MSISDN) number.

- Does not Support multi-casting.
- Two additional options are Response-to URI-HOst (RUH) and Response-to-URI-Port (RUP) which make initiating CoAP client aware of the presence of alternative interface CIMD & SMPP and UCP/UMI.

→ RUH starting size 0 to 255 B.
→ RUP of 2B with default port number 5683.
→ IANA (Internet Assigned Number Authority) registered TBD as CoAP option Numbers for registry.

- Data interchange sequence as follows:

An MS/CoAP Client sends a SMS request (SMS-SUBMIT) to SMS-C; SMS-C reports using SMS-SUBMIT-REPORT; SMS-C sends SMS (SMS-DELIVER) to MS/CoAP server; the server reports using SMS-DELIVER-REPORT; SMS-C sends -SMS-STATUS-REPORT to client.

- Authentication of a client by server provides the security.

The next page figure (a) shows a CoAP request or response Communication to a mechine, IOT device or MT.

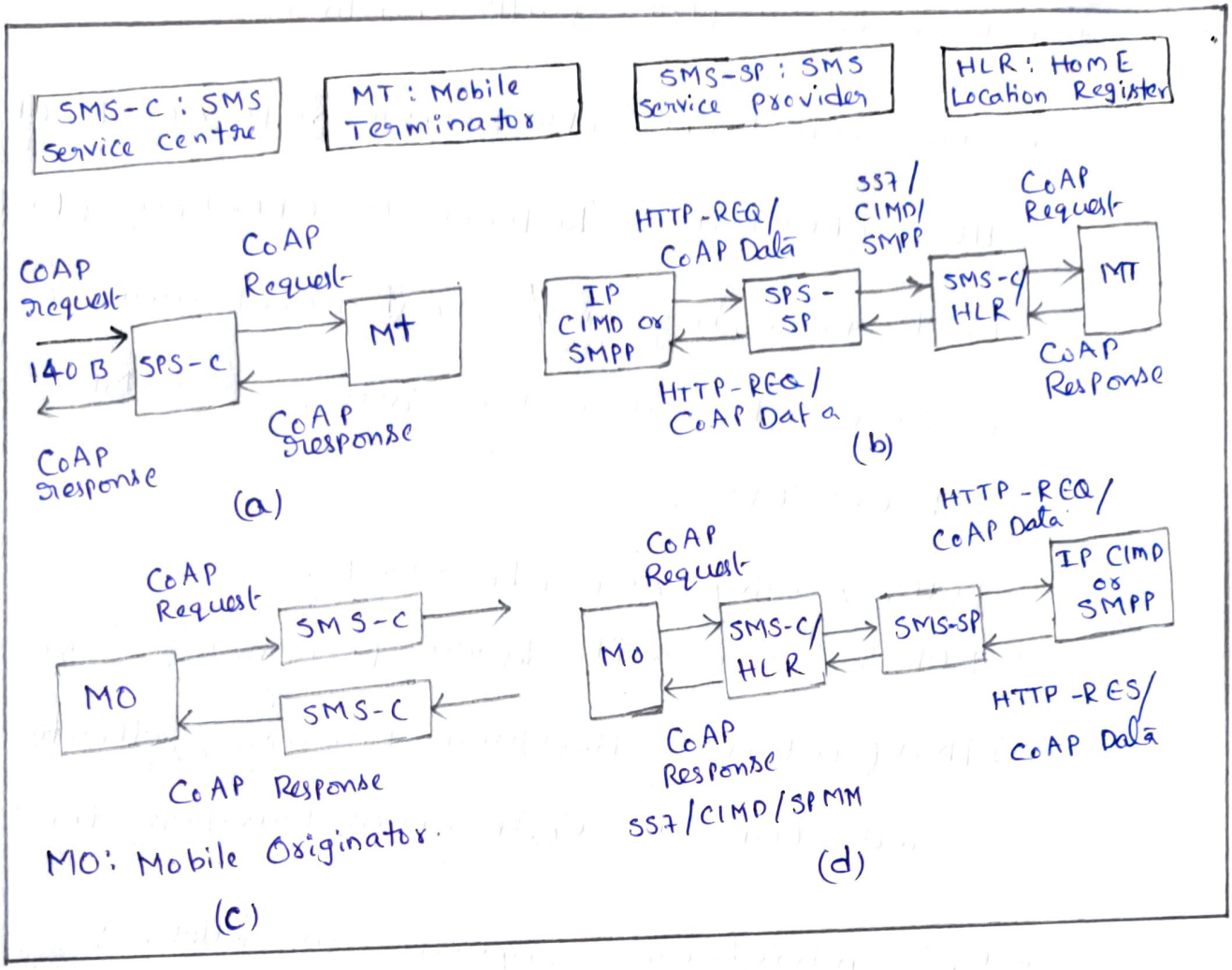


Fig :- (a). CoAP Request or Response Communication to machine or IoT device or MT (mobile terminal)

(b). A Computer or machine interface using IP Communication to a mobile service provider for data interchange with terminal

(c). A machine or IoT or mobile Origin Comm. of CoAP request or response Communication

(d). An Origin Communication using SS7/CIMD/ SMPP with Computer interface using IP Communication.

A CoAP client sends request to SMS-C which transmits the request to an MT. A CoAP server sends response to SMS-C which transmits response to Client.

The figure(b) shows a Computer or machine interface using IP for sending request or receiving the CoAP data or HTTP request (REQ) to mobile service provider using SMPP or CIMD for data interchange. The service provider Comm. using SS7 or CIMP or SMPP with machine or IoT device between node SMS-C

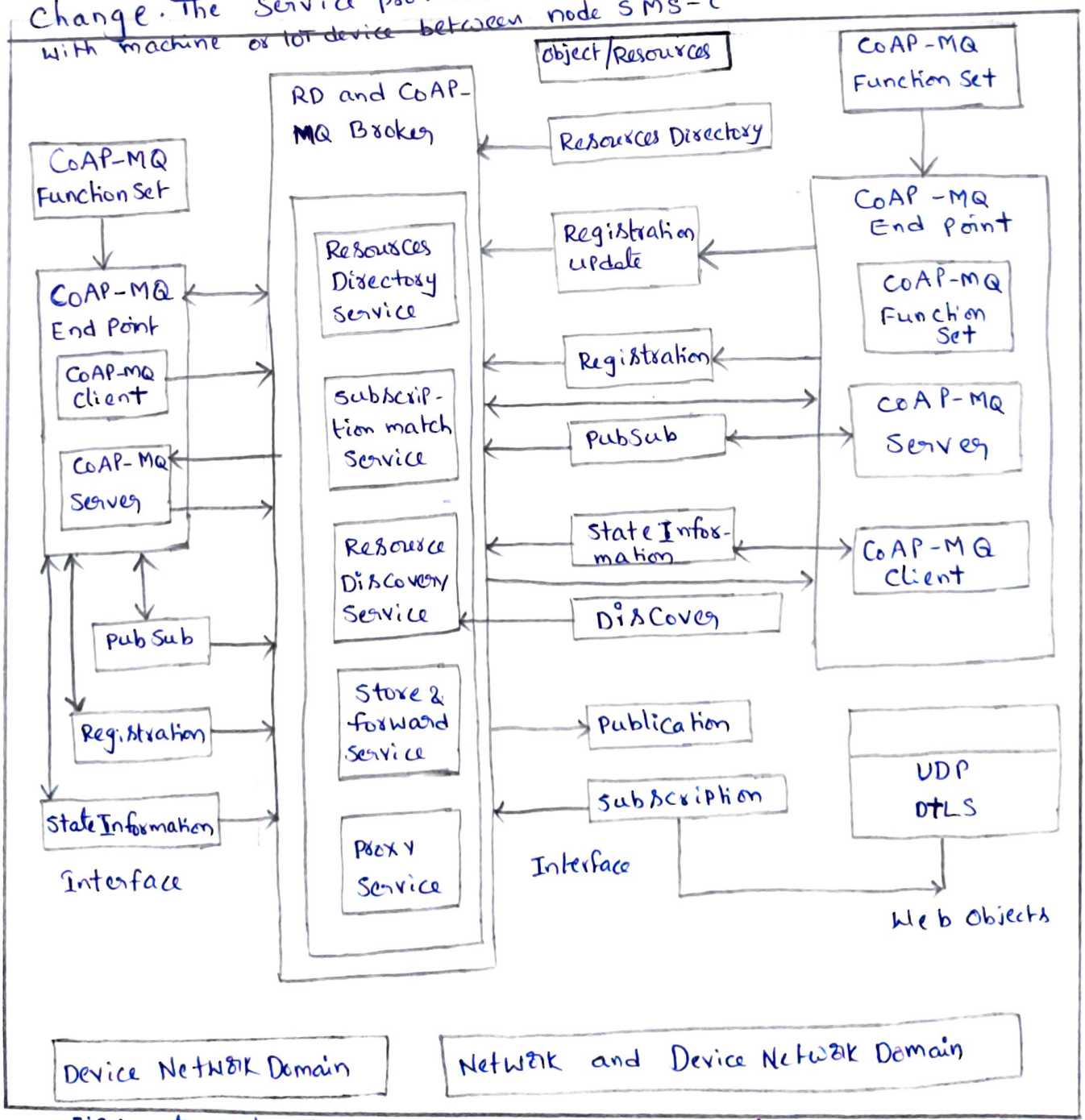


Fig 1: Data interchange between CoAP-MQ endpoints, CoAP-MQ clients, CoAP-MQ Server through CoAP-MQ broker & its servers.

~~The data interchange~~

The SMS-C Communicates that to CoAP-MQ BROKER using SS7 or CIMP or SMPP. SMS-SP receives request or sends response to machine or IoT.

Figure (c) shows a CoAP request or response Communi- from a mechine, IoT device IoT (or) MO. A CoAP client sends request to SMS-C with transmits the request. A CoAP Server sends the response to SMS-C with trans- mits to client.

Figure (d) shows Computer (or) machine interface using IP for receiving request or sending response (RES) as CoAP data or HTTP REQ mobile service provider using CIMD for data exchange. The SMS-C Communi Cates that to CoAP-MQ BROKER Using SS7 or CIMP. SMS-SP receives requests & Sends response to machine

CoAP-MQ:

It is a message queue Protocol using a broker & RD. Roles of CoAP end Points have roles as a Client & Server.

The Previous Page Figure 1 shows data inter Change between CoAP-MQ end Points, CoAP-MQ clients, CoAP-MQ Servers through CoAP-MQ broker & its Services.

Fig 1 shows CoAP-MQ Servers Provisioning for resource-

Subscription, store from publisher. The server also provisions for forwarding to subscriber & proxy service. The Fig 1 also shows that RD services are resource discovery, directory & Object registration service. The Object Comm. using CoAP client & server protocols & CoAP web objects using DTLS as Security protocol & UDP for CoAP APIs.

MQTT Protocol :-

MQTT (Message Queing Telemetry Transport) is open source protocol for M2M / IoT Connectivity. IBM created it & then donated to M2M 'Paho' project of Eclipse.

The VERSION is MQTT-SN V1.2. Sensor networks & non-TCP/IP networks, such as ZigBee can use the MQTT-SN. MQTT-SN is also publish messaging protocol.

Fig 2 denotes messages interchange between M2M / IoT device objects & web objects using MQTT broker.

Fig 2 shows MQTT - broker subscription, subscription match, store & forword, last good message retention & keep message alive. Services.

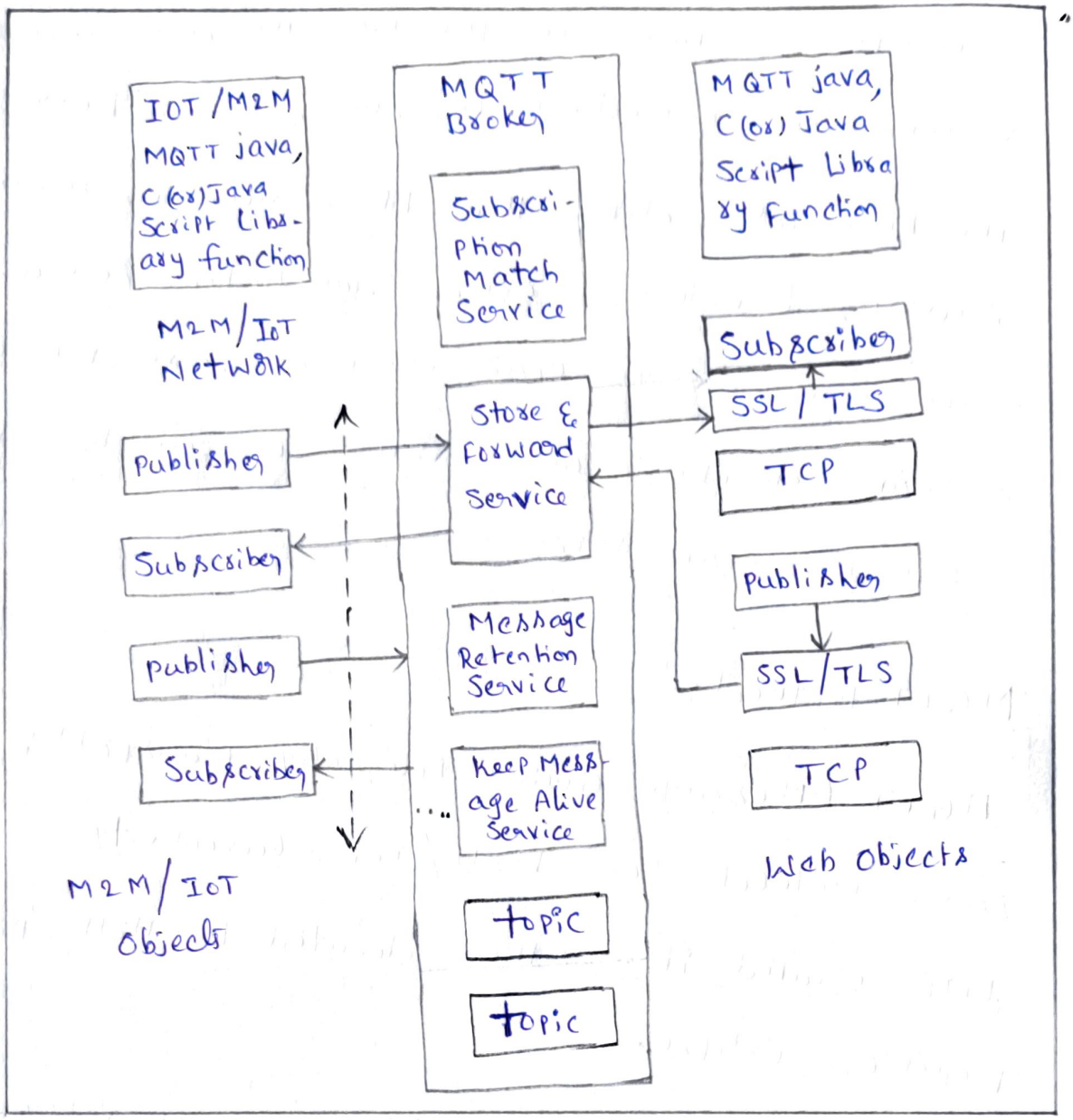


Fig 2 : Message interchange between M2M/IoT device objects & Web objects using an MQTT broker

The above figure also shows that device objects use MQTT java, C or java Script library functions. The obj. Commu. using Connected devices network protocols such as ZigBee.

MQTT Broker does following :

- Functions as a server node capable of storing messages from publishers & forwarding to clients.

- Receives topics from publishers.
- performs store-and-forward function, store topics from publishers & forward to subscribers.
- Recovers subscriptions on reconnect after a disConnection, unless the client explicitly disConnected.
- Receives subscriptions from clients on topics.
- Acts as a broker between publisher of topics & their publishers.
- Find client disConnection until DISCONNECT message receives, keeps message alive till disconnection.
- Authentication by Usesname / Password.
- support from intellegent & business Analyst server & other servers. Through MQTT Server.

XMPP (Extensible Messaging & Presence Protocol):-

XML:

XML is an open-source IETF recommended language.

XML widely used for encoding messages & text.

A text element in XML document can correspond to data, message, alert, notification obj, Command, method.

An XML tag specifies the type of encoded entity in an element. A tag may also associated an attribute (s). The interpretation & use of text with in tag pair depends on parser & associated application. The parser uses XML file as input. The application uses output from parser. The parser application may be / can be java, C# or any other programming language.

XMPP :

It is an xml based specification for messaging & presence protocols. XMPP is an open source protocol accepted by IETF.

XMPP is extensible - XSF (XMPP standard foundation) develops & publishes the 'xeps' (XMPP extension protocol). The 'xeps' enables addition of features & new applications. List of Xmpp xeps for web objects is quite long.

Examples of xeps are :

- Xep - Data-forms Format

- XEP - MUC
- XEP - publish - subscribe & Personal Eventing protocol.
- XEP - file transfer
- XEP - jingle for voice & video.

XMPP - IoT XEPs extends the use of XMPP to IoT & M2M messaging.

Features of XMPP:

- XMPP uses XML
- XML elements sends in the open - ended <stream> with tag <stream> & Corresponding end tag </stream>.
- Three basic types of XMPP stanza as (elements) are:
  * Message
  * Presense
  * iq
- Extensiblity to Constrained environment messaging & protocol as well as IP networking messaging.
- Extensibility of request - response (cli/server) architecture to iq (info through query), PubSub messaging, chat room MUC messaging & other architecture.

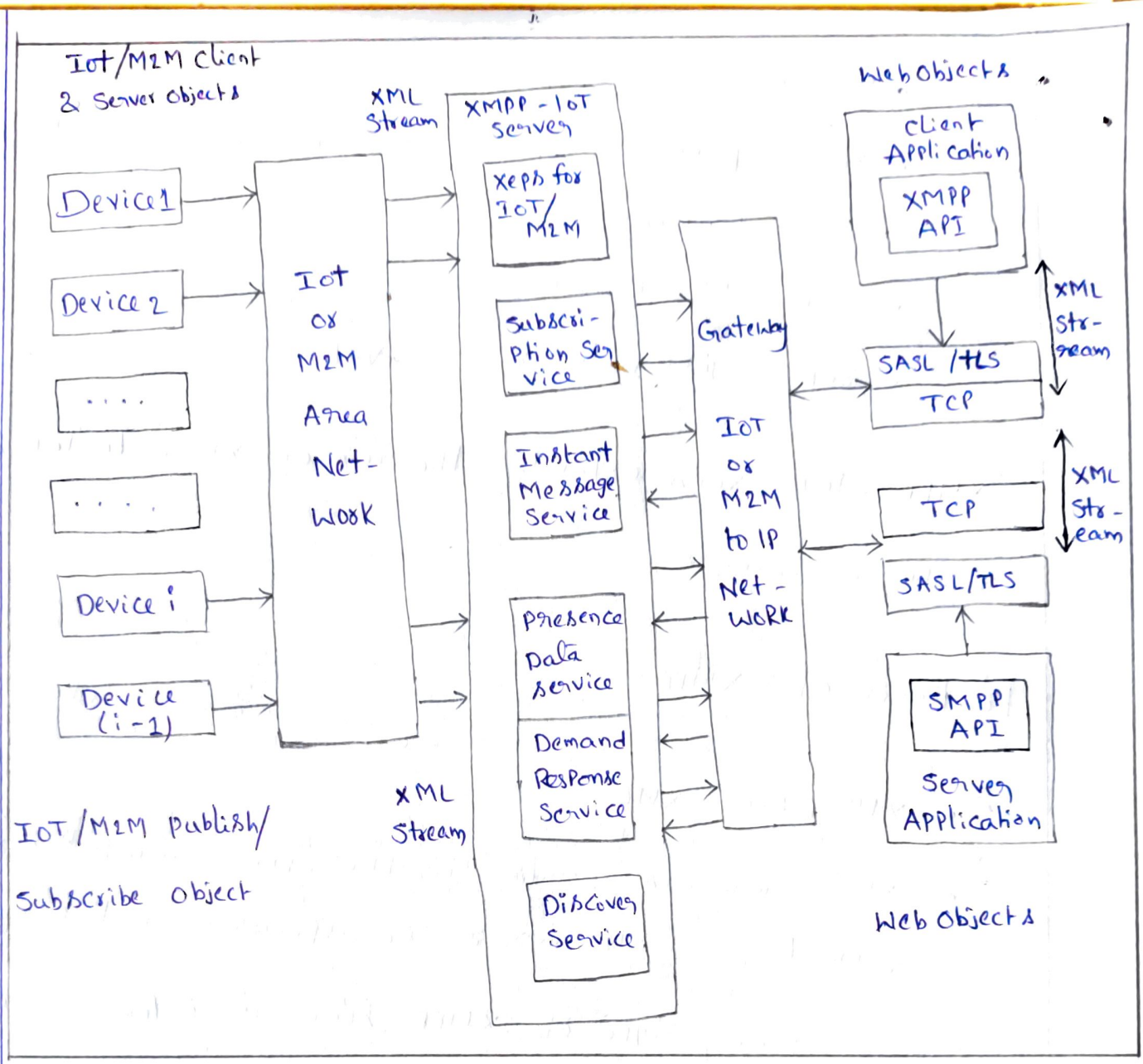


Fig : Use of XMPP and XMPP extension protocols for Connected devices & web objects. for messaging, presense notification, responses on demand using XMPP Stream.

→ The above fig shows use XMPP & XMPP extension protocols for Connected devices & web objects. The protocols are for messaging, presense notification, response-on-demand & Service discover using XML Streams.

→ The above fig. shows that XMPP - IoT Server Consists

XEPs & Services. XMPP Services are extensible to publisher / subscriber, MUC & other Services, devices & Connected devices. The XMPP-IoT Server, through a gateway between Connected devices & IP networks, Communicate through/with XMPP API.

WEB CONNECTIVITY FOR CONNECTED DEVICES NETWORK USING GATEWAY :-

Communication Gateway :-

Communication Gateway Connects 2 application layers, one at sender & the other at Receiver. The gateway also enables use of 2 different protocols, one at sender and other at Receiver ends.

The gateway facilitates the Comm. between Web Servers using the TCP/IP Protocol Conversion gateway & IoT devices. It also provide Communication between the devices using CoAP Client & Server using HTTP.

The gateway provision for one or more functions :-

- Connects the sender & Receiver ends using 2 diff. protocols. For example IoT Connects to web browser

a gateway. The server posts & gets the data using HTTP. A gate_way provide Comm. between IoT devices & webserver.

Ex:

i). ZigBee to SOAP and IP

ii). CoAP protocol version gateway for RESTful HTTP.

- Functions as proxy between the System and Server.

HTTP Request and Response Method :-

An application uses a protocol. The application layer in TCP/IP suite for Internet uses HTTP, FTP, SMTP, POP 3, TELNET. HTTP is mostly used application layer protocol for Communication Over TCP/IP. HTTP client Connect HTTP Server using TCP & then client sends a resources after eastablishing an HTTP Connection.

Problem Statement :-

Give the request & response Comm. Codes when

(a). HTTP request message Communicates to Server &

(b). Message Communicates a response to a client.

Solution :-

a). Following Code Sends request :

```
POST /items HTTP/1.1
Host : ii.jj.kk.mm
Content-Type : text/plain
Content-Length : 200.
```

b). The server first processes request and then sends an HTTP response back to client. The response also Contains the status Code & Content info :

```
200 ok
Content-TYPE : text/plain
Content-length : 200.
```

200 is for standard success Code in HTTP Connection. 400 is Standard failure Code. The status follows in that Case :

```
400 Bad request
Content-length :02
```

Data Exchanges between HTTP Web-Objects :-

An HTTP Connection enables a one-way Communication at an instance from client API to a server or from server to API.

HTTP Polling is a method for receiving new messages or updates from HTTP Server. Polling means finding whether new messages.

An HTTP transfer is stateless which means each data transfer is an independent request. Metadata is data which describes the data for interpretation in future. Each data transfer / interchange needs large headers over 100s of bytes, and greater latency in request - response exchanges.

Ways of transfering both ways at same instant are:

- Multiple TCP Connection
- HTTP requests at short, regular intervals so that responses are nearly in real time.
- Polling at successive intervals.
- HTTP Long Polling means API sends request to Server.
- Stream hidden in iFrame.

SOAP :

Applications needs to exchange objects on the internet using protocols such as HTTP, Applications may

be different languages and platforms. Simple object Access protocol (SOAP) is open source protocol.

SOAP (object) exchanges of objects between applications using XML. It is also a protocol for access to a web service. SOAP specifies the formats & way of communication the message. Its usage is independent of the application language & platform.

SOAP enables development of applications and APIs. SOAP functions Connect the GUI applications to web servers using the standards of the Internet - HTTP & XML. .NET architecture support SOAP for Internet Application Development.

SOAP uses a body element after the specifications. SOAP request could be an HTTP POST or HTTP GET request. The HTTP POST request specifies at least 2 Http headers: Content type & Content length. SOAP method uses HTTP request/response after the HTTP binding with SOAP.

REST and RESTful HTTP Web Applications :-

Rest stands for Representational State Transfer.

REST is a simpler alternative for SOAP and web Services Description Language (WSDL). REST Style Web resources & Resource-Oriented Archi (ROA) have increasingly replacing SOAP.

The architectural properties of REST are realised by applying specific interaction to data elements, Components, Connectors & objects.

REST software architecture style provisions for use of specific practices.

Client-server interactions have characteristics of performance & creation of web objects & services. Layered sys. refers to a client which Connect through an intermediate Layer (proxy, firewall, gateway etc). REST enables intermediate layer.

RESTful :-

When all interactions used in the applications fully to REST Constraints then these are called RESTful. RESTful APIs Commtly / Comply with these Constraints. REST architectural style can be used for HTTP access by GET, POST, PUT & DELETE methods for resources & building web services.

UNIT - 3.

SHORT ANSWER QUESTIONS :-

1). What is API ?

2). What is web protocol, URI ?

3). What is Communication protocol ?

4). What is REST & RESTful protocols.

5). Define CoRE ?

6). Write shot notes on CoAP.

7). What is M2M Communication protocol.

8). Define JSON format with example.

9). Write shot notes on MIME TYPE.

10). What is pull data.

11). Differences between CoAP - SMS & CoAP - MQ.

12). Define MQTT protocol.

13). Write shot notes on XMPP.

14). What is Communication Gate way.

15). What is SOAP.

16). Write a shot notes on Websockets.

ESSAY QUESTIONS :-

1). Explain in detailed about Design principles for Web Connectivity Devices.

2). Explain the Concept of Web Communication protocols for Connected Devices with neat Sketch.

3). Explain in detailed about Message Communication protocols.

4). Write a short notes on CoAP - SMS & CoAP - mq (i.e Communication protocols) with neat sketch).

5). Discuss in detailed about XMPP with neat sketch.

6). Explain in detailed about Web Connectivity for Connectivity devices.

ASSIGNMENT QUESTIONS :-

1). Explain in detailed about Web Communication protocols with neat sketch.

2). Discuss in detailed about Message Communication protocols.

UNIT-4

Introduction :-

Definition :-

Internet is a global network providing a variety of information and communication facilities, consisting of interconnected networks using standardized communication protocol.

(OR)

The internet is a globally connected network system that uses TCP/IP to transmit data via various types of media. The internet is a network of global exchanges - including private, public, business, academic and government networks connected by guided, wireless and fiber-optic technologies.

Internet Connectivity principles :-

Internet is a global network with a set of connectivity protocols for :

- Connected devices gateway for sending the data frames of the devices (or) to the devices. The data communicate over the network as a packets, which communicate through a set of routers

at the internet.

The processes manage, acquire, organise and analyse the data of IoT devices for applications, services and business processes.

• The devices perform the Controlling and monitoring functions using the messages, data stacks and Commands sent through Internet by applications.

Following are the key terms which are required for understanding Internet Connectivity principles for Communication between Connected devices network and IoT applications.

Principles :-

Header :-

→ It refers to words, which are required for processing a received data stack at a layer and which envelopes the data stack of the preceeding upper layer before transfer to the succeeding lower layer.

→ Header Consists of header fields.

→ Each word has 32-bits.

→ Each header word can have one or more fields.

→ The fields in the words are as per the processing required succeeding stages up to the destination.

IP Header :-

→ It refers to header fields, which comprise parameters & their encodings as per the IP protocol.

→ IP is a Internet Layer protocol at the Source & destination.

TCP Header :-

→ It denotes header fields containing parameters whose encoding as per the TCP protocols.

→ TCP is a Transport layer protocol at the source or destination.

Protocol Data Unit (PDU) :-

→ It is a unit of data stack maximum no. of bytes, which can be processed at a layer as per the protocol at a layer.

TCP stream :-

→ It is a sequence of words or bytes in the data

stack created at the transport layer that transmits to the destination end.

MTU (Maximum Transferable Unit) :

→ It is the unit of data stack maximum no. of bytes, which can be transferred from high layer to lower layer.

Packet :-

→ Packet is a set of bytes with a fixed maximum specified size that transfers from network layer and communicates from one to another router, until it reaches at physical, data-link and network layer at receivers end.

IP Packet :-

→ It is a data stack, which includes IP header. It communicates from a source IP address through the routers to destination IP.

Data Segment :-

→ It refers to data stack, from application supported layer for transport.

Application data is devided into Segments.

Network Interface :-

→ It is a system software Component or hardware for Providing Communication between 2 protocols Layers / Computers / nodes in a network.

Port :

→ It is an interface to the network using a protocol that sends an application layer data / stack to the lower Layer for transmission.

→ The port receives data stack at the receivers end from lower layer.

→ Each port uses an assigned number according to protocol, which is used for transmission or reception at the application Layer.

Socket :-

→ It is a software interface to the network that Links to the data stack using a port protocol & an IP address.

→ Internet data can be Considered as Communicating between the sockets.

Host :-

→ It is a device or node that Connects to network of Computers.

→ A network layer assigns a host address to each host.

IP Host :

→ It is one of that uses the Internet protocol suite.

→ An IP host has one or more IP addresses for the network interface.

Subnet :-

→ It is a subnetwork, which is logical & visible subdivision of an IP network.

→ Subdivision enables addressing a set of networked Computers in the subnet using a common & identical IP address.

Routing Prefix :-

→ 32-bits IP addresses can be devided into the msbs (most significant bits) Consisting of 8, 16 (or) 24 bits, remaining lsbs (Least Significanant bits).

→ The logical division of an IP address into 2-fields i.e a network address (or) routing prefix field and host identifier.

→ Host identifier is to identify a specific Host.

Host Identifier :-

→ The rest field may also have 2 - subfields. One for subnet id, and other for the host identifier.

→ When a network subdevides into subnet and subnet has no. of hosts.

Data Flow Graph :-

→ It denotes graphical representation using arrows from one stage to another.

→ A Circle represents a stage & Arrow represents the direction of flow of data.

→ Input at each stage Compute & Cause Outputs from the stage.

→ Generally inputs at begining of a Circle Continue flow to the next Circle, uhtill the output reach at the end of Circle.

Acyclic Data Flow Graph :-

→ ADFG refers to a DFG where only one set of inputs generate only one set of outputs for the given i/p set in the DFG Model.

→ All inputs are instantaneously available in APDFG. at each stage, except the processing interval at the stage.

Direct Acyclic Graph (DAG):

It means an ADFG in which none of the output cycles back to a previous processing stage.

4.2 : INTERNET CONNECTIVITY :-

Internet Connectivity is through a set of routers in global network of routers, which carry data Packets as per IP protocol from a source end to another and vice versa.

A source sends data to a destination using IETF standard formats.

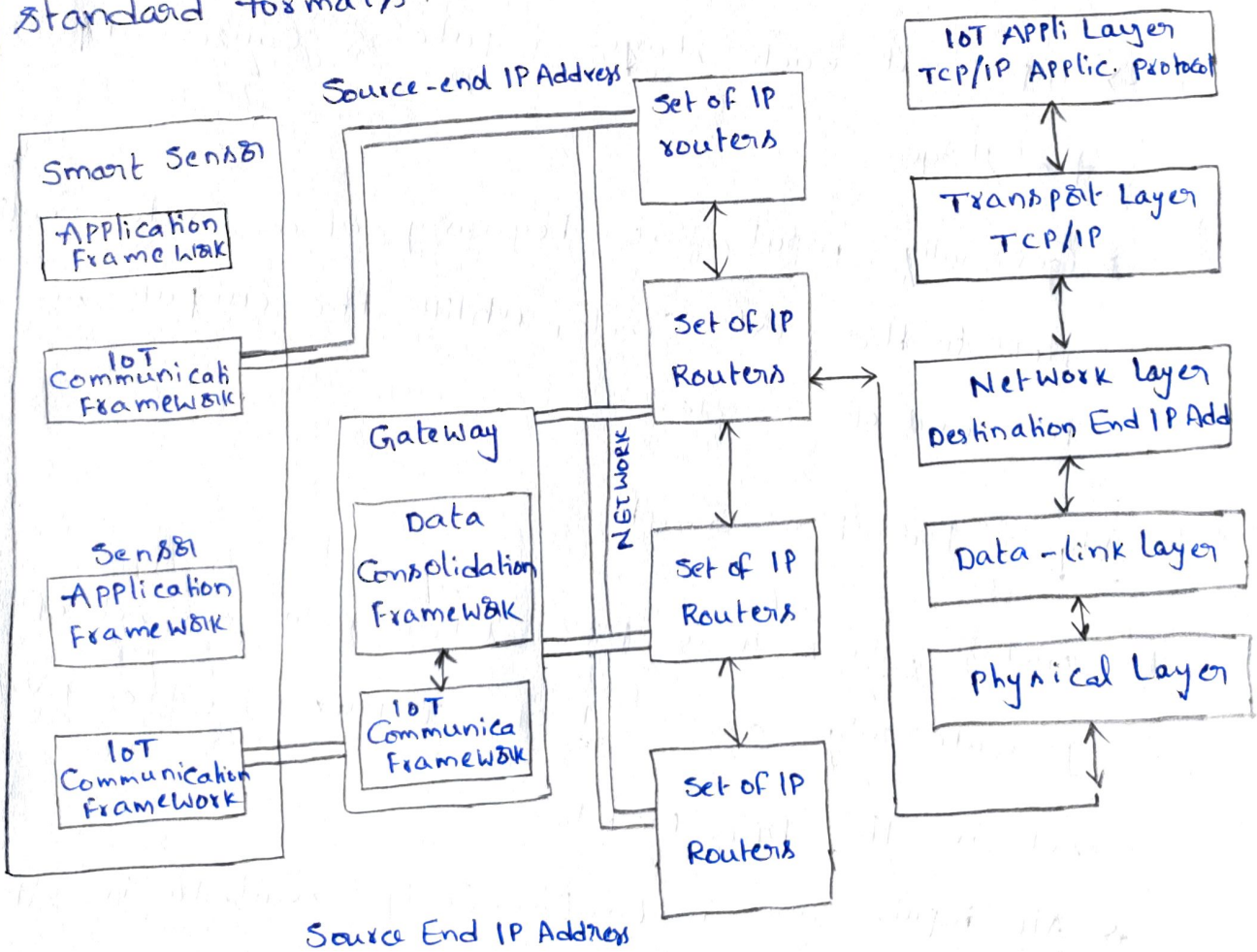


Fig: Source-end network-layer connected through a set of IP routers for data packets from an IP address & Comm. with IoT/M2M IoT Applic & Service layer using TCP/IP applic. protocol

The figure shows a source-end network-layer connected to the destination through a set of IP routers.

It also shows that a Communication Framework uses an IP address and Communicates with IoT/M2M IoT application & services layer using TCP/IP suite of application protocols to a destination IP address.

APPLICATION LAYER PROTOCALS :-

TCP/IP suite consists of number of application layer protocols.

Ex: HTTP, HTTPS, FTP, Telnet etc.

A port uses a protocol for sending & receiving messages.

TCP/IP message must be sent from right port at the transmission end and to the right port at receiver end.

HTTP :-

Hyper Text Transfer Protocol port number is 80. A web HTTP server listens to port 80 only and responds to port 80 only.

An HTTP port sends application data stack at the output to the lower layer using HTTP protocol.

An HTTP port uses a URL like http://www.mheducation.com/. Default port number is 80.

Ex: http://www.mheducation.com:80/.

HTTP is an application protocol that runs on top of the TCP/IP suite.

HTTP is method for encoding and transporting information between client & server.

→ HTTP is a primary protocol for sending information across network.

→ Using HTTP and HTML, clients can request different kinds of content (such as text, images, video and application data) from web and application server.

→ HTTP follows a request - response paradigm in which the client makes a request and the server issues a response that includes not only the requested content, but also relevent status information about the request.

→ HTTP resources such as web servers are identified across the internet using unique identifiers known as URL.

Features of HTTP :-

It is a standard protocol for requesting a URL defined web-page resource and for sending a response to web server.

An HTTP client requests an HTTP server on the Internet & server responds by sending response.

HTTP is an stateless protocol.

This is because for an HTTP request, the protocol assumes a fresh request. It means there is no ~~new~~ session is retained in the nxt exchange. This makes a current exchange by HTTP request of the previous exchanges.

The later exchanges don't depend on the current one.

HTTP is a FTP protocol, we use more effeciently than FTP.

In HTTP there are no Command line overhead.

HTTP protocol is very light (a small fonts) & thus speedy as Compared to other protocol.

HTTP is very flexible.

Assuming during a web client web Connection, the Connection breaks. The client can start by re-Connecting. Being a stateless protocol, HTTP does not keep track of the state as FTP does.

HTTP protocol is based on OOPS. methods are applied to objects identified by a URL.

Following features have been included from HTTP1.0 and 1.1 version onwards:

(a). Multimedia file Access is feasible due to provision for MIME.

Eight HTTP specific specified methods and extensions included from 1.1 version onwards. Those methods are

1. GET
2. POST
3. HEAD
4. CONNECT
5. PUT
6. DELETE
7. TRACE
8. OPTIONS

* Last 4 From 1.1

Digest Access Authentication prevents the transmission of username & password from HTTP 1.1 version onwards.

A host header field adds to suppot those ports & virtual hosts that donot accept or send IP packets. An error repsit to client when an HTTP request is with out a host header field from HTTP 1.1.

An URL is Acceptable to the server. In earlier version, only proxy servers Accept that.

Message Headers uses are: A message request from client or during a response from a server Consists of 2 parts — a start-line, none or several-headers & empty line & message body.

Message headers are:

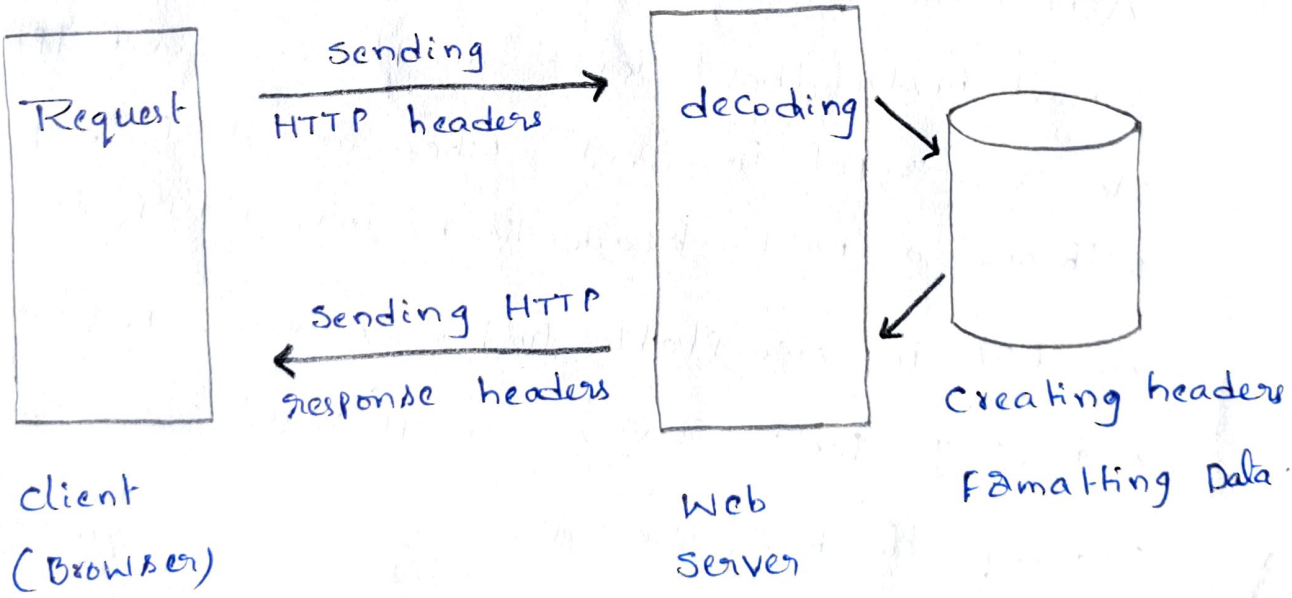
- Common headers are added when requesting a server

and responding to client.

Header includes MIME version, OPTIONS, Cacheable & transfer to close.

- Request headers are for a request & client information to a server. The header includes acceptable media or preferred specification - specifies about acceptability of HTML; or text.
- Entity headers Contain information about the entity body Contained in the message.

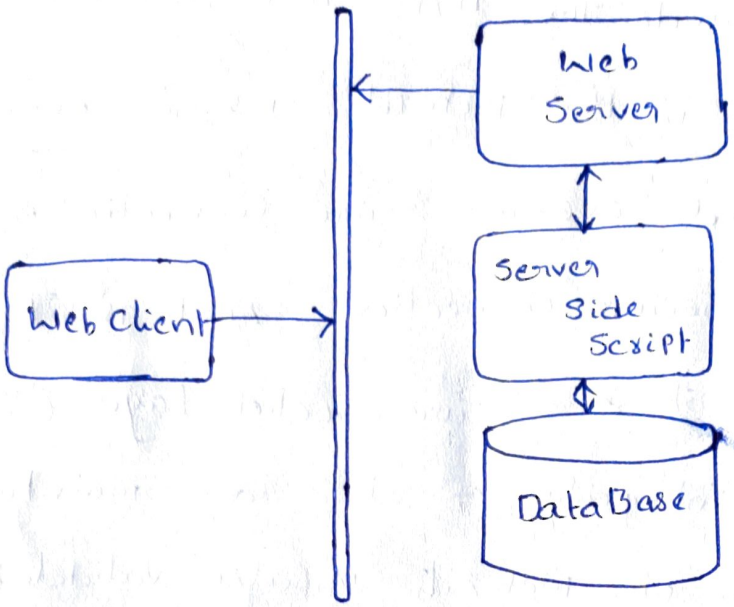
TCP/IP protocols



→ HTTP is an application protocol & relies on an underlying network-level protocol such as Transmission Control Protocol (TCP)

→ HTTP resources such as web servers are identified across the net using unique identifier known as

Basic Architechure:



The HTTP protocol is a request/response protocol based on the client/server based architechure where web browsers, robots and search engine, etc. act like HTTP clients, & the web server act as a server.

Some Common HTTP Status Codes include:

1. 200 – Successful Request (Webpage exist)
2. 301 – Moved perminently (often forwarded to new URL)
3. 401 – un-Authorized Request (-Authorization Required)
4. 403 – Forbidden (Access is nt allowed to page)
5. 500 – Internal server Error.
6. 100 – Continue (Continuing process)
7. 201 – Created new URL
8. 202 – Request Accepted.
9. 204 – No Content

HTTPS :

HTTPS stands for Hyper Text Transfer Protocol Secure. It is the protocol where encrypted data is transfered over a secure connection.

By using secure connection such as Transport Layer Security (TLS) or Secure Socket Layer (SSL), the privacy & integrity of data are maintained and authentication of website is also validated.

HTTPS ensures data security over the network - mainly Public networks like WiFi.

HTTPS encryption is done bidirectionally, which means that the data is encrypted at both the client & server sides.

Only client can decode the information that comes from server.

So HTTPS does encode ~~the~~ or encrypt the data between client & server.

which protects against eavesdropping, forging of information & tampering of data.

HTTPS is a protocol that is used to access a secure web server. When https:// is used as the prefix of a web address rather than the common http://, the

session is managed by a secure protocol, typically TLS, which supressed SSL, and the transmission is encrypted to and from the web server.

* HTTPS is an extension of the HTTP for secure communication over a computer network & widely used on Internet.

How it works :-

HTTPS keeps your stuff secrete by encrypting it as it moves between your browser and the website server. This ensures that any one listening in on the conversation can't read anything. This could include your ISP, a hacker, snooping governments, any one else who manages to position themselves b/w you and web server.

SSL is a standard protocol used by HTTPS. The newest version of SSL is now called TLS, but essentially same thing.

FTP (File Transfer Protocol) :-

The FTP is a standard network protocol used for the transfer of Computer files between a client & server on a Computer network.

FTP built on a client - Server model architecture using seperate Control & data Connections between the client & Server.

FTP users may authenticate themselves with a clear - text sign- in protocol, normally in the form of a user name & password, but can Connect anonymously if the server is Configured to allow it.

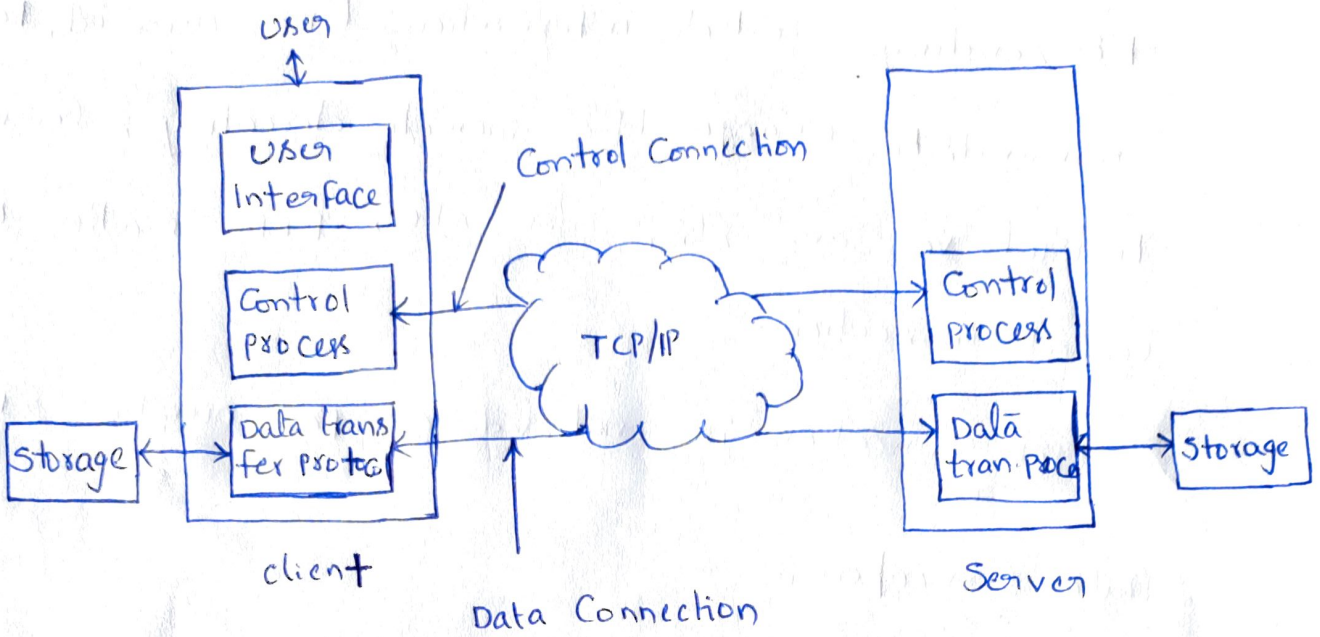
FTP Secure transmission that protects the username & password and encrypted to Content.

Ftp is often secured with SSL/TLS or replaced with SSH (Secure Shell) FTP.

The first FTP client applications were Command-line programs developed before O.S had GUI & are still shipped with most Windows, UNIX, Linux O.S.

Many FTP clients & automation utilities have since been developed for desktops, servers, mobile devices & hardware.

FTP Architecture :-



FTP is used copy the files from one host to another. FTP offers the mechanism for the same following manner :

* FTP Creates 2 processes such as Control Process and data transfer process at both ends i.e (client & Server).
* FTP establishes 2 different Connections ; one for data transfer and other is for Control information.
* Control Connection is made between Control Process while Data Connection is made between.
* FTP uses Port 21 for Control Connection & Port 20 for data Connection.

Control Connection :-

For sending control information like user id, password, commands to change the remote directory, commands to retrive and store files etc, FTP makes use of control connection.

Control Connection is initialed on port number 21.

Data Connection :-

For sending Acheal file, FTP makes use of data connection.

Data Connection is initialed on port number 20.

Data Structures :-

FTP allows 3 types of data structures.

1). File Structure :-

In file structure there is no internal structure and the file is considered to be a continous sequence of data bytes.

2. Record Structure :-

In record structure the file is made up of sequential records.

3. Page Structure :-

In Page structure the file is made up of independant indexed pages.

FTP Commands :-

USER : This Command sends the user identification to the server

PASS : This Command sends the user password to the server.

PWD : It causes the name of the current working directory to be returned in the reply.

LIST : Sends a request to display the list of all the files present in the directory.

STOR : It causes to store a file into the current directory of the remote host.

ABOR : It tells the server to abort the previous FTP service Command & any associated data transfer.

QUIT : This Command terminates a USER & if file transfer is not in progress, the server closes the Control Connection.

The first FTP client applications used the DOS command prompt with commands & syntax. Since then, many graphical user interface (GUI) clients have been developed with in O.S, making it easier for user to upload & download files.

Uses of FTP :-

1. An FTP site is a web site where users can easily upload or download files.
2. FTP Explorer is an FTP client based on Windows 95 file Manager.
3. An FTP server is a dedicated Computer which provides an FTP service.
4. An FTP client is a Computer application which access an FTP server.

Anonymous FTP :-

Anonymous FTP is enable on some sites whose files are available for public Access. A user can Access these files with out having any username and password. Insted, username is set to anonymous & password to guest by default.

Simple diagram for FTP :-

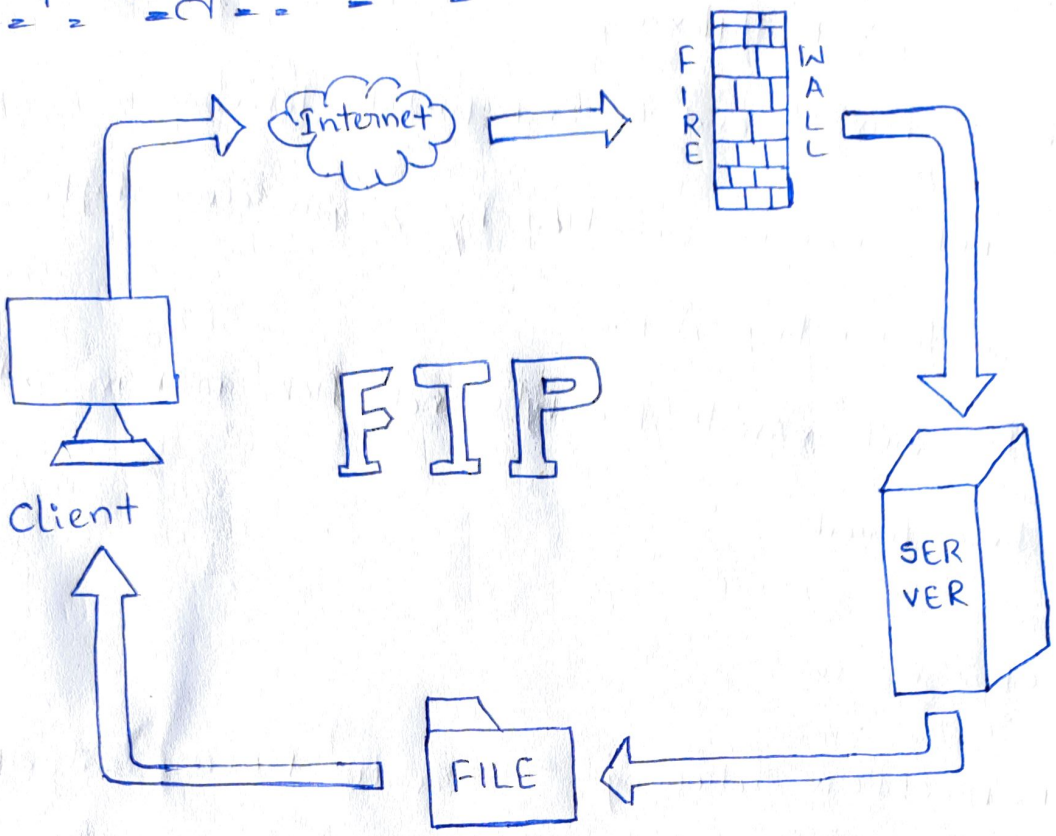


Fig : FTP.

TCP :-

It is a Connection Oriented protocol and offers end-to-end packet delivery. It acts as backbone of Connection.

Features :-

- TCP Corresponds to the Transport layer of OSI Model.
- TCP is reliable & Connection Oriented protocol.
- TCP offers :
  - → Stream data transfer
  - → Reliable
  - → Effecient Flow Control.

→ Full duplex

→ Multiplexing

- It provide end-to-end packet delivery.
- TCP ensures reliablity by sequencing bytes with a forwarding Ack.
- It retransmits the bytes not Ack with in specified time period.

TCP Services :-

TCP offers following Services to Processes at applicalion layer:

- ✱ Stream Delivery Service
- ✱ Sending & Receiving Buffers
- ✱ Bytes & Segments
- ✱ Full duplex Service
- ✱ Reliable Service
- ✱ Connection Oriented Services.

IP :- Internet protocol is a Connectionless & unreliable protocol. It ensures no guarantee of succesfully transmission of data.

In order to make it reliable, it must be paired with reliable protocol such as TCP at the transport layer.

Internet protocol transmits the data in the form of a datagram.

Features :-

→ The length of datagram is variable.

→ The datagram is devided into 2 parts : Header & data.

→ The length of header is 20 to 60 bytes.

→ Header Contains information for routing & delivery of Packet.

Telnet :-

Telnet is a protocol used to login to remote Computer on the internet.

There are no. of telnet clients having user-friendly interface.

Telnet is a protocol used on the internet or local Area Network to provide bidirectional interactive text-oriented Communication facility using a virtual terminal Connection.

User data is interspersed in-band with telnet Control info in an 8-bit byte oriented data Connection over TCP.

Telnet was developed in 1969 begining with RFC 15 extended in RFC 855 (Request For Comments) & Standardized as IETF.

→ Telnet developed in 1969, it is a protocol that provides a Command line interface for Communication with remote device (or) Server.

→ Telnet is a terminal network/tele Communication networks, it is a user Command & underlying the TCP/IP protocol for accessing remote Computers.

→ The telnet is also used to refer to the software that implements the client part of the protocol.

→ Telnet client applications are available for virtually all Computer platforms. Telnet is also used as a verb.

→ Telnet means to establish a Connection using the telnet protocol, either with Command Line or with a programatic interface.

→ Telnet Server port number is 23.

→ Telnet client and Server functionality Comes built-in most operating Systems. However, there are several thrid-party applications like putty client that enables remote Connectivity. A user can connect to a remote m/c through several access modes such as raw access, SSH access etc.

→ SSH mode offers encryption & security and hence can prevent eavesdropping by hackers. This is by far the most secure way of connecting to machine.

Ex: `telnet host port`

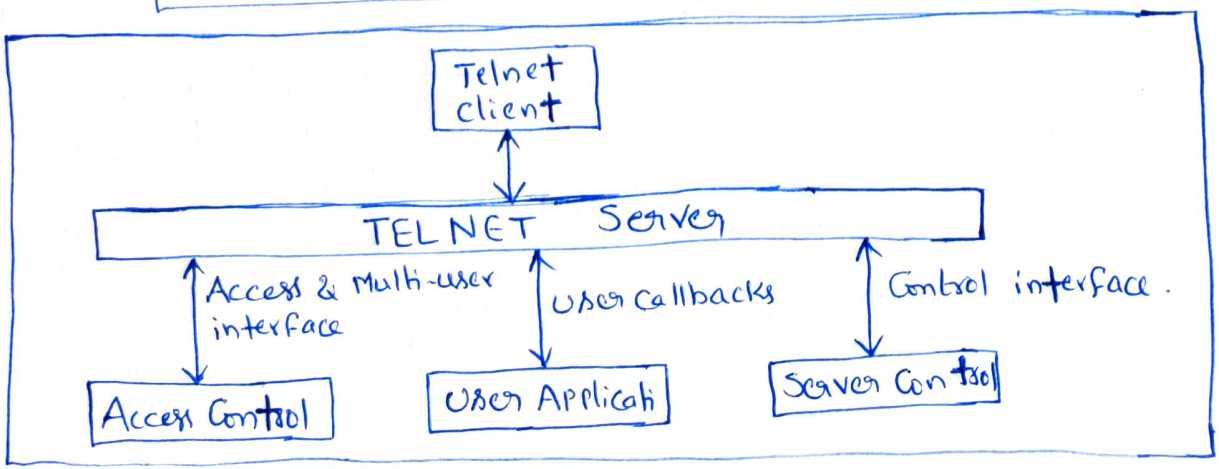
`telnet 192.168.1.15 80`

Here host with address of the service (192.168.1.15), and port with the port number on which the service runs (for example, 80 for http).

→ Command-line telnet interface clients are built into most versions of macOS, Windows, Unix & Linux. To use these clients, goto their respective Command lines.
(i.e Terminal application in macOS, shell in unix/linux, or DOS prompt in windows),

→ Enter telnet Commands on any one of the terminal.

i.e `telnet 192.168.1.15 80`



UNIT – V

Data Acquiring, Organizing and Analytics in IOT/M2M, Applications / Services / Business Processes, IOT / M2M Data and Storage, Business Models for Business Processes in the Internet of Things, Organizing Data, Transactions Business Processes, Integration and Enterprise Systems.

OBJECTIVES :-

* Apply the data – acquiring and data – storage functions for IOT / M2M devices data and message
* Classify ways of organising data
* Summarise the transactions on stored data, function for business processes and business intelligence, and the Concepts of IOT applications – integration and services architecture
* Identify the functions and usage of data analytics and data visualization for IOT applications and business processes
* Explain knowledge discovery, knowledge management and knowledge – management reference architecture

5.1 DATA ACQUIRING

* Data acquisition means acquiring data from IOT or M2M devices. The data communicates after the interactions with a data acquisition system (application).

* The application interacts and communicates with a number of devices for acquiring the needed data. The devices send data on demand (or) at programmed intervals. Data of devices communicate using the network, transport and security layer.

* An application can configure the devices for the data when devices have configuration capability. For example, the system can configure devices to send data at defined periodic intervals. Each device configuration controls the frequency of data generation.

* Application can configure sending of data after filtering (or) enriching at the gateway at the data-adaptation layer.

* Device-management software provisions for device ID or address, activation, configuring, registering, deregistering, attaching, and detaching.

* Applications configure the device for acquiring data.

5.2 ORGANIZING AND ANALYTICS IN IOT/M2M

Organised data after acquiring from the devices can be used for multiple purposes. Applications usually use the data of devices in two way – for monitoring, reporting and rule-based actions.

For Example, in Internet of Streetlights applications just do that for analytics, new facts and taking decisions based on those facts.

5.2.1 Analytics Phases

Analytics has three phases before deriving new facts and providing business intelligence.

There are :

1. Descriptive analytics
2. Predictive analytics
3. Prescriptive analytics

1. Descriptive Analytics

Descriptive analytics means finding the aggregates, frequencies of occurrences, mean values (or) Variances in values (or) groupings using selected properties and hence applying there. It enable the following

- Actions, Such as Online Analytical Processing (OLAP) for the analytics.

Descriptive Analytics Methods

- Spreadsheet-based reports and data visualisations: Result of descriptive analysis can be presented in a spreadsheet format before creating the data visuals for the user.
- Descriptive statistics-based reports and data visualisations:

Descriptive analysis can also use descriptive statistics. Statistical analysis means finding peak, minima, variance probabilities, and statistical parameters.

- Data mining and machine learning methods in analytics:

Data mining analysis means use of algorithms which extract hidden (or) unknown information or patterns from large amounts of data. Machine learning means modelling of the specific tasks.

2. Predictive Analytics

* Predictive analytics is advanced analytics. The user interprets the outputs from advanced analytics using descriptive analytics method, such as data visualisation
* The software tools make the predictive analytics easy to use and understand. The examples are as follows:
  - Predicting trends
  - Undertaking preventive maintenance from earlier models

of equipment and device failure rates

- predicting by identifying patterns, clusters with similar behaviour
- Predicting based on anomalous characteristics, anomaly detection.

3. Prescriptive Analytics

Prescriptive analytics answers not only what is anticipated (or) what will happen (or) when it will happen, but also why it will happen based on the input from descriptive analytics and busines rules.

5.2.2 Event Analytics

* Event analytics use event tracking and event reporting. An event has the following components :

- Category - an event of chocolate purchase in ACVM example belongs to one category and event of reaching predefined threshold of sell for specific chocolate flavour which belongs to other category.
- Action — Sending message from ACVM on completing predefined sell is the action taken on the event
- Value (optional) - on event, messaging the number of chocolate of that flavour sold (or) remaining.

5.2.3 In + memory Data Processing and Analytics

In - memory option of row or column formats can be selected in certain databases, for example, oracle dual format architecture database that enables to run the, real - time, adhoc, analytic queries on IOTs' data.

In - memory and On - Store Row Format Option

Consider the transactions of the type, ATM transactions or sales order transactions. Each row has separate record. For example, separate record for each ACVM or each bank Customer (or) each sales order. The columns have data associated with the record. A row format enables quick access of all columns for a record. OLTP operations run fast in the row format. There are fewer rows and more columns.

A row format, allowing row data, will be brought into the CPU with a single memory reference. Data for each record is together in - memory and on - store. There is a single copy of the table on storage.

In - memory and On - Store column Format option (Few columns and More Rows)

Consider analytics of the type, monthly sales of chocolates on the ACVMs, enterprise yearly profits.

Analytical workloads access few columns but scan the entire data set. Analytics therefore run faster on column format, more rows and few columns. Fast- for processing needs few columns and many rows. They typically require aggregation or fusion or compaction also. A columnar and many rows. They typically require aggregation (or) fusion (or) compaction also. A columnar format allows for much faster data retrieval when only a few columns in a table are selected because all the data for a column is kept together in- memory in column format option. A single memory access will load many column values into the CPU. It also lends itself to faster filtering and aggregation, making it the most optimised format for analytics.

5.2.4 Real - time Analytics Management

Real - time analytics management means ensuring faster OLTP as well as OLAP. Real - time analytics works both as direct querying using an OLTP database such as Oracle database provides in - memory row format option large speedups for OLTP applications and in - memory column format option for large speedups for OLAP applications.

5.3 DATA ACQUIRING AND STORAGE

It describe devices data, and steps in acquiring and storing data for an application, service (or) business process.

5.3.1 Data Generation :-

Data generates at devices that later on, transfers to the Internet through a gateway.

the Data generates as follows :-

=> Passive device data

=> active device data

=> Event data

=> Device real-time data

=> Event-driven device data

Passive Device Data

* Data generate at the device or system, following the result of interactions.
* A passive device does not have its own power source
* An external source helps such a device to generate and send data.

Ex :- RFID (or) an ATM debit card

Active devices data :-

* Data generates at the device (or) system as following the result of interactions.

* An active device has its own power source.

Ex :- Streetlight Sensor (or) wireless sensor node.

Event data :-

* A device can generate data on an event only once.

For Example :-

On detection of the traffic or on dark ambient conditions, which signals the event. The event on darkness communicates a need for lighting up a group of streetlights.

* A system consisting of security cameras can generate data on event of security breach or on detection of an intrusion.

Device real-time data :-

* An ATM generates data and communicates it to the server instantaneously through the Internet. This initiates and enables online Transactions processing (OLTP) in real time.

Event - driven device data :-

* A device data can generate on an event only once

Example

(i) A device receives command from controller (or) Monitor, and then performs action using an actuator. When the action completes, then the device sends an acknowledgement.

(ii) When an application seeks the status of device, then the device communicates the status.

5.3.2 Data Acquisition :-

* Data acquisition means acquiring data from IOT (or) M2M devices. The data communicates after the interactions with a data acquisition system.

* The application interacts and communicates with a number of devices for acquiring the needed data. The devices send data on demand (or) at programmed intervals.

* An application can configure the devices for the data when devices have configuration capability. For example, the system can configure devices to send data at defined periodic intervals. Each device configuration controls the frequency of data generation.

* Application can configure sending of data after filtering (or) enriching at the gateway at the data-adaptation layer.

* Device-management software provisions for device ID (or) address, activation, configuring, registering, deregistering, attaching and detaching.

5.3.3 Data Validation :-

* Data acquired from the devices does not mean that data are correct, meaningful (or) consistent.

* Data consistency means within expected range data (or) as per pattern (or) data not corrupted during transmission.

* Therefore, data needs validation checks.

* Data Validation software do the Validation checks on the acquired data. Validation software applies logic, rules and semantic annotations. The applications (or) Services depend on Valid data.

* Then only the analytics, speculations, prescriptions, diagnosis and decisions can be acceptable.

5.3.4

DATA CATEGORISATION FOR STORAGE :-

* Service, business processes and business intelligence use data. Valid, useful and relevant data can be categorised into three categories for storage - data alone, data as well as results of processing, only the results of data analytics are stored.

1. Data which needs to be repeatedly processed, referenced (or) audited in future, and therefore, data alone needs to be stored.

2. Data which needs processing only once, and the results are used at a later time using the analytics and both the data and results of processing and analytics are stored. Advantages of this case are quick visualisation and reports generation without reprocessing.

3. online, real-time or streaming data need to be processed and the results of this processing and analysis need storage.

5.3.5 Assembly Software for the Events

* A device can generate events. For Example, a sensor can generate an event when temperature reaches a present value (or) falls below a threshold.

* A pressure sensor in a boiler generates an event when pressure exceeds a critical value which warrants attention.

* Each event can be assigned an ID. A logic value sets (or) resets for an event generated and acted upon (or) not yet generated.

5.3.6

Data Store :-

A data store is a data repository of a set of objects which integrate into the store. Features of data store are :-

* Objects in a data-store are modeled using classes which are defined by the "database schemas".

* A data store is a general concept. It includes data repositories such as database, relational database, flat file, spreadsheet, mail server, web server, directory services and VM ware.

* A data store may be distributed over multiple nodes. Apache Cassandra is an example of distributed data store.

5.4

BUSINESS MODEL FOR BUSINESS PROCESSES IN THE INTERNET OF THINGS

Internet of Things (IOT) is a network of things connected to each other and refers to an ecosystem comprising of objects, connectivity and application/ services.

There are a number of projections on the size of the market by 2020 and the projections vary significantly but all projections agree to a minimum of $1 trillion plus addition to economic activity due to IOT.

5.4 INTERNET OF THINGS- VALUE CHAIN

The Value chain is perhaps the most important part of the business model. It defines how the service Clearly, the partnership formation is not easy when each of the entity would consider itself more important than the other. The position in the Value chain would define its relevance, strategy and opportunity

* The player capturing the biggest pie of the Value chain should ideally take a lead in forging Partnerships.

5.4.2

OPERATORS :-

The operators are very critical as they provide the connectivity and have had a head start over others with M2M. It is natural for them to consider themselves as frontrunners.

* IOT will drive greater data usage but due to commoditization, connectivity is at risk of becoming the 'non - value' cold component of the value chain.

* The risk of operators being reduced to just the pipe to the cloud in very real.

5.4.3

PLATFORM PROVIDERS :-

Platform is the heart of IOT and bring together the hardware, the connectivity, service providers and the vertical applications to provide industry specific IOT solutions.

* Most of the serious players are eyeing to become platform providers but the success would depend on their ability to ~~page~~ forge partnerships and drive towards common goal.

5.4.4

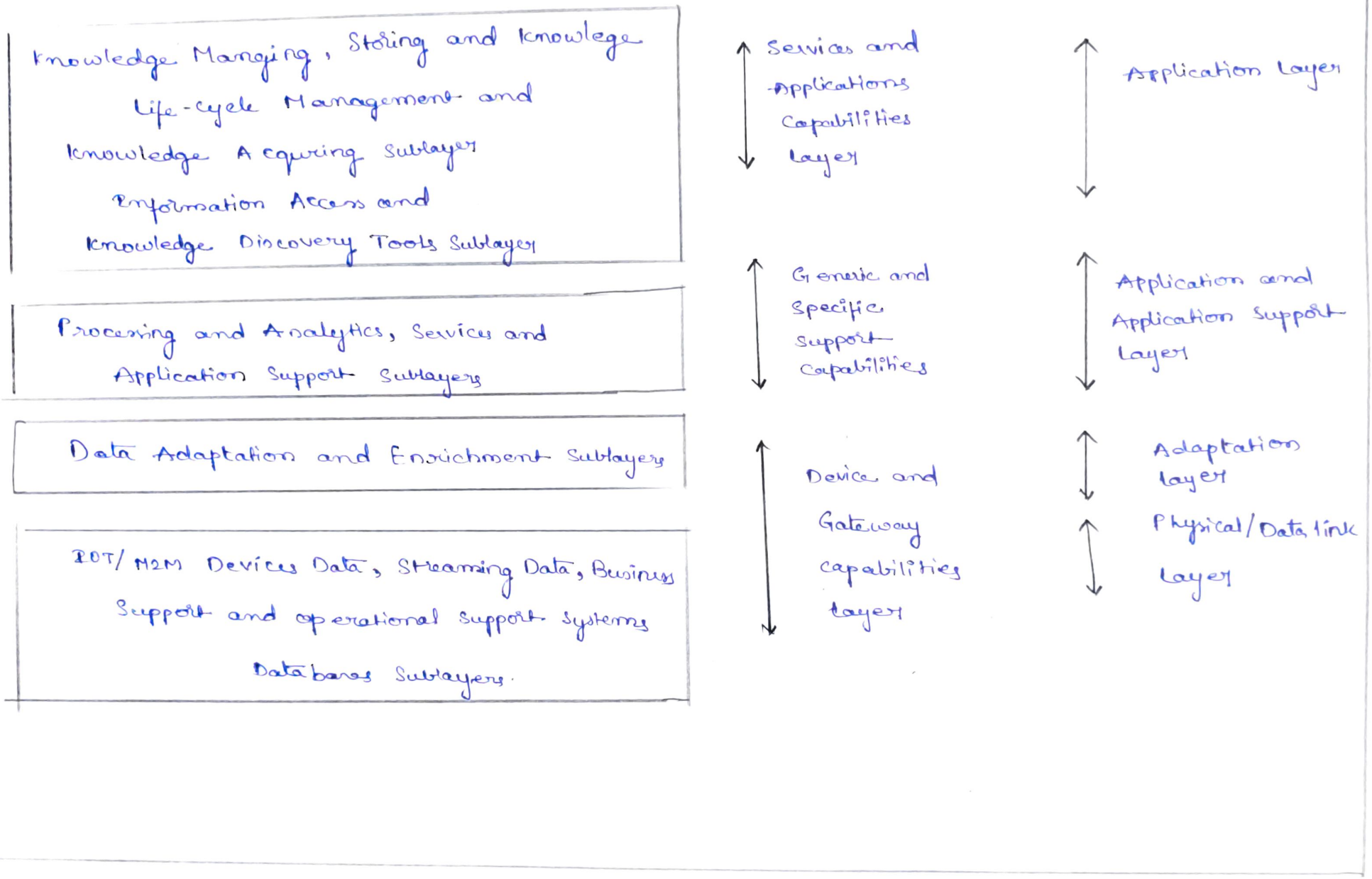
SYSTEM INTEGRATORS :-

They have a big role in the industrial internet of things. Not everything is plug and play out of box and hence we need system integrators to make the individual components of IOT to work together in the most optimal way for the customer. The best option for the System integrators is to identify their niche and enter into partnerships with large platform players.

5.4.5

APPLICATION PROVIDERS :-

No one provider has an end to end IOT solution yet and hence the only choice is to partner or perish The platform providers seem to be well placed but will need the partnerships to fully realize the potential of IOT. Device makers and operators would need to partner with platform providers and vice versa to ensure that they are not left out of the IOT ecosystem.


Knowledge Managing, Storing and Knowlege Life-cycle Management and Knowledge Acquring Sublayer
Information Access and Knowledge Discovery Tools Sublayer
Processing and Analytics, Services and Application Support Sublayers
Data Adaptation and Enrichment Sublayers
IOT/M2M Devices Data, Streaming Data, Business Support and operational Support Systems Databases Sublayers.
Services and Applications Capabilities Layer
Generic and Specific Support Capabilities
Device and Gateway capabilities layer
Application Layer
Application and Application Support Layer
Adaptation layer
Physical/Data link Layer

5.5 ORGANISING THE DATA

Data can be organised in a number of ways.

For, Example :-

Objects, files, datastore, database, relational database and object oriented database.

5.5.1 Database

One popular method of organising data is a database, which is a collection of data. This collection is organised into tables. A table provides a systematic way for access, management and update.

Relational Database

* A relational database is a collection of data into multiple tables which relate to each other through special field, called keys. Relational database provide flexibility Examples of relational database are MySQL, PostGreSQL, Oracle database created using PL/SQL and Microsoft SQL Server using T-SQL.

* Object Oriented Database (OODB) is a collection of objects, which save the objects in objected oriented design. Example, are Concept Base COU cache.

Database Management System

Database Management System (DBMS) is a software System, which Contains a set of programs specially designed for creation and management of data stored in a database.

Atomicity, Data Consistency, Data isolation and Durability Rules (ACID) :-

Atomicity :- A transaction must complete in full, treating it as indivisible. When a service request Completes, then the pending request field should also be made zero.

Consistency :- A transaction must complete in full, treating it as indivisible.

Data after the transactions should remain Consistent. For Example, Sum of chocolates sent should equal the sums of sold and unsold chocolates for each flavour after the transactions on the database

Isolation :-

To avoid the overlapping between the transaction.

Durability :- After Completion of transactions, the previous transaction cannot be recalled. only a new transaction can affect any change.

Distributed Database :-

Distributed Database is a collection of logically interrelated database over a computer network.

* DDB is a collection of database which are logically related to each other.

* Cooperation exists between the database in a transparent manner.

Consistency, Availability and Partition - Tolerance Theorem :-

Consistency, Availability and Partition - Tolerance Theorem is a theorem for distributed computing systems.

Consistency :- Every request receives the most recent write or an error. When a message (or) data is sought the network generally issues notification of time-out (or) read error. During an interval of a network failure, the notification may not reach the requesting nodes.

Availability :- Every request receives a response, without guarantee that it contains the most recent version of the information.

Partition tolerance :- 'The system continues to operate despite an arbitrary number of message being dropped by the network between the nodes'.

5.5.2 QUERY PROCESSING

* Query means an application seeking a specific data set from a database.

* The process should use a correct as well as efficient execution strategy.

* Five steps in processing are :

1. Parsing and translation : This step translates the query into an internal form, into a relational algebraic expression and then a parser, which checks the syntax and verifies the relations.

2. Decomposition to complete the query process into micro-operations using the analysis, conjunctive and disjunctive normalisation and semantic analysis.

3. Optimisation which means optimising the cost of processing. The cost means number of micro-operations generated in processing which is evaluated by calculating

Cost of the sets of Equivalent Expressions.

4. Evaluation Plan :- A query - execution engine takes a query - evaluation plan and executes that plan.

5. Returning the results of the query.

Distributed Query Processing :-

Distributed Query Processing means query processing operations in distributed database on the same system (or) networked systems. The distributed database system has the ability to access remote sites and transmit the queries to other systems.

5.5.3

SQL :-

SQL Stands for Structured Query Language. It is a language for Viewing (or) changing database. It is a language for data querying, updating, inserting, appending and deleting the database.

* SQL features are as follows:

- Create Schemas is a structure that contains descriptions of Objects created by a user.
- Create catalog consists of a set of schema that constitute the description of the database.

- Use Data Definition language (DDL) for the commands that depict a database, including creating, altering and dropping tables and establishing constraints.
- Use Data Manipulation Language (DML) for commands that maintain and query a database.
- Use Data Control Language (DCL) for commands that control a database, including administering privileges and committing data.

5.5.4

NOSQL :-

NOSQL stands for NO-SQL or Not only SQL that does not integrate with applications that are based on SQL.

NOSQL may consist of the following:

* A class of non-relational data storage systems, flexible data model and multiple schemas.
* class consisting of unordered keys and using the JSON for Example in PNUTS.
* class Consisting of name and Value in the text. for example in couchDB
* May not require a fixed table schema.

5.5.5

Extract, Transform And Load :-

ETL is a System which enables the usage of database used, especially the ones stored at a data warehouse.

Extract :- Extract means obtaining data from homogeneous or heterogeneous data sources.

Transform :- Transform means transforming and storing the data in an appropriate structure (or) format.

Load :- Load means the structured data load in the final target database (or) data store of data warehouse.

* All the three phases can execute in parallel. Data extraction takes longer time. Therefore, the System while pulling data, executes another transformation processes on already received data and prepares the already prepared data for loading.

* ETL System usage are for integrating data from multiple applications hosted separately.

5.5.6

Relational Time Series Service :-

Time Series data means an array of numbers indexed with time. Time series data can be considered as time stamped data.

* Time Series is any data-set that is accessed in a sequence of time. Software programs and an analytics program analyses in a chronological order.

* Time Series Database (TSDB) is a software system which implements a database that optimally handles mathematical operations, queries (or) database transactions on time series.

5.5.7

Real-Time and Intelligence :-

* Decision on real-time data is fast when query processing in live data has low latency. Decision on historical data is fast when interactive query processing has low latency.

* Tera Data Aster and Pivotal Greenplum are examples of MPP. In Memory and on-Store both transaction methods exist for the database. SAP Hana and Q Click View are examples of in-memory database. SAP Sybase IQ and HP Vertica are examples for columnar database for faster Analytics.

5.6 TRANSACTIONS, BUSINESS PROCESES, INTEGRATION AND ENTEPRISE SYSTEM

A transaction is a collection of operations that form a single logical unit. For, example, a database

connect, insertion, append, deletion or modification transactions. Business transactions are transactions related in some way to a business activity.

5.6.1 Online Transactions and Processing :-

OLTP means process as soon as data or events generate in real time. OLTP is used when requirements are availability, Speed, Concurrency and recoverability in database for real-time data or events.

Batch Transactions Processing :-

Batch transaction processing means the execution of a series of transactions without user interactions. Transaction jobs are set up so they can be run to completion. Scripts, command-line arguments, control files, or job control language predefine all input parameters.

Streaming Transactions Processing

Examples of the Streams are log Streams, event Streams and twitter Streams. Query and transactions processing on streaming data need specialised frameworks.

Interactive Transactions Processing

Interactive transactions processing means the transactions which involve continual exchange of

information between the computer and a user.

Real-time Transactions Processing :-

Real-time transaction processing means that transactions process at the same time as the data arrives from the data sources and data store.

Event Stream Processing and Complex Event Processing

Event Stream Processing (ESP) is a set of technologies, event processing language, complex Event processing (CEP) event visualisation, event database and event-driven middleware.

ESP and CEP does the following :-

* Processes tasks on receiving streams of event data
* Identifies the meaningful pattern from the streams.
* Detects relationship between multiple events.
* Correlates the events data

Complex Event Processing

CEP has many applications. For Example, IOT event processing applications, Stocks algorithmic-based trading and location-based services.

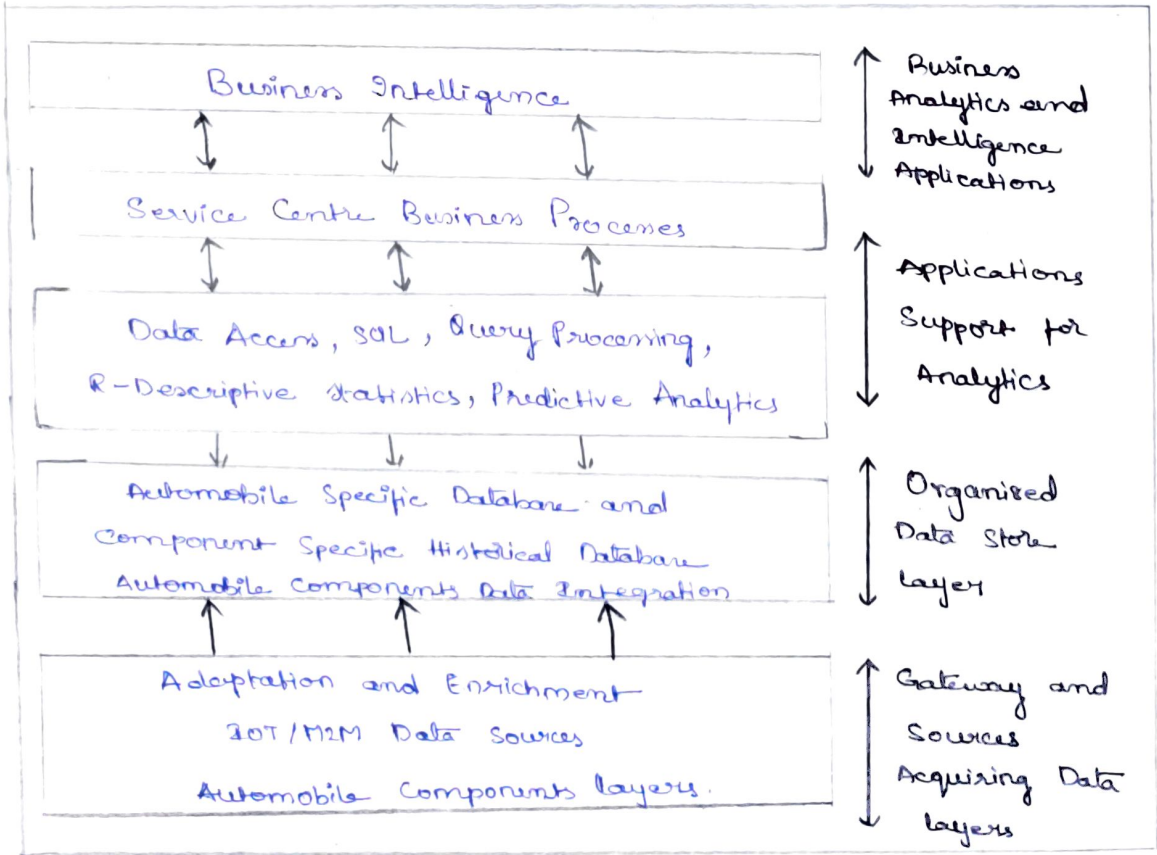
5.6.2 BUSINESS PROCESSES

A business process consists of a series of activities which serves a particular specific result. It is used when an enterprise has a number of interrelated processes which serve a particular result or goal. The results enable sales, planning and production.

IOT / M2M enables the devices data in database for business processes. The data supports the process.

5.6.3 Business Intelligence :

Business intelligence is a process which enables a business service to extract new facts and knowledge and then undertake better decisions.



Architecture reference model for the business intelligence and business processes at ACPAMS.

5.6.4 Distributed Business Process :-

Distribution of processes reduces the complexity, communication costs, enables faster responses and smaller processing load at the central System.

Distributed Business Process System (DBPS) is a collection of logically interrelated business processes in an Enterprise network. DBPS means a software system that manages the distributed BPs.

DBPS features are :-

① DBPS is a collection of logically related BPs like DDBS.

② DBPS should process location independence.

5.6.5 Complex Applications Integration and Service Oriented Architecture

* IOT Applications, services and processes enhance the existing Systems in a number of enterprises. For example, an automobile enterprise has a number of divisions. Each division has sales, Customer Relations Management, Automobile Maintenance Services, and Accounting.

* IOT - based services help in business intelligence, processes and Systems, and as post - sales services and Supply chain automation and analytics results

in Visualisation enhancement of the services from an enterprise.

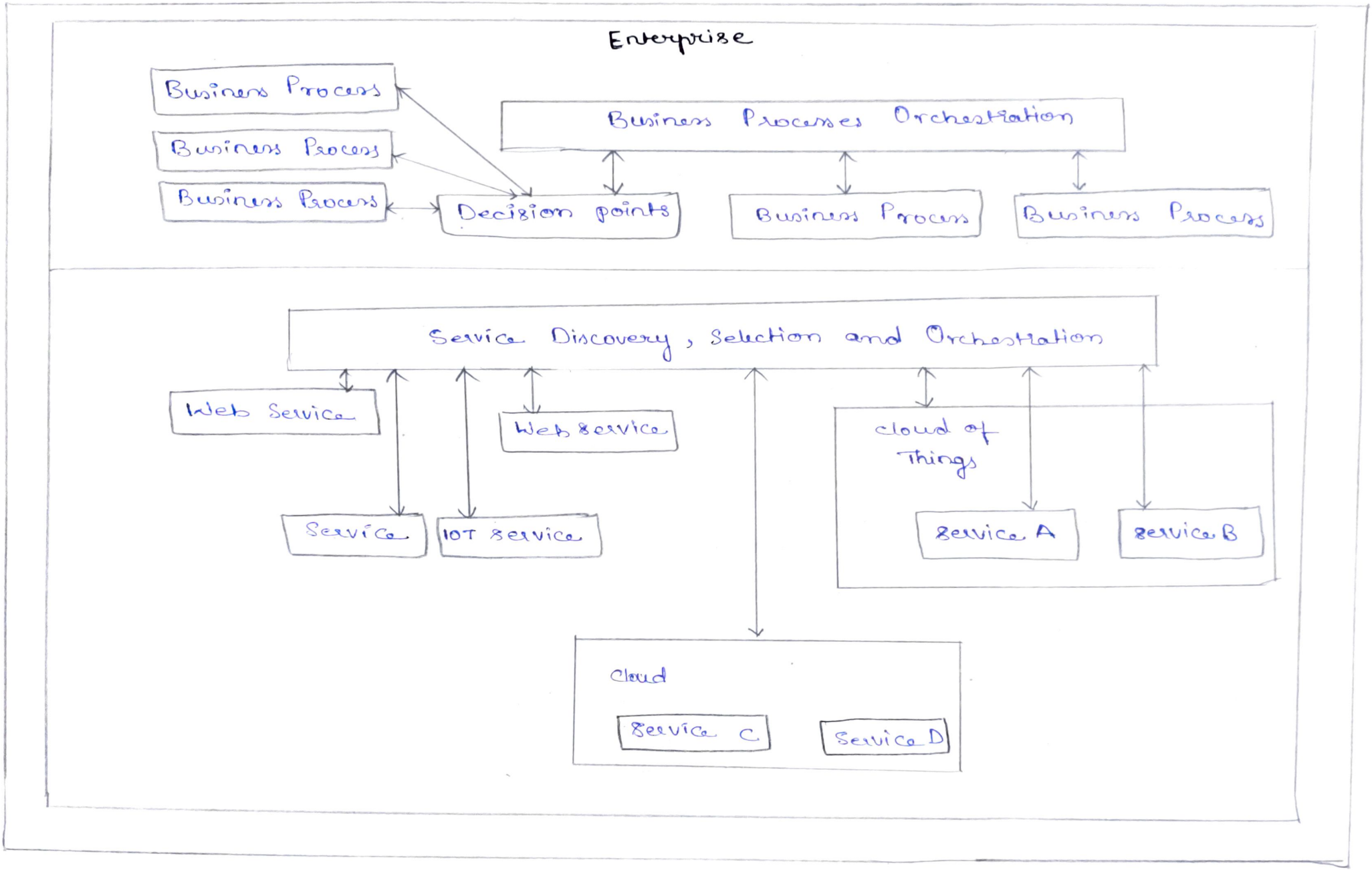
* Complex application integration means integration of heterogenous application architectures and number of processes. SOA consists of services, messages, operations and processes.

5.6.6 Integration and Enterprise Systems

* Process orchestration means a number of busines processes running in parallel and a number of processes running in sequence.

* The service discovery and selection software components select the services for application integration.

* The below figure explain the complex applications integration architecture and SOA of cloud-based IOT services, web services, cloud services and services.


Enterprise
Business Process
Business Process
Business Process
Business Processes Orchestration
Decision points
Business Process
Business Process
Service Discovery, Selection and Orchestration
Web Service
Web Service
Service
IOT Service
cloud of Things
Service A
Service B
Cloud
Service C
Service D

5.7 APPLICATIONS / SERVICES / BUSINESS PROCESSES

Applications :-

* Fast and easy communication of information
* Live and analyzed data collection.

Smart Cities :-

* Smart Parking : Monitoring of parking spaces availability in the city.
* Structural health : Monitoring of vibrations and material conditions in buildings, bridges and historical monuments.
* Noise - Urban Maps : Sound Monitoring in bar areas and centric zone in real time.
* Traffic Congestion : Monitoring of Vehicles and pedestrian levels to optimize driving and walking routes.
* Smart Lighting : Intelligent and weather adaptive lighting in street light.

Smart Environment :-

* Forest Fire Detection : Monitoring of combustion gases and preemptive fire conditions to define alert zones.
* Air Pollution : Control of CO_2 emissions of factories, Pollution emitted by cars and toxic gases generated in farms.

* Snow Level Monitoring :- Snow level measurement to know in real time the quality of ski tracks and allow Security corps avalanche prevention.

Smart Water :-

* Portable water monitoring
* Chemical leakage detection in rivers
* Swimming pool remote measurement
* Pollution levels in the sea
* River Floods.

Retail :-

* Monitoring
* Intelligent Shopping Applications
* Marketing.

Daily Life :-

* Remote control Applications
* Sensor - Based Furniture

Industry :-

* M2M Applications
* Indoor Air Quality
* Temperature Monitoring.

Services :-

* IOT Service design assumes the combined and possibly ordered activation of elementary capabilities or Service Functions (SF)

* IOT service complexity suggests robust mastering of chained SF activation and operation

* Forwarding and routing, firewall, QoS, DPI, etc.

* For the sakes of optimized resource usage and reliable service delivery.

UNIT-6

DATA COLLECTION, STORAGE AND COMPUTING USING A CLOUD PLATFORM FOR IOT/M2M APPLICATIONS/SERVICES

Syllabus:

* Introduction for Data collection, Storage and computing using a cloud platform for IOT/M2M application/services.
* Data collection, Storage and computing using cloud platform.
* Everything as a service and cloud service Models.
* IOT cloud-based services using the Xively (Pachube/cosm), Nimbits and other platforms.
* Sensor - Participatory Sensing, Actuator, Radio Frequency Identification, Wireless sensor Network Technology, Sensors Technology, Sensing the World.

Objectives:

* Outline Cloud computing Paradigm for data collection, Storage and Computing Services
* Describe Cloud Computing Service models in a software architectural concept everything as a service.
* Learning the sensor technology for sensing the real world using analog and digital sensors and examples for Sensing devices for IOT and M2M

* Learning the concepts of participatory Sensing, industrial IOT and automotive IOT.
* Learning the uses of actuators in devices and data communication
* Learning Radio Frequency Identification and Wireless Sensor Network technology.

6.1 Introduction for Datacollection, storage and computing using a cloud platform for IOT /M2M application /services.

A few conventional methods for data collection and storage as follows:

→ Saving devices data at a local server for the device nodes

→ Communicating and saving the devices data in the files locally on removable media, such as micro SD Cards and computer hard disks.

→ Communicating and saving the data and results of Computa-tions in a dedicated data store or Coordinating node locally.

→ Communicating and saving data at a local node, which is a Part of a distributed DBMS.

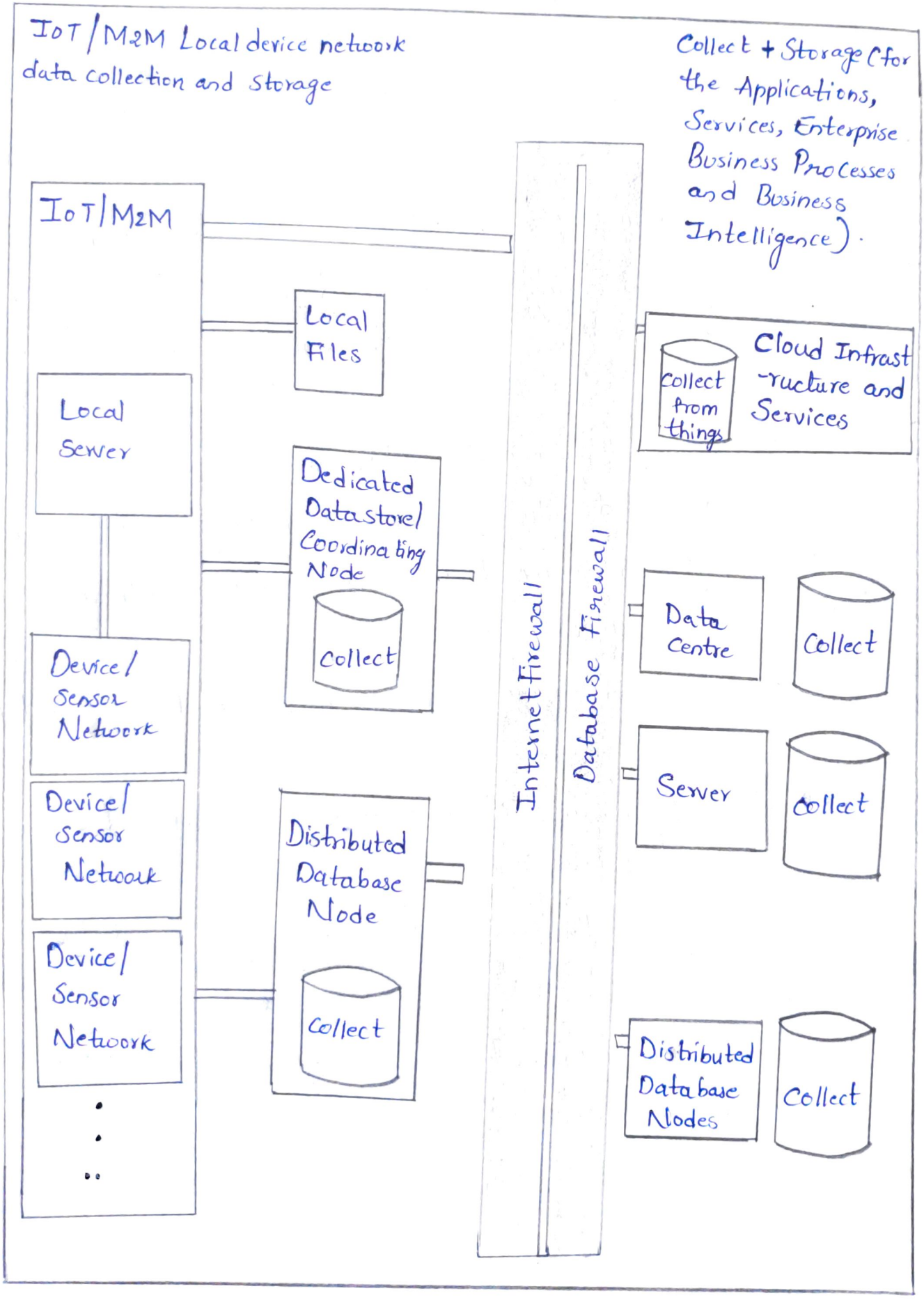
→ Communicating and saving a remote node in the distributed DBMS.

→ Communicating on the Internet and saving at a datastore in a web or enterprise server

→ Communicating on the Internet and saving at a data Centre for an enterprise.

Cloud is a new generation model for data collection, storage and computing.

6.2 Data collection, Storage and Computing using cloud platform.



6.2 Data Collection, Storage and Computing using Cloud platform.

Different methods of data collection, storage and Computing are:

i) Devices are Sensor networks data collection at a device Webserver.
ii) Local files.
iii) Dedicated data store centre at Coordinating node
iv) Local node in distributed DBMs.
v) Internet-connected data centre.
vi) Internet - connected Server
vii) Internet - connected distributed DBMs nodes and
viii) Cloud infrastructure and services.

*Cloud computing Paradigm is a great evolution in Information and Communications Technology (ICT). The new Paradigm uses XAAS at the Internet connected Clouds for Collection, Storage and Computing.

* The key terms and meanings, which need to be understand before learning about the Cloud Computing platform. The terms and their meanings are:

Resource: It refers to one that can be read, written or executed. A path specification, data point, pointer, data, object, datastore or method can also be Resource.

System Resource: It refers to an operating system (OS), memory, network, server, software (or) application.

Environment: It refers to an environment for programming, Program execution or both. For example: Google App Engine environment for creation and execution of Web applications in Python or Java.

Platform: It denotes the basic hardware, Operating System and network, and is used for software applications (or) Services over which programs can be run or developed.

Edge Computing:- It is a type of Computing that pushes the frontier of computing applications, data and services away from Centralised nodes IoT data generating nodes, that means at logical extremes of the network.

Distributed Computing:- It refers to computing and usage of resources which are distributed at multiple computing environments over the Internet.

Service:- It is a software which provides the capabilities and logically grouped and encapsulated functionalities. A service is called by an application for utilising the capabilities.

WebService:- According to W3C definition, it is an application identified by a URI, described and discovered using the XML based Web-Service Description Language.

Service-Oriented Architecture:- It consists of components which are implemented as independent services which can be dynamically bonded and orchestrated and which possess loosely coupled configurations, while the communication between them uses messages.

Web Computing:- It refers to computing using resources at computing environment of Web server(s) or Web services over the Internet.

Grid Computing:- It refers to computing using the pooled inter-connected grid of computing resources and environment in place of Webservers.

Utility Computing:- It refers to computing using focus on service levels with optimum amount of resources allotted when required and takes the help of pooled resources and environments for hosting applications.

Cloud computing:- It refers to computing using a collection of services available over the Internet that deliver computational functionality on the infrastructure of a service provider for connected systems and enables distributed grid and utility computing.

Key Performance Indicator:- It refers to set of values, usually consisting of one or more raw ~~material~~ monitered values including minimum, average and maximum values specifying the scale.

Localisation:- It means cloud computing content usage is monitored by determing localisation of QoS level and KPIs.

Seamless Cloud Computing:- It means during computing the content usages and computations continue without any break when the service usage moves to a location with similar QoS level and KPIs.

Elasticity:- It denotes that an application can deploy local as well as remote applications or services and releases them after the application usage.

Measurability:- It is something which can be measured for controlling or monitoring and enables report of the delivery of resource or service.

Homogenicity:- Homogenity of different computing nodes in a cluster or clusters refers to integration with the kernel providing the automatic migration of the processes from one to other homogeneous nodes.

Resilient Computing:- It refers to the ability of offering and maintaing the accepted QoS and KPIs in presence of the identified challenges, defined and appropriate resilience metrics, and protecting the service.

Scalability:- In cloud services refers to the ability using which an application can deploy smaller local resources as well as remotely distributed servers and resources.

Maintainability:- In cloud services refers to the storage, applications, compoting infrastructure, services, data centres and servers maintenances which are responsibilities of the remotely connected cloud services with no costs to the user.

XAAS:- It is a software architectural concept that enables deplogment and development of applications and offers services using Web and SOA.

Multitenant:- Cloud model refers to accessibility to a cloud platfom and compoting environment by multiple users who pay as per the agreed QoS and KPIs, which are defined at separate SLAs with each user.

The following subsections describe the cloud computing paradigm and deployment models.

6.2.1 Cloud computing paradigm:-

Cloud computing means a collection of services available over the Internet. Cloud delivers the compatational functionality. cloud computing deploys infrastructure of a Cloud-service Provider. The infrastructure deploys on a utility or grid computing or Web-services environment that includes network, system, grid of computer or sewers or data centres.

*Cloud platform Services:-

Cloud platform offers the following:

→ Infrastructure for laige datastorage of devices, RFIDs, in dustrial plant machines, automobiles and device networks

→ Collaborative computing and data store shaing.

→ Computing Capabilities, Such as analytics, IDE (Integrated Development Environment).

* Cloud platform Usages:-

Cloud platform usages are for connecting devices, data, APIs, applications and services, persons, enterprises, businesses and XAAS.

The following Equation describes a simple conceptual framework of the Internet Cloud:

Internet Cloud + Clients = User applications and services with 'no boundaries and no walls'.

An application or service executes on a platform which includes the operating systems (OS), hardware and Network.

Cloud Storage and Computing environment offers a virtualised Environment, which refers to a running Environment made to appear as one to all applications and services, but in fact Physically two (or) more running environments and platforms may be present.

* Virtualisation:-

Virtualisation enables provisioning for storage, network functions, Server and desktops in execution environment of multiplatforms and Servers.

Applications need not be aware of the platform, just Internet connectivity to the platform called cloud platform.

The storage is called Cloud storage. The computing is called Cloud computing. The services are called Cloud services in line with the web services which host on Web servers.

→ Virtualisation of storage :- It means user applications or service access physical storage using abstract database interface or file system or logical drive or disk drive, though in fact storage may accessible using multiple interfaces or services. For example: Apple iCloud offers storage to a user group that enables the sharing of album, music, videos, data store, editing ~~fooct~~ files and collaborating among the user group members.

→ Network Function Virtualisation (NFV) :- It means user application or service accesses the resources appearing as just one network, though the network access to the resources may be through multiple resources and networks.

→ Virtualisation of Server :- It means user application accesses not only one server but in fact accesses multiple servers.

→ Virtualised desktop :- It means the user application can change and deploy multiple desktops, though the access by the user is through their own computer platform that in fact may be through multiple OSs and platforms or remote computers.

* Cloud Computing Features and Advantages:-

Essential features of Cloud storage and Computing are:

→ On demand self-service to users for the provision of storage, computing servers, software delivery and server time.
→ Resource pooling in multi-tenant model.
→ Elasticity.
→ Massive scale availability.
→ Scalability.
→ Maintainability.
→ Homogeneity.
→ Virtualisation.
→ Resilient computing.
→ Advanced Security.
→ Low cost.

* Cloud Computing Concerns:-

Concerns in usage of cloud computing are:

→ Requirement of a constant high-speed Internet connection
→ Limitations of the Services available.
→ Possible data loss.
→ Non delivery as per defined SLA specified performance.
→ Different APIs and protocols used at different clouds.
→ Security in multi-tenant environment needs high trust and low risks.
→ Loss of User's Control.

6.2.2

* Cloud Deployment Models :-

The four cloud Deployment models are :

→ Public cloud: This model is provisioned by educational institutions, industries, government institutions or businesses or enterprises and is open for Public use.

→ Private cloud: This model is exclusive for use by institutions, industries, businesses or enterprises and is meant for private use in the organisation by the employees and associated users only.

→ Community cloud: This model is exclusive for use by a community formed by institutions, industries, businesses or enterprises, and for use within the community organisation, employees and associated users. The community specifies security and compliance considerations.

→ Hybrid cloud: A set of two or more distinct Clouds with distinct data stores and applications that blind between them to deploy the proprietary or standard technology.

6.3 Everything As a Service and Cloud Services Models.

Cloud connect the devices, data, applications, Services, Persons and business. Cloud Services can be considered as distribution Service – a service for linking the resources and for provision of coordinating between the resources.

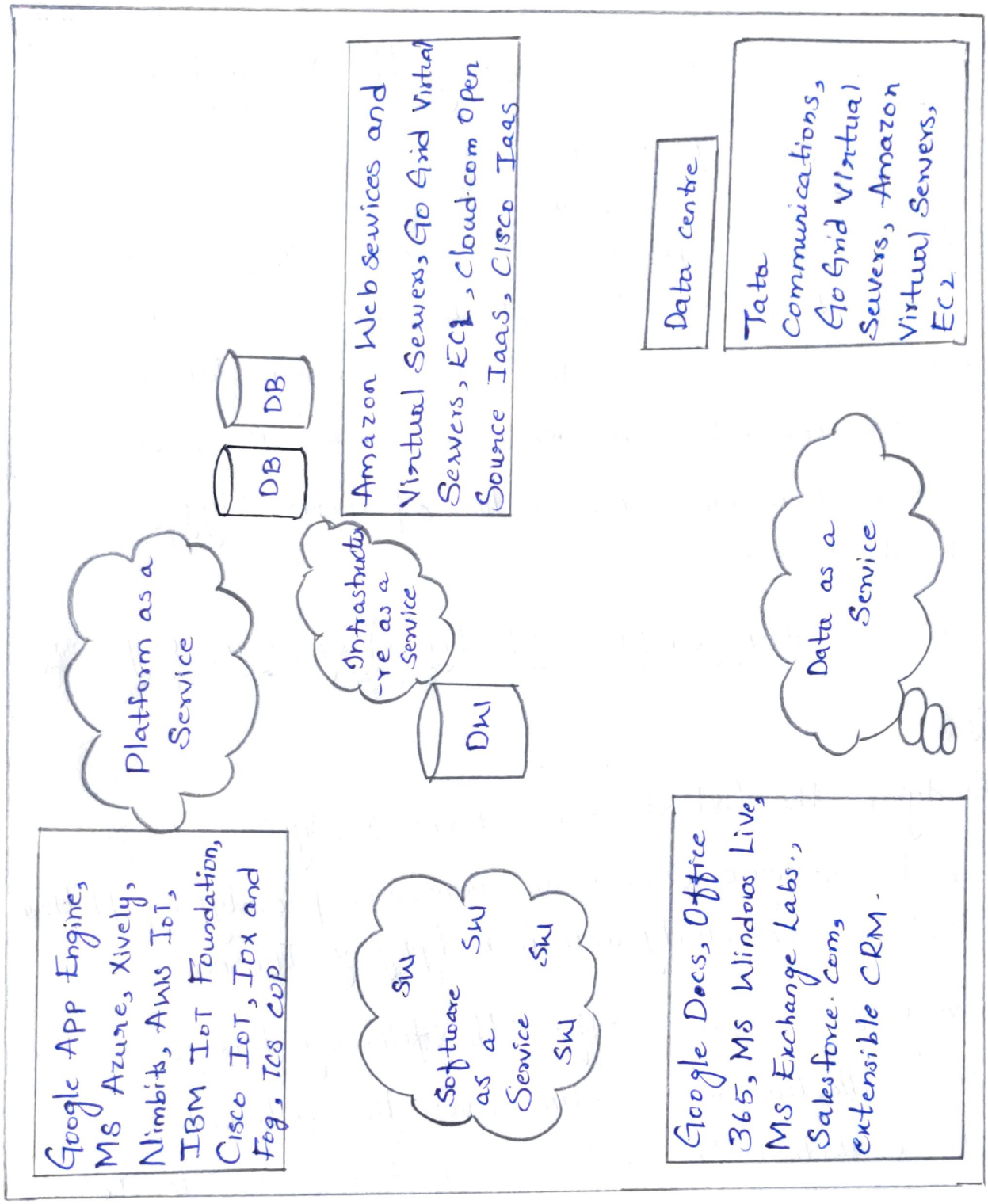
Cloud computing can be considered by a simple question:

$$\text{Cloud Computing} = \text{SaaS} + \text{PaaS} + \text{IaaS} + \text{DaaS}.$$

SaaS: SaaS means Software as a Service. The software is made available to an application or service on demand. SaaS is a Service model, Where the applications or services ~~disp~~ deploy and host at the cloud, and are made available through the Internet on demand by service User.

PaaS: PaaS means Platform as a Service. The Platform is made available to a developer of an application on demand. PaaS is a Service model where the applications and services develop and Execute using the platform which is made available through the Internet on demand for the developer of the applications.

IaaS: IaaS means Infrastructure as a Service. The infrastru-cture is made available to a user or developer of application on demand. Developer installs the OS image, datastore and application and Controls them at the infrastructure. IaaS Computing Systems, network, and Security are the responsibilities of the cloud service Provider.



PaaS, SaaS, IaaS and DaaS Cloud Service Model.

DaaS: DaaS means Data as a Service. DaaS is a service model where the data store or data warehouse is made available through the internet on demand or rent to an enterprise. The data centre management, 24x7 power, control, network, maintenance, scaleup, data replicating and mirror nodes and systems as well as physical security are the responsibilities of the data centre service provider.

6.4 IoT Cloud-Based Services Using the Xively, Nimbits and other platforms

A User is an application (or) service. The user obtains responses or feeds from the application or service. An IoT Cloud-based Service Provides for data Collection, data points, messages and calculation objections. The service also provisions for the generation and Communication of alerts, triggers and feeds to the user.

6.4.1 IoT Cloud-based Data Collection, storage, Computing using Xively:

Xively is the latest domain name. Xively is a Open Source Platform for Arduino which is an open Source prototyping platform that Provides Connectivity with web deploying Internet.

Xively PaaS Services offers the following features:

→ Data collection in real-time over the Internet.
→ Data visalation for data of connected sensors to IoT devices.
→ Graphical plots of Collected data.
→ It generates alerts.
→ Access to historical data.
→ It supports Java, Python and Ruby, and Andriod platform
→ It generatas feeds which can be real-World objects of own or others.
→ It supports REST.

Xively is based on the concept of Users, feeds, data Streams, data Points and triggers. A feed is typically a single location and data streams are of individual sensors associated with that location.

Pull or Push Methods for IoT Devices Data:-

Xively Provides two modes for data Capture, Pull method where data is collected from an http Server, and a push method where data is Written to Xively using an http client.

Data Formats and Structures:-

Number of data formats and Structures enable interaction, data Collection and Services with Xively. The support exists for JSON, XML and CSV.

Private and Public Data Access:-

A free account supports up to 10 Sensor feeds updated in near real time and the data is stored for up to 3 months.

Data Streams, Data points and Triggers:-

Xively enables data streams, data points and triggers. Data stream means Continuous Sensed data flow over the Internet.

Creating and Managing Feeds:-

Sensors or IoT devices network, such as a group of Streelights, can feed data to Xively over the Internet.

Visualing Data :-

Xively is a platform which captures data over the Internet in real-time and provides graphing, alerts and historical data access.

6.4.2 IoT Cloud-based Data Collection, storage and Computing Services Using Nimbits:

Nimbits enables IoT on an Open Source distributed Cloud. Nimbits cloud Paas deploys an instance of Nimbits Server at the device nodes. Nimbits functions as an M2M System data Store, data collector and logger with access to historical data. Nimbits architecture is a Cloud-based Google App Engine. Nimbits PaaS Services offer the following features:

→ It supports multiple programming languages including Arduino, Java Scripts, HTML.

→ It provides a data logging service and access, and stores the historical data points and data objects.

→ Storage in any format that can be serialised into a string, such as JSON or XML.

→ It filters the noise and important changes sent to another larger central instance.

→ It processes a specific type of data and can store it.

→ Time-(or) geo-stamping of the data.

→ Data visualisation for data of connected Sensors to IoT devices

The following diagram shows connected devices, Sensor nodes, network data points, Nimbits Servers, deployment at the device network nodes and networked with the Nimbits Server at Cloud for applications and services.

Data Points:-

A datapoint means a collected value of a sensor in a group of sensors. Data points Organise the data in a no. of ways. Points can be in the folders. The folders can go as deep as like in a tree a folder having several subfolders.

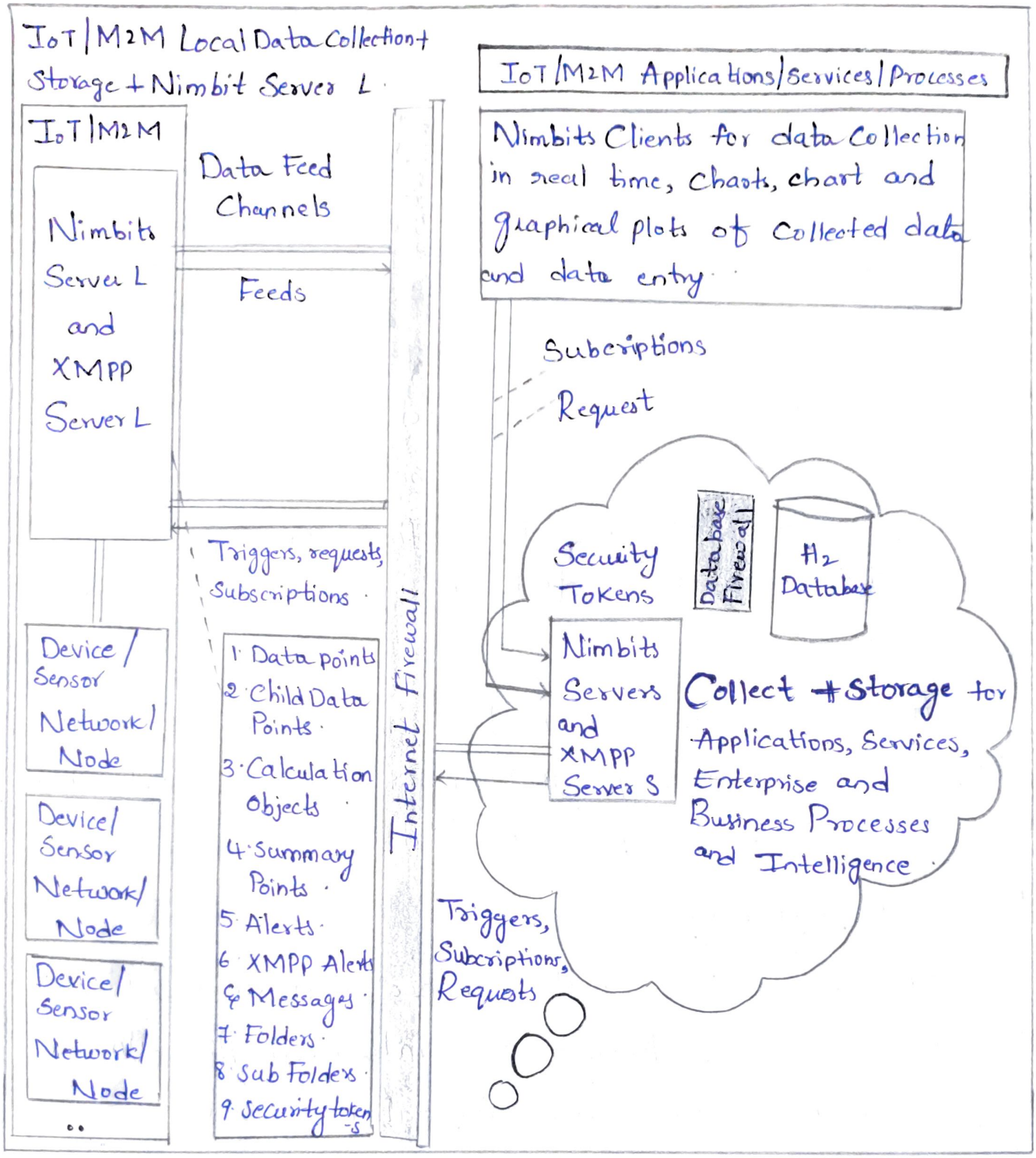
Data Channels:-

A user can create a data feed channel which shows the system events and messages that also shows data alerts which are subscribed to showup in the feed.

Using Advanced Features:-

An application can create a connection to another Nimbits application or service. Nimbits 3.6.6 introduced H2 database engine. Nimbits 3.8.10 includes H2 database engine. H2 is Java SQL database. APIs are in Pure Java, The main features of H2 are:

→ Very fast, Open Source, JDBC API.
→ Embedded and Server modes; in memory databases.
→ Encrypted database.
→ ODBC driver.
→ Full text Search
→ Multi Version Concurrency.



Security Tokens :-

Nimbits 3.9.6 Provides Security tokens in a new way.

Breakthough Performance and Data Integrity :-

Nimbits Server 3.9.10 Version launched in June 2015 Provisions for the breakthrough Performance and data integrity.

Alerts:-

A filter means applying some rule to get new data for a data point. The filter item in the tree called "Dh" is for XMPP akrts.

Jabbing:-

Jabbing means pushing the alerts or messages down quickly or pushing repeatedly. Each type of message (or) alert is assigned a Jabber ID called JID.

Subscriptions:-

A user can create many subscriptions for a single point. It may subscribe to one of the points, other user, or anyone's Public point to get the alerts.

6.43 Using Public Cloud IoT Platforms:-

Many Cloud PaaS and SaaS platforms are now available for IoT. The examples of Cloud-based platforms are:

Cloud Platform	Features.
*Spark	Distributed, Cloud-based IoT Operating System and Web-based IDE includes a Command-line interface, support for multiple languages and libraries for working with many different IoT devices.
*OpenIoT	Open source middleware, enables communication with Sensor Clouds and enables Cloud-based Sensing as a service and developed use

	cases for smart agriculture, intelligent manufacturing, Urban crowd Sensing, Smart living and Smart Camposes.
* Device Hub	Open Source backbone for IoT, a Cloud-based Service that stores IoT-related data, Provides data visualisations, allows Web-page-based consol to control IoT devices, enables developers to create applications such as tracking of vehicular data, monitoring Weather data.

6.5 Participatory Sensing:-

A web source defines Participatory Sensing as a "sensing by the individuals and groups of people contributing sensory information to form a body of knowledge. Participatory Sensing is the process coberebg individuals and communities use evermore-capable mobile phones and cloud services to collect and analyse systematic data for use in discovery.

A Participant of PS process can be Sensors used in mobile phones. Mobile phones have camera, temperature and humidity sensors, an accelerometer, a gyroscope, a compass, infrared sensors, NFC sensors, bar or QR code readers, micro-phone and Gps. Mobiles communicate on the Internet the Sensed information with time, date and location stamps.

Applications of PS include retrieving information about Weather, environment information, pollution, waste management, road faults, health of individuals and group of people,

* Phase 1 is coordination, in which the participants of a PS Process Organise after identifying the Sources.
* Next two phases, phases 2 and 3 involves data capture, communication and storage on servers or cloud.
* Next two phases, phases 4 and 5 involves PS data Processing and analytics, visualisation and knowledge discovery.
* Phase 6 is for initiating appropriate actions.

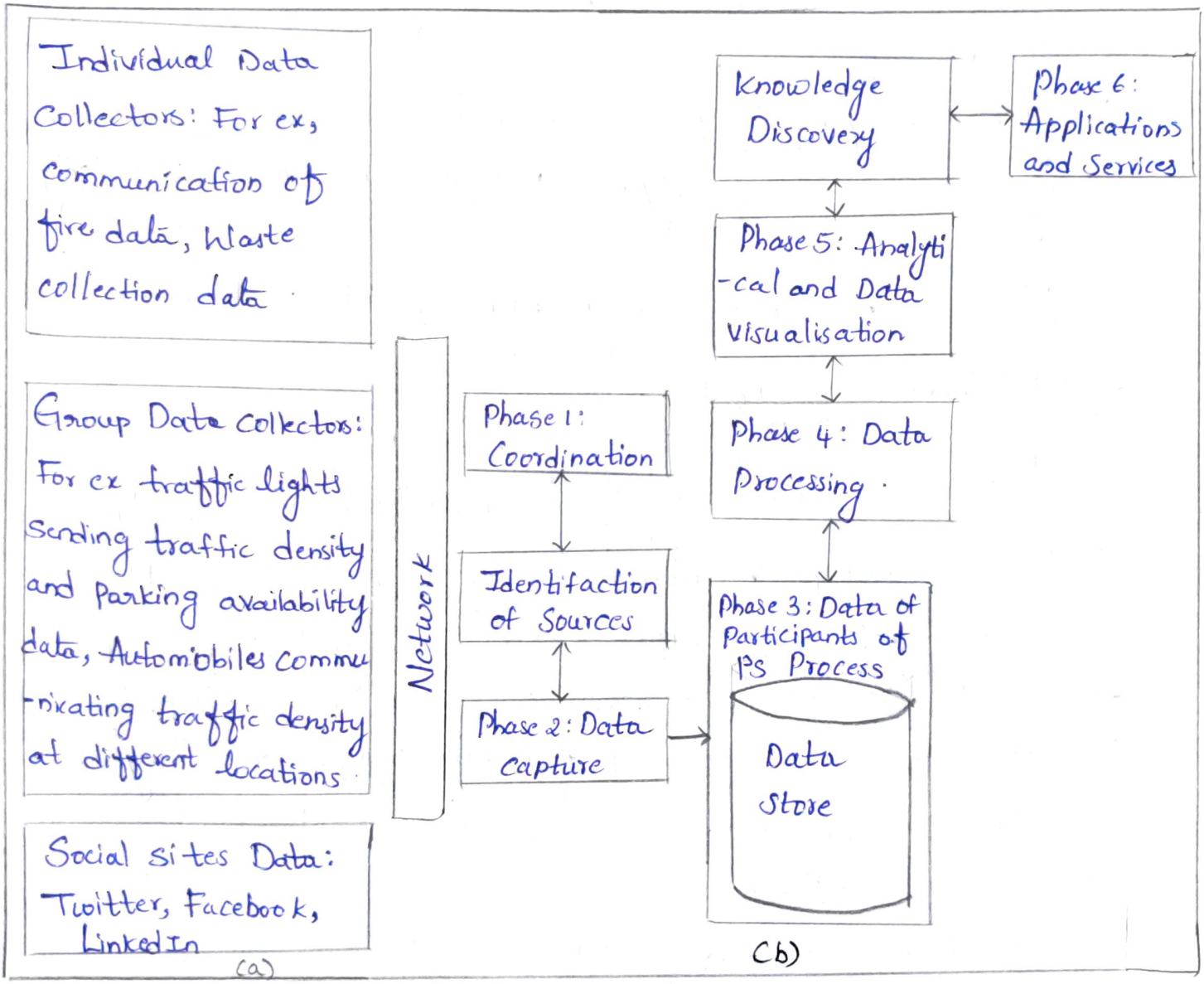
6.5.1 Industrial IoT:-

Industrial Internet of Things (IIoT) involves the use of IoT technology in manfacturing. IIOT involves the integration of complex physical machinery M2M communicat-ion with the networked sensors and use of software, analytics, machine learning and knowledge discovery.

IIoT applications are in the manfacturing, railwa-ys, mining, agriculture, oil and gas, utilities, transport-ation, logistics and healthcare services.

Industrial Internet Consortium (2014) is body which has been founded for creation of standards, open interoperability and the development of architectures for Industrial Internet of Things (IIOT).

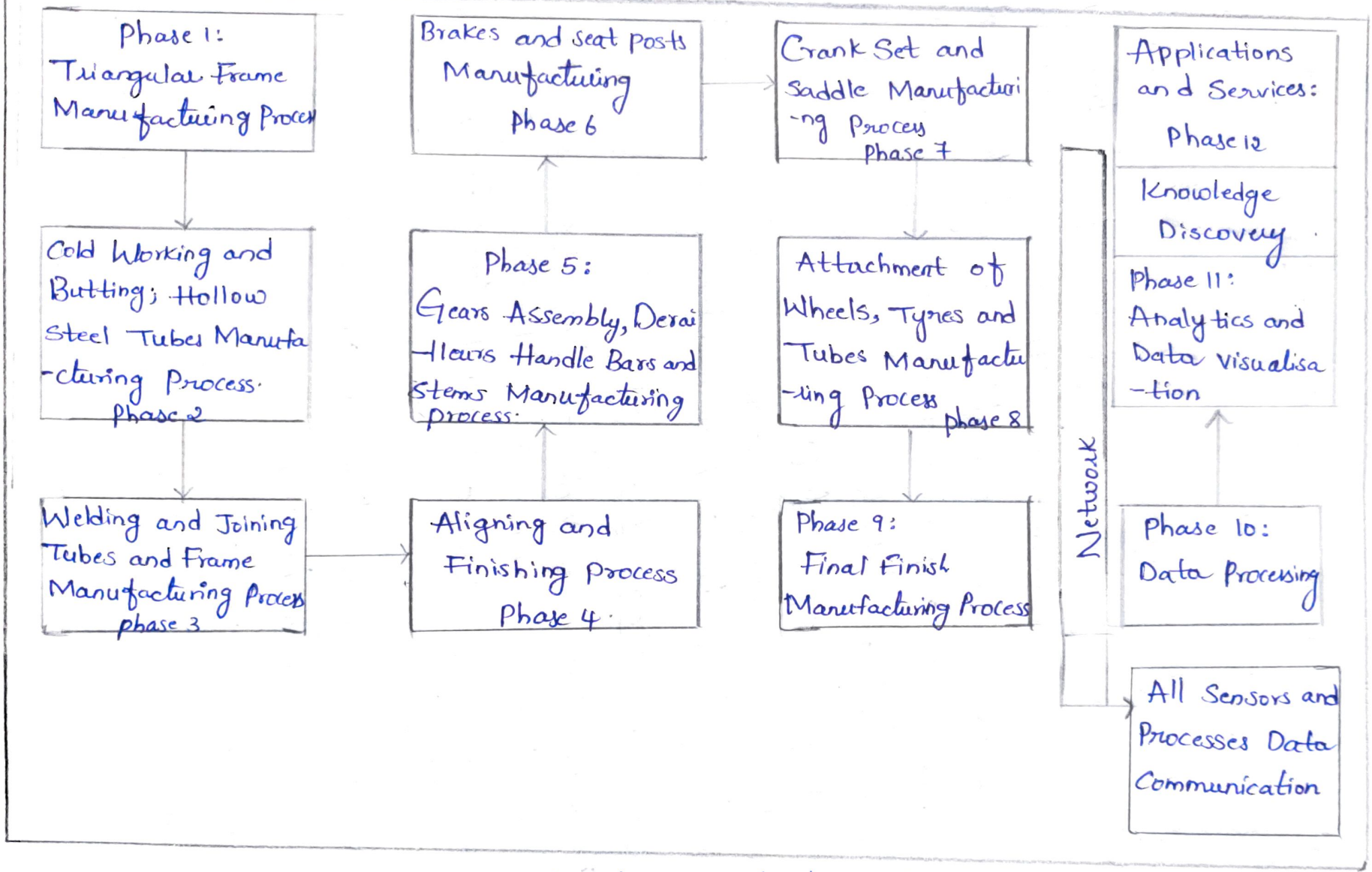
traffic congenstion, urban mobility, on disaster management, such as flood, fire, etc. Participatory Sensing has many challenges such as - Security, Privacy, repatation and ineffective incentivies to Participating entities.



(a) Sources of data in the PS Processes

(b) phases of a Participatory Sensing process for IoT applications and Services.

In (a) fig it shows the sources of data in the PS process for IoT applications. In fig(b) it shows the phases of PS Process



IIoT Phases in the bicycle manufacturing Process.

6.5.2 Automative IoT:-

Automative IoT enables the connected cars, Vehicles-to-infrastructure technology, Predictive and Preventive maintenances and autonomous cars

Connected Cars Technology:-

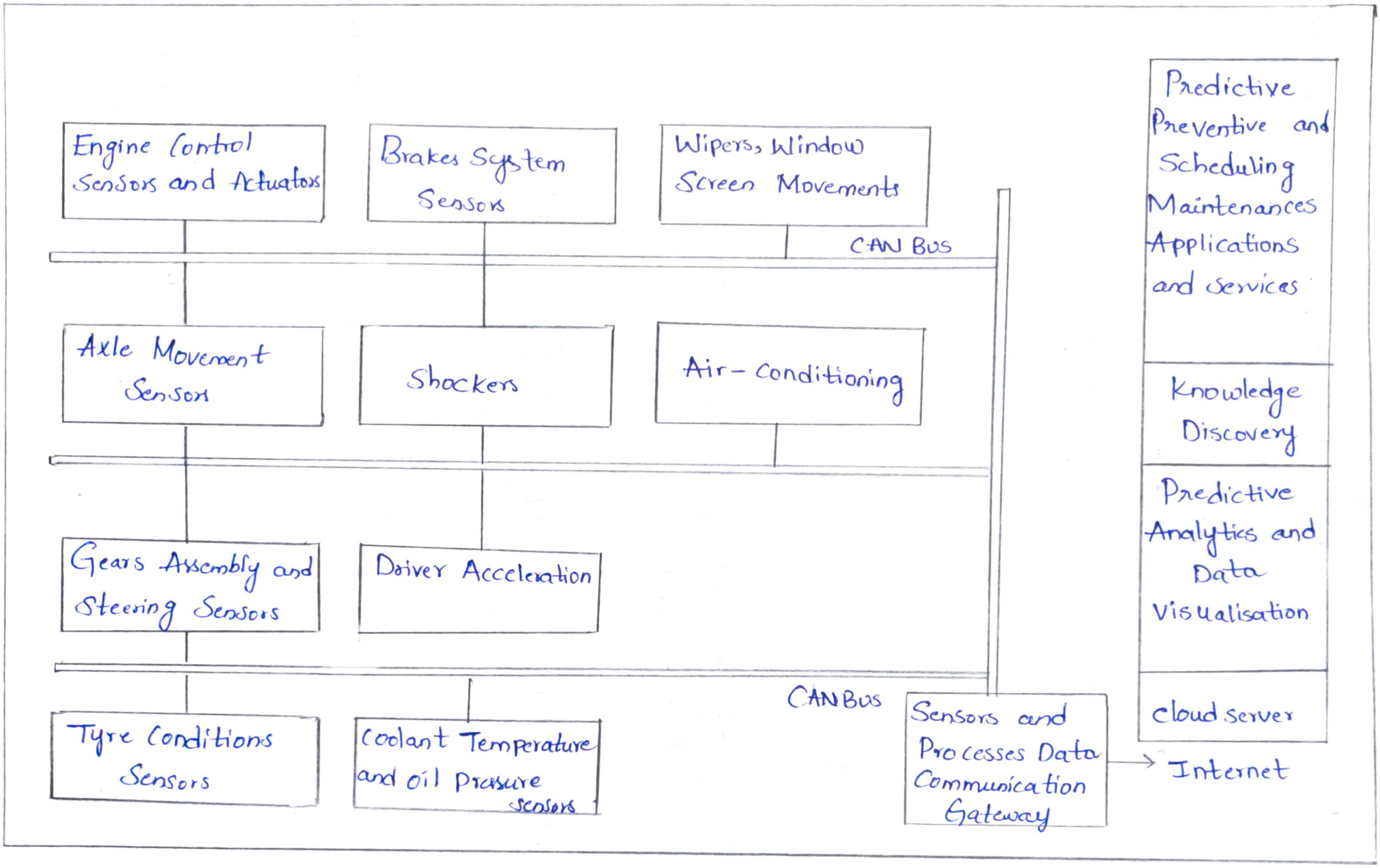
A connected car with the combination of GPS tracking and an Internet connection enables applications such as:

1) Display for driver that enables driving through the shortest route, avoiding the congested route, etc.
2) Get notifications about traffic.
3) Protecting cars against theft
4) Weather and enroute destinations.
5) Keeping a tab on driver's health and behaviour.

Vehicle-to-Infrastructure Technology:-

Automotive IoT enables Vehicle-to-Infrastructure (V2I) technology. A Vehicle communicates with other Vehicles, the Surrounding infrastructure and a WiFi LAN. Examples of V2I applications are:

1) Alerts and Warnings for forward collision.
2) Information about blind spots.
3) Notification about a vacant parking space.
4) Stream live music and News.


Engine Control Sensors and Actuators
Brakes System Sensors
Wipers, Window Screen Movements
CAN Bus
Axle Movement Sensor
Shockers
Air-Conditioning
Gears Assembly and Steering Sensors
Driver Acceleration
CAN Bus
Tyre Conditions Sensors
Coolant Temperature and oil Pressure sensors
Sensors and Processes Data Communication Gateway
Internet
Predictive Preventive and Scheduling Maintenances Applications and services
Knowledge Discovery
Predictive Analytics and Data Visualisation
cloud server

6.6 ACTUATOR:-

An actuator is a device that takes actions as per the input command, pulse or state (or) sets of 1s and 0s, or a control signal. An attached montor, speaker, LED or an o/p device converts electrical energy into physical action.

Examples of applications of actuators are:

* Light Sources
* LEDs
* Speakers
* Solenoids
* Servomotor
* Relay Switch
* Switching ~~of~~ on a set of Street lights
* Ringing of alarm bell.

Light Sources:-

Traffic lights are examples of function of light Sources as actuators controlled by the inputs.

LED:-

LED is an actuator which emits light or infrared radiation Uses of different colour LEDs.

Speakers:-

A Speaker enables synthesised music tunes and sounds. The appropriately programmed pulses generate the music, Sounds, buzzers and alarms when they are the input to the Speaker.

Solenoid :-

A Solenoid is an actuator consisting of a no of cylind -rically wound coils. The flow of current creates a magnetic field in proportion to the no. of turns in the Solenoid and the current in it.

Servomotor:-

Servomotor is geared Dc motor for applications such as robotics. It rotates the shaft of a motor.

Relay switch:-

An electronic switch can be controlled by the input 1 or 0. From the port pin of a microcontroller or through a push button switch and battery.

6.7 Radio Frequency Identification:

RFID is an identification system using tagging and labelling of objects.

6.7.1 RFID IOT systems:-

A tag enables identification of an object at different locations and times. A Product, Parcel, Postal article, Person, bird, animal, Vehicle cor) object can have a tag or label in order to make the identification feasible.

The Reader circuit of an ID can cause UART or NFC Protocol to identify the tag, When RFID tag is

at a distance less than 20cm.

A hotspot consists of a wireless transceiver (or) Wi-Fi transceiver for Internet connectivity. It receives signals from a no. of RFID tags in an organisation and transmits the data to the Webserver over the Internet.

→ Principle of RFID:-

A tag is an electronic circuit which transmits its ID using RF signals. The ID transmits to a reader, then that transmits along with the additional information to a remote Server or cloud connected through the Internet.

An RFID tag has an advantage over a barcode or QR code in terms of simpler processing of the RFID data. It can also be made invisible to a person.

→ RFID IOT Network Architecture:-

It Explained a four layered ITU-T reference model for the Internet of RFIDs, individual capabilities of the layers and data interchange.

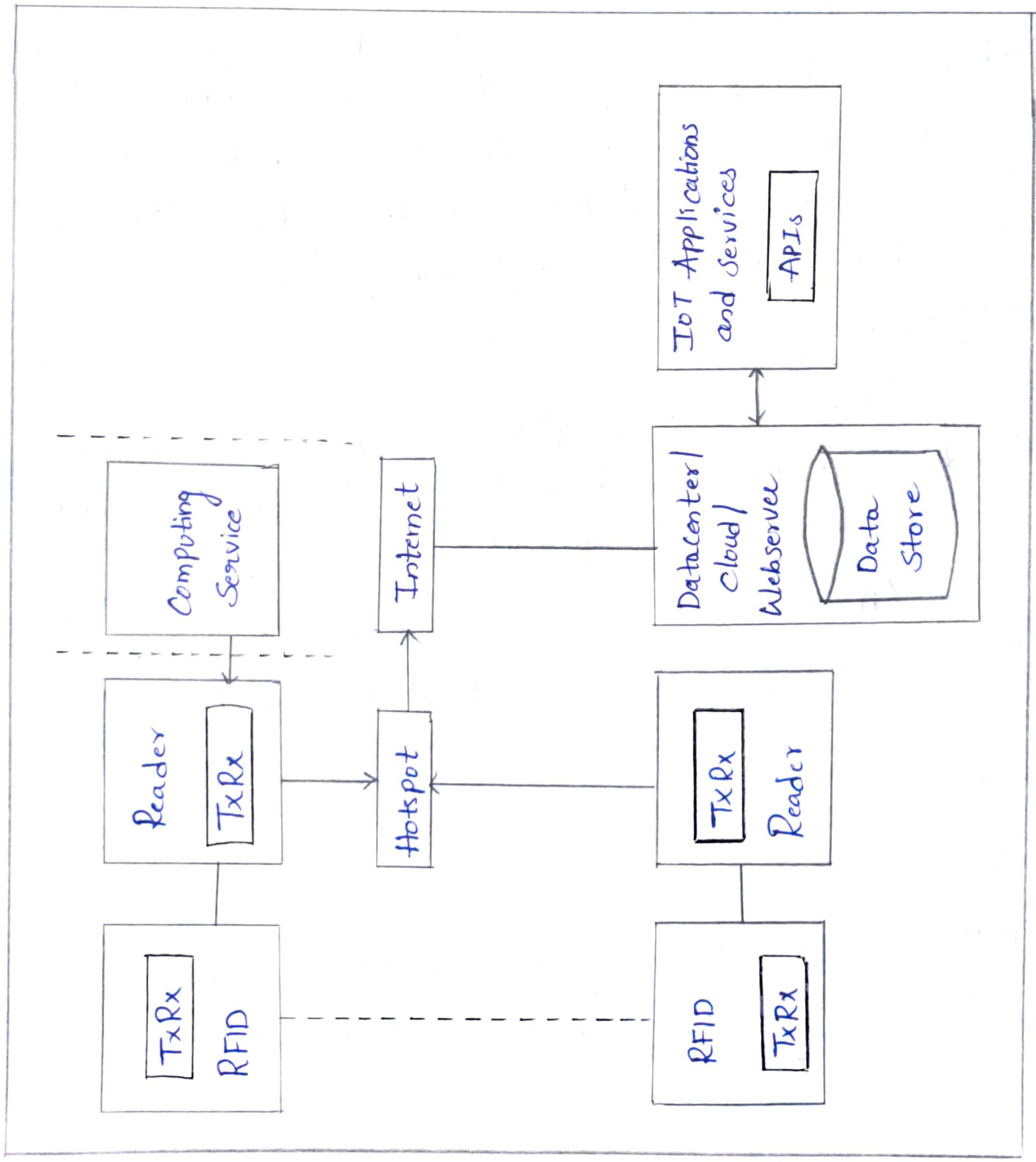
→ RFID IOT Applications:-

Examples of RFID IOT Applications are tracking and inventory control of goods, supply, chain systems, business Processes Such as for payment, leasing, insurance and quality management, access to buildings and road tolls (or) Secured store centre entries.

→ Components of an RFID System :-

The components needed in a system for IoT applications and services. The components of an RFID system are:

* Transceiver
* Data Processing subsystem
* Middleware



Components needed in a system for RFID IoT applications and services.

* Transceiver:-

A transceiver is in-built at the chip. It Communicates in a range 10cm to 200m according to the chip. Standard frequency range used can be between 120 KHz to 150 KHz, 13.56 MHz, 433 MHz, higher when Using UHF and Microwave frequencies.

* Data Processing subsystem:-

A reader associates a data processing subsystem which consists of a compoting device and a middleware and provides connectivity to the Internet, directly (or) through a gateway which includes a data adaption sublayer. The subsystem is a backend system.

* Middleware:-

Middleware are software components used at the reader, read manager, data store for the transaction data store and APIs of the applications.

Applications and services and other associated applica-tions and software use the data store at the cloud or Web Server.

→ Issues:-

The issues are:

1) Design issue:- Designing a unique ID system needs a Standard global framework.

2) Security issue: A tag is read only. It can thus interact with any reader and thus allows automat-ed external monitoring.

3) Cost issue:- RFID tag and reader become costly with data Processing and security enhancing technology.

4) Protection issue:- The tag needs protection from the adverse weather condition which may damage the tag.

5) Recycling issue:- Recycling of the tags can be an environment-al concern.

6) Active life issue:- Active RFID, which consists of battery, has limited life of up to 2 to 4 years.

6.7.2 EPC global Architecture Frame Work:-

MIT Auto-ID Labs is a group which consists of research laboratories of seven Universities. The group has designed an architectural framework for IoTs. The group works in the field of networked RFIDs and emerging Sensing technologies. The group works together with EPCglobal research group.

→ EPICS and ONS, Design Issues:-

EPC Information Service (EPICS) is a design of EPC global standards which enables EPC related data sharing within and across enterprises.

The EPC global architecture defines the following:-

* EPCIS Capturing Application (ECA) for capturing EPC-related data required for business processes.
* EPCIS Accessing Application (EAA) for enterprise business Processes supported by data captured using ECA.
* Partner applications such as postal tracking system connected with payment systems.

Design issues include governance model and architecture of the ONS. The concerns involved as:

→ Political for Control over information.
→ Capability issue of loss of domestic and strategic capability.
→ Security issue of collecting business intelligence.
→ Commercial issues.
→ Innovation Control issues.

→ Technological challenges:-

RFID technology challenges are as follows:

* Effective implementation at data processing subgystem consisting of reader and tag protocols, middleware architecture and EPC standards.
* Need of low cost tags and RFID technology.
* Design robustness.
* Data Security.

→ Security Challenges:-

The issues associated with RFID Security are:

* Discovery of foreign attacks and maintain overall data integrity.
* Unauthorised disabiling of a tag by a reader which is external, thus making the tag useless.
* Unauthorised tag manipulation by a reader which is external, thus making the tag useless.
* Cloning of the tag by anunauthorised entity.

→ IP for an RFID in Internet of RFIDs:-

Data from the RFID reader after filtering, aggregation and routing get stored at an IP address. This data is in XML format. Data accesses using HTTP and SOAP Protocols.

Internet Protocol (IP), IPV6 is 128-bit IP address, which is required for routing data on the Internet. It needs to me mapped with the 96-bit EPC. The EPC is header, manufacturer, product and Serial numbes bits.

6.7.3 Web of Things of RFIDs:-

Web of Things (WOT) means making objects a part of the World Wide Web. WOT software, architecture style such as JSON, REST, JSON and programming pattern such as Web Sockets, makes this feasible. WOT data store of objects is similar to Web pages store. The Web

receives and sends data using the Internet.

6.8 WIRELESS SENSOR NETWORS TECHNOLOGY:-

A set of sensors can be networked using a wire-less system. They cooperatively monitor the physical and environmental conditions, such as temperature, sound, vibration, Pressure or multiple and remote locations.

6.8.1 WSN Concepts:-

Definition:-

WSN is defined as a network in which each sensor node connects wirelessly and has the capability of computation, for data compaction, aggregation and analysis.

→ **WSN History:-**

Sound-Waves-based surveillance and tracking Systems for enemy Submarine were used in the 1950s. Wireless-based networked radars were also used during this time. Research on the Distributed Sensor Networks (DSN) using network communication started before 1980.

WSNs have a large no of IOT applications. Examples are smart homes and smart city.

→ Context-based Node Operations:-

A WSN Node can adapt, re-program or do another task using a sensor and associated circuit, computations, networking capabilities and context at that node.

Context can be a physical, computing, user, structural or temporal context. Following may correspond to the context for re-programming of the actions of the WSN nodes:

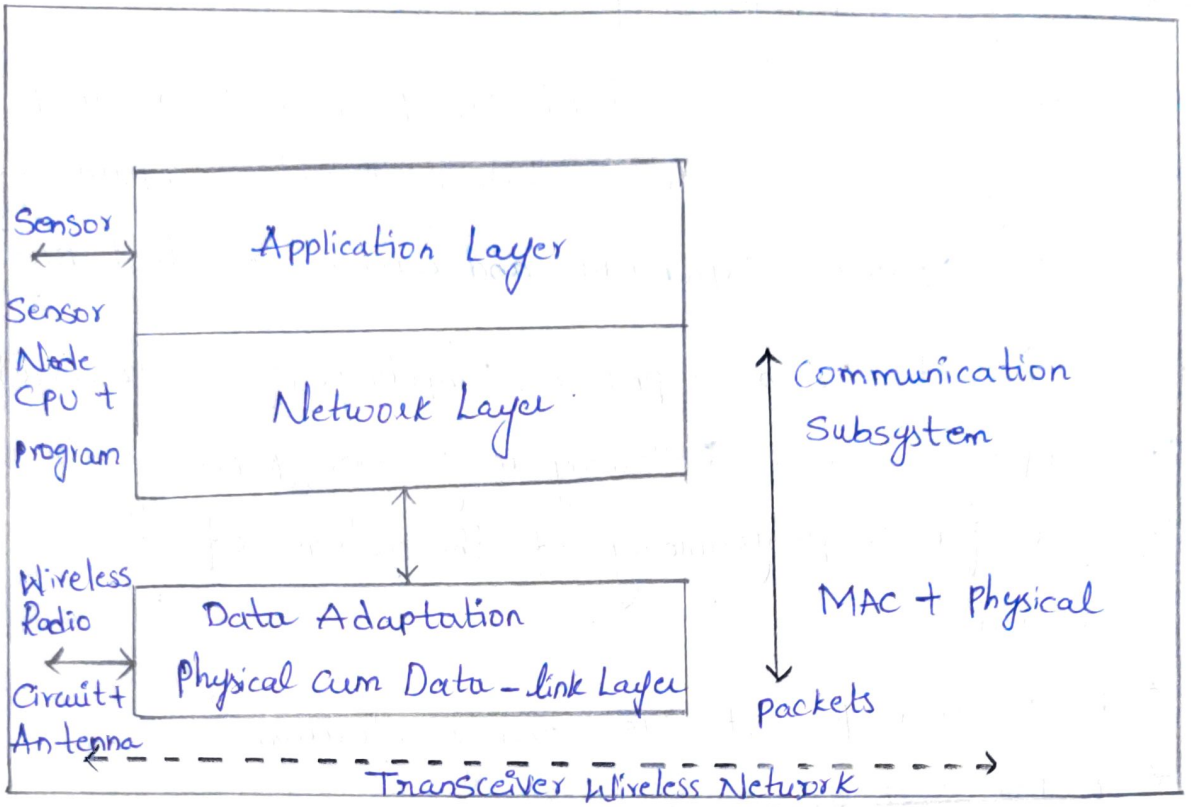
* Past and present Surrounding situations.
* Action such as the present network.
* Surrounding device or systems.
* Changes in the state of the connecting network.
* Nearest Connectivity currently available.
* Previously changed data records.

6.8.2 WSN Architecture:-

WSN Node Architecture:-

The three-layer architecture of a node. The three layers are application layer, network layer, physical cum data-link layer.

The application layer software components are Sensor management, sensor query and data dissemination, task assignment, data advertisement and application-specific protocols.



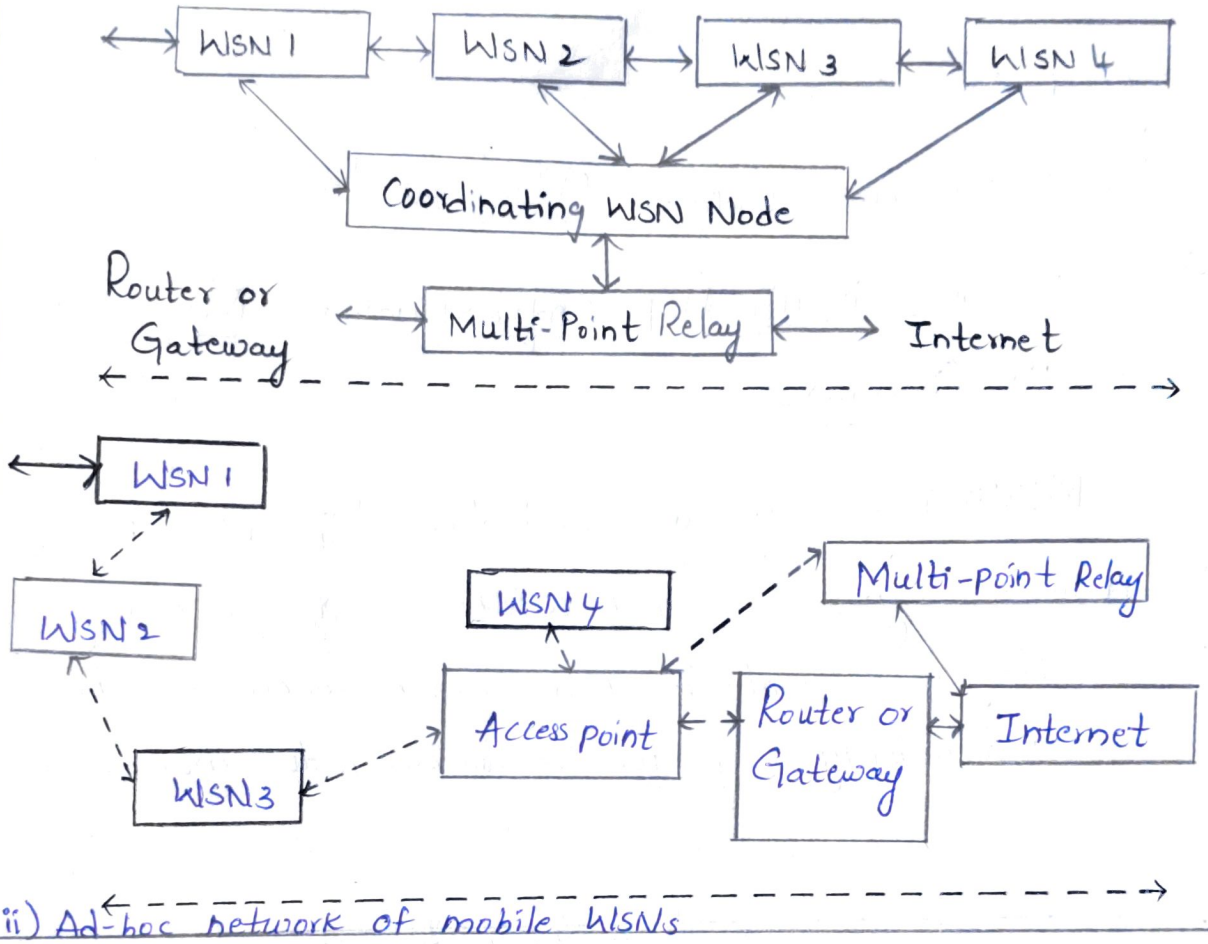
Architecture of a Wireless Sensor Node

Sensor, CPU and program sensor node constitute the application and network layers. Network layer links serially to the data-link layer. A serial link interconnects the layers to a wireless radio circuit and antenna.

Architecture for connecting nodes:-

It shows 2 architectures for connecting WSN nodes, fixed connecting infrastructure of WSN nodes, coordinators, relays, gateways and routers and mobile ad-hoc network of WSNs, access points, routers, gateways and multi-point relays.

i) Fixed Connecting Infrastructure



ii) Ad-hoc network of mobile WSNs

An access point is a fixed point transceiver to Provide the accessibility to nodes present nearby or nodes reaching in the wireless range. A multipoint relay connect to other networks such as the Internet or mobile service Provider network.

Fixed infrastructure example is a smart home network consisting of WSN at security surveillance points, refrigerator, air conditioner, microwave, Tv and computer with Wi-Fi ~~multi-point~~ access Point.

Architecture for Networking of the Nodes:-

Two basic architectures for networking of the nodes are:

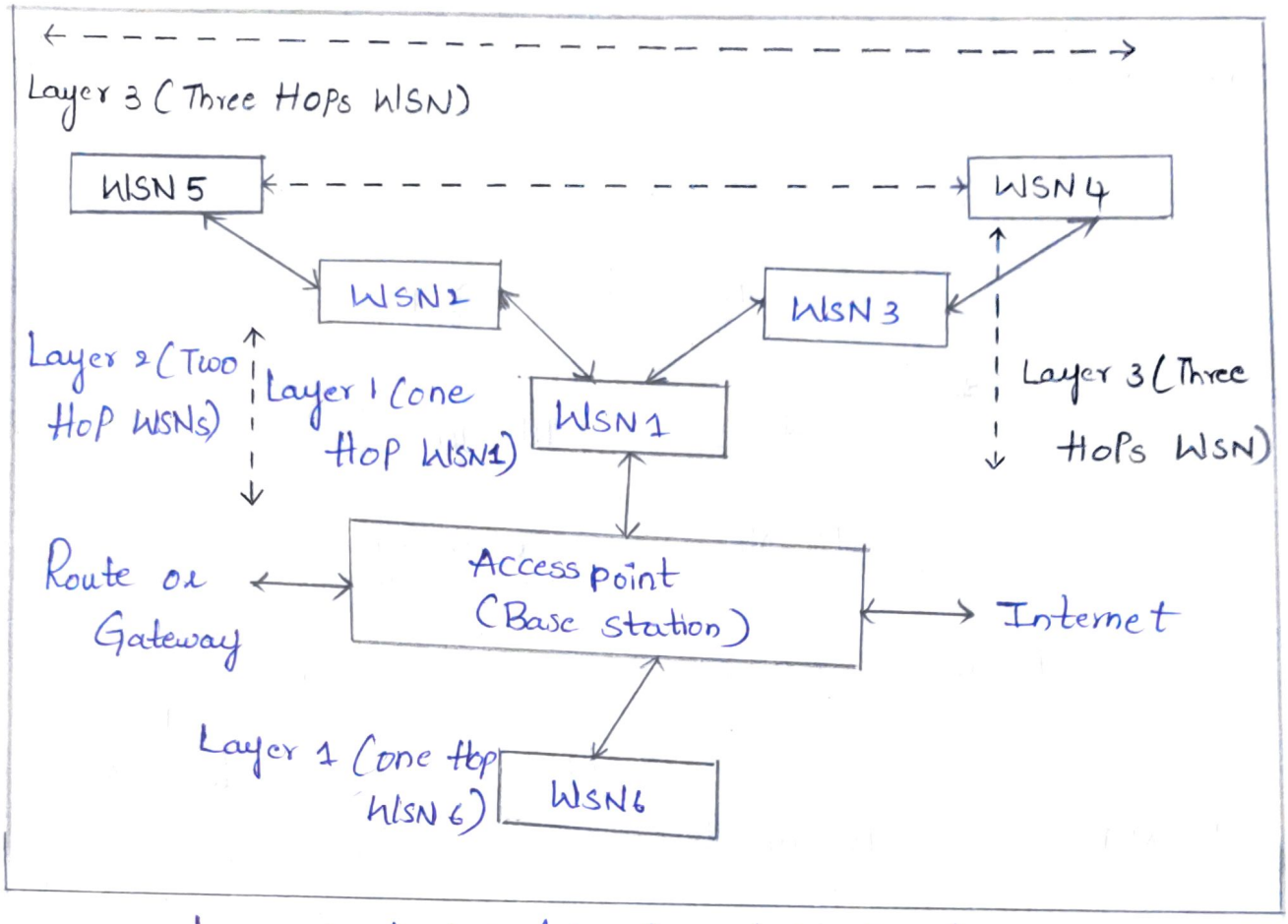
1) Layered Architecture.
2) Multicluster Architecture.

Wireless Multi-Hop Infrastructure Network Architecture (MINA):-

MINA is a layered architecture. The WSN nodes have data sensing as a well as capabilities of forwarding towards the access point. The nodes can be mobile and have coverage and mobility range for communication to remote access Points.

Each node connects to a short-distant neighbour. When the node moves to longer distances then it communicates through 2 or 3 hops to the accesspoint. Each node has low-Power transceivers to the nearest neighbouring layer WSNs.

Assume that the base station is surrounded by three layers of WSN. Layer 1 WSNs directly connect. Layer 2 WSNs first connect to layer 1 WSNs functioning as Coordinators and then directly connect. Layer 3 WSNs first connect to layer 2 WSNs functioning as coordinators, then connect to layer 1 WSNs and then connect to the access point.



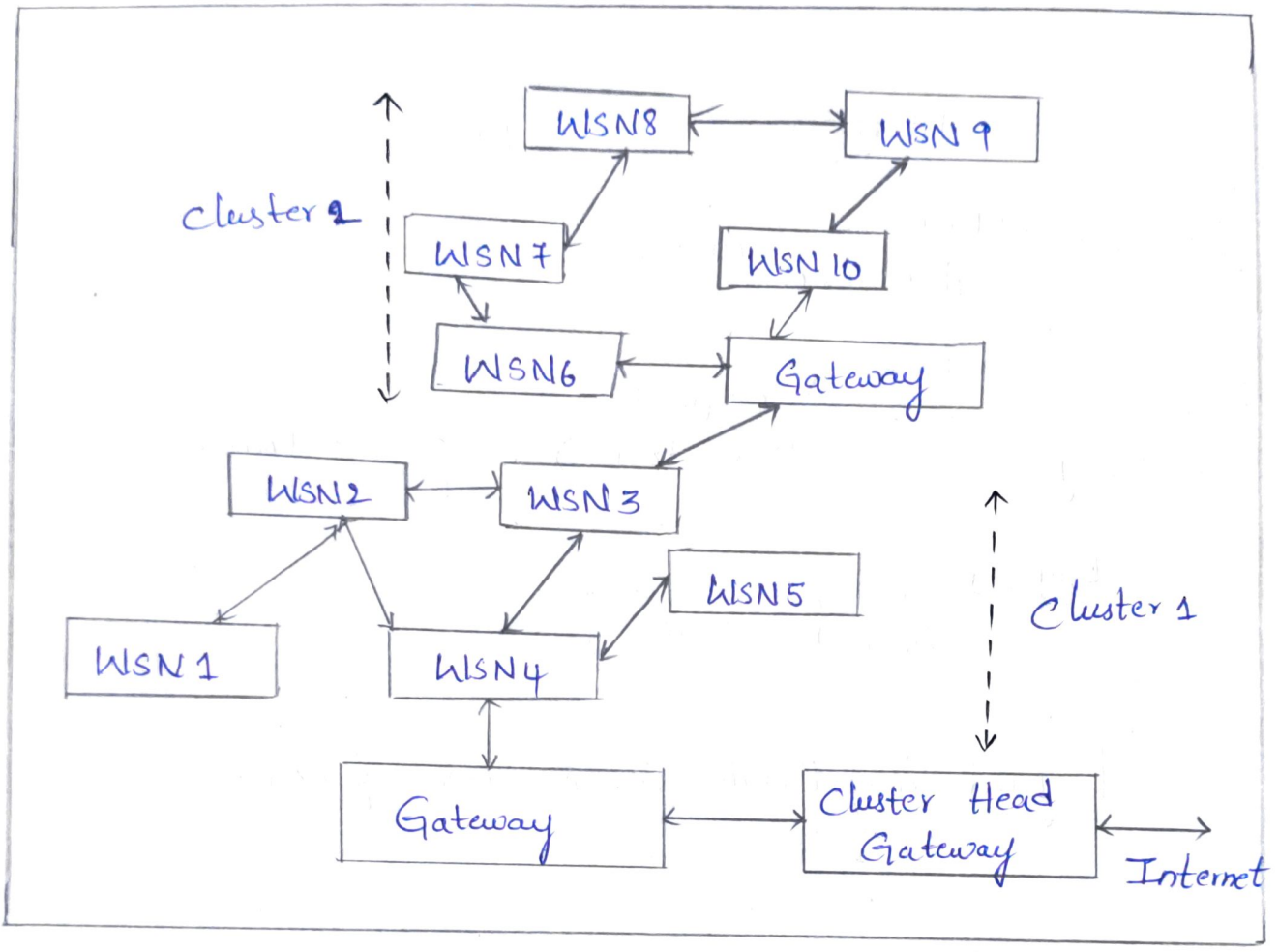
Layered Architecture for Network of nodes.

Multiple clusters Architecture:-

Each cluster has a gateway node. A set of clusters with a gateway each has one cluster with a cluster-head gateway. Multi-cluster architecture has a no of clusters, which associate a cluster-head gateway.

Cluster-head enables a tree-like topology of the clusters in a multi-cluster architecture. The cluster formation and election of cluster-heads is autonomous in distributed WSNs

and WSN clusters.



Multi-cluster architecture for network of nodes.

A number of clusters associated with the cluster-head gateway depends on the coverage required in the Network. Cluster 1 is an ad-hoc network of mobile WSNs may have mesh architecture or layered architecture. cluster 2 is an ad-hoc network of mobile WSNs.

6.8.3 WSN Protocols:-

Network protocols have design goals as:

i) limit computational needs

ii) limit use of battery power and thus bandwidth limits, Operation in self-configured ad-hoc setup mode.

iii) limit memory requirement of the protocol.

Data-link layer Media Access Control (MAC) Protocol:-

Sensor-MAC Protocol can be deployed at the data-link layer. S-MAC nodes go to sleep for prolong periods. They need to synchronise at periodic intervals.

S-MAC Protocol enables use of the energy-efficient, collision-free transmission and intermittently synchronises the operations.

Routing Protocols:-

The network layer uses multi-hop determination, energy-efficient routing, route caching and directed diffusion of data.

Routing protocols are either proactive or reactive. Pro-active Protocols keep route ~~chosen~~ cache and determine the route in advance. Reactive Protocols determine the route on demand. Routing Protocols are table-driven when a routing table guides the path available.

6.8.4 WSN Infrastructure Establishment:-

When a WSN infrastructure establishes the following steps which need considerations:

* Sensors with associated CMOS low power ASIC circuit, their radio ranges and energy-efficient coding.
* Clusters, cluster gateways, cluster-heads and clusters hierarchy.
* Routing, data aggregation, compaction, fusion and direct diffusion.
* Network topology and architecture according to the applications and services, wireless multi-hop infrastructure network architecture or multi-cluster architecture.

→ WSN Various Integration Approaches:-

WSN needs integrated approaches for:

* Nodes design and provisioning of resources.
* Nodes localisation.
* Nodes mobility.
* Sensor connecting architecture.
* Sensor networking architecture.
* Data dissemination protocols.
* Security protocols.
* Data-link layer and routing protocols.

→ Quality of service:-

Quality of service is an average weighted QoS metric over the lifetime of a network. Several metrics are as follows:

i) Average delay: Measure of the time taken in generation of sensor data and its delivery up to the destination.

ii) Life time: Time up to which the WSN functions efficiently or given energy resources of the nodes will last

iii) Throughput: Bytes per second delivered up to the destination. Low throughput means high delays.

iv) Link Quality Indicator (LQI) which means packets delivered/Packet transmitted from the nodes.

→ Configuration:-

Challenges of configuring in view of resource constraints with the nodes statically or dynamically or self-automatic are the following:

i) locations and mobility range of the WSN nodes.

ii) Clusters.

iii) gateways.

iv) Cluster-heads.

v) Sampling rate of the sensed parameters and

vi) Their aggregation compaction and fusion.

6.8.5 WSN Nodes Secure Communication :-

Sensor networks need Secure Communication for data Privacy and integrity. Authentication ensures data from the Sensing node only, maintains data integrity and disables the Communication of messages from unauthenticated Sources. Privacy ensures data Secrecy and eavesdropping.

SPINS is a suite of security protocols for the sensor networks which are:

i) Secure Network Encryption protocol (SNEP)

ii) Micro-Tesla

Distributed Sensor networks use a key management Scheme, Which can be:

i) based on probabilistic key sharing.

ii) random key pre-distribution key sharing.

→ Localised Encryption and Authentication Protocol (LEAP)

Different packets when using LEAP use different keying mechanisms. Security needs decide the mechanism. A node uses four keys: individual key, group key, Cluster key and Share a pair of key with the neighbour.

Network is data-centric network and route needs have no addressability. Three Challenges, viz. security, QoS and Configuring of nodes are as follows:

Security:-

Security challanges are:

→ Hello flood Attack: An attacker node sends hello messages repeatedly, and thus drains the energy of the attacked node.

→ Sybil attack is an attack where a single node, presents itself as different entities at diffuent times.

→ Selective forwarding attack is an attack when the attacker node does not forward the attacked node messages on receiving.

→ Sinkhole attack is when an attacked node behaves as an access point and receives the messages without forwauding them.

6.8.5 WSN IoT Applications:-

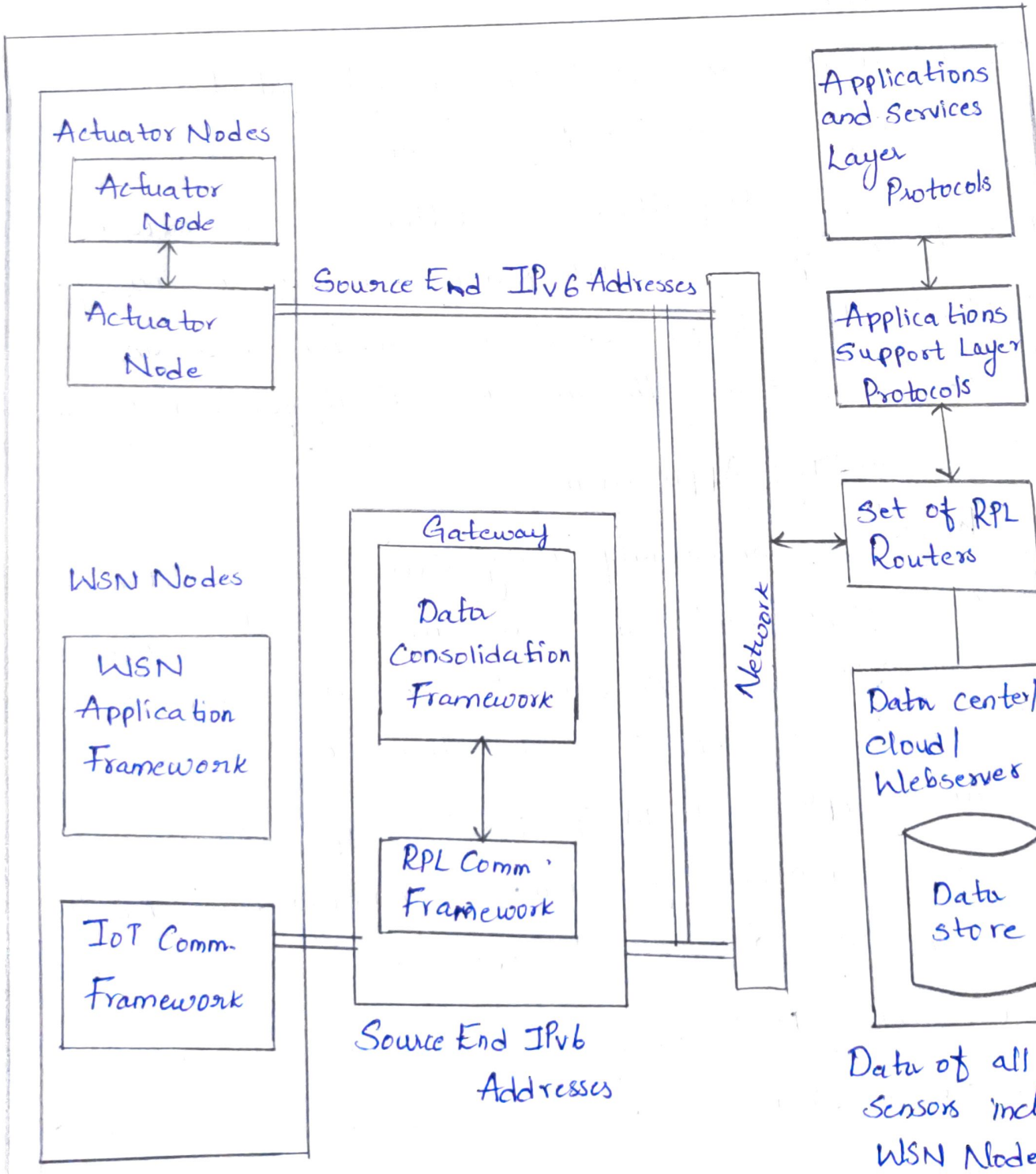
WSNs are increasingly being used as a subsystem in IoT-based applications and services. WSN can function as a source of data along with other systems and connect through a gateway and access point to the Internet.

An example of WSN specific IoT application is smart home control and monitoring system. A connected home has the following applications deployed in smart home:

1) Mobiles, tablets, IP-TV, VOIP telephony, Video conferencing, Video-on-demand, Video-surveillance, WI-FI and Internet.

2) WSN nodes and wireless actuator nodes, which can be built using ZigBee IP are nodes for home security access control and

Security alerts, lighting control, home health care, fire detection, leak detection, energy efficiency, solar panel monitoring and control, temperature monitoring and HVAC control and automated meter reading.



6.9 SENSOR TECHNOLOGY:-

Sensor technology is a technology used for designing Sensors and associated electronic readers, circuits and devices. A Sensor can sense a change in physical Parameters, such as temperature, Pressure, light, metal, smoke and proximity to an object. Sensors can also sense acceleration, Orientation, location, Vibrations or smell, Organic Vapours (or) gases. A microphone senses the voice and changes in the sounds and is used to record voice or music.

6.9.1 Sensing the Real World:-

Sensor is an electronic device in a circuit that senses a physical environment or condition. The sensor sends signals to an electronic circuit, which interconnects to a serial port interface at a microcontroller or controller or computing device.

Example of Resistive, Capacitive, Diode and Transistor-based Sensors:-

A characteristic parameter of a circuit changes with the physical conditions. Technology that facilitates such changes due to sensing is also used in mobile phone. A mobile phone can sense Surrounding conditions. The touchscreen of a mobile phone can senses a finger touch and gestures.

Analog Sensors :-

Analog Sensors Use a Sensor and an associated electronic analog circuit. Analog sensors generate analog outputs as per the physical environmental parameters, such as temperature, strain, Pressure, force, flex, Vapours, magnetic field or proximity.

The Sensor output is given to the ~~output~~ input of a Signal conditioning-cum-amplifying circuit (SC). The SC output is the input to an Analog-to-Digital Converter (ADC).

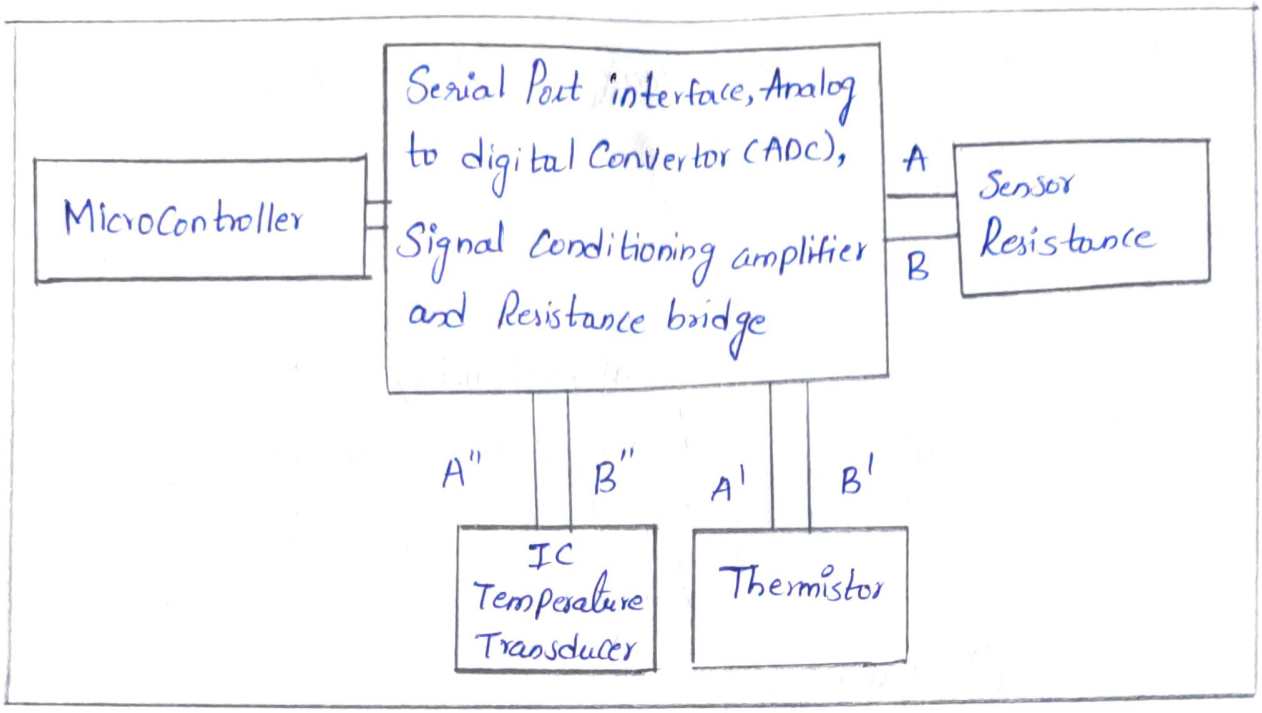
Reading Temperature from Resistance Sensor :-

The measurements can be first made using two standard or reference temperature points, such as 0°c and 100°c. An equation or table can be prepared for the sensing component resistance R values as a function of the temperature T in °C.

When temperature, such as of oil or coolant plate in an M2M or IoT device in an automobile needs to be sensed then a simple electronic device circuits uses the sensing component at the sensed object.

A transducer induces current or voltage. The output changes as per a change in the physical energy at input. An IC-based circuit for a temperature-transducer induces current in the output according to the heat energy, represented by the temperature.

Microcontroller is a computing device which reads the input at its ports, saves the reading in memory and then the reading is used for computations and communication.



Serial Port Interface:-

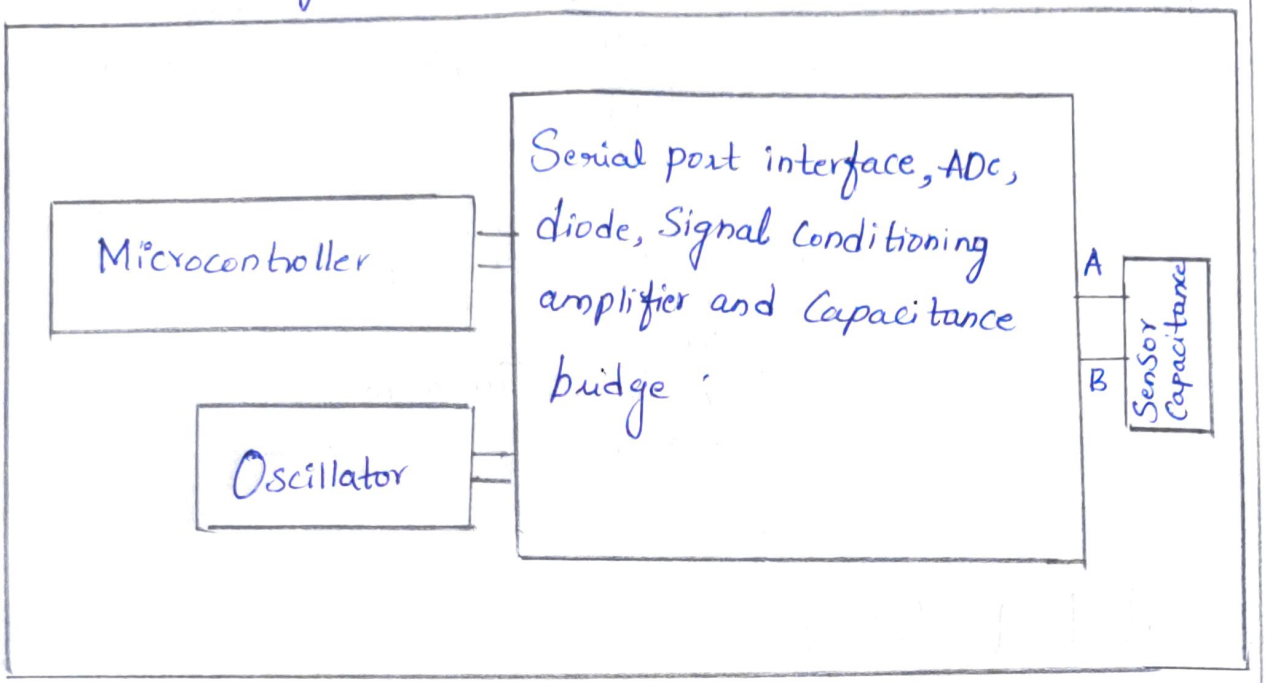
A Serial port interface with the ADC has an advanta-ge that the ADC 8 or 10 or 12 bit output is input to the interface and the inteface sends the input to the Serial Port at the ~~contro~~ microcontroller. Serial port interface has just two terminals in the output.

Reading from Capacitive Sensor:-

C of a sensor object is a part of a capacitive bridge. The associate computing devices calculates the proximity distance in case 1 and touched position in case 2 as well as the Successive Vooriations.

It shows a circuit using a capacitance bridge. The bridge consists of the sensing capacitor and three fixed capacitors. The figure shows a microcontroller based electronic circuit with port connected to four sub-circuits, serial port

interface, ADC, Signal conditioning amplifier, diode and capacitance bridge.



Analog to Digital Converter:-

A microcontroller may consist of an in-circuit ADC or multiple inputs ADC. It processes the digital output from the in-circuit ADC. Alternatively, a port accepts the digital input consisting of 1's and 0's through an external ADC. An 8-bit Port accepts the 8-bit input which corresponds to 0 to 255 decimal.

Sampling ADC:-

Sampling means that an ADC accepts input signals at Specified Periodic intervals and Converts them into digits. The interval is set as per the signal frequency and other needs. The applications of sampling ADC are many.

Signal Conditioning Amplifier:-

An SC amplifies the signal at the input as well as adds or subtracts an offset voltage in such a way that minimum $V_{in(min)}$ and maximum $V_{in(max)}$ values of the sensed physical parameter equal to 0 V and V_{ref}, respectively, at the SC outputs.

Digital Sensors:-

A specific electronic component or circuit gives digital output 1 or 0 (or) output of 1s and 0s as a binary number. A digital sensor uses the sensor and has an associated electronic circuit which gives digital output. The output 1 or 0 is read through a port in a microcontroller.

Sensing of an On-Off State:-

A no. of conditions needs detection using the concept of digital output of on-off state. The gives four cases on how to sense an on-off state, which means binary 1 and 0 output for reading by a circuit or microcontroler.

Sensing a Set of On-Off states:-

A no. of conditions together need detection in many applications. A circuit generates digital output for a set of On-Off states. A specific electronic component or circuit gives digital output such as, a set of 4 or 8 or 16 states consisting of 1s and 0s for sensing a set of discrete changes in a specific set of physical state or conditions.

6.9.2 Examples of Sensors :-

* Temperature :-

A Component called thermistor, shows larger changes in resistance within narrow environment temperature range. An NTC thermistor shows negative temperature coefficient which means a drop in the resistance value with rise in temperature.

A temperature Sensor is called PTC, When its exhibits a Positive Temperature Coefficient.

* Humidity :-

Humidity is measured in percentage. It is the relative Percentage ratio (RH%) of content of water vapours in air compared to one in a situation of maximum possible water vapour content for the air temperature at the instance of measurement.

* Distance :-

Infrared Sensor is useful for a 0.15m to 0.8m range of object. IR sensor works on the principle that when a narrow beam IRLED, sends radiation at an include angle, the nearby phototransistor FPT receives the reflected radiation after travelling two times the object distance.

* Light :-

That a photoconductor can be used to detect light in the vicinity. The sensor shows a drop in resistance with surrounding light. Alternatively, the p-n junction photodiode

or phototransistor can be used to measure incoming radiation intensity incoming from a particular direction.

* Angular Acceleration and Change in Direction :-

Gyroscope is a sensor which measures the change in angular velocity and the change in direction. An application takes measurement using a gyroscope or accelerometer and the system initiates actions as programmed.

* Sound :-

A microphone is used to sense sound. A readily available electronic board with a microphone connects to the micro-controller, which can control an actuator for actions based on the sensed sound, or recognise the voice and then take required action, such as dialing a number using the actuator circuit or switching on the car.