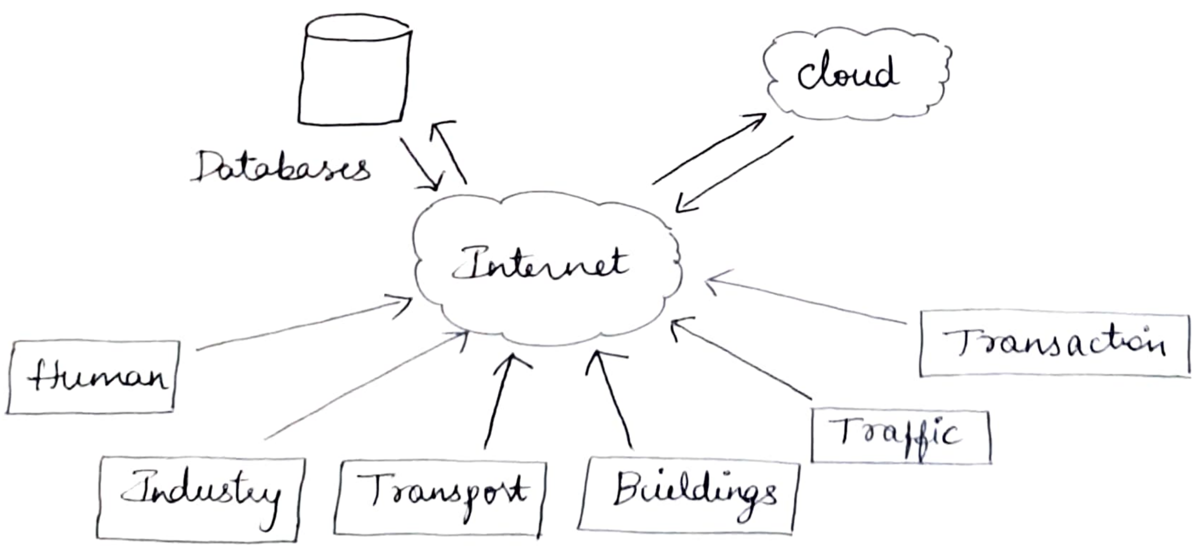
Module - 3

IOT Processing Topologies and Types

Data Format

A huge quantity and variety of data is generated regularly in the Internet. The massive volume of data generated by this huge number of users is further enhanced by the multiple devices utilized by most users. Apart from data generating sources, non human data generator sources such as sensor nodes and automated monitoring systems also add to the data load on the Internet. The variety of data such as e-mails, text documents, social media posts, videos, audio files and images.



Data generating & storage sources connected to Internet

These data can be broadly grouped into two types based on how they can be accessed and stored:

1. Structured data
2. Unstructured data.

1. Structured data:

* Structured data are typically text data that have a predefined structure.
* The data is constrained by a fixed schema and has a proper format.
* These are primarily created by using length-limited data fields such as phone numbers, social security numbers and other such information.
* Languages such as Structured Query Language (SQL) are used for accessing these data.
* Structured data is usually stored in well-defined schemas such as Databases.
* It is generally tabulated with rows and columns.
* Sources are SQL Databases, spreadsheets

such as excel, online forms, sensors such as GPS or RFID tags, Medical devices.

* operations such as updating and deleting is easy due to well structured form of data.

Unstructured data

* Unstructured data have no-predefined structure and can vary according to applications and data-generating sources.

* Data cannot be stored in the form of rows and columns as in databases.

* Data doesnot follow any rules or semantics.

* Examples of human-generated unstructured data includes text, e-mails, videos, images, phone recordings, chats.

* Examples of machine-generated unstructered data include sensor data from traffic, buildings, industries, satellite imagery, surveillance videos.

* This data type does not have fixed formats associated with it, which makes it very difficult for querying algorithms to perform a look up.

* Querying languages such as NOSQL are generally used for this data type.

Sources of unstructured data includes web pages, images (JPEG, GIF, PNG), videos, memos, reports, word documents and powerpoint presentations and surveys.

Importance of Processing in IOT

The need for processing the data has become more crucial with the rapid advancement in IOT.

There is a need for an intelligent and resourceful processing technique.

The data to be processed are classified into three types based on urgency of processing.

1. Very time critical
2. Time critical
3. Normal.

1. Very time critical :

* The Data from sources such as flight control systems, healthcare are considered as very time critical.
* They need immediate decision support.
* These data have a very low threshold of processing latency typically in the range of a few milliseconds.

* The processing requirements of data from very time-critical sources are exceptionally high.

2. Time critical :

* Data from sources that can tolerate normal processing latency.
* These data generally associated with sources such as Vehicles, traffic, machine systems, smart home systems, surveillance systems.
* Those can tolerate a latency of a few seconds.
* The processing requirements allow for the transmission of data to be processed to remote locations / processors such as clouds or through Collaborative processing.

3. Normal :

* Data which can tolerate a processing latency of a few minutes to a few hours.
* Associated with less data - sensitive domains such as Agriculture, Environmental monitoring.
* They have no particular time requirements for processing urgently and can be pursued leisurely.

Processing topologies :

The identification and intelligent selection of processing requirement of an IOT application is one of the crucial steps in deciding the architecture of the deployment.

A properly designed IOT architecture would result in massive savings in network bandwidth and conserve significant amounts of overall energy in the architecture.

There are two processing topologies :

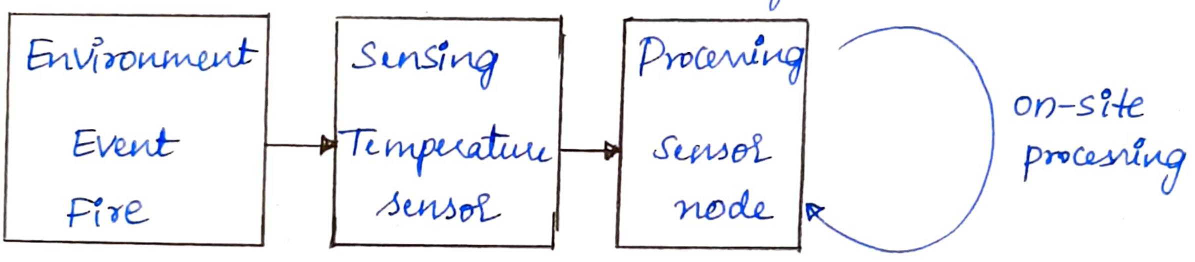
1. on-site topology
2. off-site topology.

1. On-site topology :

* The data is processed at the source itself.
* The processing infrastructure should be fast and robust enough to handle such data.
* This is crucial in applications that have a very low tolerance for latencies.
* Applications such as those associated with healthcare and flight control systems (real time systems).

For example :

* An event (fire) is detected using a temperature (node) sensor connected to a sensor node.
* This sensor node processes the information from the sensed event and generates an alert.
* The node additionally has the option of forwarding the data to a remote infrastructure for further analysis or storage.



Event detection using an on-site processing

Off-site Processing

* The sensor nodes are not required to process data on an urgent basis.
* The sensor nodes are responsible for the collection and framing of data i.e. eventually to be transmitted to another location for processing.
* The data from these sensor nodes (data generating sources) is transmitted either to a remote location (server or cloud) or to multiple

processing nodes.

* The off-site processing paradigm allow for latencies.
* Multiple nodes can come together to share their processing power in order to collaboratively process the data.

Types of off-site processing topologies are :

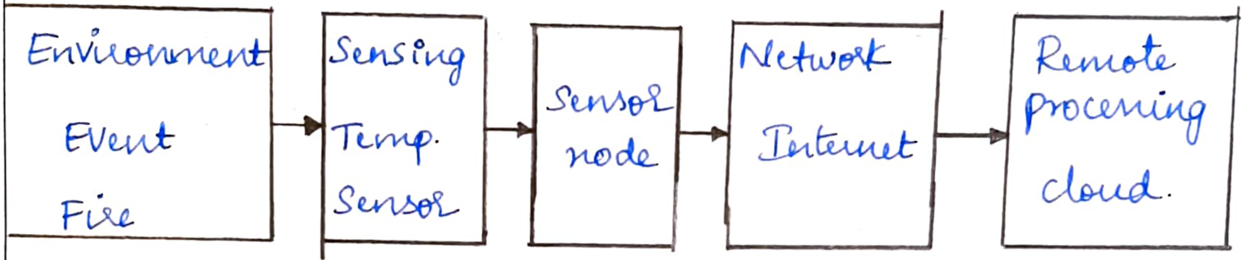
* Remote Processing
* Collaborative Processing.

Remote off-site Processing :

* This is one of the most common processing topologies prevalent in present-day IOT solutions.
* It encompasses sensing of data by various sensor nodes ; the data is then forwarded to a remote server or a cloud-based infrastructure for further processing and analytics.
* The processing of data from hundreds and thousands of sensor nodes can be simultaneously offloaded to a single, powerful computing platform.
* This results in massive cost and energy savings by enabling the reuse and reallocation of the same processing resource.

* The sensing of an event is performed locally, and the decision making is outsourced to a remote processor (Cloud)

* This paradigm uses lot of network bandwidth and relies on network Connectivity between the sensor nodes and the remote processing infrastructure.



Event detection using an off-site Remote processing

Collaborative Processing :

* This Processing topology typically finds use in scenarios with limited or no network Connectivity.

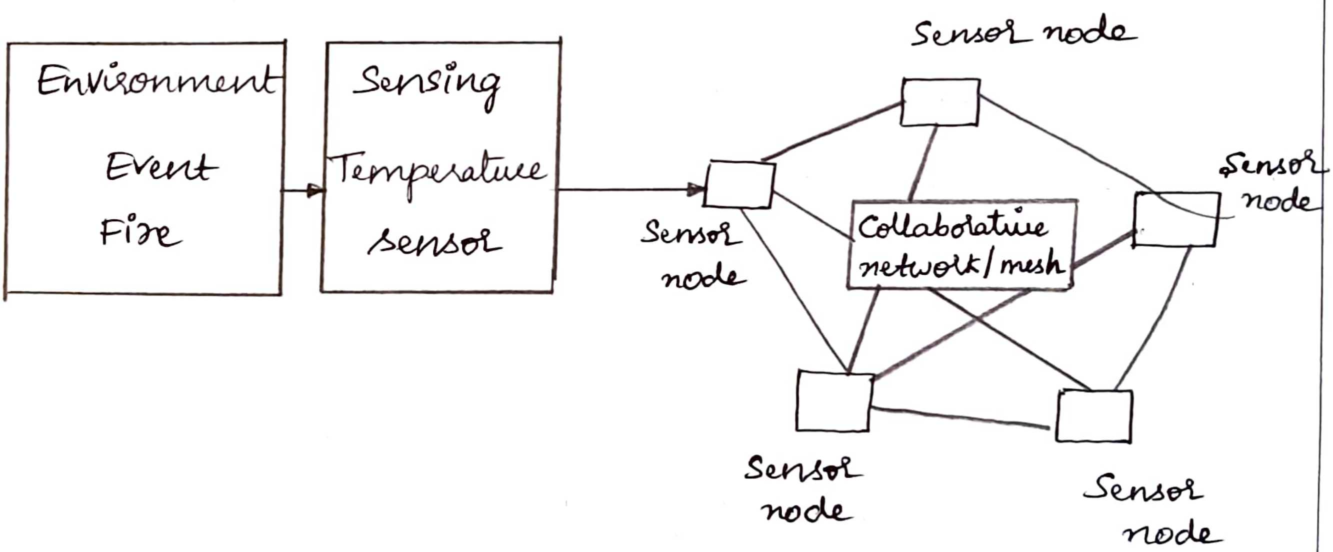
* This topology can be quite economical for large-scale deployments spread over vast areas.

* The simplest solution is to club together the processing power of nearby processing nodes and collaboratively process the data in the vicinity of the data source itself.

* This approach also reduces latencies due to transfer of data over the network. It conserves bandwidth of the network, especially ones connecting to the Internet.

* This topology can be beneficial for applications such as Agriculture as high frequency of data processing is not required as agricultural data is generally logged after significantly long intervals (in the range of hours)

* This topology makes use of mesh networks for easy implementation.



Event detection using a collaborative processing topology.

Considerations for IOT Device design

* The main considerations in an IOT solution is the selection of the processor for developing the sensing solution (sensor node)

* The important considerations are as follows:

1. Size
2. Energy
3. Cost
4. Memory
5. Processing power
6. I/O rating
7. Add-ons.

1. Size :

* Size is the crucial factor for deciding the form factor and the energy consumption of a sensor node.

* larger the form factor, larger is the energy consumption of the hardware.

* large form factors are not suitable for a significant bulk of IOT applications. nowadays sensors are made with MEMS technology. They can be extremely small.

2. Energy :

* The energy requirements of a processor is the most important deciding factor in designing IOT based sensing solutions.

* Higher the energy requirements, higher is the energy source (battery) replacement frequency.

* This principle automatically lowers the long-term sustainability of sensing hardware, especially for IOT-based applications.

3. Cost :

* The cost of the processor is also the driving force in deciding the density of deployment of sensor nodes for IOT based solutions.

* If it is cheaper, hardware enables a much higher density of hardware deployment by users.

* For example :
Cheaper gas and fire detection solutions would enable users to include much more sensing hardware for a lesser cost.

4. Memory :

* The memory requirements (both Volatile and non-Volatile memory) of IOT devices determines the capabilities of the device.

* Volatile is the fast and temporary memory. example : RAM and Cache memory.

* Non Volatile is the permanent memory. example : ROM.

* Features such as local data processing, Data storage, data filtering, data formatting rely on the memory capabilities of devices.

* Devices with higher memory are expensive.

5. Processing Power :

* The Processing Power decides :

* what type of sensors can be accommodated with the IOT node.

* what processing features can integrate on-site with the IOT device.

* The type of applications the device can be associated with.

* Applications that handle video and image data require IOT devices with higher processing power.

I/O Rating :

* The I/O rating determines the circuit complexity, energy usage and requirements of sensor types.
* Newer processors have a meager I/O voltage rating of 3.3V as compared to 5V.
* This translates to requiring additional voltage and logic conversion circuitry to interface legacy technologies and sensors with the newer processors, this adds to complexity and cost.

Add-ons :

* An IOT device need add-ons such as analog to digital conversion (ADC) units, in-built clock circuits, connections to USB and ethernet, in-built wireless access capabilities.

Processing offloading :

* The processing offloading is important for the development of densely deployable, energy-conserving, miniaturized and IOT-based solutions for sensing tasks.

* A typical outline of an IOT deployment involves various layers of processing from the primary layer of sensing to cloud based infrastructure.

* The primary layer of sensing can have multiple sensing types. the sensors are integrated with a processor using wired or wireless connections.

* The data processing is done at various levels

1. Local processing
2. Edge processing
3. Fog processing
4. cloud processing.

1. Local Processing :

The event that require immediate processing of the sensed data, an on-site topology is followed. The sensor nodes may be connected through wired or wireless connection.

* The data from the sensing layer can be forwarded to fog or cloud or can be within the edge layer.

Edge processing :

* The edge layer makes use of devices within the local network to process data.

* The devices within the local network (till the fog) communicate using short range wireless connections.

* The data which needs to be sent to the cloud uses long range wireless connection which enables access to a backbone network.

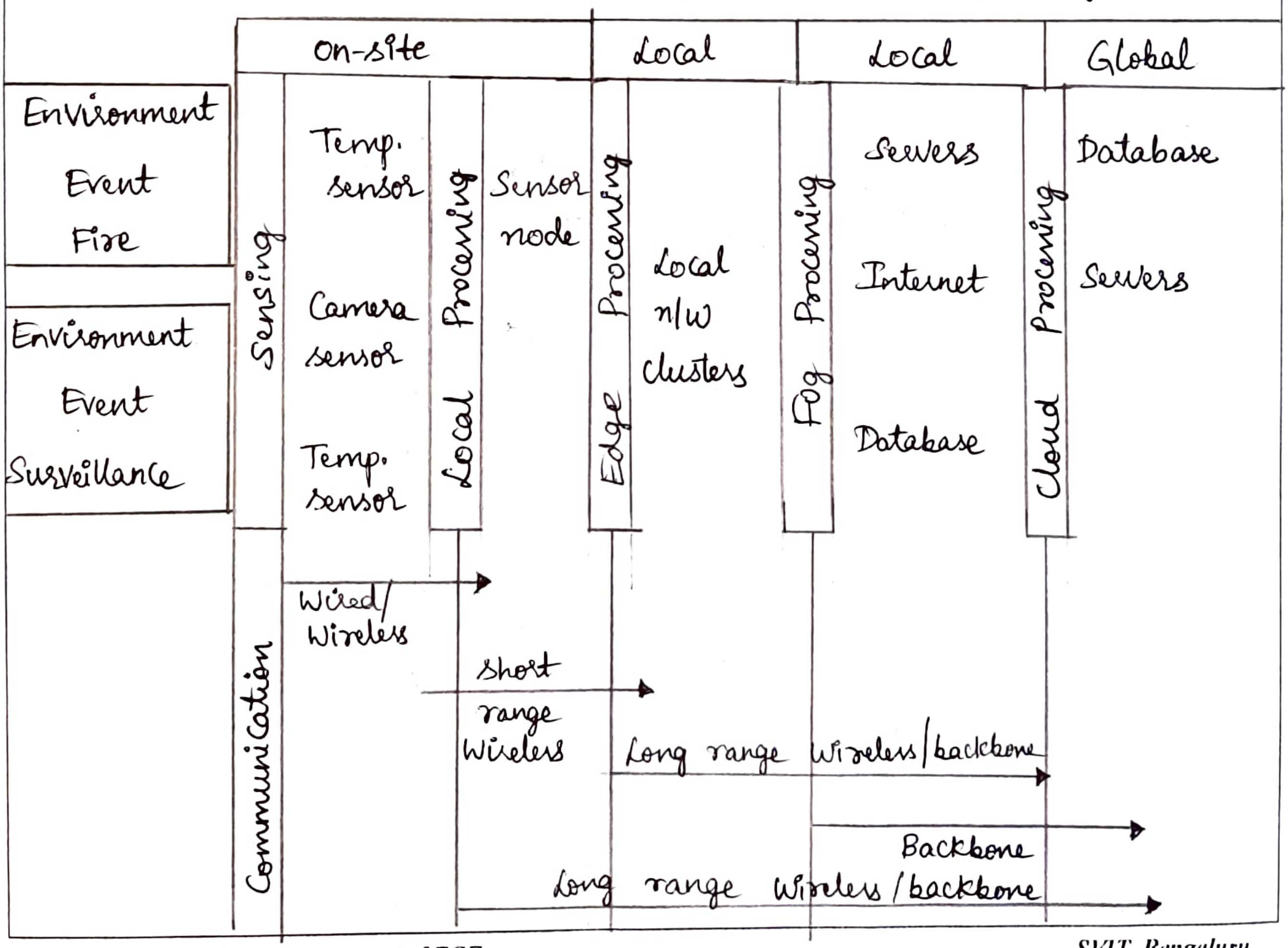
Fog based Processing :

* The fog nodes are localized within a geographic area and serve the IOT nodes within a much smaller coverage area as compared to cloud.

* Fog nodes which are at the level of gateways are accessed by the IOT devices through the internet.

Cloud based Processing :

* The devices to be connected to the Internet through long range wireless networks.
* These devices are connected to a backbone network.
* This approach is expensive and concerns network bandwidth, latency and complexity.



The data offloading is divided into three parts:
1. offload location
2. offload decision making
3. offloading considerations.

1. offload location :

The offload location is distinguished into 4 types:
* Edge
* Fog
* Remote server
* Cloud.

The choice of offload location decides the applicability, cost and sustainability of the IOT application and deployment.

1. Edge : offloading processing to the edge implies that the data processing is facilitated to a location at or near the source of data generation itself. Offloading to the edge is done to achieve aggregation, manipulation, bandwidth reduction and other data operations directly on an IOT device.

2. Fog : Fog computing is utilized to conserve network bandwidth, reduce latencies, restrict the

amount of data unnecessarily flowing through the internet and enable rapid mobility support for IOT devices. The data, computing, storage and applications are shifted to a place between the data source and the cloud.

3. Remote server :

A simple remote server with good processing power may be used with IOT-based applications to offload the processing from resource constrained IOT devices. Rapid scalability may be an issue with remote servers, and may be expensive and hard to maintain in comparison to solutions such as the cloud.

4. Cloud :

Cloud computing is a configurable computer system, which can get access to configurable resources, platforms and high-level services through a shared pool hosted remotely. A cloud can be accessed globally. Cloud enables massive scalability of solutions as they can enable resource enhancement allocated to a user or solution in an on-demand manner.

Offload decision making :

The offload decision making deals with the choice of where to offload and how much to offload in the deployment of an off-site processing topology.

This is based on data generation rate, network bandwidth, the criticality of applications, processing resource available at the offload site.

There are two approaches :

1. Naive approach :

* This is a rule based approach in which the data from IOT devices are offloaded to the nearest location based on the criteria.
* This approach is easy to implement.
* This approach is not recommended for dense deployments or deployments where the data generation rate is high or the data being offloaded is complex (multimedia or hybrid data types)

2. Bargaining based approach :

* This is a processing intensive during the decision making stages.
* It enables alleviation of network traffic Congestion.

* It enhances service QOS (quality of service) parameters such as bandwidth and latencies.

* For optimal solution of QOS, all the parameters need not have to be treated with equal importance.

* This method tries to maximize the QOS by reducing the qualities of certain parameters while the others are enhanced.

* This approach does not depend on historical data for decision making purposes.

for example : Game theory.

3. Learning based approach :

* The learning based approaches generally rely on past behaviour and trends of data flow through the IOT architecture.

* The optimisation of QOS parameters is pursued by learning from historical trends.

* It optimizes previous solutions further and enhances the collective behaviour of the IOT implementation.

* The memory requirements and processing requirements are high during the decision making stages.

The most common example of a learning based approach is machine learning.

Offloading Considerations :

There are few offloading parameters that need to be considered while deciding to choose the offloading type.

They are :

1. Bandwidth
2. Latency
3. Criticality
4. Resources.

1. Bandwidth :

* The maximum amount of data that can be simultaneously transmitted over the network between two points is the bandwidth of that network. the bandwidth of a wired or wireless network is also considered to be its data-carrying capacity and often used to describe the data rate of that network.

2. Latency :

• It is the time delay incurred between the start and completion of an operation. In the present context, latency can be due to the network (network latency) or the processor (processing latency)

* latency arises due to the physical limitations of the infrastructure, which is associated with an operation.

* the operation can be data transfer over a network or processing of a data at a processor.

3. Criticality :

* It defines the importance of a task being pursued by an IOT application.

* The more critical a task is, the lesser latency is expected from the IOT solution.

* For example :

Detection of fires using an IOT solution has higher criticality than detector of agricultural field parameters.

* The former requires a response time in the tune of milliseconds, whereas the later can be addressed within hours or even days.

4. Resources :

* It signifies the actual capabilities of an offload location.

* These capabilities may be the processing power, the suite of analytical algorithms and others.

For example :

Real time multimedia processing which are highly energy intensive, can process and analyse huge volumes of data in a short duration.

The scalar data which can be addressed using nominal resources doesn't waste much energy.