MODULE 2 - IOE

Physical Design of IoT Devices

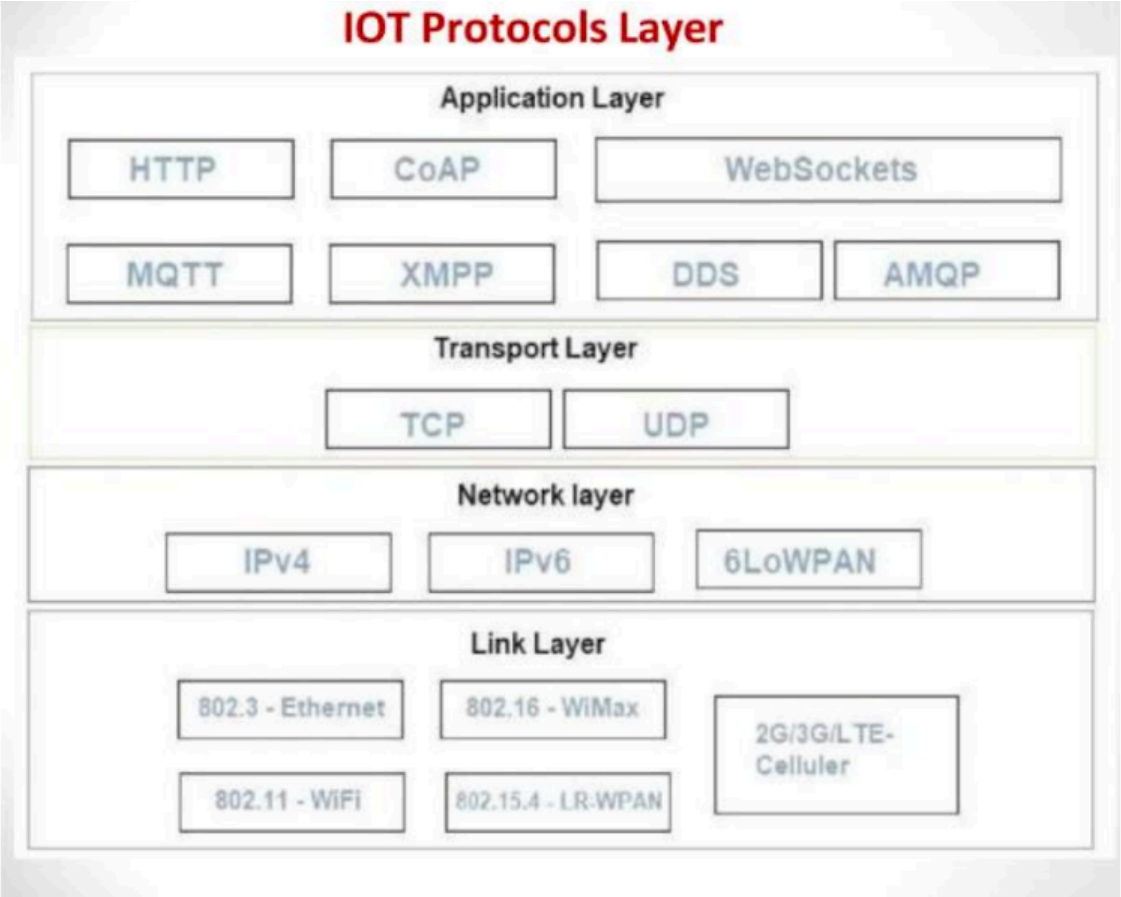
IoT (Internet of Things) refers to a **network of interconnected physical devices** embedded with **sensors, software, and connectivity** to collect and exchange data over the internet using wired or wireless connections.

Key Elements of Physical Design

1. **Sensor I/O Interfaces**
   - Used by IoT devices to connect with other devices and gather environmental data.
   - Sensors measure parameters like temperature, humidity, motion, light, etc.
   - Actuators perform actions like turning a motor, opening a valve, or adjusting brightness.

2. **Connectivity Interfaces for the Internet**
   - Provide communication between IoT devices and the internet or other networks.
   - Can include **Ethernet ports, Wi-Fi modules, cellular modems, Zigbee, Bluetooth**.
3. **Data Storage and Memory**
   - Temporary or permanent storage for collected data before processing or transmission.
   - Can be local (e.g., SD card, onboard flash memory) or cloud-based.
4. **Interfaces for Audio and Video**
   - For multimedia IoT devices such as smart cameras, voice assistants, or video doorbells.
   - Includes microphones, speakers, and camera modules.

IoT Protocols (part of the physical design answer)



1) Link Layer

- Manages **physical transmission** of data between devices.
- Defines how packets are **coded, signaled, and sent** via medium: coaxial, copper wires, radio waves.

Protocols

1. **802.3 – Ethernet (Wired)**
   - supports twisted pair, coaxial, and fiber optics.

2. **802.11 – Wi-Fi (Wireless LAN)**
   - Frequency bands : operates in 2.4 GHz & 5 GHz bands

3. **802.16 – WiMAX (Wireless Broadband)**
   - Data rates: **1.5 Mb/s** to **1 Gb/s**.

4. **802.15.4 – LR-WPAN (Low-Rate Wireless PAN)**
   - Data rates: **40 kb/s** to **250 kb/s**.

5. **2G / 3G / 4G – Mobile Communication**
   - Data rates: **9.6 kb/s (2G)** to **100 Mb/s (4G)**.

2) Network Layer

- Handles **addressing and routing** of data packets from source device to destination device
- Uses **IP addresses** to identify devices.

Protocols:

- **IPv4:** 32-bit addresses. Provides about $\mathbf{2^{32}}$ **addresses**.
- **IPv6:** 128-bit addresses (huge address space). Provides about $\mathbf{2^{128}}$ **addresses** (huge space).
- **6LoWPAN:** IPv6 over low-power WPAN (2.4 GHz, 250 kb/s).

3) Transport Layer

- Ensures end-to-end communication between IoT devices and applications.
- Provides reliable or fast delivery depending on protocol used.
- Functions: segmentation (breaking data into packets), error control, flow/congestion control.

Protocols:

- **TCP (Transmission Control Protocol):** Reliable, connection-oriented, ordered delivery. Used with HTTP, SMTP, FTP. Used in applications where accuracy is important (e.g., web, email).
- **UDP (User Datagram Protocol):** no guarantee, Connectionless, fast; used in time-sensitive apps.

4) Application Layer

- Defines how applications **interface with lower layers**.

Protocols: (Explain 3 / 4. Write full form of all.)

1. **HTTP (Hypertext Transfer Protocol)**
   - Used for web communication between browser and server.
   - **Request–response** model, **stateless protocol**.

2. **CoAP (Constrained Application Protocol)**
   - Designed for **M2M** communication.
   - Works on **low-power, constrained devices** (e.g., sensors).
   - **Lightweight** alternative to HTTP.

3. **WebSocket**
   - Provides **full-duplex (two-way) communication** over a single connection.

4. **MQTT (Message Queue Telemetry Transport)**
   - **Lightweight**, **publish–subscribe** model.
   - Uses **client-server architecture**

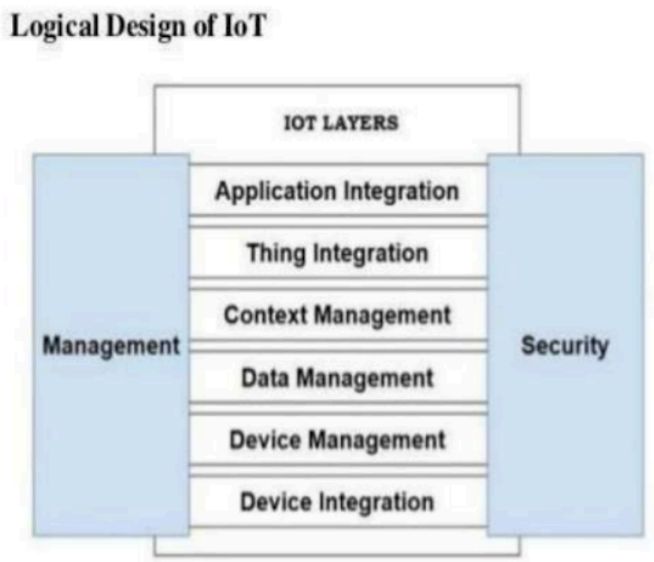
5. **DDS (Data Distribution Service)**
   - Uses **publish–subscribe model**.

6. **AMQP (Advanced Message Queuing Protocol)**
   - Supports **publish–subscribe** models.
   - Supports **point-to-point** models.

-

7. **XMPP (Extensible Messaging and Presence Protocol)**
    - Supports both **client–server** and **server–server** communication.

Logical Design of IoT Devices



Logical Design of IoT

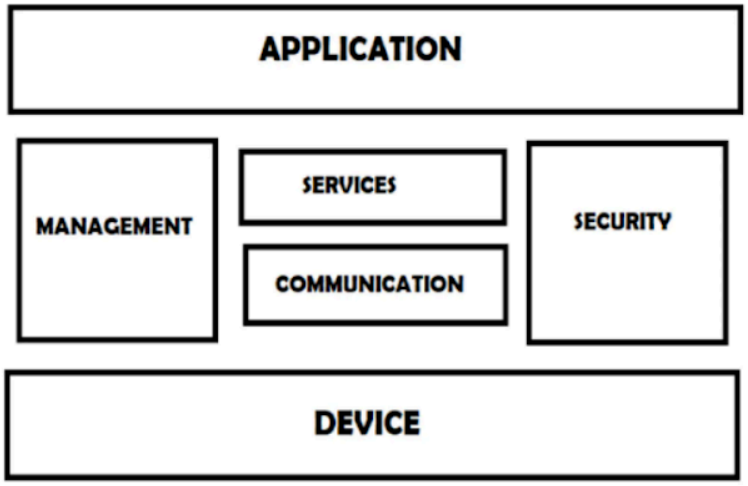
Logical design means the abstract view of an IoT system, showing entities and processes without focusing on technical implementation. It mainly covers:

- **IoT Functional Blocks**
- **IoT Communication Models**
- **IoT Communication APIs**

A] IoT Functional Blocks

An IoT system has several blocks : **(acronym - SMS CAD) (tip - monitor and control; these 2 words used too many times)**

- **Device**: Sensors and actuators used for monitoring, control, and data collection.
- **Communication**: Enables data exchange between devices.
- **Services**: Provide device monitoring and control service, data publishing services, etc.
- **Management**: This block provides various functions to govern the IoT system.
- **Security**: Ensures authentication, authorization, data integrity, and protection.
- **Application**: User interface to monitor and control devices.



B] IoT Communication Models

- Request-Response Model
- Publish-Subscribe Model
- Push-Pull Model
- Exclusive Pair Model

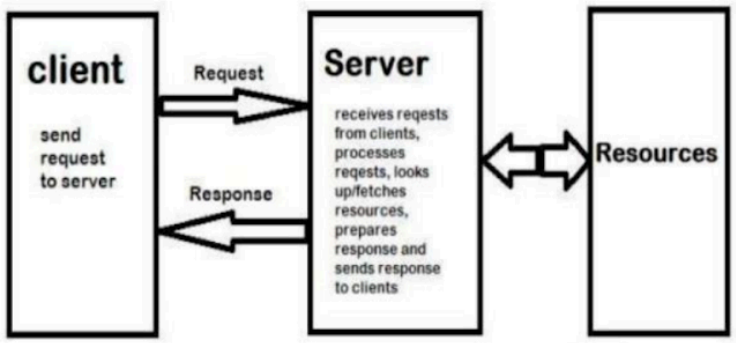
(all 4 models can be very easily explained in own words dont learn)

A] Request-Response Communication Model

The request-response model is a way of communication where a **client** sends a request to a **server**, and the server replies with a response.

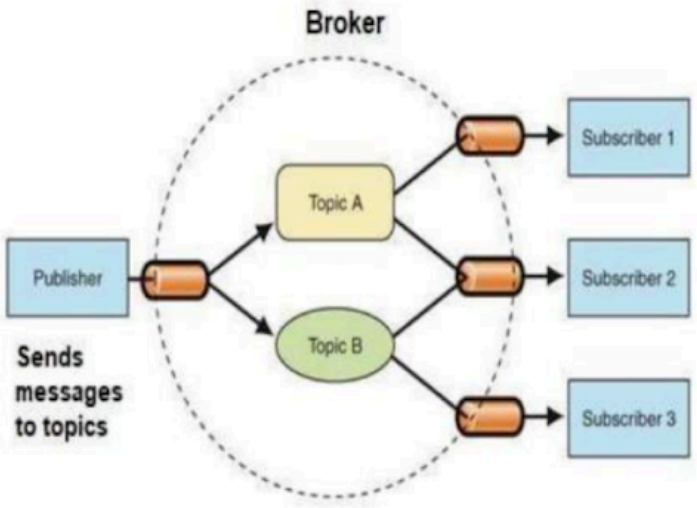
- The server, on receiving a request, processes it, fetches the required data/resources, prepares the result, and sends it back to the client.
- This model is **stateless**, meaning each request-response happens independently.

- **HTTP** is the best example: When you open a website, your browser requests the page, and the server responds with the webpage data.



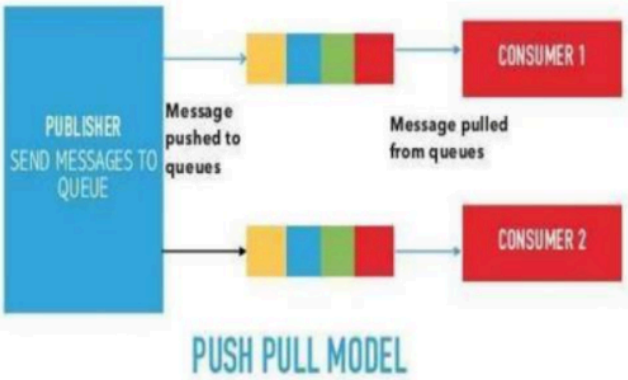
Request-Response Communication Model

(dont write server ke andar ka)



B] Publish-Subscribe Communication Model

- In this model, there are **publishers**, a **broker**, and **subscribers**.
- **Publishers** send data to topics managed by the broker.
- **Subscribers** register for topics of interest.
- The **broker** delivers data from publishers to all subscribers of that topic.
- Publishers and subscribers don't need to know each other.
- **Example:** A news app sends sports updates to all users subscribed to the "Sports" topic.



C] Push-Pull Communication Model

- **Producers** push data into queues, and **consumers** pull data from them.
- Producers and consumers work independently, without knowing each other.
- Queues act as a **buffer**, handling speed differences between producers and consumers.

D] Exclusive Pair Communication Model

•Exclusive Pair is a bidirectional, fully duplex communication model that uses a persistent connection between the client and server. Connection is setup it remains open until the client sends a request to close the connection.

•Client and server can send messages to each other after connection setup. Exclusive pair is stateful communication model and the server is aware of all the open connections.

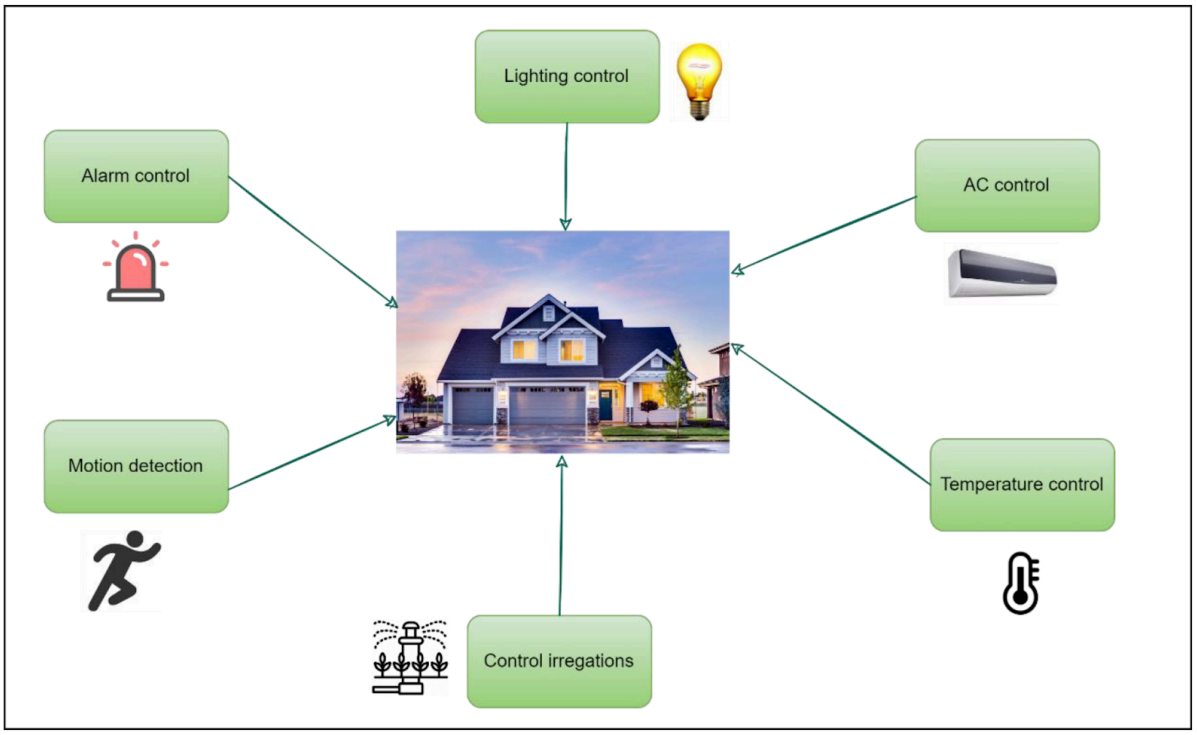
IoT Levels

IoT Level 1

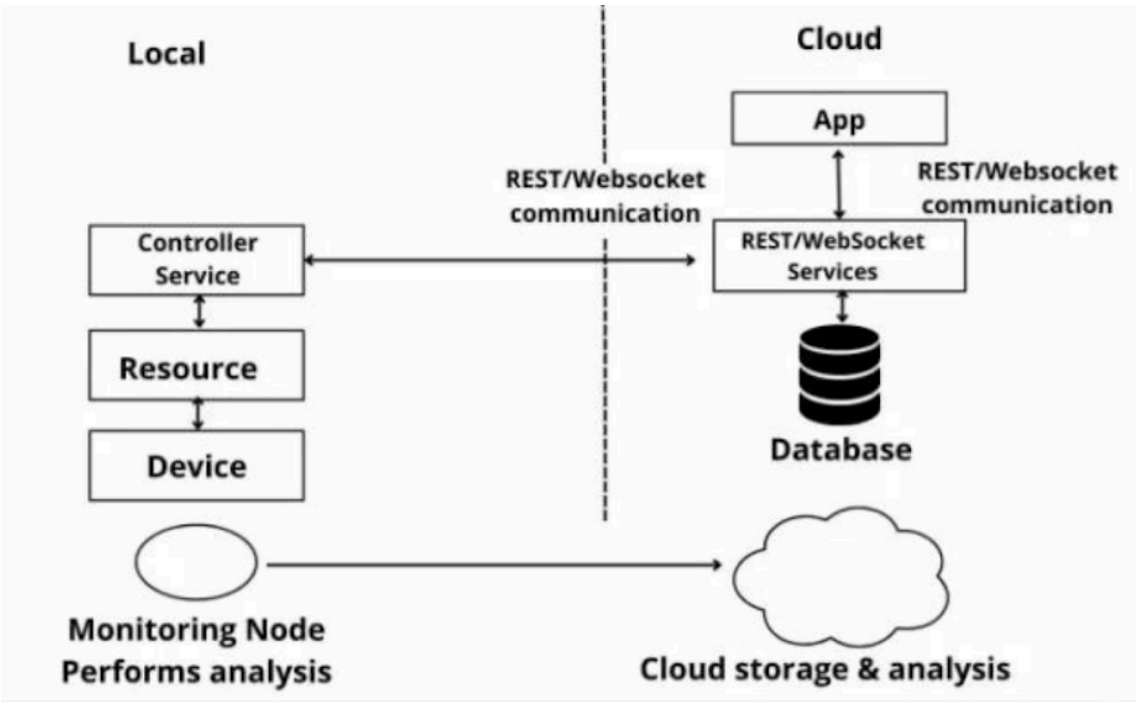
- IoT systems have a **single device** that does sensing/actuation, stores and analyzes data, and runs the application.
- Suitable for **low complexity, low-cost solutions**.

Example: Home Automation System

- Devices like sensors (temperature, smoke detector, motion sensor) collect data.
- **Data** is **stored and analyzed "locally".**
- Based on analysis, control actions are triggered (e.g., turning on AC, switching lights) via a mobile app, or simply used for monitoring (home cameras).



IoT Level 2



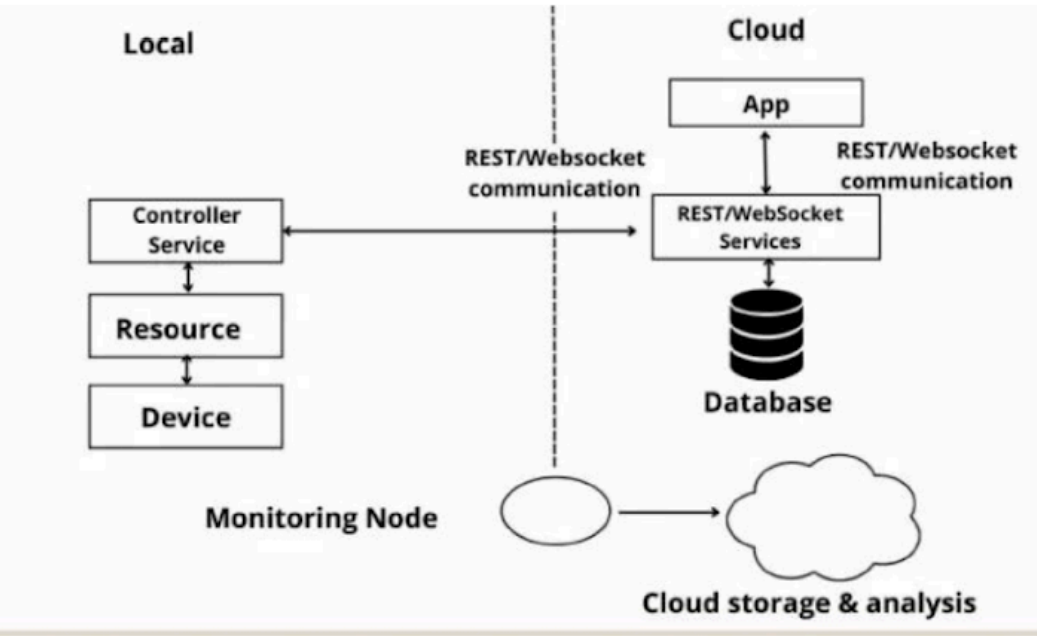
- More **complex** than Level 1, with **faster sensing**.
- Generates **large data**, so **cloud storage** is used.

- **Data analysis** is done **locally**; **Data storage** is done on **cloud**.
- Based on the analysis, actions are triggered using a **web app or mobile app**.

Examples:

- Agriculture applications (e.g., soil or crop monitoring).
- Room freshening systems using odour sensors.

IoT Level 3 (same as level 2 only 3rd point different and 1 small change in diagram)



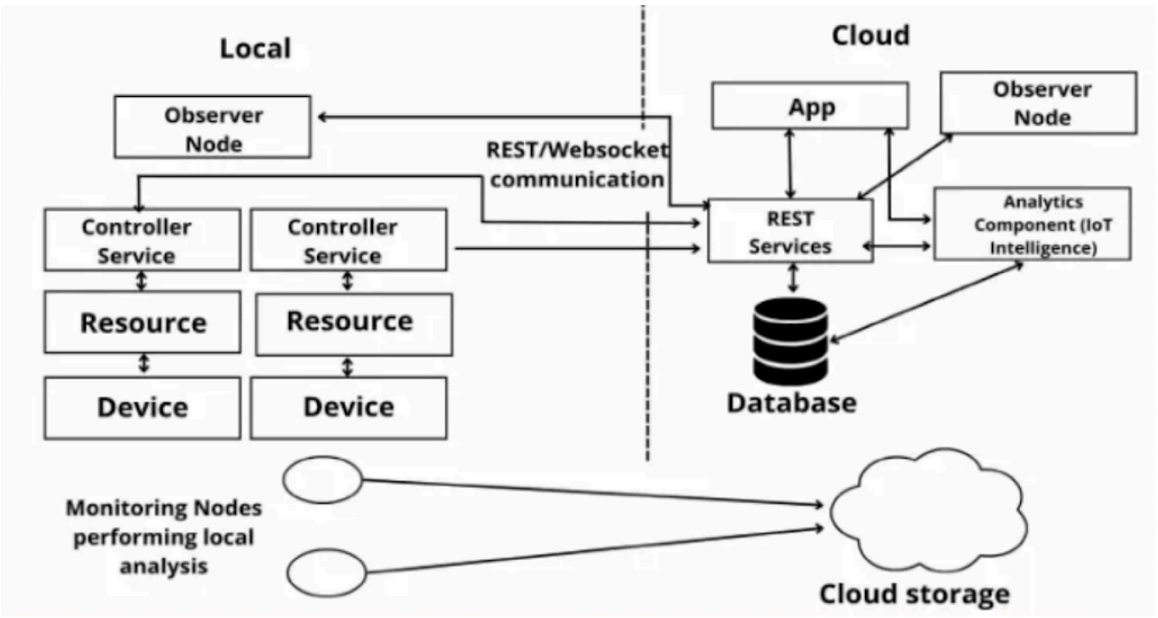
- **Fast sensing**.
- Generates **large data**, so **cloud storage** is used.
- **Data analysis and storage** are both done on the **cloud side.**
- Based on the analysis, actions are triggered using a **web app or mobile app**.

Examples : Tracking package handling

IoT Level 4 (theory same as level 2/3 only 1st different)

- **Multiple sensors (**independent of each other**), data collection, analysis and control & monitoring app.**
- Generates **large data**, so **cloud storage** is used.
- **Data analysis and storage** are both done on the **cloud side.**
- Based on the analysis, actions are triggered using a **web app or mobile app**.

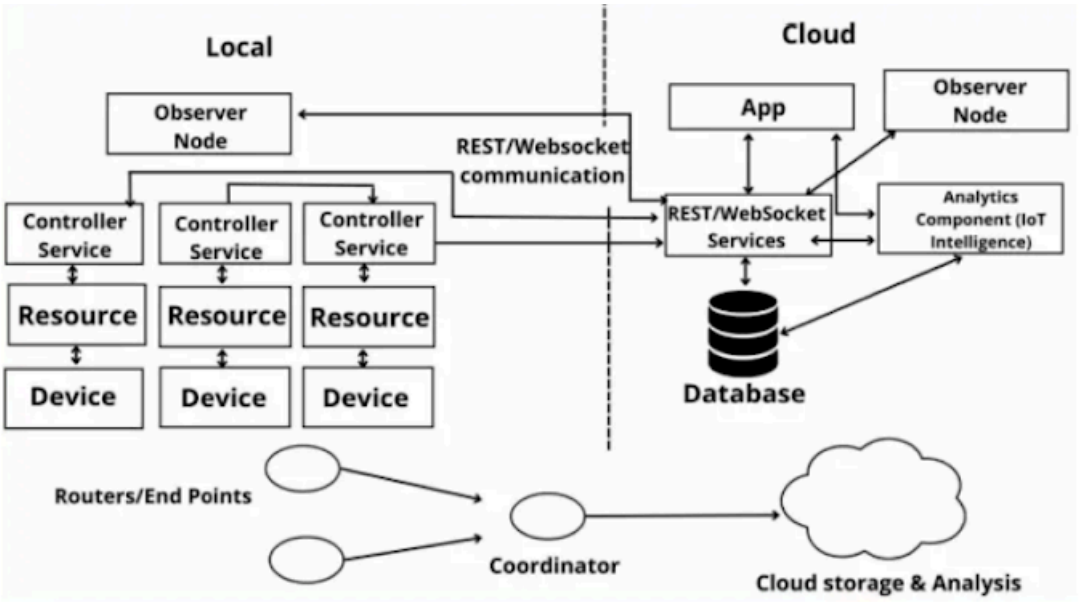
Examples : Noise Monitoring -> Sound Sensors are used



IoT Level 5 **(theory same as level 4 only 1st point different -> one word added and only one small change in diagram)**

- **Multiple sensors (**independent of each other**), coordinator node, data collection, analysis and control & monitoring app.**
- Generates **large data**, so **cloud storage** is used.
- **Data analysis and storage** are both done on the **cloud side.**
- Based on the analysis, actions are triggered using a **web app or mobile app**.

Examples : Forest Fire Detection -> Sensors measure the temperature, smoke, weather, slope of the earth, wind speed, speed of fire spread, flame length to detect fire in early stages.



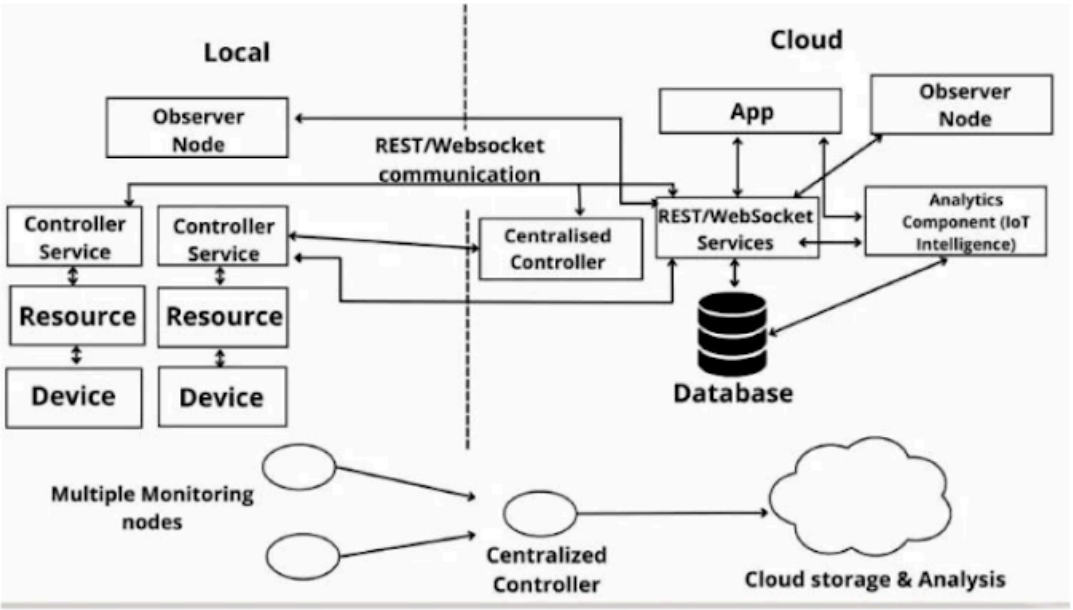
IoT Level 6

- It has **several separate end nodes** used for monitoring and/or transferring data to the cloud.
- Generates **large data**, so **cloud storage** is used.
- **Data analysis and storage** are both done on the **cloud side.**

- The **centralized controller** knows all nodes' status and delivers command signals to the nodes.

Example:

- **Weather surveillance system** with many sensors (temperature, humidity, pressure, etc.).



Developing an IoT Level Template System

Main Components

- **Device**: Sensors or actuators used for sensing or identification.
- **Resources**: Used for accessing and processing data.
- **Controller Service**: Runs on the device and interacts with web services.
- **Database**: Stores data generated by devices.
- **Web Service**: Connects IoT devices, applications, databases, and analysis tools.
- **Analysis Component**: Processes data and generates easy-to-understand results.
- **Application**: User interface to view system status and analyzed data.

IOT CONNECTIVITY TECHNOLOGIES

IoT devices rely on various connectivity technologies to communicate data. Choosing the right technology depends on several factors:

1. Area Coverage

- **Definition:** The geographic range that a connectivity technology can reliably cover.
- **Examples:**
  - **Short-range** (up to 100 meters): Bluetooth, Wi-Fi **-> Smart Homes**
  - **Medium-range** (hundreds of meters to a few km): LoRaWAN
  - **Long-range** (tens of km): Cellular (3G, 4G, 5G), Satellite **-> Smart Cities**

2. Battery Usage

- **Definition:** The battery life and power consumption characteristics of the connectivity method.
- **Examples:**
  - **Low-power:** LoRaWAN (designed for devices running for years on small batteries) **-> IOT Devices like Wearables**
  - **High-power:** Wi-Fi, Cellular (require frequent charging) **-> Wifi**

3. Bandwidth

- **Definition:** The data rate supported by the connectivity technology (how much data can be sent/received per second).
- **Examples:**
  - **Low bandwidth:** LoRaWAN (0.3–50 kbps) **-> Simple Sensors**
  - **Medium bandwidth:** Bluetooth (up to 2 Mbps)
  - **High bandwidth:** Wi-Fi (up to Gbps), 4G/5G Cellular (up to Gbps) **-> Video streaming**

4. Cost

- **Definition:** The cost of deploying and maintaining the connectivity technology, including hardware, network fees, and infrastructure.
- **Examples:**
  - **Low cost:** LoRaWAN **-> Large-scale deployments with budget constraints**
  - **Medium cost:** Wi-Fi
  - **High cost:** Cellular **-> SIM cards, data plans**

5. Conditions

- **Definition:** Environmental factors influencing connectivity technology, such as interference and physical obstacles.
- **Examples:**
  - **Indoor vs outdoor:** Wi-Fi and Bluetooth better indoors; LoRaWAN and cellular better outdoors
  - **Urban vs rural:** Cellular networks better in urban areas; LPWAN better in remote areas
  - **Mobility:** Cellular preferred for moving devices (vehicles, wearables)

Short Range IoT Connectivity

These technologies are designed for communication over relatively small distances, typically within a building or a local area.

- **High Range IoT Connectivity**

Feature	Wi-Fi	Bluetooth	Mesh Network (Zigbee)	Ethernet
Range	~50–100 m indoors	~10 m (Classic), up to 100 m (BLE)	10–100 m per node; mesh extends total range	Up to 100 m via cable
Bandwidth	High (up to Gbps)	Moderate (~1–3 Mbps)	Low to moderate	Very high (up to 10+ Gbps)
Power Consumption	High	Low	Very low	Powered via cable (PoE)
Use Cases	Smart home devices (cameras, speakers, appliances)	Wearables, health monitors	Smart lighting, home automation	Industrial IoT equipment
Pros	High-speed internet/data transfer	Low power, easy pairing	Scalable, low power	Reliable, secure, very fast
Cons	Limited range	Limited range	Lower bandwidth, complexity in setup	Requires cabling, not mobile

Feature	LPWAN	LoRaWAN	NB-IoT
Range	Several km up to 15+ km	2–15 km (urban/rural)	1–10 km
Bandwidth	Low (<50 kbps)	0.3–50 kbps	Up to 250 kbps
Power Consumption	Years of battery life	5–10 years	5–10 years
Use Cases	Smart agriculture, environmental monitoring	Agriculture, environmental monitoring	healthcare, industrial IoT,