SNS COLLEGE OF ENGINEERING

An Autonomous Institution

Kurumbapalayam(Po), Coimbatore – 641 107

DEPARTMENT OF COMPUTER SCIENCE AND DESIGN

Course Code and Name : 19IT503 Internet of Things

Unit 1

UNIT1

IoT INTRODUCTION AND APPLICATIONS

Overview and Motivations - IPv6 Role - IoT Definitions - Observations - ITU-T Views – Working Definition - IoT Frameworks - Basic Nodal Capabilities – Physical Design of IoT - Logical Design of IoT – Applications:- City Automation Automotive Applications - Home Automation - IoT Levels & Deployment Templates - IoT and M2M

UNIT2

FUNDAMENTAL MECHANISMS & KEY TECHNOLOGIES

Identification of IoT Objects and Services- Structural aspects of IoT-Environment Characteristics-Traffic Characteristics-Scalability-Interoperability-Security and privacy -Key IoT Technologies :Device Intelligence - Communication Capabilities - Mobility Support - Device Power –Sensor Technology -RFID Technology - Satellite Technology - IoT Enabling Technologies- WSN, Cloud computing, Big data Analytics, communication protocols, embedded systems

UNIT3

EVOLVING IoT STANDARDS & PROTOCOLS

IETF IPv6 Routing Protocol for RPL Roll – Constrained Application Protocol (CoAP) – Representational State Transfer (REST) – Third Generation Partnership Project Service Requirements for Machine Type Communications- Over Low Power WPAN (6LoWPAN)- IP in Small Objects (IPSO) - WPAN Technologies for IoT/M2M – ZigBee/IEEE 802.15.4, RF4CE,Bluetooth and its Low-Energy Profile, Cellular and Mobile Network Technologies for IoT/M2M.

UNIT4

IPv6 TECHNOLOGIES FOR THE IOT

Motivations - Address Capabilities - IPv6 Protocol Overview - IPv6 Tunneling - IPsec in IPv6 -Header Compression Schemes - Quality of Service in IPv6 - MOBILE IPv6 -Protocol Details -Generic Mechanisms - New IPv6 Protocol - Message Types - Destination Option - Modifications to IPv6 Neighbor Discovery - Requirements for Various IPv6 Nodes - Correspondent Node Operation

UNIT5

DESIGN METHODOLOGY & FUTURE TRENDS

IoT System Management with NETCONF-YANG: Need for IoT Systems Management – Simple Network Management Protocol (SNMP) - Limitations of SNMP, Network Operator Requirements- NETCONF-YANG-IoT Systems Management with NETCONF-YANG -IoT Platforms Design Methodology – IoT Physical Devices & Endpoints - Raspberry Pi- Linux on Raspberry Pi -Raspberry Pi Interfaces - Programming Raspberry Pi with Python

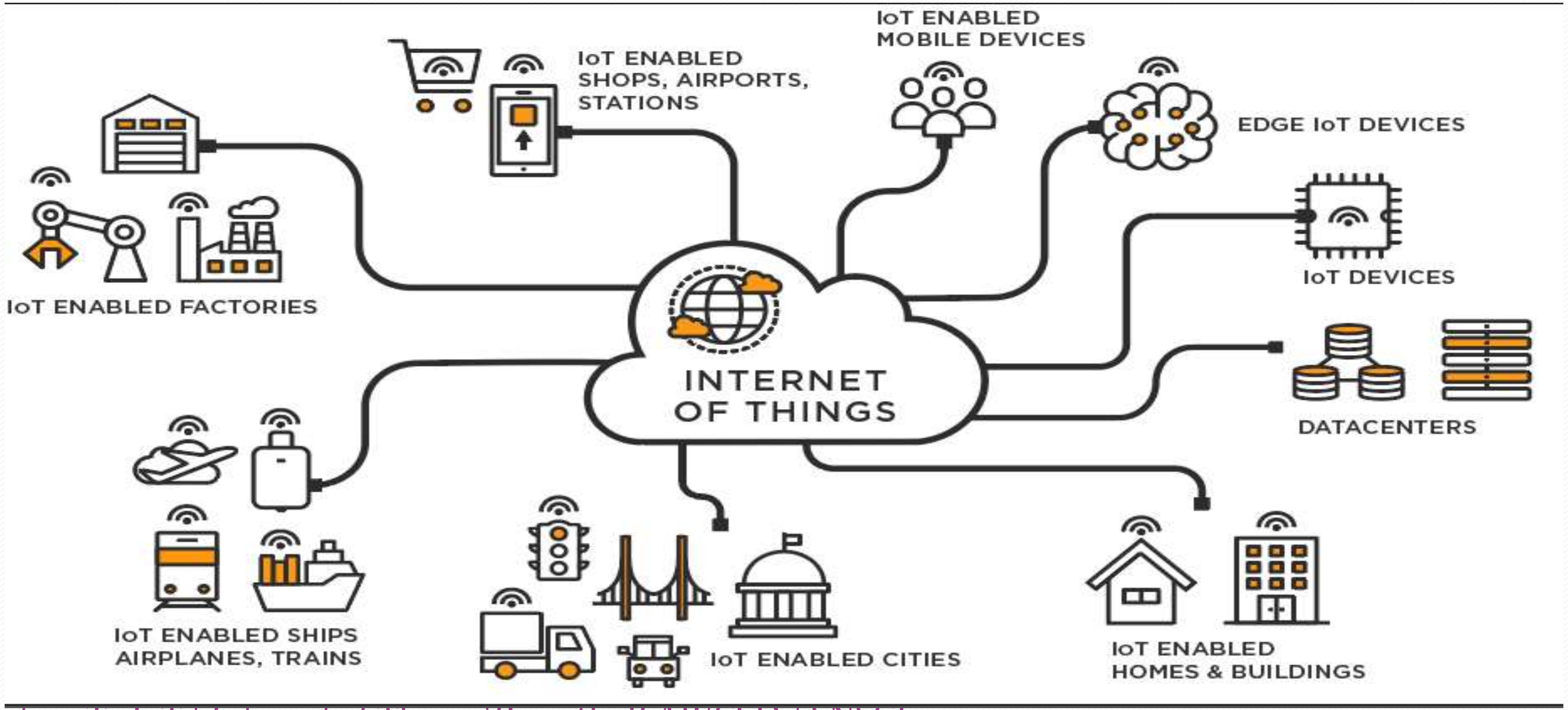
UNIT1

IoT INTRODUCTION AND APPLICATIONS

Overview and Motivations - IPv6 Role - IoT Definitions - Observations - ITU-T Views – Working Definition - IoT Frameworks - Basic Nodal Capabilities – Physical Design of IoT - Logical Design of IoT – Applications:- City Automation Automotive Applications - Home Automation - IoT Levels & Deployment Templates - IoT and M2M

Overview and Motivation

- The term 'Internet of Things' was coined in 1999 by the computer scientist **Kevin Ashton**.
- **-Assistant brand manager** at Procter & Gamble (P&G) in 1997 interested in using RFID to help manage P&G's supply chain.
- The primary purpose of IoT is to make such systems that can work without human intervention or follow a self-reporting mechanism in real-time.
- It is perceived by proponents as the "**next-generation network (NGN)** of the Internet

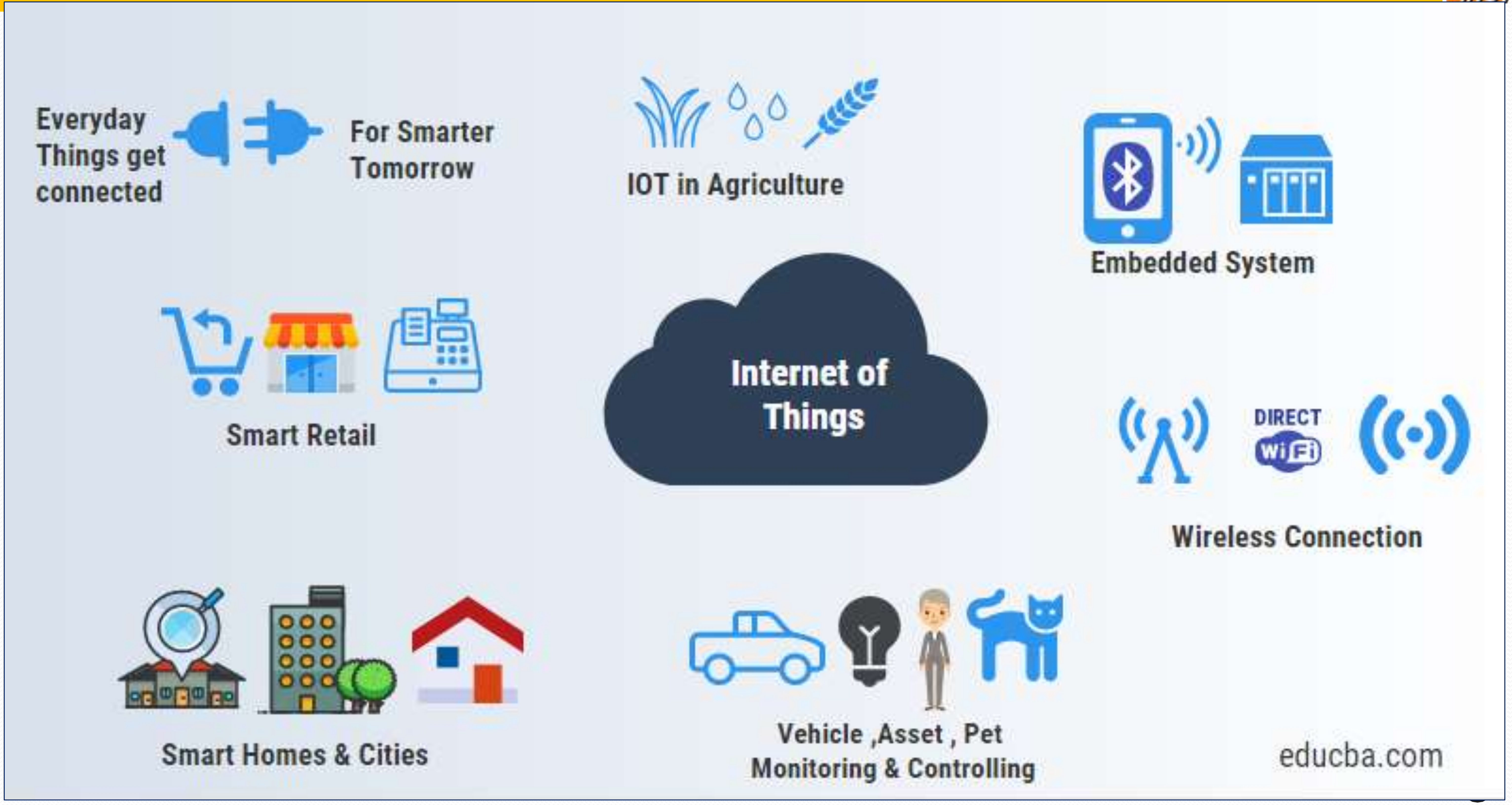

IoT ENABLED MOBILE DEVICES
IoT ENABLED SHOPS, AIRPORTS, STATIONS
EDGE IoT DEVICES
IoT ENABLED FACTORIES
IoT DEVICES
INTERNET OF THINGS
DATACENTERS
IoT ENABLED SHIPS AIRPLANES, TRAINS
IoT ENABLED CITIES
IoT ENABLED HOMES & BUILDINGS

KEVIN ASHTON

What is IOT?

- The Internet of Things (IoT) describes **physical objects embedded with sensors and actuators** that communicate with computing systems via wired or wireless networks—allowing the physical world to be digitally monitored or even controlled.

- Physical objects - embedded with sensors-which can monitor things like temperature or motion, or really any change in environment—**Actuators**—which receive signals from sensors and then do something in response to those changes. The sensors and communicate via wired (for example, Ethernet) or wireless (for example, WiFi, cellular) networks with computing systems that can monitor or manage the health and actions of connected objects and machines.


Everyday Things get connected
For Smarter Tomorrow
IOT in Agriculture
Embedded System
Internet of Things
Smart Retail
DIRECT
WiFi
Wireless Connection
Smart Homes & Cities
Vehicle ,Asset , Pet Monitoring & Controlling
educba.com

Evolution of Internet to IOT

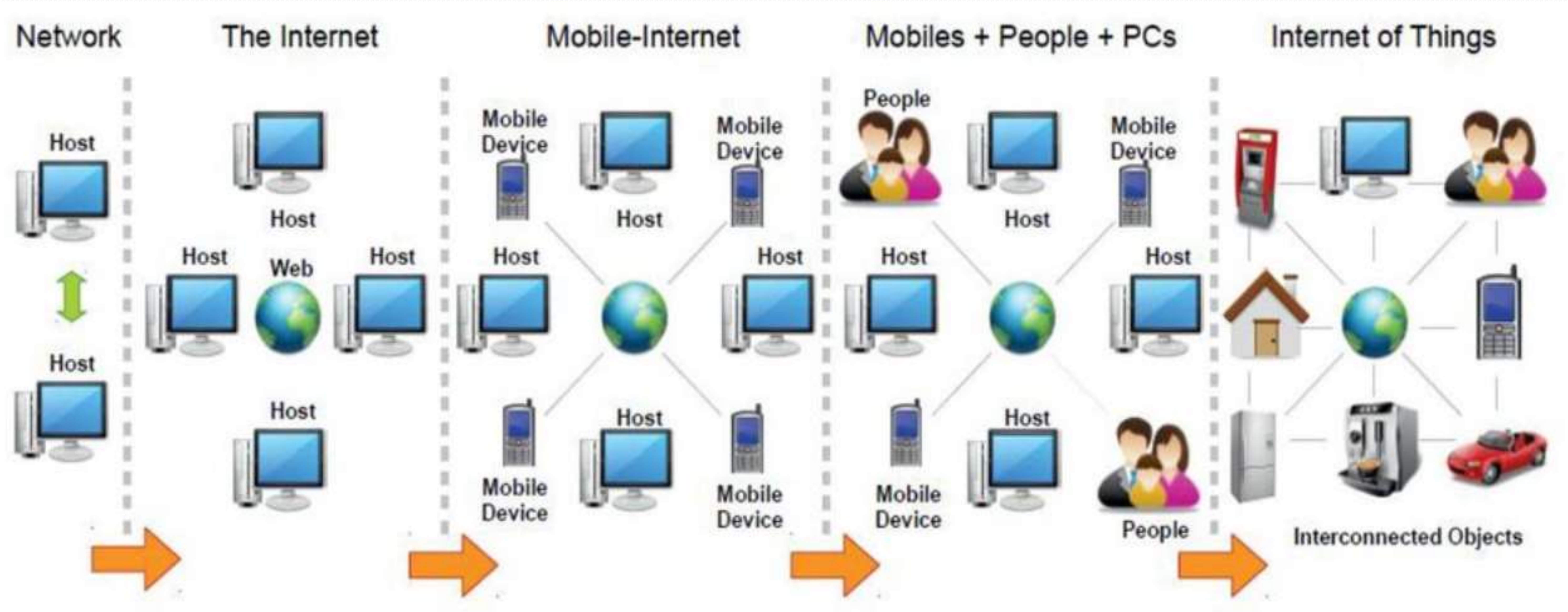
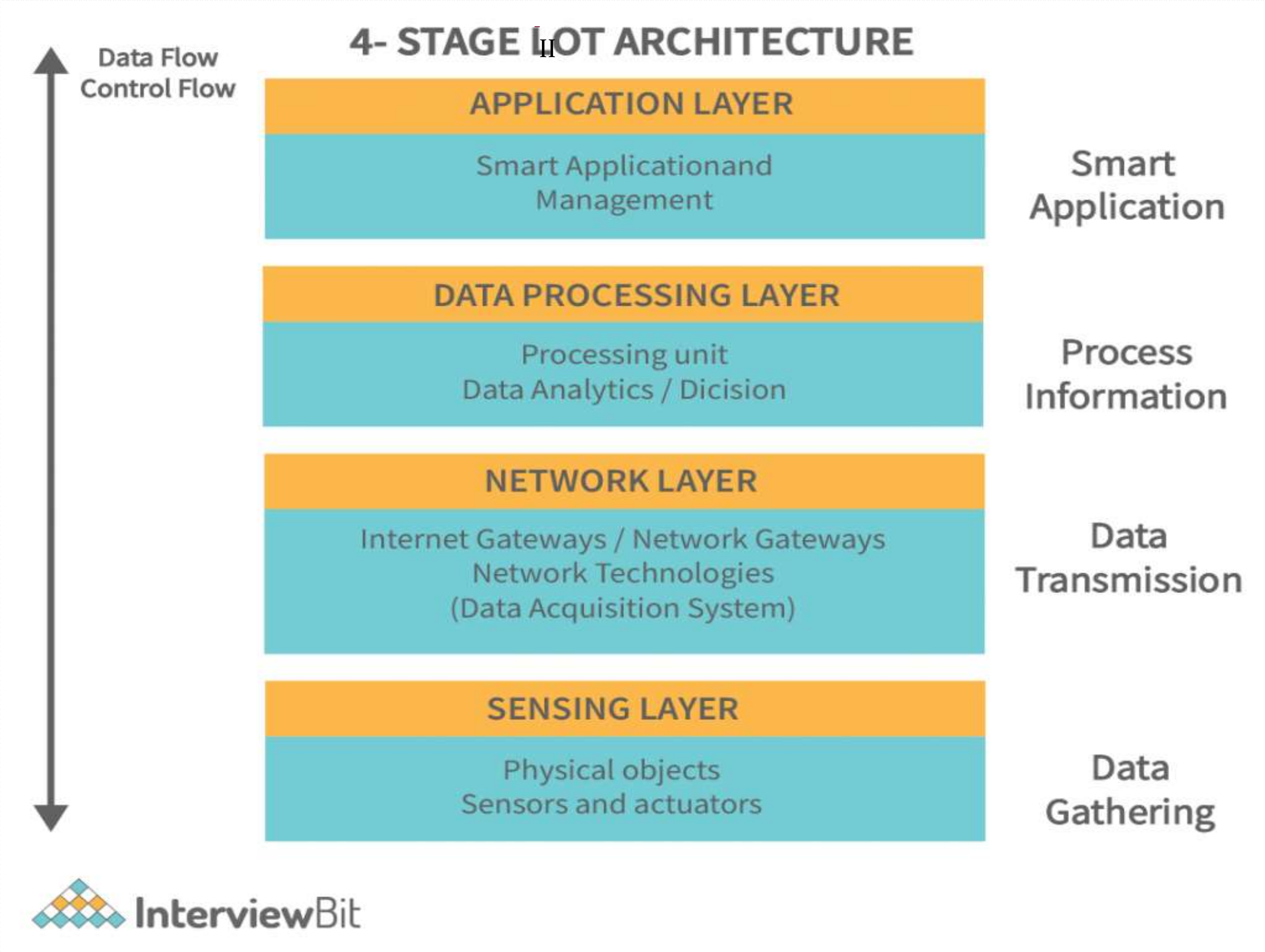


Fig. 1. Evolution of the Internet in five phases. The evolution of Internet begins with connecting two computers together and then moved towards creating World Wide Web by connecting large number of computers together. The mobile-Internet emerged by connecting mobile devices to the Internet. Then, peoples' identities joined the Internet via social networks. Finally, it is moving towards Internet of Things by connecting every day objects to the Internet.

Overview and Motivation



4- STAGE IOT ARCHITECTURE

Data Flow
Control Flow

APPLICATION LAYER

Smart Applicationand
Management

Smart
Application

DATA PROCESSING LAYER

Processing unit
Data Analytics / Dicision

Process
Information

NETWORK LAYER

Internet Gateways / Network Gateways
Network Technologies
(Data Acquisition System)

Data
Transmission

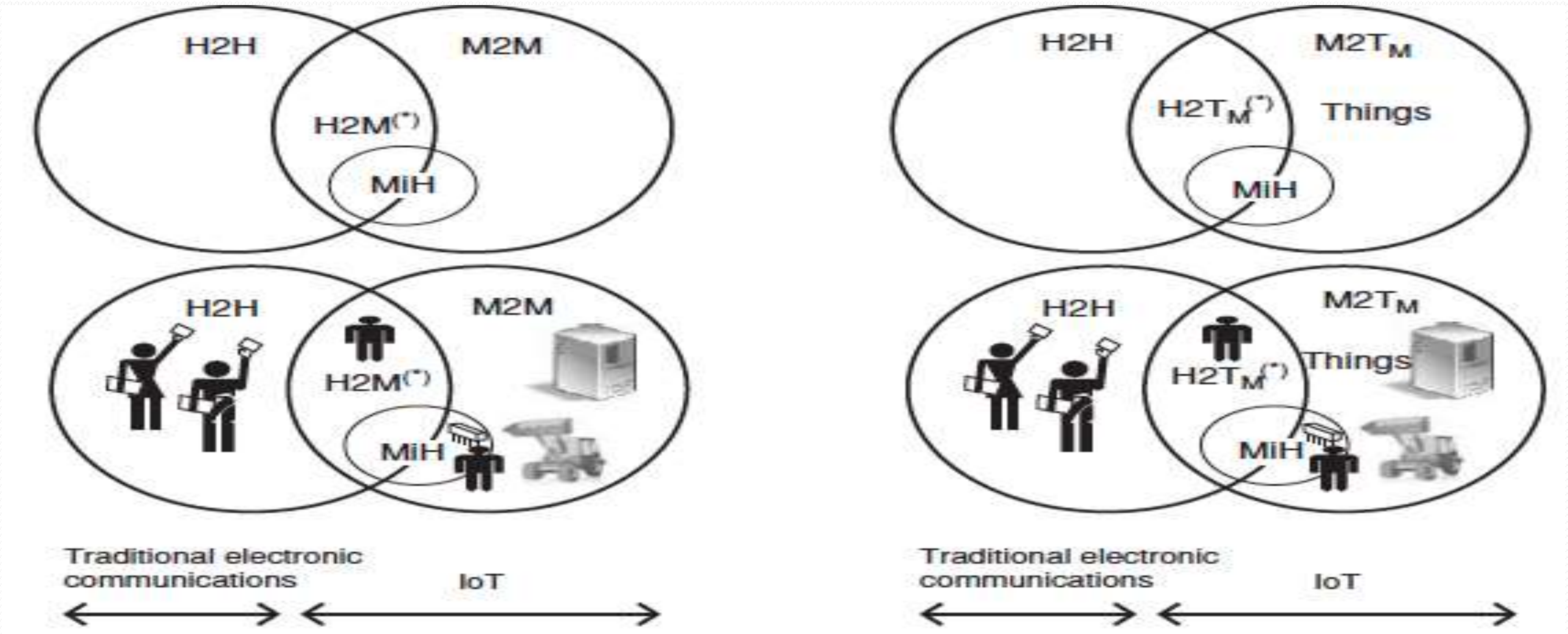
SENSING LAYER

Physical objects
Sensors and actuators

Data
Gathering

InterviewBit

Overview and Motivation

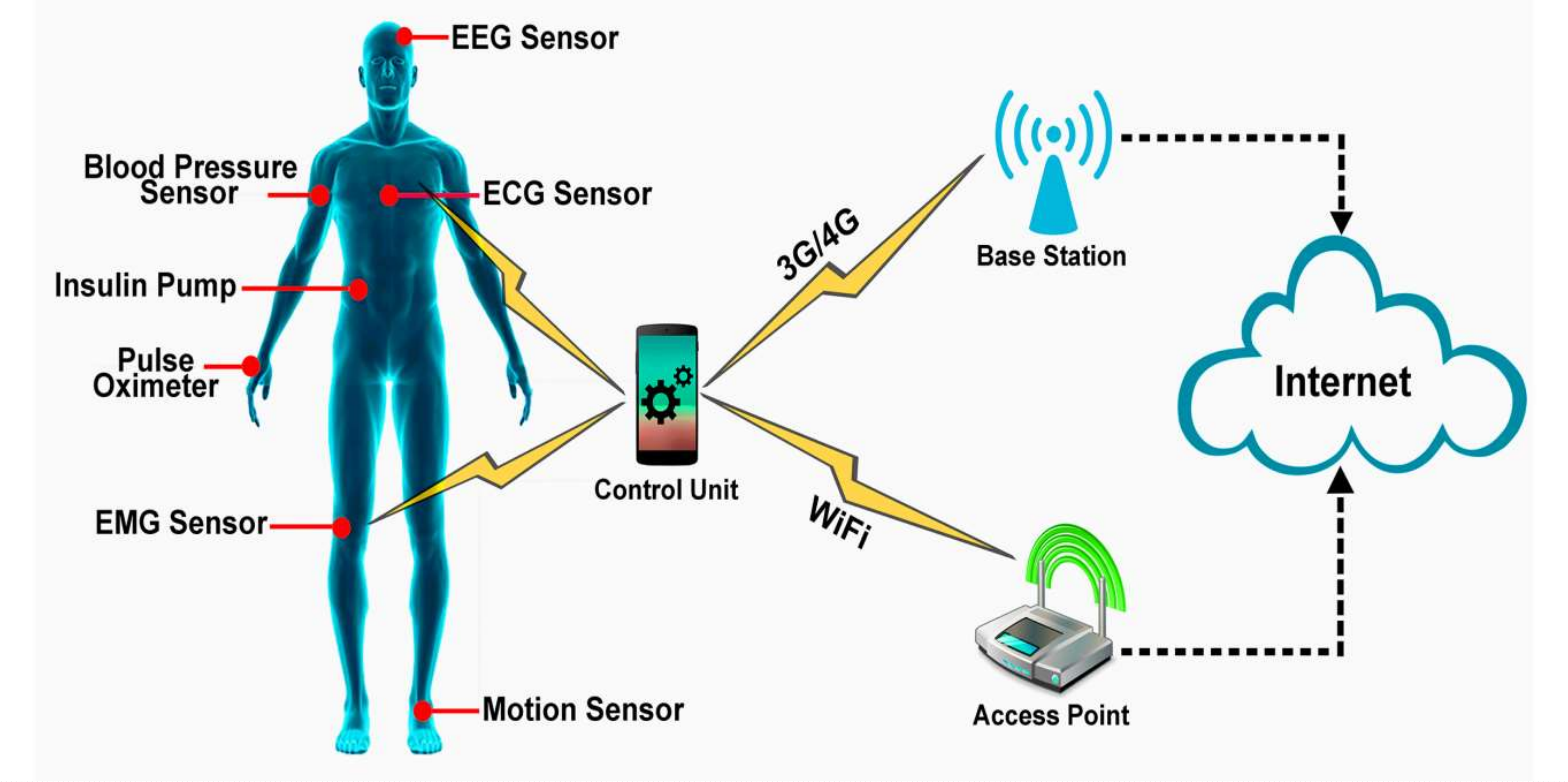


H2H: Human to Human
H2M: Human to Machine – $H2T_M$: Human to Thing with Microprocessor/Machine
M2M: Machine to Machine – $M2T_M$: Machine to Thing with Microprocessor/Machine
MiH: Machine in Humans
(e.g., medical sensors)
(also includes chips in animals/pets)

(*) People have been communicating with computers for over half-a-century, but in this context "machine" means a microprocessor embedded in some objects (other than a traditional computer)

- human-to-human (H2H) communication,
- M2M communication, H2M communications, and
- machine in (or on) humans (MiH) communications (MiH devices may include human
embedded chips, medical monitoring probes, global positioning system (GPS) bracelets,
and so on).

Machine in human



ACTIVITY

1.Who is inventor of IOT?

Write it using mirror Image

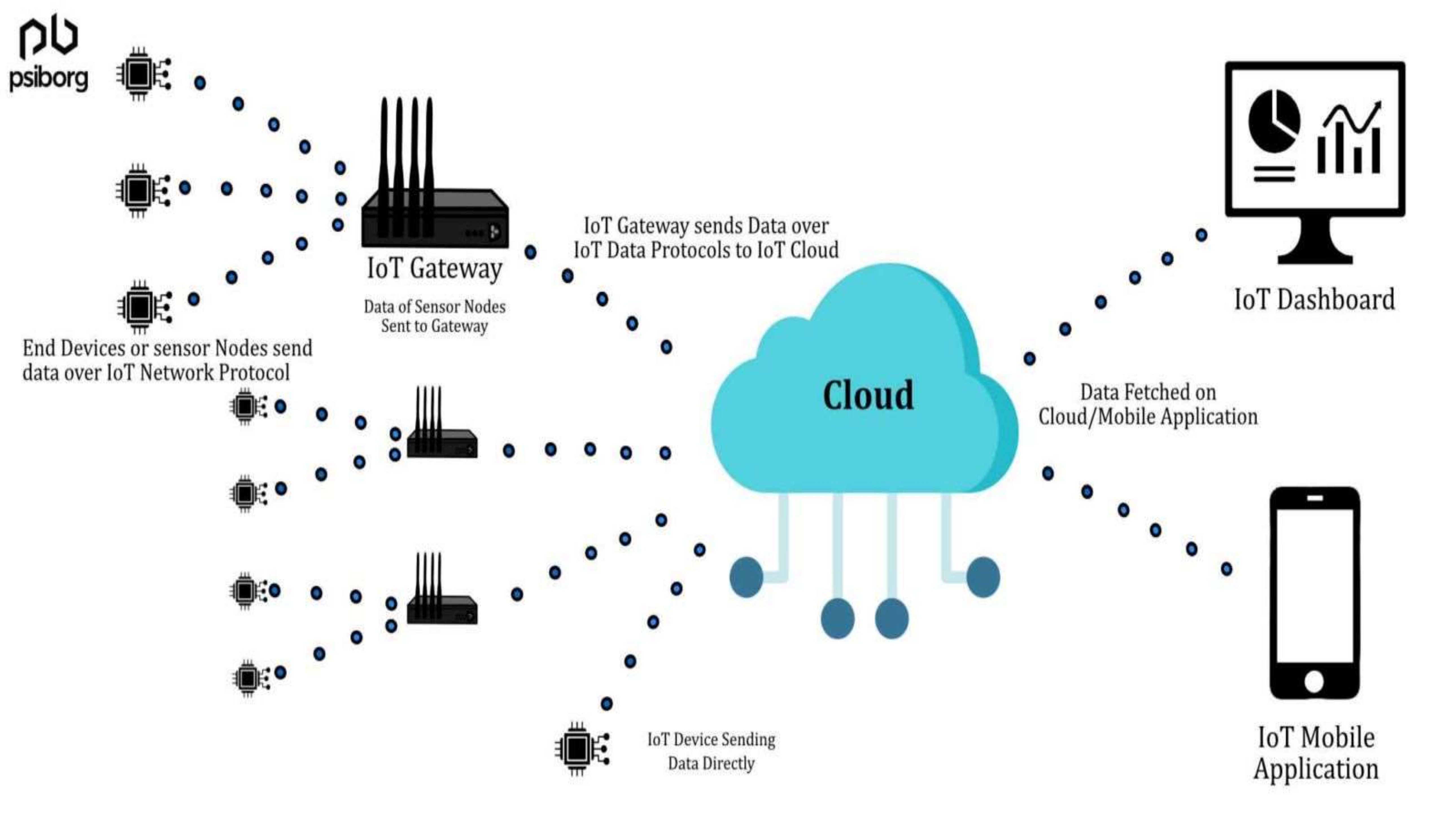
2.What industries can benefit from IoT?

Write it using mirror Image

- ***Some warm activities***

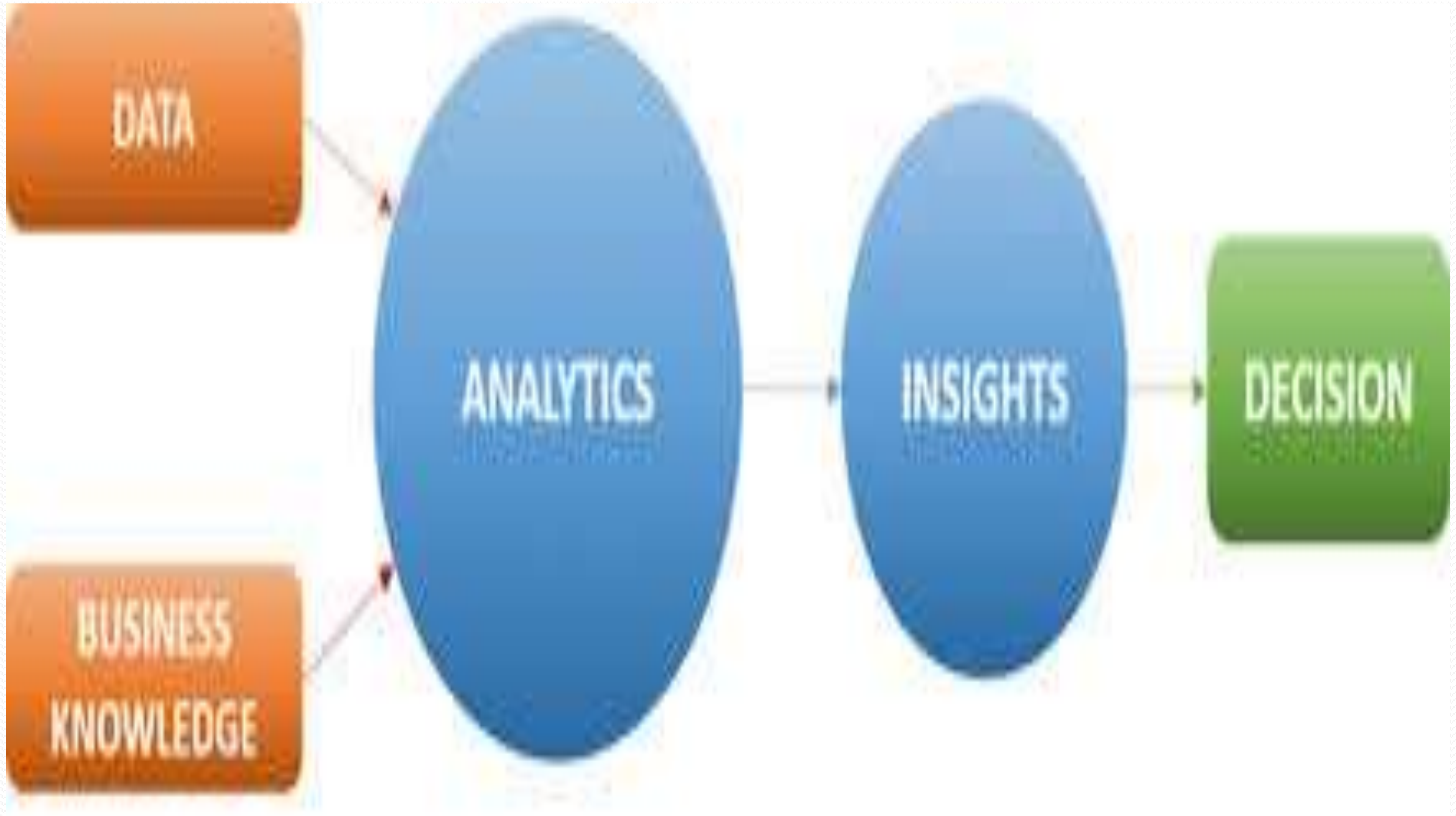
Motivation of IoT:

❖ **Connectivity and Communication**



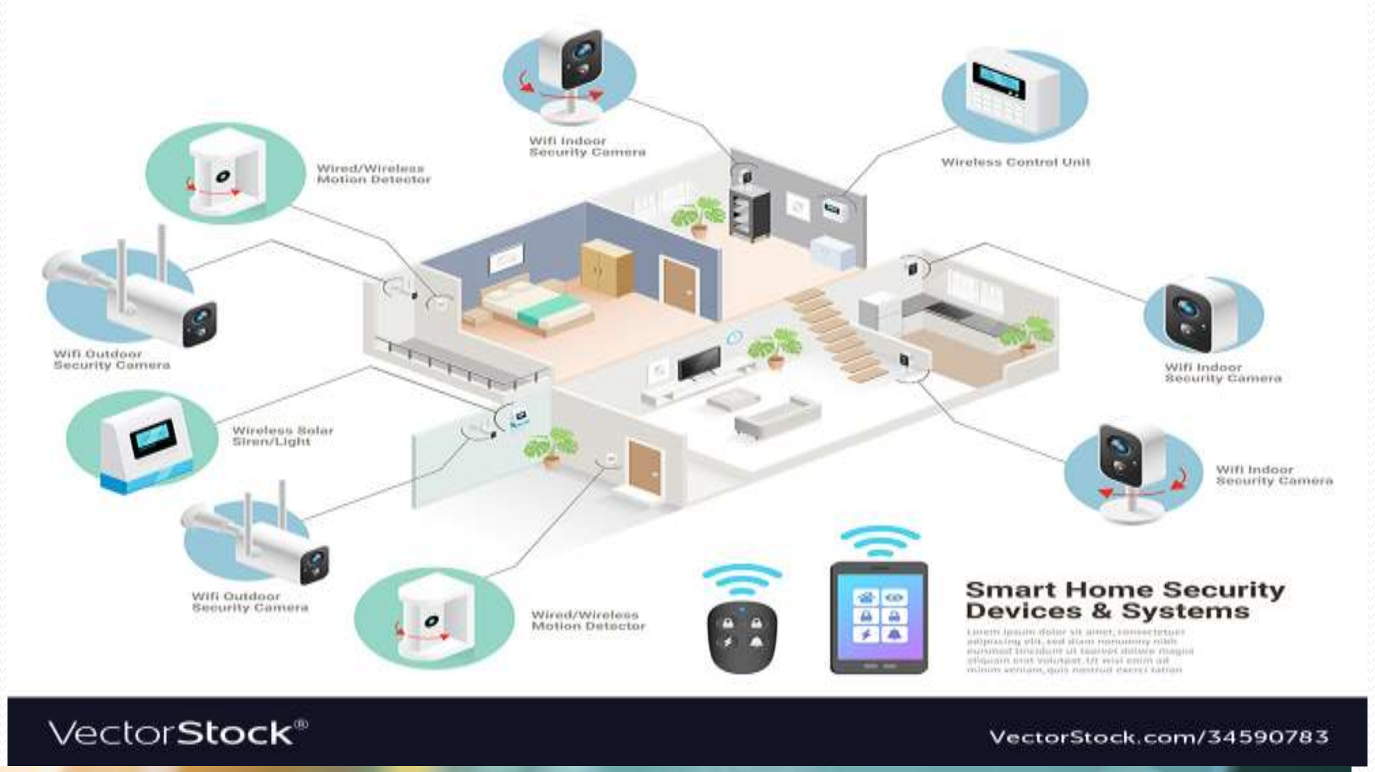
psiborg
IoT Gateway
Data of Sensor Nodes Sent to Gateway
End Devices or sensor Nodes send data over IoT Network Protocol
IoT Gateway sends Data over IoT Data Protocols to IoT Cloud
Cloud
Data Fetched on Cloud/Mobile Application
IoT Dashboard
IoT Mobile Application
IoT Device Sending Data Directly

❖ Data Insights and Analytics



❖ Automation and Efficiency

Enhanced User Experience


Wifi Indoor Security Camera
Wireless Control Unit
Wired/Wireless Motion Detector
Wifi Outdoor Security Camera
Wifi Indoor Security Camera
Wireless Solar Siren/Light
Wifi Indoor Security Camera
Wifi Outdoor Security Camera
Wired/Wireless Motion Detector
Smart Home Security Devices & Systems
VectorStock®
VectorStock.com/34590783

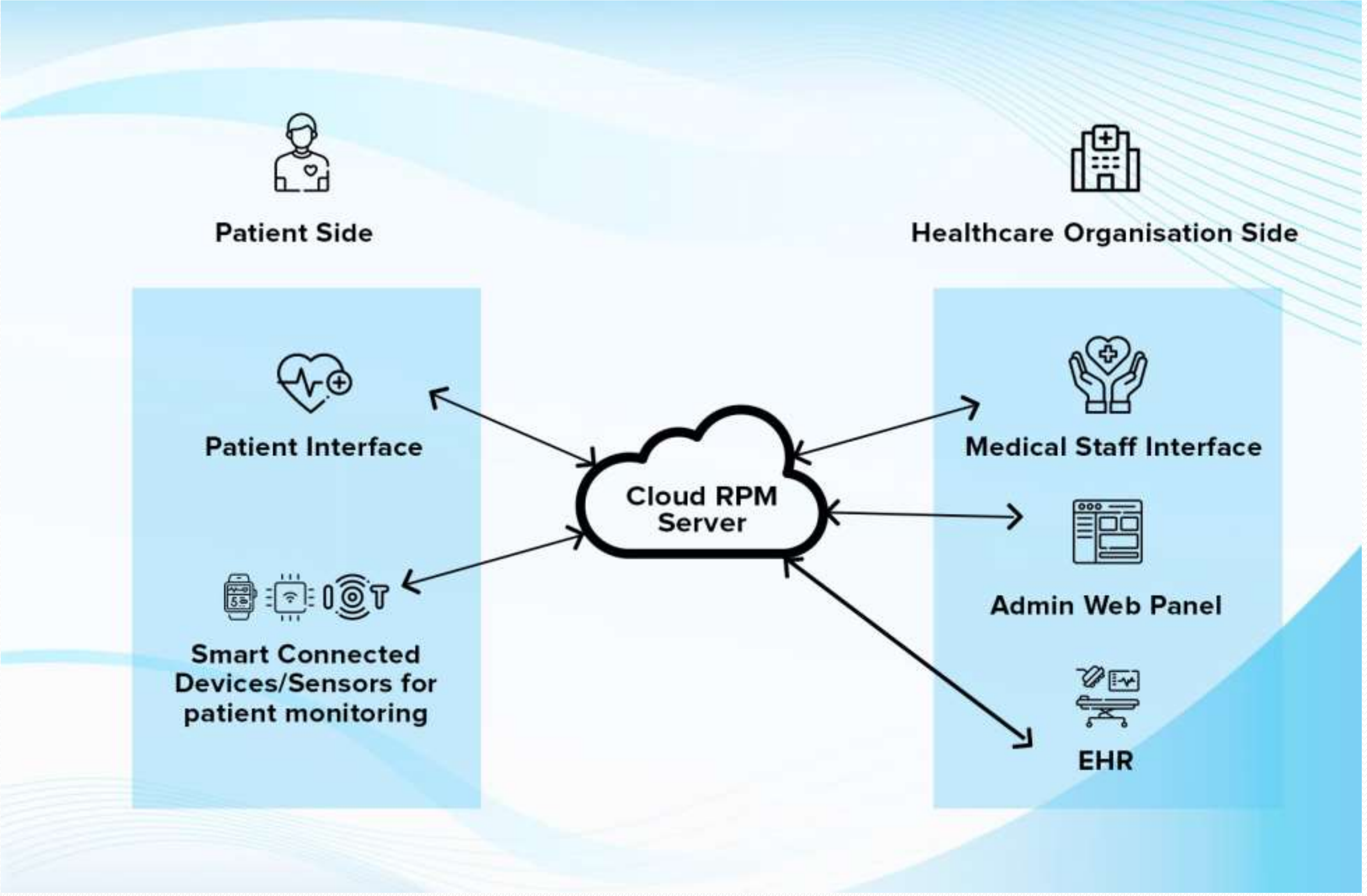
- context-aware services

ex

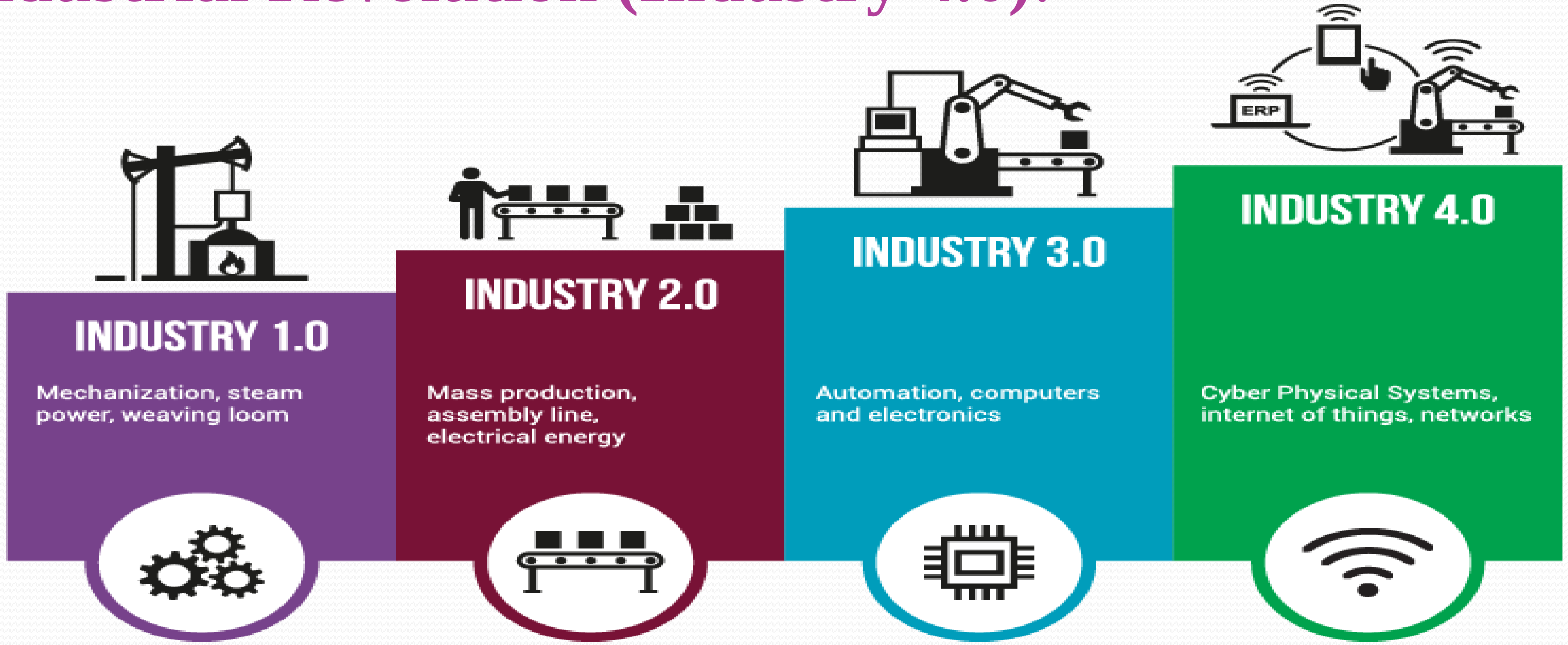
- home devices, wearable health trackers

IOT

Healthcare and Remote Monitoring



Industrial Revolution (Industry 4.0):


INDUSTRY 1.0
Mechanization, steam power, weaving loom
INDUSTRY 2.0
Mass production, assembly line, electrical energy
INDUSTRY 3.0
Automation, computers and electronics
INDUSTRY 4.0
Cyber Physical Systems, internet of things, networks
ERP

- IoT plays a central role in the fourth industrial revolution (Industry 4.0)
- automation in production processes

- **Innovation and Technological Advancements**: The growth of IoT has driven innovation in various fields, from hardware development to cloud computing, data analytics, and artificial intelligence.

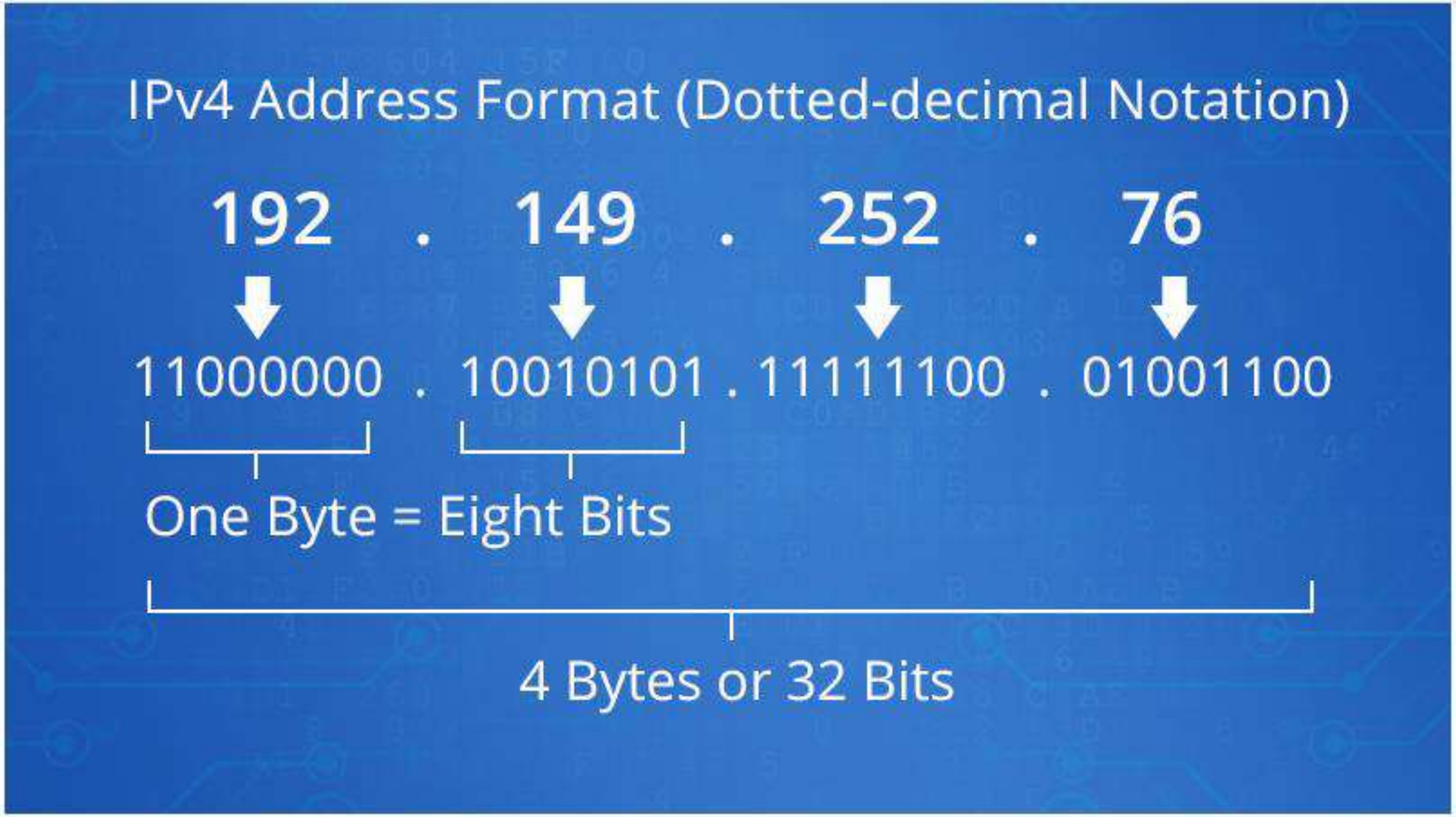
Overview and Motivation

Role of IPv6

- IPv6 with its abundant address spaces,
- globally unique object (thing) identification

- permanent unique identifier, an object ID (OID)
- unique network address (Nadr)

- IPv4



IPv4 Address Format (Dotted-decimal Notation)

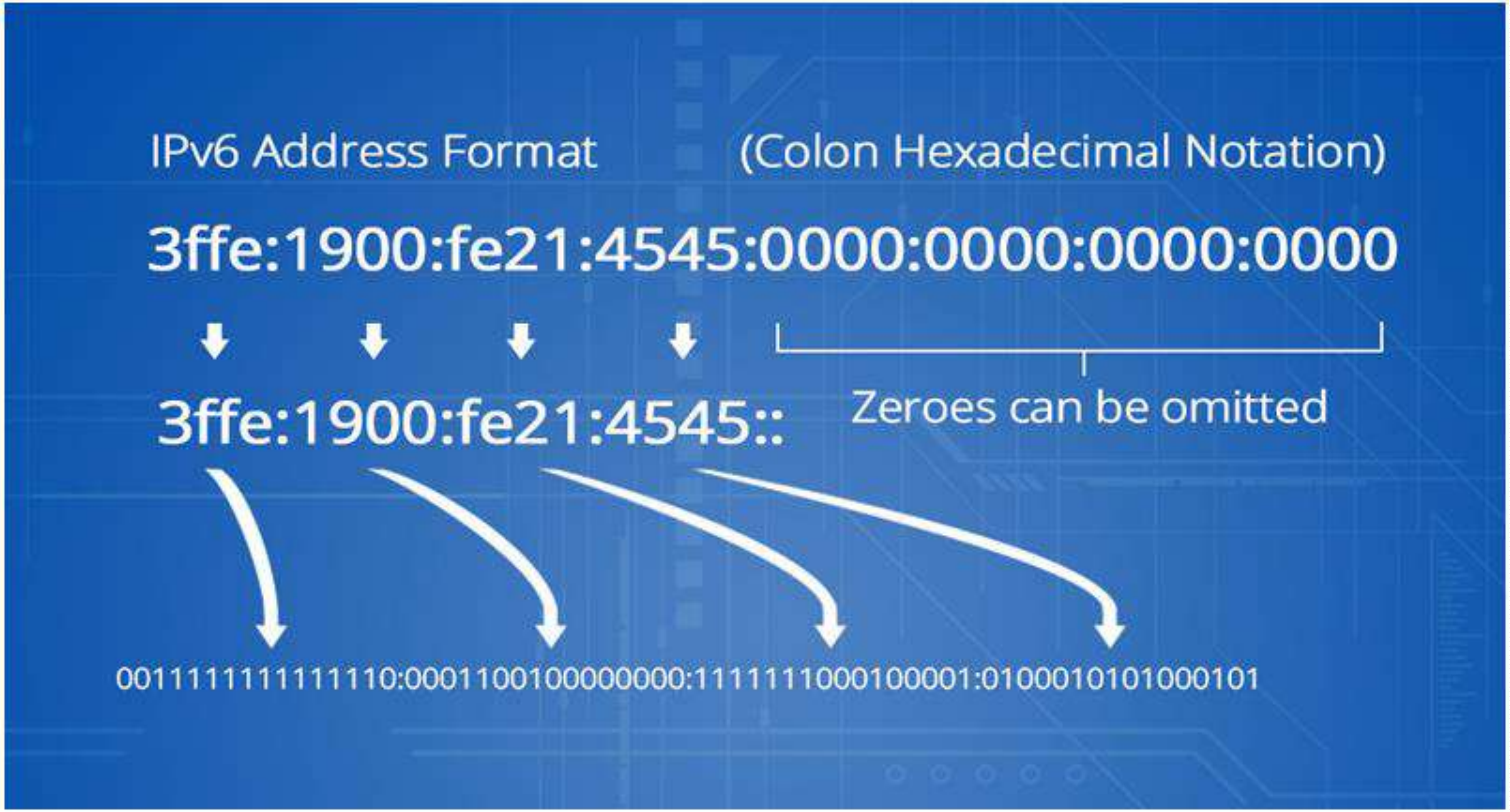
192 . 149 . 252 . 76

11000000 . 10010101 . 11111100 . 01001100

One Byte = Eight Bits

4 Bytes or 32 Bits

Ipv6



IPv6 Address Format (Colon Hexadecimal Notation)

3ffe:1900:fe21:4545:0000:0000:0000:0000

3ffe:1900:fe21:4545::

Zeroes can be omitted

0011111111111110:0001100100000000:1111111000100001:0100010101000101

Advances of IPv6

Larger Address Space: IPv6 offers 128-bit addresses, providing a vast number of unique addresses (about 3.4 × 10^38 addresses)

IPv6 has more addresses

4.3 billion addresses(ipv4)

340 trillion trillion trillion addresses

- **Autoconfiguration and Plug-and-Play**: IPv6 supports stateless address autoconfiguration (SLAAC), allowing devices to automatically generate their unique IP addresses without the need for manual configuration
- **Simplified Header Structure**: IPv6 has a simplified and more efficient header structure compared to IPv4. The IPv6 header is fixed in size (40 bytes)
- **Improved Security**: IPv6 includes built-in support for IPsec (Internet Protocol Security), which provides authentication, encryption, and data integrity services for the entire IPv6 packet.

- **Mobility Support:** IPv6 has native support for mobile devices
- **Efficient Multicast**: IPv6 introduces efficient native multicast support, which simplifies the handling of multicast traffic, making it more scalable and efficient than IPv4's multicast implementation.

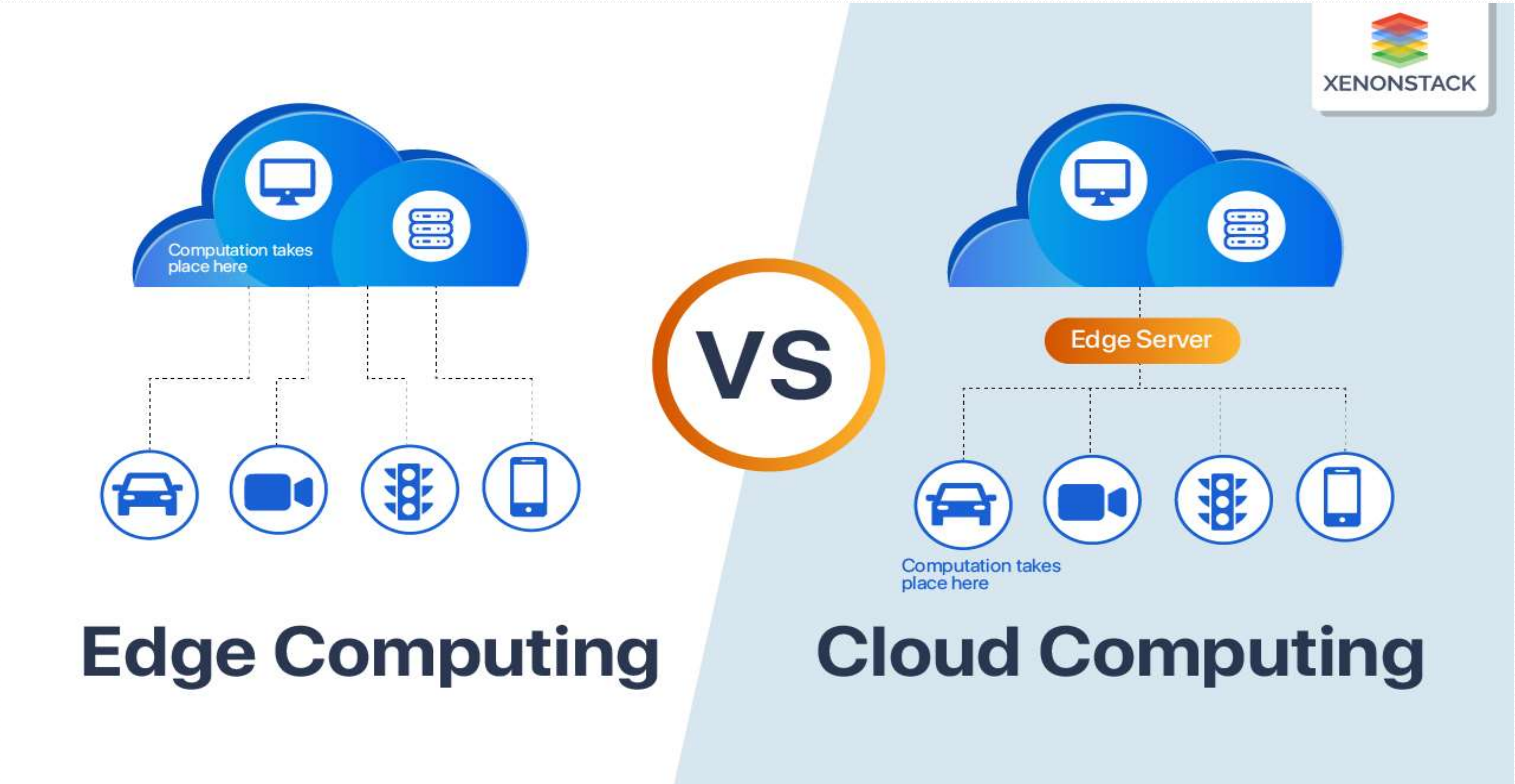
Observation

- Observation of IoT (Internet of Things) involves studying and analyzing various aspects of IoT deployments, applications, and the impact of connected devices on different domains. Researchers, businesses, and individuals can make several observations related to IoT. Here are some key observations:

- **Proliferation(growh) of Connected Devices**: IoT has led to an explosion of connected devices across various sectors, including homes, industries, healthcare, agriculture, transportation, and more.
- **Data Generation and Analysis**: IoT generates vast amounts of data from connected devices.
- **Improvement in Efficiency and Automation**: IoT enables automation and optimization of processes

- **Enhanced User Experience**: IoT devices contribute to improved user experiences by providing personalized services and real-time information. Smart homes, wearable health trackers, and smart cities are prime examples of IoT enhancing user convenience and quality of life.
- **Security and Privacy Challenges**: IoT devices often collect sensitive data, leading to concerns about security and privacy. Vulnerabilities in IoT devices can be exploited by cybercriminals, potentially causing significant harm.
- **Impact on Sustainability**: IoT has the potential to contribute to sustainability efforts. Smart energy management, water conservation, and waste reduction initiatives are some of the ways IoT can help create a more sustainable future.

- **Edge Computing**: IoT generates enormous amounts of data, making cloud-based processing sometimes impractical. Edge computing, where data is processed closer to the source



- **IoT in Healthcare**: IoT applications in healthcare, such as remote patient monitoring and connected medical devices, are revolutionizing patient care and enabling telemedicine services.
- **Smart Cities**: IoT is transforming cities into smart cities, where connected infrastructure enhances public services, improves traffic management, and optimizes resource allocation.
- **Industrial IoT (IIoT)**: IoT is reshaping industries with predictive maintenance, remote monitoring, and real-time data analytics, leading to cost savings and operational efficiency.

ITU(International Telecommunication Union) t views

- 1965-established-headquartered in Geneva, Switzerland
- The main objectives of the ITU are to promote the development and efficient use of telecommunications and ICT networks and services, as well as to allocate global radio-frequency spectrum and satellite orbits

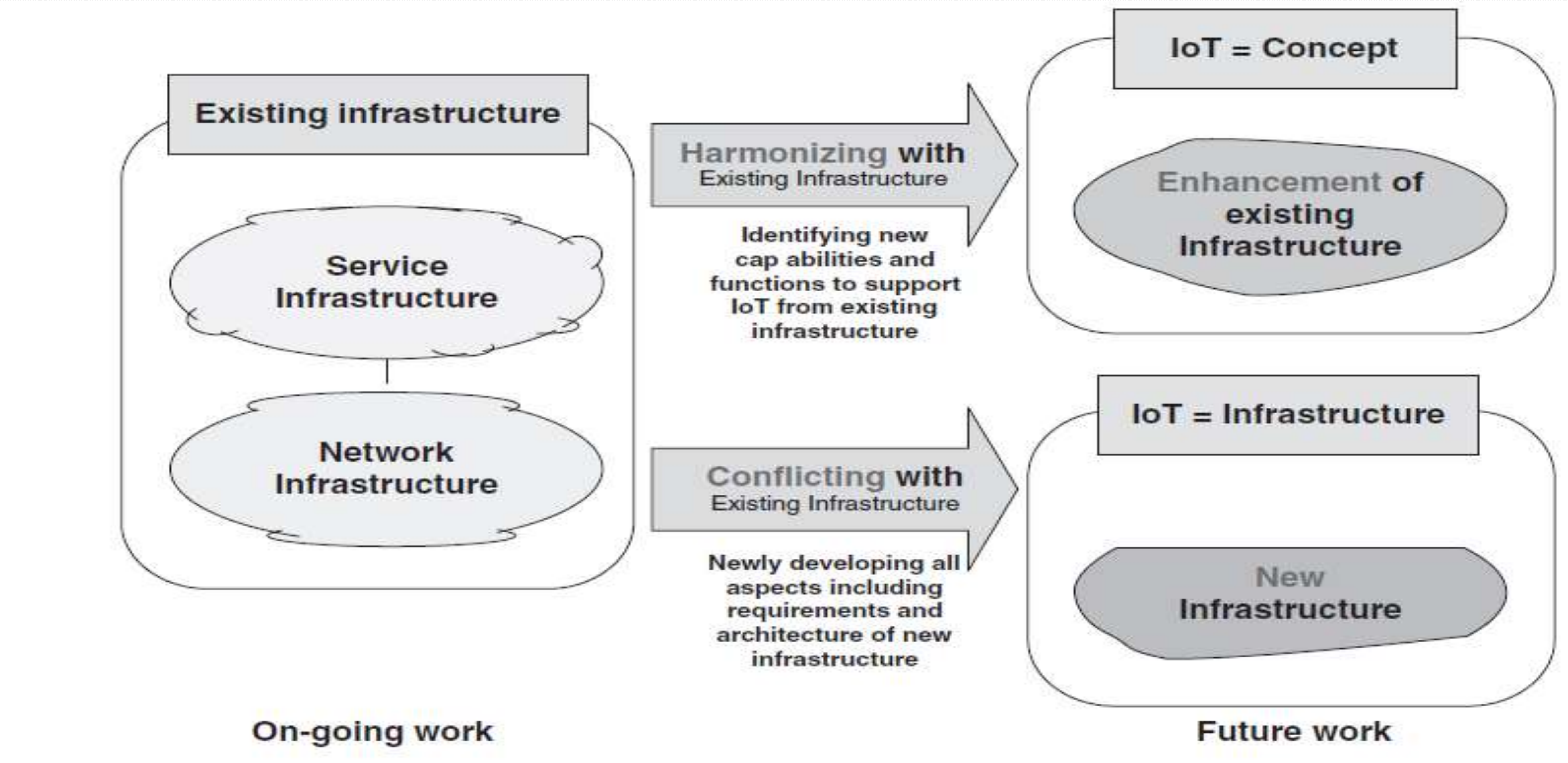
Some key functions and activities of the ITU include:

- Standardization
- Radio-frequency Allocation
- Development Sector
- Telecommunication Development Assistance
- Radiocommunication Sector
- Telecommunication Standardization Sector
- ITU Study Groups
- Conferences and Events

The ITU-T is in the process of identifying a common way to define/describe the IoT.

- Infrastructure View - Internet as an infrastructure provide a number of technological capabilities.
- Concept View - Internet as an concept provide an array of data exchange and linkage services.

- View A: IoT is just a concept (conceptual aspects of definition)
- View B: IoT is an infrastructure: The IoT refers to an infrastructure


Existing infrastructure
Service Infrastructure
Network Infrastructure
Harmonizing with Existing Infrastructure
Identifying new cap abilities and functions to support IoT from existing infrastructure
Conflicting with Existing Infrastructure
Newly developing all aspects including requirements and architecture of new infrastructure
IoT = Concept
Enhancement of existing Infrastructure
IoT = Infrastructure
New Infrastructure
On-going work
Future work

Direction for standardization according to IoT definition.

WORKING DEFINITION IOT

A broadly-deployed aggregate computing/communication application and/or application-consumption system, that is deployed over a local (L-IoT), metropolitan (M-IoT), regional (R-IoT), national (N-IoT), or global (G-IoT) geography, consisting of (i) dispersed instrumented objects (“things”) with embedded one or two-way communications and some (or, at times, no) computing capabilities, (ii) where objects are reachable over a variety of wireless or wired local area and/or wide area networks, and, (iii) whose inbound data and/or outbound commands are pipelined to or issued by a(n application) system with a (high) degree of (human or computer-based) intelligence.

Working Definition of Sensors :

Sensors are active devices that measure some variable of the natural or man-made environment (e.g., a building, an assembly line, an industrial assemblage supporting a process).

Working Definition of Actuators :

- An actuator is a mechanized device of various sizes (from ultra-small to very large) that accomplishes a specified physical action, for example, controlling a mechanism or system, opening or closing a valve, starting some kind or rotary or linear motion, or initiating physical locomotion. An actuator is the mechanism by which an entity acts upon an environment.

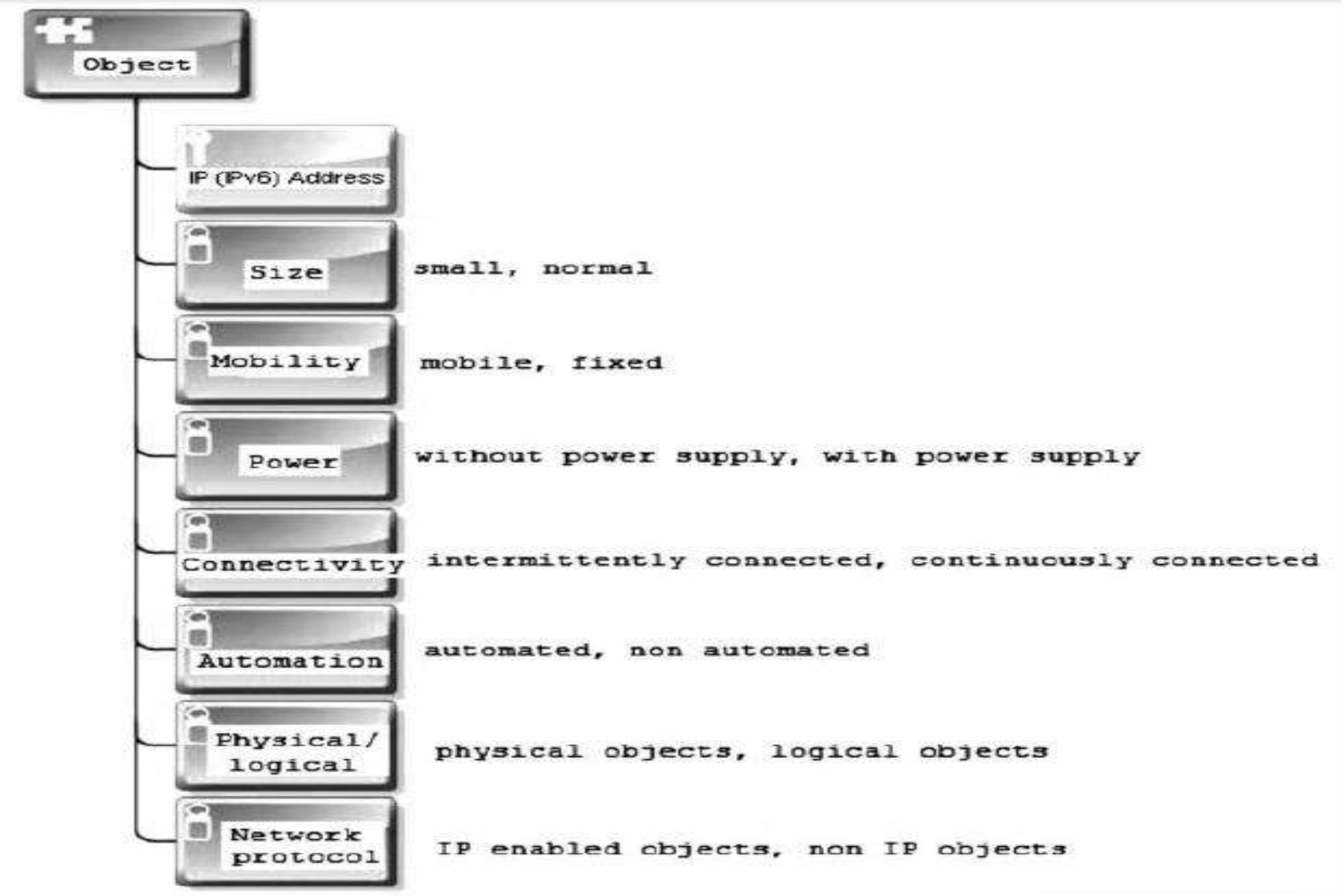
Object :

- □ An object is a model of an entity.
- □ An object is distinct from any other object and is characterized by its behaviour.
- □ An object is informally said to perform functions and offer services

Objects have the following characteristics :

- □ have the ability to sense or actuate
- □ are generally small (but not always)
- □ managed by devices, not people (but not always)

Object classification



Object

IP (IPv6) Address

Size small, normal

Mobility mobile, fixed

Power without power supply, with power supply

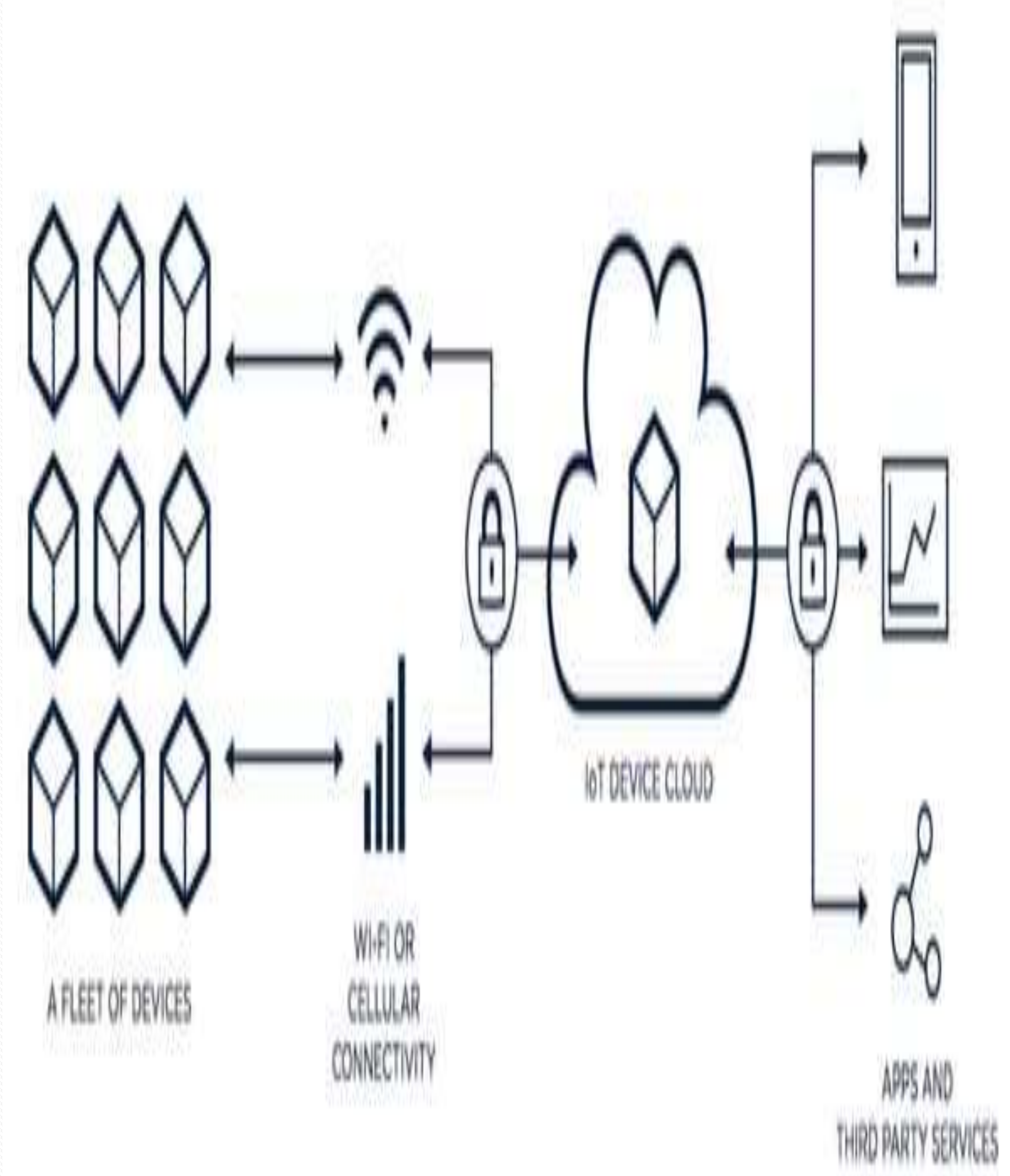
Connectivity intermittently connected, continuously connected

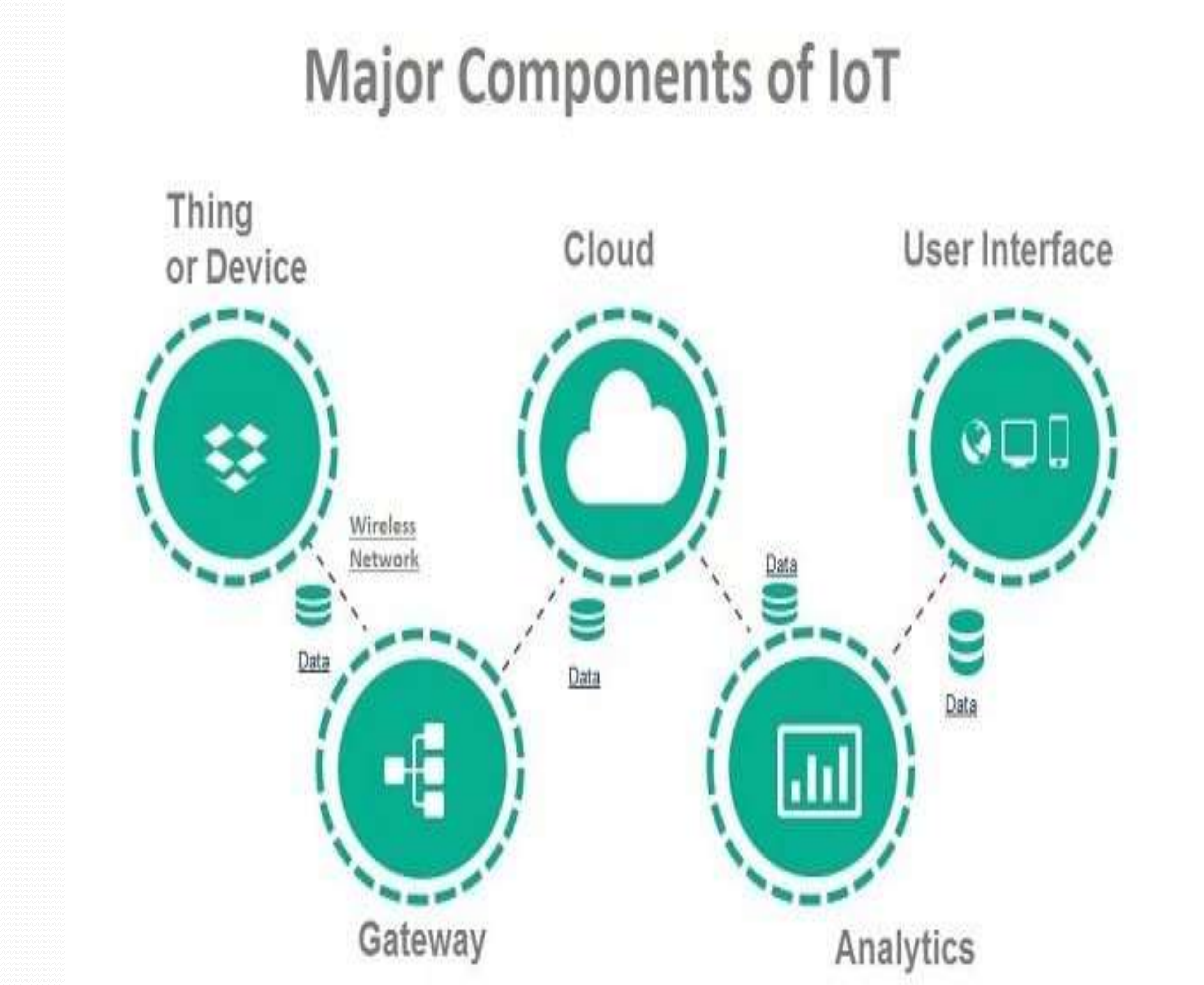
Automation automated, non automated

Physical/ logical physical objects, logical objects

Network protocol IP enabled objects, non IP objects

IOT FRAMMEWORK





IOT- Framework: IoT Frameworks With help of High Level M2M System Architecture (HLSA)

- The HLSA comprises
  - the device and gateway domain,
  - the network domain, and
  - the applications domain.

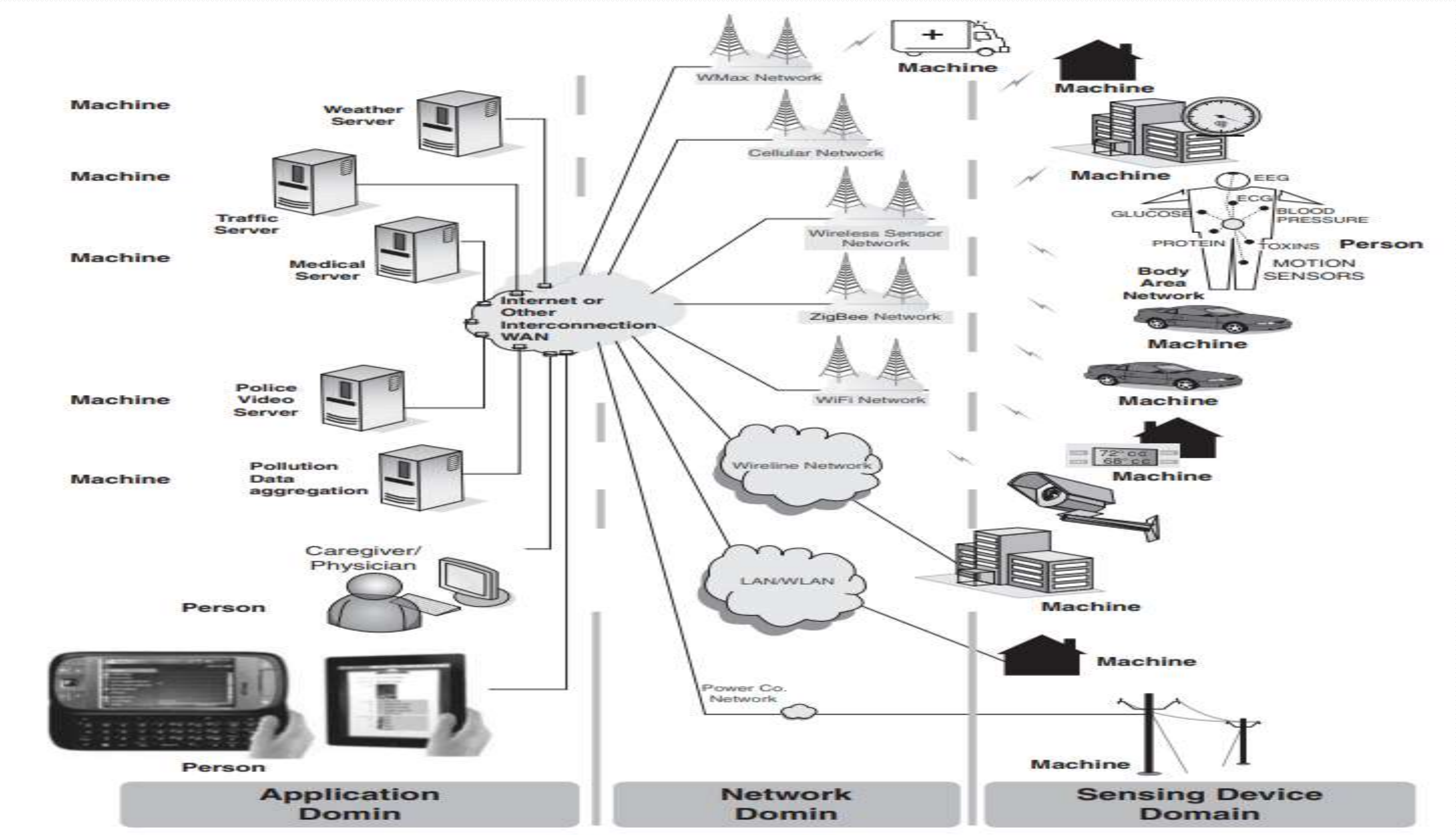


FIGURE . M2M domains.

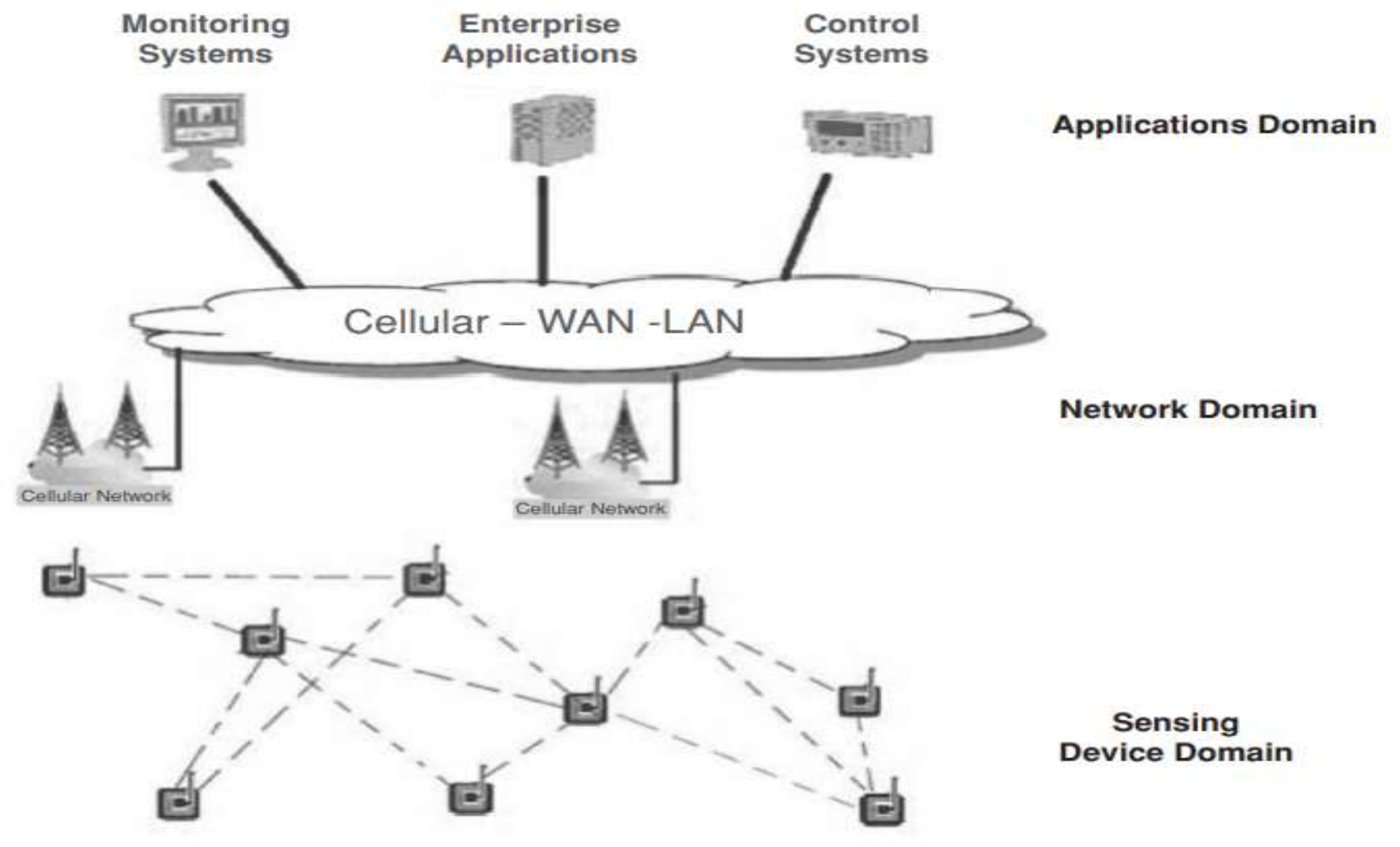


FIGURE Other example of M2M domains.

IOT-Framework: IoT Frameworks With help of High Level M2M System Architecture (HLSA)

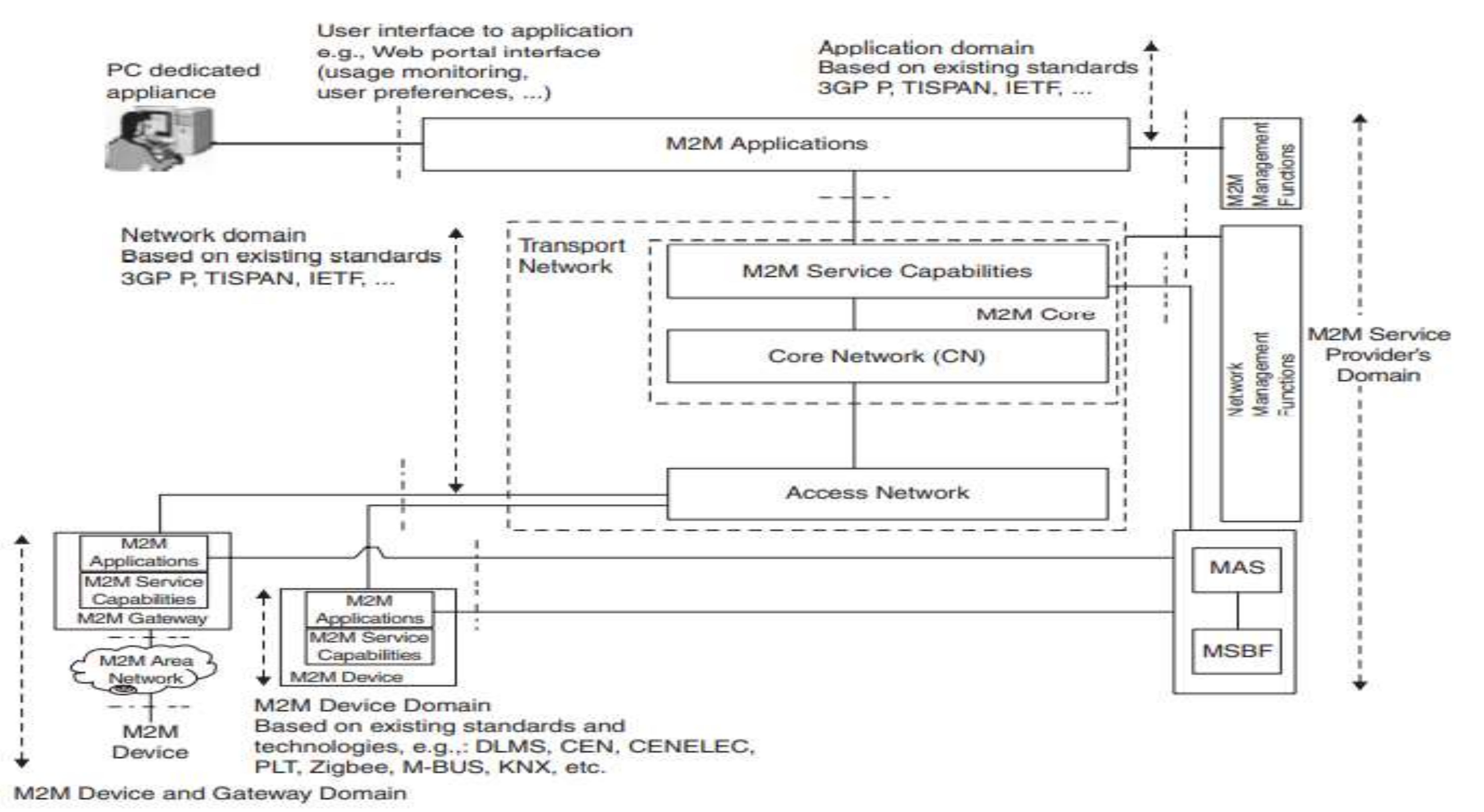
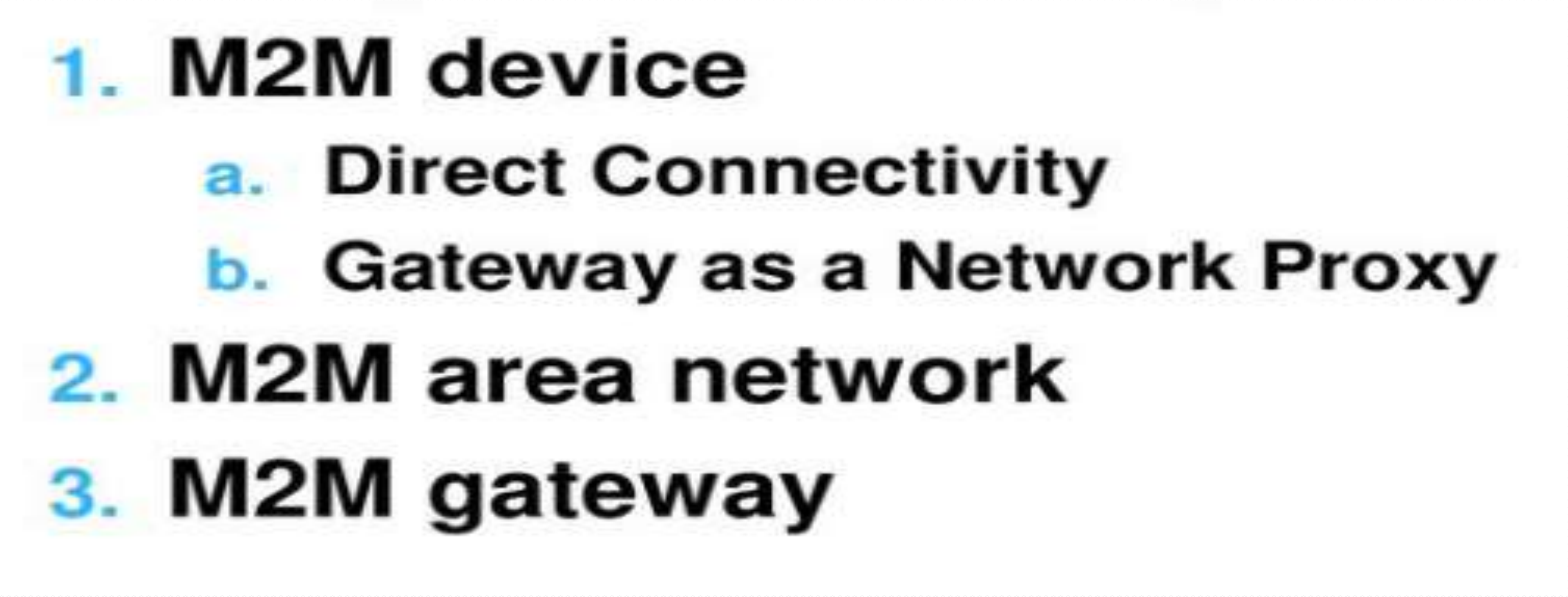

PC dedicated appliance
User interface to application e.g., Web portal interface (usage monitoring, user preferences, ...)
Application domain Based on existing standards 3GP P, TISPAN, IETF, ...
M2M Applications
M2M Management Functions
Network domain Based on existing standards 3GP P, TISPAN, IETF, ...
Transport Network
M2M Service Capabilities
M2M Core
Core Network (CN)
Access Network
Network Management Functions
M2M Service Provider's Domain
M2M Applications
M2M Service Capabilities
M2M Gateway
M2M Area Network
M2M Device
M2M Applications
M2M Service Capabilities
M2M Device
MAS
MSBF
M2M Device Domain Based on existing standards and technologies, e.g.,: DLMS, CEN, CENELEC, PLT, Zigbee, M-BUS, KNX, etc.
M2M Device and Gateway Domain

FIGURE M2M HLSA.

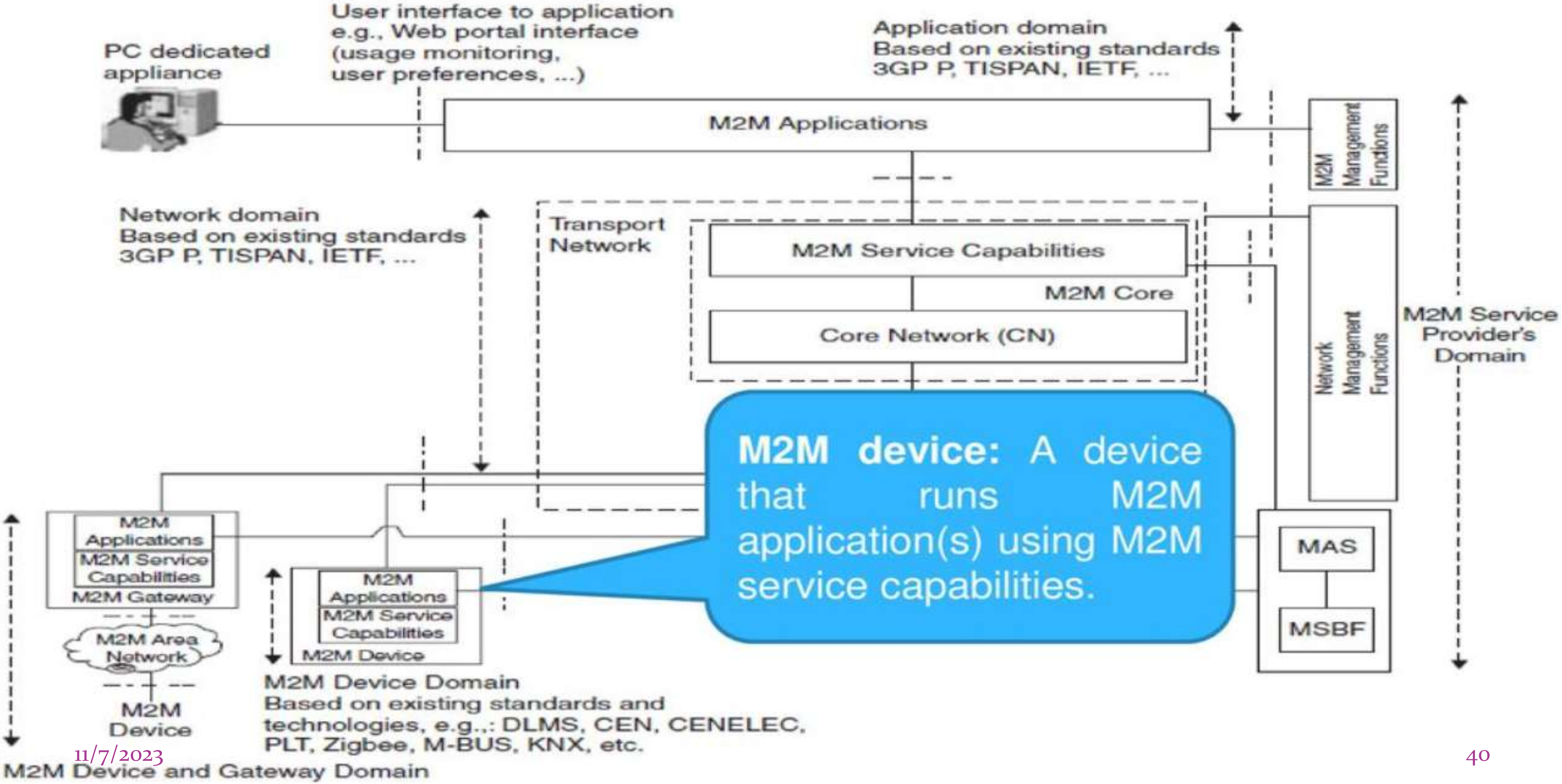
IOT- Framework

- The device and gateway domain is composed of the following elements:

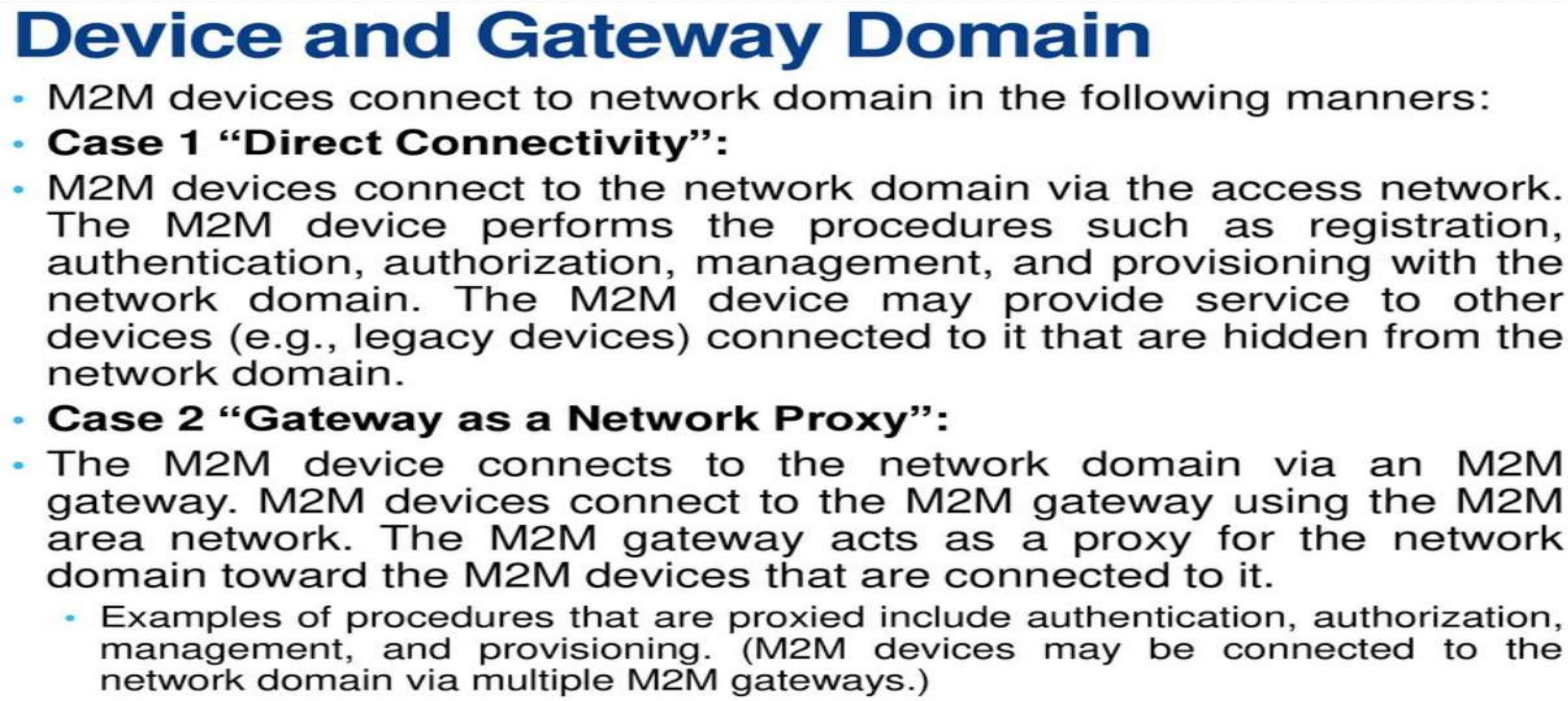


1. M2M device
   a. Direct Connectivity
   b. Gateway as a Network Proxy
2. M2M area network
3. M2M gateway

Device and Gateway Domain


PC dedicated appliance
User interface to application e.g., Web portal interface (usage monitoring, user preferences, ...)
Application domain Based on existing standards 3GP P, TISPAN, IETF, ...
M2M Applications
M2M Management Functions
Network domain Based on existing standards 3GP P, TISPAN, IETF, ...
Transport Network
M2M Service Capabilities
M2M Core
Core Network (CN)
Network Management Functions
M2M Service Provider's Domain
M2M Applications
M2M Service Capabilities
M2M Gateway
M2M Area Network
M2M Device
M2M Applications
M2M Service Capabilities
M2M Device
MAS
MSBF
M2M Device Domain Based on existing standards and technologies, e.g.,: DLMS, CEN, CENELEC, PLT, Zigbee, M-BUS, KNX, etc.
M2M Device and Gateway Domain

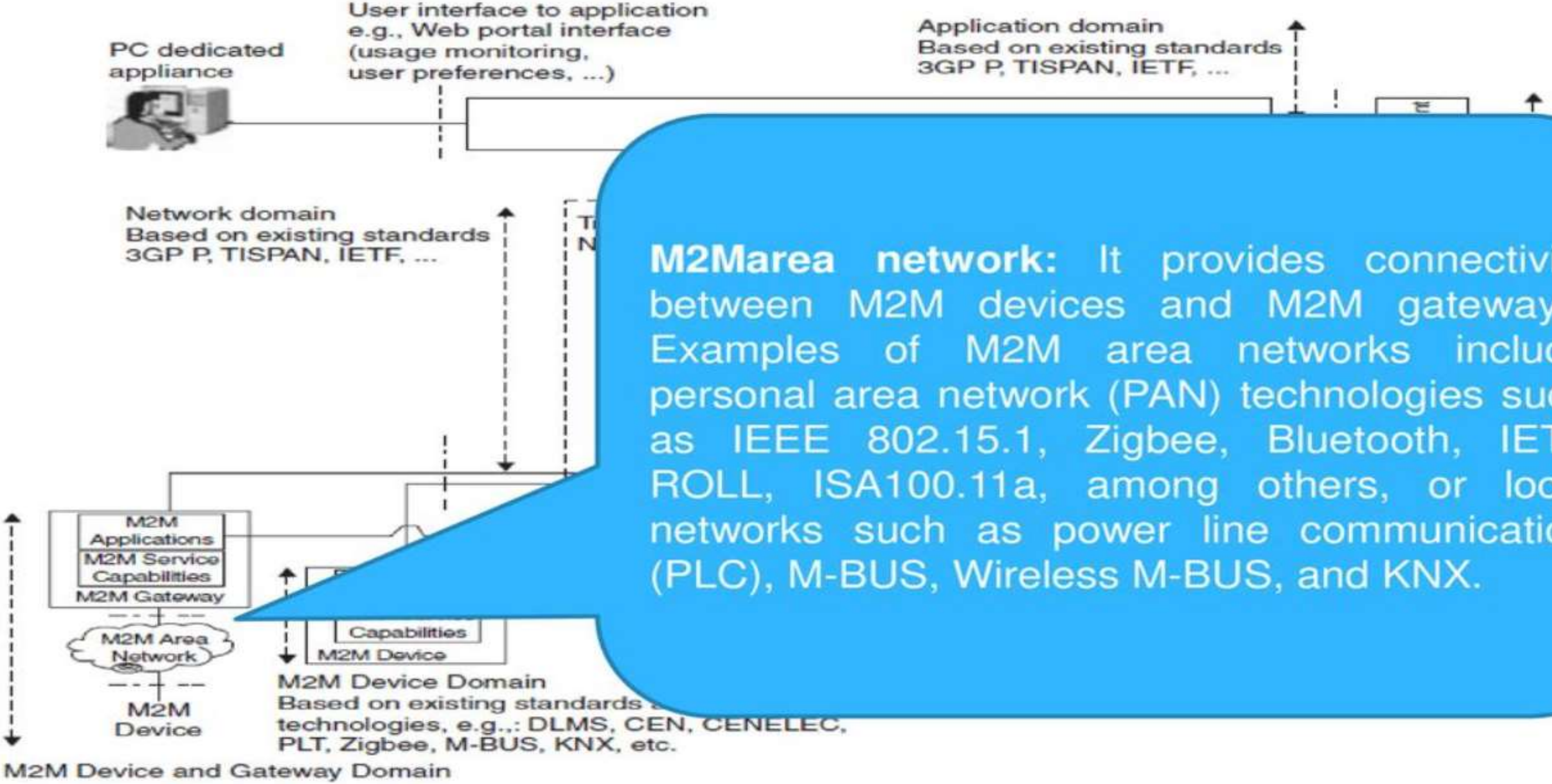
M2M device: A device that runs M2M application(s) using M2M service capabilities.



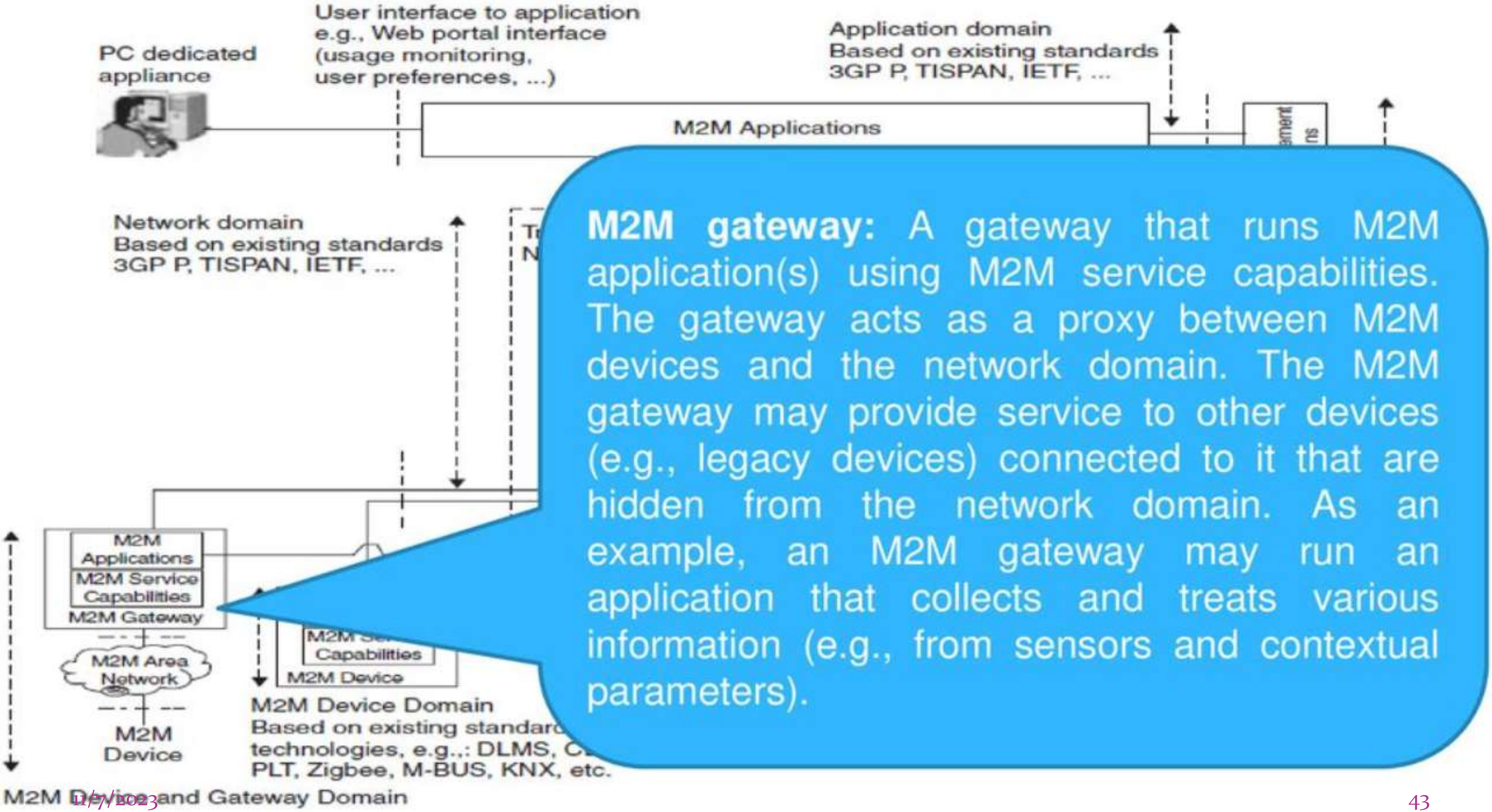
Device and Gateway Domain

- M2M devices connect to network domain in the following manners:
- **Case 1 "Direct Connectivity":**
- M2M devices connect to the network domain via the access network. The M2M device performs the procedures such as registration, authentication, authorization, management, and provisioning with the network domain. The M2M device may provide service to other devices (e.g., legacy devices) connected to it that are hidden from the network domain.
- **Case 2 "Gateway as a Network Proxy":**
- The M2M device connects to the network domain via an M2M gateway. M2M devices connect to the M2M gateway using the M2M area network. The M2M gateway acts as a proxy for the network domain toward the M2M devices that are connected to it.
  - Examples of procedures that are proxied include authentication, authorization, management, and provisioning. (M2M devices may be connected to the network domain via multiple M2M gateways.)

Device and Gateway Domain

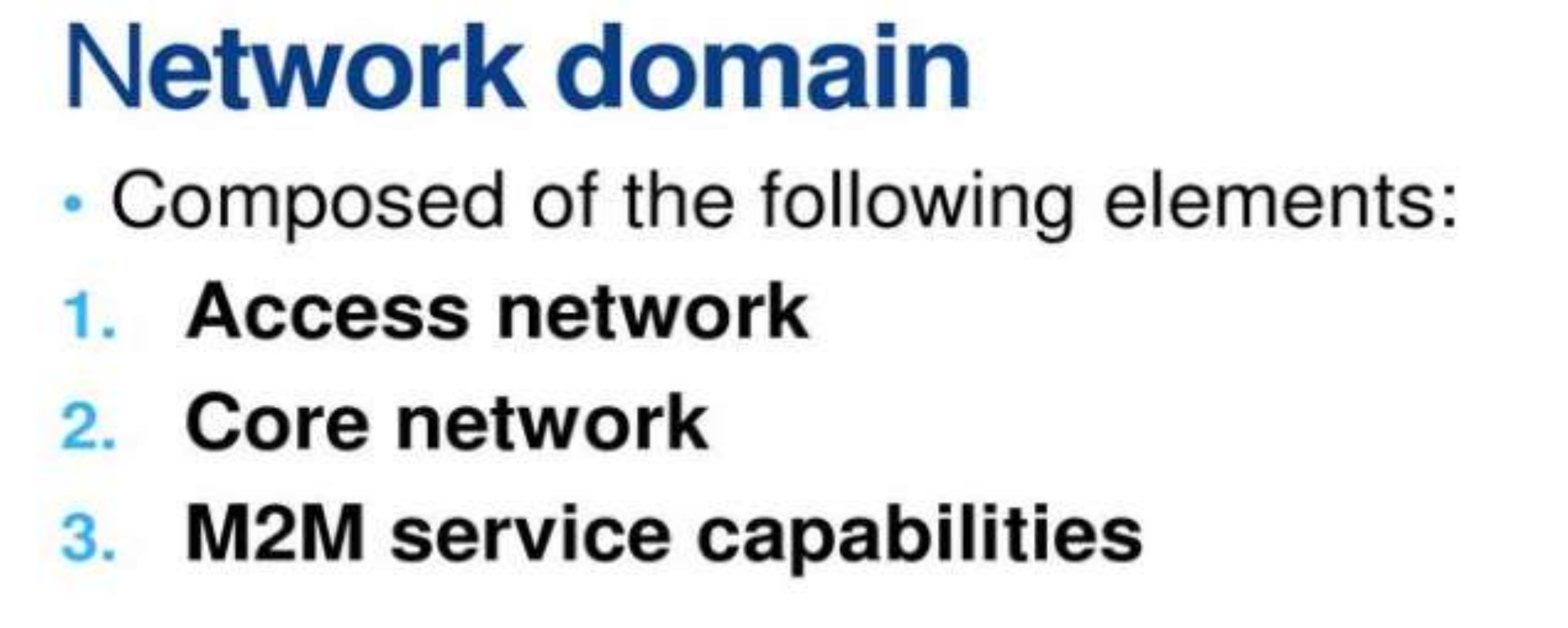


M2Marea network: It provides connectivity between M2M devices and M2M gateways. Examples of M2M area networks include personal area network (PAN) technologies such as IEEE 802.15.1, Zigbee, Bluetooth, IETF ROLL, ISA100.11a, among others, or local networks such as power line communication (PLC), M-BUS, Wireless M-BUS, and KNX.


PC dedicated appliance
User interface to application e.g., Web portal interface (usage monitoring, user preferences, ...)
Application domain Based on existing standards 3GP P, TISPAN, IETF, ...
M2M Applications
Network domain Based on existing standards 3GP P, TISPAN, IETF, ...
M2M Applications
M2M Service Capabilities
M2M Gateway
M2M Area Network
M2M Device
Capabilities
M2M Device
M2M Device Domain Based on existing standard technologies, e.g.,: DLMS, C PLT, Zigbee, M-BUS, KNX, etc.
M2M Device and Gateway Domain

M2M gateway: A gateway that runs M2M application(s) using M2M service capabilities. The gateway acts as a proxy between M2M devices and the network domain. The M2M gateway may provide service to other devices (e.g., legacy devices) connected to it that are hidden from the network domain. As an example, an M2M gateway may run an application that collects and treats various information (e.g., from sensors and contextual parameters).

IOT Frameworks- Network Domain

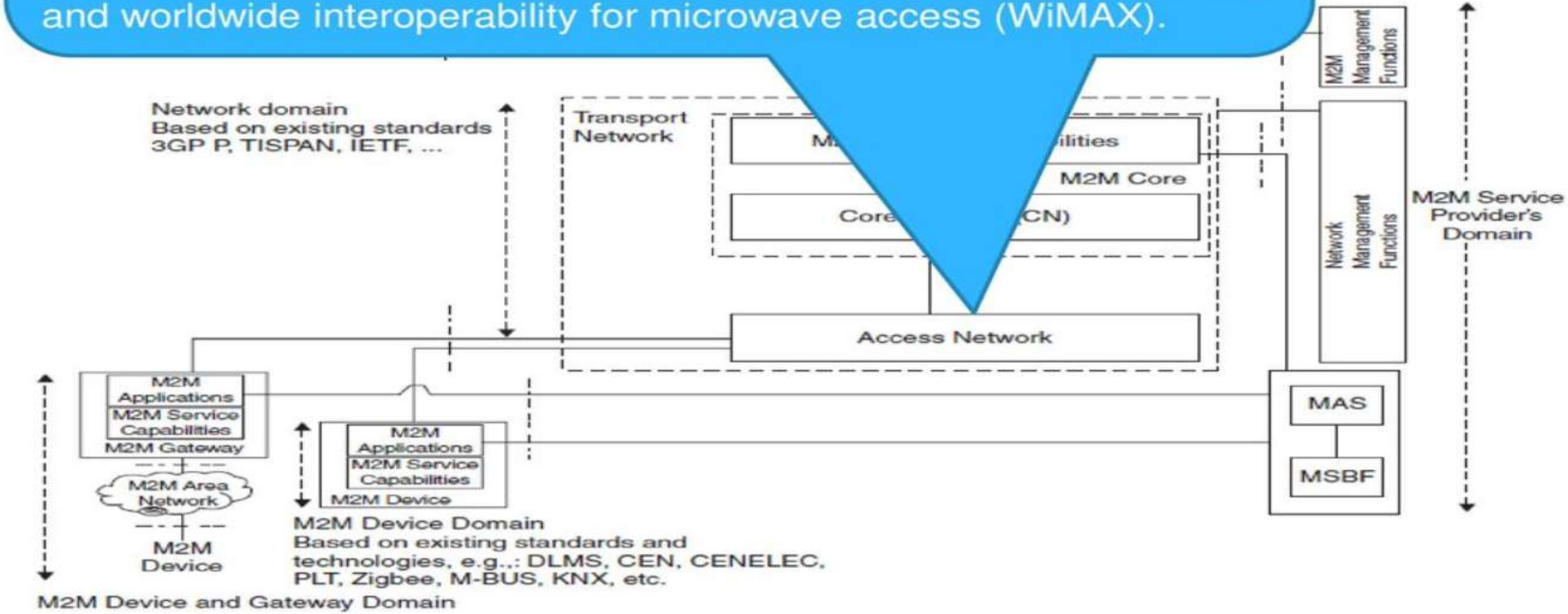


Network domain

- Composed of the following elements:

1. **Access network**
2. **Core network**
3. **M2M service capabilities**

Access network: A network that allows the M2M device and gateway domain to communicate with the core network. Access networks include (but are not limited to) digital subscriber line (xDSL), hybrid fiber coax (HFC), satellite, GSM/EDGE radio access network (GERAN), UMTS terrestrial radio access network (UTRAN), evolved UMTS terrestrial radio access network (eUTRAN), W-LAN, and worldwide interoperability for microwave access (WiMAX).



Core network: A network that provides the following capabilities (different core networks offer different features sets):

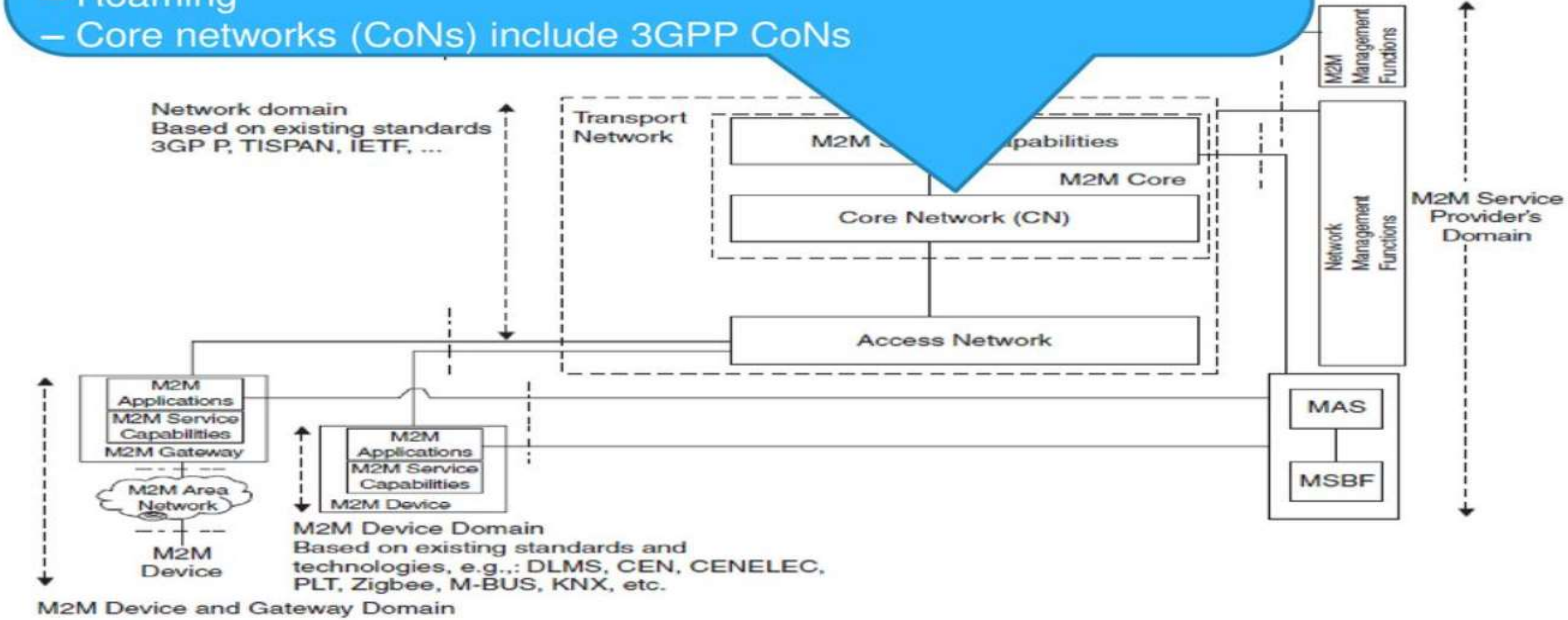
– IP connectivity at a minimum, and possibly other connectivity means

– Service and network control functions

– Interconnection (with other networks)

– Roaming

– Core networks (CoNs) include 3GPP CoNs



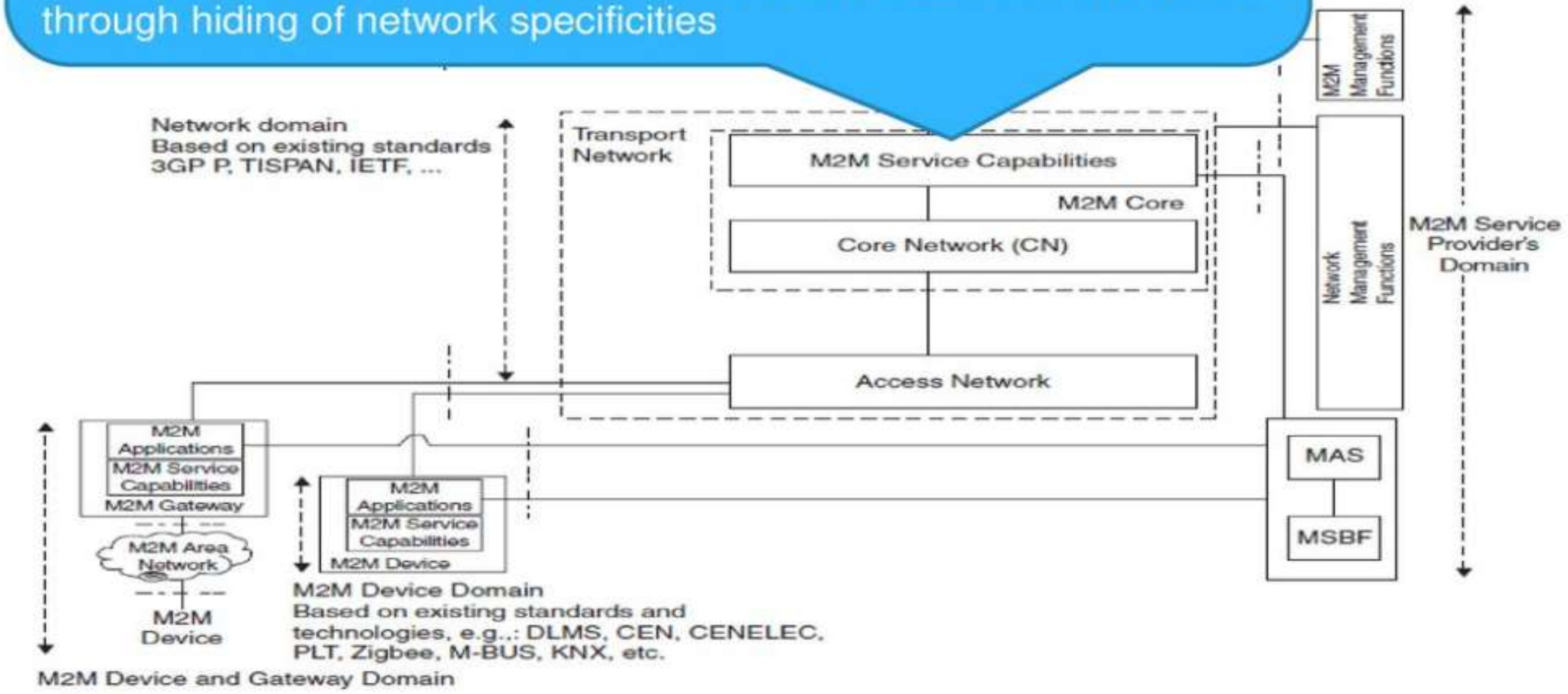
M2M service capabilities:

– Provide M2M functions that are to be shared by different applications

– Expose functions through a set of open interfaces

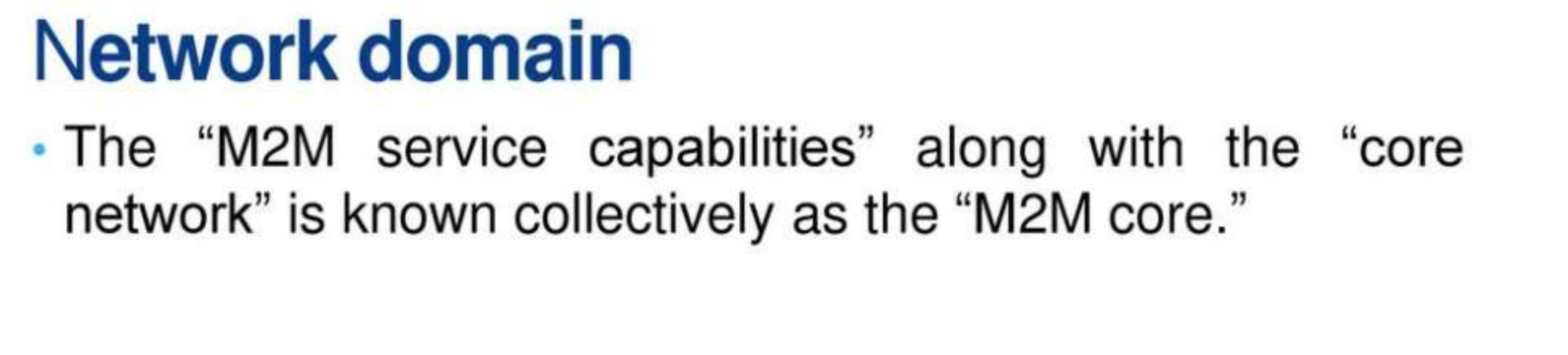
– Use CoN functionalities

– Simplify and optimize application development and deployment through hiding of network specificities



IOT Framework-Network domain

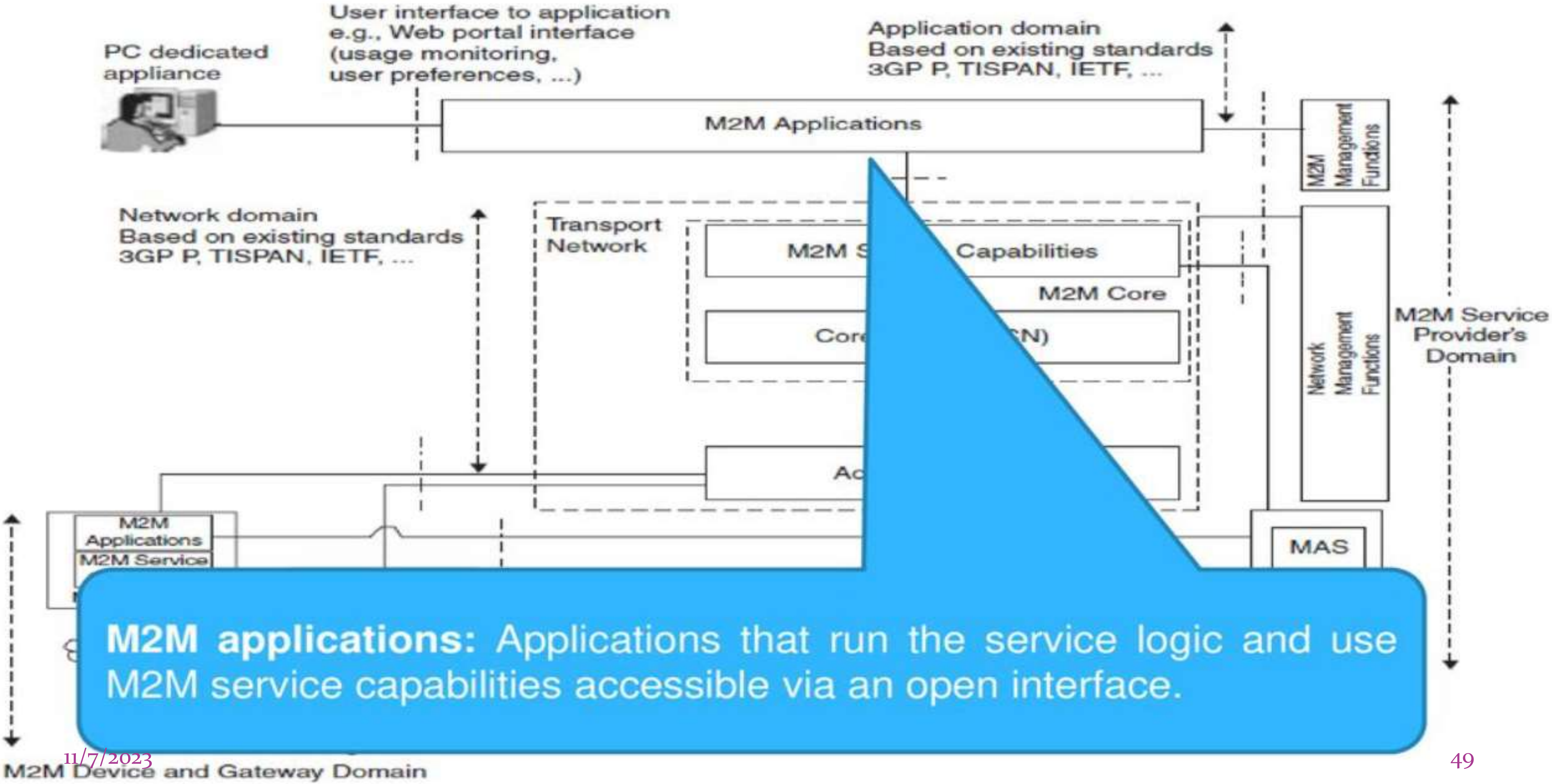
I
O
T



Network domain

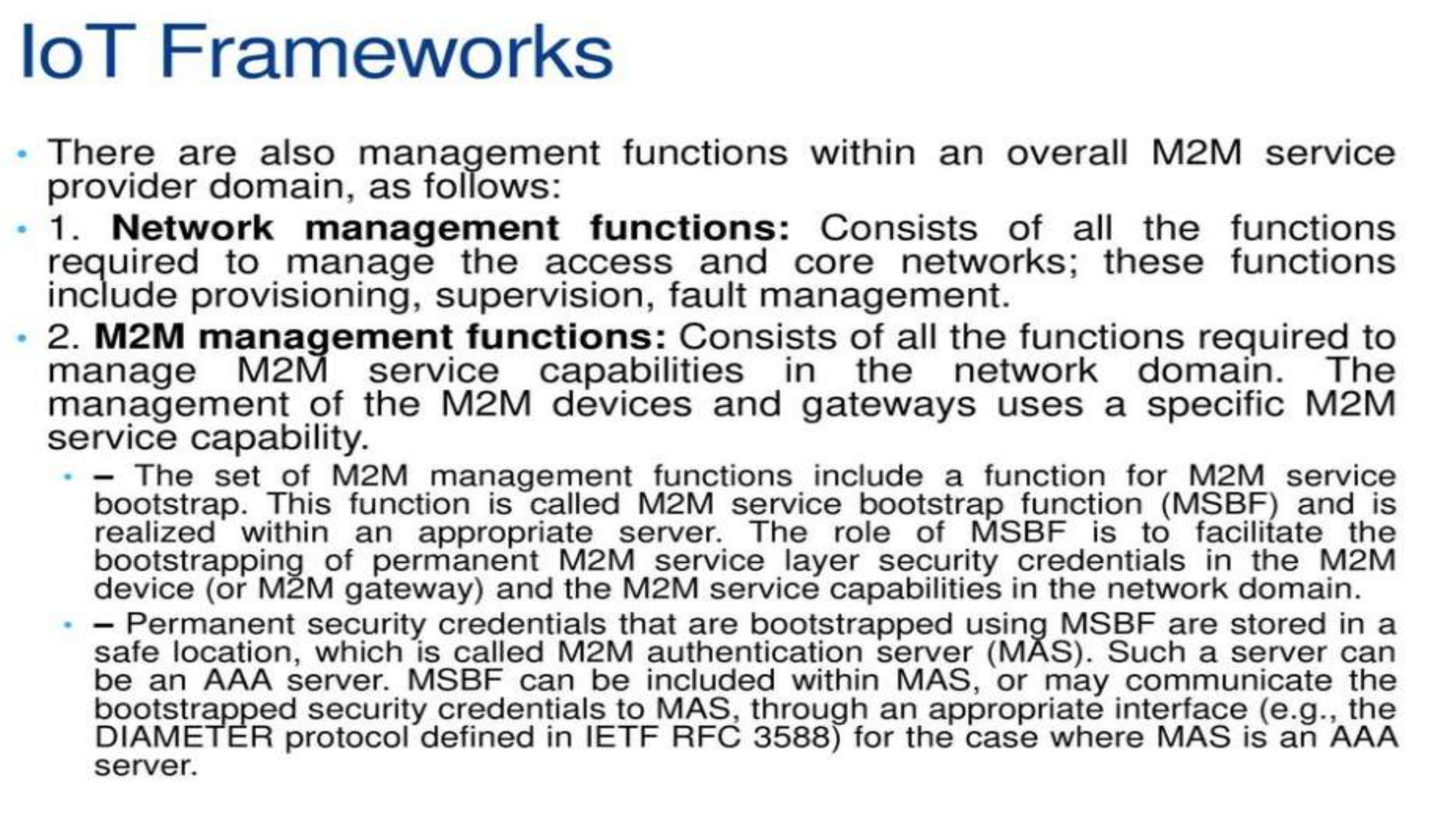
- The “M2M service capabilities” along with the “core network” is known collectively as the “M2M core.”

Applications domain



PC dedicated appliance
User interface to application e.g., Web portal interface (usage monitoring, user preferences, ...)
Application domain Based on existing standards 3GP P, TISPAN, IETF, ...
M2M Applications
M2M Management Functions
Network domain Based on existing standards 3GP P, TISPAN, IETF, ...
Transport Network
M2M S Capabilities
M2M Core
Core N)
Ac
Network Management Functions
M2M Service Provider's Domain
M2M Applications
M2M Service
MAS
M2M Device and Gateway Domain

M2M applications: Applications that run the service logic and use M2M service capabilities accessible via an open interface.



IoT Frameworks

- There are also management functions within an overall M2M service provider domain, as follows:
- 1. **Network management functions:** Consists of all the functions required to manage the access and core networks; these functions include provisioning, supervision, fault management.
- 2. **M2M management functions:** Consists of all the functions required to manage M2M service capabilities in the network domain. The management of the M2M devices and gateways uses a specific M2M service capability.
  - – The set of M2M management functions include a function for M2M service bootstrap. This function is called M2M service bootstrap function (MSBF) and is realized within an appropriate server. The role of MSBF is to facilitate the bootstrapping of permanent M2M service layer security credentials in the M2M device (or M2M gateway) and the M2M service capabilities in the network domain.
  - – Permanent security credentials that are bootstrapped using MSBF are stored in a safe location, which is called M2M authentication server (MAS). Such a server can be an AAA server. MSBF can be included within MAS, or may communicate the bootstrapped security credentials to MAS, through an appropriate interface (e.g., the DIAMETER protocol defined in IETF RFC 3588) for the case where MAS is an AAA server.

Basic Nodal Capabilities

- Remote device generally needs to have a basic protocol stack
- Basic protocol stack supports as minimum local connectivity and networking connectivity
- In addition, some higher layer application support protocols are generally needed, with varying degrees of computational/functional sophistication.

BASIC NODAL CABABILITIES

- In the context of the Internet of Things (IoT), a "node" refers to an individual device or object that is part of the IoT network. Each node typically has its own set of capabilities that enable it to interact with other nodes, collect data, and communicate with the central system or cloud. Here are some basic nodal capabilities in IoT:

Sensing: Nodes in IoT are equipped with sensors that can detect various physical parameters such as temperature, humidity, pressure, light, motion, proximity, and more. These sensors enable the node to perceive its environment and gather data.

Processing: IoT nodes often have some processing capabilities, which allow them to perform computations on the data collected from the sensors. Depending on the complexity of the node, it may have microcontrollers or microprocessors to process the data locally.

Connectivity: Connectivity is a fundamental capability of IoT nodes. Nodes are designed to communicate with other nodes or with a central server or cloud platform over the internet. Connectivity options may include Wi-Fi, Bluetooth, Zigbee, LoRaWAN, cellular networks, or Ethernet.

Data Transmission: IoT nodes can transmit the collected data to other nodes or to the central server/cloud. The data transmission may occur in real-time or at scheduled intervals, depending on the application requirements.

Data Storage: Some IoT nodes have onboard memory or storage capabilities to store data locally, especially when connectivity to the central server is intermittent or unreliable. Local storage ensures that data is not lost during temporary communication disruptions.

Energy Management: Since many IoT nodes are battery-powered or have limited power sources, energy management is critical. Nodes are designed to be energy-efficient, optimizing power consumption to prolong battery life or reduce the need for frequent recharging.

Actuation: Certain IoT nodes, often referred to as "actuators," have the ability to perform actions based on the data they receive or on commands from the central system. Actuators can control physical devices, appliances, or processes in the real world.

Security: Basic security capabilities are essential in IoT nodes to protect the data they collect and transmit. This may include encryption, authentication mechanisms, and secure communication protocols to prevent unauthorized access.

Location Awareness: Some IoT nodes may be equipped with GPS or other location tracking technologies to determine their precise location. Location awareness is crucial for applications such as asset tracking and geolocation-based services.

Over-the-Air Updates: Many IoT nodes support firmware updates over the air (OTA updates). This allows the devices to receive software updates and bug fixes remotely, without the need for physical intervention.

These basic capabilities form the foundation of IoT nodes, enabling them to operate within an IoT network, gather data, and contribute to the overall functionality of the IoT system. Specific IoT applications may require additional specialized capabilities based on their unique use cases and requirements.

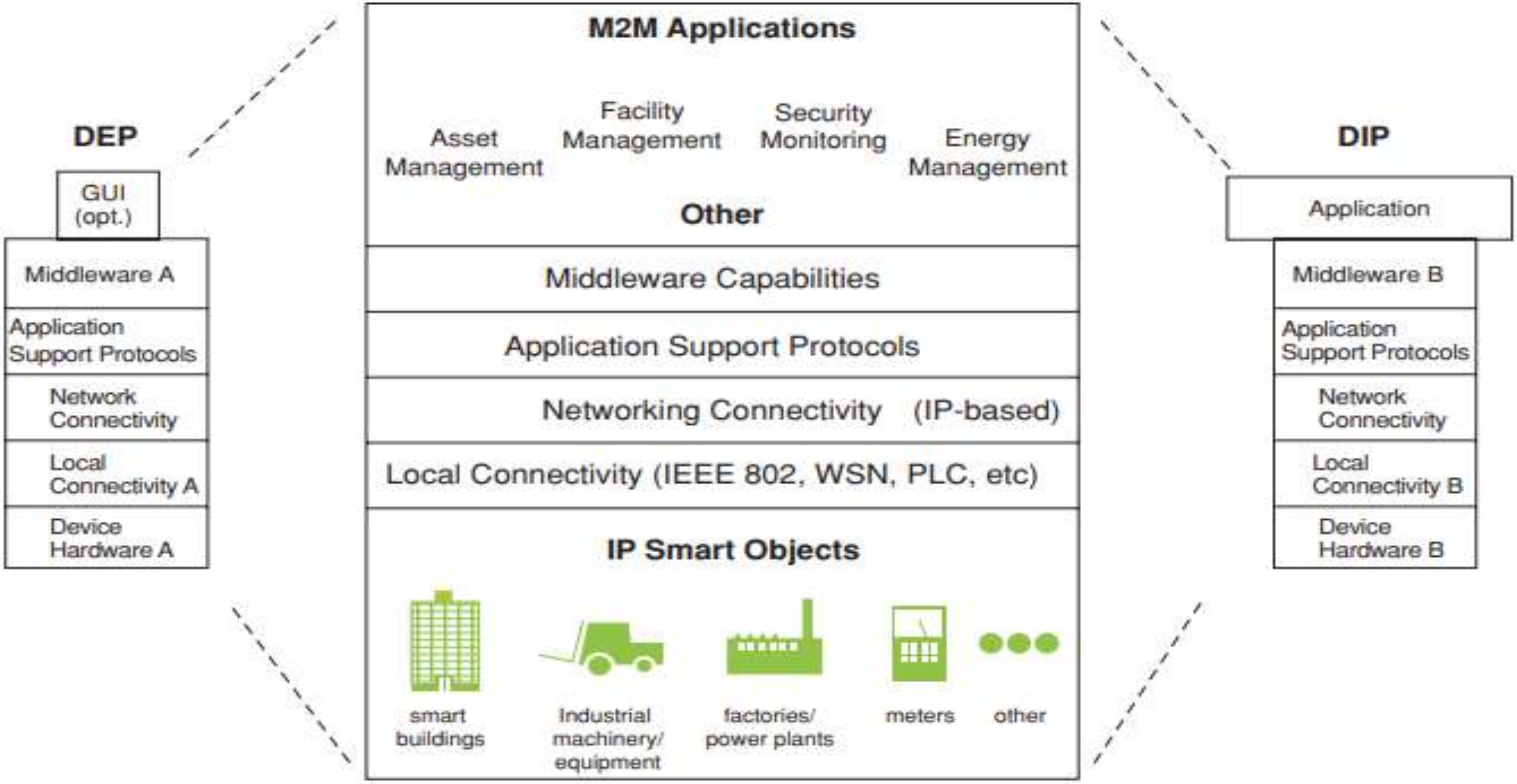


FIGURE Protocol stack, general view.

Conti…

- IoT devices may have capability differences, such as but not limited to the following: (Refer the above figure)
  - maximum transmission unit (MTU) differences,
  - simplified versus full-blown web protocol stack (COAP/UDP versus HTTP/TCP),
  - single stack versus dual stack,
  - sleep schedule,
  - security protocols,
  - processing and communication bandwidth.

Conti…

- Distributed control/M2M typically entails continuously changing variables to control the behavior of an application. Typical requirements include the following capabilities:
- Retransmission
    - Network recovers from packet loss or informs application
    - Recovery is immediate: on the order of RTTs, not seconds
- Network independent of MAC/PHY
- Scale
    - Thousands of nodes
    - Multiple link speed

Conti…

- Thousands of nodes
- Multiple link speeds
- Multicast
  - Throughout network
  - Reliable (positive Ack)
- Duplicate suppression
- Emergency messages
  - Routed and/or queued around other traffic
  - Other traffic slushed as delivered
- Routine traffic delivered in sequence
- Separate timers by peer/message

Conti...

- Polling of nodes
  - Sequential
  - Independent of responses
- Paradigm supports peer-to-peer
- Not everything is client/server
- Capabilities
- Discover nodes
- Discover node capabilities
- Deliver multisegment records (files)
- Exchange of multisegment records
- Network and application versioning
- Simple publish/subscribe parsers
- Security
  - Strong encryption
  - Mutual authentication
  - Protection against record/playback attacks
  - Suite B ciphers

IOT

The applications domain is composed of the following elements:

- □ M2M applications: Applications that run the service logic and use M2M service capabilities accessible via an open interface.

Other management functions within an overall M2M service provider domain, as follows:

- □ Network management functions - functions required to manage the access and core networks
- □ M2M management functions - required to manage M2M service capabilities in the network domain

The network domain is composed of the following elements:

- 1. Access network: A network that allows the M2M device and gateway domain to communicate with the core network.

Core network: A network that provides the following :

- – IP connectivity - minimum and possibly other connectivity means
- – Service and network control functions
- – Interconnection (with other networks)

Open Source IoT Frameworks

1)KAA IoT

2) ZETTA

3) GE PREDIX

4) ThingSpeak

5) DeviceHive

6) Distributed Services Architecture

Physical Design of IoT

Things/Devices

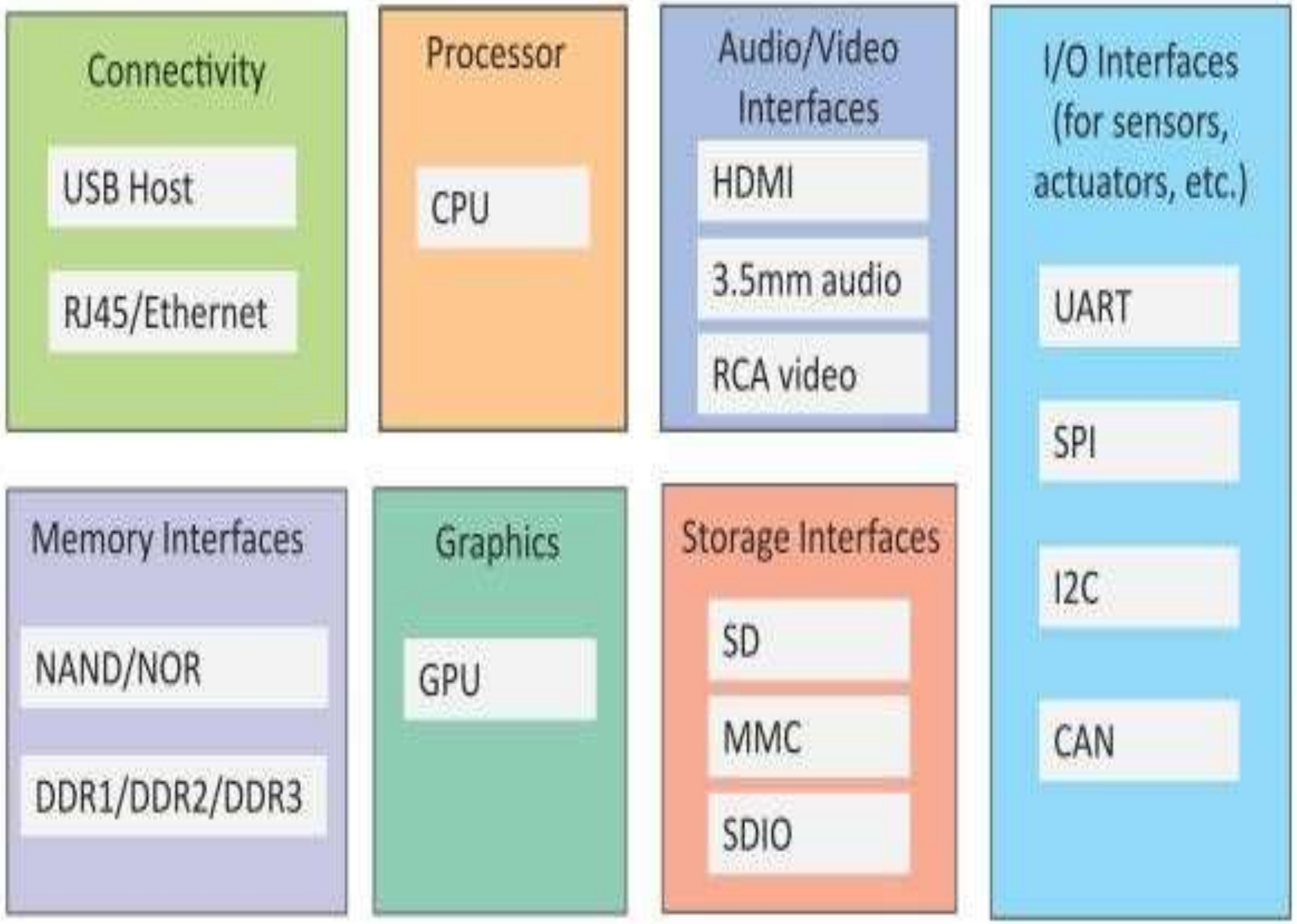
- It used to build a connection, process data, provide interfaces, storage, and graphics interfaces in an IoT system.

Physical Design of IoT

- The "Things" in IoT usually refers to IoT devices which have unique identities and can perform remote sensing, actuating and monitoring capabilities.
- IoT devices can:
  - Exchange data with other connected devices and applications (directly or indirectly), or
  - Collect data from other devices and process the data locally or
  - Send the data to centralized servers or cloud-based application back-ends for processing the data, or
  - Perform some tasks locally and other tasks within the IoT infrastructure, based on temporal and space constraints

Generic block diagram of an IoT Device

- An IoT device may consist of several interfaces for connections to other devices, both wired and wireless.
  - I/O interfaces for sensors
  - Interfaces for Internet connectivity
  - Memory and storage interfaces
  - Audio/video interfaces.



Generic block diagram of an IoT Device

Connectivity

- Devices like USB host and ETHERNET are used for connectivity between the devices and server.

Processor

- A processor like a CPU and other units are used to process the data. these data are further used to improve the decision quality of an IoT system.

Audio/Video Interfaces

- An interface like HDMI and RCA devices is used to record audio and videos in a system.

Input/Output interface

- To giving input and output signals to sensors, and actuators we use things like UART, SPI, CAN, etc.

Storage Interfaces

- Things like SD, MMC, SDIO are used to store the data generated from an IoT device.

Memory Interfaces

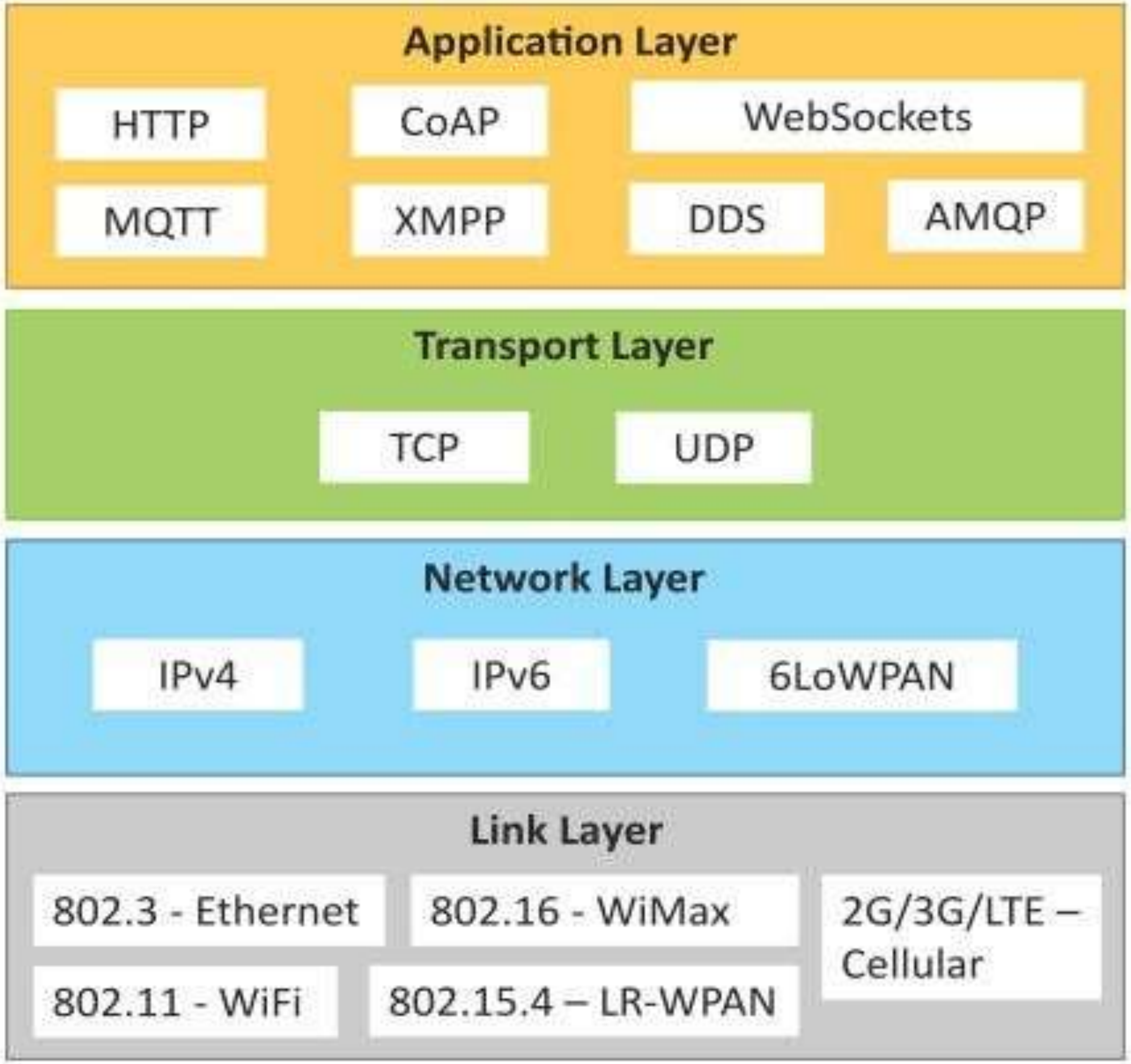
- Memory interface are usually chipsets, which makes computations / processing along with processor.

Graphics

- GPU is a processor that is specially-designed to handle intensive graphics rendering tasks.

IOT Protocols

- Link Layer
  - 802.3 – Ethernet
  - 802.11 – WiFi
  - 802.16 – WiMax
  - 802.15.4 – LR-WPAN
  - 2G/3G/4G
- Network/Internet Layer
  - IPv4
  - IPv6
  - 6LoWPAN
- Transport Layer
  - TCP
  - UDP
- Application Layer
  - HTTP
  - CoAP
  - WebSocket
  - MQTT
  - XMPP
  - DDS
  - AMQP



Application Layer

HTTP | CoAP | WebSockets

MQTT | XMPP | DDS | AMQP

Transport Layer

TCP | UDP

Network Layer

IPv4 | IPv6 | 6LoWPAN

Link Layer

802.3 - Ethernet | 802.16 - WiMax | 2G/3G/LTE – Cellular

802.11 - WiFi | 802.15.4 – LR-WPAN

IOT Protocols

Application Layer protocol

- In this layer, protocols define how the data can be sent over the network with the lower layer protocols using the application interface. These protocols including HTTP, WebSocket, XMPP, MQTT, DDS, and AMQP protocols.

HTTP

- Hypertext transfer protocol is a protocol that presents in an application layer for transmitting media documents.
- It is used to communicate between web browsers and servers.
- It makes a request to a server and then waits till it receives a response and in between the request server does not keep any data between two requests.

CoAP

- Constrained Application Protocol (CoAP) used for M2M applications, meant for constrained environments, networks and devices.
- It is used for web transfer protocol and uses request –response model
- It runs on UDP instead of TCP

XMPP

- Extensible Messaging and Presence Protocol(XMPP) is a protocol for real-time communication and streaming XML data between network entities. It provides messaging, gaming, presence, data syndication, multi-party chat, voice/video calls.

WebSocket

- This protocol enables two-way communication (Full duplex) between a client and server. It is based on TCP

MQTT

- Message Queue Telemetry Transport, is light weight messaging protocol based on publish-subscribe model. It is well suited for constrained environment with limited resources.

AMQP

- Advanced Message Queuing Protocol is a protocol for message-oriented middleware environments. It supports both point-to-point and published/subscriber model.

DDS

- Data Distribution Service (DDS) is a data-centric middleware standard for M2M communication. It uses publish-subscriber model. DDS provides QoS control and configurable reliability.

Transport Layer

- This layer is used to control the flow of data segments and handle the error control. also, these layer protocols provide end-to-end message transfer capability independent of the underlying network.

TCP

- The transmission control protocol is a protocol that defines how to establish and maintain a network that can exchange data in a proper manner using the internet protocol.

UDP

- a user datagram protocol is a part of internet protocol called the connectionless protocol. this protocol not required to establish the connection to transfer data.

Network Layer

- This layer is used to send datagrams/datastream from the source network to the destination network. we use IPv4 and IPv6 protocols as a host identification that transfers data in packets.

IPv4

- This is a protocol address that is a unique and numerical label assigned to each device connected with the network. an IP address performs two main functions host and location addressing. IPv4 is an IP address that is 32 bit long.

IPv6

- It is a successor of IPv4 that uses 128 bits for an IP address. it is developed by the IETF task force to deal with the long-anticipated problems.

6LoWPAN

- IPv6 over Low power Wireless Personal Area Network , It brings IP protocol to the low power devices with limited processing capability. It operates in 2.4 GHz at 250 Kb/s transfer rate.

Link Layer

- Link-layer protocols are used to send data over the network's physical layer. it also determines how the packets are coded and signaled by the devices.

Ethernet

- IEEE 802.3 is a collection of wired ethernet standards for the link layer. It is a set of technologies and protocols that are used primarily in LANs. it defines the physical layer and the medium access control for wired ethernet networks.

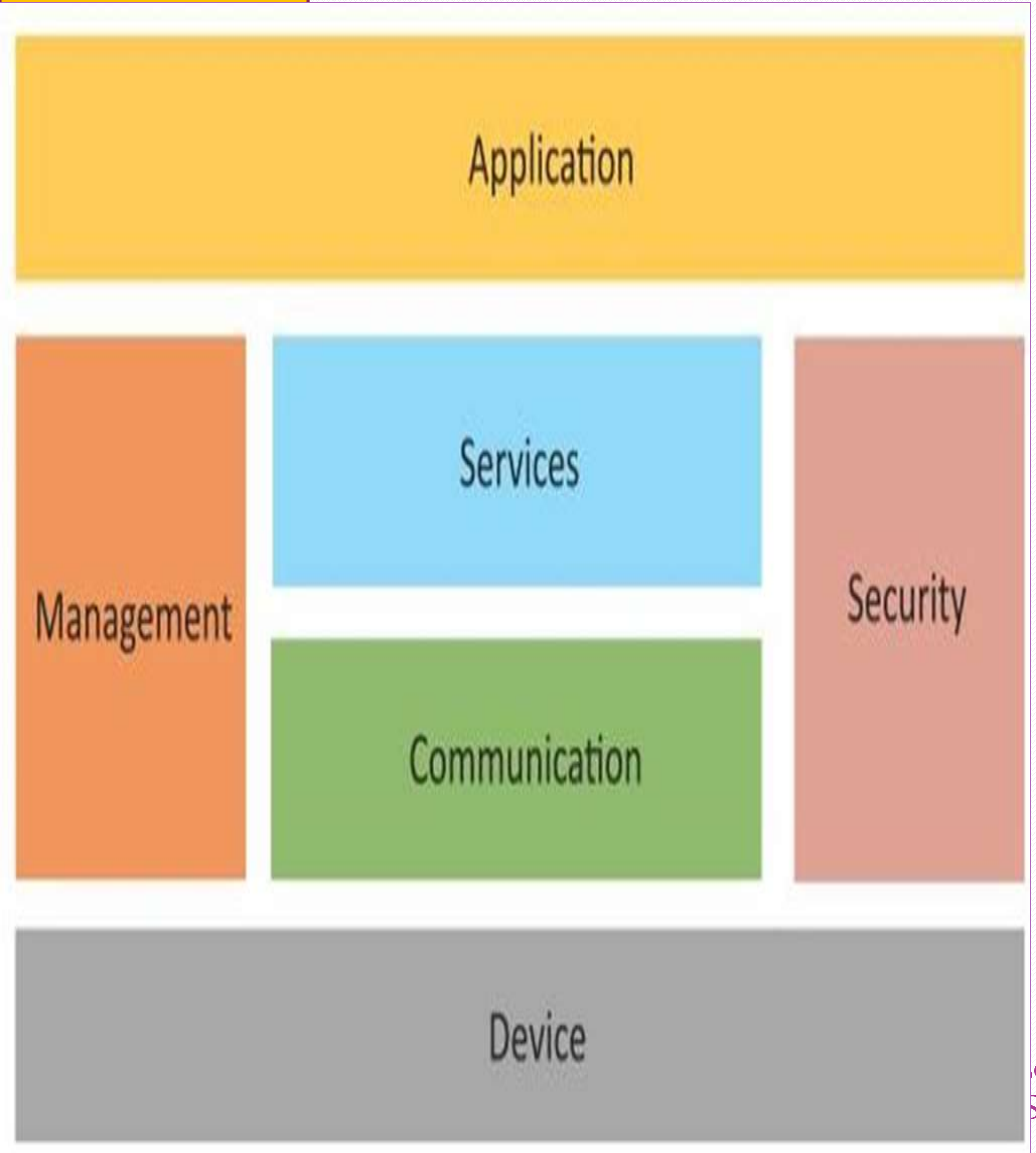
WiFi

- IEEE 802.11 is a collection of WLAN communication standards, including extensive description of link layer. It operates mostly in 2.4 Ghz/5 Ghz. Some Others operates at 60Ghz.

- WiMax: IEEE 802.16 (worldwide interoperability for microwave access) is a collection of wireless broadband standards, including extensive description for link layer. It provides data rates from 1.5 Mb/s to 1 Gb/s. The recent update 802.16m provides data rates of 100 Mbits/s for mobile stations and 1 Gbits/s for fixed stations.
- LR-WPAN: IEEE 802.15.4 is a collection of standards for low rate WPANs (LR-WPANs). These standards form the basis of specifications for higher level communication protocols such as ZigBee. These standard provides data rates from 40Kb/s 250 Kb/s.

- 2G/3G/4G-Mobile communication: This includes
  - 2G GSM and CDMA
  - 3G UMTS and CDMA2000
  - 4G including LTE

Logical Design of IoT



- Logical design of an IoT system refers to an abstract representation of the entities and processes without going into the low-level specifics of the implementation.

- An IoT system comprises of a number of functional blocks that provide the system the capabilities for identification, sensing, actuation, communication, and management.

Logical Design of the Internet of Things(IoT)

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

IoT Functional blocks

- An IoT system consists of a number of functional blocks like Devices, services, communication, security, and application that provide the capability for sensing, actuation, identification, communication, and management.

IoT communication APIs

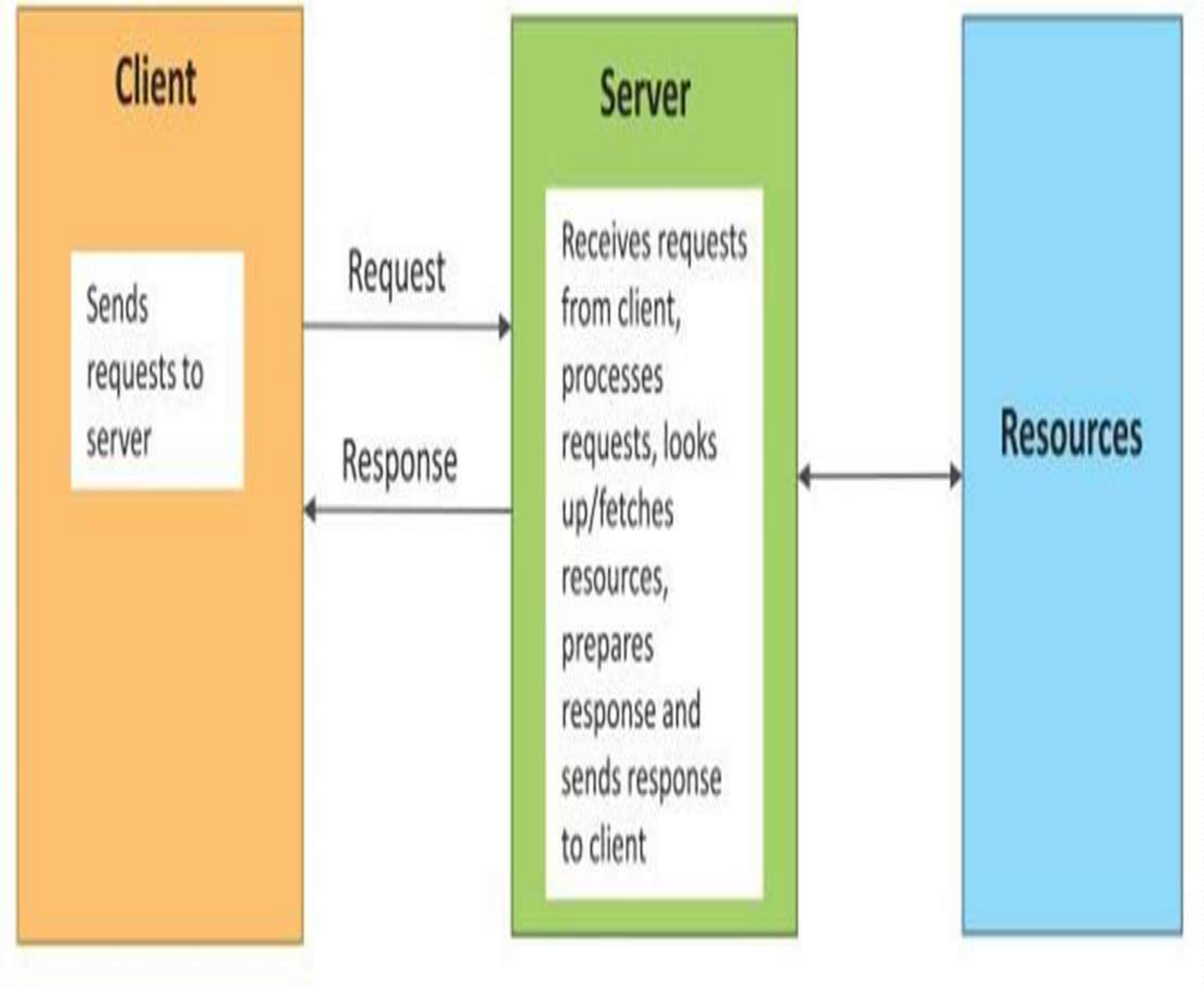
These APIs like REST and WebSocket are used to communicate between the server and system in IoT

REST-based communication APIs

- **Client-server**
- **Stateless**
- **Cacheable**

WebSocket-based communication API

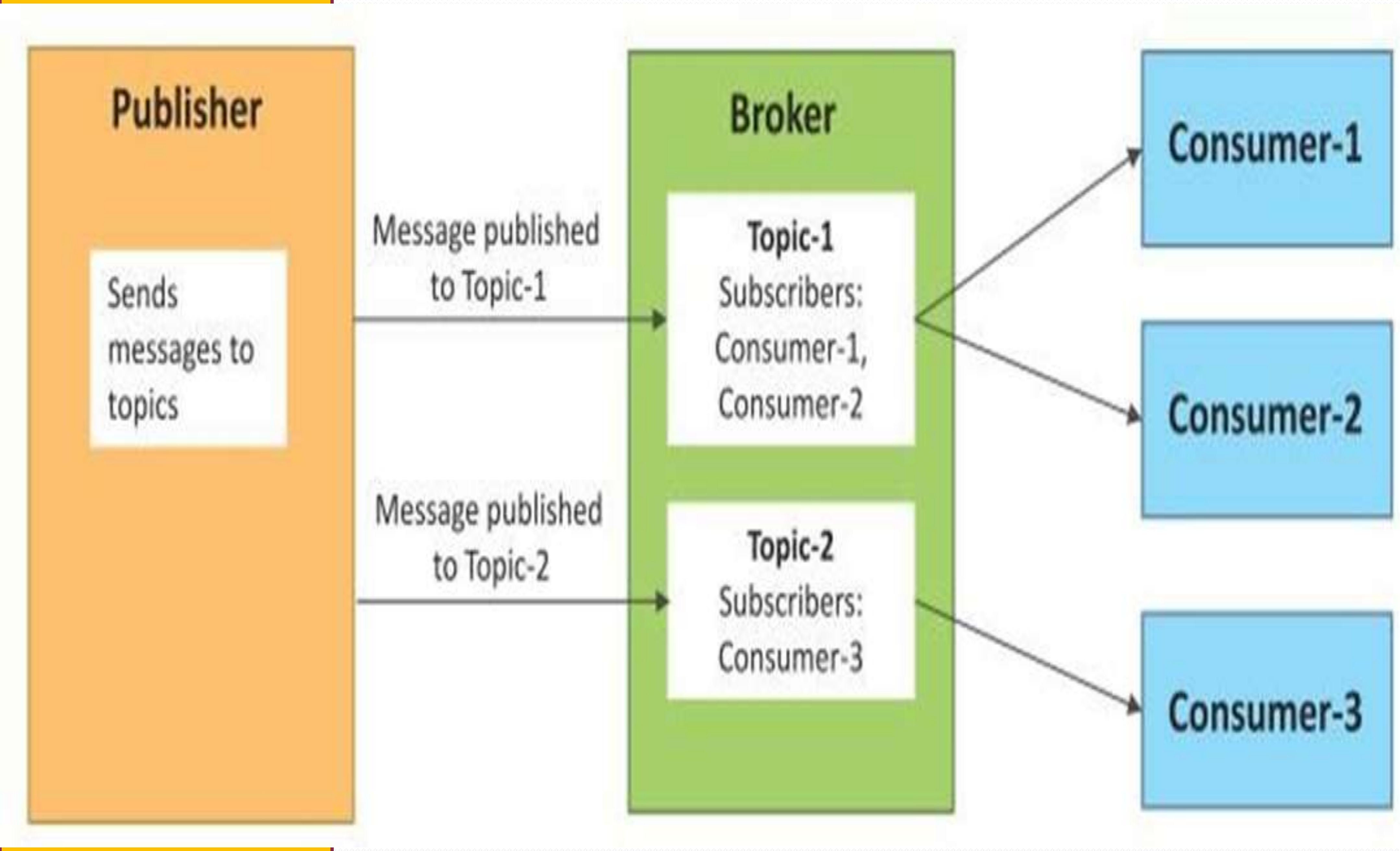
Request-Response communication model


Client
Sends requests to server
Request
Response
Server
Receives requests from client, processes requests, looks up/fetches resources, prepares response and sends response to client
Resources

- Request-Response is a communication model in which the client sends requests to the server and the server responds to the requests.

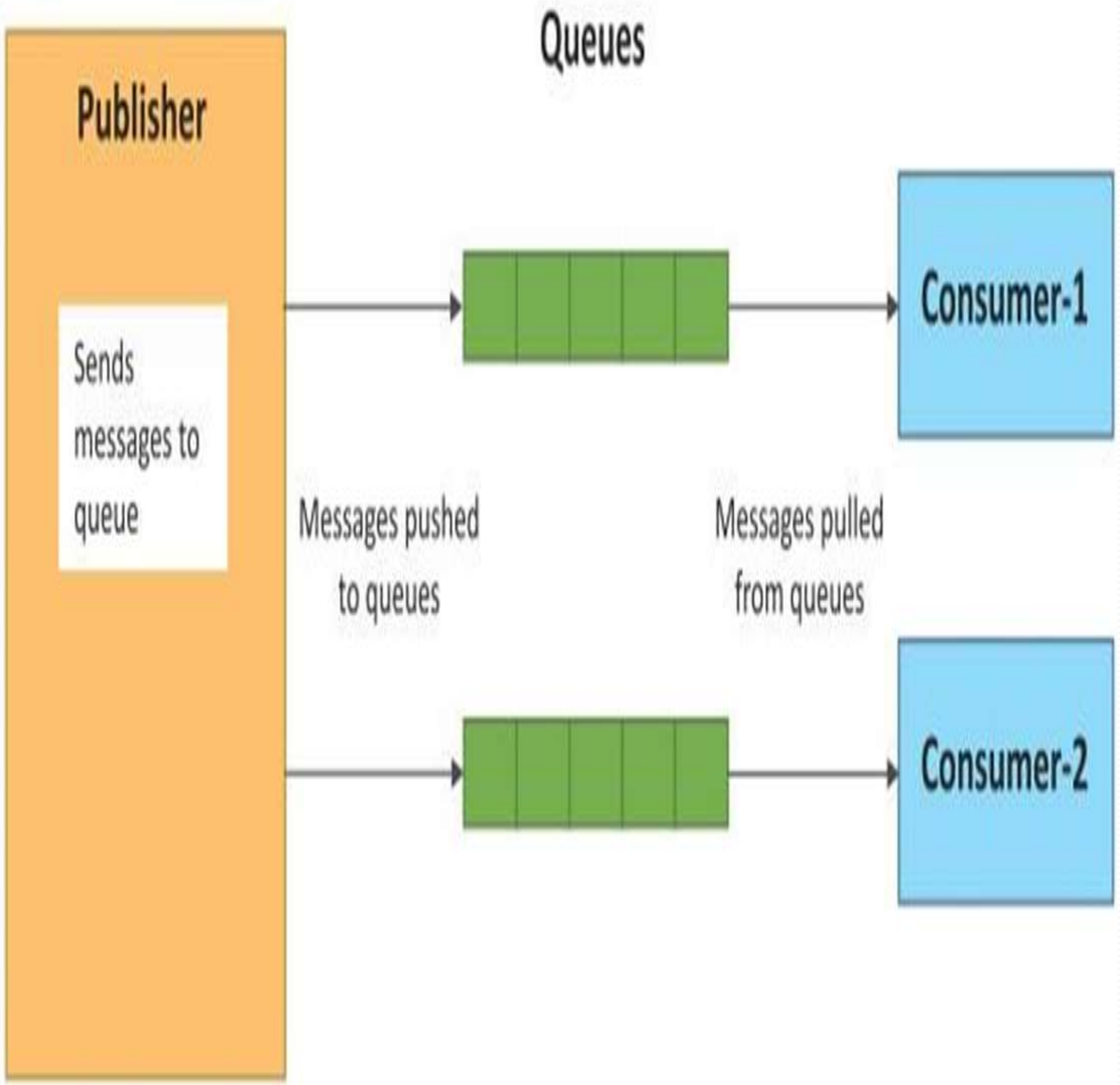
- When the server receives a request, it decides how to respond, fetches the data, retrieves resource representations,prepares the response, and then sends the response to the client.

Publish-Subscribe communication model



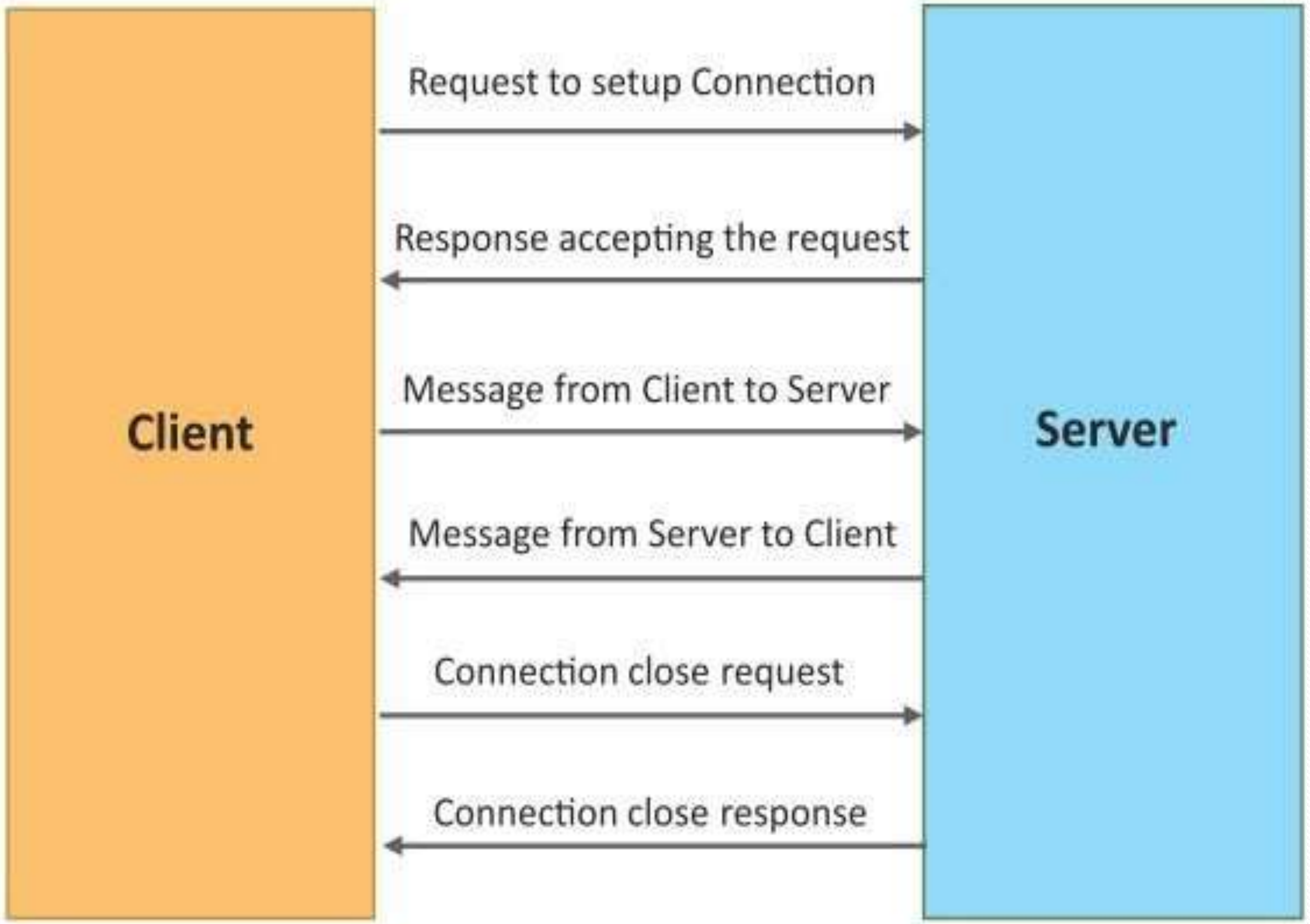
- Publish-Subscribe is a communication model that involves publishers, brokers and consumers.
- Publishers are the source of data. Publishers send the data to the topics which are managed by the broker. Publishers are not aware of the consumers.
- Consumers subscribe to the topics which are managed by the broker.
- When the broker receives data for a topic from the publisher, it sends the data to all the subscribed consumers.

Push-Pull communication model



- Push-Pull is a communication model in which the data producers push the data to queues and the consumers pull the data from the queues. Producers do not need to be aware of the consumers.
- Queues help in decoupling the messaging between the producers and consumers.
- Queues also act as a buffer which helps in situations when there is a mismatch between the rate at which the producers push data and the rate at which the consumers pull data.

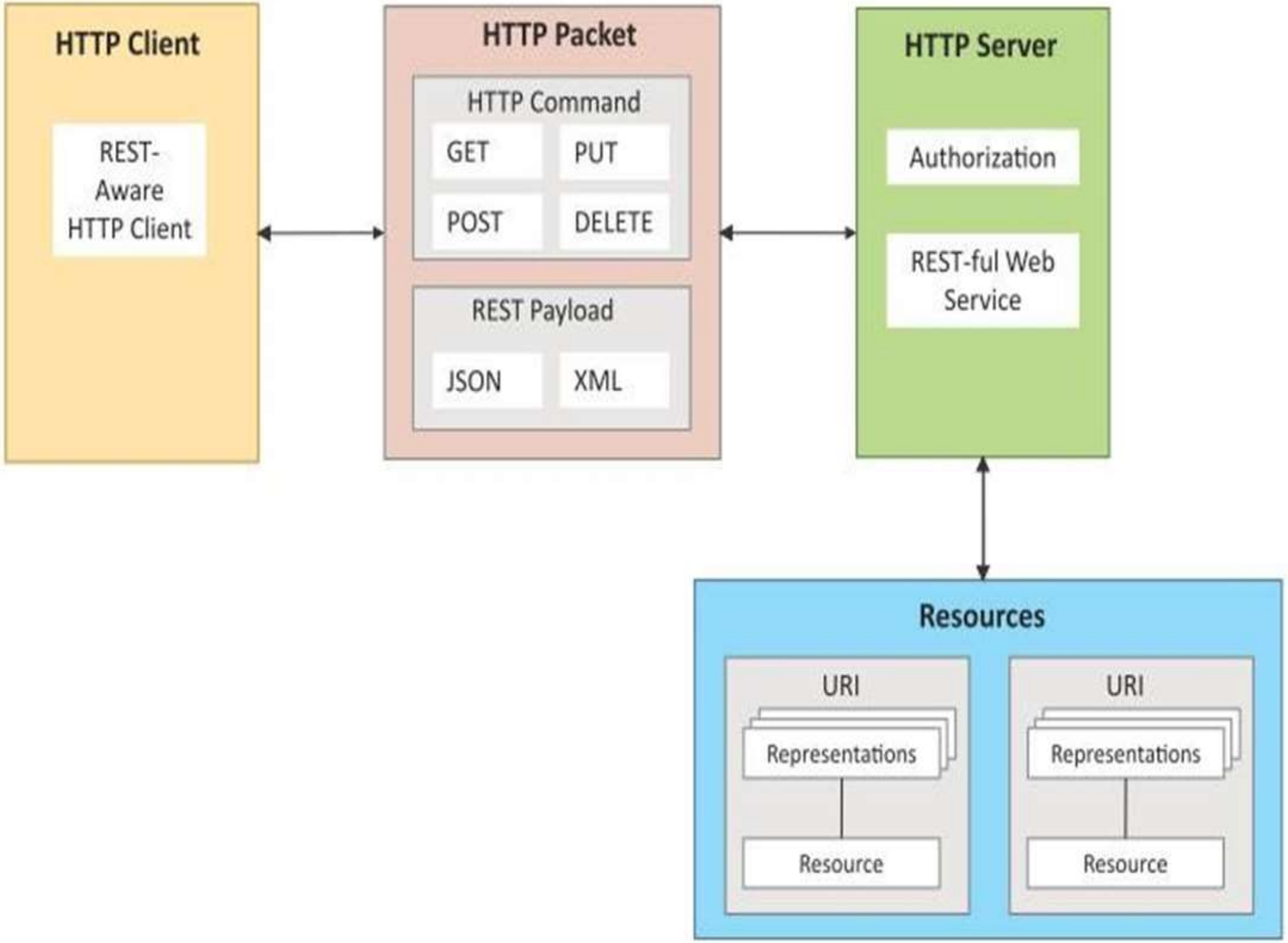
Exclusive Pair communication model



Client
Request to setup Connection
Response accepting the request
Message from Client to Server
Message from Server to Client
Connection close request
Connection close response
Server

- Exclusive Pair is a bidirectional, fully duplex communication model that uses a persistent connection between the client and server.
- Once the 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.

REST-based Communication APIs

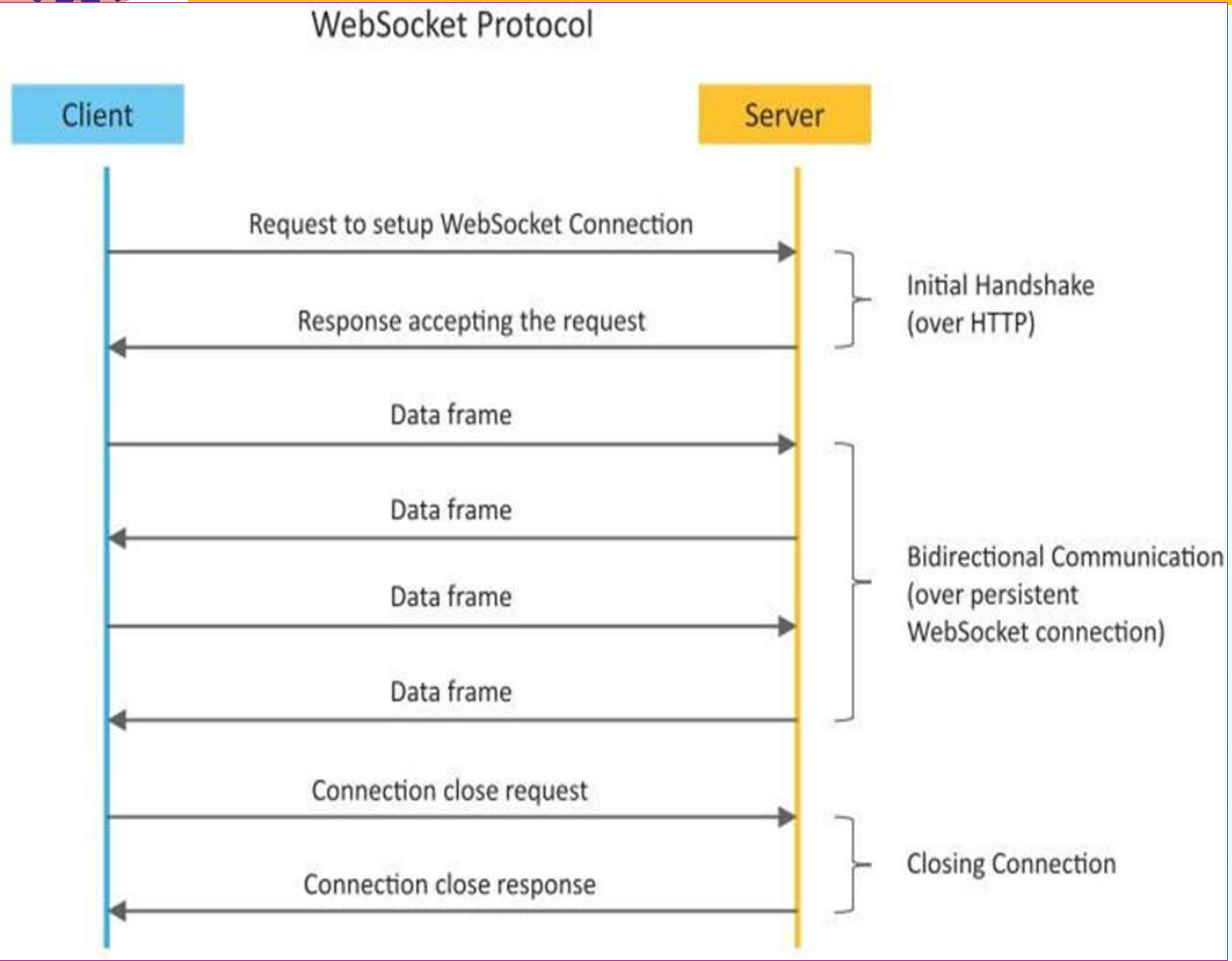


Representational State Transfer (REST) is a set of architectural principles by which you can design web services and web APIs that focus on a system's resources and how resource states are addressed and transferred.

REST APIs follow the request-response communication model.

The REST architectural constraints apply to the components, connectors, and data elements, within a distributed hypermedia system.

WebSocket-based Communication APIs


WebSocket Protocol
Client
Server
Request to setup WebSocket Connection
Response accepting the request
Initial Handshake (over HTTP)
Data frame
Data frame
Data frame
Data frame
Bidirectional Communication (over persistent WebSocket connection)
Connection close request
Connection close response
Closing Connection

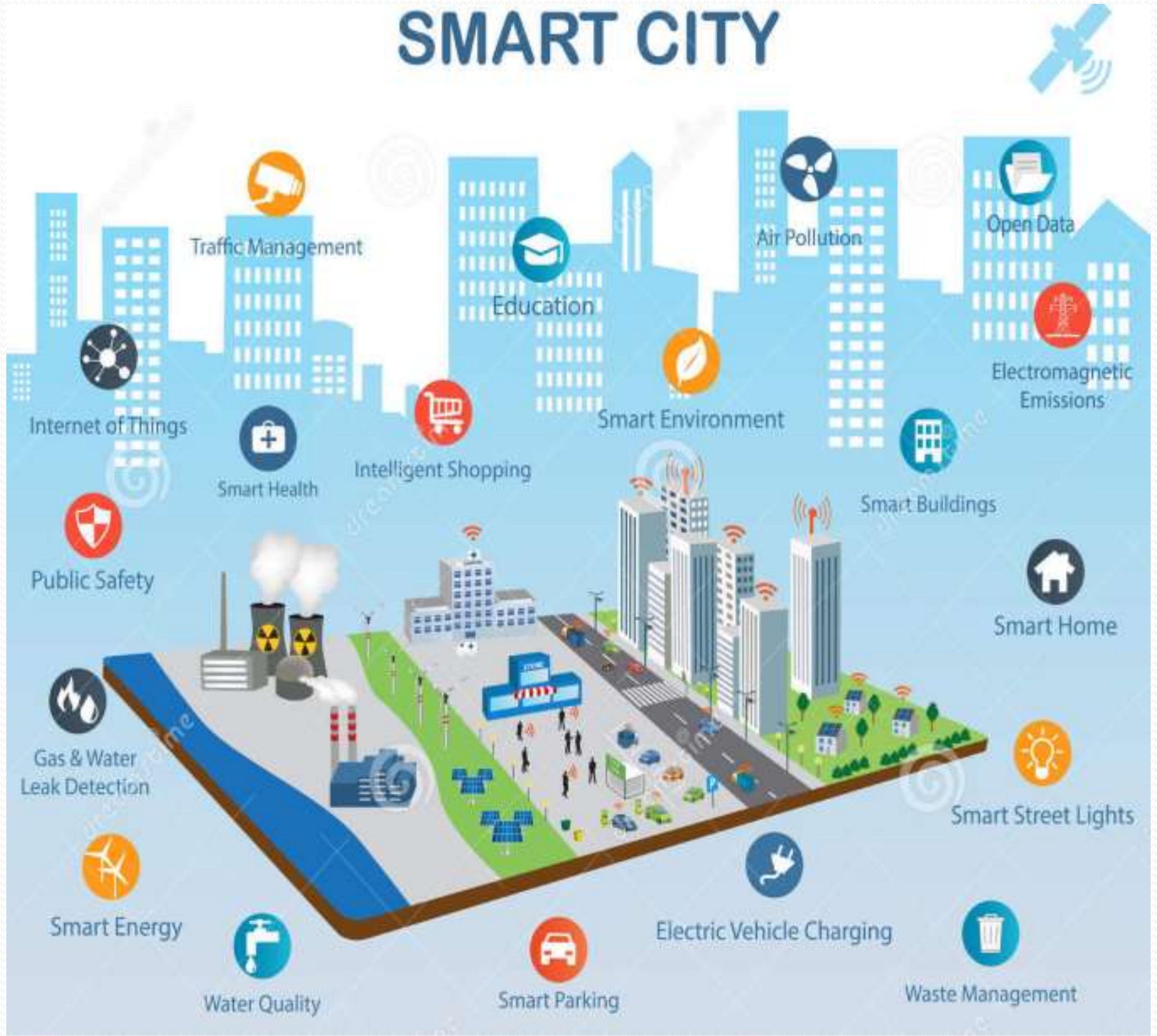
- WebSocket APIs allow bi-directional, full duplex communication between clients and servers.
- WebSocket APIs follow the exclusive pair communication model

I
O
T

City Automation

I
O
T

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.



Applications:- City Automation Automotive Applications - Home Automation

City Automation:

- **Smart Traffic Management**
- **Water Management**
- **Smart Waste Management**
- **Environmental Monitoring**
- **Public Safety and Surveillance**
- **Smart Street Lighting**
- **Smart Parking Solutions**

Automotive Applications:

- **Connected Cars**
- **Vehicle Telematics**:
- **Autonomous Vehicles**
- **In-Vehicle Entertainment and Services**
- **Remote Vehicle Control**

Home Automation:

- **Smart Lighting**
- **Smart Thermostats**
- **Home Security**
- **Smart Appliances**
- **Energy Management**
- **Voice Assistants**
- **Health Monitoring**
- Home security applications include but are not limited to:
- Door access phone
- Window locks
- Motion detector
- Smoke/ Gas / Fire alert
- Baby monitors

IoT applications in these areas have the potential to improve efficiency, convenience, and safety in our daily lives, making cities smarter, vehicles more connected, and homes more automated.

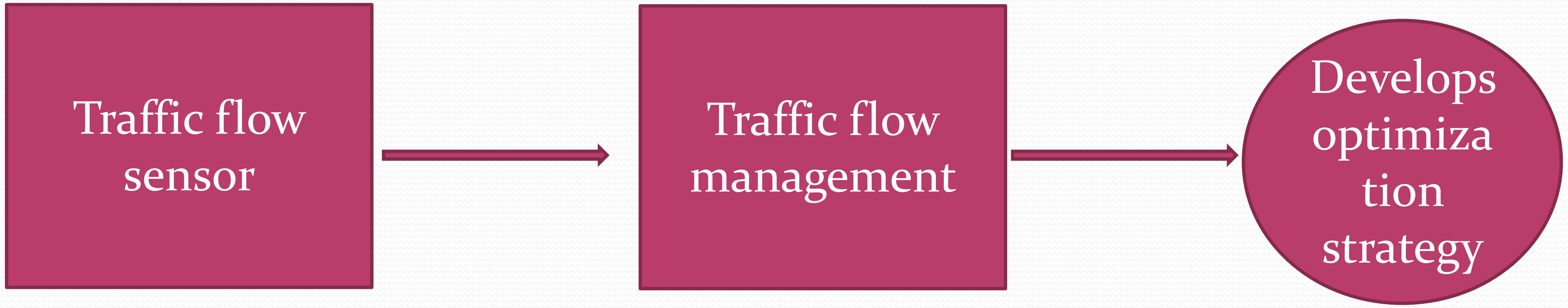
City Automation

Some applications in this domain include but are not limited to the following:

- Traffic flow management system in combination with dynamic traffic light control
- Street light control
- Passenger information system for public transportation
- Passive surveillance

Traffic flow management system in combination with dynamic traffic light control

- The flow of road traffic within cities depends on a number of factors such as the number of vehicles on the road, the time and the day, the current or expected weather, current traffic issues and accidents, as well as road construction work.



Traffic flow sensor

Traffic flow management

Develops optimization strategy

- The traffic flow management system can also interact with controllable traffic lights to extend or to reduce the green light period to increase the vehicle throughput on heavy used roads.
- Thus enabling cities to reduce fuel consumption, air pollution, congestions, and the time spent on the road

Street Light Control.

- Street lights are not required to shine at the same intensity to accomplish the intended safety goal.
- The intensity may depend on conditions such as moonlight or weather and movement of people.
- Adjusting the intensity helps to reduce the energy consumption and the expenditures incurred by a municipality.

Passenger Information System for Public Transportation.

- Public transportation vehicles, such as busses, subways, and commuter trains, operate on a schedule that may be impacted by external variables and, thus, have a degree of variability compared with a baseline formal schedule.
- Passengers need to know when their next connection is available; this information also allows passengers to select alternative connections in the case of longer delays.

- In this application, the current locations of the various public transport vehicles are provided to the central system that is able to match the current location with the forecasted location at each time or at specific checkpoints.
- Based on the time difference, the system is able to calculate the current delay and the expected arrival time at the upcoming stops.
- The vehicle location can be captured via checkpoints on the regular track or via GPS/general packet radio service (GPRS) tracking devices.

Environmental sensors

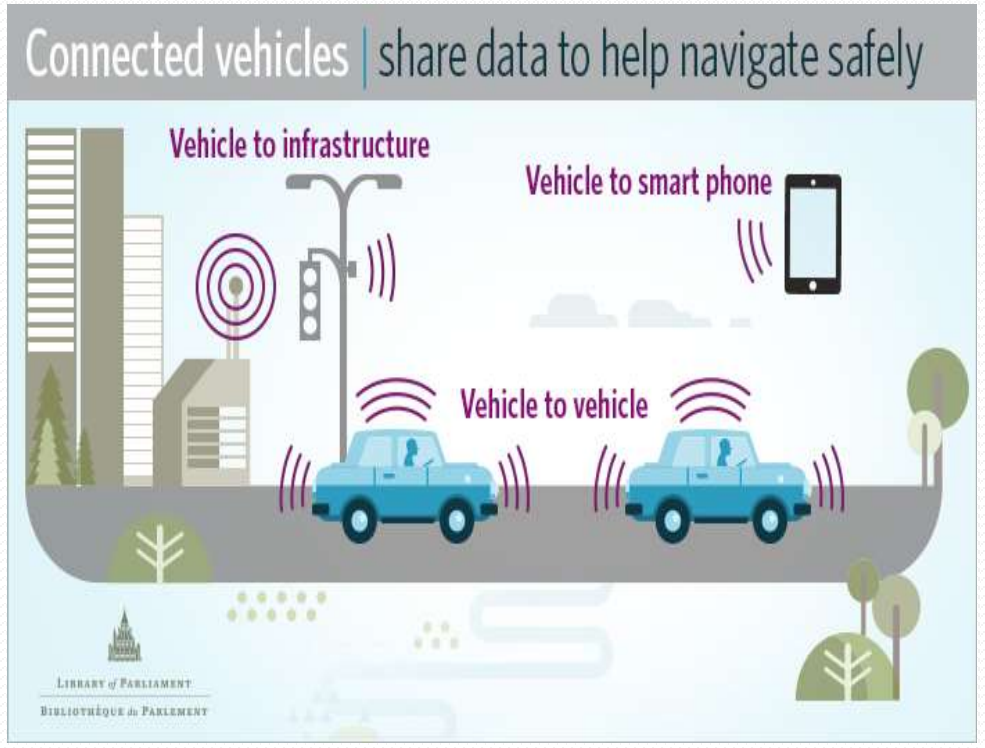
- thermal
- hygrometric
- anemometric
- sound
- gas
- particles
- light,
- other EM spectrum
- seismic

Activity sensors

- – pavement/roadway pressure
- – vehicle and pedestrian detection
- – parking space occupancy

AUTOMOTIVE APPLICATIONS

- M2M automotive and transportation applications focus on safety, security, connected navigation, and other vehicle services such as, but not limited to, insurance or road pricing, emergency assistance, fleet management, electric car charging management, and traffic optimization.


Connected vehicles | share data to help navigate safely
Vehicle to infrastructure
Vehicle to smart phone
Vehicle to vehicle
Library of Parliament
Bibliothèque du Parlement

- These applications typically entail IoT/M2M communication modules that are embedded into the car or the transportation equipment.
- Some of the technical challenges relate to mobility management and environmental hardware considerations.

Applications

- bCall (breakdown call)
- Stolen vehicle tracking (SVT)
- Remote diagnostics
  - Maintenance minder
  - Health check
  - Fault triggered
  - Enhanced bCall
  - Fleet management

- Vehicle-to-infrastructure communications
- Insurance services

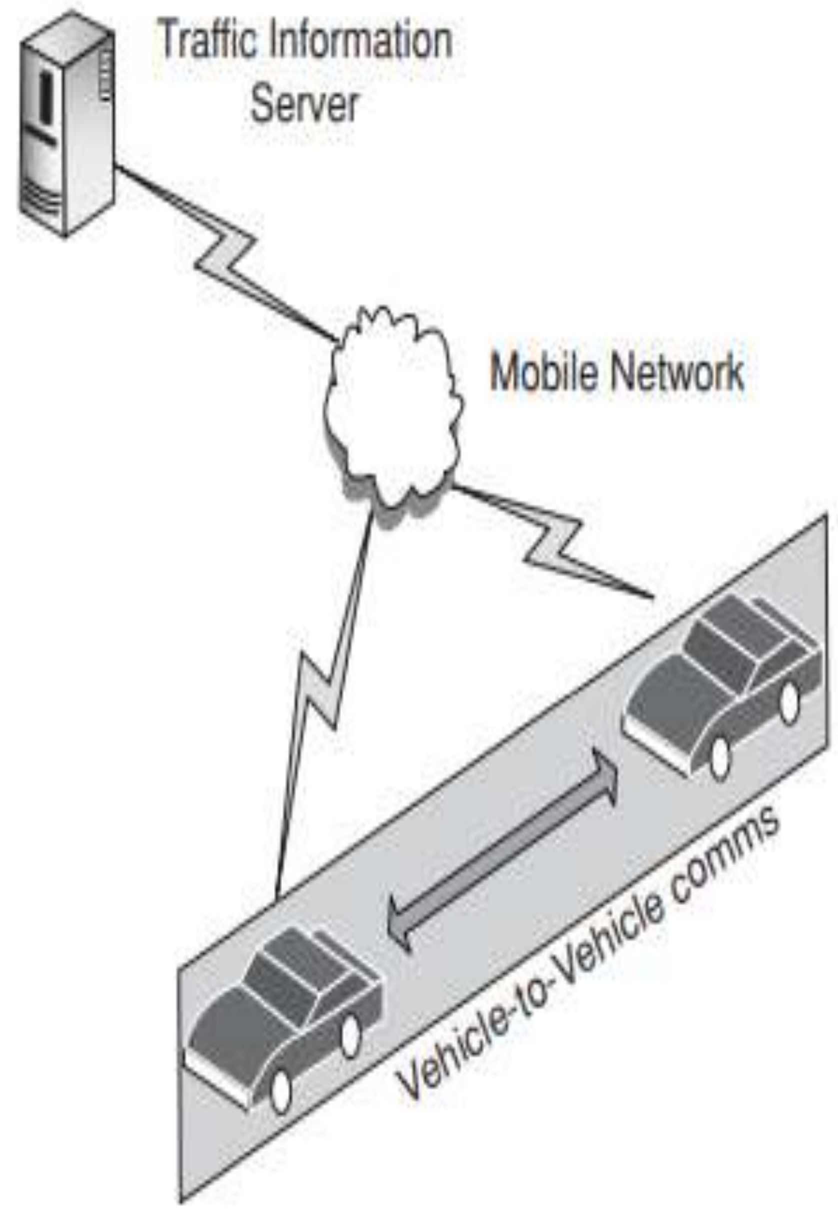


FIGURE Vehicle-to-infrastructure communications.

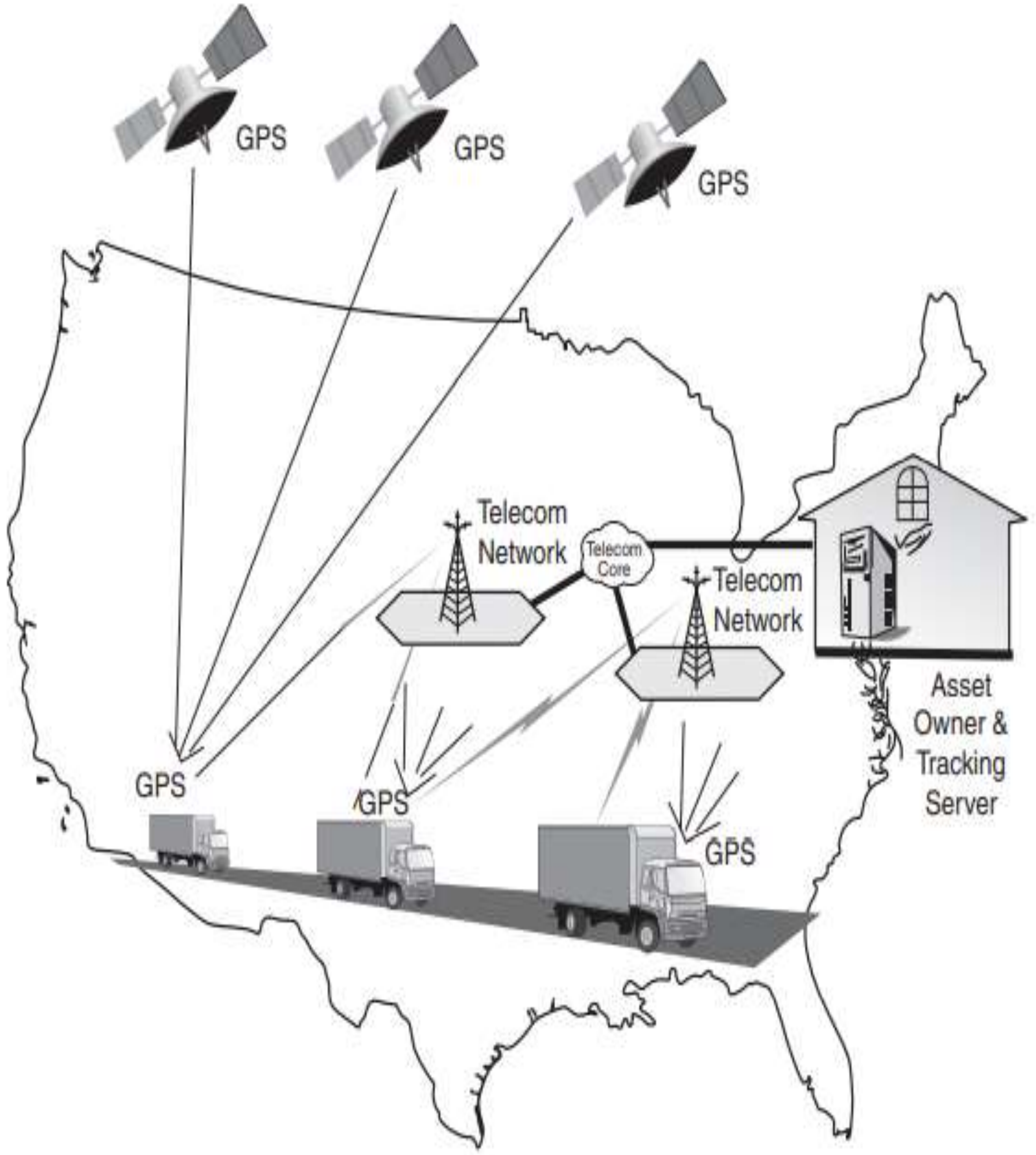


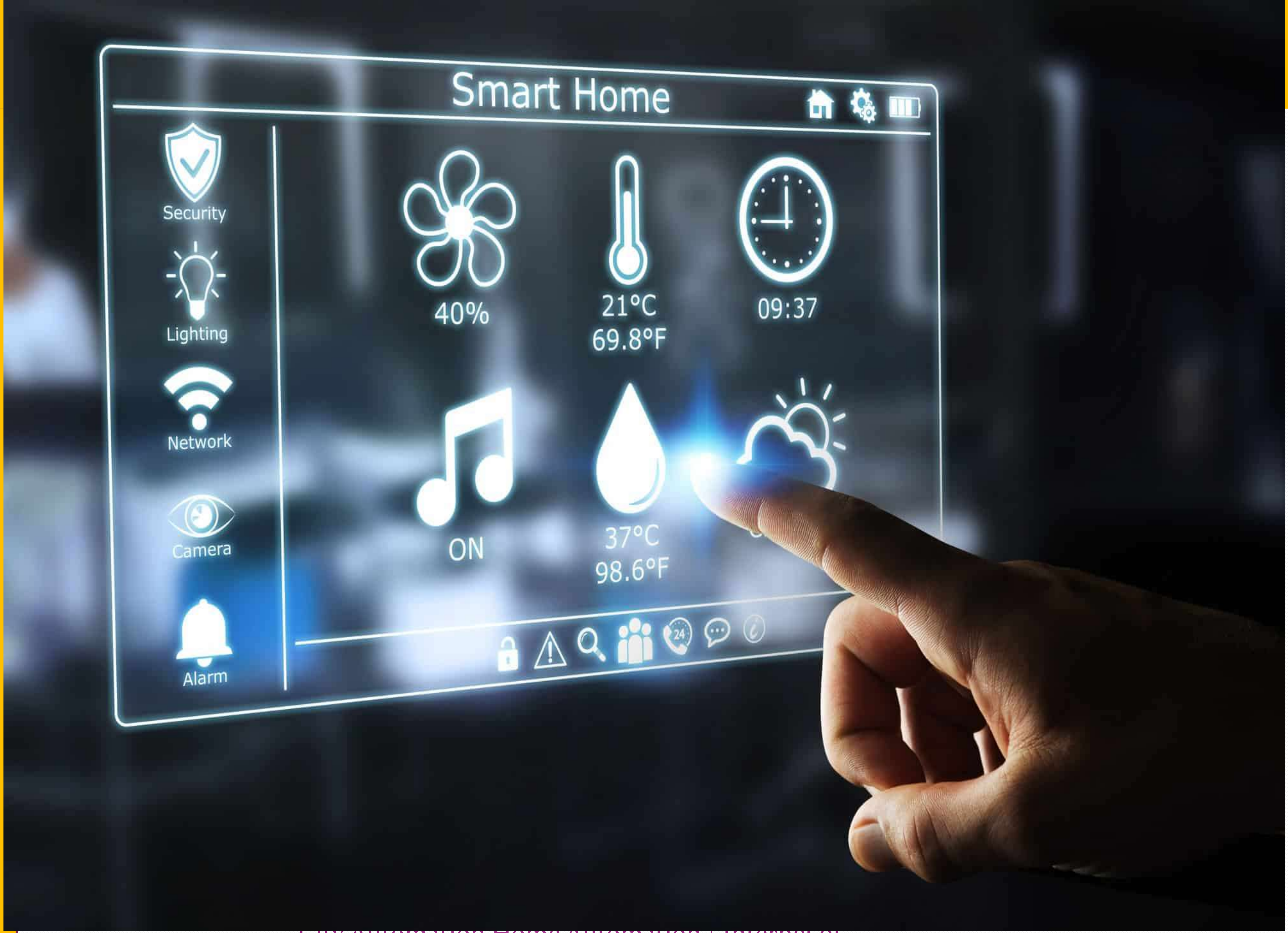
FIGURE Vehicular asset tracking.

HOME AUTOMATION

- Basic applications of the automated home include remote media control, heating control, lighting control (including low power landscape lighting control), and appliance control.
- Smart meters and energy efficiency
- Telehealth
- security and emergency services, Etc

are comes under this home automation category

IOT


Smart Home
Security
Lighting
Network
Camera
Alarm
40%
21°C
69.8°F
09:37
ON
37°C
98.6°F

Lighting
Climate
Security
Gadgets
Appliances
Starter Kits
Routers
HOME CONTROL
21°C
CONTROL
MODE

- M2M communications is expected to play a major role in residences, where instrumentation of elements supporting daily living such as,
- comfort,
- health,
- security, and
- energy efficiency can improve the quality of life and the quality of experience

Home control applications

- Home control applications include but are not limited to:
- Lighting control
- Thermostat/HVAC
- White goods
- Appliance control
- In-home displays

Home security applications

- Door access phone
- Window locks
- Motion detector
- Smoke/fire alert
- Baby monitors
- Medical pendant

I
O
T

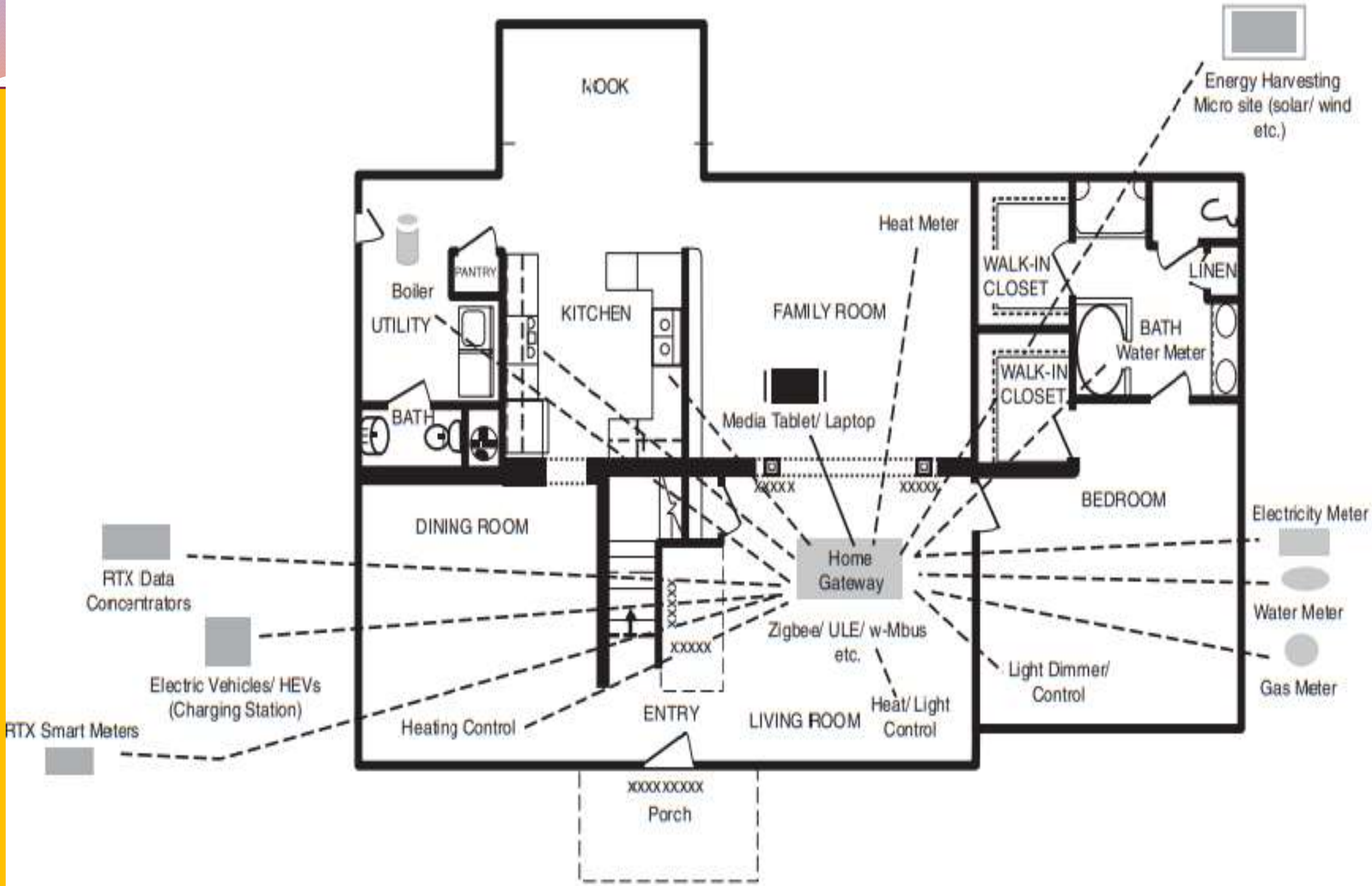


FIGURE Home automation example.

IoT Levels and Deployment Templates

An IoT system comprises the following components:

Device, Resource, Controller Service, Database, Web service, Analysis Component and **Application.**

Device :

An IoT device allows identification, remote sensing, remote monitoring capabilities.

Resource:

- Software components on the IoT device for
  - -accessing, processing and storing sensor information,
  - -controlling actuators connected to the device.
  - - enabling network access for the device.

Controller Service:

- Controller service is a native service that runs on the device and interacts with the web services.
- It sends data from the device to the web service and receives commands from the application (via web services) for controlling the device.

Database:

- Database can be either local or in the cloud and stores the data generated by the IoT device.

Web Service:

- Web services serve as a link between the IoT device, application, database and analysis components.
- It can be implemented using HTTP and **REST** principles (REST service) or using the **WebSocket** protocol (WebSocket service).

Analysis Component:

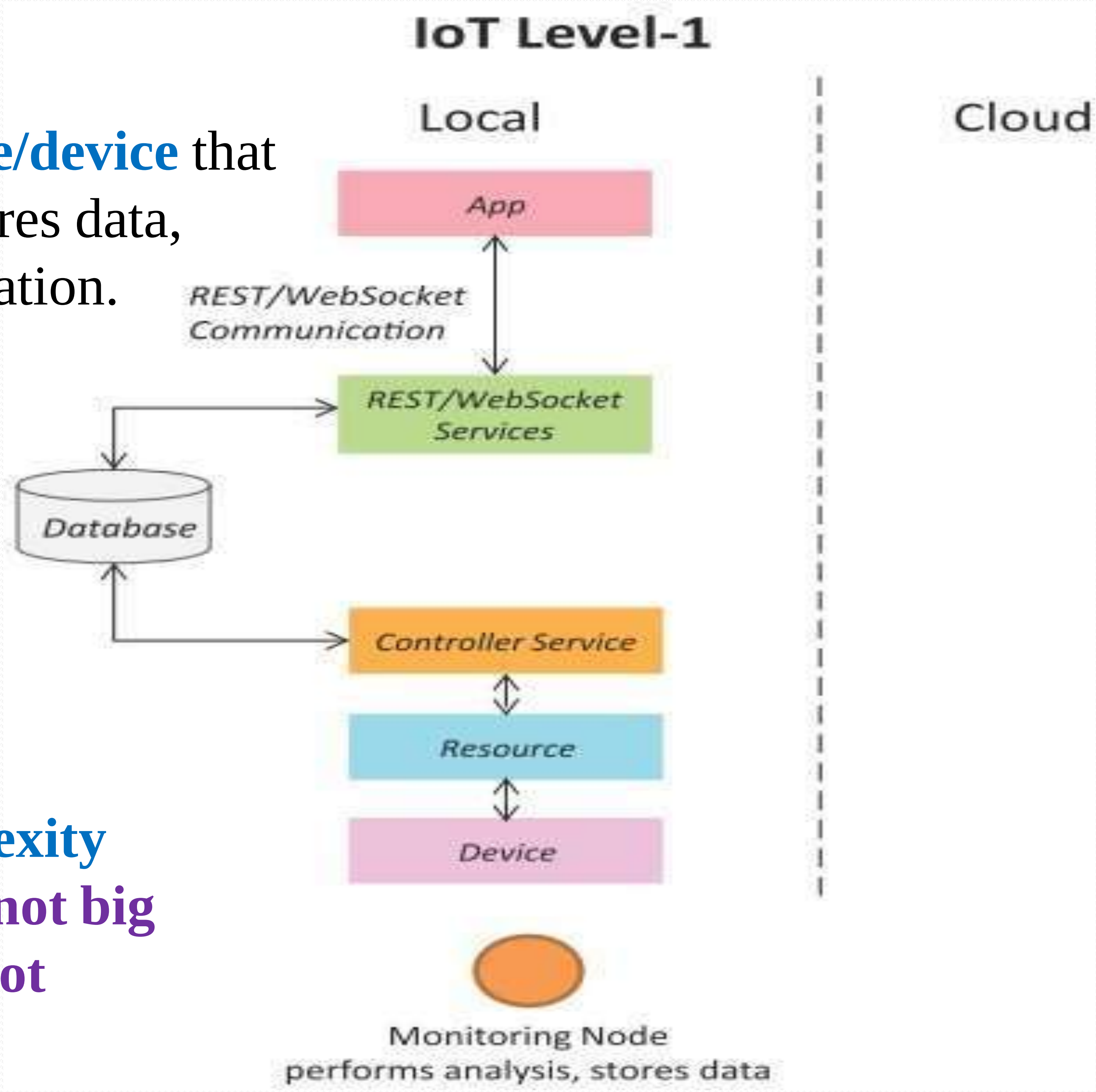
- Analysis Component is responsible for analyzing the IoT data and generating results in a form that is easy for the user to understand.

Application:

- IoT applications provide an interface that the users can use to control and monitor various aspects of the IoT system.
- Applications also allow users to view the system status and the processed data.

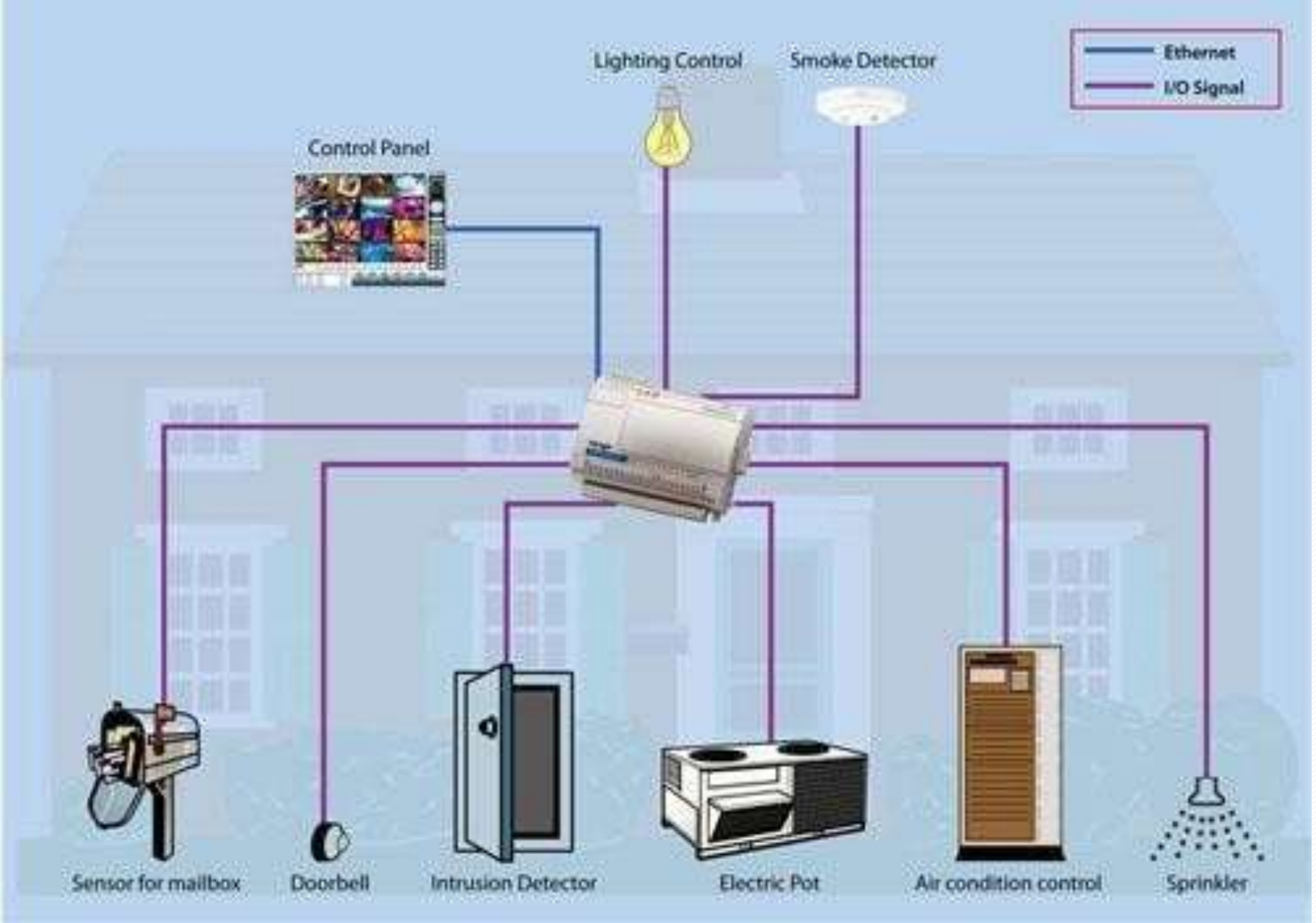
IoT Level-1

A level-1 IoT system has a **single node/device** that performs sensing and/or actuation, stores data, performs analysis and hosts the application.



Level-1 IoT systems are suitable for modelling **low- cost and low-complexity** solutions where the data involved is **not big** and the **analysis requirements** are **not computationally intensive.**

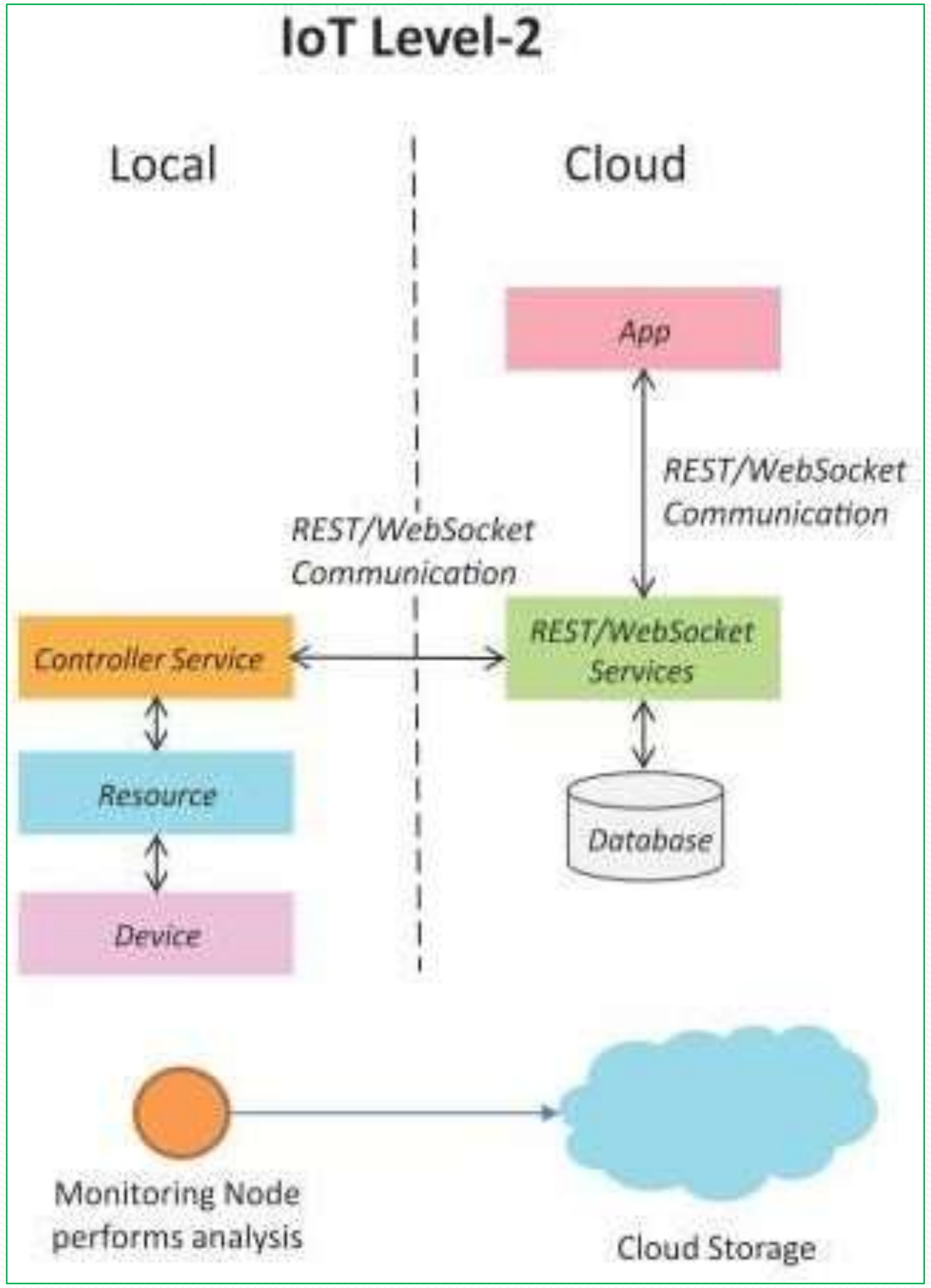
IoT – Level 1 Example : Home Automation System



IoT Level-2

- A level-2 IoT system has a **single node** that performs sensing and/or actuation and **local analysis**. **Data is stored in the cloud** and the **application is usually cloud-based**.

- Level-2 IoT systems are suitable for solutions where the **data involved is big**; however, the **primary analysis** requirement is **not computationally intensive** and can be done locally.

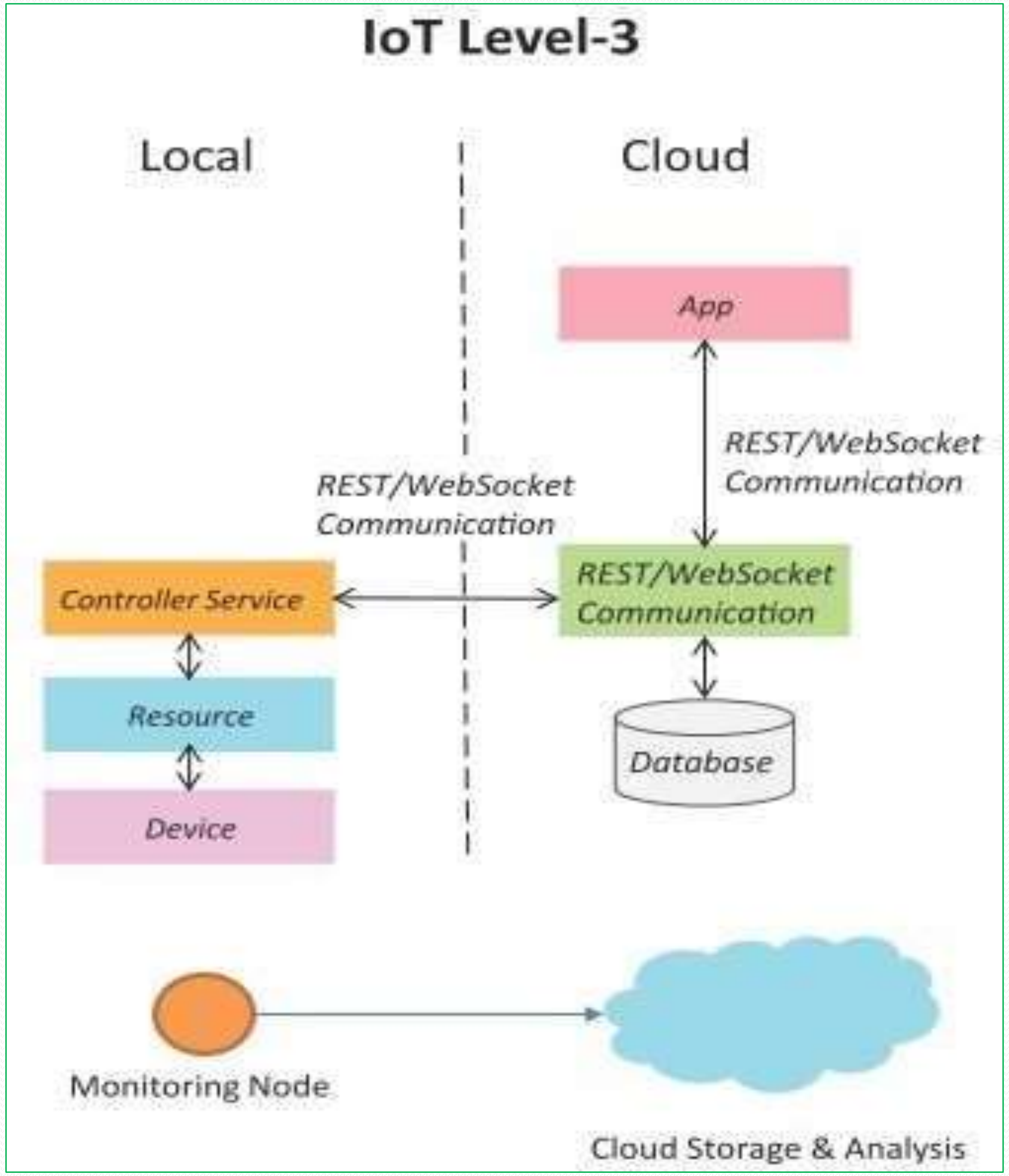


IoT – Level 2 Example: Smart Irrigation

IoT Level-3

A level-3 IoT system has a **single node**. Data is stored and **analyzed in the cloud** and the **application is cloud-based.**

Level-3 IoT systems are suitable for solutions where the **data involved is big** and the **analysis requirements are computationally intensive**.

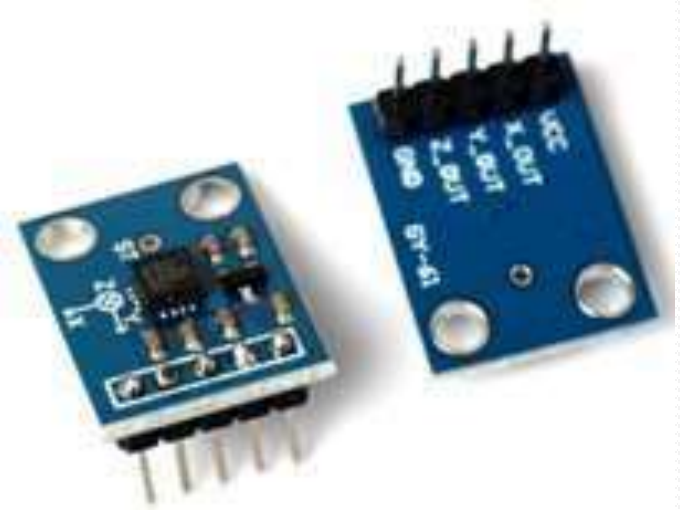


IoT – Level 3 Example: Tracking Package Handling

Sensors used

Accelrometer

sense movement or vibrations



Gyroscope

Gives orientation info

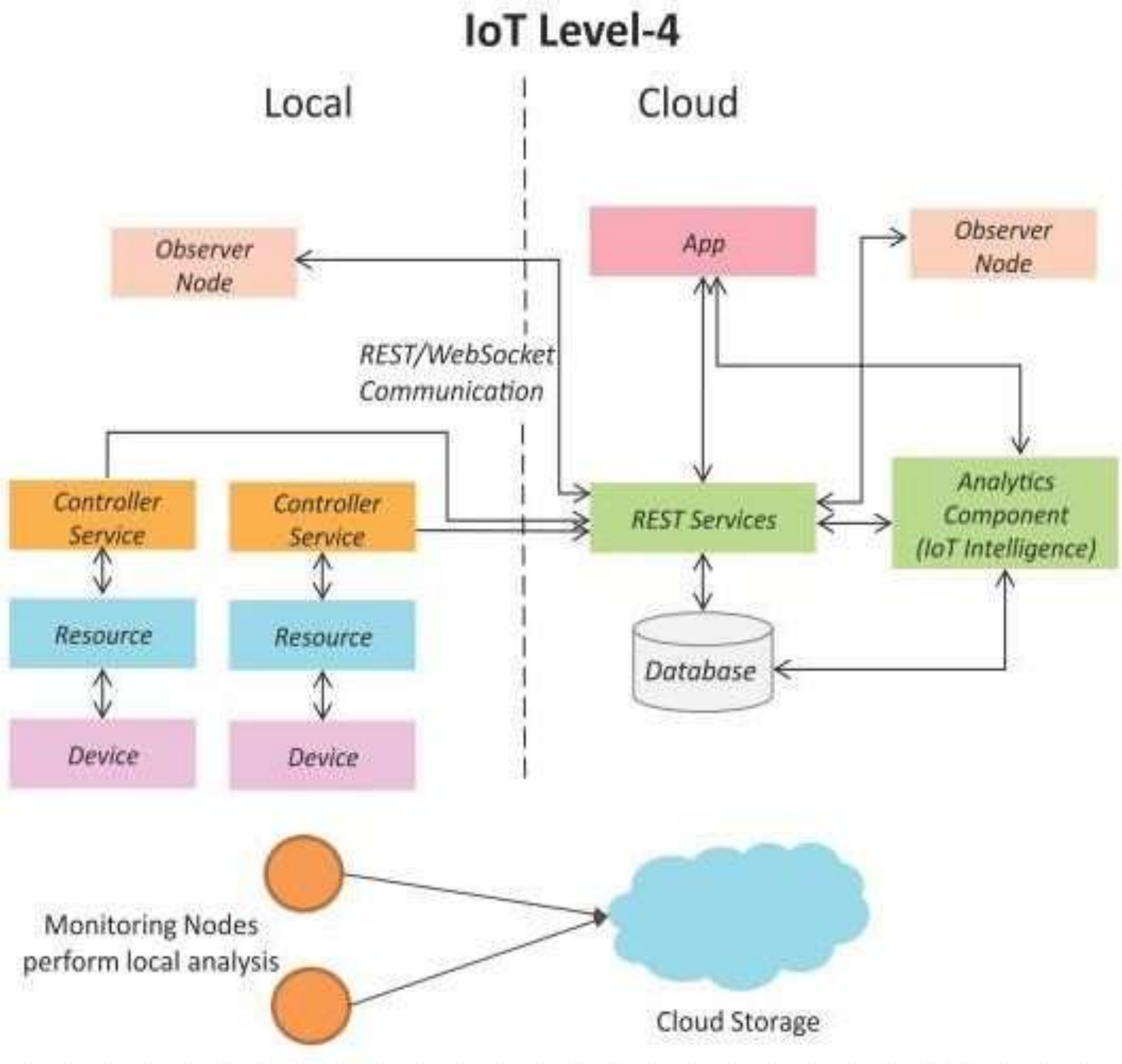
Websocket service is used because sensor data can be sent in real time.

IoT Level-4

A level-4 IoT system has **multiple nodes** that perform **local analysis**. **Data is stored in the cloud** and the application is cloud-based.

Level-4 contains local and cloud-based **observer nodes** which can subscribe and receive information collected in the cloud from IoT devices.

Level-4 IoT systems are suitable for solutions where **multiple nodes are required**, the **data involved is big** and the **analysis requirements are computationally intensive.**

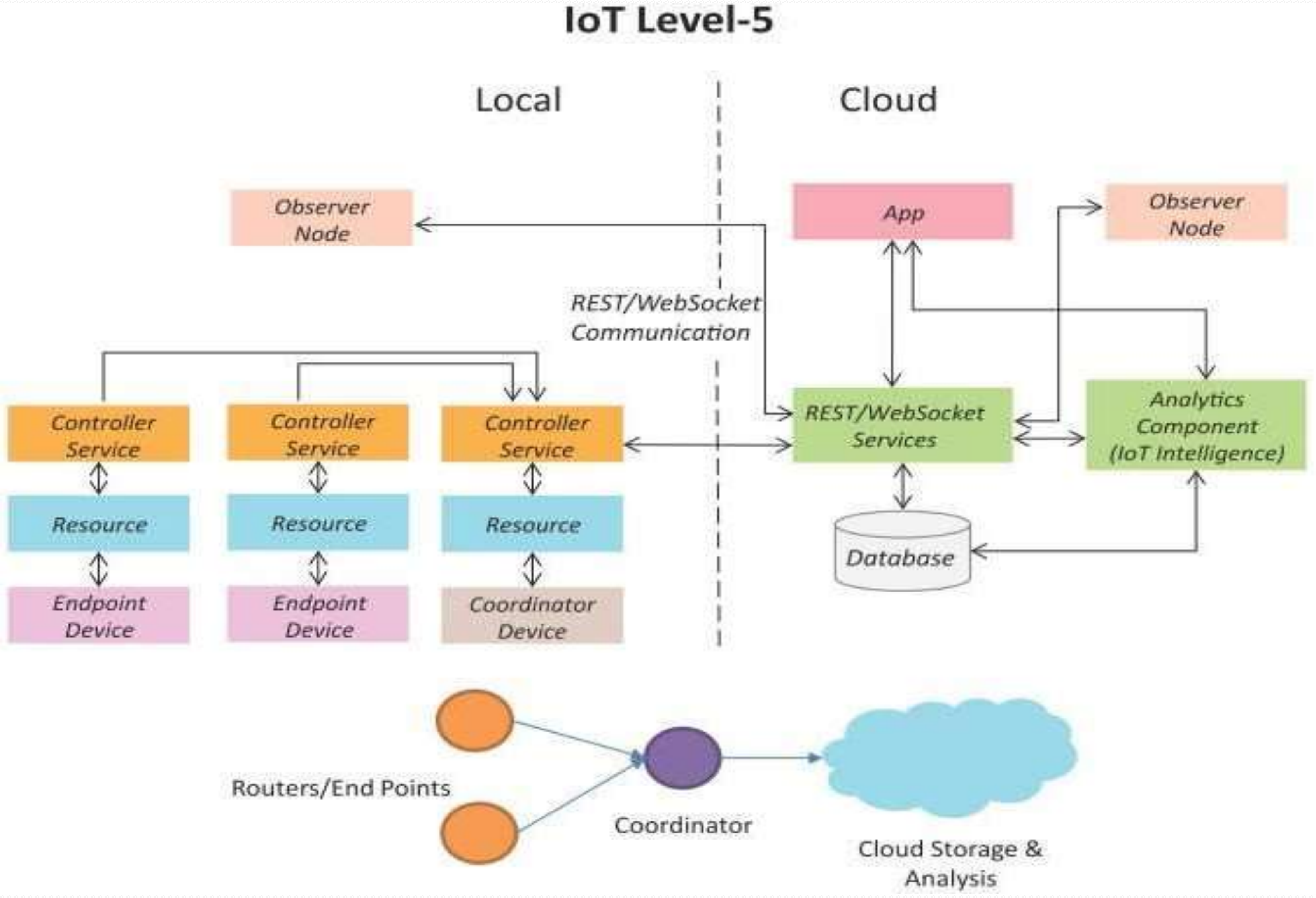


IoT – Level 4 Example: Noise Monitoring

Sound Sensors are used

IoT Level-5

- A level-5 IoT system has **multiple end nodes** and **one coordinator node.**
- The end nodes perform sensing and/or actuation.
- The coordinator node **collects data from the end nodes and sends it to the cloud.**
- Data is stored and **analyzed in the cloud** and the **application is cloud- based.**

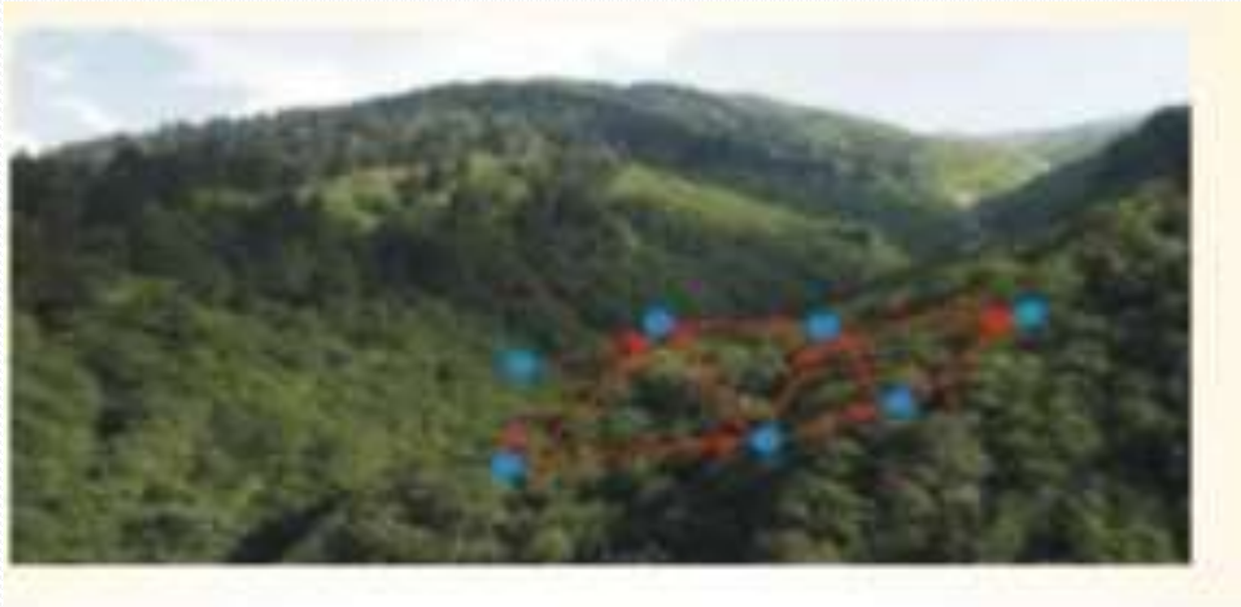


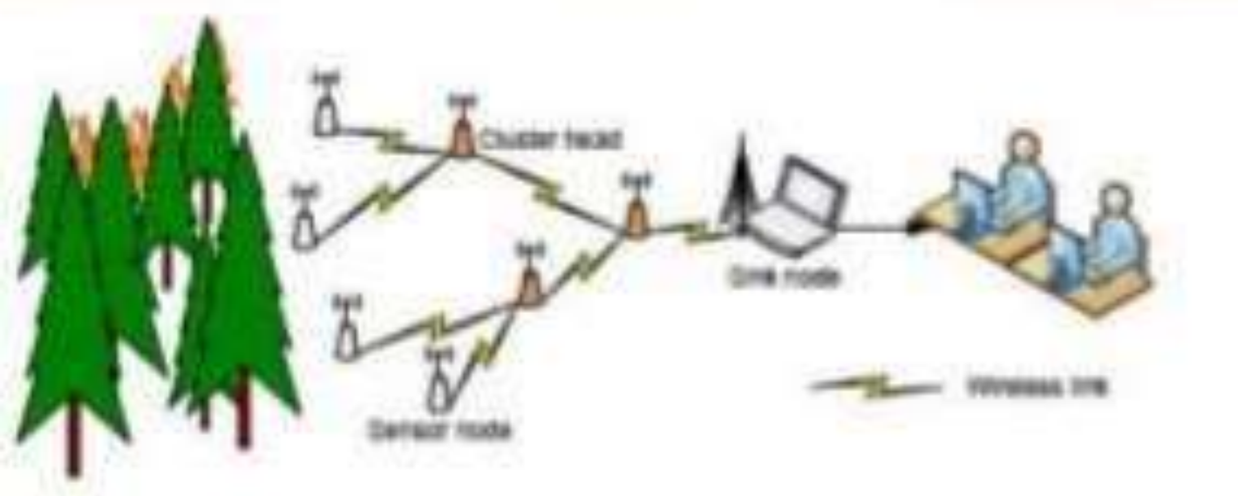
Level-5 IoT systems are suitable for **solutions based on wireless sensor networks**, in which the **data involved is big** and the **analysis requirements are computationally intensive.**

IoT – Level 5 Example: Forest Fire Detection

Detect forest fire in early stages to take action while the fire is still controllable.

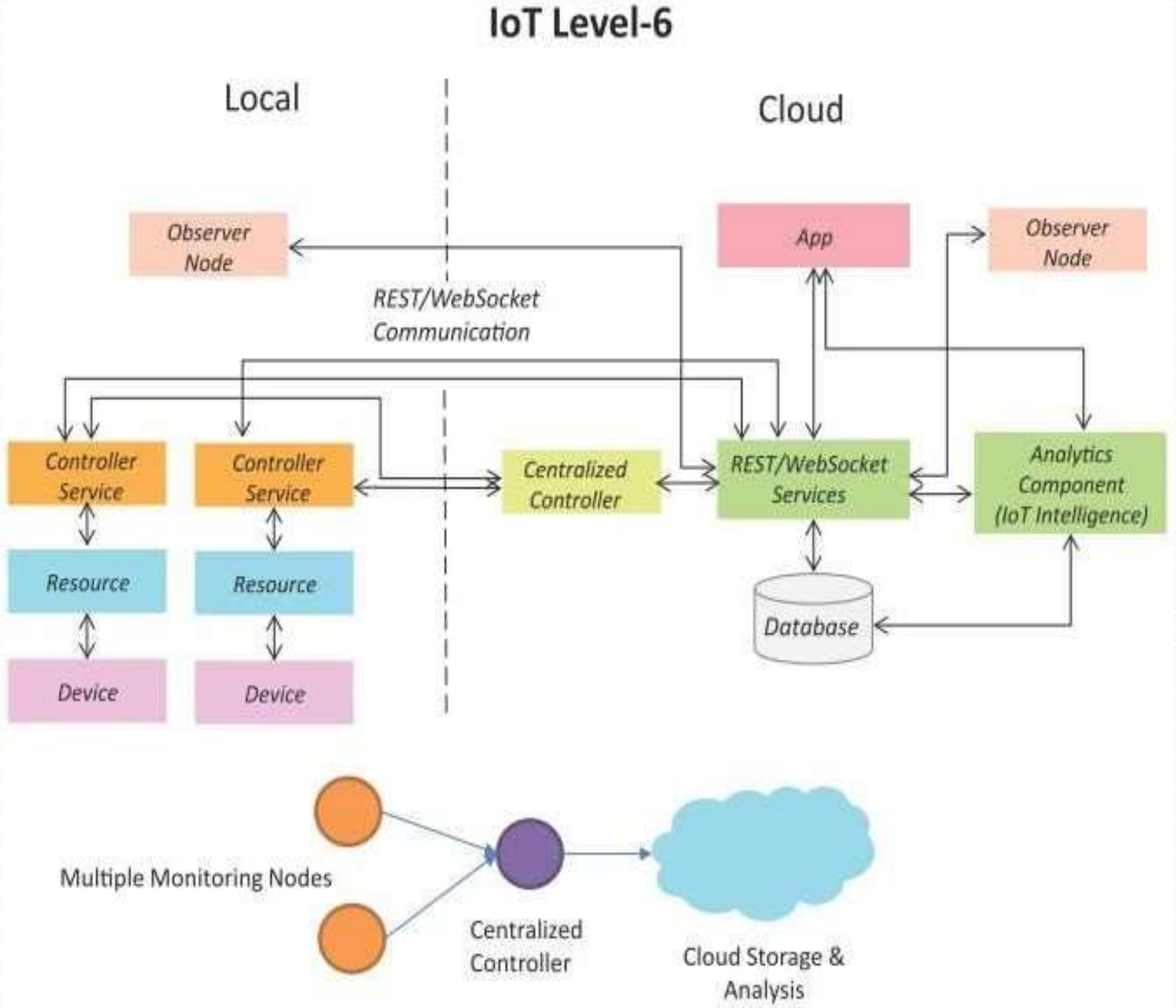
Sensors measure the temperature, smoke, weather, slope of the earth, wind speed, speed of fire spread, flame length





IoT Level-6

- A level-6 IoT system has **multiple independent end nodes** that perform sensing and/or actuation and send data to the cloud.
- **Data is stored in the cloud** and the **application is cloud-based**.
- The **analytics** component analyzes the data and stores the results in the **cloud database**.
- The results are visualized with the cloud-based application.
- The **centralized controller** is aware of the status of all the end nodes and **sends control commands to the nodes**.



I O T

IoT – Level 6 Example: Weather Monitoring System

Sensors used

Wind speed and direction
Solar radiation
Temperature (air, water, soil)
Relative humidity

Precipitation
Snow depth
Barometric pressure
Soil moisture

IoT Issues and Challenges

Security

- Cyber Attacks, Data Theft

Privacy

- Controlling access and ownership of data.

InterOperability

- Integration Inflexibility

Legality and Rights

- Data Protection laws be followed, Data Retention and destruction policies

Economy and Development

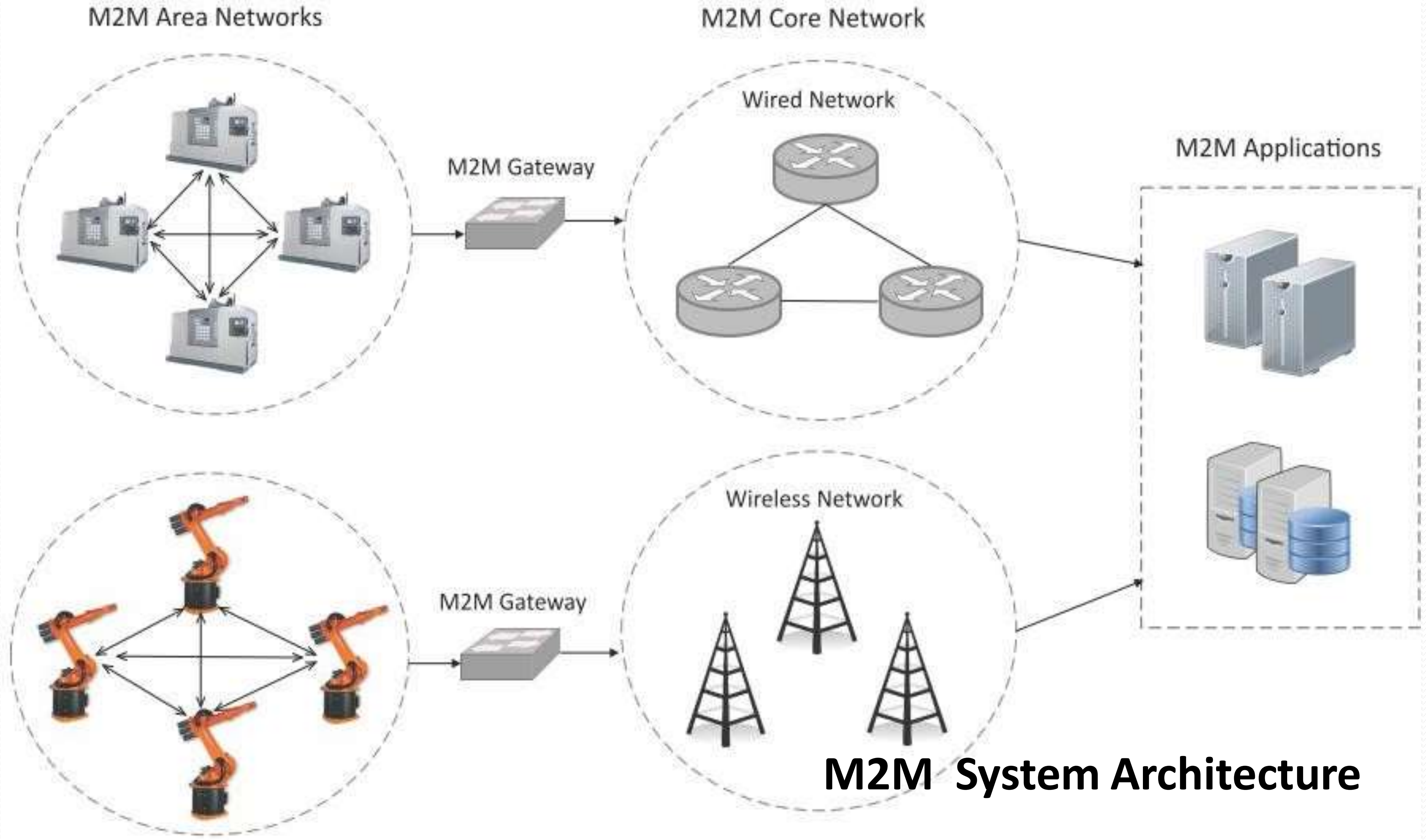
- Investment Incentives, Technical Skill Requirement

Outline

- M2M
- Differences and Similarities between M2M and IoT
- SDN and NFV for IoT

Machine-to-Machine (M2M)

- Machine-to-Machine (M2M) refers to networking of machines (or devices) for the purpose of remote monitoring and control and data exchange.



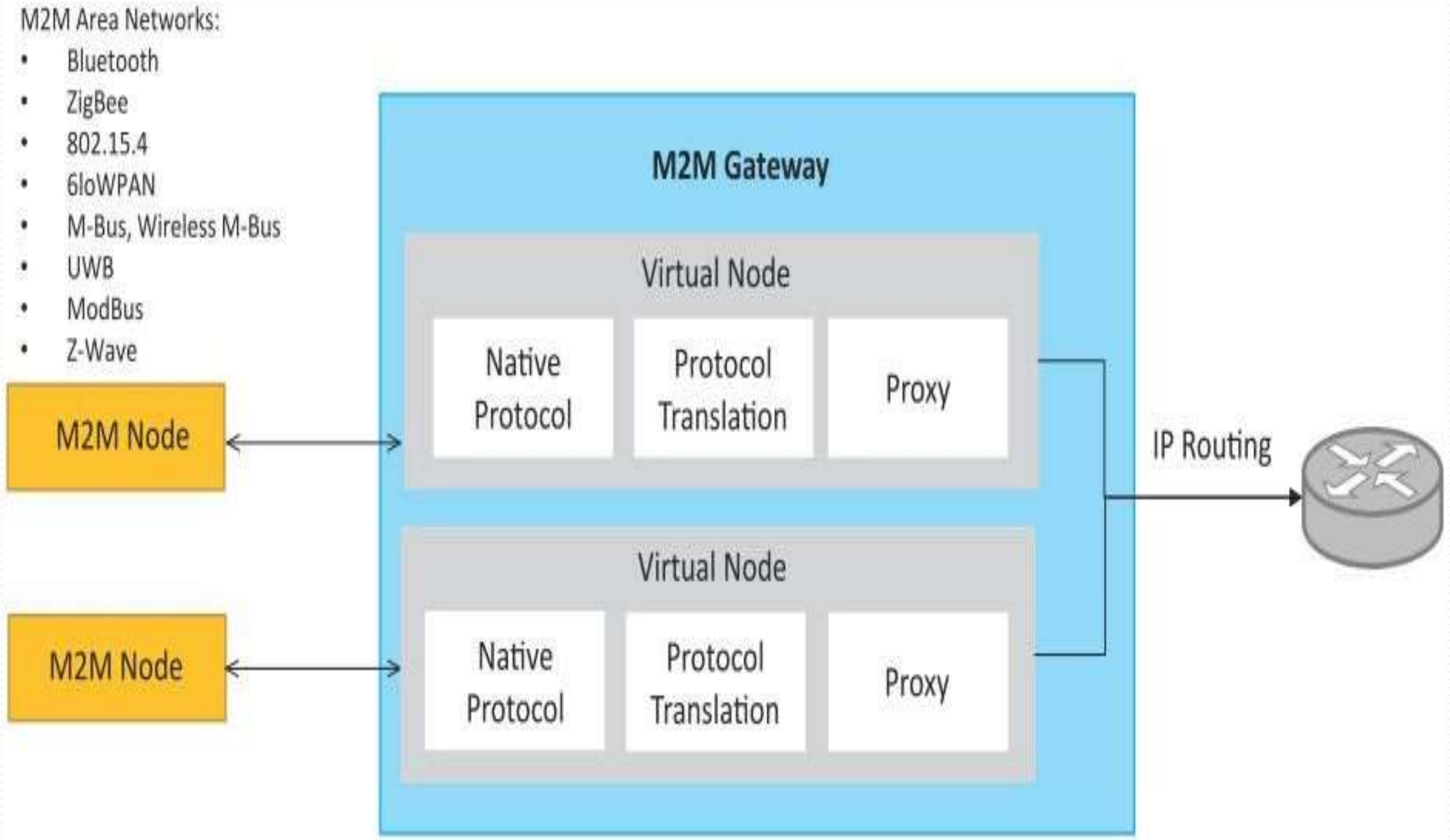
M2M System Architecture

Machine-to-Machine (M2M)

- An M2M area network comprises of machines (or M2M nodes) which have embedded hardware modules for sensing, actuation and communication.
- Various communication protocols can be used for M2M local area networks such as ZigBee, Bluetooh, ModBus, M-Bus, Wirless M-Bus, Power Line Communication (PLC), 6LoWPAN, IEEE 802.15.4, etc.
- The communication network provides connectivity to remote M2M area networks.
- The communication network can use either wired or wireless networks (IP- based).
- While the M2M area networks use either proprietary or non-IP based communication protocols, the communication network uses IP-based networks.

M2M gateway

- Since non-IP based protocols are used within M2M area networks, the M2M nodes within one network cannot communicate with nodes in an external network.
- To enable the communication between remote M2M area networks, M2M gateways are used.

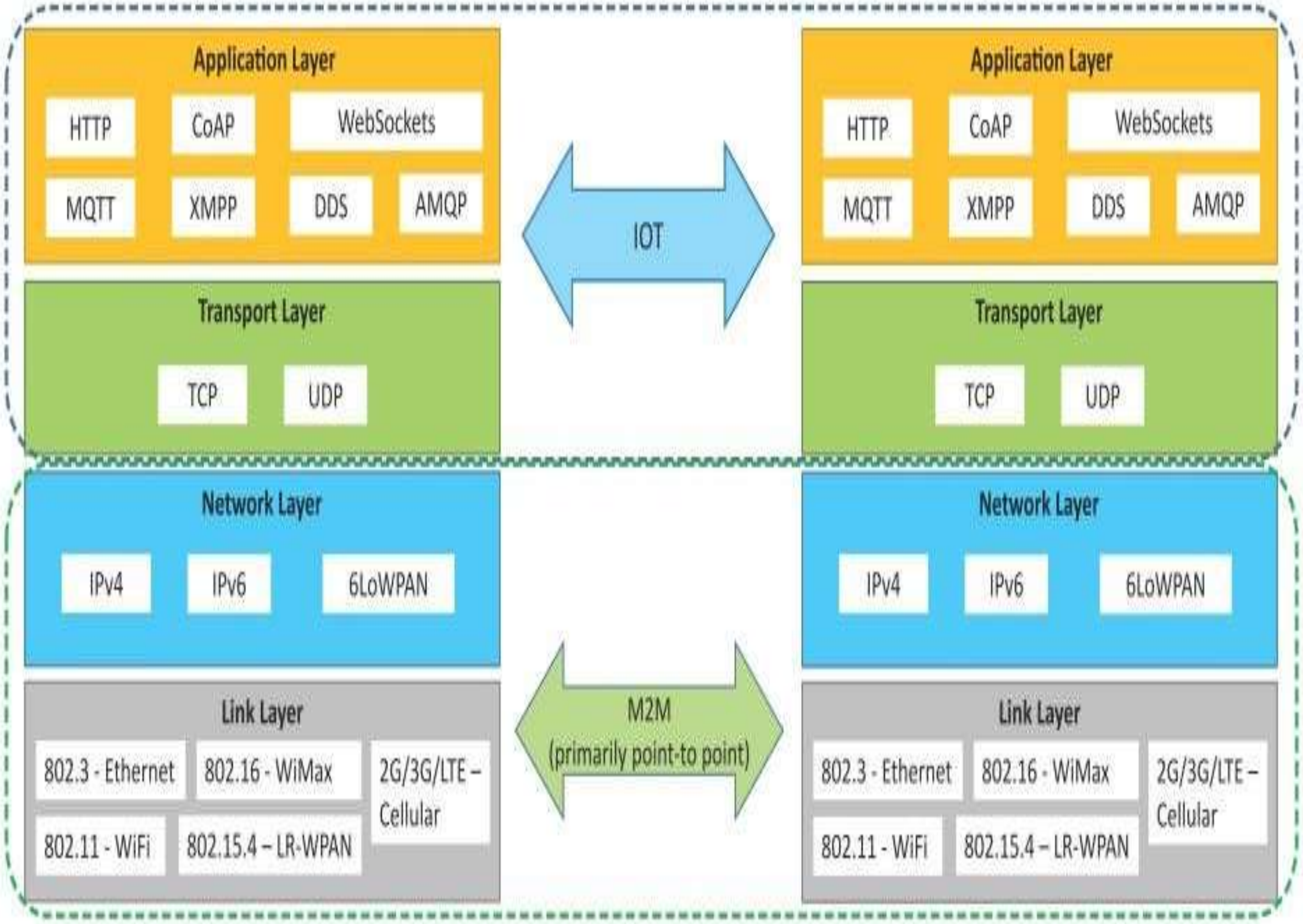


Difference between IoT and M2M

- Communication Protocols
  - M2M and IoT can differ in how the communication between the machines or devices happens.
  - M2M uses either proprietary or non-IP based communication protocols for communication within the M2M area networks.
- Machines in M2M vs Things in IoT
  - The "Things" in IoT refers to physical objects that have unique identifiers and can sense and communicate with their external environment (and user applications) or their internal physical states.
  - M2M systems, in contrast to IoT, typically have homogeneous machine types within an M2M area network.

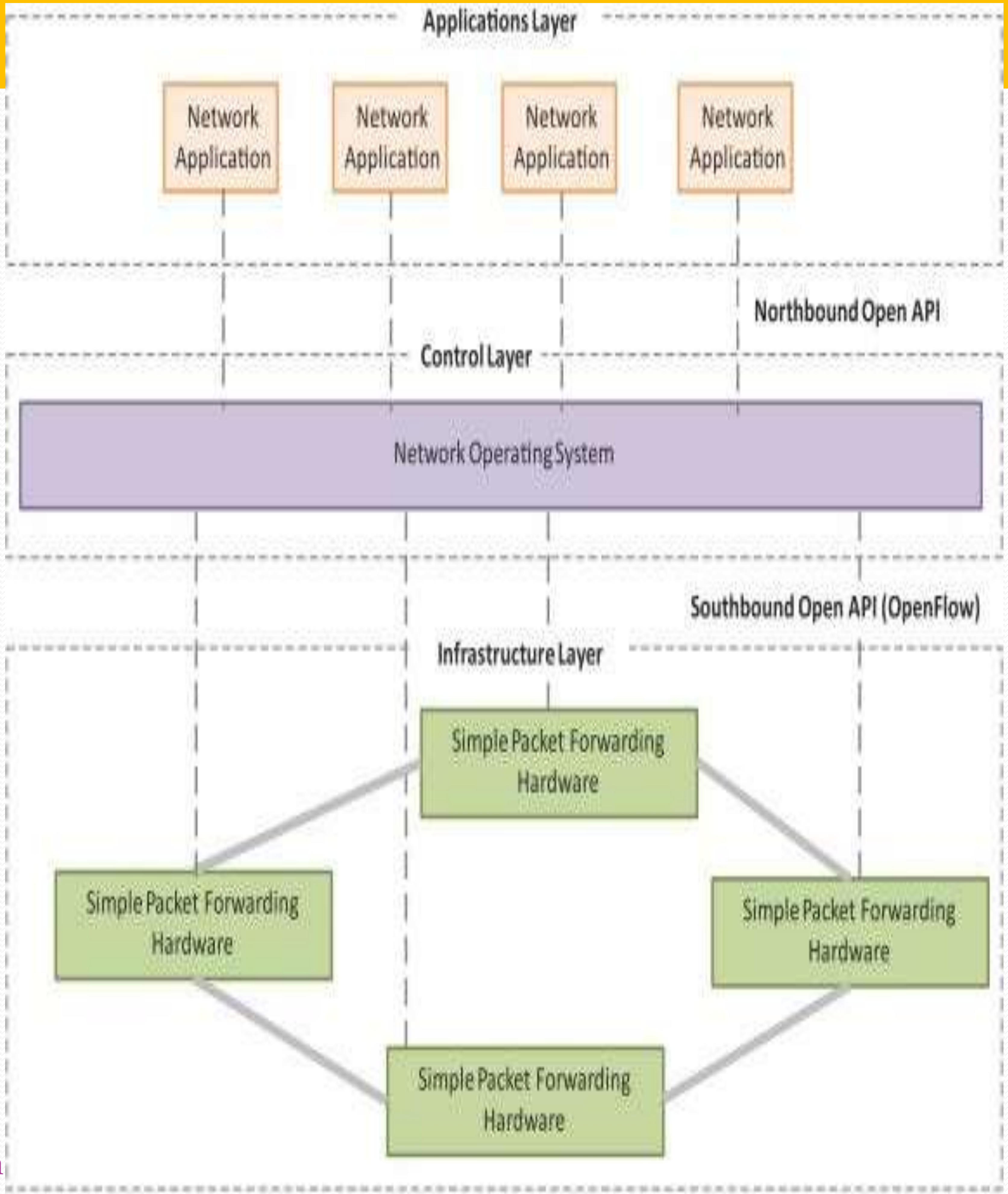
Difference between IoT and M2M

- Hardware vs Software Emphasis
  - While the emphasis of M2M is more on hardware with embedded modules, the emphasis of IoT is more on software.
- Data Collection & Analysis
  - M2M data is collected in point solutions and often in on-premises storage infrastructure.
  - In contrast to M2M, the data in IoT is collected in the cloud (can be public, private or hybrid cloud).
- Applications
  - M2M data is collected in point solutions and can be accessed by on-premises applications such as diagnosis applications, service management applications, and on- premisis enterprise applications.
  - IoT data is collected in the cloud and can be accessed by cloud applications such as analytics applications, enterprise applications, remote diagnosis and management applications, etc.


Application Layer
HTTP
CoAP
WebSockets
MQTT
XMPP
DDS
AMQP
Transport Layer
TCP
UDP
Network Layer
IPv4
IPv6
6LoWPAN
Link Layer
802.3 - Ethernet
802.16 - WiMax
2G/3G/LTE – Cellular
802.11 - WiFi
802.15.4 – LR-WPAN
IOT
M2M
(primarily point-to point)
Application Layer
HTTP
CoAP
WebSockets
MQTT
XMPP
DDS
AMQP
Transport Layer
TCP
UDP
Network Layer
IPv4
IPv6
6LoWPAN
Link Layer
802.3 - Ethernet
802.16 - WiMax
2G/3G/LTE – Cellular
802.11 - WiFi
802.15.4 – LR-WPAN

SDN

- Software-Defined Networking (SDN) is a networking architecture that separates the control plane from the data plane and centralizes the network controller.
- Software-based SDN controllers maintain a unified view of the network and make confi guration, management and provisioning simpler.
- The underlying infrastructure in SDN uses simple packet forwarding hardware as opposed to specialized hardware in conventional networks.



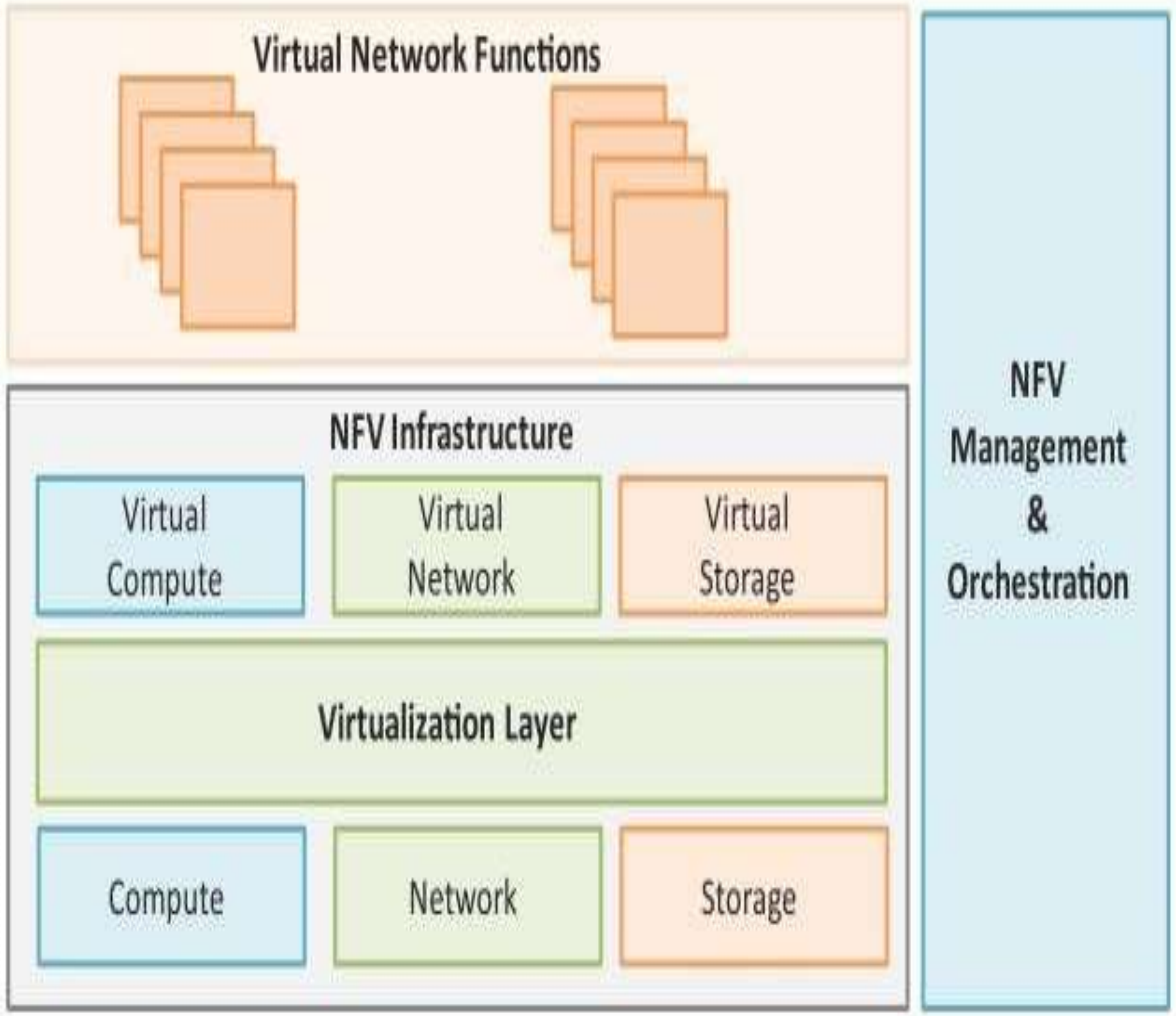
IOT

Key elements of SDN

- Centralized Network Controller
  - With decoupled control and data planes and centralized network controller, the network administrators can rapidly configure the network.
- Programmable Open APIs
  - SDN architecture supports programmable open APIs for interface between the SDN application and control layers (Northbound interface).
- Standard Communication Interface (OpenFlow)
  - SDN architecture uses a standard communication interface between the control and infrastructure layers (Southbound interface).
  - OpenFlow, which is defined by the Open Networking Foundation (ONF) is the broadly accepted SDN protocol for the Southbound interface.

NFV

- Network Function Virtualization (NFV) is a technology that leverages virtualization to consolidate the heterogeneous network devices onto industry standard high volume servers, switches and storage.
- NFV is complementary to SDN as NFV can provide the infrastructure on which SDN can run.

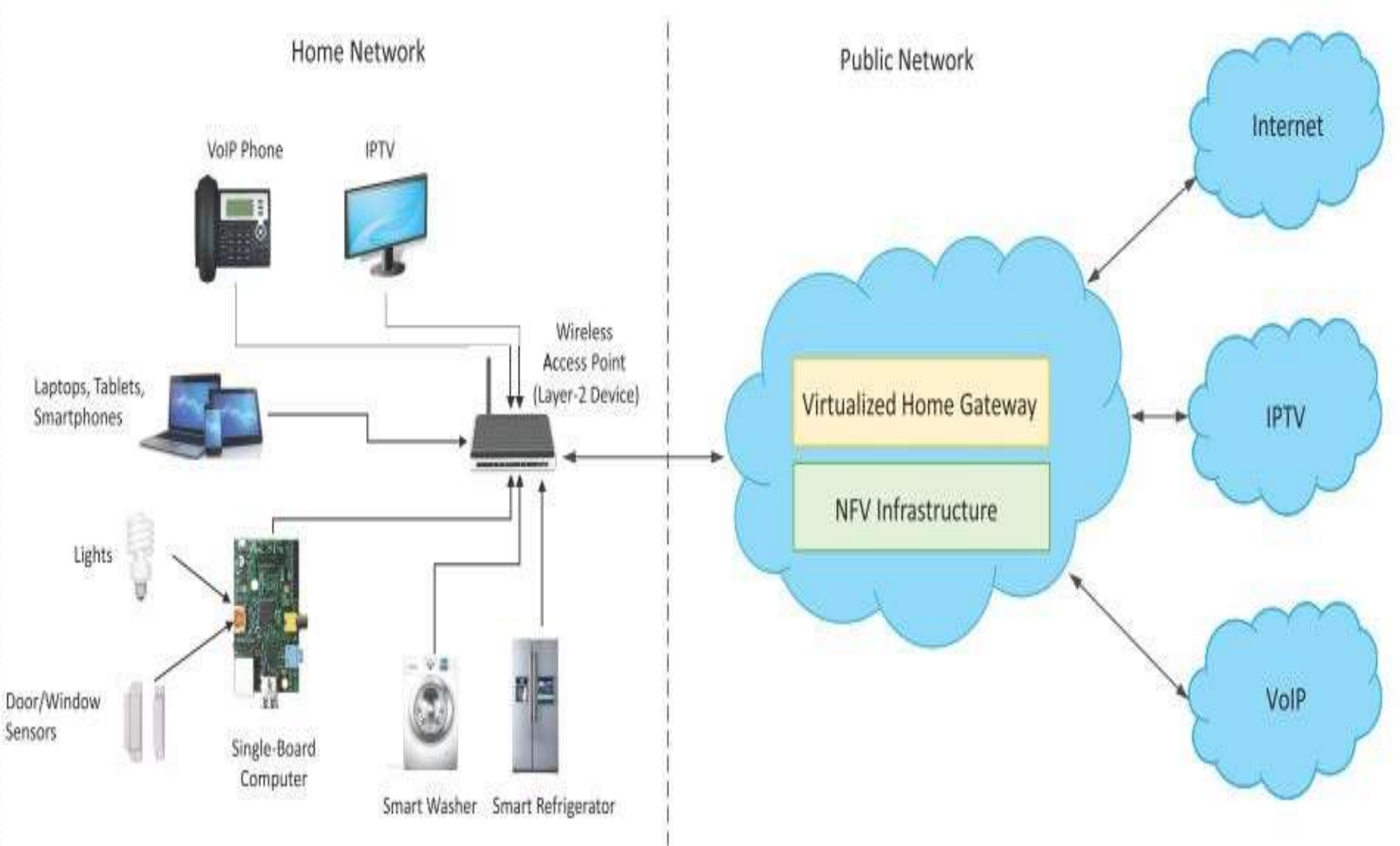


Key elements of NFV

- Virtualized Network Function (VNF):
  - VNF is a software implementation of a network function which is capable of running over the NFV Infrastructure (NFVI).
- NFV Infrastructure (NFVI):
  - NFVI includes compute, network and storage resources that are virtualized.
- NFV Management and Orchestration:
  - NFV Management and Orchestration focuses on all virtualization-specific management tasks and covers the orchestration and life-cycle management of physical and/or software resources that support the infrastructure virtualization, and the life-cycle management of VNFs.

NFV Use Case

- NFV can be used to virtualize the Home Gateway. The NFV infrastructure in the cloud hosts a virtualized Home Gateway. The virtualized gateway provides private IP addresses to the devices in the home. The virtualized gateway also connects to network services such as VoIP and IPTV.



I
O
T

THANK YOU