

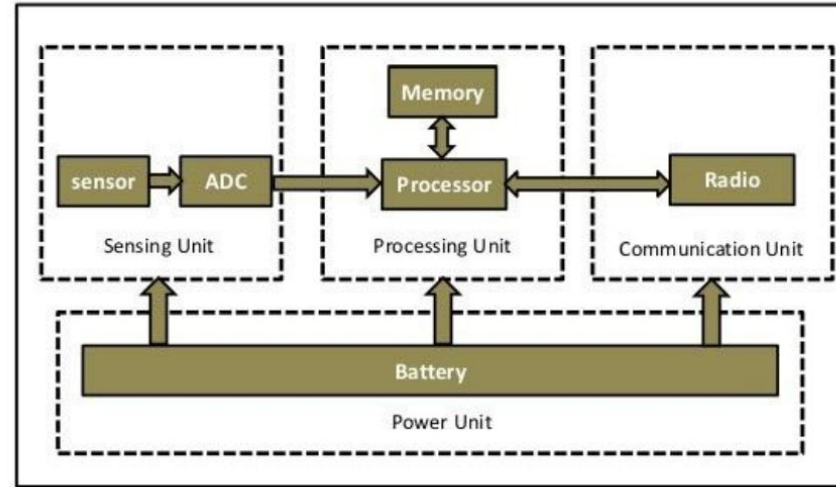
Module 6: Introduction to IoT

Introduction

- **Internet of things or IoT** is a system of connected devices through the internet.
- It involves **mechanical devices, sensors, home appliances, vehicles**, etc., apart from desktop, mobile, and laptop.
- These devices are designed in such a way that they can share data with other devices over the internet.
- IoT basically provides a **platform for devices to interact and collaborate with each other**.

Components of IoT

Node Architecture (Components of Node)



Different types of Sensors

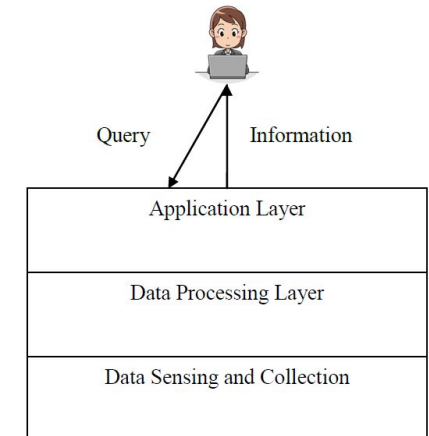
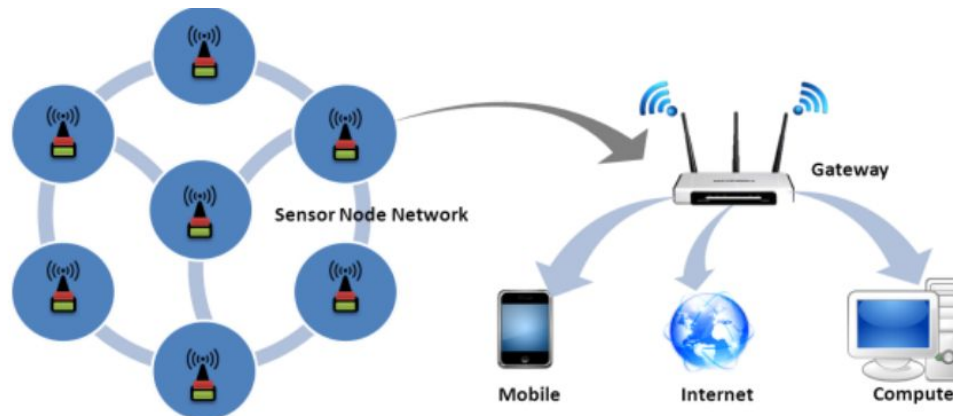


Figure 2 IoT System Layers

- **Sensor**

- The **sensor** is the layer that connects the IoT device to the **physical environment or human being**. As the name suggests, **it senses the changes and sends data to the cloud for processing**. These sensors **continuously** collect data from the environment and transmit the information to the next layer: **pressure sensor, temperature sensor, and light intensity detectors**.

- **Gateway**

- **Gateway facilitates** data flow management and protocol layer to transfer data from one device to another. It translates the network protocols for devices and provides encryption for the data flowing in the network. It's like a **layer between the cloud** and devices that filter away cyber attacks and illegal data access.

- **Connectivity**

- The data collected by sensors need to be sent to the cloud for analytics. Data need the **internet as a medium to transfer data from one device to another**. Hence all the **IoT devices** need to be connected to the internet all the time. Networks like **Wi-Fi, Bluetooth, WAN, satellite networks make it easy to stay connected**. Data can be sent to the cloud through these networks.

- **Cloud**

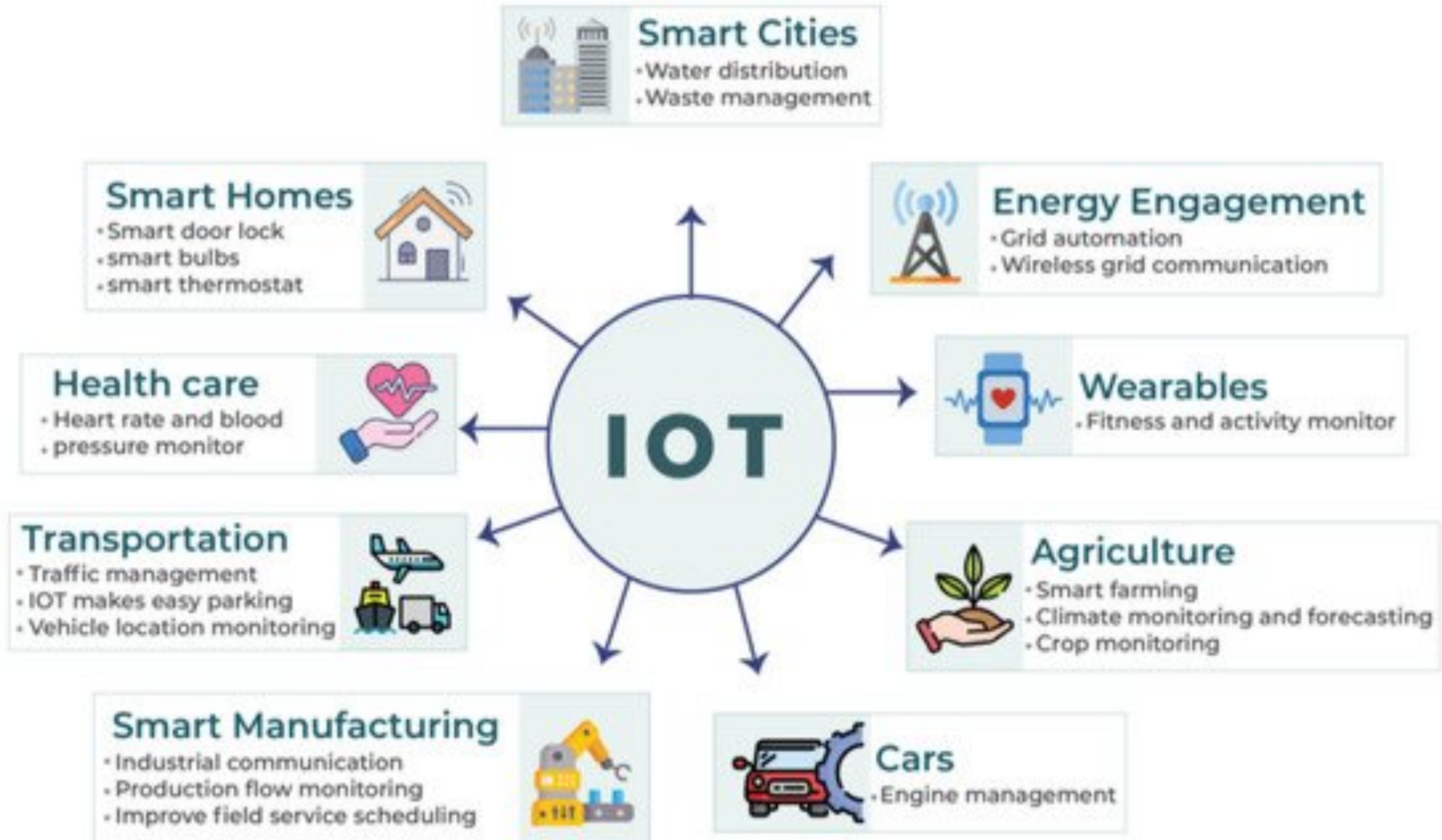
- IoT systems send massive data from devices, and this data needs to be managed efficiently to generate meaningful output. **To store this huge amount of data, the IoT cloud is used**. It provides tools to collect, process, and store data. Data is readily available and remotely accessible through the internet. It also provides a platform for analytics. IoT cloud is a sophisticated, high-performance network of servers to perform high-speed processing of a huge amount of data.

- **Analytics**
- **Analytics** is the **process of converting raw data into some meaningful form**. IoT analytics **supports real-time analysis**, which captures real-time changes and irregularities. The data is then converted into a format that is easy to understand by the end-user. As a result, users or businesses can analyze the trends shown in reports, predict the market and plan ahead for a successful implementation of their ideas.
- **User Interface**
- **The User interface is a visible, tangible part of IoT systems. It is that part of the system which interacts with the end-user.** The information can be made available either in **report format or in the form of some actions like trigger an alarm, a notification, etc.** The user can also choose to perform some actions. Therefore, it is important to create a user-friendly interface that can be used without much effort and **technical knowledge**. The easier the user interface is, the more successful the product is.

Challenges of IoT

- **Devices heterogeneity**
- **Scalability**
- **Security and privacy-preserving mechanisms**
- **Energy-optimized solutions**

Applications



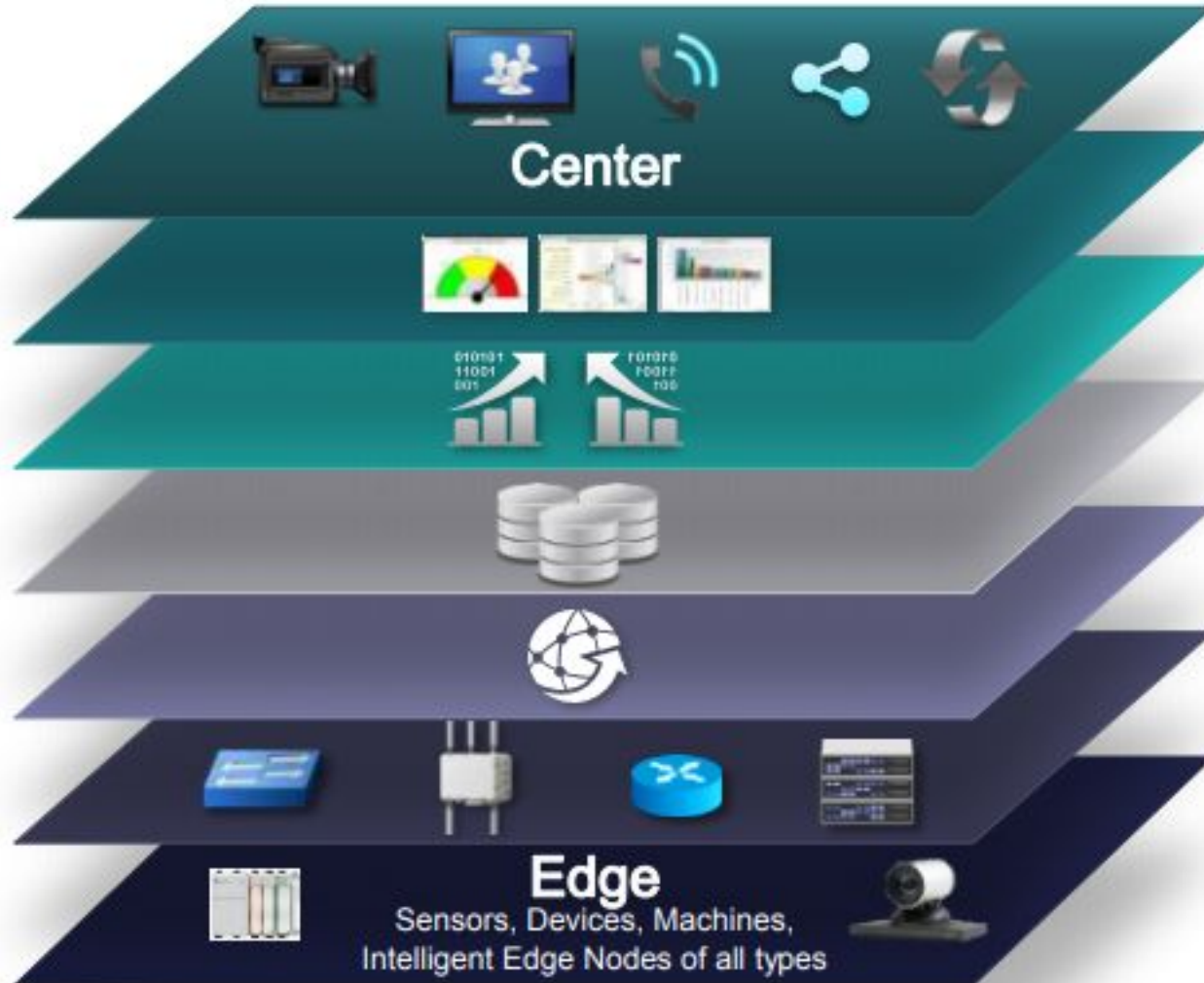
Major IoT building blocks and layers

- Before we go any further, it's worth pointing out that there is **no single, agreed-upon IoT architecture**. It varies in **complexity and number of architectural layers** depending on a **particular business task**.
- For example, the **Reference Model** introduced in **2014 by Cisco, IBM, and Intel** at the **2014 IoT World Forum** has as many as seven layers. According to an official [press release](#) by Cisco forum host, the architecture aims to *"help educate CIOs, IT departments, and developers on deployment of IoT projects, and accelerate the adoption of IoT"*

IoT World Forum Reference Model

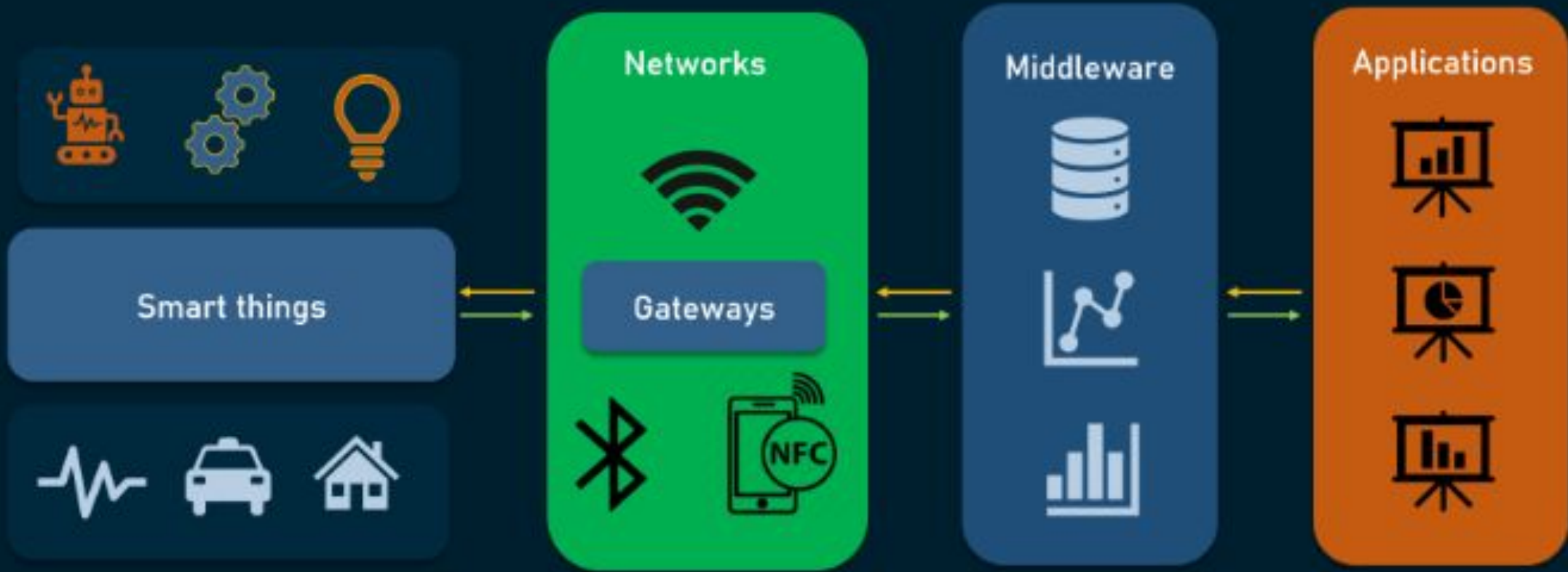
Levels

- 7 Collaboration & Processes**
(Involving People & Business Processes)
- 6 Application**
(Reporting, Analytics, Control)
- 5 Data Abstraction**
(Aggregation & Access)
- 4 Data Accumulation**
(Storage)
- 3 Edge Computing**
(Data Element Analysis & Transformation)
- 2 Connectivity**
(Communication & Processing Units)
- 1 Physical Devices & Controllers**
(The "Things" in IoT)



- But no matter the use case and number of layers, **the key building blocks of any IoT structure are always the same**, namely:
- **smart things**;
- **networks and gateways** enabling low-power devices (which is often the case in IoT) to enter the big Internet;
- the **middleware or IoT platforms** providing **data storage spaces and advanced computing engines** along with analytical capabilities; and
- **applications**, allowing end users to benefit from IoT and manipulate the physical world.

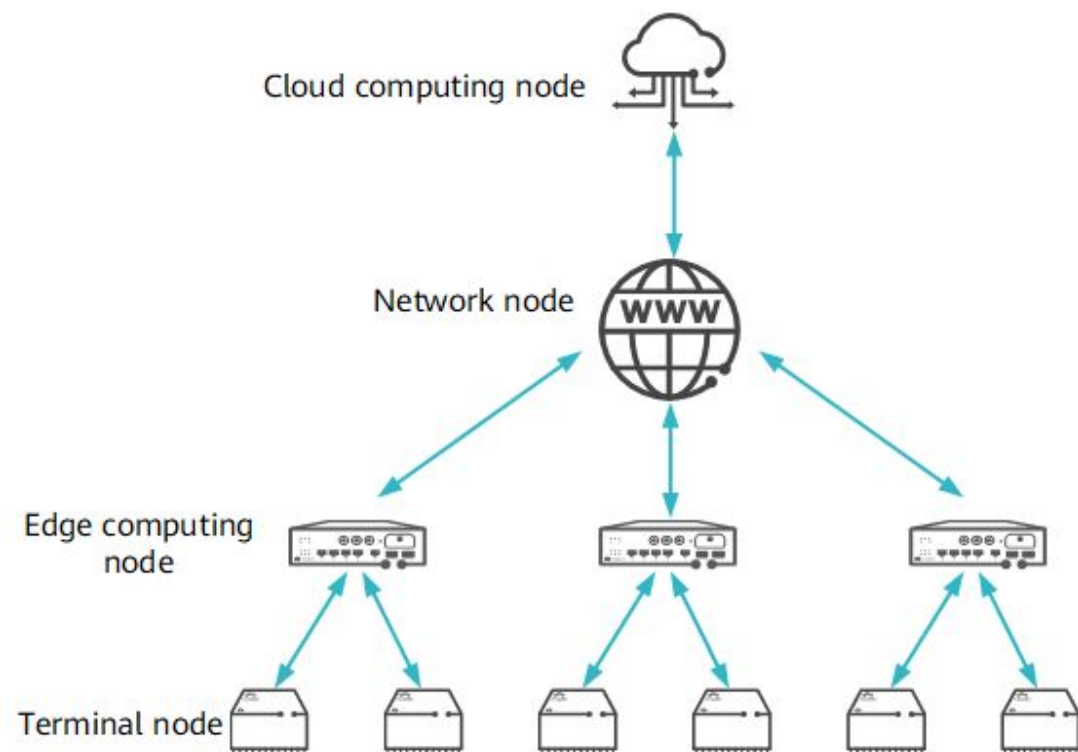
IoT KEY BUILDING BLOCKS



- **These elements make up the backbone of any IoT system** upon which effective, multi-layered architecture can be developed. Most commonly, these layers are:
- the **perception layer** hosting smart things;
- the **connectivity or transport layer** transferring data from the **physical layer / perception layer** to the cloud and vice versa via **networks and gateways**;
- the **processing layer** employing IoT platforms to accumulate and manage all data streams; and
- the **application layer** delivering solutions like analytics, reporting, and device control to end users.

- Besides the most essential components, there is three additional layers:
- the **edge or fog computing layer** performing data **preprocessing close to the edge**, where IoT things collect new information. Typically, edgy computing occurs on gateways;
- the **business layer** where businesses make decisions based on the data; and
- the **security layer** encompassing all other layers.

Often viewed as optional, these extra components none the less make an IoT project neatly fit modern business needs.



CLOUD LAYER

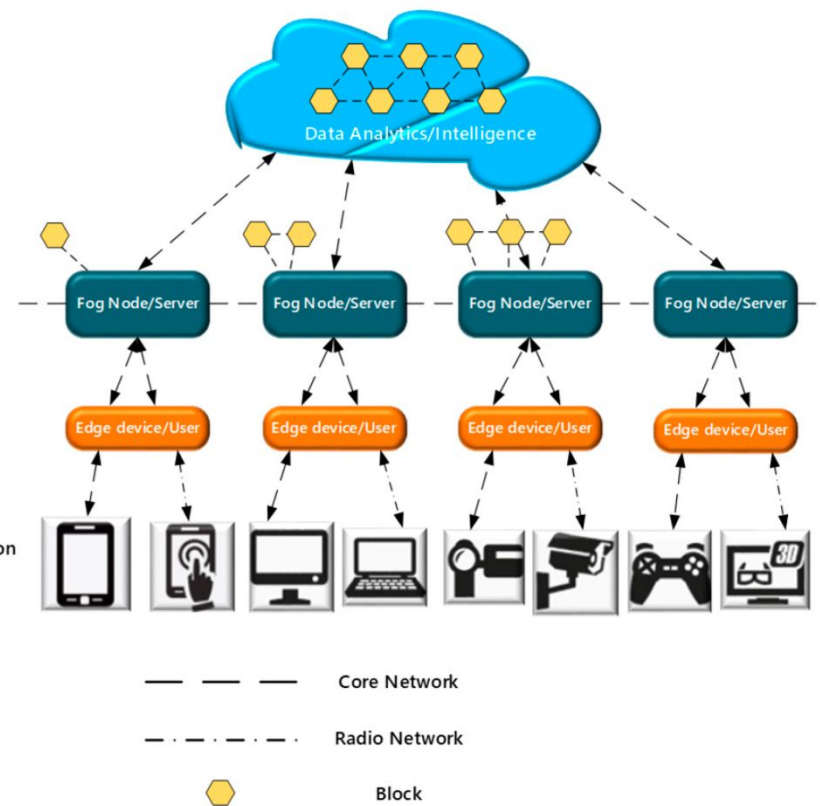
Big Data Processing
Data Warehousing
Business Logic
Blockchain Full Nodes

FOG LAYER

Local Network
Data Analysis & Reduction
Control Response
Virtualization/Standardization
Blockchain Light Nodes

EDGE LAYER

Real-time Data Processing
Micro Data Storage
Embedded Systems
At Source/On Premises Data Visualization



Perception layer or Sensing Layer

- **Perception layer: converting analog signals into digital data and vice versa**

The initial stage of any IoT system embraces a wide range of “things” or endpoint devices that **act as a bridge between the real and digital worlds**. They vary in form and size, from tiny silicon chips to large vehicles.

By their functions, **IoT things can be divided into the following large groups.**

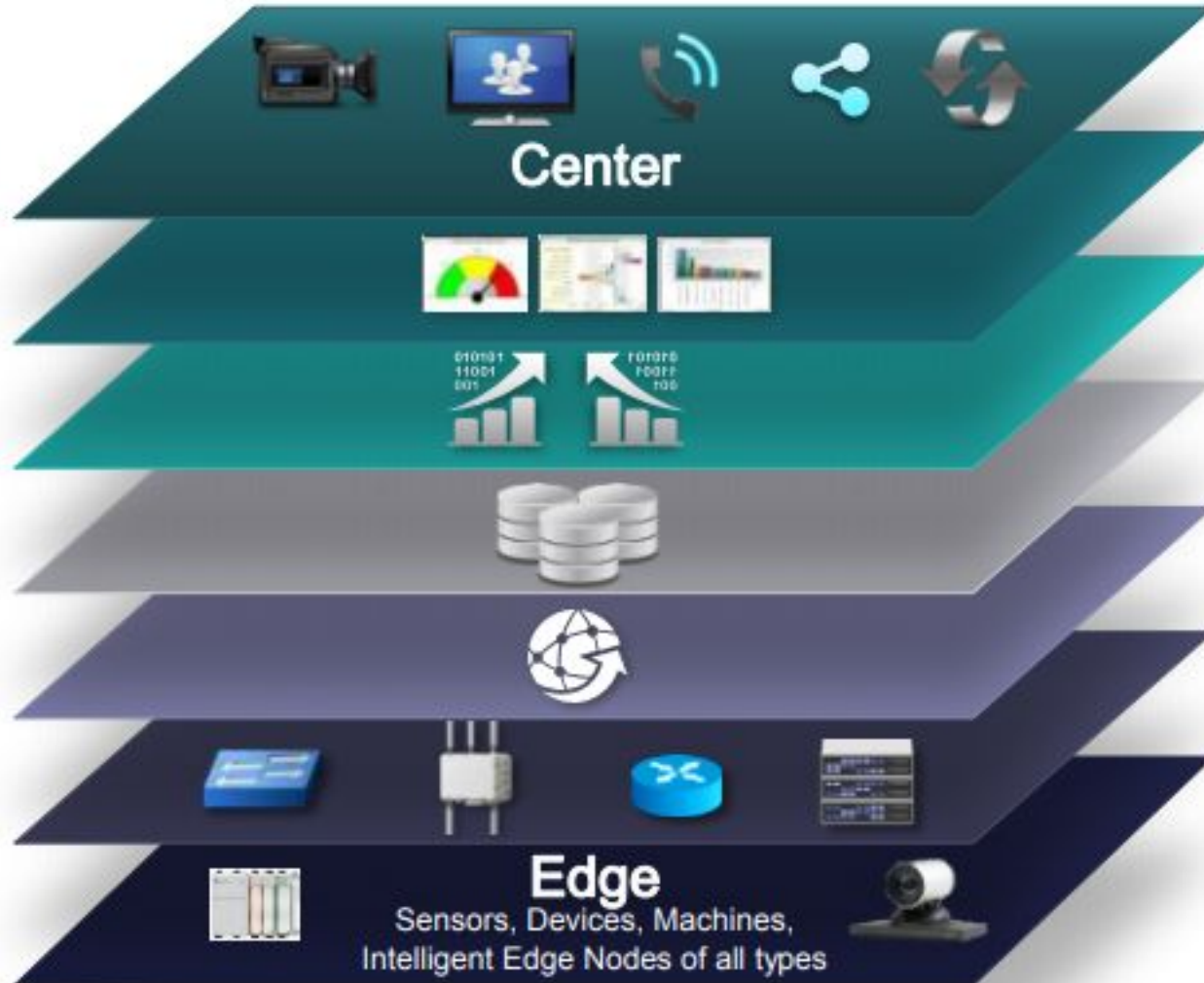
- **Sensors** such as probes, gauges, meters, and others. **They collect physical parameters like temperature or humidity, turn them into electrical signals, and send them to the IoT system.** IoT sensors are typically small and consume little power.
- **Actuators, translating electrical signals from the IoT system into physical actions.** Actuators are used in **motor controllers, lasers, robotic arms.**
- **Machines and devices** connected to sensors and actuators or having them as integral parts.

It's important to note that the architecture puts no restriction on the scope of its components or their location. The edge-side layer can include just a few “things” physically placed in one room or myriads of sensors and devices distributed across the world.

IoT World Forum Reference Model

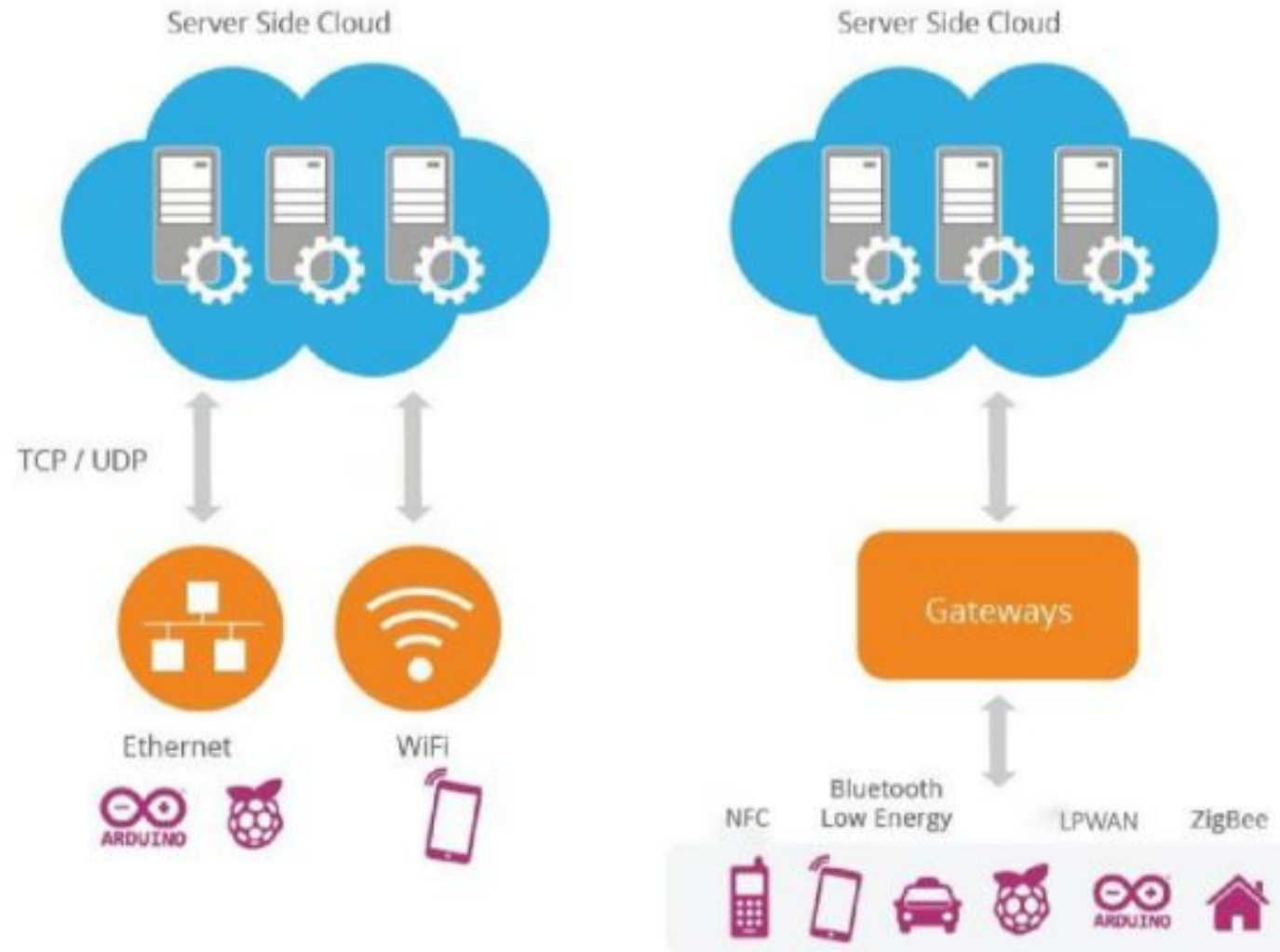
Levels

- 7 Collaboration & Processes**
(Involving People & Business Processes)
- 6 Application**
(Reporting, Analytics, Control)
- 5 Data Abstraction**
(Aggregation & Access)
- 4 Data Accumulation**
(Storage)
- 3 Edge Computing**
(Data Element Analysis & Transformation)
- 2 Connectivity**
(Communication & Processing Units)
- 1 Physical Devices & Controllers**
(The "Things" in IoT)



Connectivity layer: enabling data transmission

- The second level is in charge of **all communications across devices, networks, and cloud services** that make up the IoT infrastructure. The connectivity between the physical layer and the cloud is achieved in two ways:
- directly, **using TCP or UDP/IP stack;**
- via **gateways** — hardware or software modules performing translation between different protocols as well as encryption and decryption of IoT data.



Two key models of connectivity between physical and cloud levels in IoT. Sc

The communications between **devices** and **cloud services or gateways** involve different networking technologies.

- **Ethernet** connects stationary or fixed IoT devices like security and video cameras, permanently installed industrial equipment, and gaming consoles.
- **WiFi**, the most popular technology of wireless networking, is a great fit for data-intensive IoT solutions that are easy to recharge and operate within a small area. A good example of use is smart home devices connected to the electrical grid.
- **NFC (Near Field Communication)** enables simple and safe data sharing between two devices over a distance of 4 inches (10 cm) or less.
- **Bluetooth** is widely used by wearables for short-range communications. To meet the needs of low-power IoT devices, the Bluetooth Low-Energy (BLE) standard was designed. It transfers only small portions of data and doesn't work for large files.
- **LPWAN (Low-power Wide-area Network)** was created specifically for IoT devices. It provides long-range wireless connectivity on low power consumption with a battery life of 10+ years. Sending data periodically in small portions, the technology meets the requirements of smart cities, smart buildings, and smart agriculture (field monitoring).
- **ZigBee** is a low-power wireless network for carrying small data packages over short distances. The outstanding thing about ZigBee is that it can handle up to 65,000 nodes. Created specifically for home automation, it also works for low-power devices in industrial, scientific, and medical sites.
- **Cellular networks** offer reliable data transfer and nearly global coverage. There are two cellular standards developed specifically for IoT things. LTE-M (Long Term Evolution for Machines) enables devices to communicate directly with the cloud and exchange high volumes of data. NB-IoT or Narrowband IoT uses low-frequency channels to send small data packages.

NETWORKING TECHNOLOGIES USED in IoT

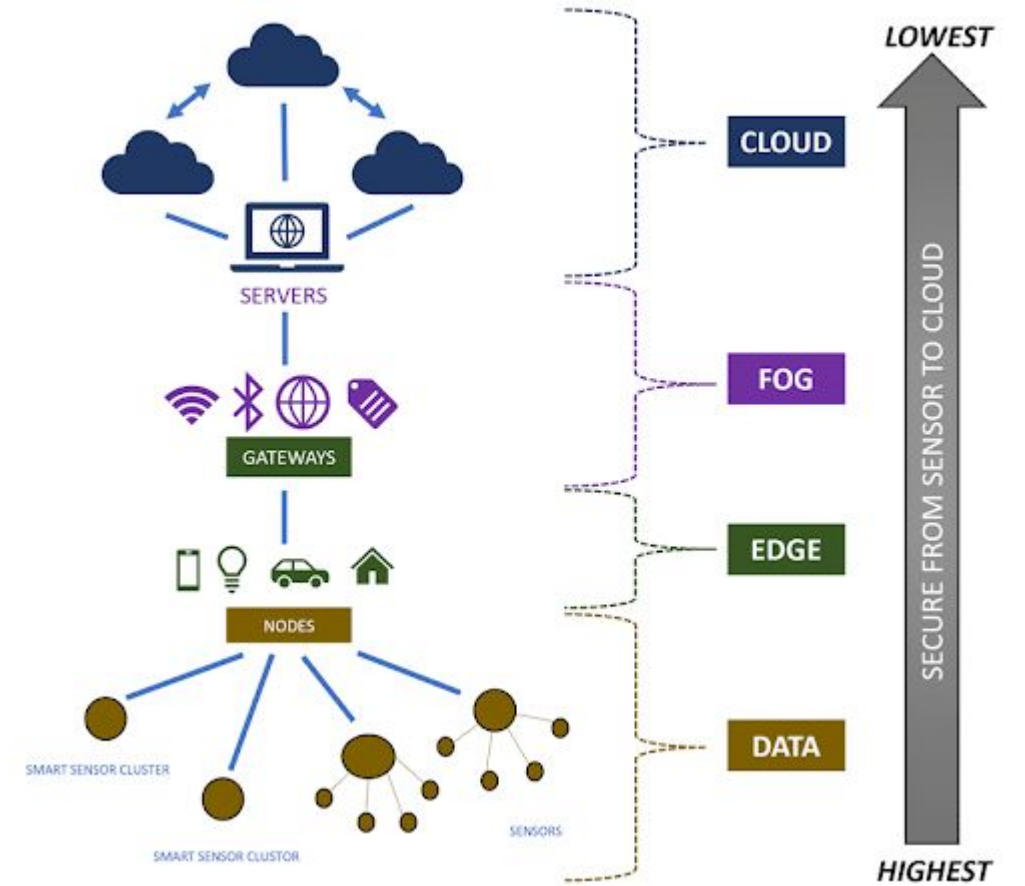
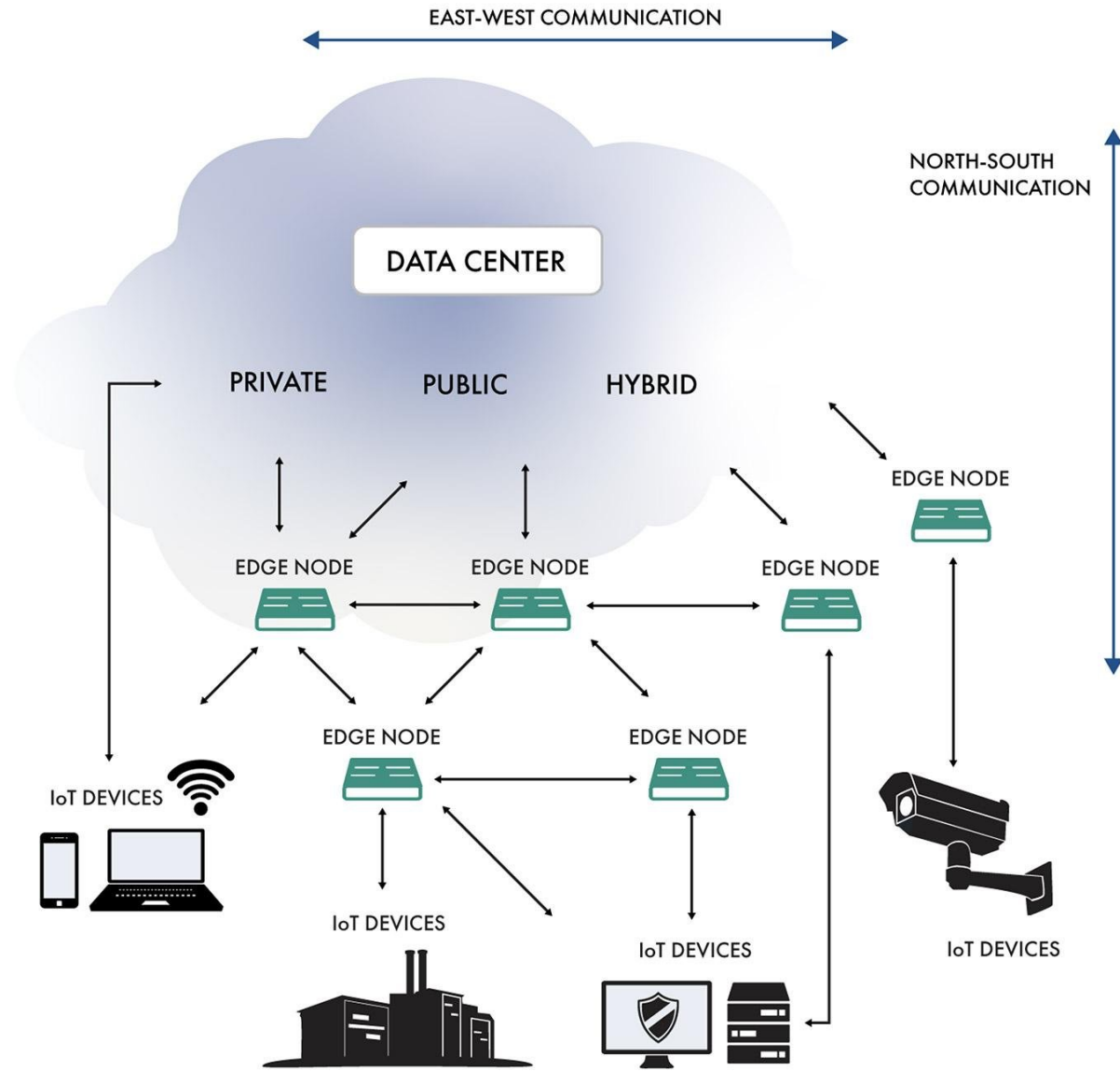
Network	Connectivity	Pros and Cons	Popular use cases
Ethernet	Wired, short-range	<ul style="list-style-type: none"> High speed Security Range limited to wire length Limited mobility 	Stationary IoT: video cameras, game consoles, fixed equipment
WiFi	Wireless, short-range	<ul style="list-style-type: none"> High speed Great compatibility Limited range High power consumption 	Smart home, devices that can be easily recharged
NFC	Wireless, ultra-short-range	<ul style="list-style-type: none"> Reliability Low power consumption Limited range Lack of availability 	Payment systems, smart home
Bluetooth Low-Energy	Wireless, short-range	<ul style="list-style-type: none"> High speed Low power consumption Limited range Low bandwidth 	Small home devices, wearables, beacons

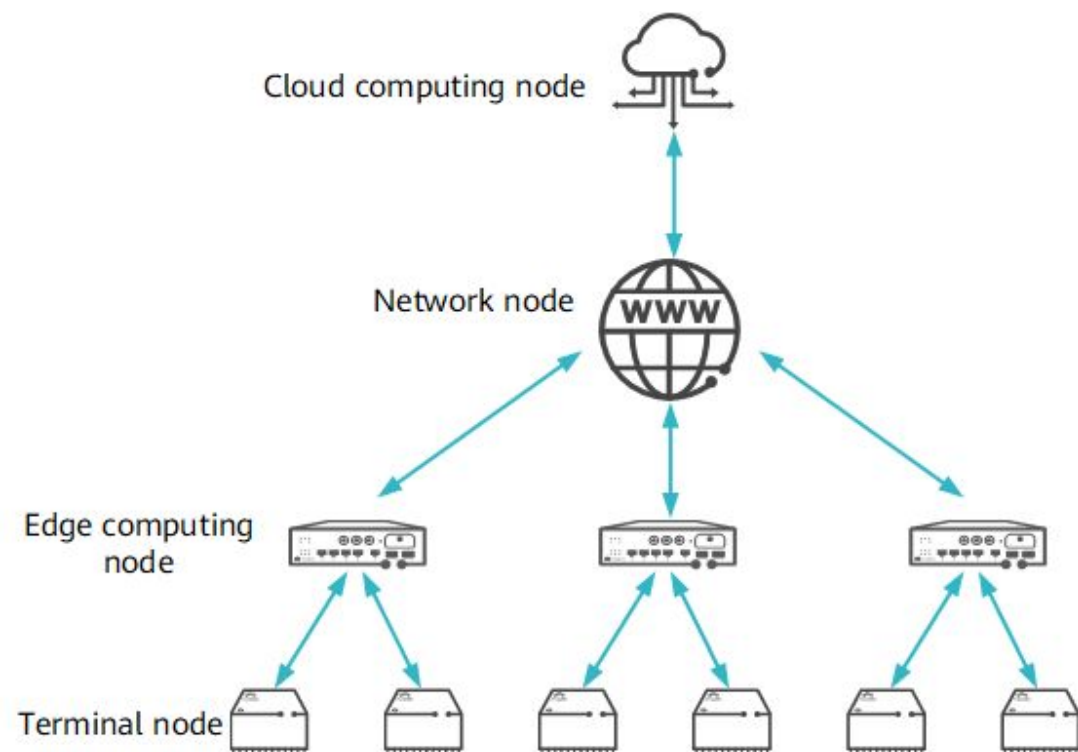
LPWAN	Wireless, long-range	<ul style="list-style-type: none"> ⊕ Long range ⊕ Low power consumption ⊖ Low bandwidth ⊖ High latency 	Smart home, smart city, smart agriculture (field monitoring)	
ZigBee	Wireless, short-range	<ul style="list-style-type: none"> ⊕ Low power consumption ⊕ Scalability ⊖ Limited range ⊖ Compliance issues 	Home automation, healthcare and industrial sites	
Cellular networks	Wireless, long-range	<ul style="list-style-type: none"> ⊕ Nearly global coverage ⊕ High speed ⊕ Reliability ⊖ High cost ⊖ High power consumption 	Drones sending video and images	

- Once parts of the IoT solution are networked, they still need **messaging protocols to share data across devices and with the cloud**. The most popular protocols used in the IoT ecosystems are:
- **DDS (the Data Distribution Service)** which directly connects IoT things to each other and to applications addressing the requirements of real-time systems;
- **AMQP (the Advanced Message Queuing Protocol)** aiming at peer-to-peer data exchange between servers;
- **CoAP (the Constrained Application Protocol)**, a software protocol designed for *constrained devices* — end nodes limited in memory and power (for example, wireless sensors). It feels much like HTTP but uses fewer resources;
- **MQTT (the Message Queue Telemetry Transport)**, a lightweight messaging protocol built on top of TCP/IP stack for centralized data collection from low-powered devices.

Edge or fog computing layer: reducing system latency

- This level is essential for enabling IoT systems to meet the speed, security, and scale requirements of the 5th generation mobile network or 5G. The new wireless standard promises **faster speeds**, **lower latency**, and the ability to handle many more connected devices, than the current 4G standard.
- The idea behind **edge or fog computing is to process and store information as early and as close to its sources as possible**. This approach allows for analyzing and transforming high volumes of real-time data locally, **at the edge of the networks**. Thus, you save the time and other resources that otherwise would be needed to send all data to cloud services. **The result is reduced system latency that leads to real-time responses and enhanced performance.**





CLOUD LAYER

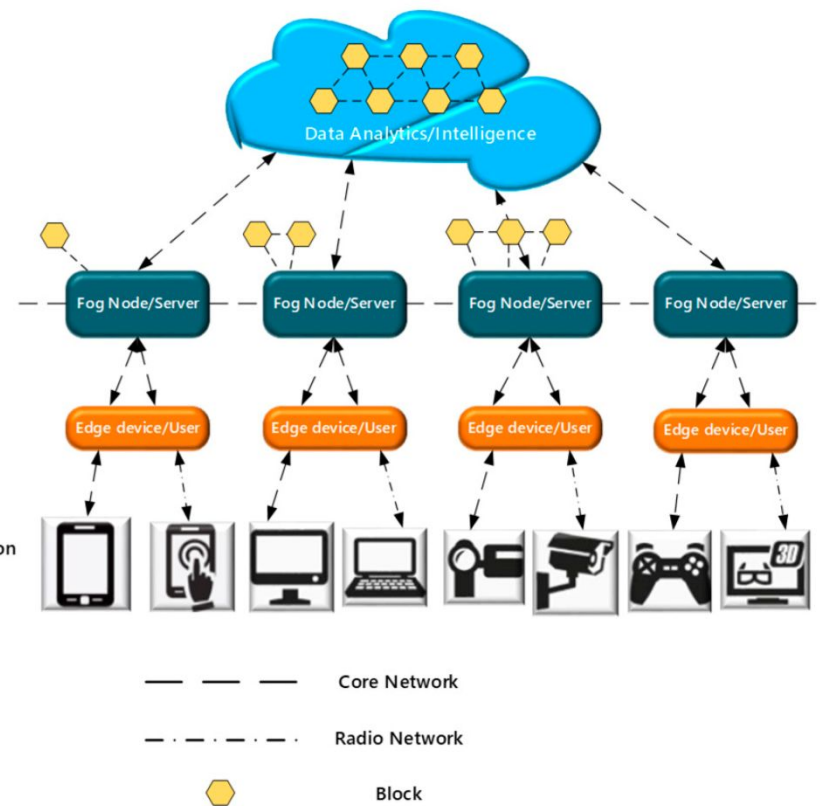
Big Data Processing
Data Warehousing
Business Logic
Blockchain Full Nodes

FOG LAYER

Local Network
Data Analysis & Reduction
Control Response
Virtualization/Standardization
Blockchain Light Nodes

EDGE LAYER

Real-time Data Processing
Micro Data Storage
Embedded Systems
At Source/On Premises Data Visualization



- **Edge computing** occurs on gateways, local servers, or other edge nodes scattered across the network. At this level, data can be:
 - **evaluated to determine if it needs further processing at higher levels,**
 - **formatted for further processing,**
 - **decoded,**
 - **filtered, and**
 - **redirected to an additional destination**

To sum up, the first three layers see data in motion, as it is constantly moving and altering. Only on hitting the next level, is data finally at rest and available for use by consumer applications.

Processing layer: making raw data useful

- The **processing layer accumulates, stores, and processes data** that comes from the previous layer. All these tasks are commonly handled via **IoT platforms and include two major stages**.
- **Data accumulation stage**
- The real-time data is captured via an **API** and put at rest to meet the requirements of non-real-time applications. The data accumulation component stage works as a transit hub between event-based data generation and query-based data consumption.
- Among other things, the stage defines whether data is relevant to the business requirements and where it should be placed. **It saves data to a wide range of storage solutions, from data lakes capable of holding unstructured data like images and video streams to event stores and telemetry databases.** The total goal is to **sort out a large amount of diverse data and store it in the most efficient way.**

Processing layer: making raw data useful

- **Data abstraction stage**
- Here, **data preparation is finalized so that consumer applications can use it to generate insights.** The entire process involves the following steps:
 - combining data from different sources, both IoT and non-IoT, including ERM, ERP, and CRM systems;
 - reconciling multiple data formats; and
 - aggregating data in one place or making it accessible regardless of location through data virtualization.
- Similarly, data collected at the application layer is reformatted here for sending to the physical level so that devices can “understand” it.

Together, the data accumulation and abstraction stages veil details of the hardware, enhancing the interoperability of smart devices. What’s more, they let software developers focus on solving particular business tasks — rather than on delving into the specifications of devices from different vendors.

Application layer: addressing business requirements

At this layer, **information is analyzed by software to give answers to key business questions**. There are hundreds of **IoT applications that vary in complexity and function**, using different technology stacks and operating systems. Some examples are:

- device monitoring and control software,
 - mobile apps for simple interactions,
 - business intelligence services, and
 - analytic solutions using machine learning.
- Currently, applications can be built right on top of IoT platforms that offer software development infrastructure with ready-to-use instruments for data mining, advanced analytics, and **data visualization**. Otherwise, IoT applications use APIs to integrate with middleware.

Business layer: implementing data-driven solutions

- The information generated at the previous layers brings value if only it results in problem-solving solution and achieving business goals. New data must initiate collaboration between stakeholders who in turn introduce new processes to enhance productivity.
- The decision-making usually involves more than one person working with more than one software solution. For this reason, the business layer is defined as a separate stage, higher than a single application layer.

Security layer: preventing data breaches

- It goes without saying that there should be a security layer covering all the above-mentioned layers. **IoT security** is a broad topic worthy of a separate article. Here we'll only point out the basic features of the safe architecture across different levels.
- **Device security.** Modern manufacturers of **IoT devices typically integrate security features both in the hardware and firmware installed on it.** This includes
 - embedded TPM (Trusted Platform Module) chips with cryptographic keys for authentication and protection of endpoint devices;
 - a secure boot process that prevents unauthorized code from running on a powered-up device;
 - updating security patches on a regular basis; and
 - physical protection like metal shields to block physical access to the device

Connection security. Whether data is being sent over devices, networks, or applications, it should be encrypted. Otherwise, sensitive information can be read by anybody who intercepts information in transit. **IoT-centric messaging protocols like MQTT, AMQP, and DDS may use standard Transport Layer Security (TLS) cryptographic protocol to ensure end-to-end data protection.**

- **Cloud security.** Data at rest stored in the cloud must be encrypted as well to mitigate risks of exposing sensitive information to intruders. Cloud security also involves **authentication and authorization mechanisms to limit access to the IoT applications.** Another important security method is device identity management to verify the device's credibility before allowing it to connect to the cloud.

The good news is that **IoT solutions from large providers like Microsoft, AWS, or Cisco come with pre-built protection measures including end-to-end data encryption, device authentication, and access control.** However, it always pays to ensure that security is tight at all levels, from the tiniest devices to complex analytical systems.

Difference between IoT and M2M

- **1. Internet of Things :**

-

IOT is known as the Internet of Things **where things are said to be the communicating devices that can interact with each other using a communication media.** Usually every day some new devices are being integrated which uses IoT devices for its function. These devices use various sensors and actuators for sending and receiving **data over the internet.** It is an ecosystem where the devices share data **through a communication media known as the internet.**

- **2. Machine to Machine :**

-

This is commonly known as Machine to machine communication. It is a concept **where two or more than two machines communicate with each other without human interaction using a wired or wireless mechanism.** M2M is an technology that helps the devices to connect between devices **without using internet.** M2M communications offer several applications such as security, tracking and tracing, manufacturing and facility management.

IoT Devices



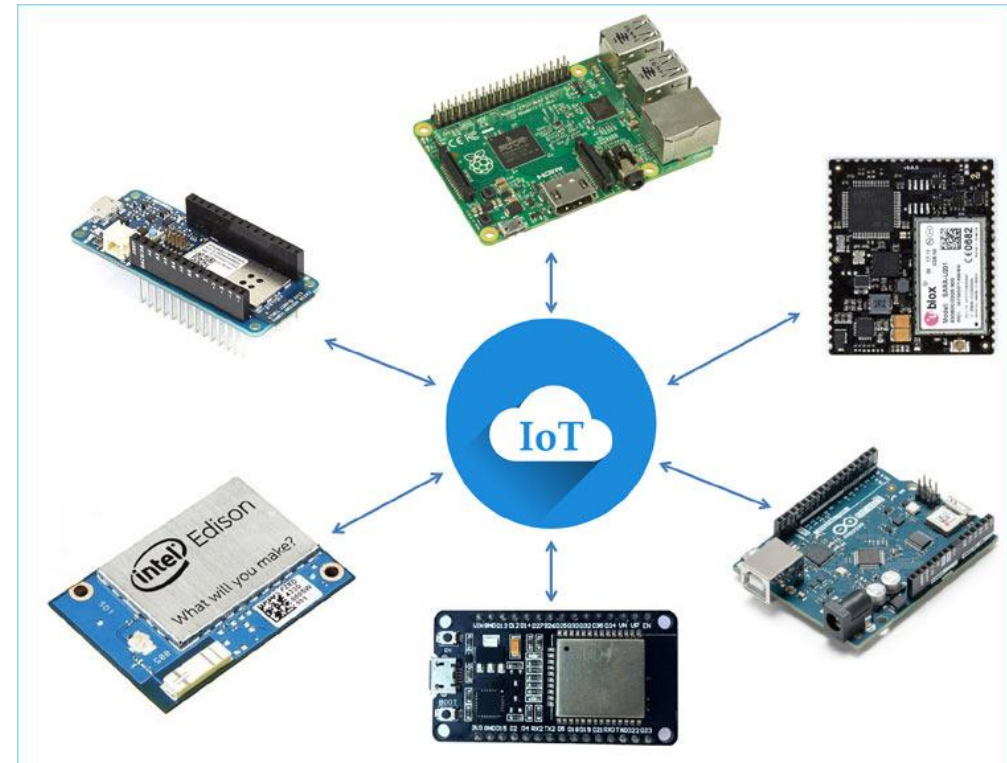
Raspberry Pi



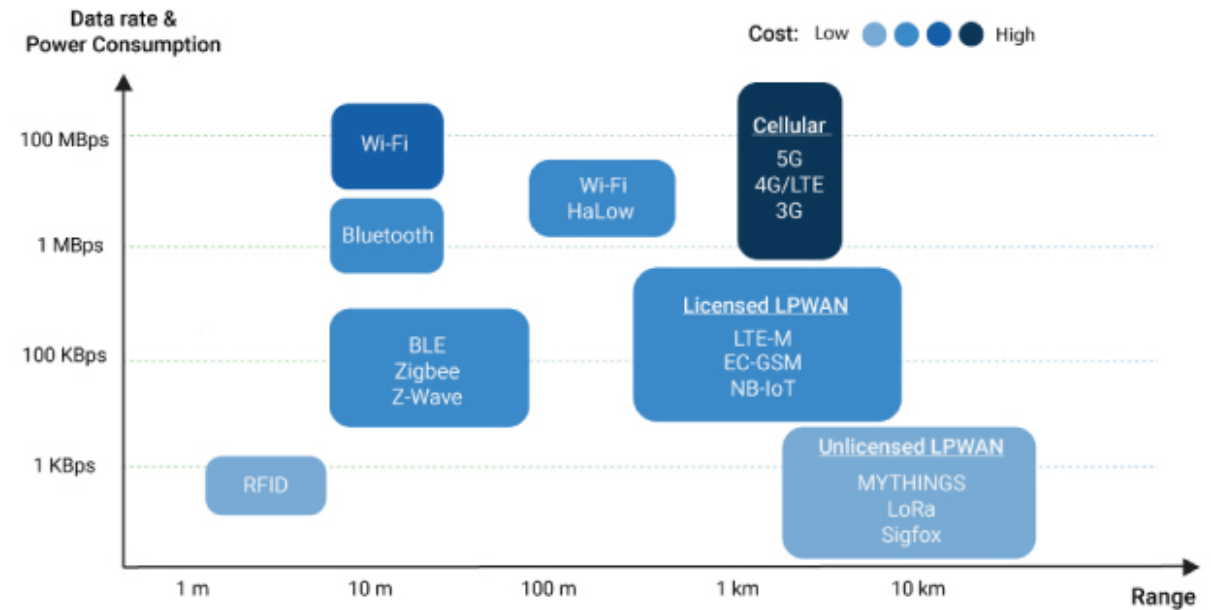
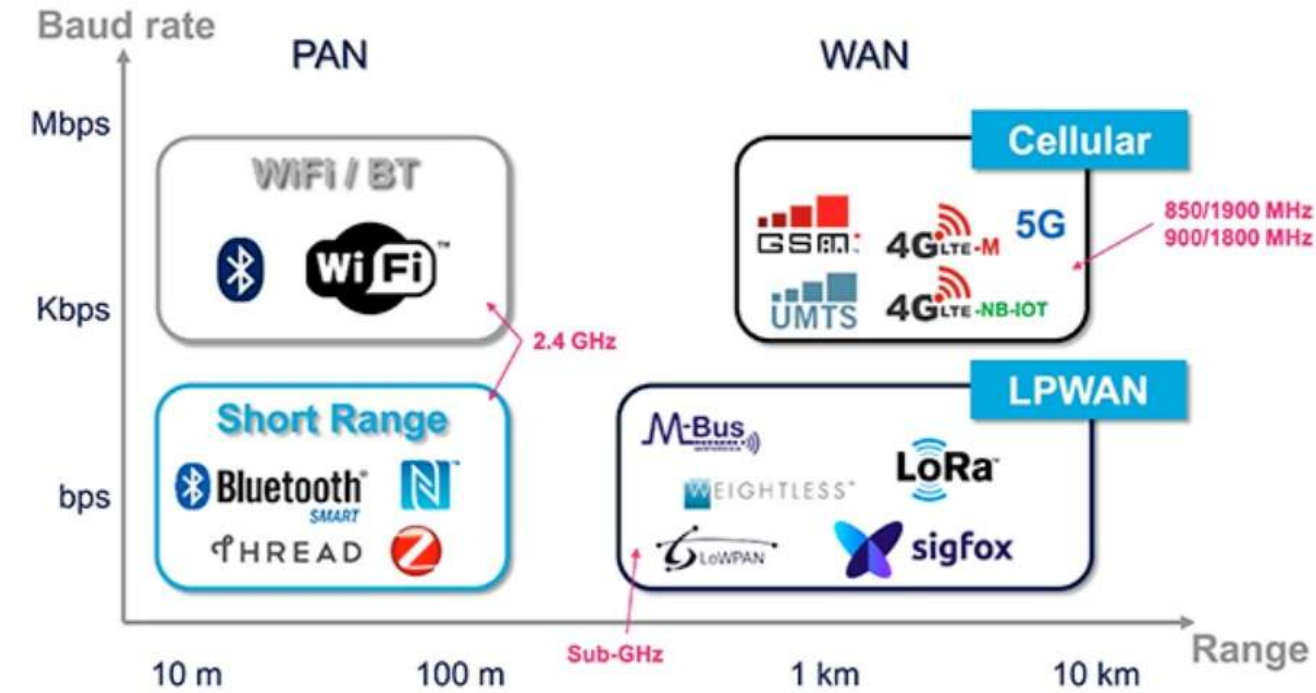
Arduino Nano



NodeMCU



Connectivity



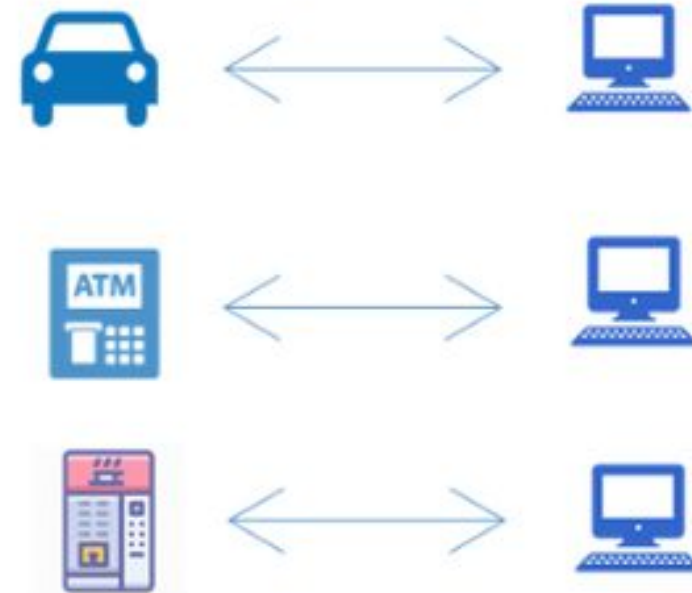
M2M

- Machine-to-machine communication, or M2M, is exactly as it sounds: **two machines “communicating,” or exchanging data, without human interfacing or interaction.** This includes serial connection, powerline connection (PLC), or **wireless communications in the industrial Internet of Things (IoT).** Switching over to wireless has made M2M communication much easier and enabled more applications to be connected.
- In general, when someone says M2M communication, they often are referring to **cellular communication for embedded devices.** **Examples of M2M communication in this case would be vending machines sending out inventory information or ATM machines getting authorization to dispense cash.**
- As businesses have realized the value of M2M, it has taken on a new name: **the Internet of Things (IoT).** IoT and M2M have similar promises: to fundamentally change the way the world operates. Just like IoT, M2M allows virtually any sensor to communicate, which opens up the possibility of systems monitoring themselves and automatically responding to changes in the environment, with a much reduced need for human involvement. **M2M and IoT are almost synonymous—the exception is IoT (the newer term) typically refers to wireless communications, whereas M2M can refer to any two machines—wired or wireless—communicating with one another.**

Differences between IoT and M2M

- What are M2M and IoT?
- **M2M** refers to machine-to-machine communication. Machines have the ability to connect to a common network and exchange data over the network. M2M connectivity is often business driven and using mobile (2G/3G/4G) networks to enable remote access of **specific business applications**. **M2M networks are often closed user groups and serve one specific application**. Examples are fleet tracking for a company, or ATM machines for a bank.
- **IoT** (Internet of Things) refers to a wider concept, which includes a variety of equipment and devices, from personal devices, consumer electronics, to business machines and vehicles. By enabling communications, the devices become more sophisticated and useful, often transforming their function entirely. Communications can be established using a range of technologies, from mobile networks (3G/4G) and WIFI, to Bluetooth and even wired Internet access. Examples are fitness trackers, home automation (Alexa etc.), but also industrial automation, sensors, connected vehicles etc. **So in summary, M2M is a subset of IoT, whereby M2M solutions are often narrowly defined for a specific function using cellular networks, while IoT based systems have much more flexibility and connectivity options.**

IoT vs M2M



M2M versus the IoT

M2M	IoT
M2M is about direct communication between machines.	The IoT is about sensors automation and Internet platform.
It supports point-to-point communication.	It supports cloud communication.
Devices do not necessarily rely on an Internet connection.	Devices rely on an Internet connection.
M2M is mostly hardware-based technology.	The IoT is both hardware- and software-based technology.
Machines normally communicate with a single machine at a time.	Many users can access at one time over the Internet.
A device can be connected through mobile or other network.	Data delivery depends on the Internet protocol (IP) network.

Key Difference

- The Internet of things (IoT) is the network of physical devices, vehicles, home appliances, and others embedded with electronics, software, sensors, actuators, and connectivity which enables these things to connect and exchange data, creating opportunities for more immediate integration between the physical world and computer-based systems.
- With regard to M2M (machine to machine) communication, there are several differences. **The M2M communications are used to connect machines to machines within the same cellular network. The IoT is used to connect machines to machines across different cellular networks.**

Differences between IoT and M2M

Basis	IoT	M2M
Abbreviation	Internet of Things	Machine to Machine
Communication	IoT sensors automation	Communicates directly between machines
Connection	The connection is via using various communication types	Point to Point Connection
Communication protocols	HTTP, Ftp, Telnet, etc are used	Communication technology techniques and traditional protocols are used.
Intelligence	Objects are responsible for decision making	Observation of some degree of intelligence
Technology	Hardware and Software based technology	Hardware-based technology

Data Delivery	Depending on Internet protocol	Devices can be connected through mobile or other networks
Internet Connection	Active Internet connection is required	Devices do not rely on internet connection
Scope	Many users can connect at a time over the Internet	Communicate with a single machine at a time
Business Type	B2C(Business to Customers) and B2B(Business to Business)	Only B2B(Business to Business) is used
Open API support	IoT supports open API Integrations	M2M does not support open API
Data Sharing	Data is shared with applications that tend to improve the end-user experience	Data is shared with the communication parties themselves.

- **Q. Is M2M an IoT?**

A. M2M is a subset of IoT that deals explicitly with connections between machines. M2M communication allows devices to communicate without direct human input. M2M is all about connecting machines to the Internet to communicate with each other, manage their tasks and data, and generally “do business” with one another. Likewise, IoT is all about connecting devices to the Internet to be monitored remotely by individuals or enterprises.

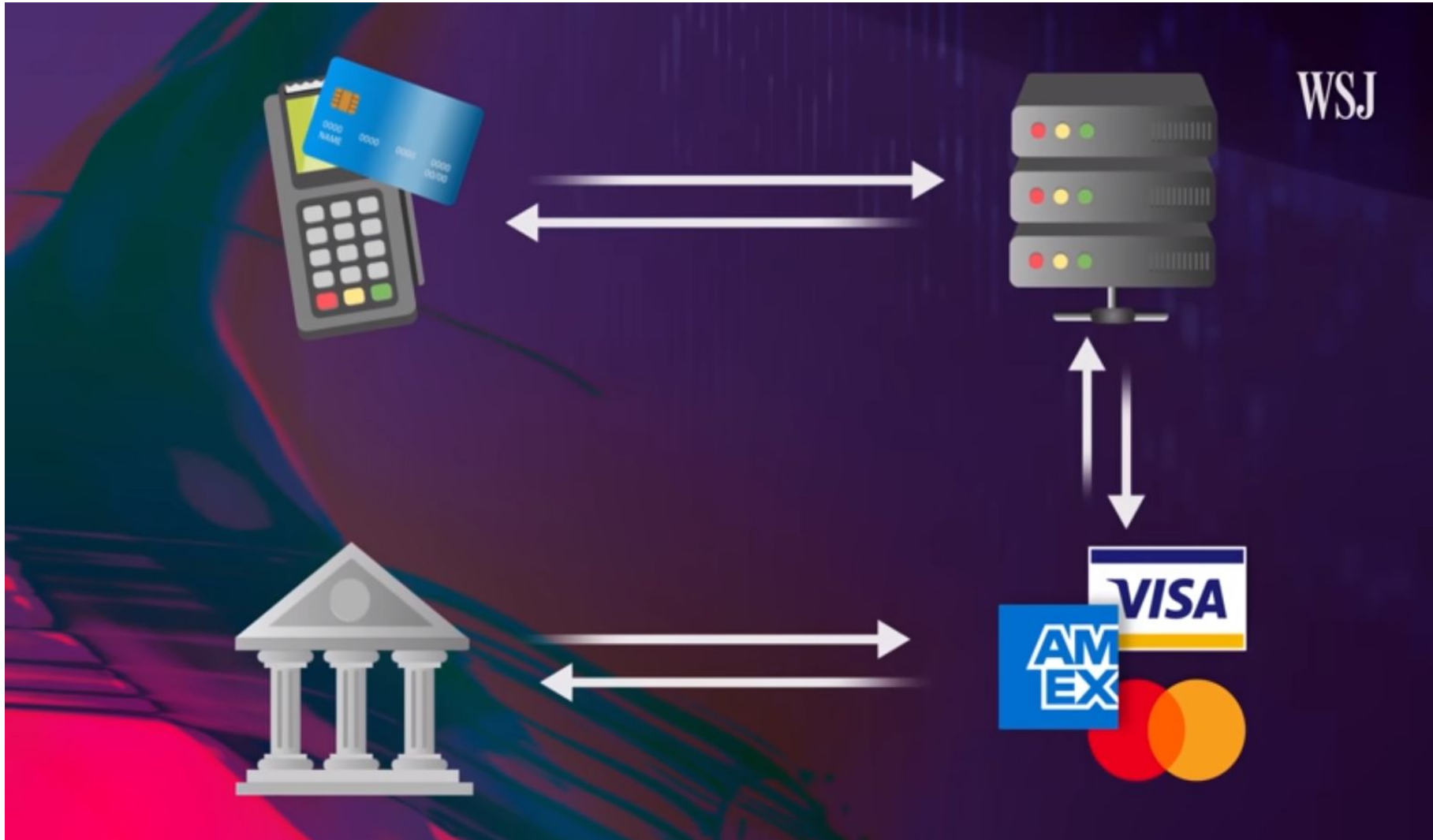
- **Q. What is the primary purpose of M2M?**

The primary purpose of Machine to Machine communications is to enable machine-to-machine (M2M) interaction, which has been proposed as a specialized application of the Internet of Things (IoT).

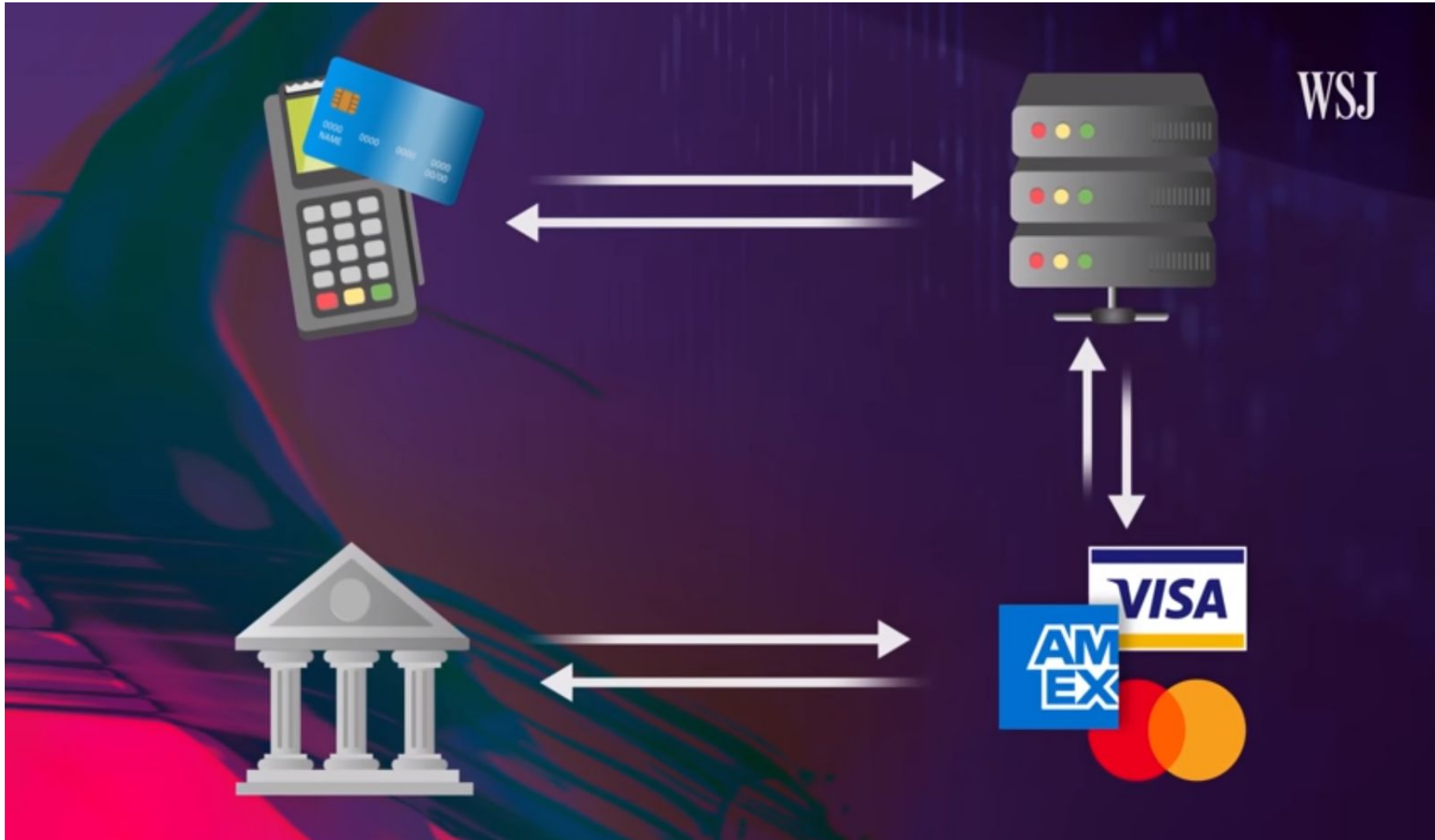
- **Q. What is the primary purpose of IoT?**

The primary purpose of IoT is to create an intelligent environment for the users. IoT brings a revolution in the process of manufacturing and management in different sectors. It provides a new way to interact with the objects in the real world by capturing real-time data from them.

Example: NFC (specific of RFID Communication) Tap-to-Pay Works



Tap-to-Pay Works



RFID

- Hotel room key

- Tap to pay
 - <https://www.youtube.com/watch?v=S7dWigl7Soc>