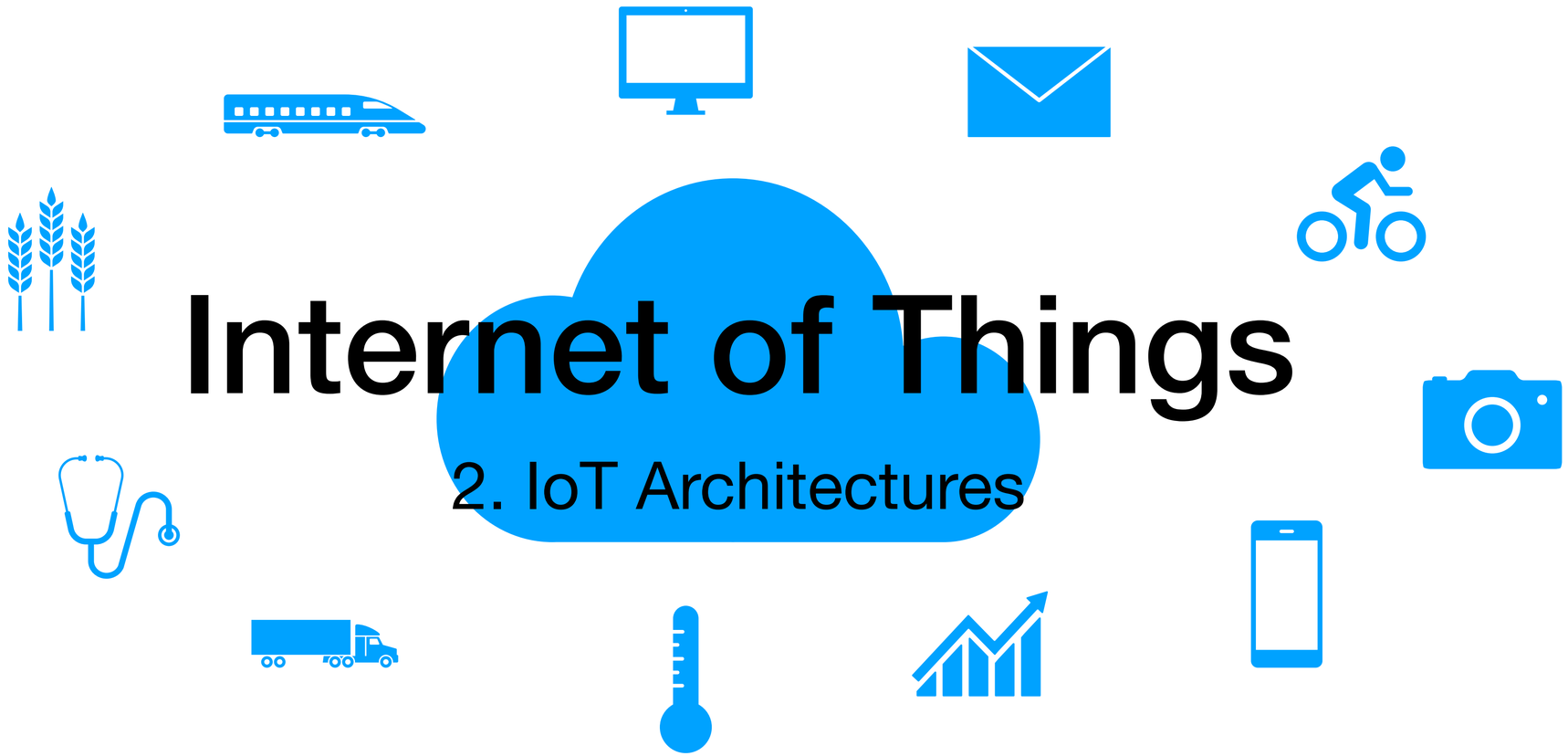


Internet of Things

2. IoT Architectures

M.Sc. Computer Science 2024-2025

Viviana Arrigoni

2.1 Drivers of IoT Architectures

IT and IoT

- Two key differences between IT and IoT:

Data

IT systems are mostly concerned with reliable and continuous support for business applications such as email, web, database, etc., IoT is all about the data generated by sensors and how data is used.

Smart Devices

IT systems are mostly running on computers and servers. A IoT system is also composed by ordinary objects that are made “smart” and produce a lot of data.

- What should we be taking into consideration when designing an IoT architecture?

Scale

- The scale of typical IT networks is very smaller than the scale of IoT endpoints.
- IPv4 address space has reached exhaustion and is unable to meet IoT's scalability requirements.
  - Scale can be met only by using IPv6

Security

- IoT devices, especially those on wireless sensor networks (WSNs), are often physically exposed to the world.
- Traditional models of IT security do not always work for the new attack vectors
  - IoT devices vary widely in hardware, software, and communication protocols
  - Many devices have limited computational resources, making it difficult to implement strong encryption, firewalls, or intrusion detection systems.
- Every IoT endpoint node on the network must be part of the overall security strategy and must support device-level authentication and link encryption.
- 3:30 p.m. on December 23, 2015, the Ukrainian power grid experienced an unprecedented cyber attack that affected approximately 225,000 customers

Constrained devices and networks

- Most IoT sensors are designed for a single job, and they are typically small and inexpensive
    - Limited power, CPU, memory, and they transmit only when there is something important to say
- Large scale of the devices + large, uncontrolled environments = lossy network supporting low data rates
    - This is very different from traditional IT networks, which enjoy Gb connections and endpoints with powerful CPUs.

Data

- In IoT, data enables business to deliver new IoT services that enhance the user experience and reduce costs.
- Most IoT generated data is unstructured.
- When all the data is combined, it can be difficult to manage and analyse it effectively.
- Data analytics capabilities need to be distributed throughout the IoT network, from the edge to the cloud. In traditional IT networks, analytics and applications typically run only in the cloud.

Compute and network assets in IoT

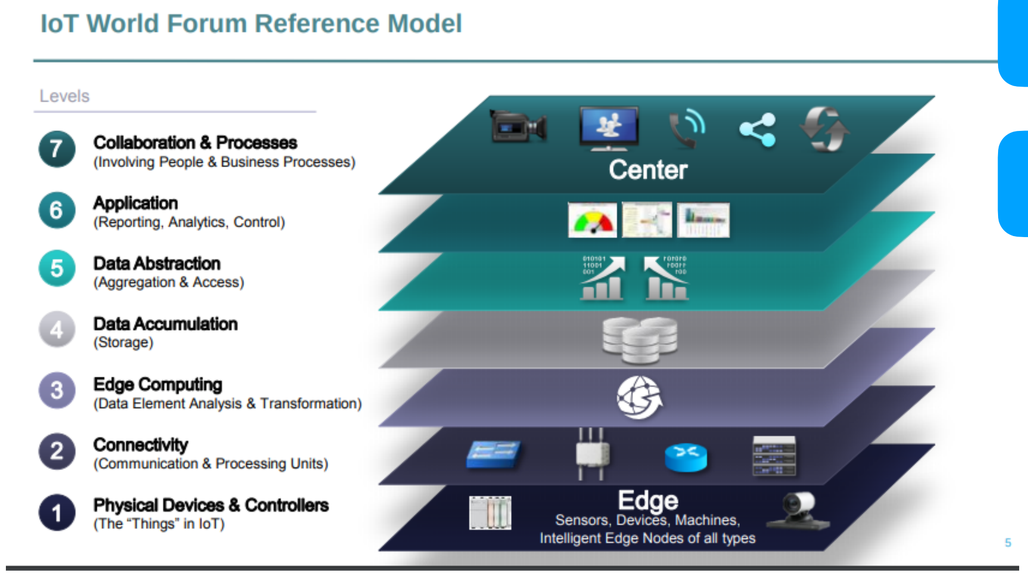
- Compute and network assets used in IoT can be very different from those in IT environments.
- This is because IoT devices are deployed in various environments, which belong to different OT environments.
- Deciding what devices to deploy is key factor:
  - Thermometers can be place near the furnaces of a steel mill, or used for cold chains
  - Devices can be placed in environments with very different humidities, or even underwater.
  - Some devices are subject to constant kinetic vibrations, or to abrupt ones

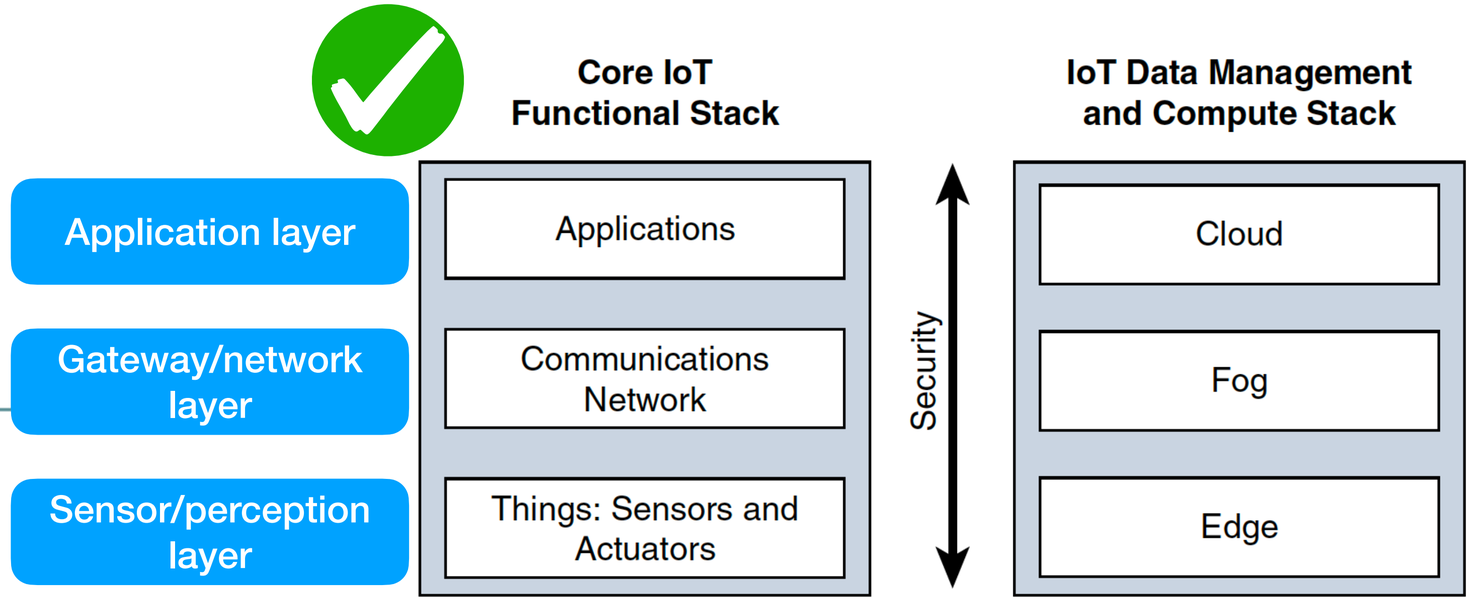
2.2 IoT Architecture layers

IoT architecture

Yaqoob et al. 2017.

Application Layer	• Provide services required by customer. • Span various sectors such as Smart Home, Smart Healthcare, Smart Transportation, etc.
Management and Service Layer	• Provide IoT application programmers with hardware abstraction. • Provide features for analysing data.
Gateway and Network Layer	• Connect IoT devices with cloud technologies. • Number of communication protocols like Zigbee, WiFi, Bluetooth, etc. work at this layer.
Sensor and Connectivity Layer	• Device sense the enviroment. • Transfers event through secure channel.





Sensor layer

- **Sensor layer** or **perception layer** consists of **heterogeneous** sensors and actuators, that sense the environment, collect information for processing to gain useful insights.
    - Different kind of sensors deployed, like temperature, motion, humidity, etc.
- The sensor layer digitalizes, creates a channel, and transfers data to the next layer.
- It is the major source of big data to be processed by next layers.

Sensor layer - classification (1)

Smart devices can be:

- **Battery powered or power connected:**
  This classification is based on whether the object carries its own energy supply or receives continuous power from an external power source.
  - Battery-powered things can be moved more easily than line-powered objects.
  - Batteries limit the lifetime and amount of energy that the object is allowed to consume, thus driving transmission range and frequency.

An outdoor battery-powered humidity sensor

A smart motion camera, power connected

Sensor layer - classification (2)

Smart devices can be:

- **Mobile or static:**
  This classification is based on whether the “thing” should move or always stay at the same location.
  - A thing can be moved because it is located on a moving object.
  - The frequency of a movement may also vary, from occasional to permanent.

Self-driving car

Smart thermometer

Sensor layer - classification (3)

Smart devices can have:

- **Low or high reporting frequency:**
  How often the object should report monitored parameters.
  - A color sensor can report data once a day.
  - A motion sensor may report acceleration several hundred times per second.
  - Higher frequencies drive higher energy consumption.

Sensor layer - classification (4)

Smart devices can produce:

- **Simple or rich data:**
  This classification is based on the quantity of data exchanged at each report cycle.
  A humidity sensor in a field may report a simple daily index value, while an engine sensor may report hundreds of parameters, from temperature to pressure, gas velocity, compression speed, carbon index, and many others.
  - Richer data typically drives higher power consumption.

Sensor layer - classification (5)

- **Sensing range:**
  Sensors might have smaller or larger sensing range, i.e., the region around the sensor where the sensor can sense and take measurements.
- **Transmission range:**
  Sensors might have smaller or larger transmission range, i.e., the region around the sensor where signal coming from the sensor can be heard.
  - Depending on the transmission range of the sensors involved, we can classify IoT networks (WSNs).

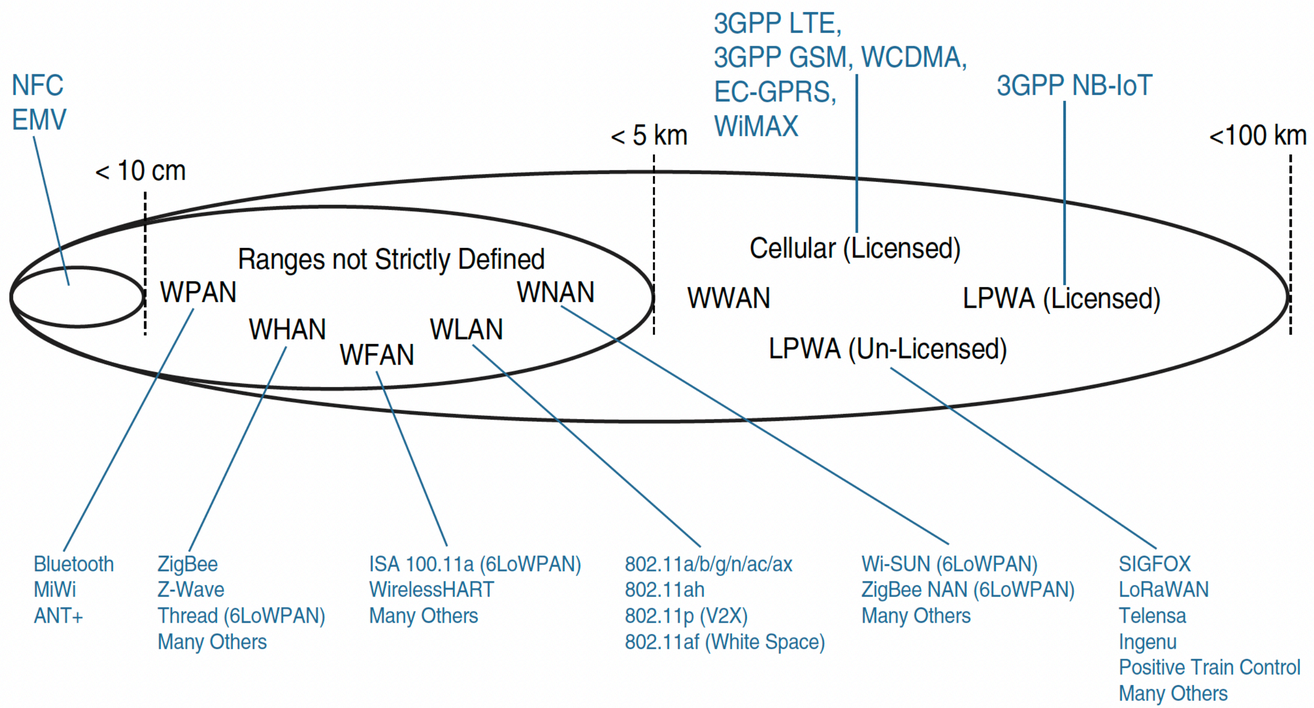
IoT architecture: gateway and network layer

- The **gateway and network layer** transfers data created by the sensor layer through secure channels.
- Uses different wireless technologies to transmit data.
- Can also perform data storage and processing.

Network layer - M2M

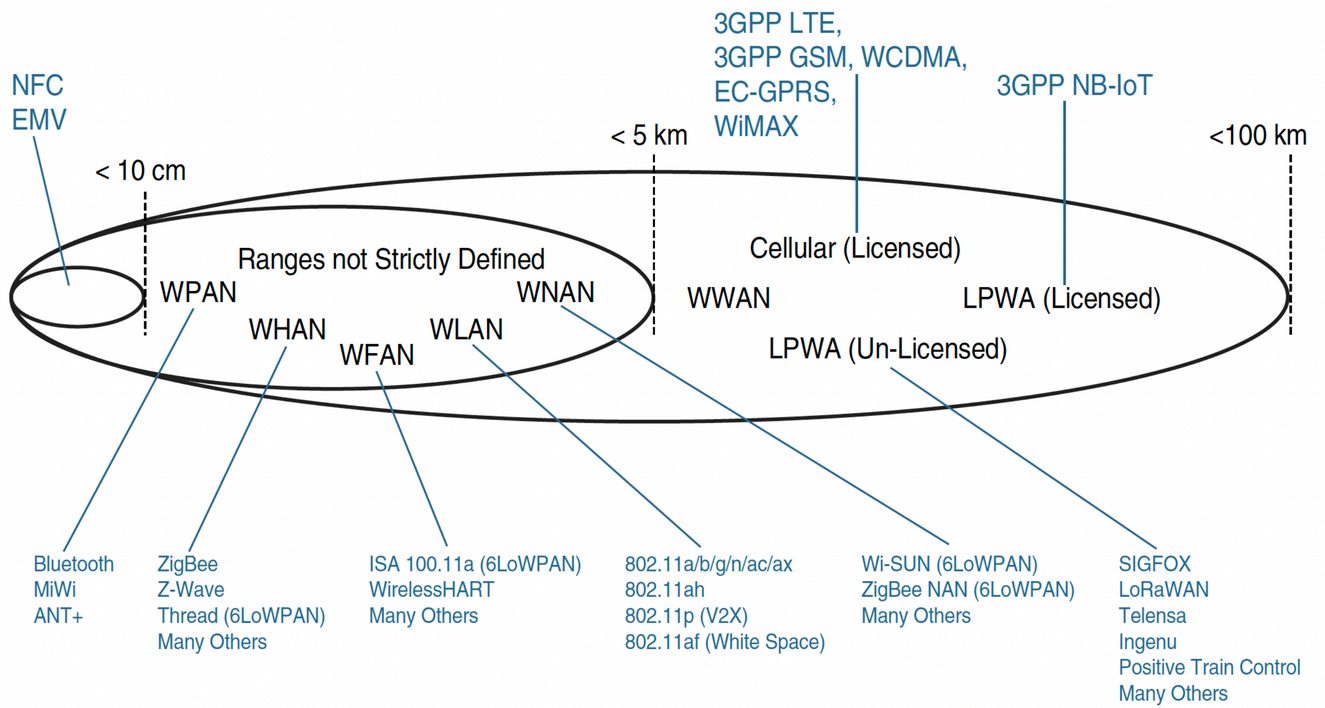
- In IoT systems, devices communicate with one another to cooperate and rely on machine-to-machine (M2M) communication protocols.
- Smart objects communicating with one another form autonomous networks which can be classified based on the transmission ranges of the devices.
- M2M communication is mostly wireless and achieved by different protocols and allows for autonomous communication.

Network layer - M2M network types (1)



- **PAN (personal area network).** Scale of a few meters. This is the personal space around a person (Bluetooth).
- **HAN (home area network).** Scale of a few tens of meters (ZigBee, BLE).
- **NAN (neighborhood area network).** Scale of a few hundreds of meters. The term NAN is often used to refer to a group of house units from which data is collected.
- **FAN (field area network).** Scale of several tens of meters to several hundred meters. FAN typically refers to an outdoor area larger than a single group of house units.
- **LAN (local area network).** This term is very common in networking, and also commonly used in the IoT space when standard networking technologies (such as Ethernet) are used

Network layer - M2M network types (2)

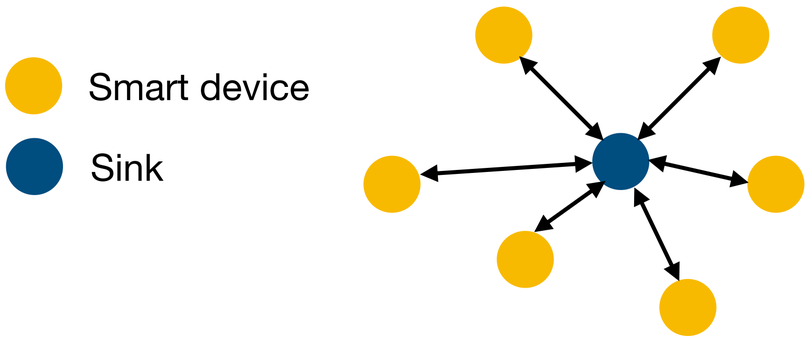


- **LPWA (Low-Power wise-area network).** Wireless telecommunication wide area network designed to allow long-range communication at a low bit rate between IoT devices.
- **Cellular networks.** A telecommunications network where the link to and from end nodes is wireless and the network is distributed over land areas called cells, each served by at least one fixed-location transceiver (such as a base station).

Network layer - M2M Topologies (1)

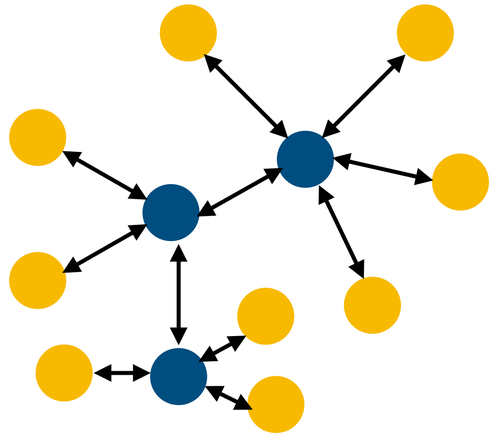
- Different M2M protocols support different topologies that define the connections between smart devices.

Star topology



- A central node act as a sink (“coordinator” in IEEE 802.15.4 standard) and peripheral nodes are connected to it via direct link without being connected with each other.
- The sink is usually the gateway connected to the outer world.

Clustered Star topology



- When there are multiple sinks connected directly or indirectly to a gateway, we have a clustered star topology

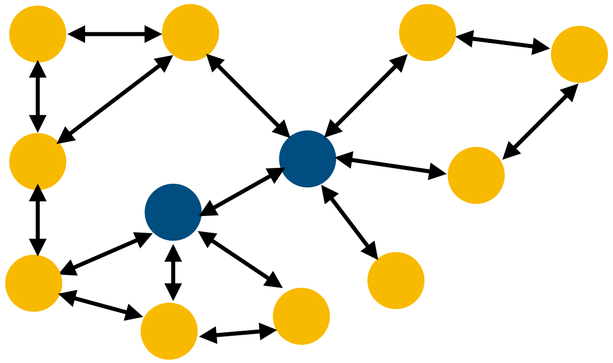
Network layer - M2M Topologies (2)

- Different M2M protocols support different topologies that define the connections between smart devices.

Smart device

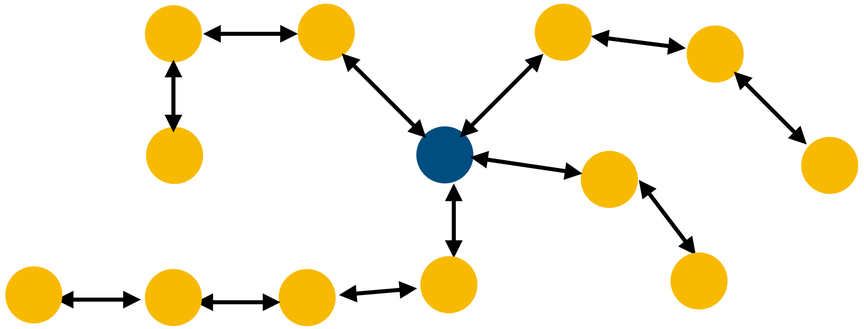
Sink

Mesh topology



- Devices are connected to each other in a multi-hop fashion. Some of them are connected to the seed

Hierarchical tree topology



- Connections form a tree in which the sink is the root and peripheral nodes are connected in layers via direct links

Network layer - M2M Topologies (3)

- The choice of what topology (and protocol) to use highly depends on the application of the IoT system.
- Mesh topology are more reliable, since they have redundant links and the failure of one link does not necessarily disconnect the network.
- Nevertheless, keeping more connections influences the power consumption of the devices.
- Topologies depend also on the **transmission range** of the devices (trivially, two devices are connected by a link in a topology only if they are within the transmission range of one another)
  - In remote areas when devices can possibly be very far apart from each other, it might be necessary to leverage satellite communication.

Network layer - Gateways (1)

- Data collected from a smart object may need to be forwarded to a central station (i.e., a server, a data center, or more generally the cloud) where data is processed.

- The station is often far from the smart object, and requires an Internet connection.

- Many small IoT devices do not support the a full TCP/IP stack, i.e., they cannot access the internet directly (when they do, they form a peer-to-peer network).

  - Gateways are responsible for **bridging** the sensors/smart devices with the internet to transmit data to a central station, but they can also act as edge servers!

Network layer - Gateways (2)

IoT Edge Computing Gateway - MFX-1

Datasheet:
https://mainflux.com/downloads/Mainflux_Labs-IoT_Gateway_Datasheet_and_Pricing.pdf

~500€

Processor	I.MX6 single to quad core Arm Cortex A9 (800MHz)
Memory and storage	Up to 2GB DDR3* eMMC (8GB)**
Network	2x Ethernet Wifi BLE 5.0 LTE Additional slots for networking options

* Double Data Rate 3 Synchronous Dynamic Random Access Memory with hight bandwidth
** embedded MultiMediaCard - soldered directly on the motherboard differently from SSDs

Network layer - IoT Gateways vs Routers

Both IoT Gateways and Routers are networking devices laying in the Internet space

IoT Gateways:

- Support wide protocol variety (protocol translation, remote monitoring and control)
- Low power consumption, edge computing (local processing capabilities)
- Handle large amounts of data (data filter and processing, data collection and aggregation, security management)
- Packet forwarding (connectivity management)

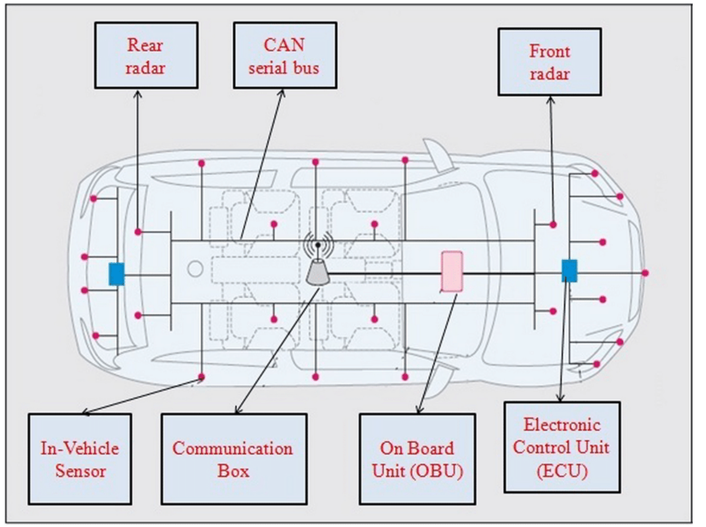
Routers:

- Limited protocol support
- Not optimised for low power
- Packet forwarding

Network gateways exist also for non-IoT networks to enable communication between different networks

Network layer - Gateways example 1

- In most cases, smart objects are static or mobile within a limited area, while the gateway is often static.
  However, some IoT technologies do not apply this model.
  - Dedicated Short-Range Communication (DSRC) allows vehicle-to-vehicle and vehicle-to-infrastructure communication.
  - In this case, the smart object's position relative to the gateway is static, as the car includes sensors and one gateway



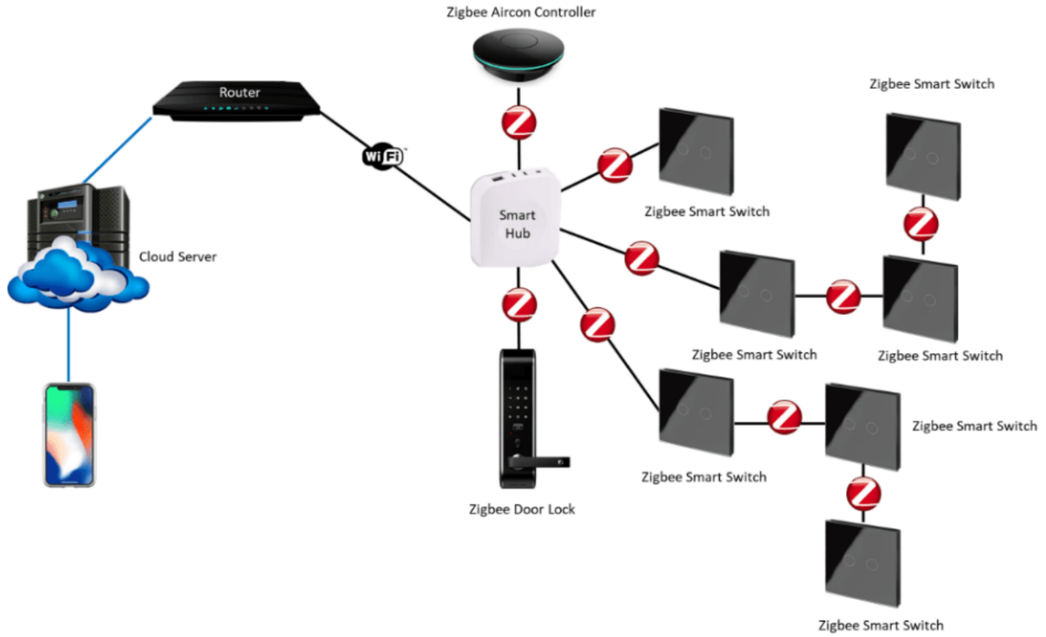
Vehicle gateway
4G LTE cellular with built-in WiFi hotspot for mobile devices.
~150-300€

Network layer - Gateways example 2

- Home automation protocols such as ZigBee typically use a gateway box plugged into wall power.

Connected to the home WiFi router with WiFi/Ethernet

~10-30€



Network layer - Gateways example 3

- Industrial gateways are usually more powerful, allowing **real time data processing and decision making**.
  - Low latency
  - Fail-safe in case of a lost connection between edge and cloud
  - Do not overload network and cloud servers

IoT architecture: application and analytics layer

- The **application and analytics layer** provides the requested services to IoT users.
- It interprets data using software applications. Applications may monitor, control, and provide reports based on the analysis of the data.
- **Data driven decision making.**

Applications and Analytics Layer - Applications

- Once data coming from different sensors is transmitted to a central station through the network via a gateway, applications running on top of the central station analyse the data.
- IoT applications are very diverse depending on the considered IoT system
  - Think of the IoT examples in the previous lecture: an application for baby monitoring has very different features and goals than an application for smart cities.
- From an architectural standpoint, we can classify IoT applications into two main types:

Control application

- Controls the behaviour of the smart objects, allowing to control complex aspects of an IoT system with a logic that cannot be programmed inside a single IoT device.
- The program is too complex to fit into the local system or requires data from other objects.
- E.g., pressure sensor connected to a pump. The control application increases the pump speed when the sensor senses a drop in pressure.

Collects data from multiple smart objects, processes it and displays the resulting information.

- The display can be about any aspect of the IoT system, e.g., historical reports, statistics, trend, individual system states.
- The application processes the data to convey a view of the network.

Applications and Analytics Layer - Analytics

- Analytics is a general term that described processing information to make sense of collected data.
- In IoT, a possible classification of the analytics function is as follows:

Data Analytics

Processing of the data collected by smart objects and combination of such data to provide an intelligent view related to the IoT system.
E.g., temperature, pressure, wind, humidity and light levels collected from thousands of sensors may be combined then processed to determine the likelihood of a storm and its possible path - involvement of complex algorithms models.
Data analytics can also monitor the IoT system itself, e.g., a robot in a factory can report data about its own movements to monitor its degradation.

Network Analytics

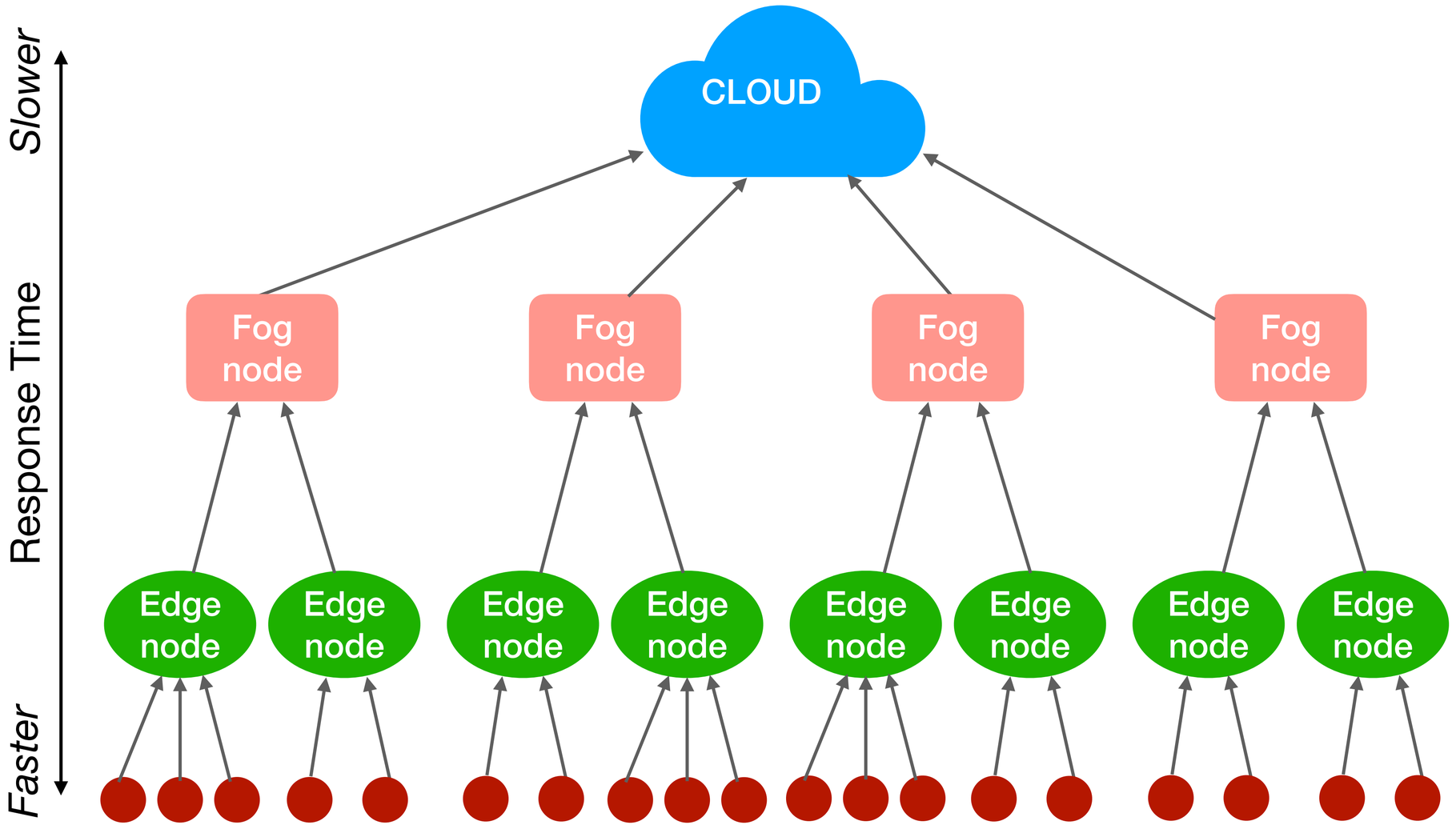
Network Analytics processes information about connectivity of the IoT system.
Loss or degradation in connectivity results in loss of data from sensors and can sometimes have dramatic effect (e.g., autonomous vehicles, factories).

2.3 Edge, Fog and Cloud Computing

Remote Computing

- Remote Computing (Fog/Cloud) offer more flexibility and scalability than hosting on a local server.
  - Reduced costs.
    - No need to buy expensive equipment, configuring and managing mainframe computers and infrastructures.
  - Increased speed.
    - Availability of enterprise applications, no need to wait weeks for the IT to respond to a request, purchase and configure supporting hardware and install software.
  - Scalability.
    - You pay for what you need instead of purchasing excess capacity that is unused most of the time.
  - Flexibility.
    - Services can be installed on remote servers and deliver better customer response time.

Cloud/Fog/Edge (1)



Cloud Layer

- Big data processing
- Business Logic
- Data storage

Fog Layer

- Local Area Network
- Data Analysis
- Control Response

Edge Layer

- Large volume real time data processing
- At source data visualisation
- Industrial PCs
- Embedded systems
- Gateways
- Micro data storage

Sensors

Edge Computing

- Edge computing is a distributed architecture in which client data is processed at the **periphery of the network**, i.e., as geographically close to the source of data as possible.
- In a IoT system, edge computing is performed by embedded systems, local computers/servers, IoT gateways to provide initial processing and filtering of all the data continuously sent from sensors.
- Data does not travel through the Internet, but is transmitted from the sensors using IoT protocols.

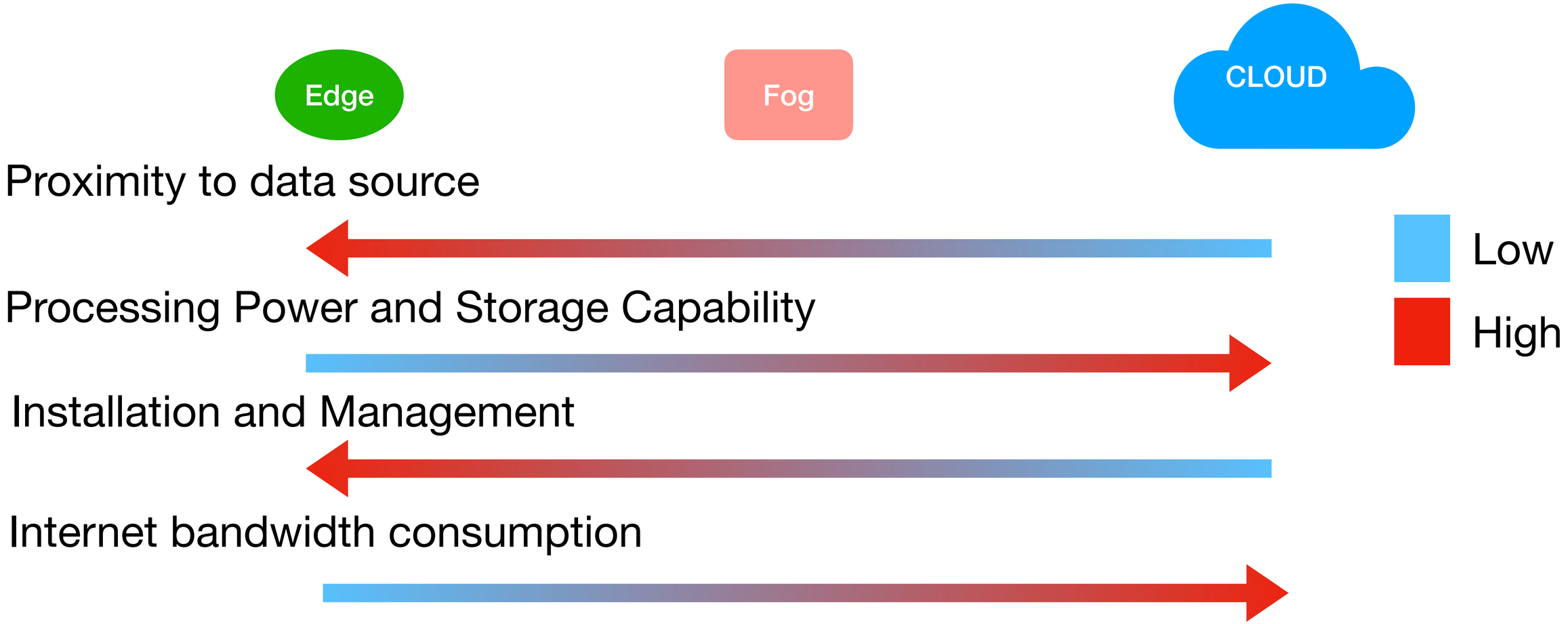
Cloud Computing

- Cloud computing is the delivery of on-demand computing resources—physical or virtual servers, data storage, networking capabilities, application development tools, software, AI-powered analytic platforms and more—over the internet with **pay-per-use pricing**.
- Resources are **dynamically assigned** and reassigned among multiple users and scale up and down in response to user's needs.
- Cloud data centres are connected to the Internet with very high speed, high capacity fibre optical cables.

Fog Computing

- Fog computing lies in between edge and cloud computing.
- The concept was introduced by Cisco and represents a decentralised infrastructure that places storage and processing components (e.g., servers) at the edge of the cloud.
- Depending on the IoT application, fog computing can provide more power resources and storage capacity than edge devices with lower costs than cloud resources.
- Fog servers usually located in LANs.
- Difference with Edge layer is blurry.

Cloud/Fog/Edge (2)



Why do we need the Fog if we have the Cloud?

- Offload the cloud, specially after the evolution of IoT.
- Optimised use of network and computing resources.
  - Reducing latency.
  - Reducing transport costs through the Internet.
- Supports cellular connections.

“X as a Service” XaaS

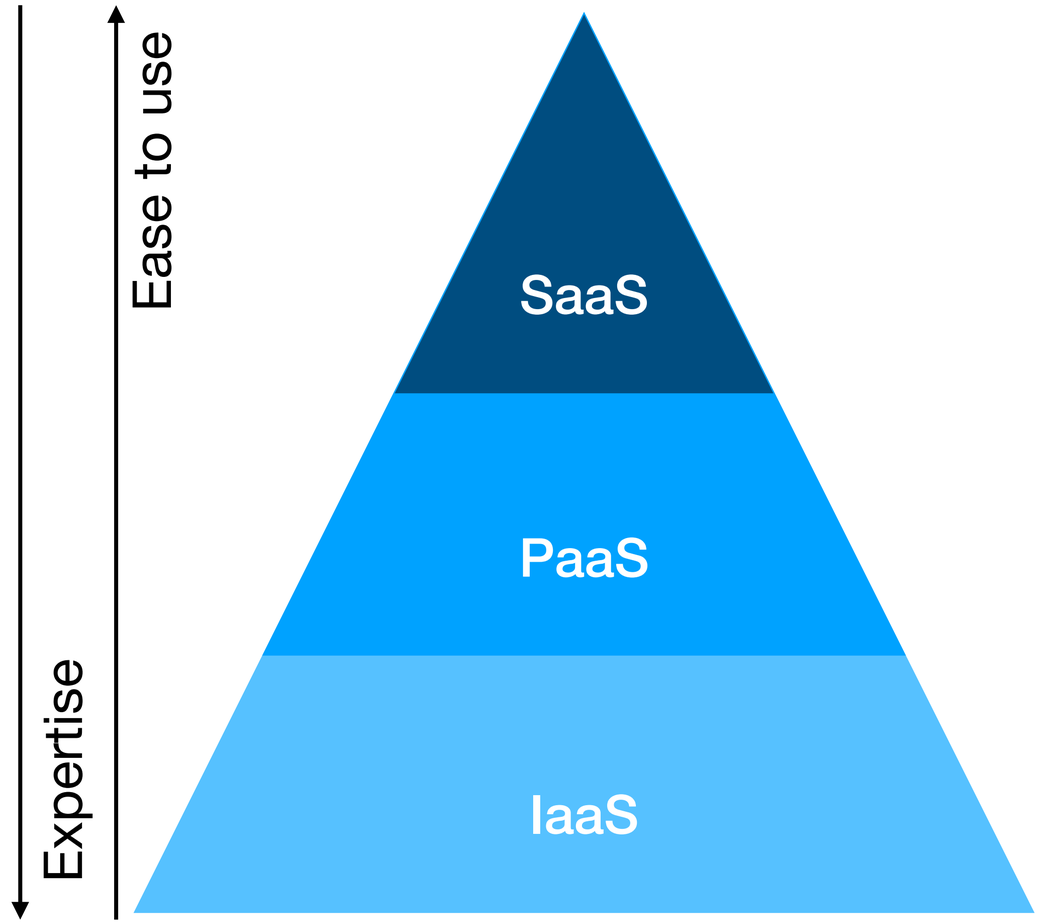
- Cloud computing allows businesses to deploy and run their applications on cloud servers.
- Users can access the services via requests to the Cloud hosting the application.
- XaaS “anything as a service”: delivery of solutions, application, products, tools and technologies as services, which exist on some Cloud platform.

IaaS/PaaS/SaaS

- The National Institute of Standards and Technology recognised three cloud service models in 2011:
  - **Infrastructure as a Service (IaaS):** cloud providers offer computing, networking and storage resources that have been virtualised by a vendor. Users can access and configure them as they want.
  - **Platform as a Service (PaaS):** cloud providers deliver a computing platform, typically including an operating system, programming language execution environment, database and web server.
  - **Software as a Service (SaaS):** users gain access to application software and databases. Cloud providers manage the infrastructure and platforms that run the applications.

Metaphor:

- **Infrastructure as a Service (IaaS) - Leasing a car.** You pay for the car, you care about its performance, you choose the model, the colour, and you can customise it. You drive the car.
- **Platform as a Service (PaaS) - Renting a car.** You do not care of most of the specs and the colour of the car. You drive it and pay for the gas and any tolls you go through.
- **Software as a Service (SaaS) - Taxi/Uber.** You do not care about any specs of the car, you do not choose anything. You do not drive or pay for the gas, all tolls are included in the price.



Top Cloud Providers (1)

- **Amazon Web Services (AWS).**
  - Offers services like database storage, computing power, networking. Can virtually host any applications, including networks like firewall, Load balancers.
  - Netflix, Airbnb, Uber and many other companies rely con AWS.

- **Microsoft Azure**
  - Offers services like AI, Machine Learning, Analytics, Blockchain, Compute, Containers, Databases, Developer Tools.
  - Microsoft has invested billions in OpenAI and provides the cloud infrastructure for running large AI models.

Top Cloud Providers (2)

2.4 Example of optimisation for task offloading on the fog

Fall Detection - minimise costs (1)

- Consider an IoT architecture with N wearable sensors $s_1, \ldots, s_N$, one for each patient.
- Every second, each sensor s_i produces some data that requires computational resource r_i (FLOPs). The data is sent every second to a dispatcher.
- The dispatcher collects the data from the sensors and has to decide how to deploy the tasks on fog nodes $f_1, \ldots, f_F$ to perform fall detection.
- Fog nodes have limited resources (FLOPS), $g_1, \ldots, g_F$, and using fog node f_j has a cost c_j. The dispatcher knows the resources and the costs.

Fall Detection - minimise costs (2)

- We want to minimise the costs for computation while performing all the tasks.
- We can introduce the decision variables:
  - $y_j = \begin{cases} 1 & \text{if } f_j \text{ is used} \\ 0 & \text{otherwise} \end{cases}$, $x_{i,j} = \begin{cases} 1 & \text{if task of sensor } s_i \text{ is offloaded on fog node } f_j \\ 0 & \text{otherwise} \end{cases}$
- And formulate the following problem:

$$\min \sum_{j=1}^{F} y_j \cdot c_j$$

Subject to: $\sum_{j=1}^{F} x_{i,j} = 1, \forall i = 1,\ldots,N$

$$\sum_{i=1}^{N} x_{i,j} r_j \leq g_j y_j, \forall j = 1,\ldots,F$$

$$y_j, x_{i,j} \in \{0,1\}$$

Minimize the total cost

All tasks are processed

The resource necessary to process all tasks arranged to fog f_j cannot exceed its capacity

Bin packing problem (NP complete)

Fall Detection - minimise costs (3)

- The problem is oversimplified. We are omitting
  - Latency
  - Dynamism
  - Queues of fog nodes
  - Etc
- How to solve this?
  - Approximation algorithms
  - Heuristics
  - Optimisation algorithms (exponential in the worst case)

Gurobi

- https://www.gurobi.com/features/academic-named-user-license/
- See attached code.

Bibliography

- Yaqoob, I., Ahmed, E., Hashem, I. A. T., Ahmed, A. I. A., Gani, A., Imran, M., & Guizani, M. (2017). Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges. IEEE Wireless Communications, 24(3), 10–16.

- David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry. “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”. Cisco press

- Federico Montori, Luca Bedogni, Marco Di Felice, Luciano Bononi “Machine-to-machine wireless communication technologies for the Internet of Things: Taxonomy, comparison and open issues”, Pervasive and Mobile Computing, Elsevier 2018

- Introduction to Cloud Computing - IBM course