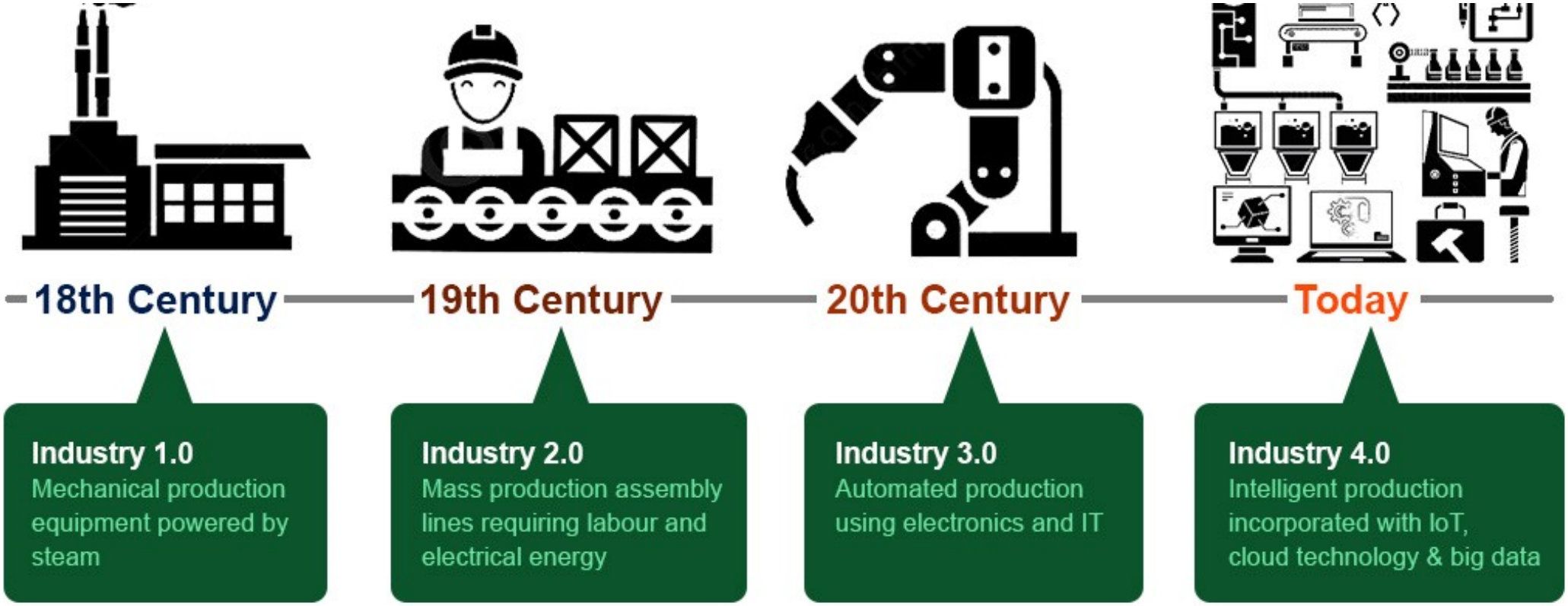
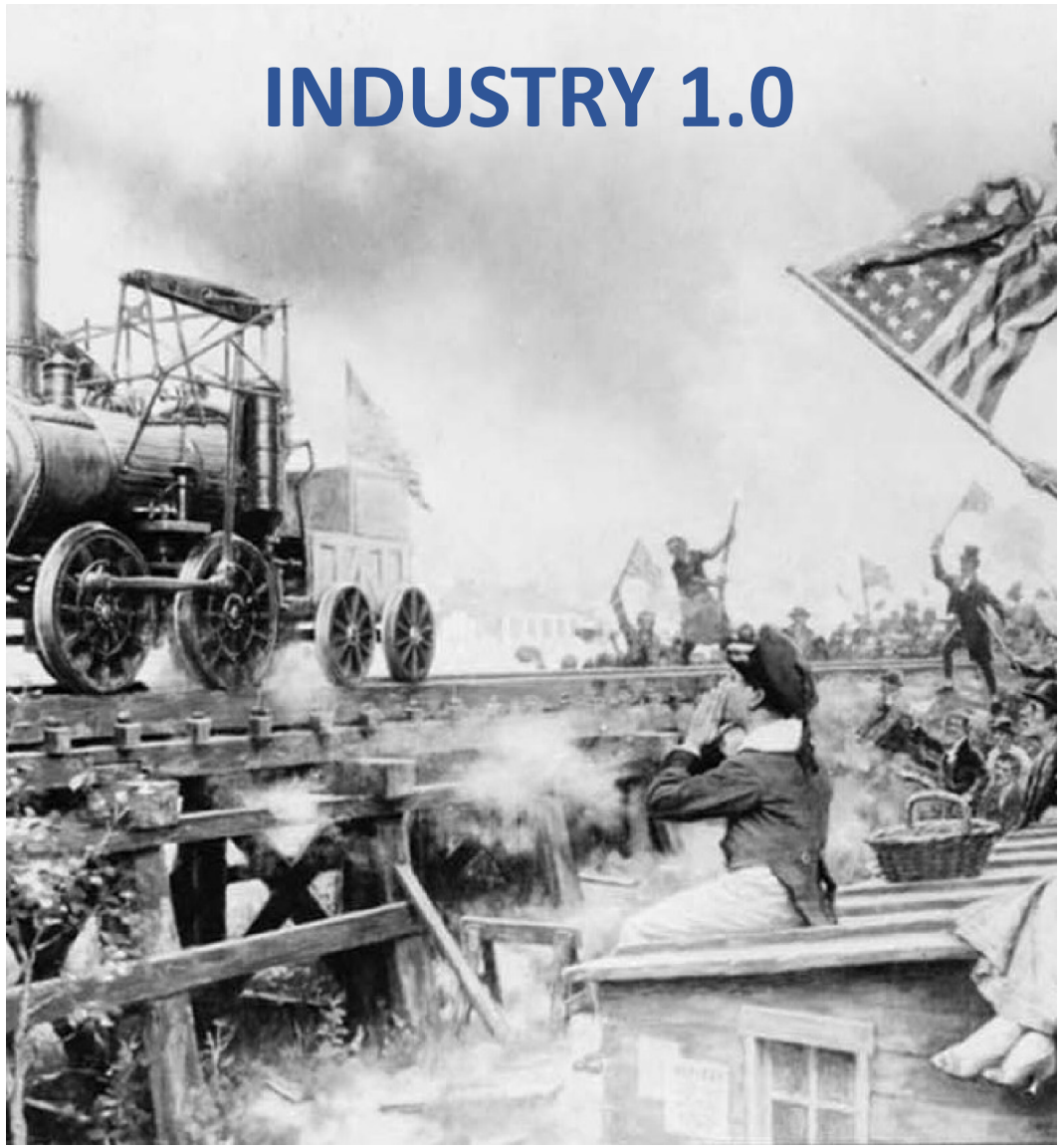
Memahami Konsep & Arsitektur IoT

Usman Nurhasan



Revolusi Industri


INDUSTRY 1.0


INDUSTRY 2.0

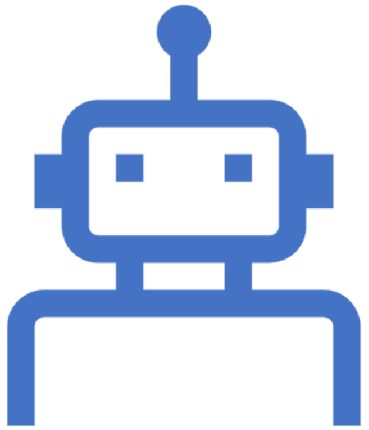

INDUSTRY 3.0
ABB

Revolusi Industri 4.0

Industry 4.0

- Industry 4.0 is the subset of the fourth industrial revolution that concerns industry. ***The fourth industrial revolution encompasses areas which are not normally classified as an industry***, such as smart cities, for instance.
- Although the terms "industry 4.0" and "fourth industrial revolution" are often used interchangeably, ***"industry 4.0" factories have machines which are augmented with wireless connectivity and sensors, connected to a system that can visualize the entire production line*** and ***make decisions on its own***.
- In essence, **industry 4.0 is the trend towards automation and data exchange in manufacturing technologies and processes which include cyber-physical systems (CPS), the internet of things (IoT), industrial internet of things (IIOT), cloud computing, cognitive computing and artificial intelligence**.

Cognitive Computing



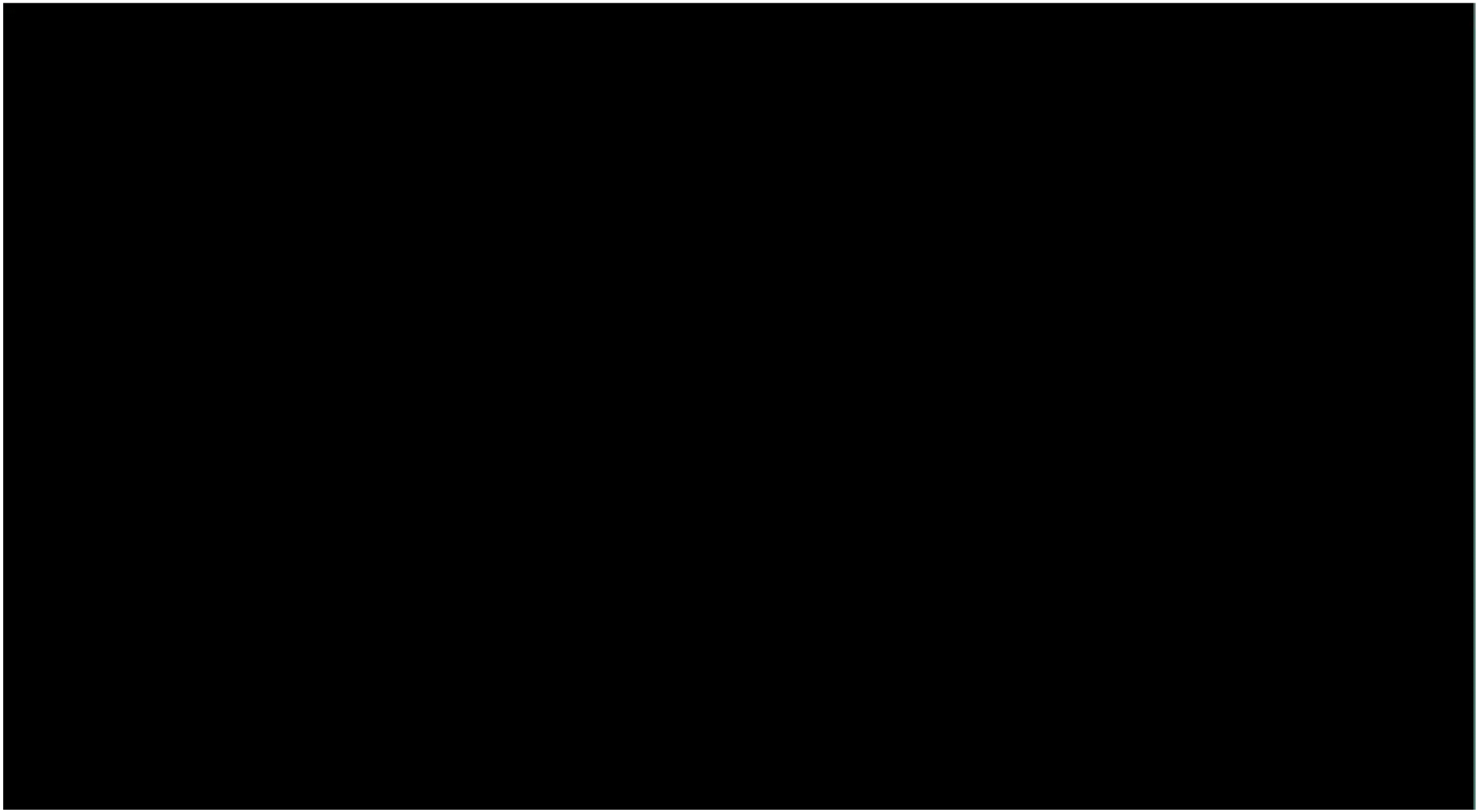
- Cognitive computing (CC) describes technology platforms are based on the scientific disciplines of artificial intelligence and signal processing.
- These platforms encompass **machine learning**, **reasoning**, **natural language processing**, **speech recognition** and **vision** (object recognition), **human–computer interaction**, **dialog and narrative generation**, among other technologies.

Industri 4.0 Technology

o	Technology	Description
	Advanced Robotic	• Autonomous, cooperating industial robots • Numerous integrated sensors and standarized interfaces
	Additive Manufacturing	• 3D printing, particularly for spare parts and prototypes • Decentralized 3D facilities to reduce transport distances and inventory
	Augmented Reality	• Augmented reality for maintenance, logistics and all kinds of SOP • Display of supporting information, eg. Through glasses
	Simulation	• Simulation of value networks • Optimization based on real time data from intellegent system
	Horizonal/Vertical Integration	• Cross company data integration based on data transfer standards • Precondition for a fully automated value chain (from supplier to customer, from management to shop floor)
	Industrial Internet	• Network of machines and products • Multidirectional communication between networked objects • Internet of Things
	Cloud	• Management of huge data volumes in open systems • Real time communication for production systems
	Cybersecurity	• Operation in networks and open systems • High level of networking between intelegent machine, products, and systems
	Big Data and Analytics	• Full evaluation of avalaible data (eg. From ERP, SCM, MES, CRM and machine data) • Real time decision making support and optimization

Smart Car

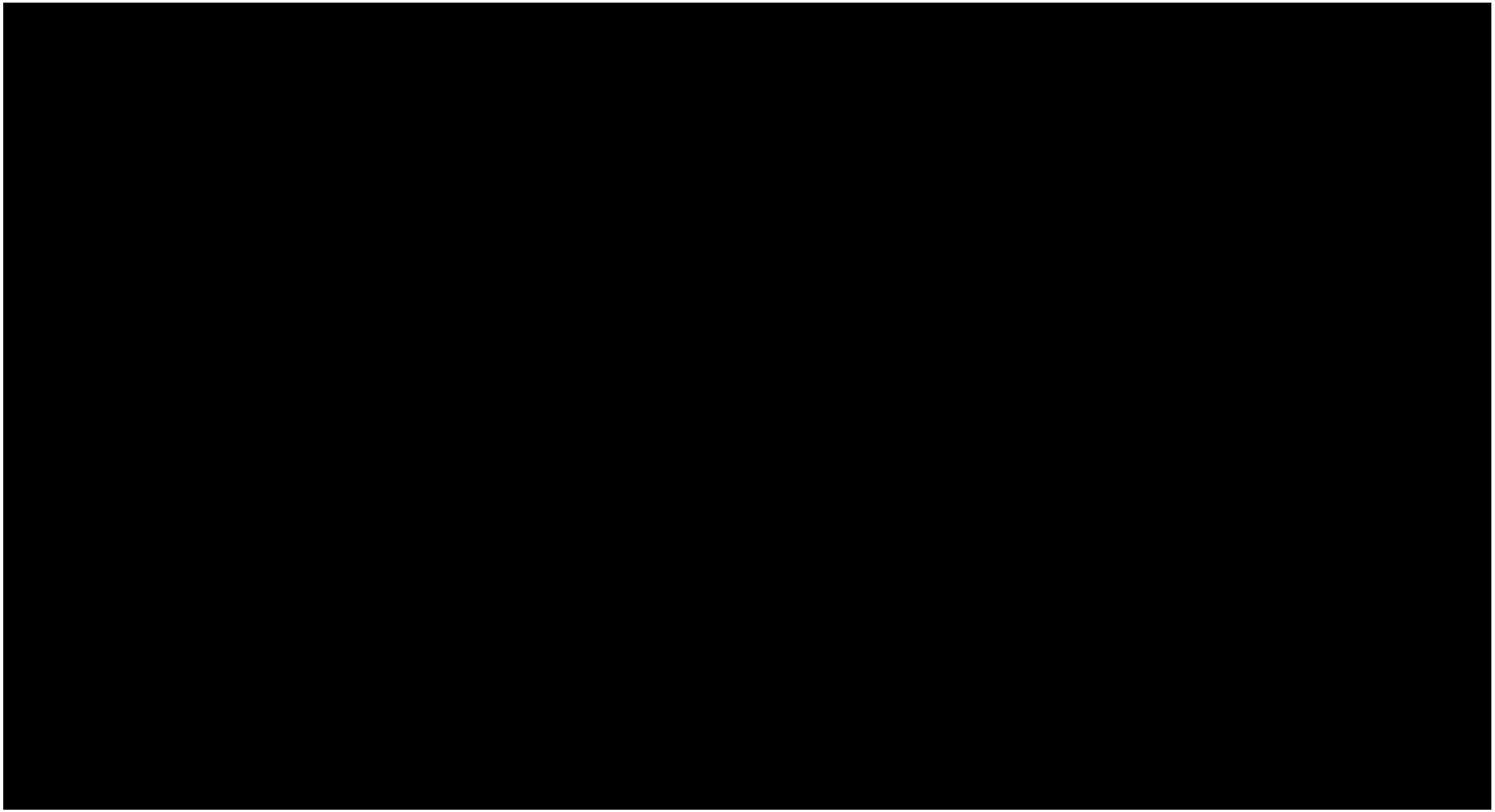
Smart Robot



Smart Home


CNN Money
FaceTime
Calendar
Photos
Camera
Clock
Maps
Contacts
Home
Reminders
Videos
Notes
News
iTunes Store
App Store
Settings

Smart Farming



IoT (Internet of Things)

Definition

Internet of things, atau IoT, adalah sistem perangkat komputasi yang saling terkait, mesin mekanik dan digital, benda, hewan atau orang yang dilengkapi dengan pengidentifikasi unik (UID – Unik Identifier) dan kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer.

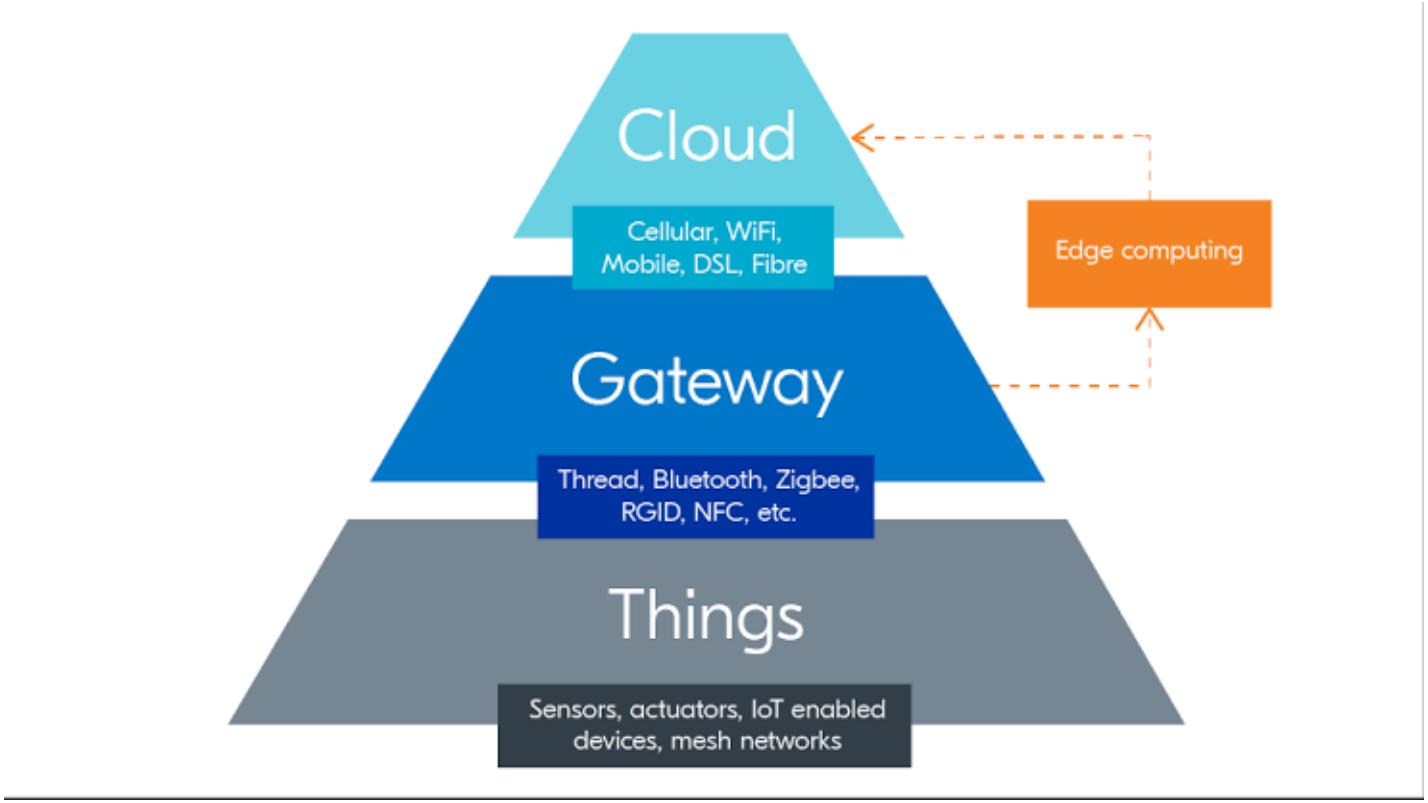
Things dalam internet dapat berupa manusia dengan implan monitor jantung did alamnya, hewan ternak dengan transponder biochip, sebuah mobil auto yang di dalamnya terdapat sensor built-in untuk memberi tahu pengemudi ketika tekanan ban rendah atau segala dalam kondisi normal atau segala sesuatu buatan manusia yang dapat ditugaskan ke alamat IP dan dapat mentransfer data melalui network

IoT (Internet of Things)

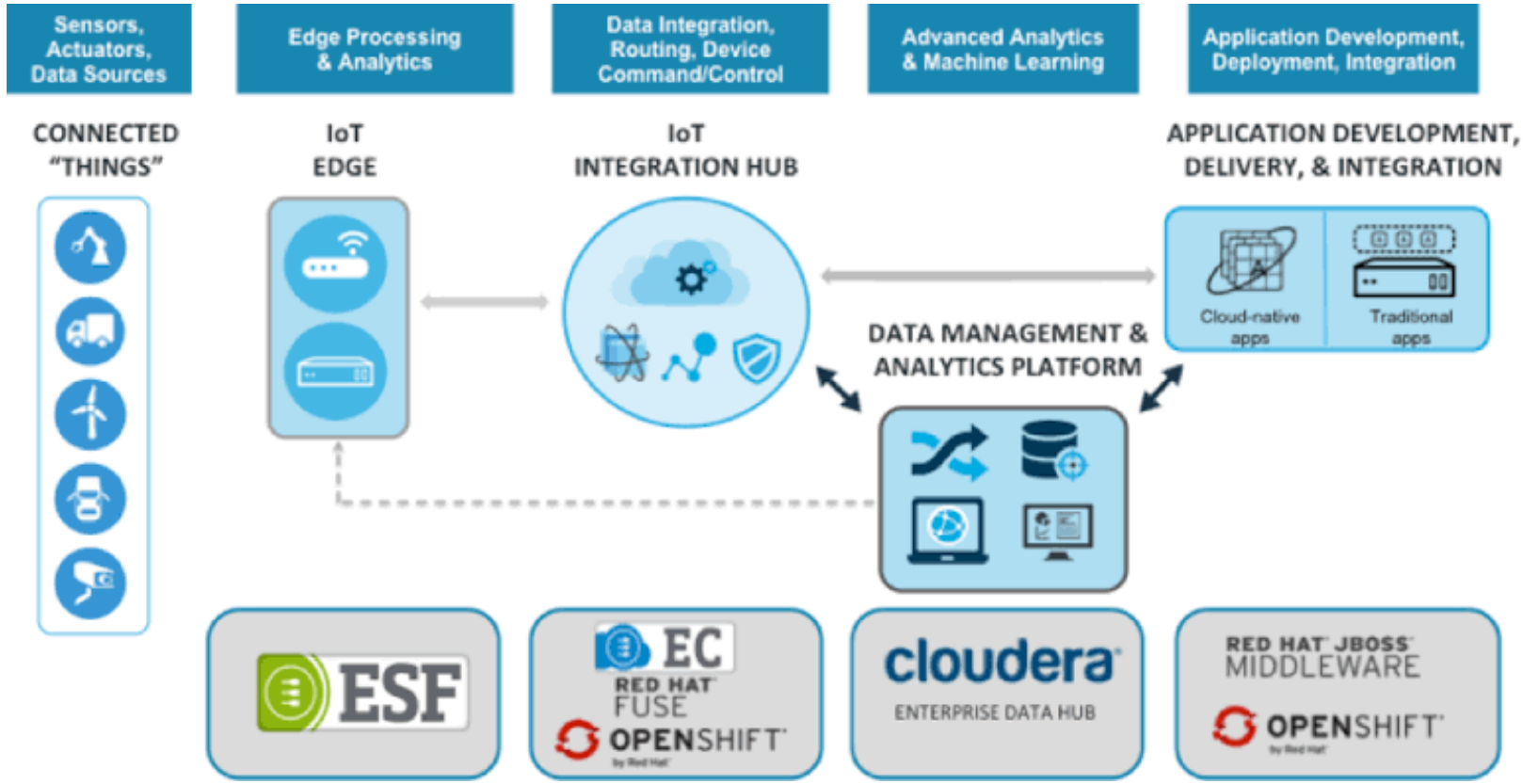
How IoT works

- Ekosistem IoT terdiri dari perangkat pintar berkemampuan web yang menggunakan prosesor tertanam, sensor, dan perangkat keras komunikasi untuk mengumpulkan, mengirim, dan bertindak berdasarkan data yang mereka peroleh dari lingkungan mereka.
- Perangkat IoT berbagi data sensor yang mereka kumpulkan dengan menghubungkan ke gateway IoT atau edge device lainnya dimana data dikirim ke cloud untuk dianalisis atau dianalisis secara lokal. Terkadang, perangkat ini berkomunikasi dengan perangkat terkait lainnya dan bertindak berdasarkan informasi yang mereka dapatkan dari satu sama lain. Perangkat-perangkat tersebut melakukan sebagian besar pekerjaan tanpa campur tangan manusia, meskipun orang dapat berinteraksi dengan perangkat - misalnya, untuk mengaturnya, memberi mereka instruksi atau mengakses data.
- Konektivitas, jaringan dan protokol komunikasi yang digunakan dengan perangkat yang mendukung web ini sangat tergantung pada aplikasi IoT spesifik yang digunakan.

IoT (Internet of Things) - How IoT works Stack Architecture



IoT (Internet of Things) - How IoT works Stack Architecture



IoT (Internet of Things)

Why IoT is important

- Internet of Things membantu manusia hidup dan bekerja lebih cerdas serta mendapatkan kontrol penuh atas kehidupan mereka. Selain menawarkan perangkat pintar untuk mengotomatisasi rumah, IoT sangat penting untuk bisnis. IoT memberikan bisnis secara real-time untuk melihat ke dalam bagaimana sistem perusahaan mereka benar-benar bekerja, memberikan wawasan tentang segala sesuatu mulai dari kinerja alat berat hingga rantai pasokan (supply chains) dan operasi logistik.

- IoT memungkinkan perusahaan untuk mengotomatisasi proses dan mengurangi biaya tenaga kerja. Ini juga mengurangi pemborosan dan meningkatkan pengiriman layanan, membuatnya lebih murah untuk memproduksi dan mengirimkan barang serta menawarkan transparansi dalam transaksi dengan pelanggan.

- IoT menyentuh setiap sisi industri, termasuk kesehatan, keuangan, ritel, dan manufaktur. Kota pintar membantu warga mengurangi konsumsi limbah dan energi dan sensor yang terhubung bahkan digunakan dalam pertanian untuk membantu memantau hasil panen dan ternak serta memprediksi pola pertumbuhan.

Benefits of IoT

Internet of things menawarkan sejumlah manfaat bagi organisasi, memungkinkan mereka untuk:

- Memantau keseluruhan proses bisnis mereka
- Meningkatkan pengalaman pelanggan
- Menghemat waktu dan uang
- Meningkatkan produktivitas karyawan
- Mengintegrasikan dan mengadaptasi model bisnis
- Membuat keputusan bisnis yang lebih baik
- Menghasilkan lebih banyak pendapatan.
- IoT mendorong perusahaan untuk memikirkan kembali cara mereka mendekati bisnis, industri, dan pasar mereka dan memberi mereka alat untuk meningkatkan strategi bisnis mereka.

Keuntungan dan Kerugian IoT

Keuntungan IoT meliputi:

- Kemampuan untuk mengakses informasi dari mana saja kapan saja di perangkat apa pun
- Peningkatan komunikasi antara perangkat elektronik yang terhubung
- Mentransfer paket data melalui jaringan yang terhubung akan menghemat waktu dan uang
- Mengotomatiskan tugas-tugas akan membantu meningkatkan kualitas layanan bisnis dan mengurangi kebutuhan akan intervensi manusia.

Kerugian IoT meliputi:

- Dengan meningkatnya jumlah perangkat yang terhubung dan lebih banyak informasi yang dibagikan di antara perangkat, potensi pencurian informasi rahasia oleh peretas juga meningkat
- Perusahaan enterprise pada akhirnya mungkin harus berurusan dengan sejumlah besar bahkan jutaan perangkat IoT, mengumpulkan dan mengelola data dari semua perangkat tersebut akan menjadi tantangan.
- Jika ada bug dalam sistem, kemungkinan setiap perangkat yang terhubung akan rusak
- Karena tidak ada standar kompatibilitas internasional untuk IoT, sulit bagi perangkat dari berbagai produsen untuk berkomunikasi satu sama lain.

IoT standards and frameworks

- **6LoWPAN (IPv6 over Low -Power Wireless Personal Area Networks)**, an open standard defined by the Internet Engineering Task Force (IETF). The 6LoWPAN standard enables any low-power radio to communicate to the internet, including 804.15.4, Bluetooth Low Energy and Z-Wave (for home automation).
- **ZigBee**, a low-power, low data-rate wireless network used mainly in industrial settings. ZigBee is based on based the IEEE 802.15.4 standard. The ZigBee Alliance created Dotdot, the universal language for IoT that enables smart objects to work securely on any network and understand each other.
- **LiteOS**, a Unix-like operating system for wireless sensor networks. LiteOS supports smartphones, wearables, intelligent manufacturing applications, smart homes and Internet of Vehicles (IoV). The operating system also serves as a smart device development platform.

IoT standards and frameworks

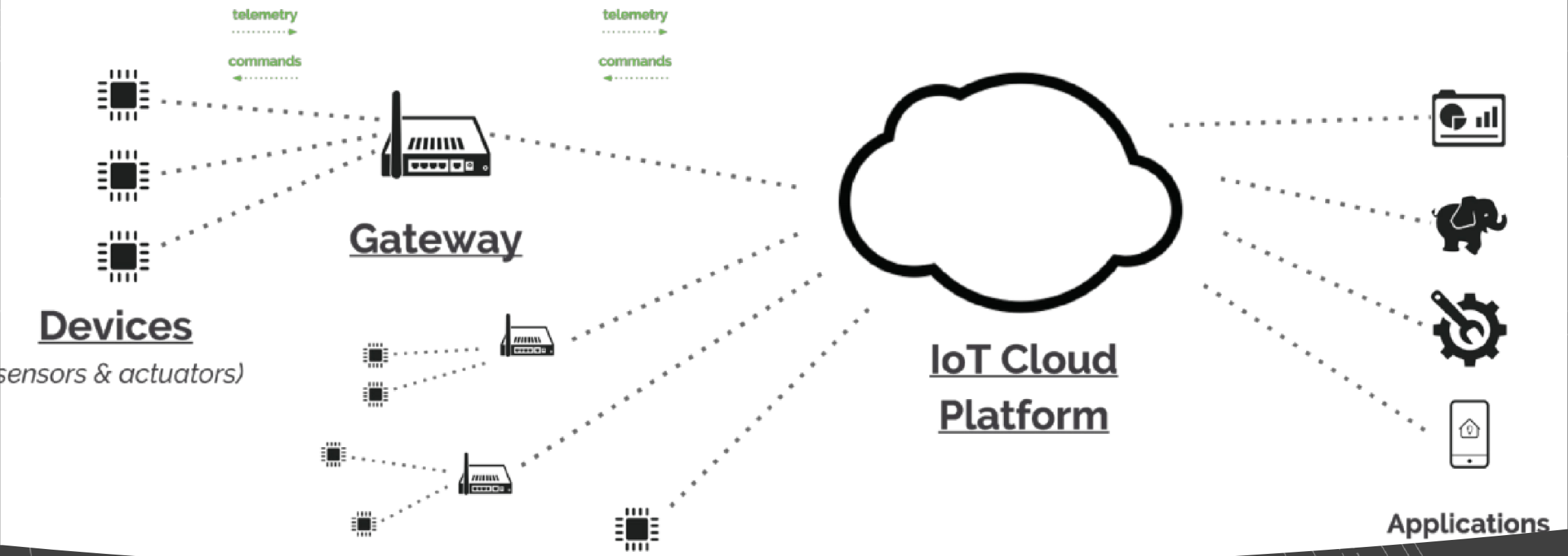
- **OneM2M**, a machine-to-machine service layer that can be embedded in software and hardware to connect devices. The global standardization body, OneM2M, was created to develop reusable standards to enable IoT applications across different verticals to communicate.
- **DDS (Data Distribution Service)** was developed by the Object Management Group (OMG) and is an IoT standard for real-time, scalable and high-performance machine-to-machine communication.
- **AMQP (Advanced Message Queuing Protocol)**, an open source published standard for asynchronous messaging by wire. AMQP enables encrypted and interoperable messaging between organizations and applications. The protocol is used in client/server messaging and in IoT device management.

IoT standards and frameworks

- **AMQP (Advanced Message Queuing Protocol),** an open source published standard for asynchronous messaging by wire. AMQP enables encrypted and interoperable messaging between organizations and applications. The protocol is used in client/server messaging and in IoT device management.
- **CoAP (Constrained Application Protocol),** a protocol designed by the IETF that specifies how low-power compute-constrained devices can operate in the internet of things.
- **LoRaWAN (Long Range Wide Area Network),** a protocol for wide area networks, it's designed to support huge networks, such as smart cities, with millions of low-power devices.
- **MQTT (MQ Telemetry Transport)** is an open OASIS and ISO standard lightweight, publish-subscribe network protocol that transports messages between devices. The protocol usually runs over TCP/IP; however, any network protocol that provides ordered, lossless, bi-directional connections can support MQTT. It is designed for connections with remote locations where a "small code footprint" is required or the network bandwidth is limited.

edureka!

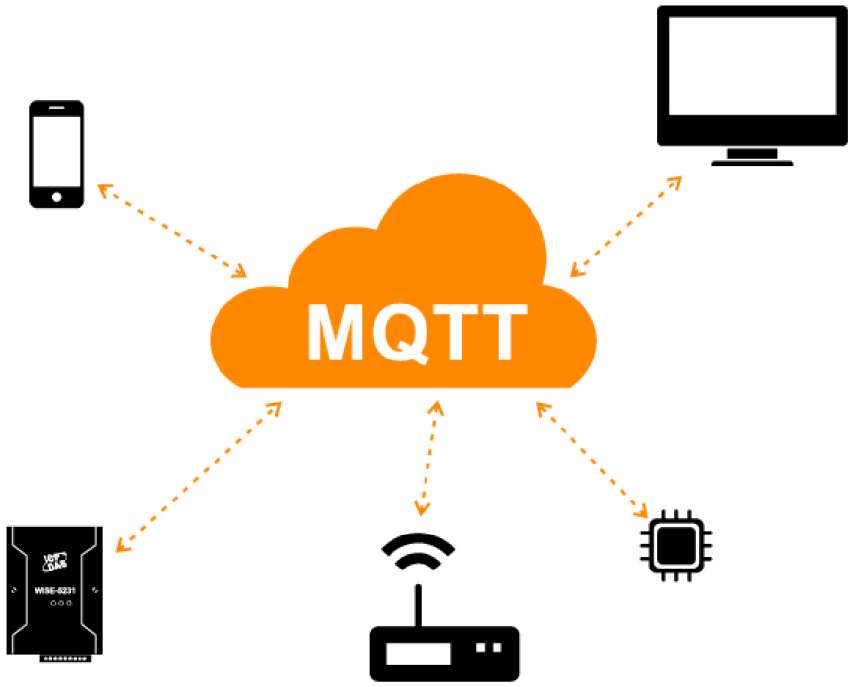

edureka!
IOT
AI



Stack IoT Architecture

A typical IoT solution is characterized by many devices (i.e. things) that may use some form of gateway to communicate through a network to an enterprise back-end server that is running an IoT platform that helps integrate the IoT information into the existing enterprise. The roles of the devices, gateways, and cloud platform are well defined, and each of them provides specific features and functionality required by any robust IoT solution.

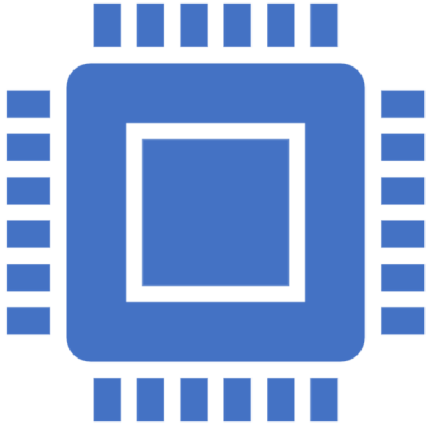
MQTT Protocol Transport



MQTT is one of the most commonly used protocols in IoT projects. It stands for ***Message Queuing Telemetry Transport***.

In addition, it is designed as a lightweight messaging protocol that uses publish/subscribe operations to exchange data between clients and the server. Furthermore, its small size, low power usage, minimized data packets and ease of implementation make the protocol ideal of the “machine-to-machine” or “Internet of Things” world.

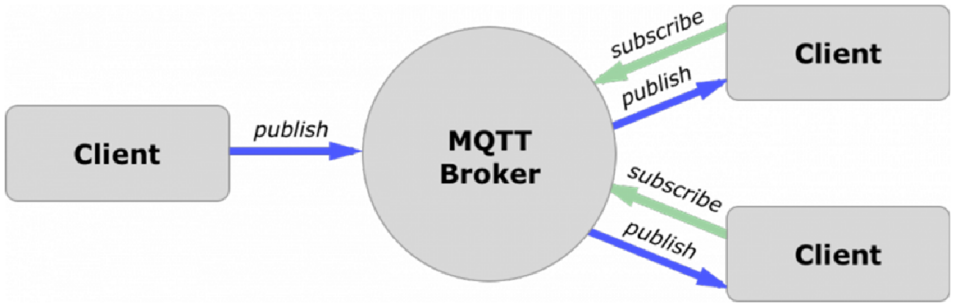
Why MQTT?



MQTT has unique features you can hardly find in other protocols, like:

- It’s a lightweight protocol. So, it’s easy to implement in software and fast in data transmission.
- It’s based on a messaging technique.
- Minimized data packets. Hence, low network usage.
- Low power usage. As a result, it saves the connected device’s battery.
- It’s real time! That’s is specifically what makes it perfect for IoT applications.

How MQTT works

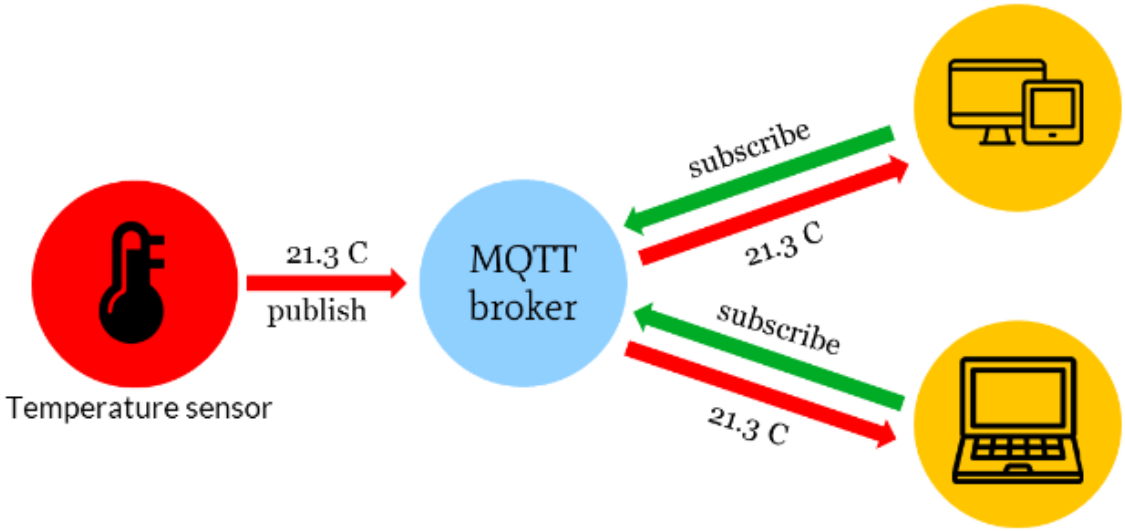


MQTT is based on clients and a server. Likewise, the server is the guy who is responsible for handling the client's requests of receiving or sending data between each other.

MQTT server is called a broker and the clients are simply the connected devices.

- When a device (a client) wants to send data to the broker, we call this operation a "**publish**".
- When a device (a client) wants to receive data from the broker, we call this operation a "**subscribe**".

Example



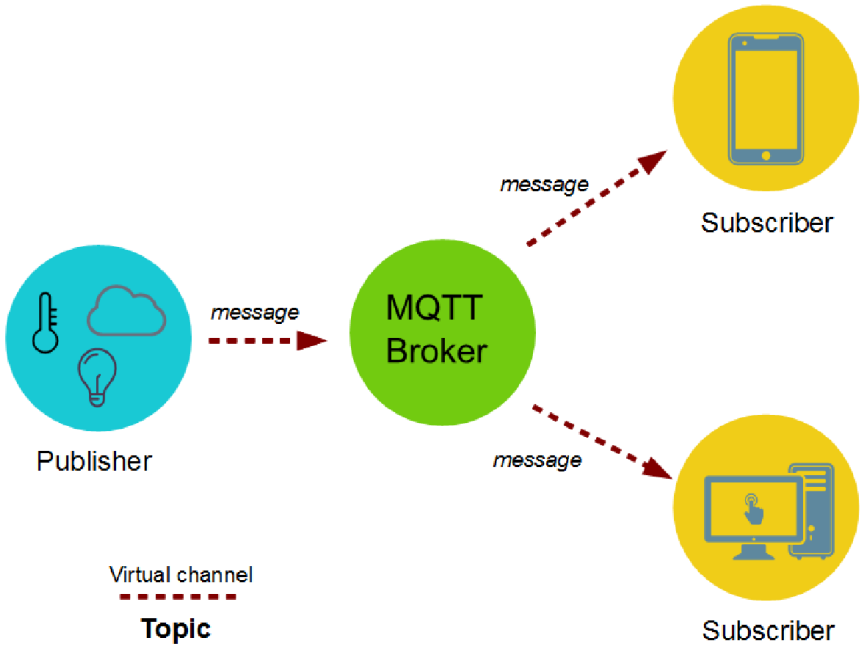
Schematic data flow from sensor (machine) to devise (machine)

Let’s say there is a device that has a temperature sensor. Certainly, it wants to send his readings to the broker. On the other side, a phone/desktop application wants to receive this temperature value. Therefore, 2 things will happen:

- The device defines the topic it wants to publish on, ex: “temp”. Then, it publishes the message “temperature value”.
- The phone/desktop application subscribes to the topic “temp”. Then, it receives the message that the device has published, which is the temperature value.

Again, the broker role here is to take the message “temperature value” and deliver it to phone/desktop application.

MQTT Components



That takes us to the MQTT components, which are 5 as follows:

- Broker, which is the server that handles the data transmission between the clients.
- A topic, which is the place a device want to put or retrieve a message to/from.
- The message, which is the data that a device receives “when subscribing” from a topic or send “when publishing” to a topic.
- Publish, is the process a device does to send its message to the broker.
- Subscribe, where a device does to retrieve a message from the broker.

Why not HTTP

HTTP is slower, more overhead and power consuming protocol than MQTT. So, let’s get into each one separately:

- Slower: because it uses bigger data packets to communicate with the server.
- Overhead: HTTP request opens and closes the connection at each request, while MQTT stays online to make the channel always open between the broker “server” and clients.
- Power consuming: since it takes a longer time and more data packets, therefore it uses much power

Mosquitto broker

- Mosquitto is an open source message broker that implements the MQTT protocol. It's lightweight and suitable for use on all devices from a low power single board like Arduino, ESP8266 to full computers and servers.
- But rather than using the Mosquitto on a local PC, you will need to use a cloud-based server that implements the Mosquitto broker. That's necessary to make your IoT projects controllable over the internet.
- Cloud-based Mosquitto brokers are many, like:
  - ThingMQ
  - ThingStudio
  - MQTT.io
  - Heroku
  - CloudMQTT

IoT Cloud Server

- https://thingspeak.com
- https://thingsboard.io
- https://thinger.io
- https://www.blynk.cc
- etc