C1: Introduction to IoT

IoT Definitions, Approaches and Models

IoT Concepts

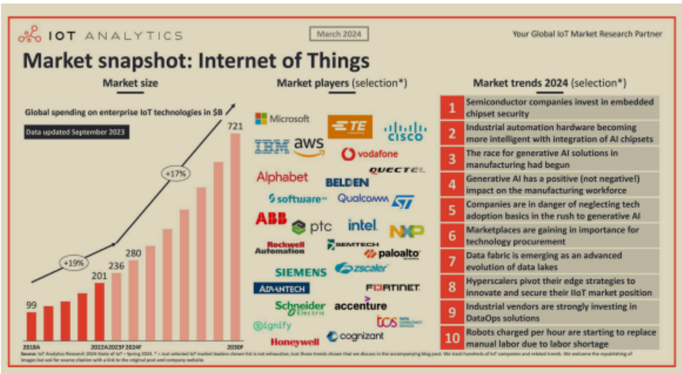
IoT things: These objects acting as sensors / actuators执行器 are able to interact with each other in order to reach a common goal
Internet of Things (IoT) paradigm is based on **intelligent** and **self configuring nodes (things)** interconnected in a dynamic and global network infrastructure
IoT refers to a **'world-wide network of interconnected objects'** uniquely addressable based on standard communication protocols whose point of convergence汇聚点 is the Internet

3 elements of IoT

- **Identity**
- **Intelligence**
- **Communication**

IOT Markets

Number of connected devices
- Overtook the number of people (Year 2011)
- Estimated at 46 billion (Year 2021)
- Expected to reach 125 billions (Year 2030))



Source: Knud Lasse Lueth, 14 March 2024, https://iot-analytics.com/state-of-iot-10-emerging-iot-trends-driving-market-growth/

Volume of Connected data

- International Data Corporation (IDC) estimates that the amount of data in the world will more than quadruple in the next five years to reach **175 zettabytes** in 2025. 国际数据公司（IDC）估计，未来五年全球数据量将翻两番以上，到 2025 年将达到175 ZB。

Factors that derives hype炒作 of IoT

- Cheaper Technology – Moore's Law has lowered the cost of sensors, processors, and networking, with WiFi making integration easier.
- IPv6 Expansion – Allows an unlimited number of connected devices.
- Industry Support – Major companies (Cisco, IBM, GE, Amazon) are enhancing networks with Fog and Swarm layers to simplify and reduce costs.

<u>IOT Common Communication Model (4)</u>

1: Device-to-Device Communication

- Two or more devices <u>connect and communicate directly</u> (no need intermediary application server)
- They communicate over networks (IP networks / Internet), Often use protocols like Bluetooth, Z-Wave, ZigBee
- Applications: home automation - residential IoT devices like light bulbs, door locks (use small data packets of information to communicate between devices with relatively low data rate requirement

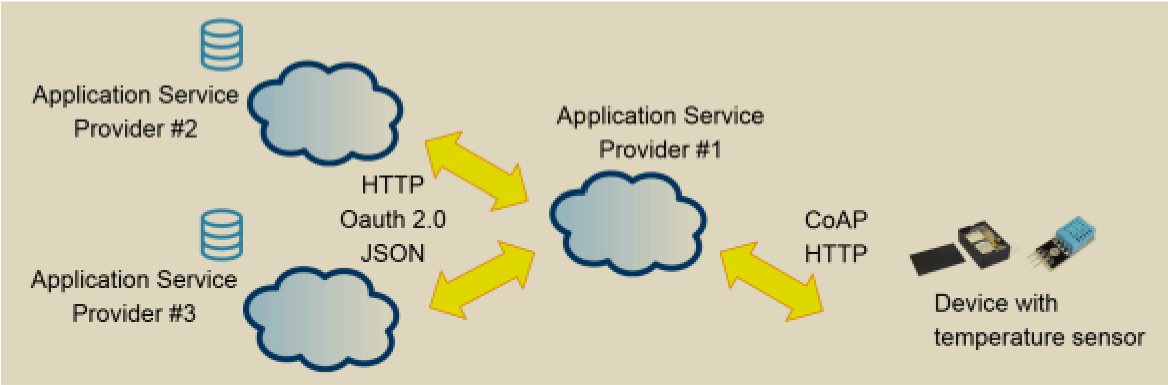
2: Device-to-Cloud Communication

- IoT devices <u>connect directly to an Internet cloud service</u> (application service provider to exchange data & control message traffic
- Takes advantage of existing communication mechanisms (wired Ethernet / Wi-Fi connections) to establish a connection between device & IP network, ultimately connects to the cloud service
- Applications: Samsung SmartTV - television uses Internet connection to transmit user viewing information to Samsung for analysis / enable interactive voice recognition feature

3: Device-to-Gateway Communication

- Aka Device-to-application-layer gateway (ALG) model, IoT devices <u>connect through ALG service as a conduit管道 to reach a cloud service</u>
- typically , there is application software operating on local gateway device(intermediary between devices- cloud service, provides security/func like data and protocol translation转换
- Applications: smartphones act as a gateway, relaying data between the device and the cloud - Personal fitness trackers lack direct cloud connectivity and depend on a smartphone app to sync and transmit data.

4: Back-End Data-Sharing Model



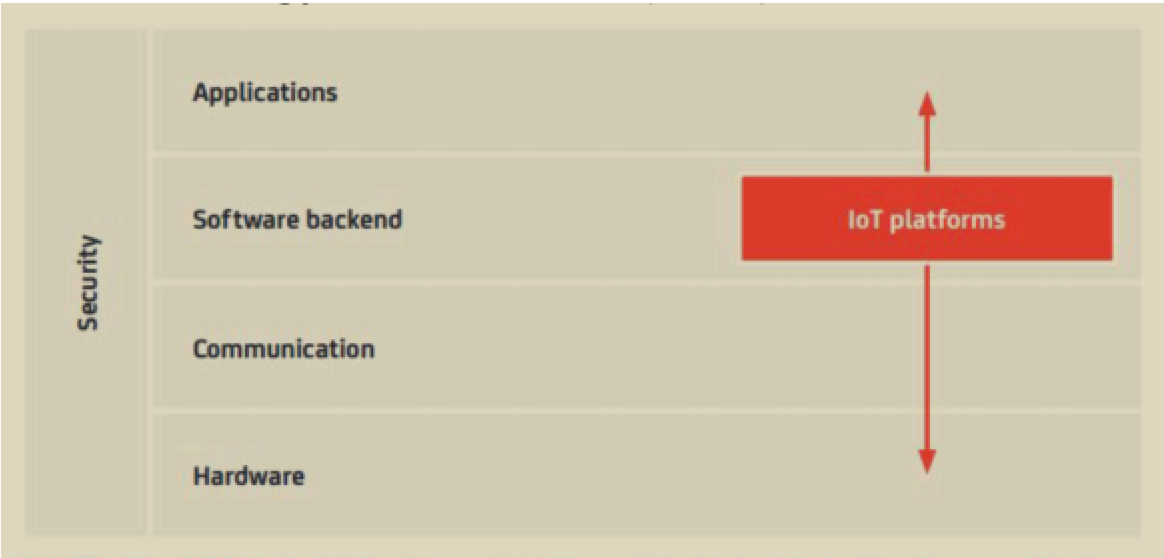
- Communication architecture that <u>enables users to export & analyze smart object data from a cloud service in combination with data from other sources</u>
- Allows data collected from single IoT device data streams to ge aggregated and analyzed
- enables interoperability互操作性 through federated cloud services or APIs.
- Applications: An office complex can analyze energy consumption by aggregating IoT sensor data. A well-designed back-end architecture allows seamless data access, portability, and integration, breaking down data silos. 办公大楼可以通过汇总物联网传感器数据来分析能源消耗。精心设计的后端架构可实现无缝数据访问、可移植性和集成，打破数据孤岛。

C2: Common Structure of IOT System

IoT Development and IoT Stack: Application, Data Processing (Softtware Backend), Network (Communication), Sensing (Hardware)

IoT Development

Technology Infrastructure (4+1)



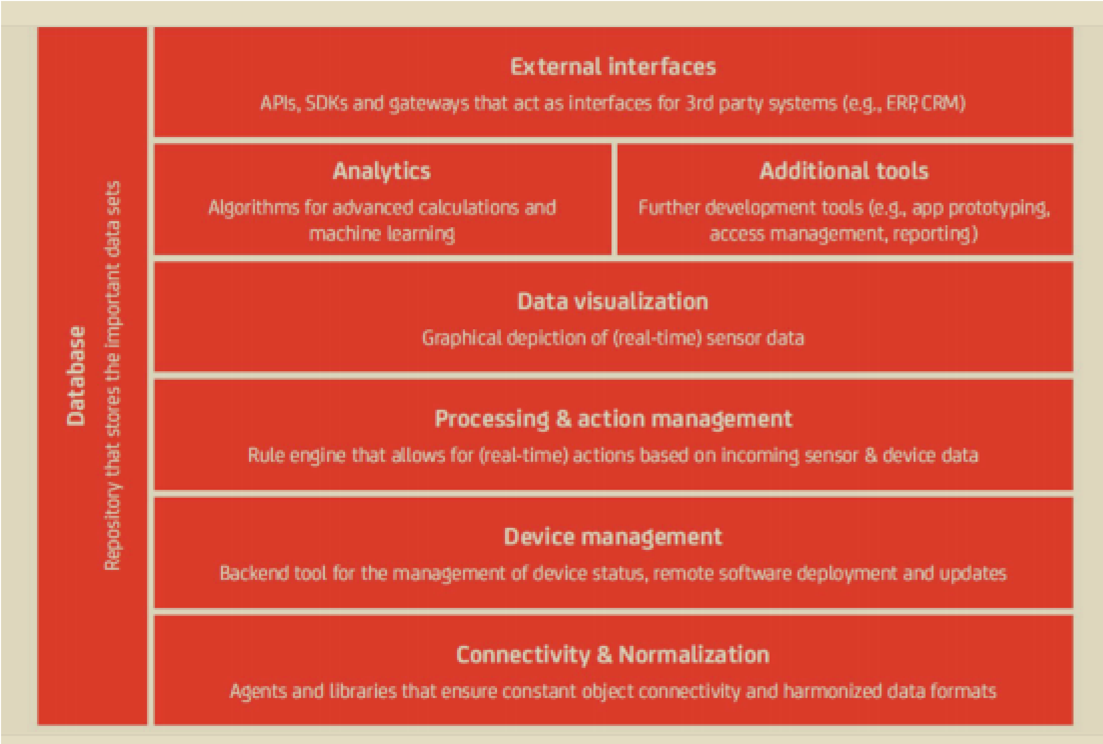
Applications: The software that interacts with users and other systems to provide functionality.
Software Backend: The data processing and storage systems that support the applications.
Communication: The network infrastructure that enables data transfer between devices and the cloud.
Hardware: The physical devices (sensors, actuators, etc.) that interact with the environment.
Security: A critical component to protect data and ensure the integrity of the system.
IoT technology architecture is currently far from being standardized or accurately defined and it is evolving very quickly.
There are hundreds of different hardware units, connection protocols, low-level software languages, and an increasing number of IoT platforms. It is a relatively young infrastructure.

Platforms Architecture

IoT Platform

- facilitates communication, data flow, device management, and application functionality.
- Links machines, devices, applications, and people to data and control centers.
- Employs advanced search engines and data storage systems.

- Most elements are *cloud-based* and run on *wireless connectivity.*
- Provides scalability, security, and data analysis capabilities.



a modern IoT platform architecture

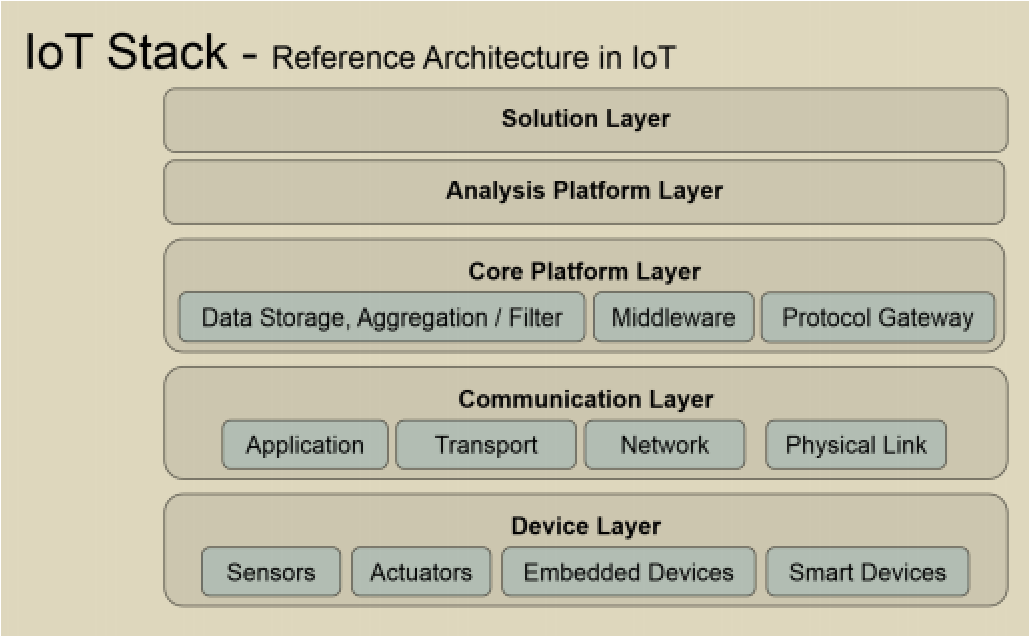
In its simplest form, an IoT platform is just about enabling connectivity between "things" or devices. The architecture may also consist of a software platform, an application development platform or an analytics platform. In a more sophisticated form, a true end-to-end IoT platform consists of eight important architectural components:

1. Connectivity & Normalization
    - Brings different protocols and data formats into a unified software interface.
    - Ensures accurate data streaming and seamless device interaction.
2. Device Management
    - Ensures connected devices function properly.
    - Handles software patches and updates for devices and edge gateways.
3. Database
    - Provides scalable storage for device data.
    - Supports hybrid cloud-based databases to manage data volume, variety, velocity, and veracity.
4. Processing & Action Management

- Enables rule-based event-action triggers.
- Executes smart actions based on specific sensor data.

5. Analytics
    - Performs complex analysis, including data clustering, machine learning, and predictive analytics.
    - Extracts valuable insights from IoT data streams.
6. Visualization
    - Displays data patterns and trends through dashboards.
    - Supports various chart types (line, stacked, pie) and 2D/3D models.
7. Additional Tools
    - Provides IoT developers with prototyping, testing, and deployment tools.
    - Supports app development for device visualization, management, and control.
8. External Interfaces
    - Integrates with third-party systems and IT ecosystems.
    - Uses APIs, SDKs, and gateways for seamless interaction.

IoT Stack



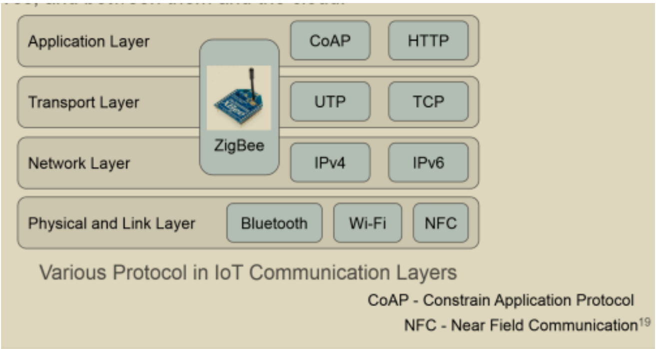
5-Device Layer

Devices are the “things” of the Internet of Things.

Sensors	• Detects events and reads changes in its environment. • Can be programmed to sense specific environmental parameters and convert them into meaningful data. Types of Sensors 1. Connected Sensor ◦ Captures information from a physical resource within a specific area. ◦ Transmits data over a network for further processing. ◦ Example: Smart water sensors measure water quality and pollution levels by detecting harmful ingredients in the water. 2. Single-Form Sensor ◦ Standalone sensor that emits signals based on the target object's movement. ◦ Provides alerts through regular beeps. ◦ Example: A medicine bottle with a sensor that beeps to remind patients to take their medication on time.
Actuators	• Interact with the physical world by executing actions based on received inputs. Examples of actuators: DC/AC Servo Motor, Step Motor Example Use Case: Connected door locks controlled remotely, where the actuator controls the movement of the lock motor. Actuator = Executes actions; Indicator = Show somethings
Embedded System (Controller)	Serves as a prototype for testing IoT use cases. Availability of controllers becomes an ideal prototyping choice, and leads to building a commercial connected product. Popular controller boards: • Arduino • Raspberry Pi • BeagleBone Functions: • Connects with various sensors and actuators. • Provides network ports for internet connectivity.
Smart Devices	The falling cost of controller boards allows embedding them in various objects for real-time data processing. Examples of Embedded Smart Devices: Consumer Electronics: Smartphones, smart tablets, smartwatches. Healthcare Industry: Glucometers, heart rate and blood pressure monitors, imaging systems. Automotive Industry: Tire pressure monitors, connected transportation services.

	Supply Chain Industry: RFID tagging for efficient tracking. Energy & Utilities Industry: Smart grids, smart meters.

4-Communication Layer



utilizes devices in device layer for communicating among themselves, and between them and the cloud
Protocols: Various communication protocols used in IoT, including:
Application Layer: MQTT, CoAP, AMQP, WebSocket, etc.
Transport Layer: TCP, UDP, etc.
Network Layer: IPv4, IPv6, etc.
Physical and Link Layer: Wi-Fi, Bluetooth, ZigBee, NFC, etc.

Communication Strategy	Communication Protocol	Application Protocol	Industrial Protocol
Gateway ■ Device ■ Smart (Edge) ■ Smartphone Connectivity ■ Direct ■ Device-to-device ■ API Connectivity	○ Wi-Fi ○ Ethernet ○ Cellular ○ Bluetooth Low Energy ○ RFID ○ NFC ○ ZigBee / Z-wave	○ MQTT ○ CoAP ○ AMQP ○ WebSocket ○ AllJoyn ○ DDS	○ BACnet ○ SCADA ○ Modbus

3-Core Platform Layer

- contains the cloud services and resources that **support resource management (in Analytic Platform Layer)** and processing of IoT tasks that reach the cloud.

- There are **no standardized protocols** where all vendors can converge on. Therefore, IoT stack needs to provide support for commonly used protocols, industry protocol and any evolving standards in future.
- Protocol Gateway is required to convert proprietary protocol to the protocol supported by the platform for communication.
    - In IoT stack, protocol gateway provides connectivity to the devices over the messaging platform or middleware like MQTT or AMQP

Messaging Middleware	• Acts as the backbone for message communication between enterprise systems.
	• Serves as an integration layer for various distributed systems.
	• Provides a highly scalable and high-performance middleware to support the rapid growth of connected devices.
	    ○ Example: 4.9 billion connected devices in 2015, projected to reach 25 billion by 2020.
	Key Functions:
	1. Device Management:
	    ○ Handles device registration and secure connectivity.
	    ○ Stores device data and ensures access control for authorized personnel.
	2. Data Handling & Storage:
	    ○ Holds device data for a specific interval.
	    ○ Uses a dedicated storage service to scale, compute, and analyze data efficiently.

Data Storage	• Handles the continuous data stream from connected devices.
	• Requires highly scalable storage services (terabytes of data).
	• Ensures fast data retrieval for real-time processing and analysis.
	Key Storage Technologies:
	1. NoSQL Databases (e.g., MongoDB, Cassandra)
	    ○ Enables faster data retrieval and computation.
	    ○ Simplifies processing for data analysis.
	    ○ Example:
	        ■ A connected car has a fixed ID field, but its speed value changes every second.
	        ■ The speed values are collected over time using a Time Series Database.
	2. Enterprise IoT Applications
	    ○ Stores structured, semi-structured, and unstructured data.
	    ○ Helps correlate different data types to derive actionable insights.
	    ○ Example:
	        ■ Equipment manual information (unstructured data) can be stored.

	■ Sensor data (structured data) can be correlated with manuals to generate functional alerts and suggest corrective actions.

2-Analytics Platform Layer

-

1-Solutions Layer

Solution Template: Includes device models, SDKs, application models, and machine learning models.
IoT Application: Customized applications that leverage fog computing to deliver intelligent solutions.
Summary
Device Layer: Physical components and sensors.
Communication Layer: Protocols and strategies for data transfer.
Core Platform Layer: Middleware, storage, and data processing.
Analytics Platform Layer: Resource management and data analysis.
Solutions Layer: End-user applications and services.

C3: Physical Component of IoT System

Embedded System Components

What are the challenges for IoT ?

1. Security and Privacy
    - Data Protection: Ensuring the security of data transmitted between devices and the cloud is a major concern. IoT devices often have limited processing power and storage, making it difficult to implement robust encryption.
    - Authentication and Authorization: Managing access control and ensuring that only authorized users and devices can interact with the system.
    - Privacy: Protecting user privacy, especially when IoT devices collect sensitive data such as health information or personal habits.
2. Interoperability
    - Protocol Diversity: IoT devices often use different communication protocols (e.g., MQTT, CoAP, ZigBee, Z-Wave), making it challenging to ensure seamless communication between devices from different manufacturers.
    - Standardization: Lack of standardized protocols and architectures can lead to fragmentation and difficulties in integrating devices into a cohesive system.
3. Scalability
    - Device Management: Managing a large number of devices, especially as the number of connected devices grows exponentially.
    - Data Handling: Processing and storing the vast amounts of data generated by IoT devices, which can strain existing infrastructure.
4. Reliability and Performance
    - Network Latency: Ensuring low-latency communication, especially for time-sensitive applications like industrial control or healthcare monitoring.
    - Device Reliability: Ensuring that IoT devices operate reliably in various environmental conditions and over long periods.
5. Power Consumption
    - Battery Life: Many IoT devices are battery-powered, and optimizing power consumption to extend battery life is crucial.
    - Energy Harvesting: Implementing energy harvesting techniques to power devices in environments where traditional power sources are not available.
6. Complexity in Deployment and Maintenance
    - Installation: Deploying IoT systems can be complex, requiring specialized knowledge and skills.
    - Maintenance: Regular updates and maintenance are necessary to keep devices secure and functioning correctly, which can be challenging for large-scale deployments.
7. Cost
    - Initial Investment: The cost of purchasing and deploying IoT devices and infrastructure can be high.
    - Operational Costs: Ongoing costs for data storage, processing, and network connectivity can add up over time.
8. Regulatory and Compliance Issues

    - Regulations: Navigating the complex regulatory landscape, especially for IoT devices that collect personal data or are used in critical sectors like healthcare or finance.
    - Compliance: Ensuring that IoT systems comply with local and international regulations, such as GDPR in Europe or HIPAA in the United States.

Infrastructure and Technology

Arduino

- Description: Arduino is an open-source electronics platform based on easy-to-use hardware and software.
- Features:
    - Can read inputs (e.g., light on a sensor, a finger on a button) and turn them into outputs (e.g., activating a motor, turning on an LED).
    - Uses the Arduino programming language (based on Wiring) or the Arduino Software (IDE) for programming.
- Community: A worldwide community of makers, including students, hobbyists, artists, programmers, and professionals, contributes to the platform.
- Applications: Suitable for a wide range of projects, from simple home automation to complex industrial applications.
- Resource: Arduino Official Website

BeagleBone

- Description: BeagleBone is a low-cost, fan-less single-board computer based on low-power Texas Instruments processors.
- Features:
    - Features the ARM Cortex-A series core with expandability similar to desktop machines.
    - Various designs available, including BeagleBone Black, Blue, Green, AI, etc.
- Applications: Ideal for projects requiring more processing power than Arduino, such as robotics, media servers, and industrial control.
- Resource: BeagleBoard.org

Espressif

- Description: Espressif provides low-cost, low-power system-on-chip microcontrollers with integrated Wi-Fi and dual-mode Bluetooth.
- Features:
    - The ESP32 series employs a Tensilica Xtensa LX6 microprocessor in dual-core and single-core variations.
    - Includes built-in antenna switches, power amplifier, low-noise receive amplifier, filters, and power-management modules.
- Applications: Suitable for IoT devices requiring Wi-Fi and Bluetooth connectivity, such as smart home devices and wearables.
- Resource: Espressif Official Website

Raspberry Pi

- Description: Raspberry Pi is a series of small single-board computers developed to promote the teaching of basic computer science.
- Features:
    - Provides low-cost, high-performance computers for learning, problem-solving, and fun.
    - Supports various operating systems, including Raspbian (a Debian-based OS).

- Includes GPIO pins for interfacing with hardware components.
- Applications: Widely used in education, home automation, media centers, and general-purpose computing.
- Resource: [Raspberry Pi Foundation]

Raspberry Pi series

Source: http://socialcompare.com/en/comparison/raspberrypi-models-comparison

	Raspberry Pi 4 Model B	Raspberry Pi 3 Model B+	Raspberry Pi 2 Model B v1.2	Raspberry Pi Model B+	Raspberry Pi Model A+
Processor Chipset	Broadcom 2711, Cortex-A73, 64-bits SoC	Broadcom BCM2837B0, Cortex-A53, 64-bits SoC	Broadcom BCM2837 64Bit SoC	Broadcom BCM2835 32Bit SoC full HD	Broadcom BCM2835 32Bit SoC full HD
Processor Speed	Quad-core @ 1.5GHz	Quad-core @1.2 GHz	Quad-core @900 MHz	Single Core @700 MHz	Single Core @700 MHz
RAM	1GB, 2GB, 4GB LPDDR4 SDRAM	1GB SDRAM @ 400 MHz	1GB SDRAM @ 400 MHz	512 MB SDRAM @ 400 MHz	256 MB SDRAM @ 400 MHz
Storage	MicroSD	MicroSD	MicroSD	MicroSD	MicroSD
USB 2.0	2 USB 2.0, 2 USB 3.0	4x USB Ports	4x USB Ports	4x USB Ports	1x USB Port
Max Power Draw/voltage	3A @ 5V	2.5A @ 5V	1.8A @ 5V	1.8A @ 5V	1.8A @ 5V
GPIO	40 pin	40 pin	40 pin	40 pin	40 pin
Ethernet Port	Yes	Yes	Yes	Yes	No
WiFi	Built in	Built in	No	No	No
Bluetooth LE	Built in	Built in	No	No	No

Python Programming

- Description: Python is a versatile programming language widely used in IoT development.
- Features:
  - Easy to learn and use, with a large standard library and extensive community support.
  - Suitable for developing applications, scripts, and automation tasks.
- Applications: Used for developing software on IoT devices, such as Raspberry Pi and Arduino.
- Resource: [Python Official Website]

Node-RED

- Description: Node-RED is a flow-based programming tool originally developed by IBM.
- Features:
  - Allows users to create applications by dragging nodes from a palette into a workspace and wiring them together.
  - Extensible with community-created nodes and can be shared as JSON files.
- Applications: Ideal for creating IoT workflows, integrating devices, and automating tasks.
- Resource: [Node-RED Official Website]

C4: Processing Requirement in IoT System

Common Communication Standards, Protocols, Security and Privacy

Data Communication Standards and Protocols

Serial Communication

- Universal Asynchronous Receiver Transmitter (UART):
  - Uses a single data line for transmitting and one for receiving data.
  - Common voltage levels: 3.3V or 5.0V.
  - Data format: 1 start bit (low level), 8 data bits, 1 stop bit (high level).
  - Synchronization: Asynchronous communication, relies on start and stop bits.
  - Use case: Suitable for short-range, low-speed communication between devices.
- Universal Synchronous/Asynchronous Receiver Transmitter (USART):
  - Combines UART with synchronous communication capabilities.
  - Adds a clock signal to improve timing accuracy.
  - Use case: Provides more reliable communication over longer distances compared to UART.
- Electronic Industries Association (EIA) Standards:
  - RS-232: Common data format for serial communication, suitable for short distances.
  - RS-485: Suitable for longer distances and higher noise environments.

Serial Peripheral Interface (SPI)

- Master sends a clock signal, and data is transferred bit by bit.
- Signal names:
  - SCLK: Serial Clock
  - MOSI: Master Out Slave In
  - MISO: Master In Slave Out
  - SS: Slave Select (allows multiple slaves on the bus)
- Use case: High-speed communication between a master device and one or more slave devices.

Inter-Integrated Circuit (I^2C)

- Uses two wires:
  - SCL: Serial Clock
  - SDA: Serial Data
- Supports multiple devices on the same bus.
- Data transfer:
  - Initiated with a start bit (SDA pulled low while SCL is high).
  - Data bits are sampled on the rising edge of SCL.

    - Ends with a stop bit (SDA pulled high while SCL is high).
- Use case: Suitable for low-speed communication between multiple devices on a single bus.

Communication Strategy

- Gateway:
    - Device Gateway: Connects incompatible networks or protocols, enabling devices to communicate with the Internet.
        - Example: Sensors connect to a gateway via Bluetooth and communicate with a broadband router via Wi-Fi.
    - Smart Gateway (Edge Gateway): Has local storage and embedded applications for data analytics.
        - Example: Fog computing allows data to be processed locally without sending it over the network.
    - Smartphone as Gateway: Utilizes Bluetooth, cellular networks, and sensors for communication.
        - Requires manual intervention to set up the communication process.
- Connectivity:
    - Direct Connectivity: Devices connect directly to the Internet via Wi-Fi or cellular networks.
        - Example: SmartTVs connecting directly to the Internet for streaming services.
    - Device-to-Device Connectivity: Forms mesh networks for information sharing and alerting.
        - Example: Home security systems where devices communicate directly with each other.
    - API Connectivity: Uses RESTful APIs for communication between services.
        - Example: A device triggers an action on another service provider via a REST URL.
        - Eliminates the need for a shared gateway.

Communication Protocols

- Application Protocols:
    - MQTT (Message Query Telemetry Transport):
        - Lightweight, publish/subscribe model with hierarchical topics.
        - Suitable for low-bandwidth networks.
        - Example: Devices publish data to a broker, which then forwards it to subscribers.
    - CoAP (Constrained Application Protocol):
        - Optimized for devices with limited power and processing capabilities.
        - Uses UDP for message-based communication and HTTP for request/response interactions.
        - Example: Sensors sending data to a server with minimal power consumption.
    - AMQP (Advanced Message Queuing Protocol):
        - Supports queue-based and publish/subscribe models.
        - Operates on TCP transport.
        - Example: Used in enterprise systems for reliable message delivery.
    - WebSocket:

        - Standardized by W3C and IETF.
        - Maintains a persistent connection between client and server over a single TCP connection.
        - Example: Real-time communication between a web application and a server.
    - AllJoyn:
        - Open-source framework for device discovery, publishing, and communication.
        - Promotes interoperability between different platforms (e.g., Android, iOS, Linux).
        - Example: Enabling seamless communication between smart home devices.
    - DDS (Data Distribution Services):
        - Part of the Object Management Group (OMG) IoT standards.
        - Enables interoperability between machines, enterprise systems, and mobile devices.
        - Example: Used in industrial automation for real-time data distribution.
- Industrial Protocols:
    - BACnet:
        - Enables smart building automation by providing infrastructure for monitoring and controlling devices.
        - Supports information exchange between devices for handling communication in various areas.
        - Example: Used in HVAC systems for energy management.
    - SCADA (Supervisory Control and Data Acquisition):
        - Enables remote control and monitoring of industrial devices.
        - Includes RTUs (Remote Terminal Units) and PLCs (Programmable Logic Controllers) for processing sensor data.
        - Example: Used in manufacturing plants for process control.
    - Modbus:
        - Serial communication protocol used in SCADA networks.
        - Uses standard RS-485 and RS-232 interfaces.
        - Example: Used for communication between industrial controllers and sensors.

Security and Privacy

- Data Security and Privacy:
    - Potential risks of data breaches and unauthorized access.
    - Importance of securing data transmission and storage.
    - Example: RFID tags in passports can be read from a distance, posing a security risk.
- Object-Level Security:
    - Moving security closer to the object as technology advances.
    - Addressing challenges related to identification, authentication, and data access.
    - Example: Smart meters with tamper detection to prevent unauthorized access.
- Changing Security Priorities:

  - Increasing importance of security as IoT becomes more integrated with critical infrastructure.
  - Example: Security failures in automotive systems can have severe consequences.
  - Role of object gateways and consolidators in enhancing security.
- Standardization:
  - Importance of standardization in addressing security and privacy issues.
  - Future discussions on trends in IoT evolution and standardization.

Summary

- Data Communication Standards and Protocols: UART, USART, SPI, I^{2}C, RS-232, RS-485.
- Communication Strategy: Gateway (Device, Smart, Smartphone), Connectivity (Direct, Device-to-Device, API).
- Communication Protocols: MQTT, CoAP, AMQP, WebSocket, AllJoyn, DDS, BACnet, SCADA, Modbus.
- Security and Privacy: Data security, object-level security, changing priorities, standardization.

C5: IoT Example

Smart Home, Smart Campus, and Industrial IoT Applications

Overview

1. Smart Home
2. Smart Campus
3. Smart Manufacturer
4. Smart Retailer
5. Smart City

Smart Home

What is the Current Status of Smart Home?

- Vendors in Malaysia:
  - Icon2U: https://www.icon2u.com.my/
  - Vyrox: https://vyrox.com/
  - SmartZone: https://www.smartzone.info/
- Global Platforms:
  - Amazon Alexa: https://www.amazon.com/alexa-smart-home/
  - Apple HomeKit: https://www.apple.com/my/ios/home/
  - Google Home: https://developers.google.com/actions/smarthome/
  - Samsung SmartThings: https://www.samsung.com/us/smartthings/
  - Nest Labs: https://nest.com/
  - HiveHome: https://www.hivehome.com/
  - Intel Smart Home: https://www.intel.com/content/www/us/en/internet-of-things/overview.html

Home Automation Using IoT

- Definition: A system that uses computers or mobile devices to control basic home functions and features automatically through the Internet from anywhere in the world.
- Purpose: To provide conveniences to human, save electric power, and reduce human energy.
- Technologies Needed:
  - Sensors/hardware interface module
  - WiFi
  - Home Automation Server

Home Automation System Functions and Capabilities

- Capabilities to control and monitor the following components:
  - Temperature and humidity

- Motion detection
- Fire and smoke detection
- Light level
- Control of lights (on/off/dim)
- Control of fans (on/off)
- Control of other home appliances

- Example:
  - If the temperature exceeds a threshold, the cooler turns on automatically.
  - If gas leakage is detected, an alarm is raised.
  - Lights turn on/off automatically based on external light conditions.
  - Users can monitor and control appliances remotely via the Internet.

Home Automation Systems (Functions)

- Connectivity: The nervous system of the home, connecting and routing data seamlessly.
- Voice: Spoken interfaces that recognize speech patterns.
- Vision: Vision processors and algorithms that detect objects and understand incidents.
- Cognition: The ability to compute and make sense of data from devices to understand context.

Home Automation Systems (Software Design)

Front End Design: HTML

Cloud Storage: Gmail (example of cloud storage)

Cloud Computing: Using remote servers to manage, store, and process data

Categories:

Infrastructure-as-a-Service (IaaS)	• Follows traditional utilities model • Provides services on demand with the consumer paying accordingly
Platform-as-a-Service (PaaS)	• Allows application constructions within a provider's framework • Eg.: Google's App Engine
Software-as-a-Service (SaaS)	• Enables customers use application via browser • Eg.: Gmail, where u can access ur stored data from any computer with internet

Advantages of Home Automation System

Wireless Home Automation Systems:	Wired Home Automation System:
• Reduced installation costs	• Higher installation costs

○ no cabling required ● System scalability and easy extension ○ New connection requirement is easy ● Aesthetic benefits ○ Allows building with all-glass architecture	○ due to expensive cabling ● Rewiring is required for changes in system layout ○ Cabling process is tedious and labor-intensive ● Better suited for planned construction with hiding wiring ○ more complex and costly

Conclusion

- The home automation system using IoT has been proven to work effectively by connecting simple appliances and allowing remote control via the Internet.
- The system monitors sensor data and also actuates a process according to the requirement (For example: switching on the light when it gets dark.)
- Future work could include expanding the system to include home security features, energy monitoring, or environmental monitoring.

Smart Campus

Case Study: Huawei Solution

- Challenges:
  - Scattered resources and classrooms reduce flexibility and efficiency.
  - Lack of energy consumption monitoring leads to wasted power and increased costs.
  - Old video surveillance systems do not support incident warnings, reducing campus safety.
  - Terminals do not support unified management, affecting teaching and administration.
- Huawei's Smart Campus Solution:
  - Cloud Resource Sharing: Distributed cloud architecture enables local caching and resource sharing.
  - Energy Consumption Monitoring: Advanced Metering Infrastructure (AMI) monitors power consumption.
  - Safe Campus: Intelligent analysis functions (e.g., intrusion detection) enhance security.
  - Smart Management: Interconnected devices support unified management and collaboration.

Smart Manufacturer

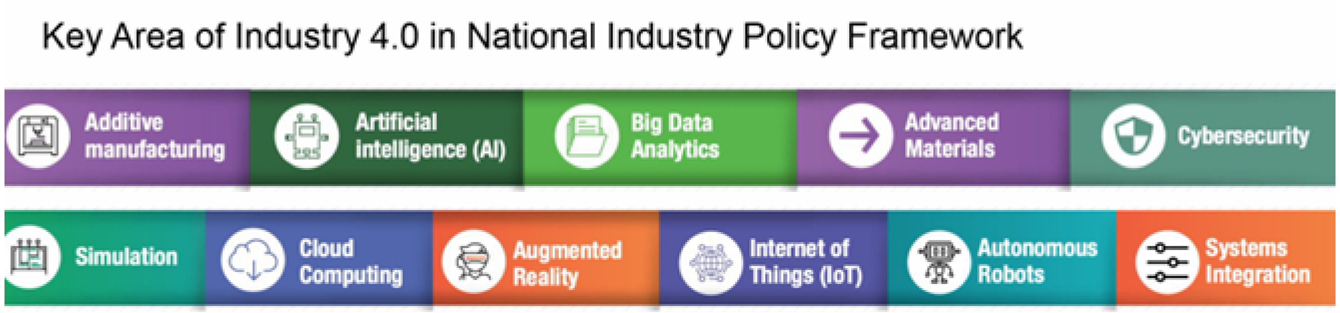
Overview

- Also known as:
  - Smart Factory
  - Industrial Internet of Things
  - Industry 4.0
  - Cyber-Physical System
- Components:

- Input/Output Connectivity
- Communication
- Assembly
- Production
- Process Monitoring
- Controlling
- Warehouse Management
- Human-Machine Interface

Revolution of Industry 1.0 to 4.0

- Industry 1.0: Mechanization using water and steam power.
- Industry 2.0: Mass production with the help of electricity.
- Industry 3.0: Further automation and computerization.
- Industry 4.0: Cyber-physical systems, IoT, and big data analytics.



IOT Works with Various Latest Technologies in IR4.0

- Centralized Panel Control
- Portable Mobile Application
- Augmented/Mixed Reality

Industry 4.0 Framework

- New Industrial Master Plan 2030
- Key Area of Industry 4.0 in National Industry Policy Framework
- Reference Architectural Model Industrie 4.0 (RAMI 4.0)

Smart Retailer

Challenges

- Customer tastes and demands change quickly.
- Leading retailers need to stay ahead of demand to survive.

- Key components for a smarter retailer:
    - Business Process Management (BPM)
    - Business Rules Management (BRM)
    - Service-Oriented Architecture (SOA)

Objectives

- Better demand forecasting
- Elimination of stock-outs
- Shorter lead time for new products
- Optimized retail space usage
- Unified experience between online and in-store shopping
- More efficient logistics
- Responsive value chain

Examples

- T-Systems International GmbH
- Microsoft Solution using Azure and Windows 10 IoT

Smart City

Key Drivers of the Smart City

- Population Growth and Urbanization: Threaten to worsen existing problems like congestion, crime, smog, and aging infrastructure.
- Climate Change and Resource Limitations: Increasing urbanization, need for resource conservation.
- Interconnected Sectors: Mobility, energy, and other sectors are merging.
- Complex Challenges: Integrated solutions are needed to address issues like CO_2 emissions and energy supply.
- Changing Human Behavior: Shifts like car-sharing and intermodal transportation.

Smart City Examples

- Chicago:
    - Investment in infrastructure (fiber-optic ring for gigabit speed).
    - Focus on economic development and community engagement.
    - Digital Skills Initiative, Connectivity, Smart Health Centers.
- Rio de Janeiro:
    - Safety and security (disaster prevention and management).
    - Center of Operations for emergency response.
    - Press room for media and information flow.
- Stockholm:
    - Focus on citizen engagement and open communication.
    - Energy and transport efficiency (Royal Seaport smart district).

    - Kista Science City (ICT innovation cluster).
- Boston:
    - Smart Streets (cameras, sensors, data analysis).
    - Hubway (public bike share system).
    - Self-driving cars, Wicked Free Wi-Fi, App Showcase (e.g., parking tickets, pothole reporting).
- Participatory Urbanism:
    - Citizens Connect: Mobile app for reporting service problems.
    - Community PlanIt: Online platform for community meetings.
    - Innovation District Welcome Home Challenge: Business growth competition.

Malaysia Smart City Initiatives

- Iskandar Malaysia Urban Observatory (IMUO)
- Malaysia Smart City Framework (2018)
- Putrajaya Smart City Blueprint
- Kuala Lumpur Smart City Master Plan 2021 - 2025
- Smart Selangor
- Melaka Maju Jaya 2035
- Penang2030
- MCMC - Smart City

C6: IoT Example

Consumer, Commercial, and Infrastructural IoT Applications

Case Study: Logistic / Supply Chain Management

Drug Discovery and Supply Chain

- Stages:
  - Drug Discovery
  - Development
  - Manufacturing
  - Warehousing
  - Distribution
  - Sales & Marketing
  - Patients
- IoT Applications:
  - Remote Monitoring: Real-time tracking of assets and conditions.
  - Predictive Maintenance & Analytics: Monitoring equipment health to prevent failures.
  - Real-time Visibility: Enhanced visibility across the supply chain.
  - Warehouse Operations: Optimized management of inventory.
  - Cold Chain Condition & Tracking: Ensuring proper storage conditions for sensitive products.
  - Shop Floor Visibility & Analytics: Real-time insights into production processes.
  - Traceability: Tracking products from origin to end-user.
  - Higher Fleet Efficiency: Remote management of transportation assets.

Smart Retailer

Challenges

- Customer Demand: Rapidly changing tastes and preferences.
- Business Process Management: Key components include BPM, BRM, and SOA.
- Objectives:
  - Better demand forecasting.
  - Elimination of stock-outs.
  - Shorter lead times for new products.
  - Optimized retail space usage.
  - Unified experience between online and in-store shopping.
  - More efficient logistics.
  - Responsive value chain.

Examples

- T-Systems International GmbH
- Microsoft Solution using Azure and Windows 10 IoT
  - [Microsoft Azure Architecture for Retail]

Smart City

Key Drivers

- Population Growth and Urbanization: Increased strain on existing infrastructure.
- Climate Change and Resource Limitations: Need for sustainable urban development.
- Interconnected Sectors: Mobility, energy, and other sectors are merging.
- Complex Challenges: Integrated solutions for CO2 emissions and energy supply.
- Changing Human Behavior: Shifts towards shared mobility and intermodal transportation.

Examples

- Chicago:
  - Investment in infrastructure (fiber-optic ring for gigabit speed).
  - Focus on economic development and community engagement.
  - Digital Skills Initiative, Connectivity, Smart Health Centers.
- Rio de Janeiro:
  - Safety and security (disaster prevention and management).
  - Center of Operations for emergency response.
  - Press room for media and information flow.
- Stockholm:
  - Focus on citizen engagement and open communication.
  - Energy and transport efficiency (Royal Seaport smart district).
  - Kista Science City (ICT innovation cluster).
- Boston:
  - Smart Streets (cameras, sensors, data analysis).
  - Hubway (public bike share system).
  - Self-driving cars, Wicked Free Wi-Fi, App Showcase (e.g., parking tickets, pothole reporting).
- Participatory Urbanism:
  - Citizens Connect: Mobile app for reporting service problems.
  - Community PlanIt: Online platform for community meetings.
  - Innovation District Welcome Home Challenge: Business growth competition.

Malaysia Smart City Initiatives

- Iskandar Malaysia Urban Observatory (IMUO)

- Malaysia Smart City Framework (2018)
- Putrajaya Smart City Blueprint
- Kuala Lumpur Smart City Master Plan 2021 - 2025
- Smart Selangor
- Melaka Maju Jaya 2035
- Penang2030
- MCMC - Smart City