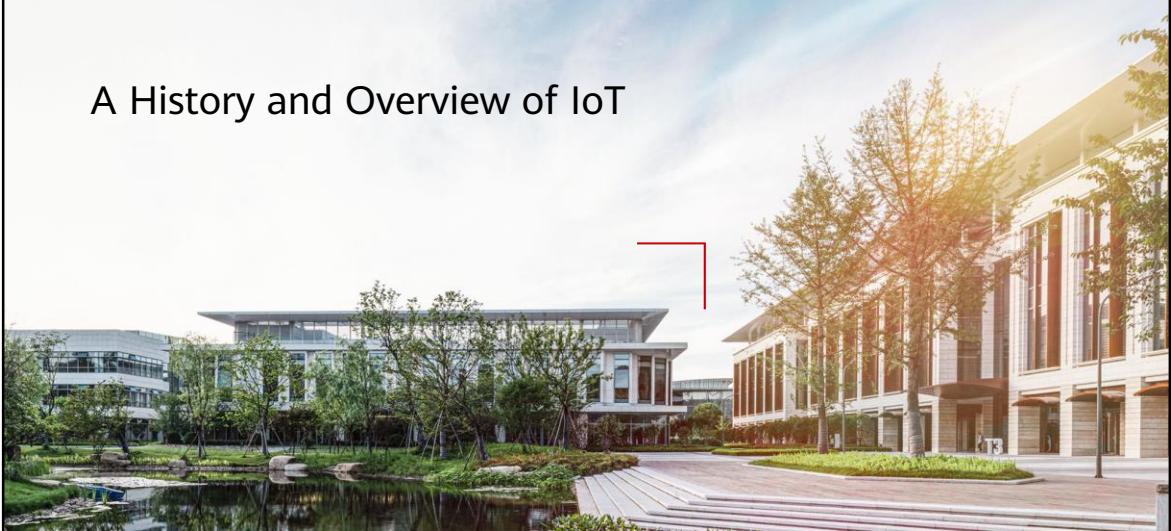


A History and Overview of IoT



Foreword

- The Internet of Things (IoT) is a next-generation information technology.
- This is a milestone of the information era and is known as the third wave of industry development after computers and the Internet. It is widely used for network convergence, integrating communications and sensing technologies (intelligent sensing, identification, and pervasive computing).

Objectives

- Upon completion of this course, you will have learned:
 - IoT development history
 - Basic IoT concepts
 - IoT architecture layers
 - How IoT engineering relates to engineers.

Contents

- 1. IoT Development History**
2. IoT Overview and Architecture
3. Huawei IoT Solution
4. IoT Engineering and Engineers

Origin of the IoT

- Trojan Room coffee pot in 1991: At the Computer Laboratory in Cambridge University, scientists would go downstairs only to find that the coffee was not yet ready.
- To solve this problem, they wrote a program, installed a portable camera next to the coffee pot, and used image capture technology to check whether the coffee was ready.

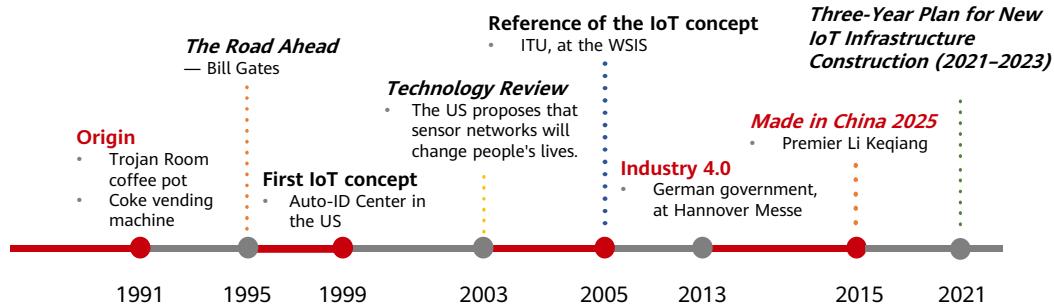


5 Huawei Confidential

 HUAWEI

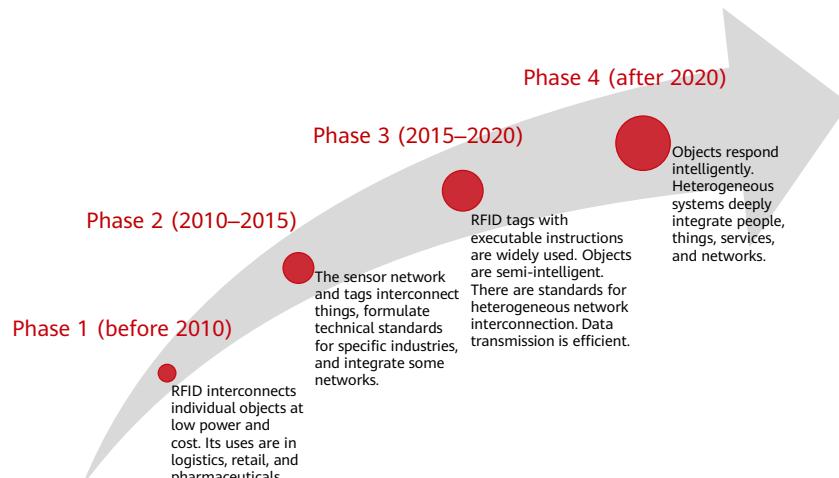
- There is a "Coke" version about the origin of IoT. In 1990, some programmers at the Carnegie Mellon University decided to modify a Coca-Cola vending machine to detect if the cans were cold enough.

Development of the IoT



- The concept of the IoT dates back to Bill Gates's book, *The Road Ahead* 1995. However, due to the limited development of wireless networks, hardware, and sensors, his idea did not attract much attention.
- On November 17, 2005, the International Telecommunication Union (ITU) released the *Internet Reports 2005: The Internet of Things* at the World Summit on the Information Society (WSIS). The report pointed out that the ubiquitous IoT era was coming. All objects in the world, from tires to toothbrushes, and from houses to tissues, can exchange information through the Internet.
- On August 7, 2009, China's former Premier Wen Jiabao delivered an important speech during the inspection in Wuxi city, proposing the "Sensing China" strategy and saying that China should seize the opportunity and vigorously develop IoT technologies. Later on November 3 in the same year, he delivered a speech to the science and technology circle of Beijing. He emphasized again the importance of strategic emerging industries and instructed that China should focus on making breakthroughs in sensor networks and the IoT.
- Smart Logistics, proposed in Industry 4.0, integrates logistics resources through the Internet, IoT, and logistics networks, to maximize the efficiency of existing logistics resource suppliers. So demanders can quickly obtain service matching and logistics support.

EU and National: IoT Industry Development Plans



- South Korea's government proposes the concept of ubiquitous sensing network.
- The IT Strategy Department of the Japanese government has formulated a next-generation informatization strategy.

IoT Key Event: LoRa International Approval



In December 2021, LoRaWAN was recognized by the International Telecommunication Union (ITU) as standard. The approval affirms the IoT achievements and prospects of LoRa and its ecosystem.

- The *ITU-T Recommendation Y.4480: Low Power Protocol for Wide Area Wireless Networks* is delivered by the Study Group 20 of International Telecommunication Union-Telecommunication Standardization Sector (ITU-T) and was approved by ITU. LoRaWAN has officially become a global standard, boosting the IoT industry.
- Image source: <http://3ms.huawei.com/km/static/image/detail.html?fid=61070>

IoT Key Event: NB-IoT Officially Incorporated into 5G Standards



ITU-R WP5D meeting on July 9, 2020: **NB-IoT and NR are now 5G standards.** So far, NB-IoT has been officially defined as the cornerstone for building a fully connected, intelligent world.

- Image source: <http://3ms.huawei.com/km/static/image/detail.html?fid=61941>
- NB-IoT can be deployed in NR in-band mode to support access to the 5G core network. In addition, NB-IoT 3GPP specifications will evolve and become the core standard for 5G mMTC scenarios. 5G mMTC requires 1 million device connections per square kilometer. In this case, NB-IoT can provide 100,000 connections with its 200 kHz frequency. From the technical perspective, NB-IoT does meet ITU's requirements for low power consumption and wide 5G coverage.
- 5G can better satisfy e-commerce industries in terms of IoT development and help them to go digital. NB-IoT is designed for the communication between things. In recent years, NB-IoT has been widely used, improving and even changing the production and operations of some industries. With NB-IoT applications, smart parking, smart firefighting, smart water, smart street lamps, bicycle sharing, and smart home appliances have greatly changed consumer behavior. NB-IoT is a stepping stone for 5G to enter the e-commerce industry.

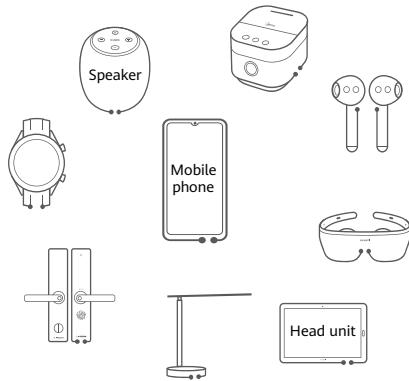
IoT Industry Status: Consumer IoT

- Consumer IoT: Products and services that we directly buy and use in all aspects of life.

Examples: Smart bands, VR/AR glasses, body fat scales, smart locks, smart speakers, shared bikes/electric bikes, and autonomous driving.

MarketsandMarkets reports that the global consumer IoT market was USD46.8 billion in 2018 and will reach **USD104.4 billion** by 2023.

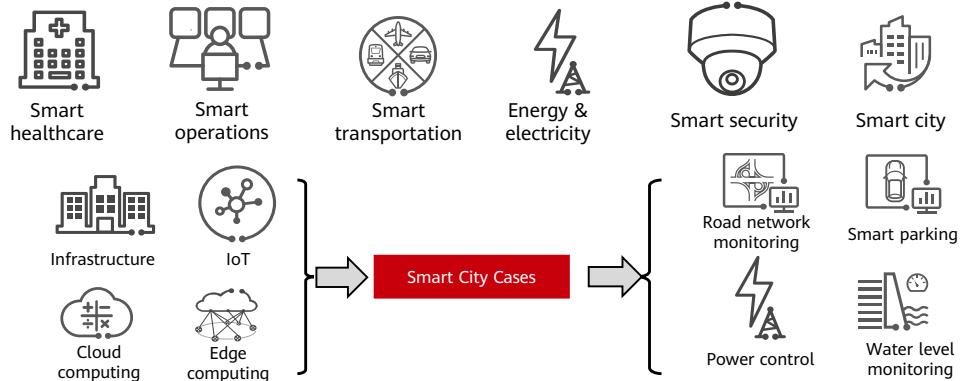
The compound annual growth rate (CAGR) from 2018 to 2023 is 17.39%.



- Currently, there are consumer IoT and industry IoT. Industry IoT is the combination of policy-driven IoT and production IoT.

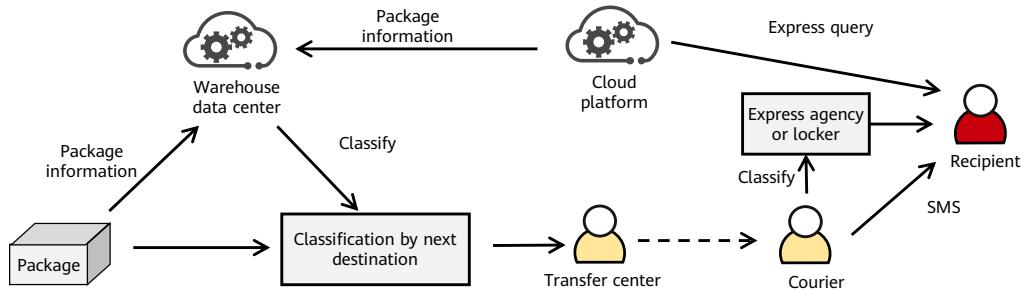
IoT Industry Status: Policy-Driven IoT

- Mainly used for city management (fire fighting, security, system integration, public utilities, lighting, and parking).

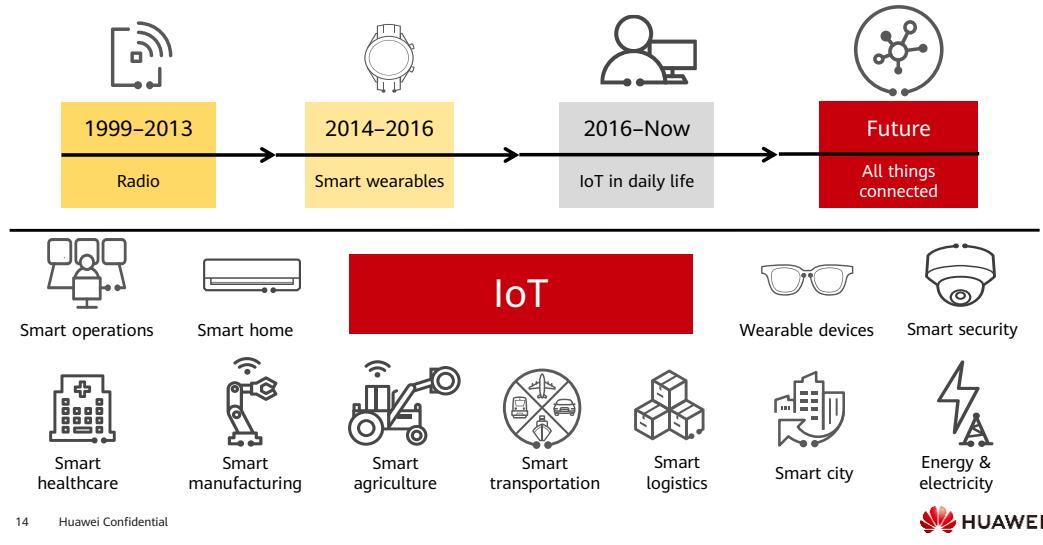


IoT Industry Status: Production IoT

- Covers smart industry, Internet of Vehicles (IoV), smart logistics, and smart agriculture. For example, an IoT application checks the status of a package periodically and uploads this data to the cloud.



Application and Development of IoT



Contents

1. IoT Development History
- 2. IoT Overview and Architecture**
3. Huawei IoT Solution
4. IoT Engineering and Engineers

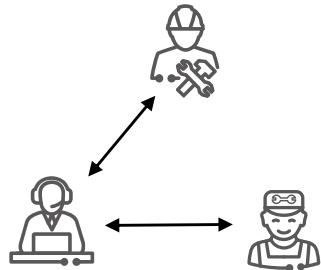
What is IoT?



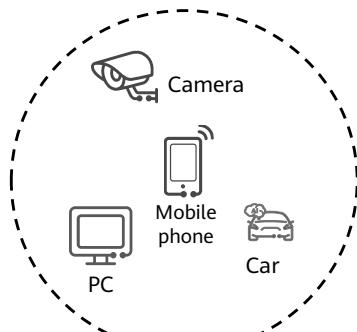
By making network access ubiquitous, IoT connects things-to-things and things-to-humans. It offers intelligent sensing, identification, and management of things and processes based on sensors, radio frequency identification (RFID), global positioning systems (GPSs), and other technologies. IoT is an information carrier built on the Internet and traditional telecommunications networks. It connects all common objects with independent functions through networks. IoT is an Internet where all things are interconnected.

- IoT is an extension and expansion of the internet. It connects sensors with the internet to form a giant network, implementing interconnection of people, machines, and things anytime, anywhere.
- Image source: <http://3ms.huawei.com/km/static/image/detail.html?fid=61711>

IoT: From Internet of People to Internet of Things

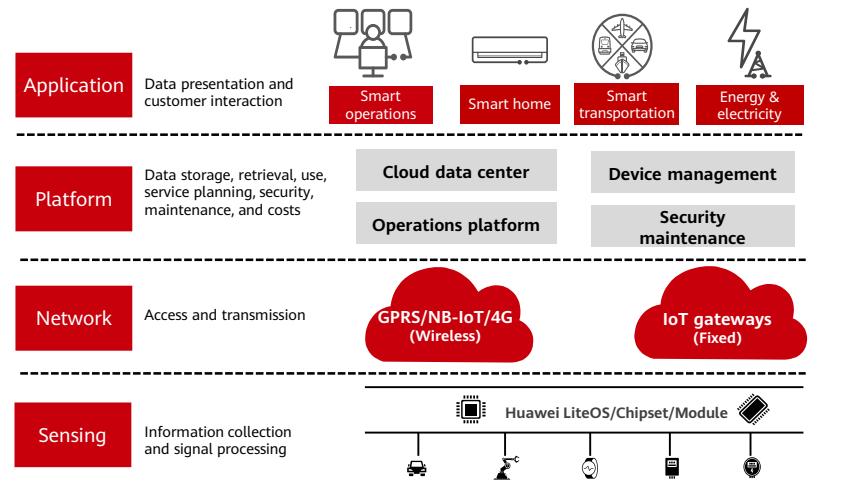


Information input and shared between people is output as text, voice, or video.



Information shared between devices is either of the device itself or collected by the device. Examples: device status and the light intensity of the surroundings.

Layers of the IoT



18 Huawei Confidential



- The sensing layer collects information and processes signals. The sensing and identification technologies enable objects to "speak" and distinguish the IoT from other networks. The sensing layer also includes smartphones, Pads, multimedia players, and laptops that can manually generate information. The sensing layer is the IoT foundation.
- The network layer directly accesses and transmits information from the sensing layer by using the Internet, mobile communications network, and satellite communications network. The network layer connects to the sensing layer and the platform layer, and has a strong link function.
- In a high-performance network computer environment, the platform layer can use computers to integrate massive information resources in the network into a large intelligent network that can be interconnected. This layer can solve the data problems of storage (database and mass storage technologies), retrieval (search engine), use (data mining and machine learning), and protection against abuse (data security and privacy protection). The platform layer is above the sensing and network layers and below the application layer. It is the brain of the IoT. That's why the IoT applications are usually named as "smart", such as smart grid, smart transportation, and smart logistics.
- The application layer is a user interface of the IoT system and provides abundant specific services for users by analyzing processed sensor data. Specifically, the application layer receives information from the network layer, processes the information, makes a decision, and then sends the information by using the network layer, to control devices at the sensing layer. IoT applications are centered on the "objects" or physical world and cover object tracking, environment sensing, smart logistics, smart transportation, and smart customs.

Contents

1. IoT Development History
2. IoT Overview and Architecture
- 3. Huawei IoT Solution**
4. IoT Engineering and Engineers

Huawei 1+2+1 IoT Solution

Smart home Smart transportation Smart parking Smart metering



IoT platform

Home gateway
Intelligent gateway for enterprises

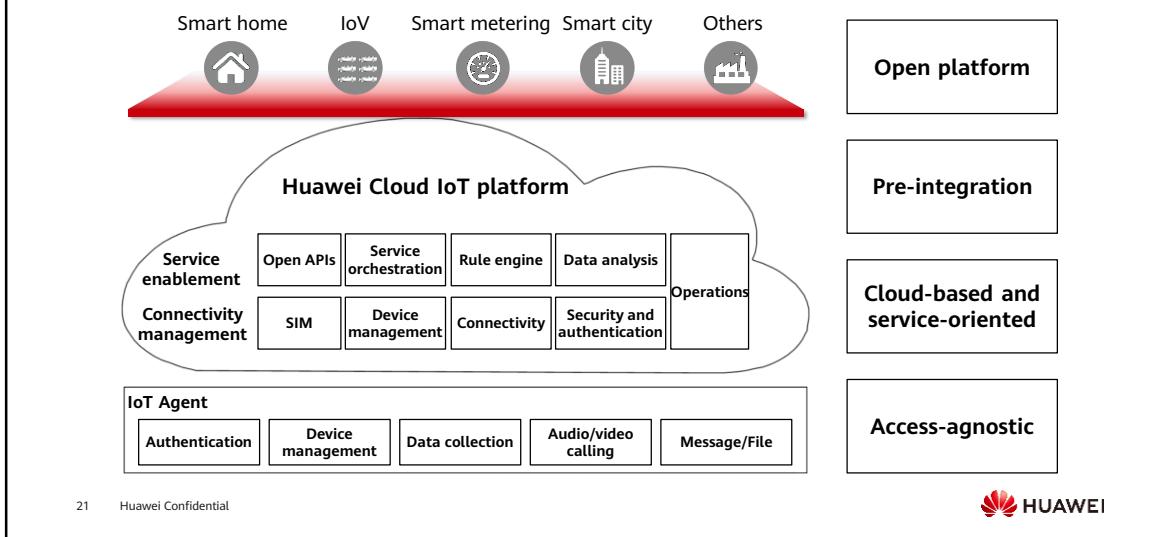
2G/3G/4G/
NB-IoT/5G

Huawei LiteOS/modules



- One IoT platform, two access methods, and one IoT operating system

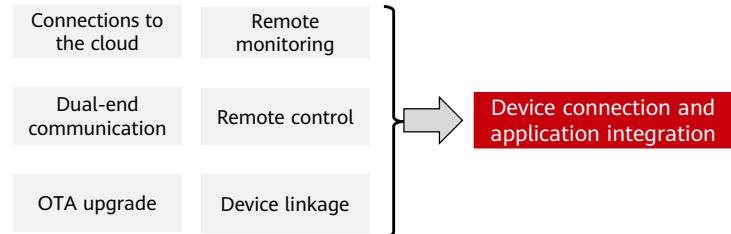
Architecture of the Huawei Cloud IoT Platform



- A secure, reliable platform that supports decoupling from devices and applications is required to develop IoT services.
- The Huawei Cloud IoT platform provides device access, security verification, service orchestration, data management, and multi-protocol communications, implements connectivity management and capability openness, and connects to carriers' pipes and platforms.

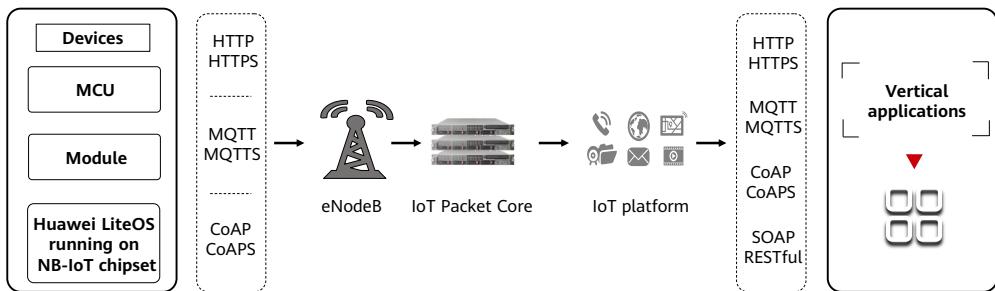
Huawei Cloud IoTDA

- IoT Device Access (IoTDA) is a Huawei Cloud IoT platform.
- IoTDA simplifies device access in all scenarios while providing high-concurrency communications and full-link self-diagnosis. IoTDA makes solutions cost-effective.



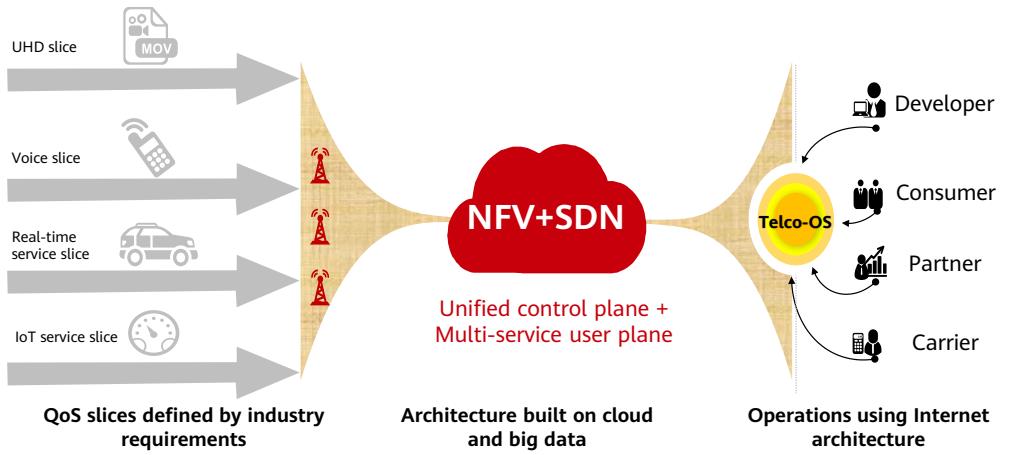
NB-IoT: Connecting Devices over Cellular Networks

- NB-IoT end-to-end solution



- NB-IoT: Narrowband Internet of Things
- NB-IoT is a new narrowband cellular communications LPWAN technology.
LPWAN stands for low-power wide-area network. NB-IoT has strong connection, wide coverage, low cost, and low power consumption.
- CoAP: Constrained Application Protocol
- MQTT: Message Queuing Telemetry Transport
- SOAP: Simple Object Access Protocol
- HTTP: Hypertext Transfer Protocol

5G Architecture: One Network Behind Hundreds of Industries

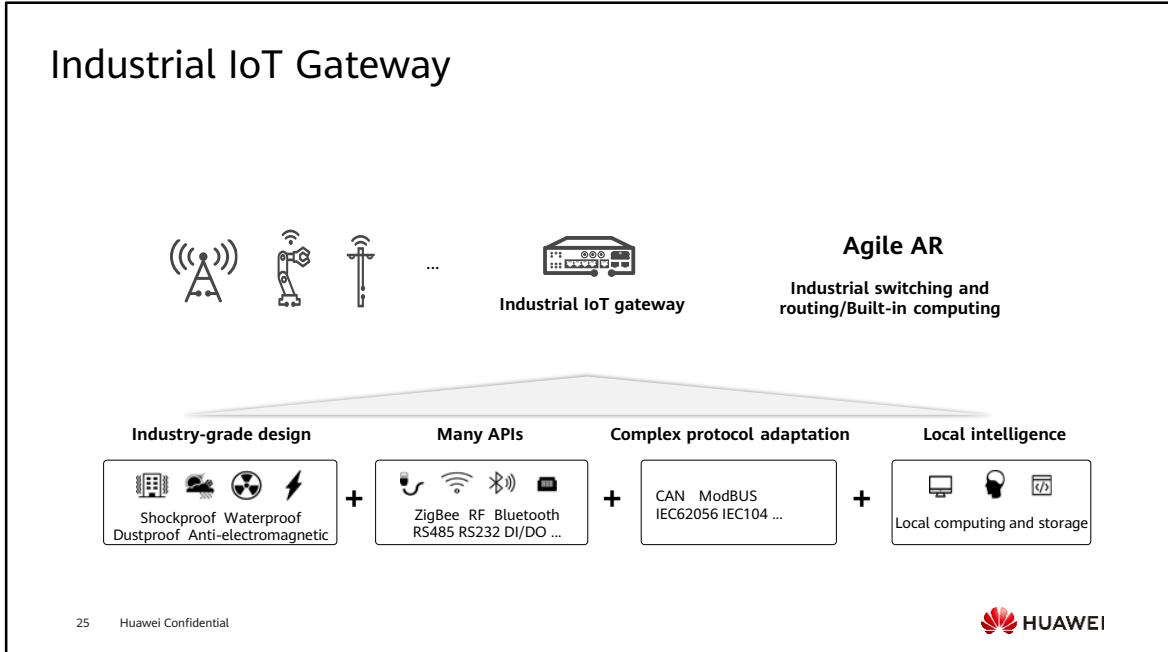


24 Huawei Confidential



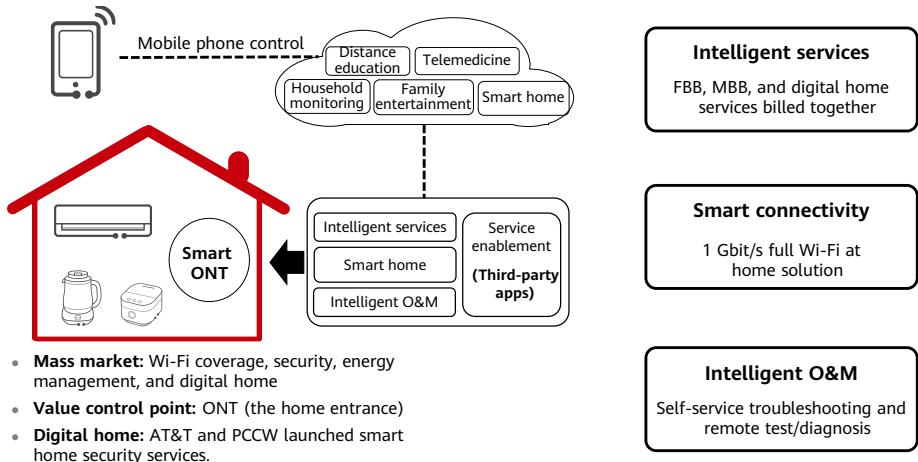
- Network function virtualization (NFV): Uses universal hardware (such as x86) and virtualization to process software with many functions, reducing network device costs. Using software and hardware decoupling and function abstraction, it enables network device functions to be independent of dedicated hardware. Resources can be flexibly shared, new services can be quickly developed and deployed, and automated deployment, auto scaling, fault isolation, and self-healing can be performed as required.
- Software-defined networking (SDN): An innovative network architecture proposed by the Clean Slate Program research team from Stanford University. It is an approach to implement network virtualization. Its core technology, OpenFlow, separates the control plane from the data plane of a network device, thereby network traffic can be flexibly controlled. This makes the network pipe more intelligent and provides a good platform for innovation of the core network and applications.
- Network slicing virtualizes a physical network into multiple end-to-end (E2E) networks, which provide different functions to meet different service requirements.
- Network slices have three key features: customization, end to end, and isolation.
 - A network slice is an end-to-end network, covering the RAN, transport network, and core network. It requires a cross-domain slice management system.
 - Network slices require isolation of resources, security, and OAM. Different domains can use different technologies to achieve such isolation. For example, the core network uses virtualization technology.
 - Network slices can be customized to provide specific network functions and features.

Industrial IoT Gateway



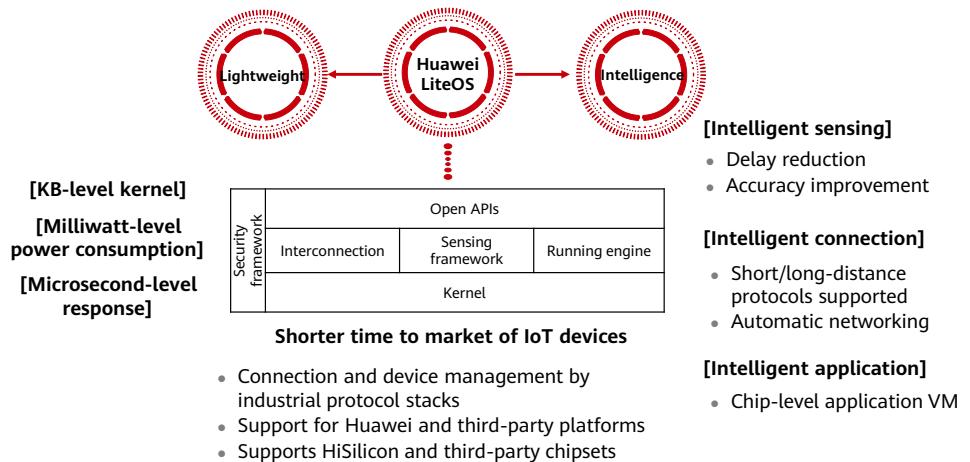
- The industrial IoT gateway supports multiple network standards and provides interfaces that comply with RS-485, PLC, and ZigBee. The gateway can directly communicate with devices on the sensor network, converge heterogeneous sensor networks that adopt different technologies, and remotely transmit sensor data using communications networks.
- The industrial IoT gateway functions just like a common Internet router and connects to the 3G mobile network and Internet in wireless or wired mode. The industrial IoT gateway can also interconnect with carriers' platforms to provide better services.
- The industrial IoT gateway supports Quality of Service (QoS) to ensure that each area or application corresponds to an independent queue, so applications can be scheduled in real time. It also provides dedicated bandwidth and preferential forwarding for key services to secure the service quality.
- The industrial IoT gateway provides security mechanisms to ensure authenticity and accuracy of IoT big data. In addition, the gateway devices must be industrialized, support a wide temperature range, and be dustproof, waterproof, and strong electromagnetic-proof to adapt to different scenarios.

Fully Open Smart ONT



- Carriers are in urgent need of a product that can function as an entry to access home networks to realize smart home. That is why smart ONT comes into being. Smart ONT has one core, two open advanced architectures, and a variety of innovative business models. Smart ONT provides smart full Wi-Fi and IoT connection extended by USB dongles to achieve ubiquitous connections. It extends flexible pipes from carriers' equipment rooms to users' homes, integrates applications, and builds a platform for smart home. Smart ONTs delivers security, home control, energy management, distance education, and telemedicine to help carriers expand the home market. The advantages of smart ONTs in smart connectivity, intelligent services, and intelligent O&M help carriers increase revenue and reduce expenditure.

Lightweight, Intelligent Huawei LiteOS



- Huawei LiteOS is a unified IoT OS with middleware. It is lightweight (basic kernel size less than 10 KB), low power consumption, interconnected, and secure.
- Currently, Huawei LiteOS is used on smart hardware in smart home, wearable devices, IoV, smart metering, and industrial Internet. It can also connect to LiteOS hardware to improve user experience.
- The LiteOS features low power consumption, small kernels, and quick response. In addition, the LiteOS has established an open source community to support chipsets such as the HiSilicon PLC chipset HCT3911, media chipset 3798M and 3798C, IP camera chipset Hi3516A, and LTE-M chipset.

Contents

1. IoT Development History
2. IoT Overview and Architecture
3. Huawei IoT Solution
- 4. IoT Engineering and Engineers**

Definition of IoT Engineering and IoT Engineers

IoT engineering

- Blends computer science and technology, information and communications engineering, microelectronics, detection and automation, and instruments science and technology. Specialties: control theory and control engineering, microelectronics detection, communications engineering, and computer and information correspond to IoT control, sensing, transmission, and information processing.

IoT engineers

- Research and develop IoT architectures, platforms, chips, sensors, and smart tags. They also design, test, maintain, and manage IoT engineering.

Career Paths

IoT embedded development

- Perceptual control, IoT application protocols, and IoT networking and communication

IoT application development

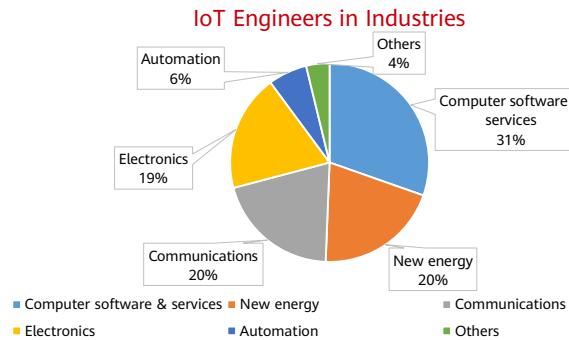
- IoT platform applications, IoT edge computing system applications, and IoT mobile applications

IoT system integration and management

- IoT device installation and commissioning, IoT system deployment, IoT system operation and maintenance, and IoT technical consulting and services

IoT Engineers in Industries

- IoT professionals are mainly working at computer software, new energy, electronics, and communications industries. Huawei is among the first to develop the IoT and has a large number of IoT engineers, covering the underlying technology and product R&D. More and more small- and medium-sized enterprises are also craving for IoT engineers to fulfill project planning and design, system implementation, and O&M.



Huawei IoT Certification

- Following the "Platform+Ecosystem" development strategy, Huawei Certification is a new collaborative architecture of ICT infrastructure based on "Cloud-Pipe-Device".
Huawei has set up a complete certification system consisting of ICT infrastructure certification and platform and service certification. It is the only certification system that covers all ICT technical fields in the industry.
- HCIA-IoT Training and Certification aims to train and certify engineers who are able to develop E2E IoT services based on the Huawei IoT solution architecture.
- Engineers certified by HCIA-IoT master basic IoT knowledge and the Huawei IoT solution architecture, and are able to develop E2E services and perform O&M based on this architecture.

- The IoT is widely used in smart home, smart wearables, smart city, and smart agriculture. Currently, commercial IoT is craving for professionals and the payment is handsome. Still, you need to earn your way through hard work.

IoT Engineers and Huawei IoT Certification

Major Tasks of IoT Engineers

- Research, IoT technologies, architectures, protocols, and standards
- Research, design, and development for the IoT chips and software and hardware system
- Project planning, integration, deployment, and implementation instructions for the IoT system
- Installation, commission, maintenance of the IoT system
- Monitoring and management of the IoT system
- Consultation and support

Huawei IoT Certification

- A History and Overview of IoT
- IoT Industry Applications and Solutions
- IoT Industry Today
- Data Collection Technologies
- MCU Basics
- IoT OS Overview
- IoT Communications Technologies
- IoT Communications Protocols
- AT Commands for IoT Communication Modules
- IoT Platform Overview
- IoT Device-Cloud Connection Development

- The main tasks of IoT engineers of the Ministry of Human Resources and Social Security can be matched with the course content of IoT certification.

Quiz

1. (True or false) IoT is the Internet where things are interconnected. IoT devices can be connected only through wireless networks.
2. (Multiple) What are the layers of the IoT architecture?
 - A. Sensing layer
 - B. Network layer
 - C. Platform layer
 - D. Application layer

- Answer:
 - F
 - ABCD

Summary

- Upon completion of this course, you have learned the origin and development of IoT, what is IoT, and its layered architecture.
- You should understand the work content of IoT engineering. IoT certification helps you step on right career path.

Acronyms or Abbreviations

- AR: Augmented Reality
- IoT: Internet of Things
- ITU: International Telecommunication Union
- NB-IoT: Narrowband Internet of Things
- RFID: Radio Frequency Identification
- VR: Virtual Reality

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and
organization for a fully connected,
intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive
statements including, without limitation, statements regarding
the future financial and operating results, future product
portfolio, new technology, etc. There are a number of factors
that could cause actual results and developments to differ
materially from those expressed or implied in the predictive
statements. Therefore, such information is provided for reference
purpose only and constitutes neither an offer nor an acceptance.
Huawei may change the information at any time without notice.



IoT Industry Applications and Solutions



Foreword

- With the development of the Internet of Things (IoT), its impact has penetrated into every aspect of society. IoT is everywhere, from smart homes to smart cities.
- Here, we analyze existing problems in four IoT industry scenarios: smart city, all-in-one smart home, Internet of Vehicles (IoV), and industrial IoT. We also present solutions provided by IoT and success stories.

Objectives

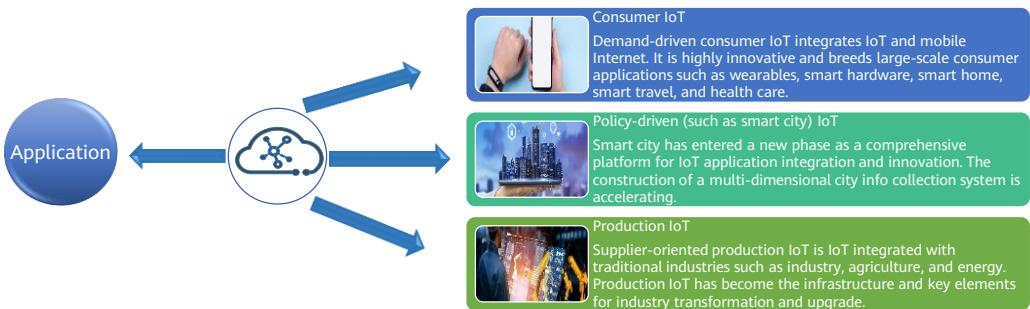
- Upon completion of this course, you will understand:
 - IoT application types
 - Common problems in smart cities and the corresponding solutions
 - Trends of the all-in-one smart home industry and the corresponding solutions
 - Driving forces of IoV development and the corresponding solutions
 - Requirements and challenges of industrial IoT and the corresponding solutions

Contents

- 1. IoT Application Types**
2. Smart City Solution
3. All-in-One Smart Home Solution
4. IoV Solution
5. Industrial IoT Solution

The Three Main Types of IoT Applications

- Consumer IoT, policy-driven (such as smart city) IoT, and production IoT. Together, policy-driven IoT and production IoT are called industry IoT.

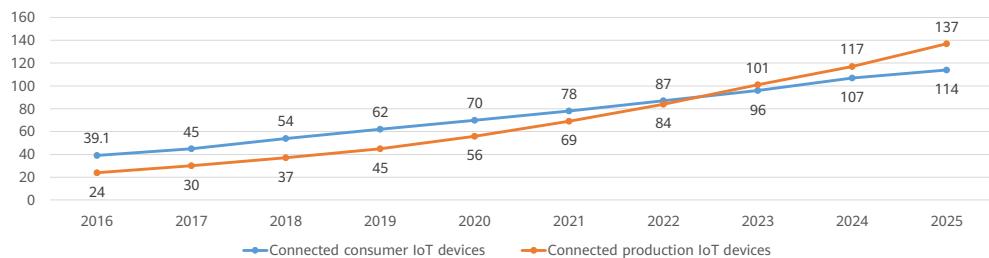


- Image 1: <http://3ms.huawei.com/km/static/image/detail.html?fid=58990>
- Image 2: <http://3ms.huawei.com/km/static/image/detail.html?fid=61742>
- Image 3: <http://3ms.huawei.com/km/static/image/detail.html?fid=61726>

Global Perspective: Surging Production IoT Connections

- Production IoT and consumer IoT are developing at the same time. Both parties want to solve and optimize development issues of the industry, energy, and transportation sectors, as well as enterprises.

Global connected devices in the consumer IoT and production IoT from 2016 to 2025 (unit: 100 million)



- The adoption of IoT in more industries brings more device connections. Production IoT is becoming the promising player instead of consumer IoT. GSMA Intelligence, one of the three top international organizations in mobile communications, predicts that the number of production IoT connections will increase by 4.7 times from 2017 to 2025, while that of consumer IoT connections will increase by 2.5 times.

Consumer IoT

- Internet enterprises optimize this ecosystem and innovation for consumer products, driving the expansion of the industry.

Main applications of consumer IoT

AI speaker

The global market is experiencing explosive growth. In 2021, the annual shipment of AI speakers reached 190 million units.

Sharing economy (bicycle and power bank)

China Sharing Economy Development Report (2022) released by the State Information Center showed that China's sharing economy transactions reached around CNY3688.1 billion in 2021, up about 9.2% from a year earlier. Direct financing was about CNY213.7 billion, up about 80.3% year-on-year.

All-in-one smart home

In 2018, the total expenditure on smart home devices, systems, and services worldwide was close to USD96 billion. Expenditure is expected to reach USD15 billion in 2023.

Wearable

According to IDC, the global demand for smart wearables will grow strongly from 2020 to 2025. Its compound annual growth rate (CAGR) is predicted to reach 25%. The estimated shipment will reach 13.58 billion units in 2025.

Smart lock

According to All View Cloud's online data, 4.58 million sets of smart locks were sold in China in 2021. The industry is developing well.

More possibilities

There are more possibilities for IoT application in the consumer field.



Policy-Driven IoT Example: Smart Cities

- Led by the concept of "digital twin cities", urban IoT will have a bigger scale, wider applications, and higher integration. However, city pain points, requirements, and deployment costs affect the development of smart city IoT.

No.	Industry	IoT Application
1	Safety	With policies such as safety and Sharp Eyes projects, the security industry is witnessing rapid growth. In 2017, the security market scale in China exceeded CNY630 billion, and the number of security manufacturers exceeded 7000. Large expansion of the security industry is a favorable environment for IoT. IoT integrates with smart security more deeply, and the number of connected smart security devices is surging. "AI + security" has become a typical feature of the IoT application in the security industry.
2	Public utilities	Public utilities use low power wide area networks (LPWANs) for intelligent upgrade. Intelligent upgrade of public utilities such as urban water supply, power supply, and heating is the most typical livelihood project of smart city in recent two years. More suitable LPWANs such as NB-IoT and LoRa are introduced into public utilities.
3	Firefighting	A survey on major firefighting enterprises in the past two years showed that overall revenue growth rate remained between 20% and 40%. Only a few managed to reach 100%. According to a conservative estimate that the CAGR of China's smart firefighting industry will reach 20% in the next five years, the market scale of China's smart firefighting industry will reach about CNY100 billion in 2026.
4	Electric bicycle	<i>China Sharing Economy Development Report (2022)</i> released by the State Information Center showed that China's sharing economy transactions reached around CNY3688.1 billion in 2021, up about 9.2% from a year earlier. Direct financing was about CNY213.7 billion, up about 80.3% year-on-year. The successful pilot of the electric bicycle management solution will witness large-scale connections of electric bicycles in smart cities.

Production IoT Applications Are New Opportunities

1

Urgent market requirements: Traditional industries need IoT to address industry pain points, expand market size, and promote transformation and upgrade.

2

Higher technology level: Dedicated IoT networks meet the requirements of wide coverage and low power consumption in agriculture. New technologies bring more opportunities.

3

Active participation of related industries: Telecom carriers, equipment vendors, and Internet enterprises drive IoT adoption in traditional industries.

Main Applications of Production IoT

No.	Domain	IoT Application
1	Industrial IoT	According to MarketsandMarkets, infrastructure and industrial development in emerging economies such as China and India have promoted the growth of industrial IoT in Asia Pacific. The development mode of the industrial IoT has four application modes: intelligent production, network-based collaboration, personalized customization, and service-oriented transformation.
2	Agricultural IoT	According to the <i>14th Five-Year Plan for National Informatization</i> , smart agriculture has evolved, the informatization rate of agricultural production has reached 27%, and the annual online retail sales of agricultural products have exceeded CNY800 billion. 100 national digital agricultural innovation bases have been built, and 200 agricultural and rural informatization demonstration sites have been certified. The agricultural and rural big data system has launched, and a national agricultural and rural big data platform built, forming a map of agricultural and rural data resources. The innovation system of agricultural and rural informatization is further improved. The independent innovation capability continues to grow. New breakthroughs have been made in key core technologies and products. More than 60 national digital agricultural and rural innovation centers, branches, and key labs have been built.

Contents

1. IoT Application Types
- 2. Smart City Solution**
3. All-in-One Smart Home Solution
4. IoV Solution
5. Industrial IoT Solution

Overview and Objectives

- For IoT to become concentrated enough for the next leapfrog, at least three conditions must be met: dense population, solid industrialization foundation, and national integrated market. Only China meets these three conditions. IBM proposed the smarter planet in November 2008 and released its smarter planet in China plan in August 2009, officially unveiling IBM's smarter planet strategy in China.
- As a pilot project under smarter planet, smart city covers a large number of scenarios. It faces many difficulties and challenges. This section describes the difficulties encountered by smart city in different scenarios and their corresponding solutions.

Common Problems - Traffic and Parking Management

Frequent traffic congestion and accidents		Increased travel time	Intensified pollution	
Parking management	Imbalance	Inconvenient charging	Difficult inspection	Increasing congestion
Parking experience	Difficulty finding parking spaces	Difficulty finding vehicles	Difficulty entering and leaving parking lots	Outdated payment systems
	<ul style="list-style-type: none"> Lack of resource integration and sharing Severe tidal effects 	<ul style="list-style-type: none"> Low labor efficiency and high costs Frequent payment evasion 	<ul style="list-style-type: none"> Low efficiency Difficulty confirming paid fees 	<ul style="list-style-type: none"> Time wasted locating parking spaces Lack of guidance and reservation services
	<ul style="list-style-type: none"> Difficulty locating empty parking spaces Lack of parking guidance facilities 	<ul style="list-style-type: none"> Difficulty locating parked vehicles Difficulty navigating complex environments 	<ul style="list-style-type: none"> Inconvenient parking, card collection, and payment Congestion at parking lot entrances and exits during peak hours 	<ul style="list-style-type: none"> Congestion due to manual charging Lack of charging modes

13 Huawei Confidential



- Parking difficulties
 - In China, for example, parking difficulties are common in first and second-tier cities. The average parking time for each vehicle is 18 minutes. Parking difficulties lead to parking violations and congestion. According to one survey, about 30% of traffic congestion is the result of drivers continuously looking for parking spaces. This increases emissions, causing more pollution.
- Low utilization
 - The vacancy rate of parking spaces in Beijing, Shanghai, Guangzhou, and Shenzhen parking lots is 44.6%. Vacancy rates are especially high in underground parking lots.
- Information silos of parking lots
 - Real-time information cannot be exchanged between parking lots and drivers, leading to low parking space usage. Tidal traffic is switched between residential areas and office areas. Parking space usage in residential areas during daytime is lower than 30%, and almost all parking spaces in office areas are available at night.

Common Problems - Street Lamp Management

Reliable lighting

Provides reliable lighting for urban roads, which is the core responsibility of the Street Lamp Administration.

Asset management

Protects street lamps from being damaged or stolen.

Energy conservation

Turns off lights on time after daybreak. Reduces illumination in the middle of the night. Lowers brightness when there are no pedestrians or vehicles.

Emergency lighting

Starts the emergency lighting in bad weather or special weather during daytime.

Simplified O&M

Promptly detects and repairs faulty street lamps. Considers the employment impact when applying advanced technologies.

Revenue growth

Rents lamp poles to advertisement companies and tower companies for profit. (It is difficult to obtain commercial benefits with current systems.)



Common Problems - Firefighting Management

- Nine small public areas are vulnerable in urban fire safety.
 - Fire risks: chaotic environment, group rentals, random stacking of flammable things, and electricity piracy.
 - Weak firefighting facilities: no firefighting facilities or outdated firefighting equipment.
 - Delayed fire warning: delayed fire detection and insufficient fire information due to the time-consuming manual inspection.



Firefighting facility



Flammable material stacking



Old cables

- The nine small public areas refer to small schools and kindergartens, hospitals, shops, restaurants, hotels, dance halls, Internet cafes, beauty salons and bathhouses, and manufacturing enterprises.

Common Problems - Manhole Cover Management

- Manhole covers are embedded in city streets like screws on giant machines. The manhole covers belong to administrative departments for water, communications, gas, heat, power, and traffic management. Problems of manhole cover management are as follows:
 - Difficult management due to large quantity.
 - Disordered identity management due to complex ownership.
 - Theft, loss, and shifting.
 - Secondary injuries due to security risks.



 HUAWEI

- Manhole cover management affects the safety of roads, people, and underground public equipment. Manhole covers are distributed widely in numbers, which makes management difficult.

Common Problems - Environmental Sanitation Management



Outdated facilities, high O&M costs, low work efficiency, and poor work quality.



Outdated management models, limited management methods, lack of basis for decision-making, and serious resource waste.



Inconsistent operating standards, random operating status, serious interference caused by human factors, and difficult command and dispatch.



Lack of innovation, slow overall development, slow application of new modes, new devices, and new concepts, low informatization levels, and high management cost.

Summary of City Management Problems

Unclear facility information

The informatization rate of manhole covers, street lamps, garbage cans, garbage stations, trees, pipelines, dangerous sources, and bridges is low. Manual inspection is heavily relied on.

Untimely issue identification

Issues such as road occupation, facility damage, and garbage overflow cannot be detected on time. As a result, there is little interaction between citizens, and public satisfaction is low.

Difficult collaboration across departments

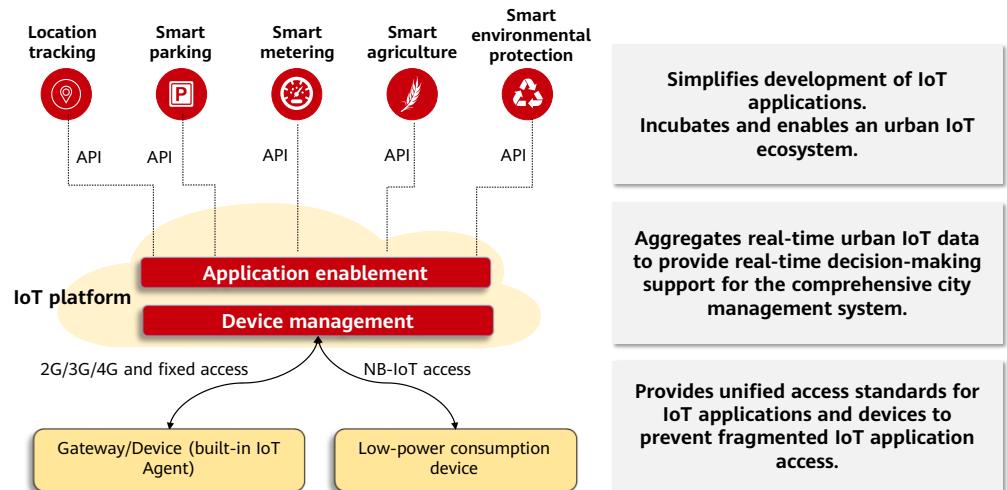
Incidents such as random unloading of slag trucks and water pipe bursts involve multiple departments, such as those for sanitation, law enforcement, gardening, city appearance, housing and construction, environmental protection, public security, transportation, and civil affairs. These departments are difficult to coordinate.

Difficult decision-making across isolated systems

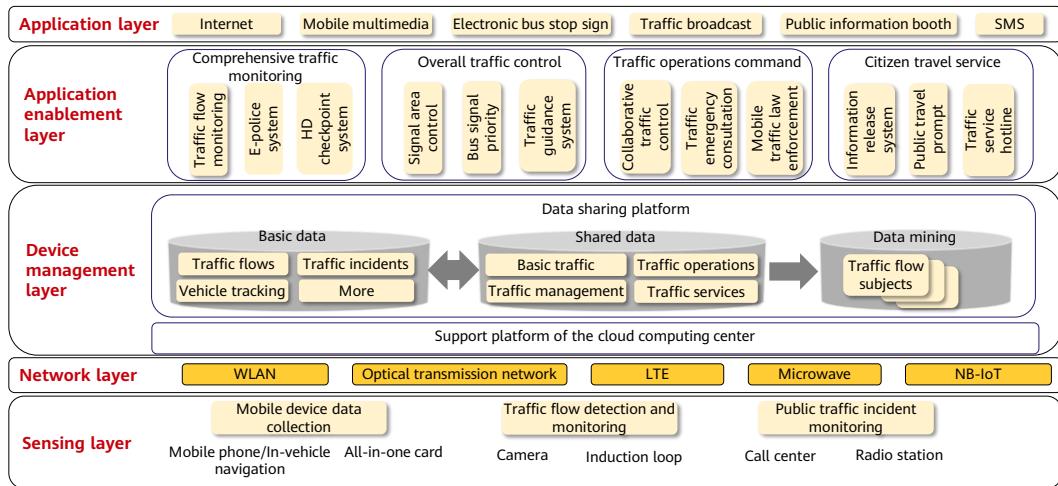
Repeated platform construction, isolated systems, difficult data aggregation, and lack of a unified data analysis and scientific decision-making systems

How do we use innovative technologies to achieve smart city management?

Smart City Solution



Smart Transportation Solution (1)



Smart Transportation Solution (2)

Relieve traffic congestion

- Constructs application systems such as e-police, speed detection, signal control, and guidance systems to maximize traffic guidance, reduce traffic accidents, and reduce accidents and property loss.
- Improves road patrol methods and patrol efficiency by displaying road conditions in multiple modes to implement electronic and automatic patrol.
- Uses mobile law enforcement to efficiently and quickly process traffic violation information, vehicle information, and driver information.

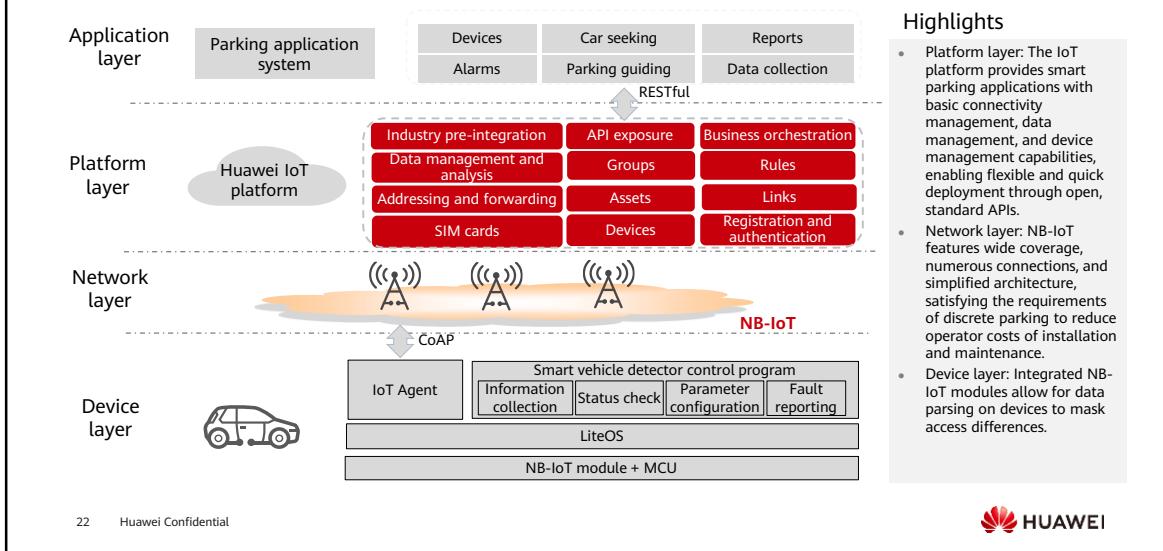
Facilitate citizen travels

- Collects details about rush hour commutes to optimize travel routes and reduce travel times.
- Citizens can obtain real-time traffic details and plan travel routes accordingly.
- Reduced commute times improve citizen satisfaction.

Improve environmental protection

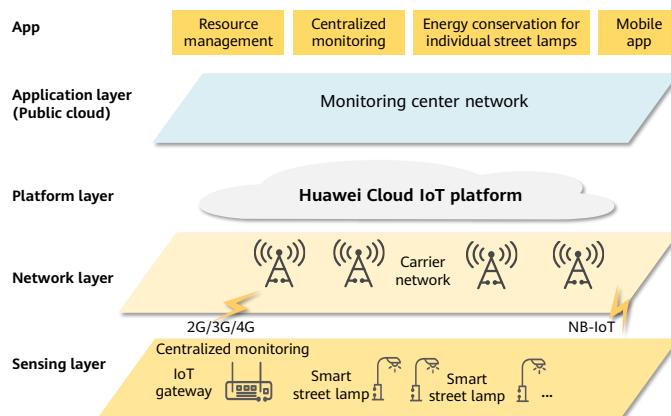
- Smooth traffic improves vehicle speeds and reduces emissions.
- Optimizes public transportation to encourage citizens to choose public transport and reduce exhaust emissions.
- Improves urban environments so that citizens choose eco-friendly travel modes to further reduce emissions.

Smart Parking solution



- The NB-IoT solution provides ubiquitous, easy-to-maintain, and smart connections for parking, including roadside parking spaces and toll parking lots.

Smart Street Lamp Solution



Customer benefits

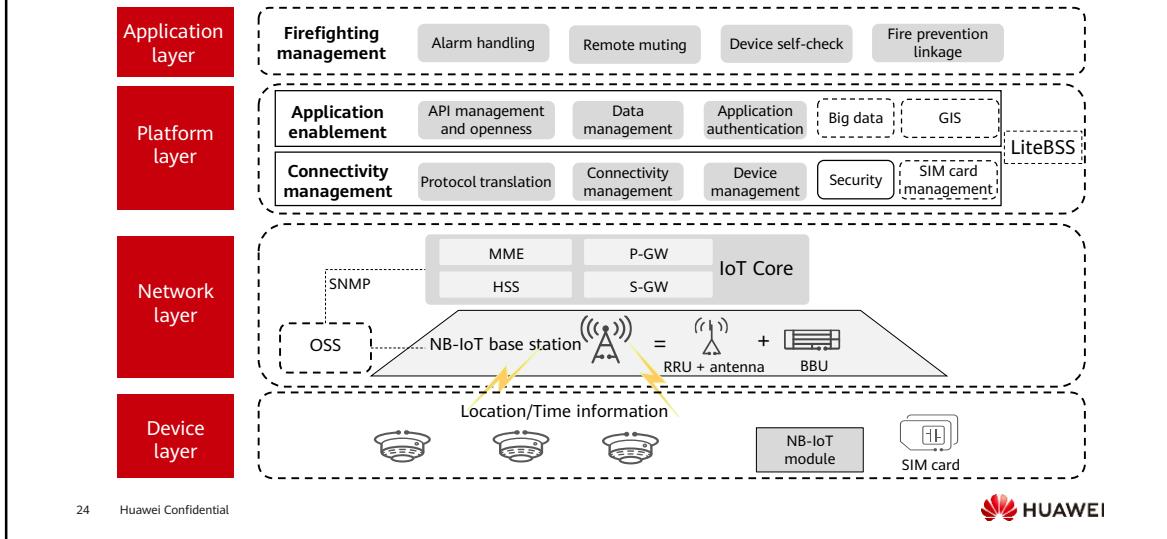
- Construction of shared collection devices in cities:** Unified installation of functional facilities in multiple cities, and unified data aggregation and backhaul reduce the construction costs of city infrastructure.
- Platform-based and unified O&M:** Unified O&M and monitoring of sensors mounted in each bureau improves O&M efficiency and reduces costs.
- On-demand lighting to reduce consumption and save energy:** Lighting duration is adjusted dynamically, and brightness is adjusted based on the time segment. An energy conservation plan is made based on comprehensive analysis of overall lighting power consumption.

Scenarios

- Periodic tasks:** enable or disable lamps and adjust brightness during different time segments.
- Intelligent light adjustment:** automatically detects passing vehicles and adjusts the brightness of lamps.
- Automatic O&M:** automatically reports faults to the service system if a street lamp is faulty.



Smart Firefighting Solution

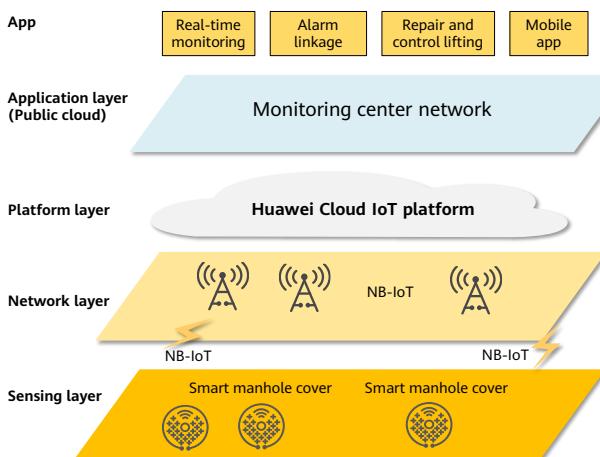


24 Huawei Confidential



- LiteBSS: Lite Business Support System, a lightweight IoT charging system
- The market space of independent smoke sensors is large. Zhejiang, Guangdong, and Jiangsu require more than 10 million sensors.
- Traditional smoke sensors are difficult to construct and manage.
 - Difficult cabling: Traditional networked smoke sensors are mature and reliable. However, their network and power cables need to be re-routed. Reconstruction costs are high for most already-constructed and furnished buildings. Reconstruction maybe unfeasible.
 - Difficult management: Independent smoke sensors are cost-effective and easy to deploy, but cannot be connected to the network for alarm reporting and management.

Smart Manhole Cover Solution



Customer benefits

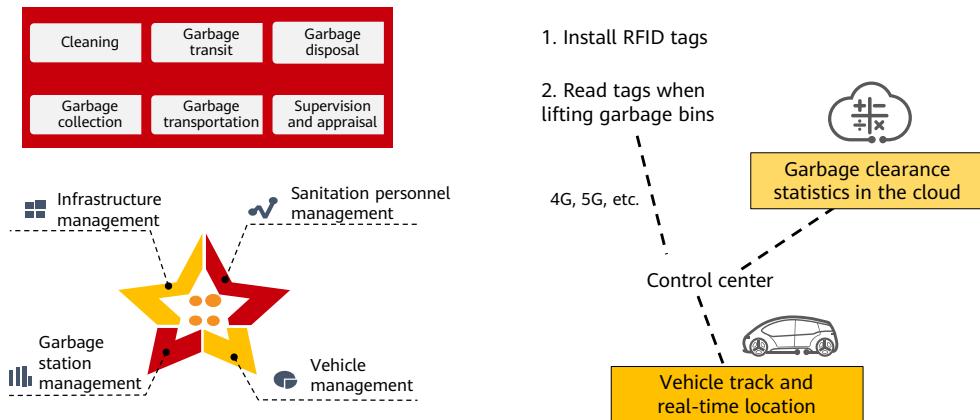
- Construction of shared collection devices in cities:** Unified installation of functional facilities in multiple cities, and unified data aggregation and backhaul reduce the construction costs of city infrastructure.
- Platform-based and unified O&M:** Unified O&M and monitoring of sensors mounted in each bureau improves O&M efficiency and reduces costs.
- Alarm reporting and anti-theft:** By monitoring manhole covers in real time, the system can detect incidents (theft, displacement, and damage), generate alarms, as well as notify construction organizations or policing platforms to take immediate action, eliminating security risks and ensuring city security.

Scenarios

- Real-time monitoring:** Manhole covers in a large area are monitored in real time and intelligently maintained.
- Alarm reporting:** Monitoring manhole covers that are abnormally open helps identify incidents such as theft, displacement, and damage. Alarm reports send the location to the monitoring center and policing platform. Then the monitoring center schedules construction vehicles to maintain these manhole covers and the policing platform dispatches officers to the incident location.



Smart Sanitation Solution



Manages people, vehicles, objects, and events in real time.

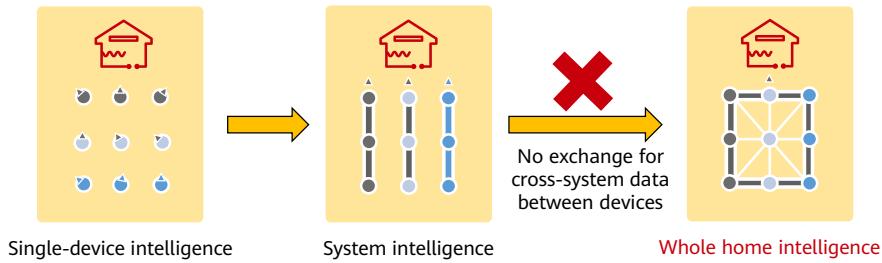
- Manages the entire process of sanitation personnel, vehicles, and garbage cleaning in real time.
- Supervises the operating time, routes, fuel consumption, and water-spraying of sanitation vehicles.
- Monitors garbage bin status, garbage collection routes, and garbage transit stations in real time.
- The solution is based on real-time, visualized video surveillance. It enables refined management, standardizes operations, and intelligently appraises urban sanitation, making sanitation orderly, effective, and controllable.

Contents

1. IoT Application Types
2. Smart City Solution
- 3. All-in-One Smart Home Solution**
4. IoV Solution
5. Industrial IoT Solution

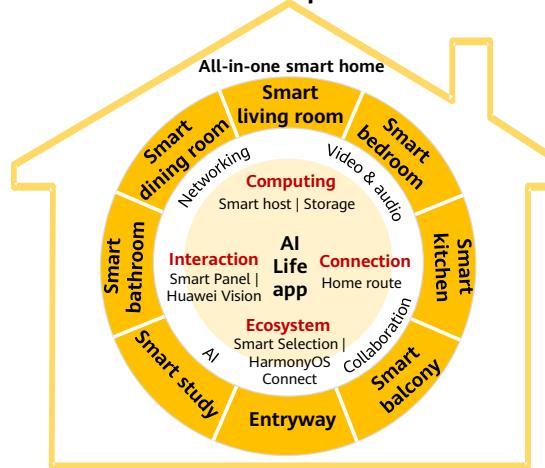
Common Issue: Different Connection Standards Bring Fragmented Interaction Experience

Home devices: not so smart due to no exchange for cross-system data between devices



- Smart home industry: Different connection standards bring fragmented interaction experience.

Huawei All-in-One Smart Home Solution: Building All-scenario Smart Spaces

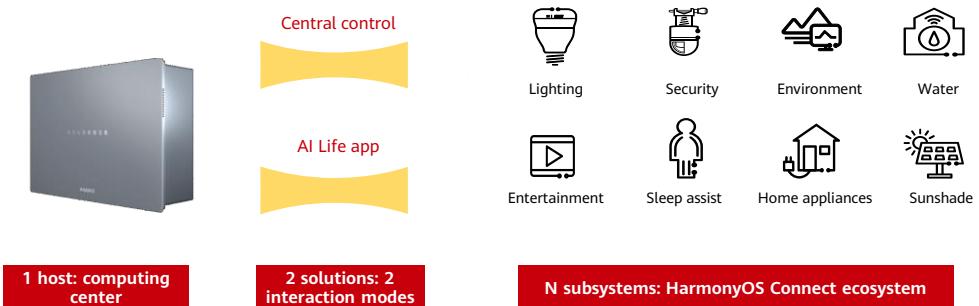


Huawei All-in-One Smart Home Solution

- **Portal:** AI Life app
- **Core products:** interaction, computing, connection, ecosystem
- **Basic capabilities:** whole-house networking, collaboration, video & audio, and AI
- **Smart spaces:** living room, bedroom, kitchen, balcony, entryway, study, bathroom, dining room, etc.

- Bring digital to every home/hotel.

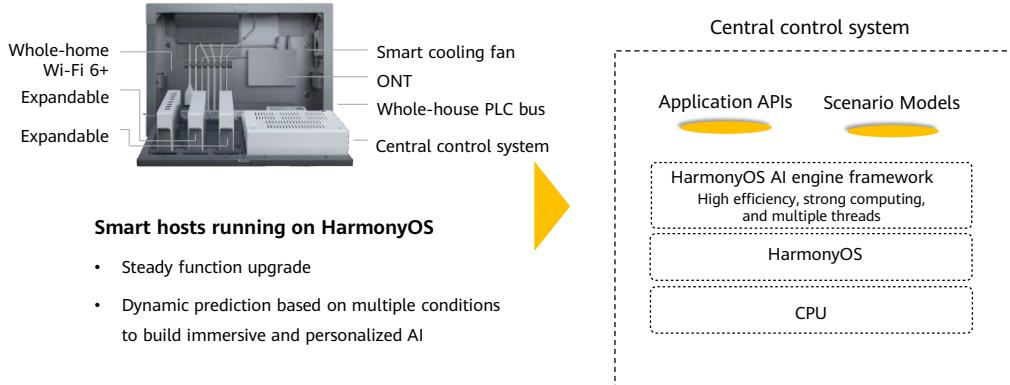
Huawei All-in-One Smart Home "1+2+N" Solution



**Core: Harmony-powered smart host + two interaction modes + diverse
HarmonyOS Connect ecosystem products**

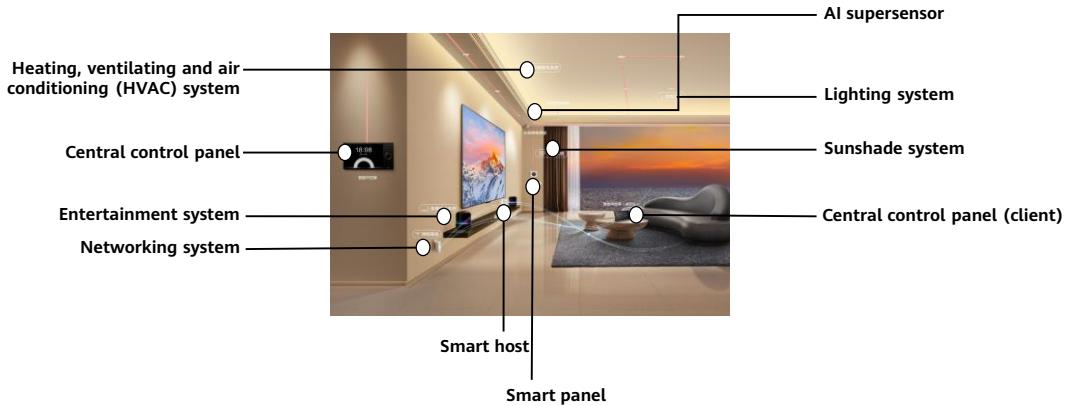
- <https://consumer.huawei.com/cn/wholehome/wholehome-solution/>

Smart Hosts for All-in-One Smart Home



- Powered by HarmonyOS, the smart home host is the center that integrates learning, computing, and decision-making.
- <https://consumer.huawei.com/cn/wholehome/wholehome-solution/>

Huawei All-in-One Smart Home



- <https://consumer.huawei.com/cn/wholehome/wholehome-solution/>

Contents

1. IoT Application Types
2. Smart City Solution
3. All-in-One Smart Home Solution
- 4. IoV Solution**
5. Industrial IoT Solution

Common Challenges in the Automotive Industry



- Delayed detection of vehicle faults
- Impact of faulty vehicles on safety of other vehicles
- Influence of weather changes on driving



- Fixed insurance rates
- Private use of company cars
- Fleet management
- ETC, parking fee recharge, etc.



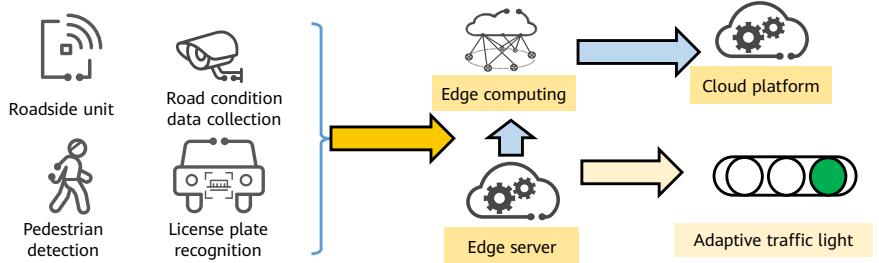
- Traffic citation for speeding
- Traffic block due to road repair
- No smooth music playing
- No smart control on traffic lights

What Is IoV?

- Internet of Vehicles (IoV) means that **vehicle-mounted devices** use **wireless communication technologies** to make full use of all dynamic vehicle information on the **information network platform** and **provide various functions and services** during vehicle running.
- IoV has the **following characteristics**: Provides assurance for a distance between vehicles to reduce a probability of vehicle collision accidents. Provides real-time navigation for drivers and communicates with other drivers and network systems to improve traffic efficiency.

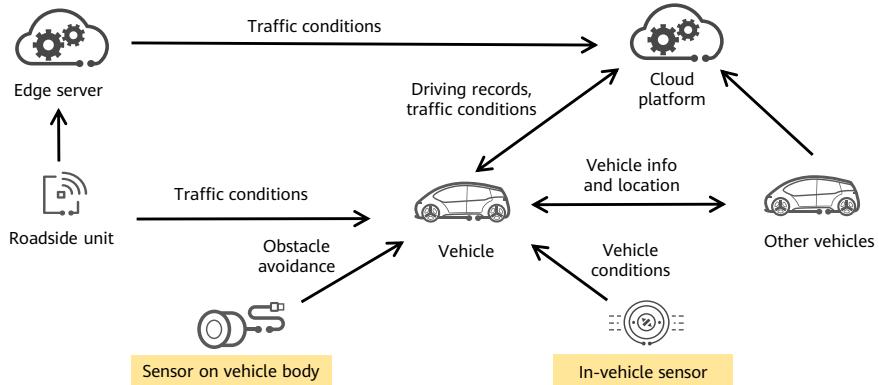
Building Smart Intersections Based on the IoV Solution

- Traffic conditions change at all hours of the day, and traffic lights are key to controlling traffic.
- Traditional solution: Traffic lights turn on and off in a specified sequence, which cannot change according to road conditions.
- Smart intersections based on IoV: Road condition data is collected to the edge to adjust the traffic lights. The cloud platform stores such data for big data analytics, providing data basis for future urban planning.



Vehicle to X (V2X)

- V2X is closely related to smart transportation and intelligent roads.



- Vehicle-To-Network (V2N) is one of the most widely used connections, which allows vehicles to connect to cloud servers through a mobile network, thereby implementing application functions such as navigation, entertainment, and anti-theft. A common application scenario is that a vehicle-mounted system connects to the Internet through a mobile phone hotspot to obtain real-time navigation information or play online music. Currently, many vehicle-mounted systems provide card ports, which can be directly inserted with data cards for the connection to the Internet.
- Vehicle-To-Vehicle (V2V) aims to implement collision warning and congestion avoidance functions through information exchange between vehicles. For example, before entering the corner of a winding road, a veteran driver would honk to warn the vehicles in the blind zone. With the V2V technology, drivers can communicate with each other without using speakers. In addition, information such as speed and direction can also be transmitted among them.
- Vehicle-To-Infrastructure (V2I) enables vehicles to exchange data with roads and roadside infrastructure, such as traffic lights, road signs, and even roadblocks. In addition, vehicles can collect information about the surrounding environment. This kind of connection seems to be not helpful to drivers, but plays an obvious role in the automated driving field because the vehicle can find an obstacle in the front by using this connection.
- V2P (Vehicle-To-Pedestrian) is similar to V2V and aims to prevent collisions through information exchange. In addition to cameras and sensors, information interconnection is also the most effective way to detect pedestrians. For example, a terminal used by a pedestrian, such as a mobile phone, a tablet, or a wearable device, can implement interconnection between a person and a vehicle. This helps to prevent staged accident fraud, or "pengci".
- As one of the IoV solutions, Vehicle to Everything (V2X) integrates digital technologies such as cloud, IoT, AI, and 5G to implement efficient collaboration among people, vehicles, roads, and clouds, build smart vehicles and roads, and

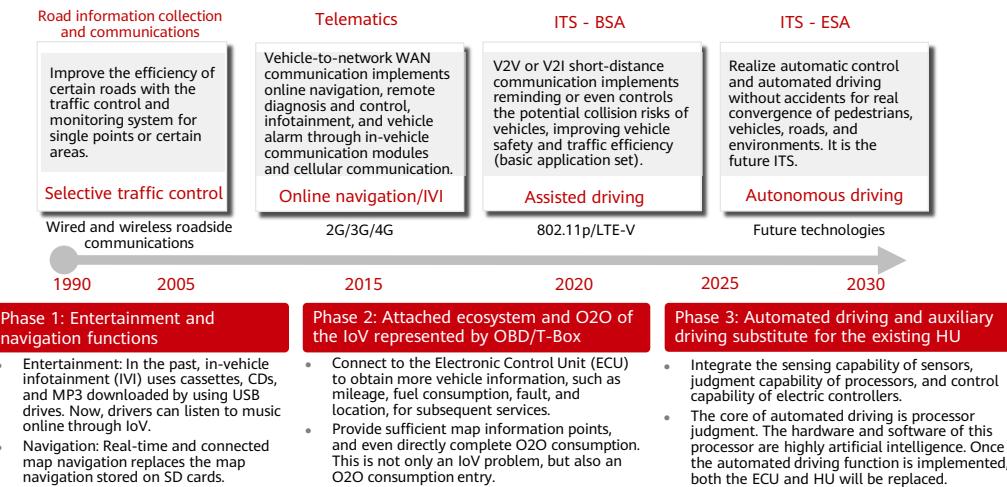
provide intelligent mobility services and experience for users.

Digital Road Infrastructure Service (DRIS)

- DRIS implements digital information exchange among pedestrians, vehicles, roads, and networks, improves driving safety and traffic efficiency, and facilitates large-scale commercial use of autonomous driving.
- DRIS consists of **V2X Server in the cloud** and **V2X Edge at the edge**.
 - V2X Server provides digital road infrastructure services and edge-cloud synergy services such as data analysis and roadside computing unit management.
 - V2X Edge provides real-time service processing such as roadside sensor data collection, event identification, and communications.
- The goal of DRIS is to connect multiple roadside sensors for digital perception and provide information for efficient traffic.

- DRIS provides the following functions:
 - Establishes traffic events and delivers them to RSUs, which then forward the events to vehicle-mounted devices. The main scenarios are vehicle signs and bad weather events.
 - Real-time traffic monitoring in cities and areas covers key indicators such as the number of traffic accidents, number of connected vehicles, congested roads, vehicle model distribution, traffic flow, ranking of accident-prone locations, and number of online devices.
 - Monitors road side units (RSUs) and other devices in real time, helping enterprises learn about device status at any time and rectify device faults in a timely manner.
 - Provides the edge-cloud synergy capability to deploy and update the software of cloud V2X.
 - Provides data storage and data openness interfaces.
- Edge DRIS identifies road traffic events through cameras and radar devices and delivers the events to the RSU through the cloud DRIS. The RSU forwards the events to vehicle-mounted units. The main scenarios are abnormal vehicle and road conditions.

IoV Development History

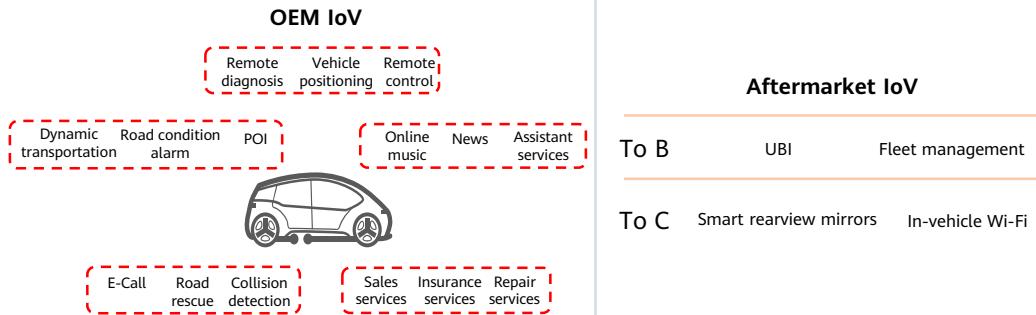


39 Huawei Confidential



Today's IoV

- IoV is evolving from IVI services to smart transportation. The OEM mode focuses on internal services of OEMs, vehicle data collection, and personal entertainment information service. The aftermarket mode focuses on industry applications, supplemented by personal IVI information service.

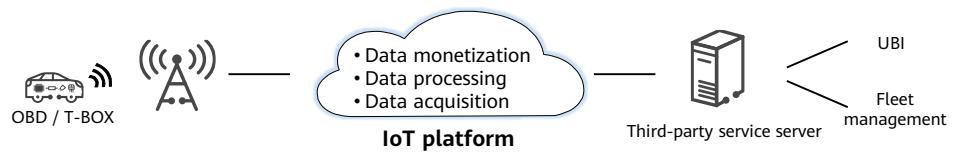


Heads Up Display (HUD)

- Driving safety is the motivation for using HUD. With HUD, drivers do not need to shift the attention to the dashboard or touch the head unit during driving. The information displayed by HUD is the driving condition indicators of the vehicle, for example, speed and fuel volume.
- HUD also makes vehicles more intelligent by providing functions including navigation, SMS, phone, email, and simple interactions.

- Three HUD implementations:
 - Use the front windshield as the display screen, which is a common practice in OEM mode, for example, HUD provided by automakers such as BBA.
 - Use an additional display screen, which is a common practice in aftermarket mode, for example, HUD provided by Nandy, the star product for this solution.
 - Use the reflection of the screen on the front windshield (or protective film), for example, AutoNavi's HUD. This implementation delivers the worst performance.

OBD/T-BOX



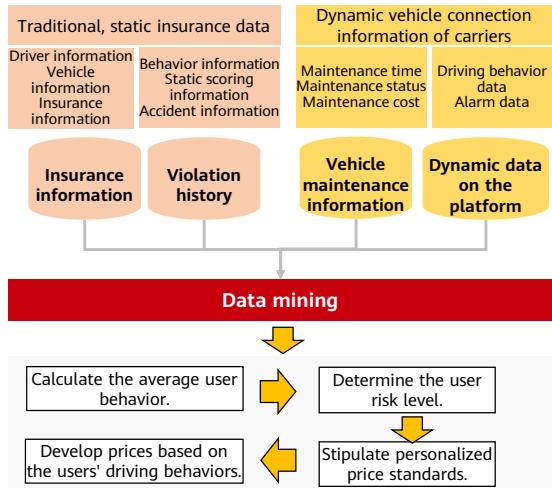
42 Huawei Confidential



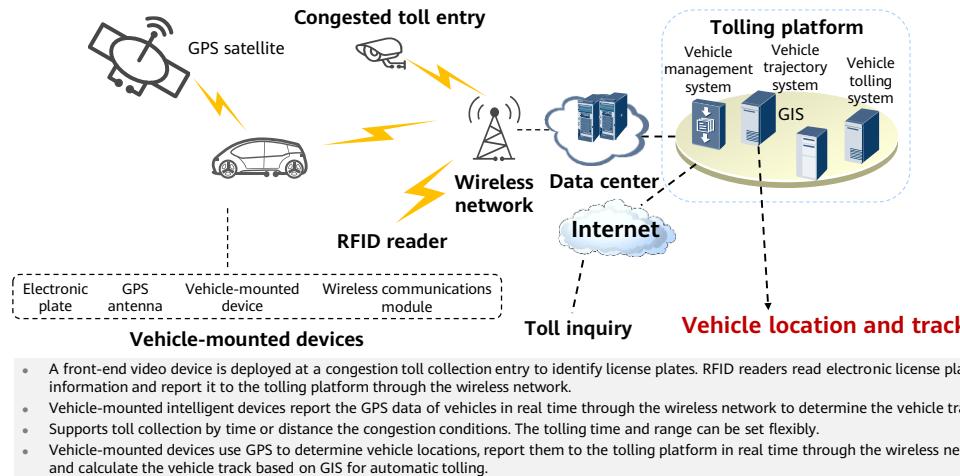
- OBD: On-Board Diagnostics

UBI

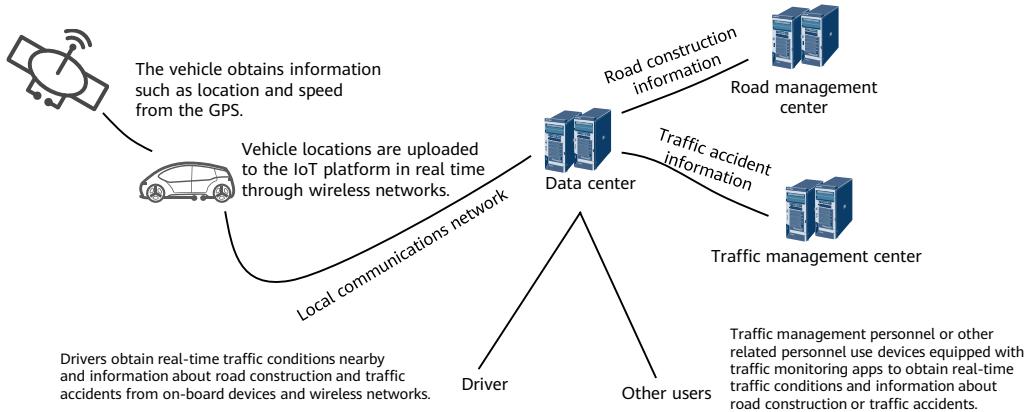
Customer Category	Business Strategy
High-risk customer	<ul style="list-style-type: none"> Do not provide insurance services.
Customers with poor behavior	<ul style="list-style-type: none"> Provide daily driving behavior scoring reminders and driving behavior suggestions. Regularly communicate with the customer based on the customers' driving behavior through the call center. Encourage customers to improve their driving behavior through discounts.
Customers with neutral behavior	<ul style="list-style-type: none"> Provide daily driving behavior scoring reminders and driving behavior suggestions. Encourage customers to improve their driving behavior through discounts.
High-value customers	<ul style="list-style-type: none"> Provide more additional services to retain customers. Perform regular surveys through the call center to increase customer loyalty.



Automatic Tolling and Vehicle Trajectory Data



Smart Transportation



Accurate, real-time traffic data relieves traffic congestion.

Contents

1. IoT Application Types
2. Smart City Solution
3. All-in-One Smart Home Solution
4. IoV Solution
- 5. Industrial IoT Solution**

Today: The Momentum of a New Industrial Revolution

- After mechanization, electrification, and automation, we have ushered in the fourth industrial revolution represented by intelligence. Intelligence is embedded into every connected things and all business processes.
- ICT technologies, such as big data analytics, cloud computing, mobility, and IoT, lay the foundation of the fourth industrial revolution.



Crossing the Chasm and Developing Smart Manufacturing with Efficiency and Refinement

Focus only on the automation of production and manufacturing.



Industry chain Information silos



No visibility into production data

Chasm: more efficient and refined smart manufacturing

Fully connect people, data, and machines, and combine big data analytics to develop towards more efficient and refined manufacturing.



Vertical integration of production information

- Dynamic sensing and intelligent O&M
- Dynamic production control
- 3D virtual factory



Horizontal integration of industry chain

- Crowdsourcing and collaborative R&D
- E-commerce for marketing procurement
- IoV

Production IoT

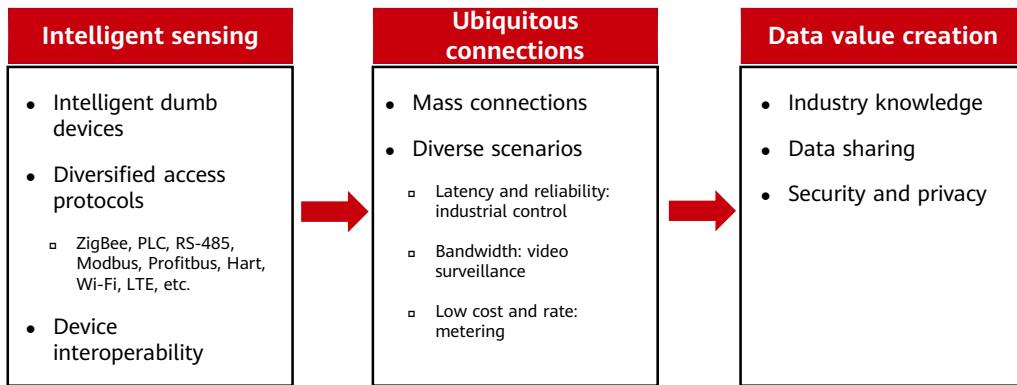
Industrial cloud and intelligent big data analytics

Mobility, real-time grasp of production information

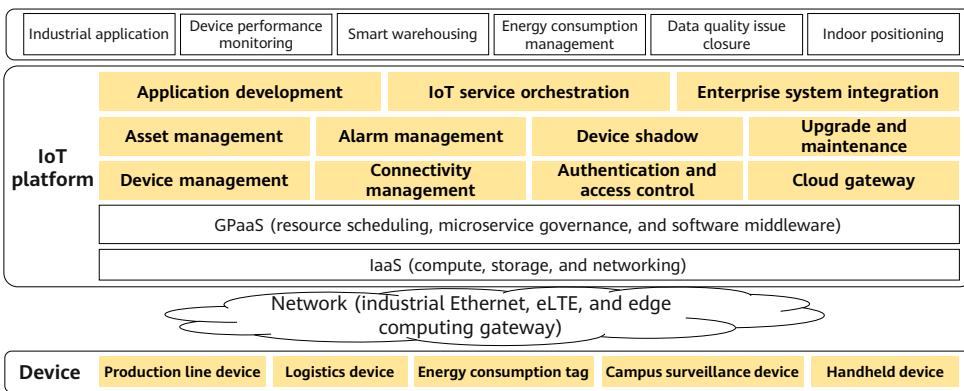
Industrial IoT Development Plans Across Countries

	2014–2017	2020–2022 Industrial IoT	2025
Germany	✓ Implemented Industry 4.0.	✓ Complete manufacturing communications standardization .	✓ Unify EU Industry 4.0 standards.
China	✓ Released <i>China Manufacturing 2025</i> initiative.	✓ Establish a standards system for smart manufacturing communications equipment. ✓ Chinese industrial robots occupy 50% of the market. ✓ Large-scale use of industrial wireless networks with a bandwidth of 500 Mbit/s	✓ Breakthroughs in 10 fields, such as automotive, healthcare, and energy ✓ Large-scale use of industrial wireless communications networks with a bandwidth of 2 Gbit/s
US	✓ Released the <i>Revitalize American Manufacturing and Innovation Act</i> .	✓ Complete flexible production line assembly within 24 hours.	✓ Invest in 1.9 billion dollars to build 45 innovation organizations. ✓ Complete flexible production line assembly within 8 hours.
Japan	✓ Released new strategies for robots and IoT. ✓ Released the Industrial Value Chain Initiative (IVI).	✓ Complete international standardization of manufacturing robots. ✓ Large-scale commercial use of service robots.	✓ Improve manufacturing informatization level from 30% to 50% .
South Korea	✓ Proposed <i>Manufacturing Industry Innovation 3.0</i> .	✓ Develop IoT/smart manufacturing, make 30% existing factories intelligent, and develop 10,000 intelligent production lines.	✓ Invest 23 billion dollars in 13 industries, such as UAVs, smart vehicles, and healthcare. The export amount exceeds Japan's.

ICT as the Production System of Smart Manufacturing Enterprises

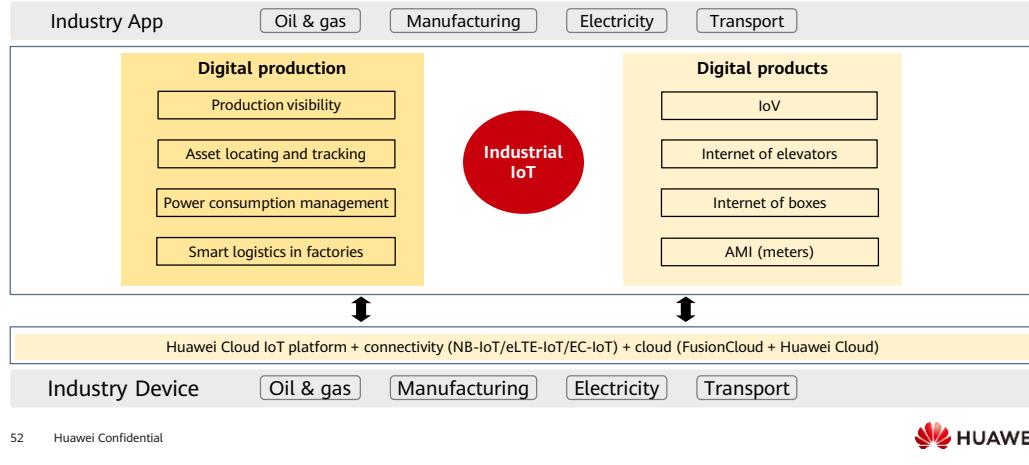


Technical Architecture of Huawei Industrial IoT Solution



Digital Production in the Factory and Digital Products Outside the Factory

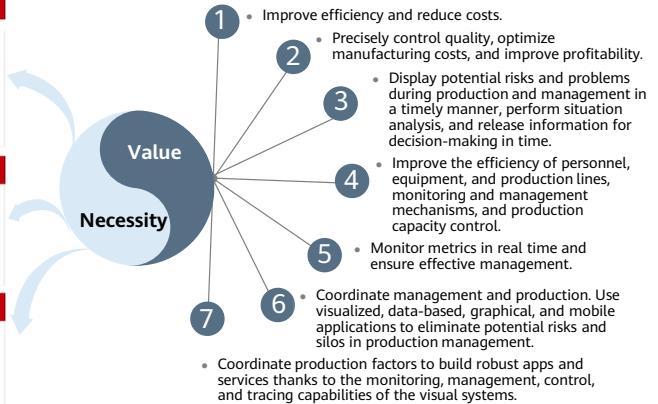
- Huawei provides the Industrial IoT platform, and ecosystem partners develop industry-specific devices and applications.



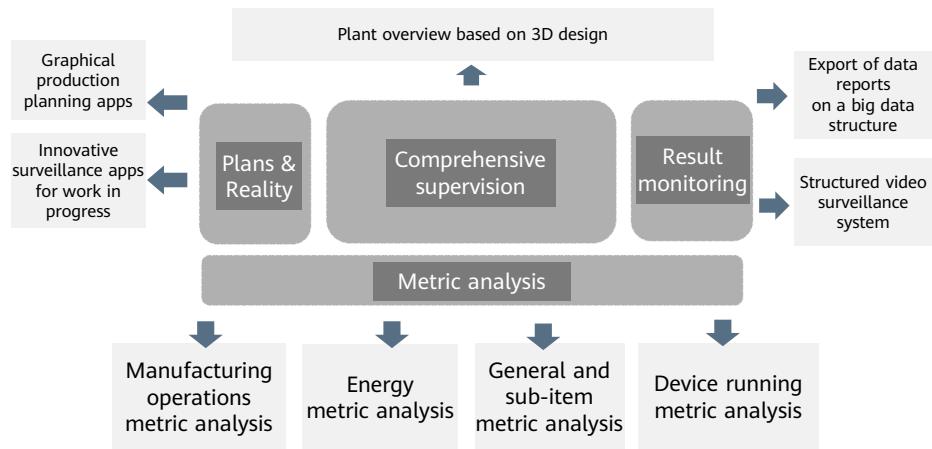
Necessity and Value of Visual Production Systems

Make data intuitive for production and decision-making.

For industry
Improve production reliability, equipment efficiency, and management effectiveness, helping automakers stay competitive in a long term.
Improve the supervision, management, and control on production processes, ensure the quality and reputation, and reduce costs.
Provide valuable experience for product update, iteration, and upgrade.
For development
Provide managers and decision makers with a strong basis to know and decide.
Integrate intelligence and management, as well as automation and efficiency, to realize automated and intelligent vehicle manufacturing and more efficient personnel management.
For data application
Control asset running, predict energy consumption, and improve profitability.
Make data smart and use it to make informed decisions quickly.



Integrated Smart Factory Production Management Platform



Asset Locating and Tracking Demands



Location query

Display the locations of people, vehicles, objects, and tools in the workshop in real time.



Intelligent judgment

1. Determine whether the cart or materials are at the correct location.
2. When an employee enters an undesignated station, an audible and visual alarm is triggered and the system records the event. The aim is to prevent cross-work.

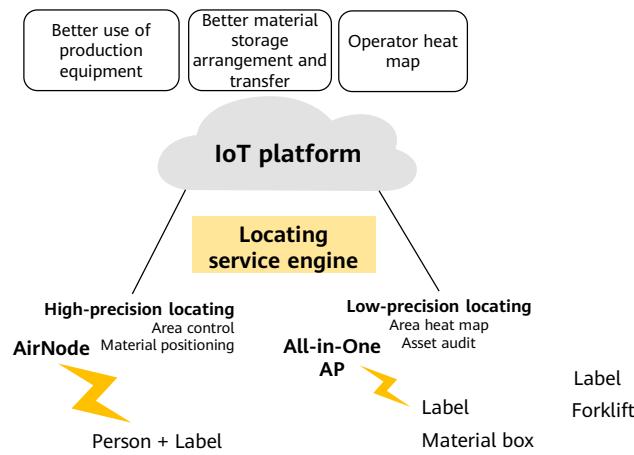
Bidirectional transmission of production dispatch instructions

After the AGV arrives at the station, information transmission is automatically triggered. The LED screen displays the work order and item attributes. The worker confirms the operation and returns the instruction.

Area division

1. Divide areas to set alarms for each area.
2. Taking away production tools and materials in specified areas will trigger alarms.
3. Unauthorized persons, objects, or vehicles will trigger alarms to the surveillance center.

Asset Locating and Tracking Solution



Platform and application systems

The IoT platform synchronizes the location data sent by the AirNode/All-in-One AP and calculates the location using the positioning engine. Based on the output coordinates of the positioning engine, the application system matches the indoor map and provides services such as heat map analysis, foolproof operation alarm, and personnel tracing and optimization according to service needs.

Network location

The positioning base station (AirNode/All-in-One AP) receives beacons sent by positioning labels and sends the beacons to the upper-layer IoT platform for parsing positioning information. High-precision positioning accuracy: 30 cm; low-precision positioning accuracy: 3 to 5 m

Location data collection

The key is labeling location information and periodically reporting location beacons. Labels use ultra-low power consumption chips and work continuously for more than three months. (The time changes according to the data reporting frequency.)



Quiz

1. (True or False) Wi-Fi is commonly used for parking, fire fighting, and manhole cover management in smart cities.
2. (True or False) From the perspective of applications, IoT applications can be classified into consumer IoT and industry IoT.

- Answers:
 - F
 - T

Summary

- This course explains the classification of IoT applications (consumer IoT and industry IoT) in common scenarios, such as smart city, smart home, IoV, and industrial IoT. In smart cities, there are more specific use cases, where you can learn about the challenges faced by each industry and the changes that digital solutions can bring to these industries.

Acronyms and Abbreviations

- API: Application Programming Interface
- IoT: Internet of Things
- LoRa: Long Range Radio, a low-power LAN wireless standard
- LTE: long term evolution
- NB-IoT: Narrowband Internet of Things
- PLC: programmable logic controller
- RFID: radio frequency identification
- RRU: remote radio unit

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and
organization for a fully connected,
intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive
statements including, without limitation, statements regarding
the future financial and operating results, future product
portfolio, new technology, etc. There are a number of factors
that could cause actual results and developments to differ
materially from those expressed or implied in the predictive
statements. Therefore, such information is provided for reference
purpose only and constitutes neither an offer nor an acceptance.
Huawei may change the information at any time without notice.



IoT Today



Foreword

- IoT has been developing for more than a decade and gaining momentum in recent years. Its market potential has been widely recognized, and new IoT technologies and applications are constantly springing up.
- IoT has been defined as a pillar of China's new infrastructure. It supports the development of digital economy. However, its development is held back by fragmentation, security risks, and high costs.
- This section describes how IoT has been developing, IoT fragmentation issues, and some of the solutions.

Objectives

- Upon completion of this course, you will understand:
 - How the IoT industry has been developing.
 - IoT fragmentation issues and some of the solutions.

Contents

1. Developmental Trends of the IoT Industry

- Global IoT Connectivity and Its Structure
 - IoT Developmental Trends
 - IoT Infrastructure Integration and Core Value Transformation
 - IoT Ecosystem Expansion and Security

2. IoT Fragmentation and Its Solutions

Increasing Global IoT Connectivity

- IoT is growing fast worldwide and has huge market potential. According to GSMA's *The Mobile Economy 2020*, the number of global IoT connections reached 12 billion in 2019 and will reach 24.6 billion in 2025, representing a compound annual growth rate (CAGR) of 13%.

Company	Connections in 2019	Connections in 2025
Ericsson	10.7 billion	24.6 billion
GSMA	12 billion	24.6 billion
IoT Analytics	8.3 billion	21.5 billion
Machina Research	10.7 billion	25.1 billion

- Data source: *13th Five-Year Plan Assessment Report on IoT* released by China Academy of Information and Communications Technology (CAICT)
- Data sources: Ericsson, GSMA, IoT Analytics, and Machina Research

Changes in the Structure of IoT Connections



Consumer IoT has been leading IoT development because it has a large user base, simple user requirements, mature technologies, and many different products. Most IoT connections are from consumer-oriented smart home products and wearables.

Industry IoT connections will take up more of the total IoT connections because more and more industries start using IoT for digitalization. According to GSMA Intelligence, the number of industry IoT devices will exceed that of consumer IoT devices by 2024. In 2019, the numbers of industry and consumer IoT connections in China are about even. It is estimated that by 2025 industry IoT will contribute most of the new connections, probably accounting for 61.2% of the total. According to predictions by various consulting companies, smart manufacturing, smart transportation, smart health, and smart energy will witness rapid increase of IoT connections.

- Source: CAICT

Internal and External Factors That Affect IoT Development

The COVID-19 accelerates the use of IoT.

Industry requirements drive commercial adoption.



5G Release 16 was frozen.
This standard provides technical support for all-scenario IoT coverage.



IoT is used as new infrastructure and the foundation of a digital economy.



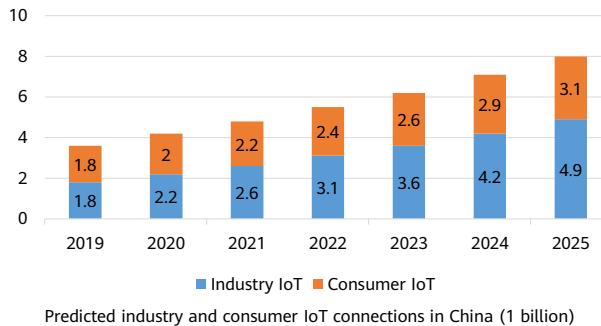
In global economic recession, IoT accelerates economic restructuring to stimulate demand.



- Image source: <http://3ms.huawei.com/km/static/image/detail.html?fid=59393>

Predicted Industry and Consumer IoT Connections in China

- Industry IoT connections will take up more of the total IoT connections because more and more industries start using IoT for digitalization. According to GSMA Intelligence, the number of industry IoT devices will exceed that of consumer IoT devices by 2024.



- According to GSMA Intelligence, the number of industry IoT devices will exceed that of consumer IoT devices by 2024. In 2019, the numbers of industry and consumer IoT connections in China are about even. It is estimated that by 2025 industry IoT will contribute most of the new connections, probably accounting for 61.2% of the total. According to predictions by various consulting companies, smart manufacturing, smart transportation, smart health, and smart energy will witness rapid increase of IoT connections.

Three Pillars of Resource Openness and Integration

Internal cross-layer integration	Upstream and downstream industry chain collaboration and developer gathering
<p>Leading enterprises integrate clouds, devices, connections, applications, and services to build comprehensive IoT platforms.</p> <ul style="list-style-type: none">• China Mobile: OneLink + OneNet• SAP: SAP HANA + SAP Leonardo• Baidu Tiangong: IoT Hub + IoT Parser + IoT Device• Alibaba Cloud: IoT Platform + Link Develop + Link Market <p>Profiles digitalize devices and describe device statuses, details, and functions, enabling unified connections of devices from different manufacturers to a platform.</p> <ul style="list-style-type: none">➢ OCF, OMA, Bluetooth SIG, ZigBee Alliance co-established a One Data Model group to integrate vendors' model strengths and standardize profiles in 2019.➢ Two-step strategy: ① Establish a universal model framework and language. ② Standardize profiles.➢ Industry standards: Carriers, device manufacturers, Internet service providers, and vertical industry giants co-formulated unified IoT profile standards, which were approved by China Communications Standards Association (CCSA). These standards were first applied in the smart home industry and then extended to logistics, new retail, and other industries.	<p>Alibaba Cloud Link IoT works with eight chip manufacturers to launch platform-independent communications models, which can be used to connect devices to platforms. This forms a partner ecosystem.</p> <ul style="list-style-type: none">• Huawei Cloud IoTDA provides developer training and certification, OpenLab, a global marketing platform, and an application developer ecosystem.• AWS IoT has built a huge cooperative network of IoT device manufacturers to break down hardware compatibility barriers for IoT applications.• IoT startups are encouraged to use its platforms by providing SDKs and training.

9 Huawei Confidential



- Open Connectivity Foundation (OCF) was jointly established by Microsoft, Cisco, General Electric, Qualcomm, Electrolux, Intel, and Samsung on February 19, 2016. It aims to create IoT specifications and protocols to ensure compatibility of devices from different manufacturers.
- Open Mobile Alliance (OMA) was co-founded by the WAP Forum and Open Mobile Architecture (OMA) in June 2002.
- IoT Connectivity Alliance (ICA) is a standard organization jointly initiated by Alibaba Cloud IoT and its partners in June 2017.

Contents

1. Developmental Trends of the IoT Industry

- Global IoT Connectivity and Its Structure
- IoT Developmental Trends
- IoT Infrastructure Integration and Core Value Transformation
- IoT Ecosystem Expansion and Security

2. IoT Fragmentation and Its Solutions

Industry Convergence Promotes the Integration of IoT and Blockchain

Integration of IoT and Blockchain

Industry IoT requires upgrades to product design, production, and circulation. The integration of IoT and blockchain (BioT) streamlines processes within and across enterprises.

BioT enables process information exchange. For example, logistics are made more traceable.

BioT enables value sharing between enterprises. For example, if multiple enterprises collaborate to deliver complex products at scale, design, supply, production, and logistics processes need to be streamlined for efficient delivery.

- The blockchain technology writes hashed off-chain data to a blockchain as the hash digest. This prevents data theft and reduces storage requirements of data transmission. On-chain data cannot be tampered with and can be traced. This improves data reliability and provides a trust foundation for data collaboration. The blockchain technology improves IoT device security. IoT also ensures off-chain data authenticity.
- IoT is extended from the Internet, while the blockchain technology is based on IoT to transfer value. The integration of blockchain and IoT not only improves IoT device security and scalability, but also expands blockchain application scenarios and simplifies blockchain use for traditional industries.

Intelligence Highlights Key IoT Processes

Device side

Widespread adopting of IoT drives edge intelligence requirements such as real-time data analysis, processing, decision-making, and autonomy. According to International Data Corporation (IDC), more than 50% of our data will need to be analyzed, processed, and stored at the edge.

Edge intelligence is attracting a lot of attention. Every industry is exploring edge intelligence and cloud-edge synergy.

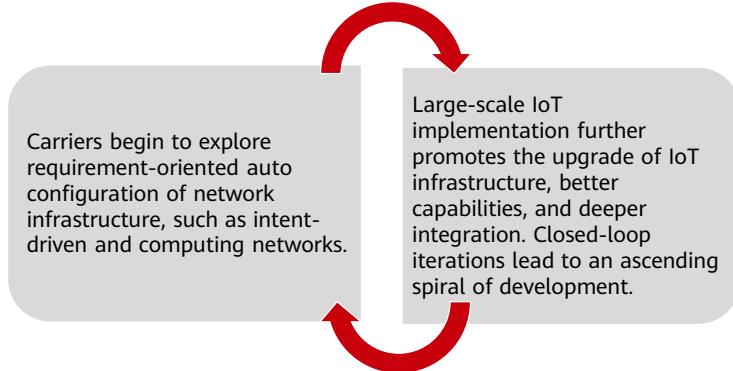
Service side

Certified Strategic Management Analyst (CSMA) estimates that by 2025 the value of IoT platforms, applications, and services will grow the fastest, contributing 67% of the total IoT revenue, while IoT connections will account for only 5%.

As the number of IoT connections grows exponentially, deployment of intelligent applications and services will grow fast as well.

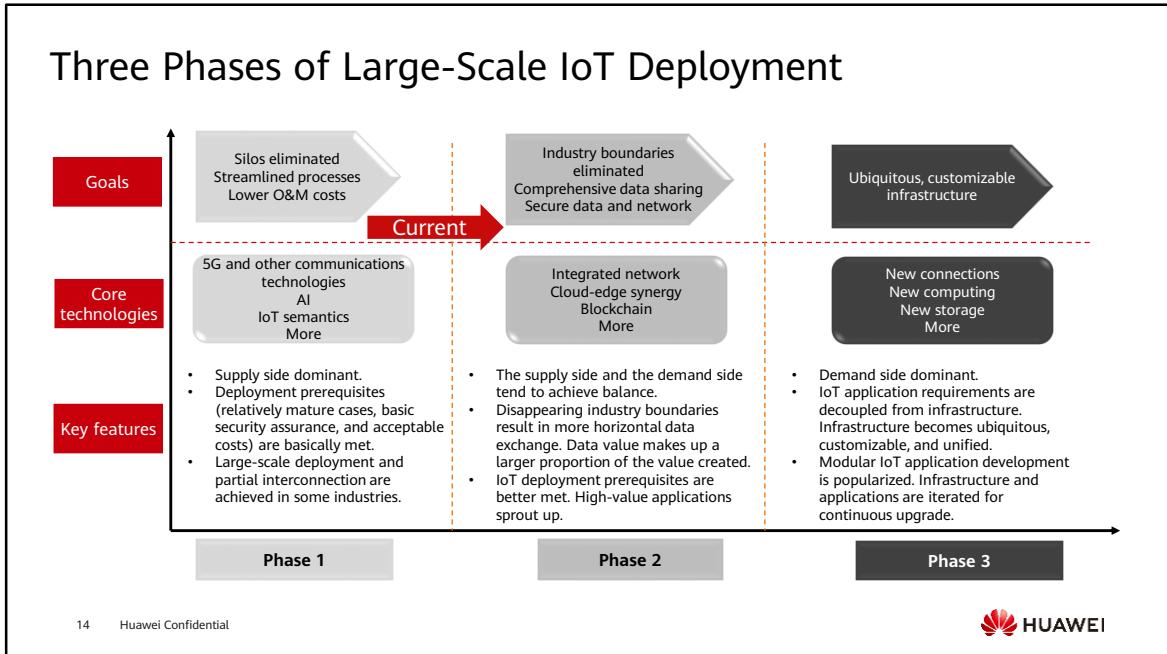
Customizable Infrastructure

- Customizable IoT infrastructure allows users to build IoT hardware capabilities based on their requirements.



- Customizable infrastructure includes the resource pools for different industry requirements. It provides comprehensive support, such as application development management, network resource scheduling, and hardware configuration.
- Customizable infrastructure simplifies IoT application development and expands the use of IoT at scale.
- Intent-driven network: oriented to user intents or business goals. Intents can be narrowed down to the intents of network O&M personnel and network architects. The system receives, translates, verifies, and executes user requirements while preventing deviations.
- Computing network: new information infrastructure that flexibly allocates and schedules compute, storage, and network resources among clouds, edges, and networks based on service requirements.

Three Phases of Large-Scale IoT Deployment



- Now the prerequisites of IoT deployment burst are not fully met.

Three Challenges of Large-Scale IoT Deployment

Fragmented requirements and complex technologies

- IoT scenarios and requirements are **fragmented**, and the technologies are **not unified**: heterogeneous devices, multiple modes of communication, isolated platforms, diverse applications, and poor interoperability and interconnection between different vendors' devices or products.
- According to Microsoft's *IoT Signals*, **complexity and technical challenges** are major barriers to companies wanting to use IoT.

Frequent security issues and increasing risks

- **Security is a top concern** in IoT development. In recent years, there have been numerous IoT security issues. Smart home products, cameras, and even power grids were attacked. These issues affect enterprise production and society in general. They cause huge economic losses.

High costs and a lack of clear business models

- Nearly 70% of enterprises hesitate to deploy IoT because of cost concerns.

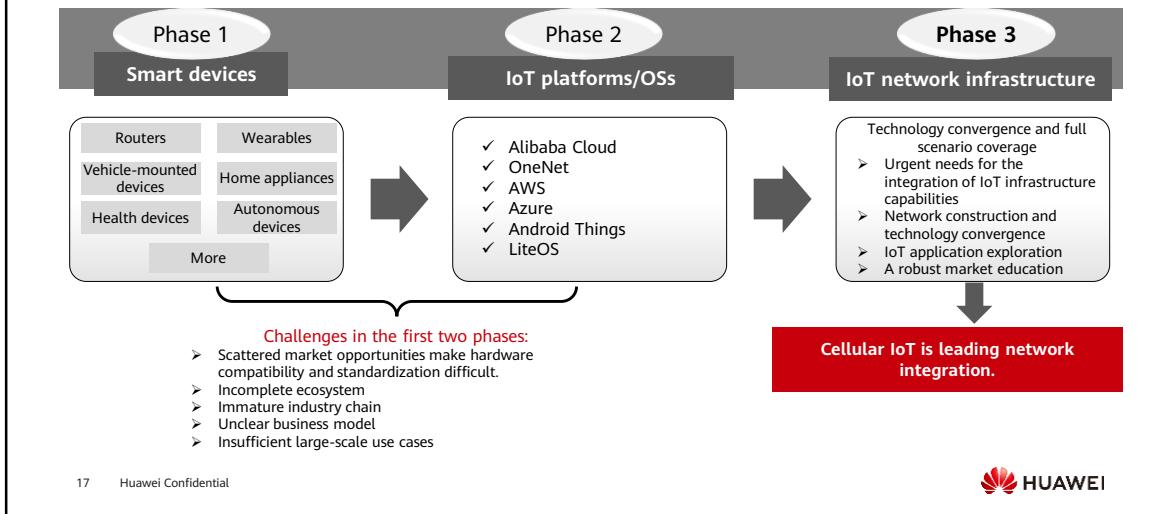
Contents

1. Developmental Trends of the IoT Industry

- Global IoT Connectivity and Its Structure
- IoT Developmental Trends
- **IoT Infrastructure Integration and Core Value Transformation**
- IoT Ecosystem Expansion and Security

2. IoT Fragmentation and Its Solutions

IoT Infrastructure Integration Is Entering a New Phase



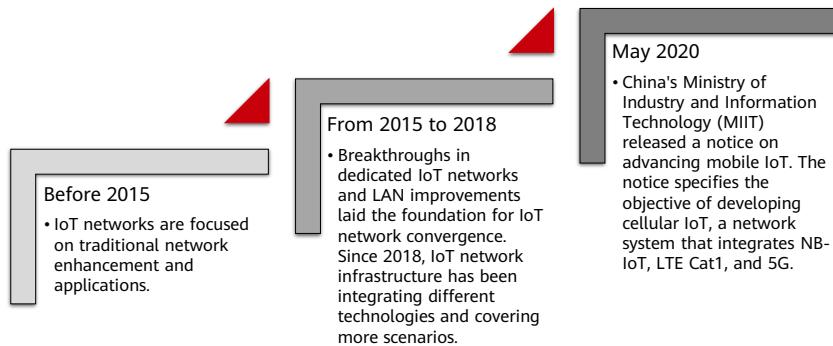
17 Huawei Confidential



- Scenario-specific infrastructures are continuously integrated.

Evolution into Space-Air-Ground Integrated Networks

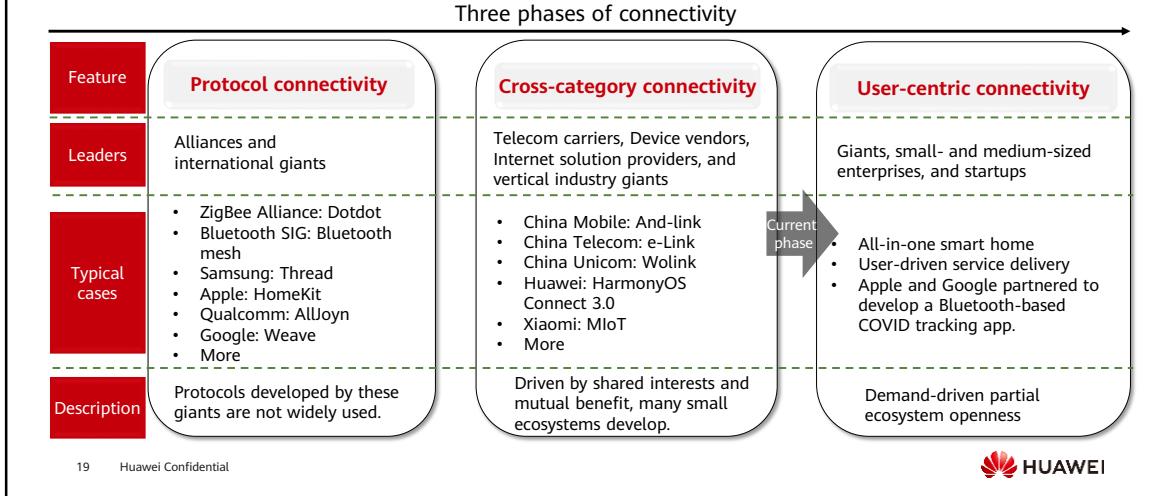
- Mobile networks (cellular IoT and unlicensed IoT networks), local area networks (LANs), satellite networks, drones, and hot air balloons form global IoT networks that integrate space, air, and ground. They provide reliable access for global IoT applications anytime, anywhere.



- In May 2020, China's MIIT released a notice on advancing mobile IoT, which is different from the notice for advancing NB-IoT released in 2017. The new notice specifies the objective of developing cellular IoT, a network system that integrates NB-IoT, LTE Cat1, and 5G.

IoT Connectivity: from Enterprise-Driven to User-Centric

Scenario-specific user demands replace the supply side as the main driving force of connectivity.



- ZigBee is a wireless communications technology designed for smart home. It complies with Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 running at the 2.4 GHz frequency band. Dotdot is different from ZigBee. ZigBee is a complete solution. Dotdot is not limited to IEEE 802.15.4. It can use other connection methods, such as Wi-Fi, Bluetooth, Ethernet, and even wired communications.

Contents

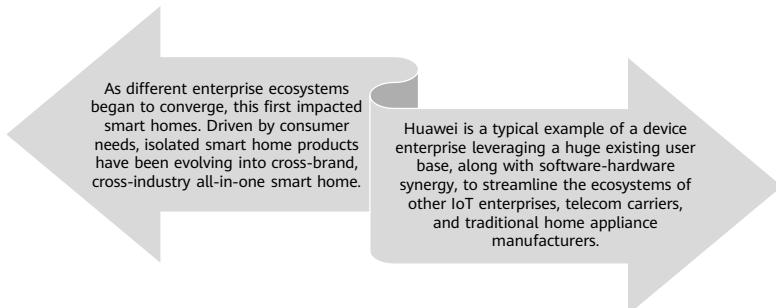
1. Developmental Trends of the IoT Industry

- Global IoT Connectivity and Its Structure
- IoT Developmental Trends
- IoT Infrastructure Integration and Core Value Transformation
- IoT Ecosystem Expansion and Security

2. IoT Fragmentation and Its Solutions

Collaboration Promotes Ecosystem Convergence

- It is difficult for enterprises to build competitive IoT ecosystems because of IoT fragmentation and the limitations of their products and partners.



- Huawei and JD achieved cloud interconnection. Huawei devices (such as mobile phones, routers, and wearables) can connect with JD's full series of smart hardware. Huawei also cooperated with manufacturers in vertical industries, such as water purification, lighting, home robots, and real estate to build an all-in-one smart home solution.

Leveraging Open Source Technologies and Profiles

- According to Eclipse's survey on commercial adoption of IoT, open source pervades IoT with 60% of companies including open source in their IoT deployment plans. Currently, 82% of IoT code is open source. Open source code helps enterprises leverage collaborative strengths, accelerate R&D, reduce development costs, and shorten time to market (TTM).

Market-driven open source ecosystem construction

- The Connected Home Over IP (CHIP) working group, led by ZigBee Alliance, was established in 2019. It developed a new smart home standard that allows different hardware to connect to the Internet, improving compatibility and simplifying development. CHIP is also supported by Amazon Alexa, Apple Siri, Google Assistant, and other mainstream smart assistants or platforms.

Profile research

- ZigBee Alliance, Bluetooth SIG, OCF, oneM2M, OMA, and W3C are all building internal profiles. ODM worked with leading enterprises worldwide to standardize IoT data models and released the first versions of various standards.

- With the increasing demand for simplified device development and information exchange, profiles have become a hot research topic. Profiles will break information silos between different devices, software/hardware platforms, operating systems, and network environments. Devices from different vendors can easily connect to a platform using profiles.

Strengthened IoT Security Supervision

Developed countries have been strengthening IoT security regulations.

Released code of practice

Established standards

Released regulatory proposals

Carried out laws

UK

In October 2018, *Code of Practice for Consumer IoT Security* was released. It stipulates 13 guidelines for manufacturers.

In February 2019, ETSI formulated *ETSI TS 103 645 Cybersecurity for Consumer IoT* based on the 13 guidelines.

In May 2019, regulatory proposals for consumer IoT security were released. Retailers are mandated to only sell consumer IoT products with IoT security labels.

In January 2020, the first three of the 13 guidelines were incorporated into legislation. Manufacturers must not use default passwords and must provide public points to report vulnerabilities and keep software updated.

Japan: mandatory nationwide blind testing

Released guidelines

- In 2017, the Ministry of Internal Affairs and Communications (MIC) announced comprehensive IoT security measures, including the establishment of an IoT vulnerability defense system, defense exercises, and international collaboration.

Conducted blind testing

- In February 2019, NICT (authorized by the MIC) launched the NOTICE project to test nationwide IoT device security without notifying device owners.

US: starting from IoT devices purchased by the government

- IoT-related legal documents have been released since 2016. *Internet of Things Cybersecurity Improvement Act* was passed in June 2019 and has gradually grown more influential over time.
- National Institute of Standards and Technology (NIST) released an IoT security report.
- The US federal government must comply with NIST guidelines.
- NIST collaborates with cybersecurity researchers and industry experts to release action guidelines on vulnerability disclosure.
- Contractors and resellers that provide IoT devices to the government must follow consistent vulnerability disclosure guidelines on how to disclose vulnerabilities in a timely manner.

Contents

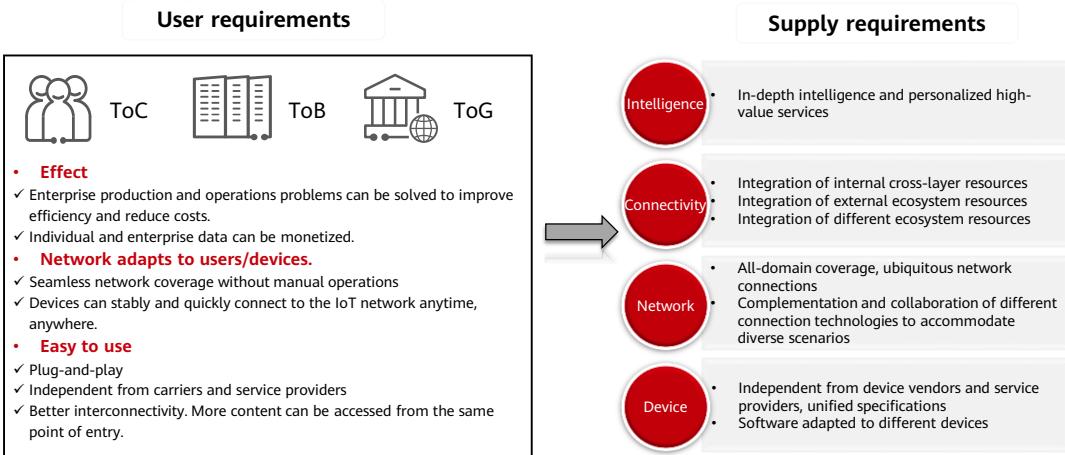
1. Developmental Trends of the IoT Industry

2. IoT Fragmentation and Its Solutions

- Fragmentation Issues

- eSIM, IPv6, and IoT
- Integrated IoT Network Infrastructure
- AIoT

IoT Fragmentation on the User and Supply Sides



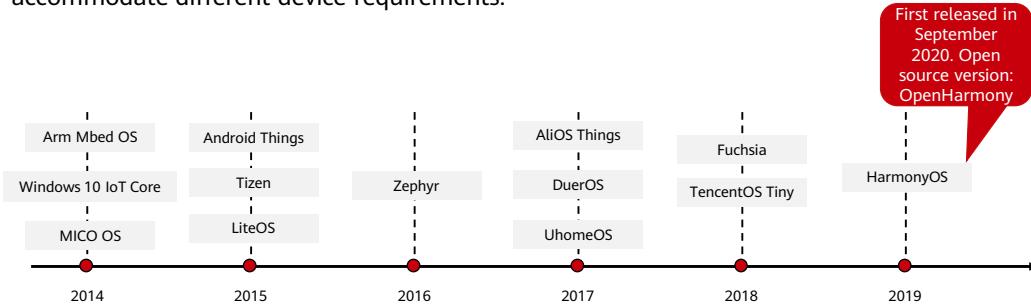
25 Huawei Confidential



- ToB: oriented to commercial businesses
- ToC: oriented to individual consumers
- ToG: oriented to governments

IoT OS Development to Accommodate Diverse Range of Devices

- IoT operating systems (OSs) coordinate and control southbound software and hardware resources and provide northbound APIs for developers and users. There are many IoT OSs to accommodate different device requirements.



- Three development paths: a. OSs represented by Google Android Wear OS and Apple watchOS and tvOS; b. embedded real-time OSs added with IoT functions such as CoAP and MQTT; c. dedicated IoT OSs that support scalability, real-time performance, and reliability to better meet IoT application requirements.

Three Developmental Trends of Dedicated IoT OSs

OS customization in specific IoT industries

- Some automobile manufacturers create custom OSs based on Linux, Android, and QNX.

One OS for different devices in consumer IoT

- Example: HarmonyOS

Device-edge-cloud linked OSs developed by leading enterprises for more efficient resource configuration and scheduling

- Example: Huawei Cloud, OpenEuler, and HarmonyOS jointly build cloud-edge-device linkage.

IoT OS Maturation Takes Longer

1.

Consumers have existing usage habits. New OSs must be compatible with the apps of mature, mainstream OSs.

2.

An OS is not a single product but an entire ecosystem. It takes time to build a new ecosystem, develop applications, cultivate developers, build alliances with hardware vendors, partner with existing application stores, and establish business relationships.

Contents

1. Developmental Trends of the IoT Industry

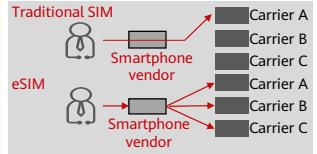
2. IoT Fragmentation and Its Solutions

- Fragmentation Issues
- eSIM, IPv6, and IoT
- Integrated IoT Network Infrastructure
- AIoT

IoT eSIM Decouples Devices from Carriers

eSIM: OTA functions, flexible carrier switching, enriching IoT products

- OTA functions enable remote download, activation, and configuration. Users can flexibly switch carriers. Multiple devices can share the same number.
- Physical features: heat-resistant, shock-resistant, miniaturized, and integrated. It improves connection reliability, waterproofing, and dustproofing, enabling devices to be used in more IoT scenarios.
- eSIM IDs (EIDs) unify interfaces and provide seamless IoT connectivity.
- To address fragmented supply chains, undifferentiated eSIM cards prevent device manufacturers from purchasing different IoT cards.



Large-scale eSIM use in the world

Global carriers, device manufacturers, and card vendors launched a series of eSIM products, which are used in smart vehicles and smart water meters.

- US: In 2014, a global eSIM solution was launched and used in smart vehicles.
- Japan: In 2017, SoftBank established an eSIM platform and deployed it globally to help the development of the new IoT industries.
- Europe: In 2016, Vodafone and Giesecke & Devrient (G&D) jointly launched an eSIM management solution targeting smart vehicles.
- Germany: In 2018, G&D, BMW, Intel, and Deutsche Telekom jointly developed a management solution to provide entertainment and information services for users based on eSIM.
- Russia: In 2020, Russia mandated that new vehicles use the ERA GLONASS eCall system.

VS

In China, eSIM is mainly used for wearables.

China's eSIM development is still in the early stages. Currently, only independent numbers for wearables and number sharing across devices have been deployed commercially.

- China Mobile: developed proprietary eSIM modules and chips in 2019, which are planned to be used in IoT and smart firefighting.
- China Unicom: worked with BMW, Volkswagen, Volvo, and other automobile enterprises 2018 and has been completing commissioning with more and more automobile enterprises.
- China Telecom: launched eSIM smart pipe modules in 2018, which serve as the basis for China Telecom to build comprehensive IoT services.
- In 2019, MIIT approved China Unicom to provide eSIM application services in IoT and other fields.



Challenges Faced by eSIM

Challenges

GSMA predicts that eSIM will be mainly used in IoT, logistics, energy, and public utilities in China. Progress has been slow and there is still a long way to go.

The industry is not mature and lacks established rules. There is friction between the different roles and business models in the industry chain. The three major carriers are focused on different solutions and scenarios. There are no guidelines for business development and product design.

Standardization and policy guidance are insufficient. There are different eSIM technical specifications. So far, industry authorities have only approved the use of eSIM in smart wearables.

IPv6 and IoT for the Internet of Everything

IPv6: urgently needed for connecting IoT devices at scale

- 4.3 billion IPv4 addresses have been assigned since November 2019. Global IPv4 addresses have run out. It is expected that the number of IoT connections will reach 25 to 50 billion by 2025.

IPv6 + IoT

- **Address assignment at scale:** IPv6 uses stateless address assignment. The network side does not need to save node address statuses. This simplifies address assignment and enables addresses to be assigned at scale. Very few resources are needed.
- **Mobility support:** IPv6 introduces mobile node detection, allowing real-time monitoring in any area. It also notifies the platform of prefix addresses in real time, which is a requirement for IoT.
- **Quality control:** IPv6 provides refined service quality control, facilitating flexible IoT QoS management. It also meets timeliness and other service quality requirements of IoT applications.

Continuous upgrade and reconstruction

- **Fast network reconstruction to support IPv6:** Fixed networks have been upgraded to support IPv6. The three major carriers in China have upgraded LTE networks and metropolitan area networks (MANs) of 30 provinces. IPv6 and IPv4 network quality is the same for the nodes directly connected to the backbone network.
- **Platform reconstruction in progress:** More than 70% of public cloud products have completed IPv4 and IPv6 dual-stack reconstruction, but there are not many regions where IPv6-supported public cloud products are available.
- **Insufficient IPv6 support for IoT devices:** Although 90% of home gateways and routers support IPv6, IPv6 management is weak. Although promising in terms of future adoption, industry IoT devices do not currently support IPv6.

Contents

1. Developmental Trends of the IoT Industry

2. IoT Fragmentation and Its Solutions

- Fragmentation Issues
- eSIM, IPv6, and IoT
- **Integrated IoT Network Infrastructure**
- AIoT

A Network That Adapts to Users in Every Scenario

Exploration of space-air-ground integrated networks

Coverage and technologies	Indoor Road	Short distance	mmWave and terahertz ZigBee 3.0, Bluetooth 5.0, and Wi-Fi 6	NB-IoT + satellites Usage: Ubiquitous NB-IoT networks based on satellites can be used in agriculture, transportation, navigation, and emergency response domains and cover oceans and mountains. Status quo: Startups, such as Ligado Networks and OQ Technology, dominate the market. They will cooperate and compete with carriers' cellular networks. The World's first successful NB-IoT satellite IoT data connection In August 2020, MediaTek developed satellite-enabled devices based on standard NB-IoT chipsets and successfully established bidirectional links with Inmarsat's Geostationary Orbit (GEO), enabling global IoT coverage.
	Base station Forest	Medium- and long- distance	Cellular mobile communications Unlicensed WAN	
	Ocean Desert	Long- distance	Satellite communications	
	Integration of 5G and non-terrestrial network (NTN)			
<p>5G Release 16: focused on the integration of the 5G air interface and NTN.</p> <p>5G Release 17 evolution: focused on the integration of NB-IoT/eMTC and NTN. Normative work on 5G NR enhancements will be carried out to support NTN access for satellites and high-altitude platforms.</p> <p>Studies will be conducted to introduce NB-IoT and eMTC support for satellites.</p>				Cellular network + unmanned aerial vehicles (UAVs)/hot air balloons Leading international enterprises explored the network layout in which UAVs and hot air balloons are used as hotspots to supplement cellular networks. <ul style="list-style-type: none"> Facebook: Project Aquila explored the air-space network coverage using the cellular network and UAVs. Google: Project Loon is intended to provide telecom services for remote areas using high-altitude hot air balloons.

Contents

1. Developmental Trends of the IoT Industry

2. IoT Fragmentation and Its Solutions

- Fragmentation Issues
- eSIM, IPv6, and IoT
- Integrated IoT Network Infrastructure
- AIoT

AI + IoT

- AIoT integrates AI and IoT. Vast quantities of data generated and collected are stored in the cloud or at the edge. Through big data analysis and AI processing, everything can be digitalized and intelligently connected.

Supply side

The rapid development of AI chips, hardware, algorithms, and platforms reduces the costs of data collection, analysis, and storage, and simplifies AIoT use.

The commercial use of 5G enhances connections and has driven the explosive growth of IoT data. This means we need more efficient data analysis.

Demand side

Consumers want a better experience and more convenience from products such as smart life assistants. They are more willing to buy products (such as XR headsets) that use premium pricing, and they are willing to try customized smart products.

Industries need automated and intelligent production equipment and systems. As labor costs rise and digital transformation continues, industries prioritize efficiency and costs of production and operations.

A smart city solution needs to accommodate requirements of different scenarios, including transportation, municipal administration, environmental protection, livelihood, service efficiency, and security.

Quiz

1. (Multiple-answer question) Which of the following are the key problems facing large-scale IoT deployment?
 - A. Fragmented requirements and complex technologies
 - B. Frequent security issues and increasing risks
 - C. High costs and lack of clear business models
 - D. Lack of communication modes suitable for IoT scenarios
2. (True or false) To solve IoT fragmentation, enterprises collaborate with each other and leverage open source technologies and profiles.

- Answers:
 - ABC
 - T

Summary

- In this section, you have learned about the latest developmental trends of the IoT industry, scenarios-based infrastructure integration, and IoT connectivity change.
- You also learned about IoT fragmentation issues and their solutions: using eSIM to decouple devices from carriers, using IPv6 in IoT, and promoting AIoT for device-pipe-cloud integration.

Acronyms or Abbreviations

- AIoT: the combination of artificial intelligence (AI) and the Internet of Things (IoT).
- eSIM: Embedded SIM
- IoT: Internet of Things
- NB-IoT: Narrowband Internet of Things
- RFID: Radio Frequency Identification
- SIM: Subscriber Identity Module
- TPU: Tensor Processing Unit

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and
organization for a fully connected,
intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive
statements including, without limitation, statements regarding
the future financial and operating results, future product
portfolio, new technology, etc. There are a number of factors
that could cause actual results and developments to differ
materially from those expressed or implied in the predictive
statements. Therefore, such information is provided for reference
purpose only and constitutes neither an offer nor an acceptance.
Huawei may change the information at any time without notice.



Data Collection Technologies



Foreword

- In the IoT era, data is an important element throughout the four-layer IoT architecture. Data is collected at the sensing layer, transmitted at the network layer, processed at the platform layer, and finally used at the application layer. It is important to understand how data is generated.
- This section describes three major data collection methods in the IoT industry: sensor data collection, tag identification information collection, and location data collection.

Objectives

- Upon completion of this course, you will understand:
 - Sensor data collection.
 - Sensor classification and working principles.
 - Classification and working principles of automatic identification technologies.
 - Classification and working principles of location data collection technologies.

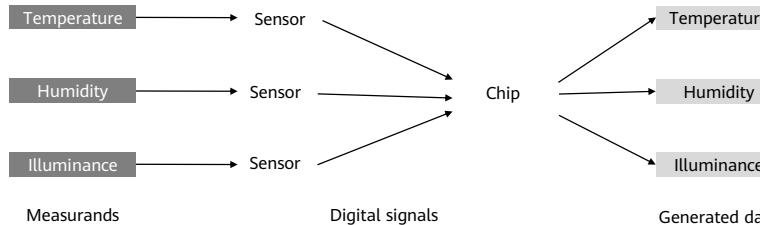
Contents

1. Sensing Technologies

- Overview
 - Sensor Classification
 - Working Principles of Common Sensors
- 2. Tag Identification Technologies
- 3. Location Data Collection Technologies

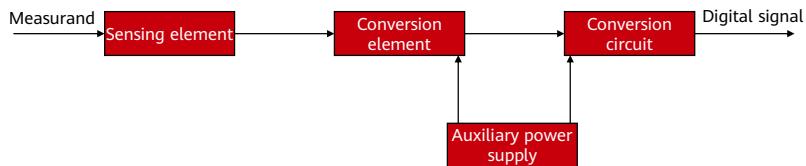
Sensing Technologies

- Information technology is built on three main pillars: computers, communications, and sensing technologies. In terms of IoT, sensing technologies are an important indicator to measure a country's informatization level.
- Sensing technology refers to sensors that can sense something about the ambient environment or that detect some special substance. They sense things like gas, light, temperature, humidity, human bodies, and more. Sensors convert analog signals into digital ones and send them to CPUs for processing. The final output may be gas concentration parameters, light intensity parameters, temperature and humidity data, and so on.



Sensors

- Definition
 - A sensor is a device that senses a measurand and converts it into a usable signal based on certain rules. It usually consists of sensing and conversion elements.
- Application Scenarios
 - Sensors are used in the military, energy, robotics, automatic control, environmental protection, transportation, healthcare, household appliances, remote sensing, and chemicals industries.



- IoT devices collect a lot of data. Device data comes from sensors. Sensors enable devices to see, hear, smell, and feel their surroundings like human beings, but in a more precise and powerful way. For example, a human being is not able to tell the precise temperature of an object by touching it, to touch an object that is a thousand degrees Celsius, or to identify subtle temperature changes. Sensors provide these functions.

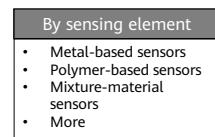
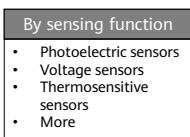
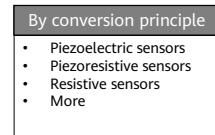
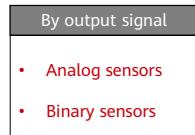
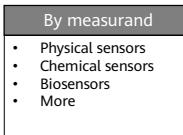
Contents

1. Sensing Technologies

- Overview
 - Sensor Classification
 - Working Principles of Common Sensors
2. Tag Identification Technologies
 3. Location Data Collection Technologies

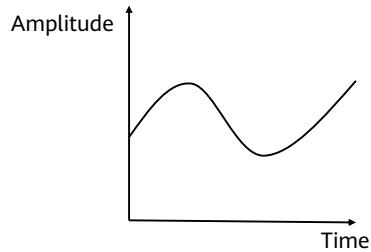
Sensor Classification

- There are various types of sensors with different working principles. A physical quantity can be measured using different technologies. Sensors with the same working principle can measure different measurands.
- Sensors can be classified using different methods:



Analog Sensors

- In the real world, all physical quantities, such as illuminance, temperature, and distance, are analog quantities. Analog quantities are **continuous**. They vary constantly within a certain range.
- Analog sensors output continuous signals and present measurand values using voltage, current, and temperature. The figure shows the current signal output by a thermistor. Any value on the curve represents a temperature value.



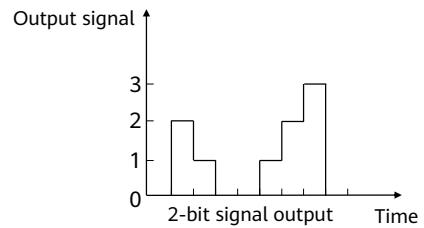
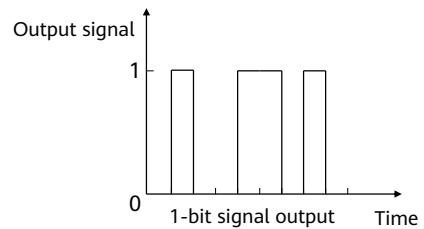
- Voltage signals output by a thermocouple are analog signals. The temperature will never suddenly jump, so the voltage signals are continuous in terms of time and value. Any value on the voltage signal curve has a physical meaning. It represents a corresponding temperature.

Binary Sensors

- Signals output by binary sensors are **not continuous**. Binary sensors reflect two status signals: **open** and **closed**, similar to 1 and 0 of digital sensors. They are the most typical type of sensors and are used to detect object statuses, actions, or positions.
- Common binary sensors include proximity switches, photoelectric switches, and magnetic switches.
- Take an infrared human body sensor as an example:
 - Features: A detection element converts received infrared radiation into weak voltage signals. The field effect transistor (FET) in the probe amplifies and outputs the signals.
 - Use cases: automatic lighting control, non-contact temperature measurement, automatic access control, etc.
 - Output: high level (3.3 V) when there is no person; low level (0 V) when there is a person.

Digital Sensors (1)

- Computers represent data using binary digits 1 and 0. Digital quantities are discrete physical quantities in terms of time and value because their changes are **not continuous** over time.
- Digital sensors transform the A/D conversion modules of analog sensors to output digital signals. A digital sensor contains an **amplifier, analog-to-digital converter (ADC), microprocessor, memory, communication interfaces, and temperature detection circuit**.
- The figures on the right show the digital sensor output, which is adjusted based on the ADC resolution.



- For example, an electronic circuit is used to record the number of parts produced on an automatic production line. A signal is sent to the electronic circuit each time a part is produced, and the signal is recorded as 1. When no part is produced, the signal sent to the electronic circuit is 0. Signals for recording the number of parts are not continuous in terms of time and value, so they are digital signals. The minimum quantity unit is 1.

Digital Sensors (2)

- Digital sensors convert analog quantities into digital quantities in four steps: **sampling**, **holding**, **quantization**, and **encoding**.
- Analog-to-digital conversion formula:

$$A/D = \frac{U_A}{V_{DD}} \cdot 2^n = \frac{2^n}{V_{DD}} \cdot U_A$$

- Where,
 - A/D : digital quantity
 - U_A : analog quantity
 - n : ADC resolution (number of bits)
 - V_{DD} : power supply voltage of the conversion circuit
- If an 8-bit sensor is used and the power supply voltage is 3.3 V, the digital quantity is $\frac{256}{3.3} \cdot U_A$.

Contents

1. Sensing Technologies

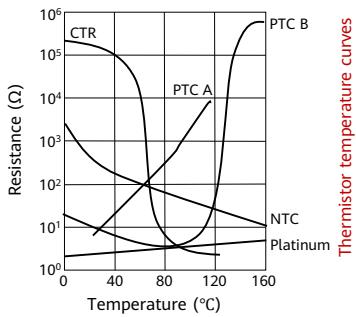
- Overview
- Sensor Classification
- **Working Principles of Common Sensors**

2. Tag Identification Technologies

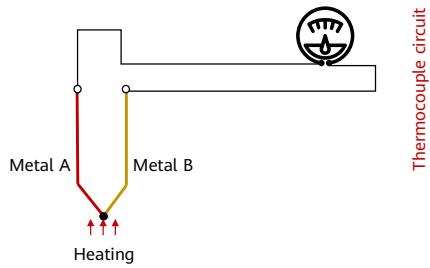
3. Location Data Collection Technologies

Temperature Sensors

- Temperature sensors sense temperatures and convert them into usable signals. Based on the characteristics of the sensing materials and electronic components used, temperature sensors can be classified as **resistance temperature detectors (RTDs)** or **thermocouple sensors**. An RTD is a thermistor. Metal resistance varies with the temperature. In a thermocouple sensor, two different metal wires at each end are connected. When the connected metal wires are heated, there is an electric potential difference in the thermocouple circuit, and the electric potential difference can be used to calculate the temperature.



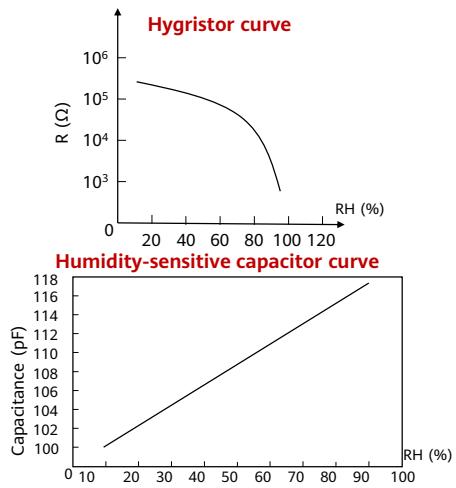
Thermistor temperature curves



Thermocouple circuit

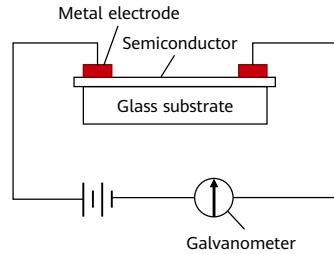
Humidity Sensors

- Humidity sensors can sense environmental humidity changes and convert non-electrical quantities into electrical ones by detecting physical or chemical changes.
- They can be classified as **resistive** or **capacitive** sensors. A hygristor uses a humidity-sensitive film on a substrate to measure humidity. When water vapor in the air attaches to the film, the element resistivity and resistance change.
- A humidity-sensitive capacitor has a thin polymer film. When the ambient humidity changes, the dielectric constant of the humidity-sensitive capacitor changes, and its capacitance changes accordingly. The changed capacitance is directly proportional to the relative humidity.



Photoelectric Sensors

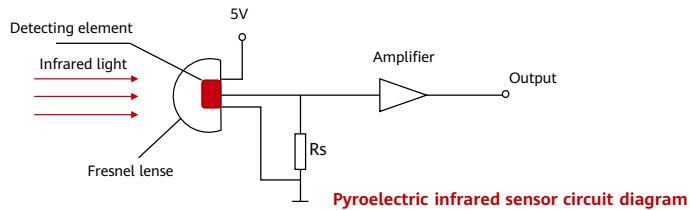
- Photosensitive sensors use photosensitive elements to convert optical signals into electrical signals. Their physical basis is the **photoelectric effect**. Many electrical features of semiconductor materials change when exposed to light. There are three types of photoelectric effects: external, internal, and photovoltaic.
- Photoresistors are based on the **internal photoelectric effect**. Electrons break the bonding and become free by absorbing the photon energy, causing a material conductivity change. As a result, the resistor resistance varies with the light intensity.
- Photoresistors are sensitive to light. More intense light reduces resistance. When there is no light, the resistance is high.



Photoresistor structure

Infrared Sensors

- Infrared sensors can measure infrared radiation emitted by monitored objects. Any object which has a temperature emits infrared radiation. Infrared sensors do not physically contact measured objects, so there is no friction. They are highly sensitive and respond quickly.
- An infrared sensor consists of an optical system, a detecting element, and a conversion circuit. Optical systems can be classified based on their structure as either transmission or reflection system. Detecting elements can be classified as **thermosensitive** or **photoelectric** elements, depending on how they work.
 - The thermistor resistance changes when there is more heat (infrared radiation). Based on this principle, a thermosensitive element converts infrared signals into electric signals using a conversion circuit.
 - A photoelectric element converts received infrared signals into electrical signals based on the photoelectric effect.



Pyroelectric infrared sensor circuit diagram

Contents

1. Sensing Technologies
2. **Tag Identification Technologies**
 - Barcode Identification
 - RFID
3. Location Data Collection Technologies

Barcode Identification Technologies

- IoT implements global goods tracking and information sharing, greatly improving management and operations efficiency. It affects each process of global supply chains, involving manufacturing, warehousing, logistics, goods tracking, retail, and public service industries. Automatic identification technologies using barcodes and electronic tags are a cornerstone of IoT. They enable objects to speak.
- One-dimensional (1D) barcodes are based on laser identification and two-dimensional (2D) barcodes are based on image identification. They are two types of **barcode information collection** technologies. Radio frequency identification (RFID) is a **radio information collection** technology.

1D Barcodes

- Barcodes are a type of automatic identification technology. They remove bottlenecks in data recording and collection and are widely used for materials and production management.
- A barcode is an image consisting of **black lines (bars)** and **white lines (spaces)** with different widths arranged based on special encoding rules. It contains information based on these highly contrasting parallel lines.
- Barcode scanners identify the light reflected by barcodes based on the photoelectric effect. The spaces reflect **strong signals**, and the bars reflect **weak signals**.

Barcode Type	Description
EAN-13	Used for products globally. The code contains 13 digits and supports numbers 0-9. Notches are included.
EAN-8	Used for products globally. The code contains 8 digits and supports numbers 0-9. Notches are included.
UPC-A	Used for products in the US and Canada. The code contains 12 digits and supports numbers 0-9. Notches are included.
EAN-128	All ASCII characters are supported. Its length is theoretically unlimited. It is the standard for the EAN/UCC system and is used for logistics units.
Code 39	Supports uppercase letters A-Z, numbers 0-9, and 43 characters. It can encode variable-length data and is mainly used for logistics tracking and production line workflow.
Code 128	128 ASCII characters are supported. Its length is theoretically unlimited. Code 128 has code sets A, B, and C.



Code-128 example



2D Barcodes

- 2D barcodes are graphical images that store information horizontally and vertically using black and white cells. The information density of a 2D barcode is several times that of a 1D barcode, so they can hold much more information.
- 2D barcodes can be classified by their encoding method as either **stacked** or **matrix** barcodes.



Stacked 2D barcodes are also called stack, multi-row, or multi-layer barcodes, in which two or more rows of 1D barcodes are stacked. Common stacked 2D barcodes include PDF417 and MicroPDF417.



Matrix 2D barcodes are checkerboard-like rectangular patterns that encode information as binary code using black and white pixels. Black dots indicate 1s and white spaces indicate 0s. The most popular matrix 2D barcodes include QR Codes, Data Matrix, MaxiCode, and Han Xin Code.

- 1D barcodes contain less information. As technology advanced, we developed 2D barcodes. They can represent more information in a smaller space.
- 2D barcodes not only identify a group of items but also describe a specific item. This is a great step forwards for barcode technologies. 2D barcodes also provide alignment patterns and error correction. Alignment patterns allow 2D barcodes to be scanned in any direction. Error correction enables correct identification and restoration of partial or damaged barcodes.
- 2D barcodes feature large storage capacity, high security, high traceability, strong resistance to damage, high availability, and cost-effectiveness. They can be used in forms, for information security, for tracking codes, permits and other certificates, stocktaking, and document backup.

Barcode Comparison

Item	1D Barcode	2D Barcode
Space utilization	Low (horizontal only)	High (horizontal & vertical)
Capacity	About 30 characters	1,850 characters or more
Information content	Letters and numbers	Letters, numbers, symbols, and figures
Expression method	1D barcodes cannot present product information directly. They need to connect to a database.	2D barcodes can present product information directly. They do not need to connect to a database.
Error correction	Not supported. Damaged barcodes cannot be read.	Supported. 2D barcodes can be read from any direction. At the highest error correction level, complete information can still be read from a 2D barcode that is only 50% complete.
Code systems	Code 128, EAN, ISBN, etc.	QR codes, PDF417, etc.
Application scenarios	Workflow management, transportation, warehousing, postal services, etc.	Mobile payments, website links, account login, information reading, e-commerce, and anti-counterfeiting, etc.

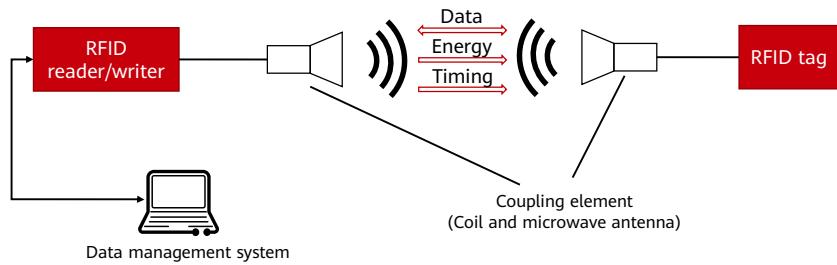
- 1D barcodes connect to databases where product data is managed. They are necessary for large retailers because they can improve stocktaking accuracy and efficiency.

Contents

1. Sensing Technologies
2. **Tag Identification Technologies**
 - Barcode Identification
 - **RFID**
3. Location Data Collection Technologies

RFID Overview

- RFID is a **contactless** automatic identification technology that uses radio waves for two-way data communications. Recording media (electronic tags or radio frequency cards) can be read or written to identify targets and exchange data.
- An RFID system consists of a reader, an RFID tag, and a data management system.



- A reader reads information from or writes information into tags. It can be a read/write device depending on the structure and technology used. It is an information control and processing center of the RFID system. When the RFID system is working, the reader emits radio frequency energy within an area to form an electromagnetic field, and the size of the area depends on the transmit power. When the tag in the area covered by the reader is triggered, data stored in the tag is sent or modified based on reader instructions. A reader can communicate with computer networks using APIs. A reader usually consists of a transceiver antenna, frequency generator, phase-locked loop, modulation circuit, microprocessor, memory, demodulation circuit, and peripheral interface.
- An electronic tag consists of a transceiver antenna, an AC or DC circuit, a demodulation circuit, a logic control circuit, a memory, and a modulation circuit.

RFID Application Scenarios



Logistics
Package tracking, auto information collection, warehousing, postal services

Anti-counterfeiting
Valuables and tickets

Information management
Archive digitization, collection information reading, and storage location management

Transportation
Taxi management, bus hub management, and railway car identification

Asset management
Valuables, large number of similar materials, dangerous items

Search
Archive information and storage location

Identification
e-Passports, student cards

Food management
Fruits, vegetables, fresh foods

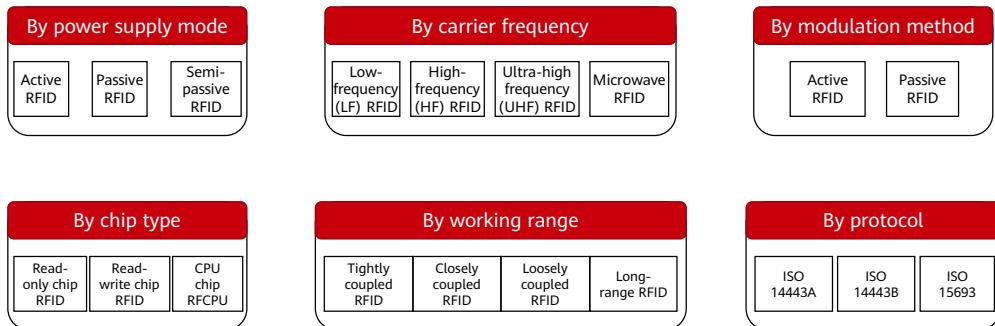
Security control
Real-time monitoring and alarm reporting for physical archives



- Source of image 1:
<http://3ms.huawei.com/km/static/image/detail.html?fid=60324>
- Source of image 2:
<http://3ms.huawei.com/km/static/image/detail.html?fid=57509>

RFID Classification

- RFID tags can be classified into the following categories based on their power supply, carrier frequency, modulation method, working range, and chip type:



- Passive RFID

- Passive RFID is the earliest, most mature, and most widely used RFID system. A passive RFID tag receives microwave signals from an RFID reader and obtains energy through an electromagnetic induction coil to supply power to itself for a short time, so as to complete information exchange. As a passive RFID tag does not have the power supply system, its size can be reduced to cubic centimeters. It is cost effective and has a simple structure, low failure rate, and long service life. However, the effective identification range of passive RFID tags is short, so they are usually used for short-range identification. Passive RFID works at low frequency bands, such as 125 kHz and 13.56 MHz. Its typical applications include bus cards, ID cards, and dining cards.

- Active RFID

- Active RFID is an emerging technology, but it has been widely used and become a must in various fields, for example, in the electronic toll collection system for expressways. An active RFID has a power source and sends signals to an RFID reader actively. It has a large size, long transmission range, and high transmission speed. A typical active RFID tag can connect to an RFID reader 100 meters away, with a read rate of up to 1,700 reads per second. Active RFID works at high frequency bands, such as 900 MHz, 2.45 GHz, and 5.8 GHz. It can identify multiple tags at a time. As active RFID supports long-range transmission and is highly efficient, it is indispensable in scenarios that require high performance and long-range transmission.

RFID Advantages and Disadvantages

Advantages	Disadvantages
<ul style="list-style-type: none">• Waterproof, anti-magnetic, and heat-resistant• Good data storage capacity• Long service life• High efficiency and simultaneous identification of multiple objects	<ul style="list-style-type: none">• Not mature enough• High cost• Not secure enough• Inconsistent technical standards



- Advantages:
 - RFID can update existing information more conveniently while reducing workforce, material resources, and financial resources.
 - RFID uses computers to store information. Its maximum capacity is several megabytes, so it can store a large amount of information. This ensures smooth work.
 - RFID has a long service life. It can be reused with careful handling.
 - RFID makes information processing more convenient and enables simultaneous identification of multiple objects, greatly improving efficiency.
- Disadvantages:
 - It is not mature enough. RFID is an emerging technology, which is not very mature. As UHF waves can bounce off metal objects and liquids, it is difficult to use UHF RFID tags for metal and liquid products.
 - Its cost is high. RFID tags are dozens of times more expensive than common barcodes. Using many RFID tags will cause a high cost. This has made them less popular to some degree.
 - It is not secure enough. Major security issues faced by RFID are unauthorized reading and tampering of RFID tag information.

- Its technical standards are inconsistent.

Comparison of Automatic Identification Technologies

Item	Barcode	RFID
Reading methods	A scanner is required for reading the data.	Data can be read without a light source or even through packaging.
Reading rate	Only one barcode can be read at a time.	Multiple tags can be identified at once.
Data capacity	Small. A few dozen to a few thousand characters are allowed.	Large. Dozens of kilobytes are allowed.
Effective identification range (onsite)	Short	Long (active RFID)
Service life	Short. Information cannot be collected from incomplete or damaged barcodes.	Long. It can be used in polluted and radioactive environments with dusts and oil stains based on wireless communications.
Flexibility	Low. Tag data cannot be modified after being printed.	High. Tag data can be dynamically changed.
Communications mode	Supports only one-way communications. It cannot provide instant information or Internet data transmission.	Supports real-time communications. As long as an object with an RFID tag is within the reader detection range, its location can be dynamically tracked.
Cost	Low	High (if used in large quantities)

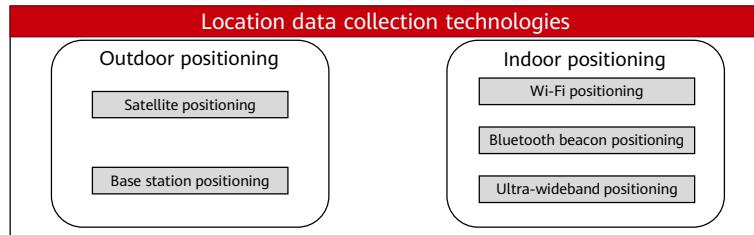
- Why application scenarios are not compared?
 - Barcodes and RFID have similar application scenarios. When a company chooses from the two technologies, costs are a key factor. The cost of barcodes includes the costs of paper and ink. An active RFID tag with a memory chip costs more than USD2, and a passive RFID tag costs more than USD1. From the technical perspective, RFID tags have more advantages than barcodes.

Contents

1. Sensing Technologies
2. Tag Identification Technologies
- 3. Location Data Collection Technologies**
 - Satellite Positioning
 - Base Station Positioning
 - Indoor Positioning

Overview

- Various positioning technologies can be used to send geographical locations to users, communications systems, or third-party positioning clients.
- In the IoT architecture, location awareness or positioning technologies are indispensable at the sensing layer because they provide location data for the entire IoT system. For applications, location services will be used in many IoT scenarios for service differentiation.
- Many companies have replaced their traditional systems with geolocation-based asset management solutions to locate tangible assets such as devices, products, and vehicles.
- Positioning technologies are very familiar to us. They have seen widespread adoption in navigation and aerospace as well as location search and traffic management.
- Positioning technologies can be classified as **indoor positioning** or **outdoor positioning** technologies based on application scenarios. Different positioning technologies are used for different scenarios and to meet different requirements.



Satellite Positioning

- Satellite positioning uses artificial earth satellites for precise positioning. It provides military and civilian users in air, sea, ground, or space with high-precision position, velocity, and time data in all weather conditions.
- There are four main global navigation satellite systems (GNSS): BeiDou Navigation Satellite System (BDS), Global Positioning System (GPS), Galileo Satellite navigation System (Galileo), Global Navigation Satellite System (GLONASS).
- Satellite positioning is commonly used for vehicle navigation, atmospheric physics observation, sailing route planning, missile guidance, and more.

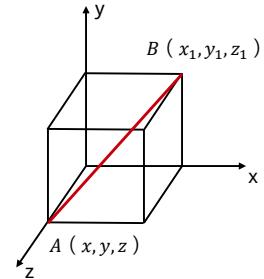


 HUAWEI

- Image source: <http://3ms.huawei.com/km/static/image/detail.html?fid=61780>

Working Principles of Satellite Positioning

- Before calculation, you need to obtain each satellite's accurate time t_0 and locations x_0 , y_0 , and z_0 . With these four elements, you can calculate the distance between a satellite and the target object using the 3D Pythagorean theorem formula.
- In the figure on the right, the distance s between A and B is calculated as follows:
$$s = \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2}$$
- Electromagnetic waves travel at the speed of light, c . If data transmitted by a satellite at t_0 is received by the target object at t , the transmission time is $t - t_0$. Therefore, the preceding formula can be written as follows:
$$s = c * (t - t_0) = \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2}$$
- There are three unknown numbers: x , y , and z . Theoretically, as long as signals of three satellites are received, you can calculate the remaining values using this equation. Devices often use crystal oscillators, which have limited precision, as clocks, so there is a concept called **clock difference**. In this case, another unknown number is introduced. This requires four satellites to be locked at the same time to locate the target object.



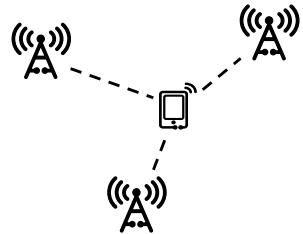
- The preceding methods are for ideal conditions. The real calculation process is much more complex. Researchers proposed many error correction methods. For example, when more than four satellites are locked, the satellites are divided into multiple groups based on their constellations, and each group contains four satellites. The group with the minimum error is used for decoding and positioning.

Contents

1. Sensing Technologies
2. Tag Identification Technologies
- 3. Location Data Collection Technologies**
 - Satellite Positioning
 - Base Station Positioning
 - Indoor Positioning

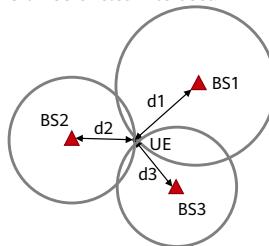
Base Station Positioning

- Base station positioning uses mobile telecommunication base stations to locate end users of carrier networks. The positioning service based on base stations is called location-based service (LBS).
- Although satellite positioning can provide precise positioning for positioning apps to locate terminals, only base station positioning works where satellite signals are unavailable, for example, in tunnels, underground parking lots, and roads under viaducts. Base station positioning and satellite positioning need to be used together to provide services in different outdoor scenarios.



Technical Principles of Base Station Positioning

- Base station positioning methods can be network-based or terminal-based. Each method includes many different positioning principles. Of these principles, trilateration is most commonly used.
- Trilateration:
 - The signal becomes weaker when it is farther away from a base station, so the signal strength received by a mobile phone can be used to estimate the distance between the mobile phone and the base station.
 - A base station is unique within a mobile network and has a unique geographical location. If you have three base stations, the distances between the mobile phone and each base station can be obtained. Then you just draw three circles with a base station as the center and the corresponding distance as the radius, and the phone will be at the point where the three circles intersect.



- Network-based positioning
 - Technology used in 2G, 3G, and 4G networks: time of arrival (TOA) positioning
 - Technology used in 5G networks: angle of arrival (AOA) positioning under massive multiple input multiple output (MIMO)
- Terminal-based positioning
 - Base ID positioning
 - Trilateration positioning
 - Positioning based on scenario analysis

Comparison of Outdoor Positioning Technologies

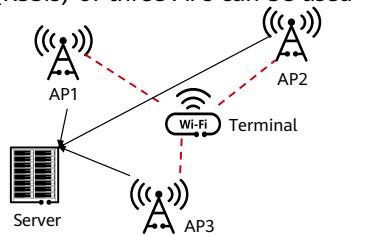
Item	Satellite Positioning	Base Station Positioning
Precision	High precision (5–10 m)	Low precision (20–200 m in urban areas and 1,000–2,000 m in suburban areas)
Power consumption	High. Terminals need to provide high-voltage power for GPS modules.	Low. Only base stations need to collect data.
Advantages	High-precision outdoor positioning and wide coverage	Fast positioning; Low power consumption; Unaffected by weather or high-rise buildings and not location dependent.
Disadvantages	High cost; High power consumption; The antenna of the GPS system must be placed outdoors and under the open sky. Otherwise, positioning will fail. It is affected by weather and location dependent.	Low precision The terminal has a registered SIM card and can receive signals from at least three base stations.

Contents

1. Sensing Technologies
2. Tag Identification Technologies
- 3. Location Data Collection Technologies**
 - Satellite Positioning
 - Base Station Positioning
 - Indoor Positioning

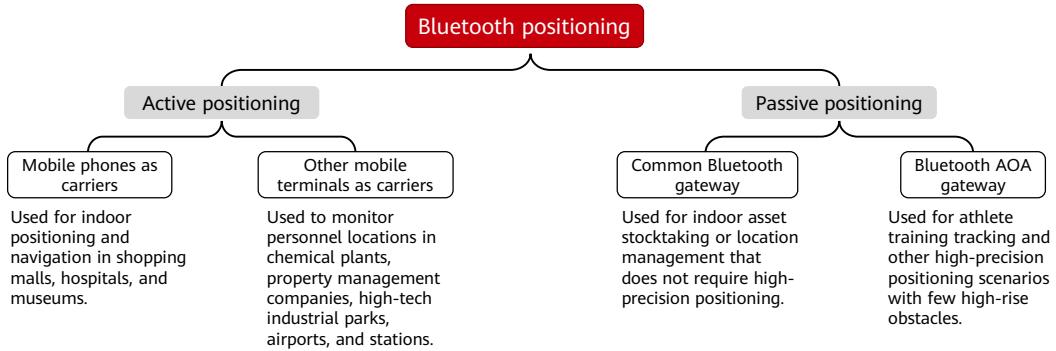
Wi-Fi Positioning

- Wi-Fi positioning is the most popular positioning method. It can be used for large-scale positioning, detection, and tracking in places with Wi-Fi networks. Network node self-positioning is the foundation of Wi-Fi positioning.
- Wi-Fi positioning uses the k-nearest neighbor (KNN) algorithm. The hotspot or base station that the terminal is closest to is used as the location of the terminal. If there are multiple access points (APs), the received signal strength indicators (RSSIs) of three APs can be used for trilateration positioning.



Bluetooth Beacon Positioning

- As Bluetooth is a very commonplace technology, many projects use Bluetooth beacons for positioning.
- Bluetooth beacon positioning uses a principle and approach similar to Wi-Fi positioning but delivers higher precision.



Ultra-Wideband Positioning

- Ultra-wideband (UWB) positioning uses an **anchor node** (master device) and **bridge nodes** (at least two slave devices) and a **blind node** (tag to be located). The anchor and bridge nodes are deployed at known locations. Then, triangulation or fingerprinting method is used for positioning.
- UWB signals can easily penetrate common obstacles to obtain individual or object location data. They can also be used for precise indoor positioning and navigation in many scenarios, including tunnels, prisons, chemical plants, factories, coal mines, construction sites, power plants, nursing homes, exhibition halls, vehicles, equipment rooms, and airports.

Comparison of Indoor Positioning Technologies

Item	Wi-Fi Positioning	Bluetooth Beacon Positioning	Ultra-Wideband Positioning
Frequency range	2.4 GHz and 5 GHz	2.4 GHz	3.1–10.6 GHz
Transmission rate	Up to 1 Gbit/s	Up to 2 Mbit/s	Up to 500 Mbit/s
Latency	3–5s	3–5s	Less than 1 ms
Transmission range	Preferably within 50 m, but up to 100 m	Preferably within 25 m, but up to 100 m	Preferably within 50 m, but up to 200 m
Positioning precision	Less than 10 m	Bluetooth RSSI: 5–10 m Bluetooth AOA: 0.5–1 m	Less than 50 cm
Security	High	High	Relatively high
Penetration	Strong	Weak	Strong
Anti-interference	Strong	Weak	Strong
Power consumption	High	Relatively low	Medium
Cost	High	Low	High

Quiz

1. (Multiple-answer question) Which of the following types can sensors be classified as based on their output signals? ()
 - A. Binary sensors
 - B. Physical sensors
 - C. Digital sensors
 - D. Analog sensors
2. (Multiple-answer question) Which of the following are indoor positioning technologies? ()
 - A. Satellite positioning
 - B. Ultra-wideband positioning
 - C. Base station positioning
 - D. Wi-Fi positioning

- Answers:

- ACD
 - BD

Summary

- Having completed this section, you should now understand how sensors collect data, how they are classified, what the working principles of common sensors are, the classification and working principles of automatic identification technologies, and how indoor and outdoor positioning technologies work.

Acronyms or Abbreviations

- ADC: Analog-to-Digital Converter
- DAC: Digital-to-Analog Converter
- LBS: Location Based Service
- NB-IoT: Narrowband Internet of Things
- RFID: Radio Frequency Identification
- UWB: Ultra Wideband

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and
organization for a fully connected,
intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive
statements including, without limitation, statements regarding
the future financial and operating results, future product
portfolio, new technology, etc. There are a number of factors
that could cause actual results and developments to differ
materially from those expressed or implied in the predictive
statements. Therefore, such information is provided for reference
purpose only and constitutes neither an offer nor an acceptance.
Huawei may change the information at any time without notice.



MCU Basics



Foreword

- A microcontroller unit (MCU) is the main component at the sensing layer of the Internet of Things (IoT). Each sensing layer device has at least one MCU.
- MCUs are the core of an IoT system. With rapid development of electronic IT, MCUs are everywhere: wearables, household appliances, industrial instruments, and smart home/transportation/city.

Objectives

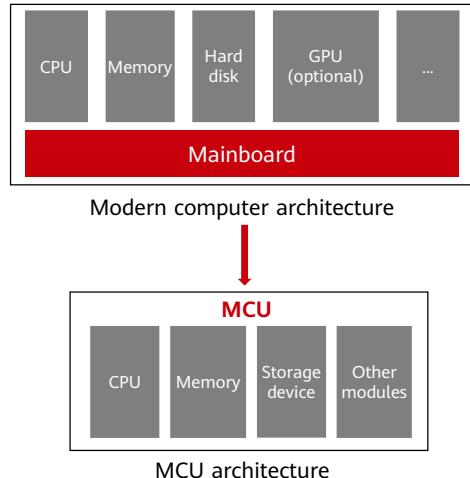
- Upon completion of this course, you will understand:
 - MCU development history
 - MCU architecture
 - Working principle of MCUs
 - Working principle of common MCU peripherals
 - Common MCU bus communication protocol

Contents

- 1. MCU Overview**
2. MCU Architecture and Working Principle
3. MCU Peripherals

From the PC to the MCU

- If you have ever assembled a computer (PC) by yourself, you would know its components:
 - Central processing unit (CPU), memory, hard disk, graphics processing unit (GPU), and mainboard.
- Although computers have evolved for a long time, their architecture remains unchanged: **CPU + memory + storage device**. MCU architecture is the same as PC architecture. However, the CPU + memory + storage device are now **integrated** into one chip, the MCU.
- MCUs greatly simplify the electronics, and enable hardware engineers to design them with less experience. Currently, an MCU performs just as well as earlier computers.



- Why use an integrated processor?
 - Although some components (such as I/O) perform worse after they are integrated, many other components perform significantly better because data transmission between processors is instant. In addition, an integrated processor has fewer components on the circuit board, so on-chip debugging is easier, board-level faults are fewer, the power consumption is much lower, the size is smaller, and the price is lower. However, components are integrated into a processor instead of a circuit board, so the flexibility of adding, changing, or removing some functions of an integrated processor is relatively poor.

MCU Development History

- Four stages: exploration, upgrade, full-fledged, diversification

Exploration Stage (1976–1978)

- Represented by MCS-48 series MCUs. Their launch started the exploration in the industrial control field. The manufacturing process was backward, the integration technology was immature, and dual chips were used. The MCU technology was being developed and innovated.

Upgrade Stage (1978–1982)

- Functional components such as the CPU, parallel interface, timer, RAM, and ROM were integrated into one chip, but the MCU had low performance, few types, and few application scenarios.

Full-fledged Stage (1982–1990)

- The 8-bit MCU was fully functional. The storage capability, addressing range, interrupt source, parallel I/O port, and timing/counter were improved. In addition, the full-duplex serial communication interface was integrated into the MCU. The instruction system supported multiplication and division, bit operation, and bit instructions.

Diversification Stage (since 1990)

- The MCU has been used in various fields. The 8/16/32/64-bit universal MCU with high speed, large addressing range, and high compute has been developed. The 16-bit MCU and 8-bit high-performance MCU are more popular. The MCU gradually improves integration capability, strengthens functions, and speeds up compute. In addition, it allows users to use special languages for industrial control.

- In the first three stages, an MCU functions as some special hardware for automatic control, for example, collecting temperature data cyclically and reporting an alarm if the temperature reaches the threshold. Generally, one while loop statement is enough to achieve the function. Some run a small OS on an MCU to make better use of it.
- With the MCU being widely used in mobile phones and tablets, the MCU becomes an application processing unit (APU) after integrating with the CPU, RAM, and Flash and connecting to an external DDR storage or a larger flash storage. In addition, functional modules such as the DSP are integrated into the chip to process audio signals. Compared with the former application, the latter uses the MCU as a device designed for general functions.
- In one sentence, a non-computer device with computing capabilities is an embedded device.

MCU and IoT

- We have learnt that the MCU predates the IoT. Before the concept of IoT, the function of the MCU and the program running on it were very simple: collecting data and reporting an alarm if the data reaches the threshold to achieve automatic control.
- After the concept of IoT, people began to realize the significance of data networking and data linkage across devices, and started making many landmark applications and products.
- Multiple MCUs and communication modules form a simple IoT system. Growing IoT also grows MCU applications.



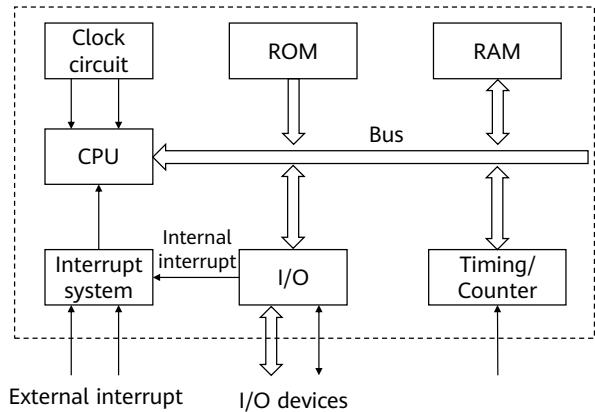
 HUAWEI

- Image source: <http://3ms.huawei.com/km/static/image/detail.html?fid=61636>

Contents

1. MCU Overview
2. **MCU Architecture and Working Principle**
 - MCU Architecture
 - MCU Working Principle
3. MCU Peripherals

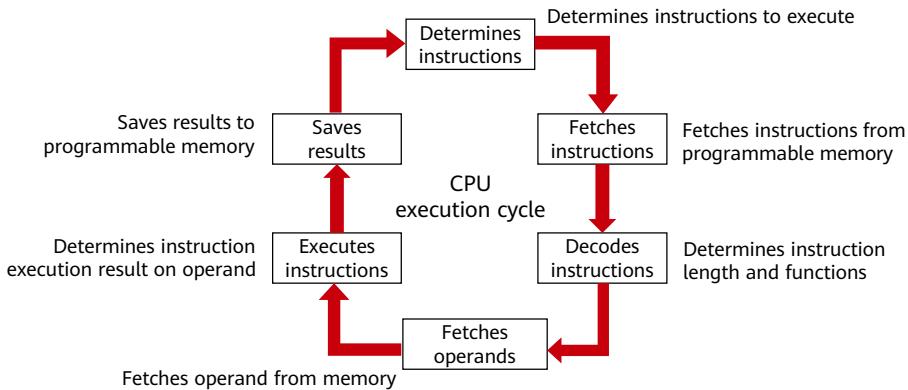
MCU Architecture



- Generally, the complexity of the main processor determines whether it is a microprocessor or a microcontroller. A microprocessor integrates only a few memory and I/O interfaces, while a microcontroller integrates most of the system memories and I/O interfaces into a chip. However, these traditional definitions do not fit modern processor design. For example, microprocessors can integrate more components.
- An interruption is a signal triggered by an event during the execution of an instruction stream by the main processor. This means that an interruption can be asynchronously triggered by events such as external hardware devices, resets, and power failures, or synchronously triggered by instruction-related activities such as system calls or invalid instructions. Once such a signal is triggered, the main processor stops executing the current instruction stream and starts to process the interruption.
- Interruption signals are mainly from software, internal hardware, and external hardware.

CPU

- A CPU is a core component of a computer. A multi-core CPU has multiple CPU cores, which are actual data processing units. A CPU **fetches**, **decodes**, and **executes** instructions.



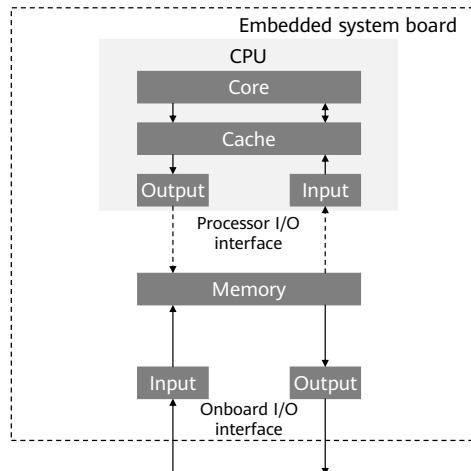
- CPU and system clock
 - The execution of a processor is ultimately synchronized by an external system or primary clock located on the mainboard. The primary clock consists of an oscillator and some other components. It generates a on/off pulse signal sequence at a fixed frequency. The control unit and some other components on the embedded system board work following the primary clock. The frequency of primary clocks varies, but the frequency must meet the requirements of the slowest component on the circuit board. In some cases, the primary clock can use clock signals at different frequencies generated by other components on the circuit board.

Memory

- A CPU needs to fetch data and instructions to be processed from memory.
- Common memory includes **read-only memory (ROM)** and **random access memory (RAM)**.
 - The on-chip ROM is the memory integrated into a processor. It stores data and instructions even when the system runs out of power, so ROM is generally used to store programs.
 - RAM is usually called main memory. All its addresses can be directly accessed (at random, not at a sequence from a specific start point). The data in RAM can be modified multiple times. Unlike ROM, RAM data will be erased if the system runs out of power, so RAM is volatile.

Input/Output Interface

- The input/output interfaces are processor I/O interfaces and onboard I/O interfaces.
 - Processor I/O interfaces mainly transfer data across other components of the processor, memory on the circuit board, and the external I/O interface of the processor.
 - The onboard I/O interfaces mainly exchange information between a system board and I/O devices connected to the embedded system.
- For processor I/O interfaces, except integrated circuit (IC) interfaces, basic features are essentially the same as when directly deployed on a circuit board.



- I/O can be further divided into input devices, output devices, or input/output devices. An output device receives data from onboard I/O components and displays the data in a specific way, such as printing on a paper, storing on a disk, displaying on a screen, or using blinking LEDs. An input device transmits data to an onboard I/O component, such as a mouse, a keyboard, or a remote control device. An I/O device may input or output data, for example, a network device may exchange data with the Internet.

Bus

- This mainly refers to the processor bus. A processor bus interconnects the main internal components (such as the CPU, memory, and I/O interface) of the processor, and transmits signals among them.
- A bus can be classified by what it transmits: **data bus, address bus, control bus**.
 - Data bus: a **bidirectional** bus used to transfer data between a processor, memory, and an I/O interface.
 - Address bus: used to select the memory location the processor reads from or writes to. The data flow of the address bus is **unidirectional**, from the processor to the memory and I/O interface.
 - Control bus: consists of multiple independent control/signaling lines. A typical signal includes a read/write line indicating **a data traveling direction with a data bus**.

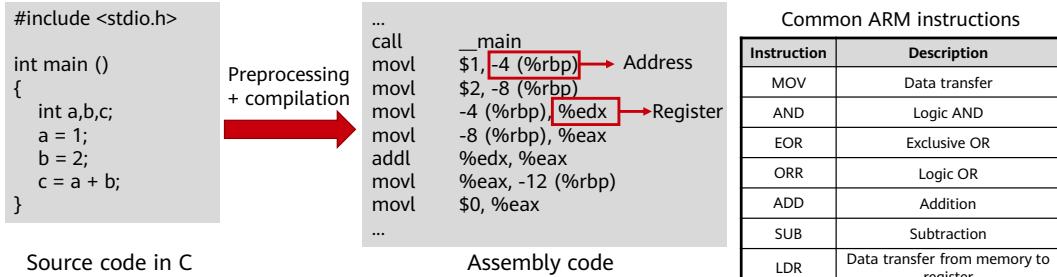
- In addition to the processor bus, there is the CPU bus and onboard bus:
 - A CPU bus connects other CPU components, such as the ALU, register, and CU.
 - An onboard bus connects all other important components (such as the main processor, I/O component, and memory) on the embedded mainboard. The onboard buses are classified into system buses, backplane buses, and I/O buses.

Contents

1. MCU Overview
2. **MCU Architecture and Working Principle**
 - MCU Architecture
 - **MCU Working Principle**
3. MCU Peripherals

Instruction Sets

- A computer instruction directs the work of a machine. A program is a series of instructions in a certain sequence. Program execution is the working process of a computer. An instruction set calculates and controls a computer system in a CPU. Each new type of CPU is designed with a series of instruction systems that match other hardware circuits. How advanced an instruction set depends on CPU performance, making it an important indicator of the CPU performance. Each CPU is designed with a series of instruction systems that match its hardware circuit. Instruction strength is also an important CPU metric. An instruction set is one of the most effective tools to improve microprocessor efficiency.



Source code in C

Assembly code

Common ARM instructions



15 Huawei Confidential

- l, w, and b are qualifiers with operation attributes in AT&T Assembly Language, and l indicates long (4 bytes).
- EAX is a 32-bit register, and AX indicates a 16-bit register.

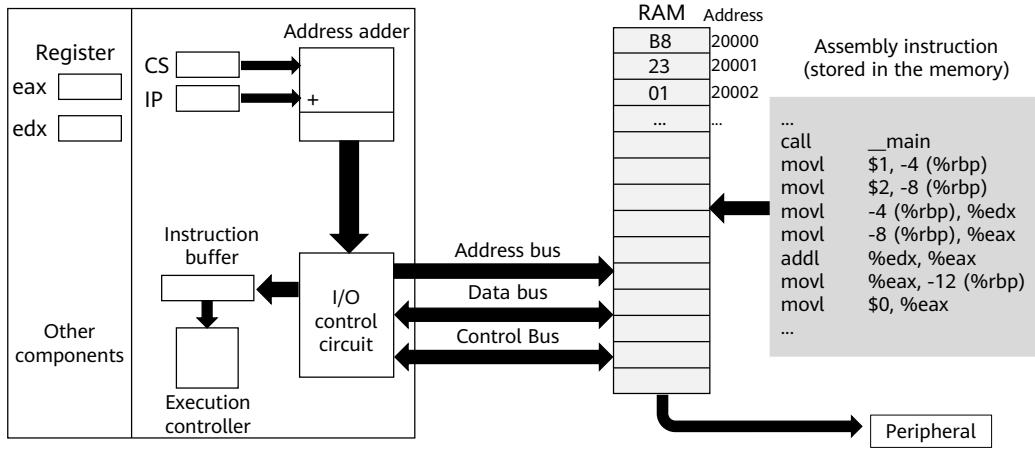
RISC & CISC

- CPU instruction sets are classified by architecture: complex instruction set computer (CISC) or reduced instruction set computer (RISC)

Item	CISC	RISC
Instruction system	Many instructions	Few instructions (usually less than 100)
Execution duration	Long, especially instructions such as copying the whole content of memory or copying the contents of multiple registers to a memory.	Short
Code length	Variable (max 15 bytes)	Fixed (usually 4 bytes)
Addressing mode	Various	Simple
Operation	Arithmetic and logical operations can be performed on memory and registers.	Arithmetic and logic operations can be performed only on registers.
Compilation	Efficient object code programs generated using optimized compilers are not supported.	Efficient object code programs generated using optimized compilers are supported.

- CISC has been used since the birth of computers. Early desktop software is designed using CISC, which is still applied now, such as, the popular x86 architecture. Microprocessor vendors, such as Intel, AMD, TI, and VIA, have been developing CISC microprocessors. In a CISC microprocessor, instructions of a program and operations in each instruction are executed in sequence. Sequential execution is easy to control, but it is slow and parts of a computer cannot be fully used. CISC servers mainly use the IA-32 architecture, and most of them are mid-range and low-end servers.
- RISC is a microprocessor that can execute instructions of a few types. It originated from the MIPS machine (that is, RISC machine) in the 1980s. Microprocessors used in a RISC machine are RISC processors. In this way, it can perform operations faster (millions of instructions per second). Transistors and circuit elements required for different instruction types vary, so the larger the instruction set, the more complex the microprocessor and the slower the execution.

MCU Working Principle



17 Huawei Confidential



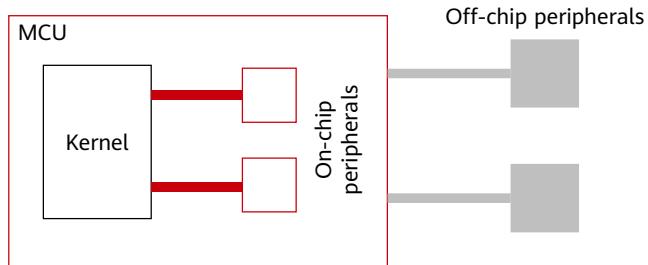
- The initial values of CS and IP are set before the CPU is delivered.
 - The memory reads the program from the Flash.
 - The CPU obtains the instructions from the specified position in the memory based on the address calculated by the CS and IP registers.
 - The instructions are executed in the execution controller, and the intermediate result is stored in an internal register (such as eax and edx).
 - After the instructions are executed, the final calculation result is output to the memory.
 - The memory transmits the data to the peripheral through the buses to display the result.

Contents

1. MCU Overview
2. MCU Architecture and Working Principle
- 3. MCU Peripherals**
 - MCU Peripheral Overview
 - Serial Port Bus Overview

MCU Peripherals

- An MCU peripheral is an MCU external functional module. Examples: analog/digital (A/D) converter, pulse width modulation (PWM), and timer. They are controlled through the I/O, SPI, or I2C bus.
- MCU peripherals can be on-chip and off-chip.



- In the early stage, the IC integration technique is not developed, so many peripherals are used, such as PWM, ADC, and CAN, and DSP chips. Even now, there are still independent ADC chips, such as ADS8364. For now, the PWM and ADC have been integrated into the DSP chip. To differ from peripherals of other types, peripherals integrated into a chip are called on-chip peripherals, for example, devices which are on the DSP microprocessor but do not belong to the DSP microprocessor are on-chip peripherals.
- Take STM32 MCU as an example, STM32 MCU is developed by ST based on Cortex-Mx. When ST uses the Arm kernel to design the chip, devices (such as the serial port, Flash, I2C, SPI, ADC, and DAC) that are added to the chip but are independent of the kernel are peripherals. These peripherals are on the chip, so they are called on-chip peripherals. When we use the STM32 chip to develop an application system, external devices (such as the display screen and SD card) that are not on the chip are off-chip peripherals.

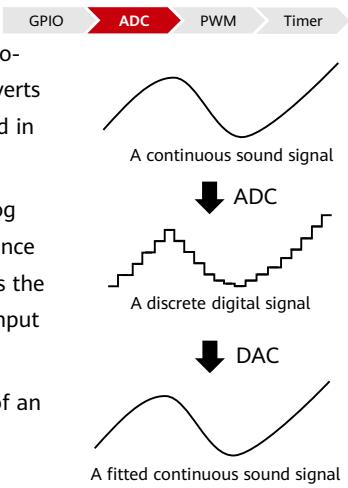
GPIO



- General-purpose input/output (GPIO) refers to a set of pins on a computer mainboard or an additional card. These pins send or receive electrical signals, but they are not designed for any particular purpose. That is why they are called general-purpose I/Os.
- A GPIO controller manages all GPIO pins by group. Each group of GPIO pins is associated with one or more registers. The GPIO controller **manages the pins by reading data from and writing data to the registers**. Pins can output high and low levels or read the pin status (high or low level).
- You can use the pins to exchange data with hardware (such as UART), control hardware (such as LED and buzzer), and read hardware working status signals (such as interrupt signals).

ADC

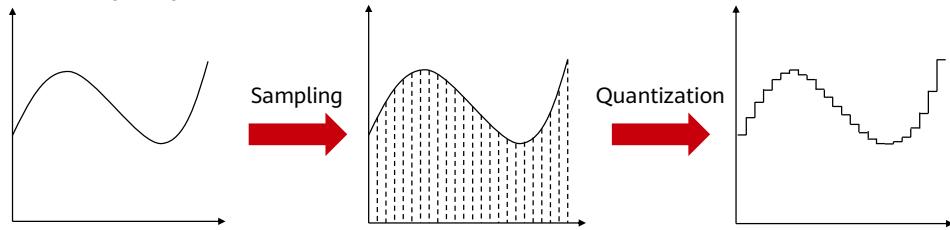
- The digital virtual world makes our life convenient, and the analog-to-digital converter (ADC) bridges the virtual and real worlds. ADC converts **analog** signals (such as temperature, pressure, and sound) generated in the real world into **digital** signals that are easier to process.
- A common ADC converts the analog quantity processed by the analog comparator into binary discrete signals, so any ADC requires a reference analog quantity as the conversion standard. The common standard is the maximum convertible signal size. The output digital quantity is the input signal size relative to the reference signal size.
- The function of a digital-to-analog converter (DAC) is the opposite of an ADC. An ADC converts analog signals to digital signals, while a DAC converts digital signals in an MCU to analog signals.



ADC Working Principles

GPIO → ADC → PWM → Timer

- Analog-to-digital conversion includes **sampling**, **quantization**, and **encoding**.
 - Sampling: samples values of a continuous input signal at discrete intervals in time.
 - Quantization: replaces a continuous analog signal with a limited number of discrete values at a certain interval.
 - Encoding: represents quantized values using binary values and then converts those values into binary or multi-level digital signals.



PWM



- **Pulse Width Modulation (PWM)** modulates the width of a series of pulses to obtain the required waveform (including the shape and amplitude) and performs digital coding on the analog signal level by adjusting the duty cycle to changes of signals and energy. The duty cycle **is the percentage of the time during which a signal or system has high voltage**.
For example, the duty cycle of a real square wave is 50%.



Square wave with 50% duty cycle

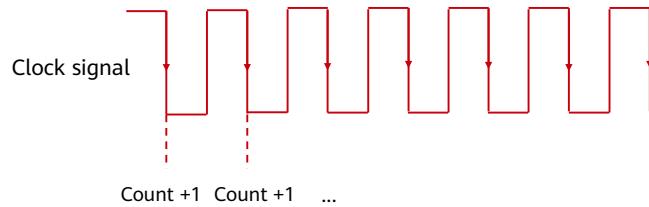
$$\text{Duty Cycle} = \frac{T_{HL}}{T} \times 100\%$$

Square wave with 25% duty cycle

Timer

GPIO > ADC > PWM > **Timer**

- This important MCU module is widely used in the detection and control fields. Developers often use a timer for time-related operations.
- An MCU timer works as a counter of the number of times a pulse reaches a rising or falling edge of a square wave signal.



Contents

1. MCU Overview
2. MCU Architecture and Working Principle
- 3. MCU Peripherals**
 - MCU Peripheral Overview
 - Serial Port Bus Overview

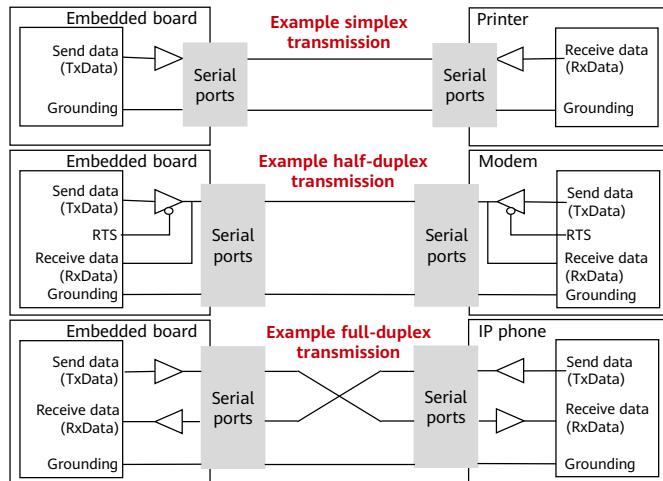
Serial I/O

- In **serial communications**, data in a message is sent one bit at a time in sequence. Serial communications use just one transmission cable for low costs but slow transmission.
- Serial communications are classified by transmission direction as **simplex**, **half-duplex**, or **full-duplex communications**.
- The serial I/O transmission can be **synchronous** or **asynchronous**. Synchronous transmission transmits a stable data stream at regular intervals adjusted by a CPU clock. Asynchronous transmission transmits a data stream intermittently at irregular (random) intervals.

- Parallel transmission refers to transmit multiple bits of data simultaneously. It is not described in this course.

Simplex, Half-Duplex, and Full-Duplex

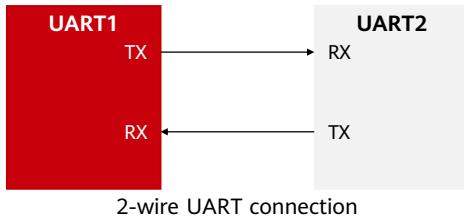
- Data can be transmitted between two devices unidirectionally (simplex), time-division bidirectionally (half-duplex), or simultaneously bidirectionally (full-duplex).
 - Simplex: data streams can be transmitted or received only in one direction.
 - Half-duplex: data streams can be transmitted and received bidirectionally, but only in one direction at a time.
 - Full-duplex: data streams can be transmitted and received in any direction at the same time.



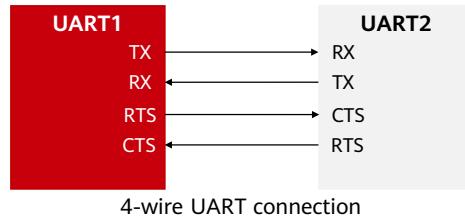
UART

UART → I2C → SPI

- Universal Asynchronous Receiver/Transmitter (UART) is a universal serial data bus used for **asynchronous communication** in **full-duplex** mode.
- UART is widely used to print information for debugging or to connect to various peripherals such as GPS and Bluetooth.
- UART has different interface standards and bus standards: RS-232, RS-499, RS-423, RS-422, and RS-485. Their main difference is the level range.



2-wire UART connection



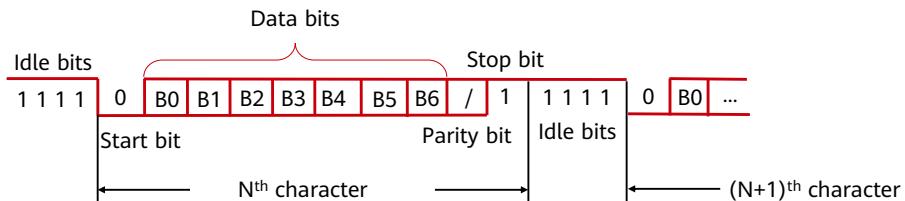
4-wire UART connection

- TX: UART transmitter. It is connected to the RX of the peer UART.
- RX: UART receiver. It is connected to the TX of the peer UART.
- RTS: Request to Send signal, indicating whether the local UART is ready to receive data. It is connected to the CTS of the peer UART.
- CTS: Clear to Send signal, indicating whether the local UART is allowed to send data to the peer end. It is connected to the RTS of the peer UART.

UART Frame Structure

UART > I2C > SPI

- The UART transmitter and receiver must have the same settings for attributes such as **baud rate** and **data format (start bit, data bits, parity bit, and stop bit)** before they start to communicate. A UART sends data to the peer end over the TX and receives data from the peer end over the RX. When the buffer used by a UART for storing received data reaches a preset threshold size, the Request to Send (RTS) signal of the UART changes to the data cannot be received, and the peer UART is prevented from sending data by the Clear to Send (CTS) signal.

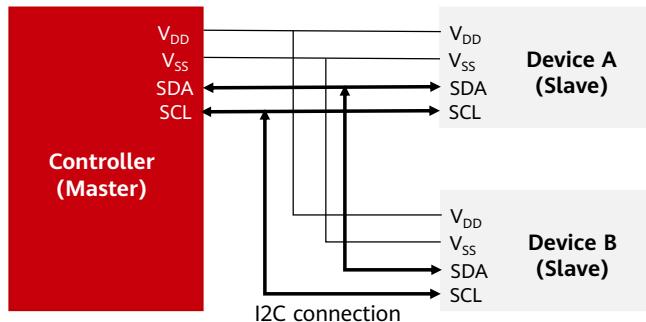


- Start bit: A logical 0 signal is sent first, indicating the start of character transmission.
- Data bits: next a few bits after the start bit. The number of data bits may be 4, 5, 6, 7, 8, which decided by the character they form. Generally, the ASCII code is used.
- Parity bit: bit after data bits. This bit is used to change the bit of 1 to an even number (even parity check) or an odd number (odd parity check) to check whether data transmission is correct.
- Stop bit: the end flag of a character. It can be 1-bit, 1.5-bit, or 2-bit high level.
- Idle bits: in the logical "1" state, indicating that no data is transmitted on the current cable.
- Baud rate: data transmission rate. It can be 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 43000, 56000, 57600 and 115200. The data sender and receiver need to use the same baud rate to ensure correct data transmission.

I2C

UART → I2C → SPI

- The Inter-Integrated Circuit (I2C) is a simple, bidirectional, two-wire, and synchronous serial bus developed by Philips.
- With I2C, one controller communicates with one or more devices through the serial data line (SDA) and serial clock line (SCL).

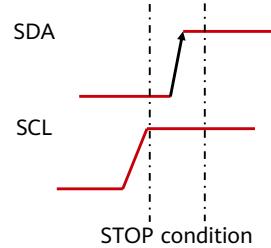
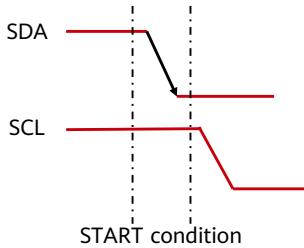


- VCC is the power supply voltage of the circuit, and VDD is the operating voltage of the chip.
- The I2C module provides a set of APIs for I2C data transfer, including:
 - Managing, opening, or closing an I2C controller
 - Performing custom transfer via a message array
- Each I2C node is recognized by a unique address and can serve as either a controller or a device. When the controller needs to communicate with a device, it writes the device address to the bus through broadcast. A device matching this address sends a response to set up a data transfer channel.
- Communication features: serial, synchronous, level, and low rate
 - Serial communication. All data is transmitted in serial mode on SDA lines by bit (8 bits each time).
 - Synchronous communication. The two communication parties work using the same clock. Party A transmits its clock to party B through a CLK signal cable, and then party B works following the clock of party A.
 - Non-differential. The I2C communication rate is low and the two communication parties are close to each other, so level signals are used.
 - Low rate. I2C is generally used between two ICs on the same board and a small amount of data needs to be transmitted, so the I2C communication rate is low (generally hundreds of kHz). Different I2C chips may have different communication rates, which is limited by the maximum rate allowed by devices during programming.

I2C Communication Timing (1)

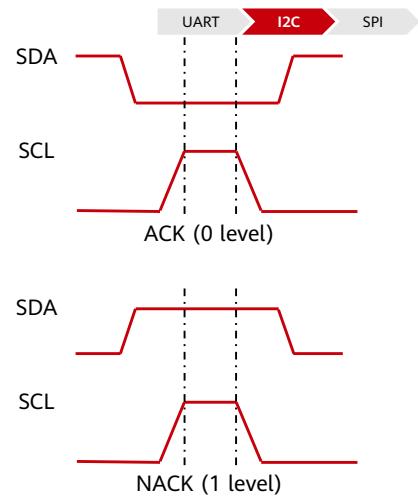
UART → I2C → SPI

- I2C timing consists of four elements: **START condition**, **STOP condition**, **acknowledge (ACK) condition (0)**, and **not-acknowledge (NACK) condition (1)**.
 - START condition: a high-to-low voltage transition on the SDA line while the SCL voltage is high. It is a level transition timing signal, not a level signal.
 - STOP condition: a low-to-high voltage transition on the SDA line while the SCL voltage is high. It is a level transition timing signal, not a level signal.



I2C Communication Timing (2)

- Response signals:
 - All data on the I2C bus is transmitted in **8-bit bytes**.
Each time the sender transmits a byte, the data line is released during the clock pulse, and the receiver feeds back a response signal.
 - When the response signal is low, it is defined as an ACK (the receiver successfully received the byte).
When the response signal is high, it is defined as an NACK (the receiver failed to receive the byte).

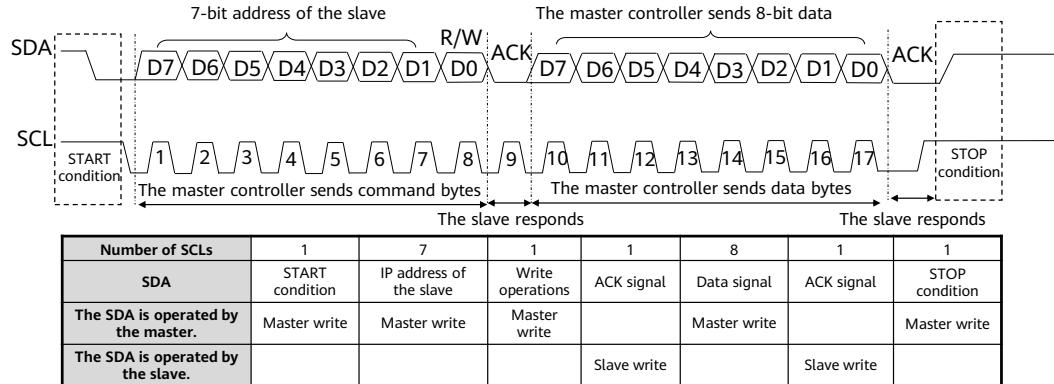


- The requirement for ACK is that the receiver pulls the SDA line low during the low level before the ninth clock pulse and ensures that the SDA line is at a stable low level during the high level of the clock. If the receiver is the master controller, after it receives the last byte, it sends a NACK signal to notify the sender to stop data transmission and release the SDA line so that the master receiver can send a STOP signal.

I2C Data Transmission

UART → I2C → SPI

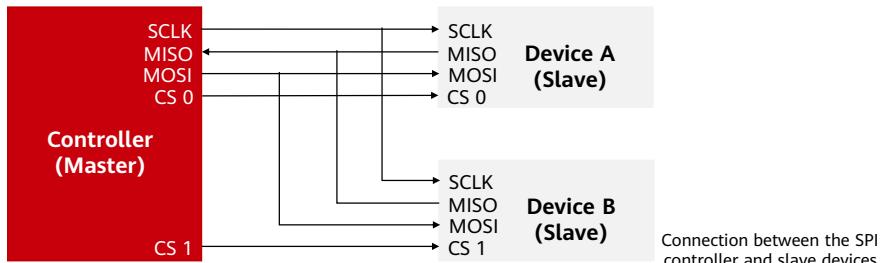
- The I2C data transmission must start with a START condition and stop with a STOP condition. The data is transferred bit by bit in bytes. The upper bits are transmitted first.



SPI

UART → I2C → SPI

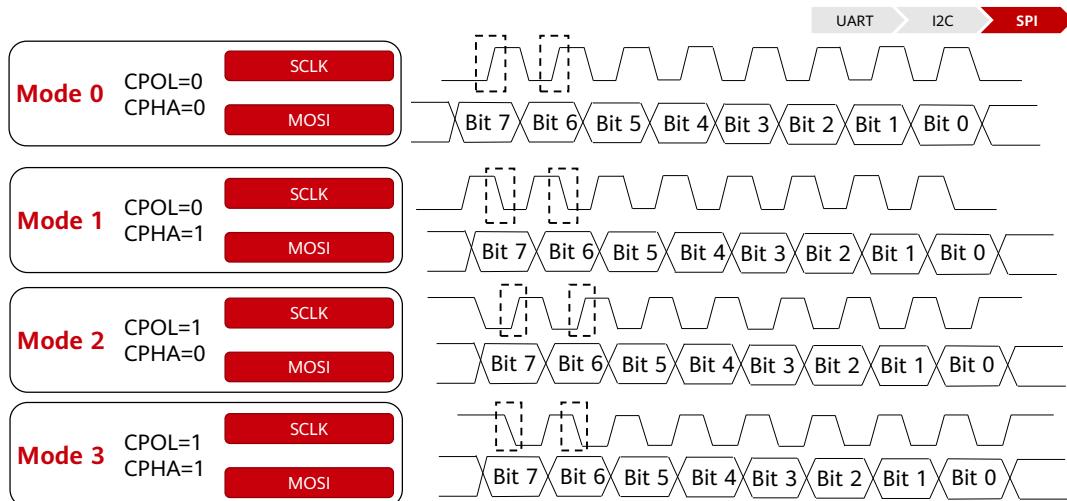
- Serial Peripheral Interface (SPI) is a high-speed, full-duplex, and synchronous communication bus developed by Motorola.
- It is commonly used for communication with flash memory, real-time clocks, sensors, and ADCs.
- The figure below shows the connection between one controller and two slave devices (device A and device B). Device A and device B share three pins (Serial Clock (SCLK), Master In Slave Out (MISO), and Master Out Slave In (MOSI)) of the controller. CS 0 of device A and CS 1 of device B are connected to CS 0 and CS 1 of the controller, respectively.



Connection between the SPI controller and slave devices

- SPI works in master/slave mode. Generally, there is one SPI controller that controls one or more SPI slave devices. They are connected via four wires:
 - SCLK: clock signal output from the SPI controller
 - MOSI: data output from the SPI controller to a device
 - MISO: data output from an SPI device to the controller
 - Chip select (CS): output from the SPI controller to indicate that data is being sent. It is controlled by the SPI controller.
- SPI communication is usually initiated by the controller and is performed as follows:
 - The SPI controller selects a device to communicate on the select line. Only one device can be selected at a time.
 - SCLK provides clock signals to the selected device.
 - The SPI controller sends data to the device via MOSI, and receives data from the devices via MISO.

SPI Data Transmission



36 Huawei Confidential



- During SPI data transmission, you need to determine the following:
 - Whether data is sampled on the rising edge or falling edge of the clock.
 - Whether the idle status of the clock is at a high level or low level.
- SPI can work in one of the following modes according to the combination of Clock Polarity (CPOL) and Clock Phase (CPHA) of the clock signal:
 - If both CPOL and CPHA are **0**, the clock signal level is low in the idle state and data is sampled on the first clock edge (rising edge).
 - If CPOL is **0** and CPHA is **1**, the clock signal level is low in the idle state and data is sampled on the second clock edge (falling edge).
 - If CPOL is **1** and CPHA is **0**, the clock signal level is high in the idle state and data is sampled on the first clock edge (falling edge).
 - If both CPOL and CPHA are **1**, the clock signal level is high in the idle state and data is sampled on the second clock edge (rising edge).

Comparison Between UART, I2C, and SPI

Item Serial Port	Number of Signal Cables	Device Subordinate Relationship	Communication Mode	Communication Rate
UART	TX, RX (CTS, RTS)	/	Full-duplex	100 bit/s–1.152 Mbit/s
I2C	SDA, SCL	There are master and slave devices. The I2C selects the slave device by address.	Half-duplex	100 kbit/s (standard) 400 kbit/s (fast) 3.4 Mbit/s (high- speed)
SPI	SCLK, MOSI, MISO, CS	There are master and slave devices. The SPI selects the slave device by CS signal.	Full-duplex communication	Up to dozens of Mbit/s

Quiz

1. (True or false) Data in ROM will not be lost during a power failure. Data can be written in ROM only once and then read only.
2. (Single-answer question) Which of the following signal lines is not an SPI signal line? ()
 - A. MOSI
 - B. MISO
 - C. SCL
 - D. CS

- Answers:

- T
 - D

Quiz

3. (Multiple-answer question) Which of the following operations are performed by a CPU? ()
- A. Fetching instructions
 - B. Decoding instructions
 - C. Executing instructions
 - D. Fetching operands

- Answer:
 - ABCD

Summary

- In this course, you have learned the basic concepts, the components, module functions, and working principles of the MCU. In addition, you also learned the concepts and working principles of some common MCU peripherals, as well as the principles and data structures of some common serial port buses.

Acronyms or Abbreviations

- ADC: Analog-to-Digital Converter
- CPU: Central Processing Unit
- DAC: Digital-to-Analog Converter
- I2C: Inter Integrated Circuit, a two-wire bidirectional binary synchronous serial bus
- MCU: Microcontroller Unit
- PVM: Pulse Width Modulation
- RAM: Random Access Memory
- ROM: Read Only Memory
- SPI: Serial Peripheral Interface

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

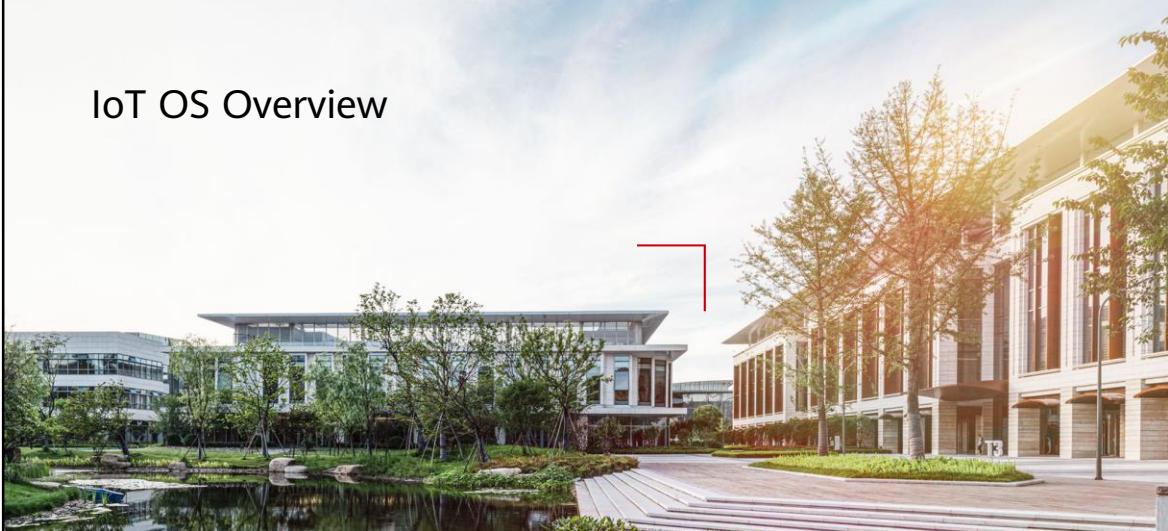
Bring digital to every person, home, and
organization for a fully connected,
intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive
statements including, without limitation, statements regarding
the future financial and operating results, future product
portfolio, new technology, etc. There are a number of factors
that could cause actual results and developments to differ
materially from those expressed or implied in the predictive
statements. Therefore, such information is provided for reference
purpose only and constitutes neither an offer nor an acceptance.
Huawei may change the information at any time without notice.



IoT OS Overview



Foreword

- Decades have passed since the first computer operating system (OS) was created. OSs play a vital role in our lives, ranging from initially implementing human-computer interaction to controlling computers and other devices. What are the opportunities and challenges of OSs in the IoT era?
- This section describes how OSs have evolved over time, along with some basic concepts. It also introduces Huawei LiteOS and OpenHarmony, including their technical architectures, features, and the relationships between them.

Objectives

- Upon completion of this course, you will:
 - Have a better understanding of OSs and their history.
 - Understand the functions of IoT OSs.
 - Understand Huawei LiteOS.
 - Understand OpenHarmony.
 - Understand the Huawei IoT OS ecosystem.

Contents

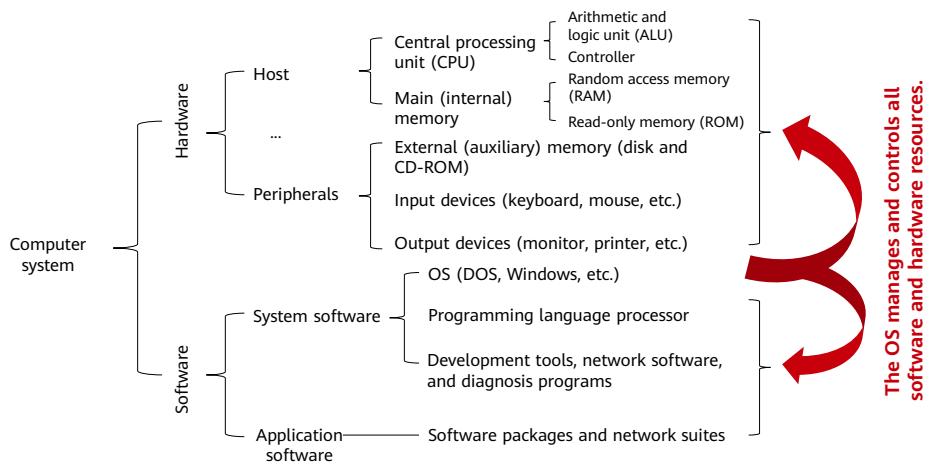
1. OS Overview

- OS
 - IoT OS
- 2. Huawei LiteOS
- 3. OpenHarmony
- 4. Huawei IoT OS Ecosystem

OS

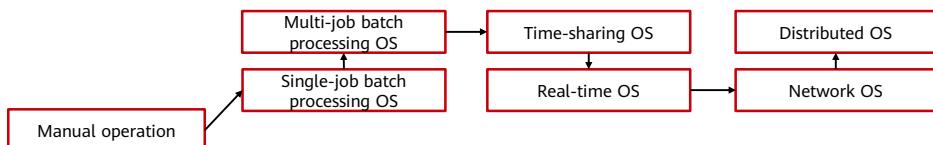
- What is an OS?
 - An OS is a **computer program that manages computer hardware and software resources**. The transactions that an OS processes include managing and configuring **memory**, setting priorities when allocating system resources, controlling **I/O devices**, and managing the network and file systems. Additionally, an OS provides an interface for users to interact with systems.
- In a computer, an OS is the most important basic system software. From the user perspective, the OS provides various services. From the programmer perspective, it is the main interface that users log in to. From the designer perspective, it is the connection between various modules and units to implement different functions. After decades of development, the computer OS has become one of the largest and most complex software systems out there.

Computer System Structure

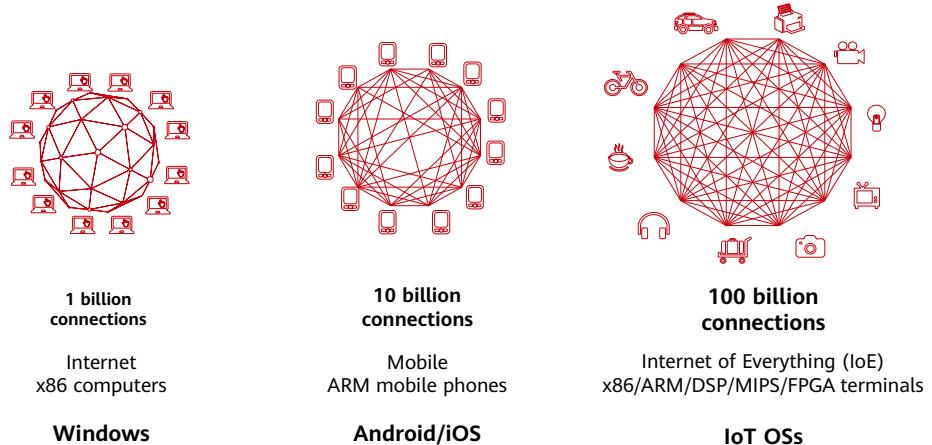


OS Types

- Batch processing OSs
 - Single-job: Programs are loaded into the memory and executed individually.
 - Multi-job: Multiple programs are loaded into the memory and executed simultaneously.
- Time-sharing OSs
 - OSs that interleave the execution of each program among users in short time slots. Each user can interact with the computer through a terminal.
- Real-time OSs
 - OSs that implement a specific function within a defined time frame. Real-time OSs are divided into soft real-time OSs and hard real-time OSs.
- Network and distributed OSs
 - Sharing of various resources in the network and communication between computers. The difference between distributed OSs and network OSs is that in the former, several computers cooperate with each other to complete the same task.



From Internet to Mobile to IoE



Contents

1. OS Overview

▫ OS

▪ IoT OS

2. Huawei LiteOS

3. OpenHarmony

4. Huawei IoT OS Ecosystem

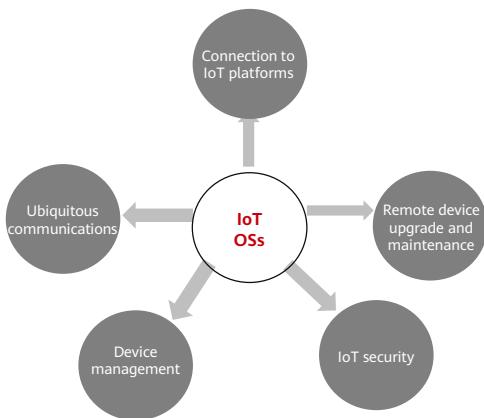
IoT OS Requirements

- As IoT applications propagate, consumers have become IoT service objects. New IoT services differ greatly from the traditional dedicated services. The O&M, network architectures, and data processing are all different. IoT OSs need to evolve from siloed and closed to being more open and interconnected.
- To be open and interconnected, IoT OSs must be able to adapt to different types of terminals.



IoT OS Technical Capabilities

- To provide the full functionality of IoT applications, IoT OSs must have the following technical capabilities:



Three Ways of IoT OS Development

- Because existing OSs cannot meet IoT application requirements, IoT OSs often require the following:

Tailoring and customization
based on Android/iOS

More connectivity based on
embedded OSs

The development of new IoT OSs

Contents

1. OS Overview
- 2. Huawei LiteOS**
3. OpenHarmony
4. Huawei IoT OS Ecosystem

Huawei LiteOS Overview

- Huawei LiteOS is a lightweight IoT OS developed by Huawei. It is widely used in smart homes, wearables, Internet of Vehicles (IoV), urban public services, and manufacturing. Since Huawei LiteOS is open source, it has helped partners in the NB-IoT market with new technologies, ecosystems, solutions, and commercial support. Huawei LiteOS has facilitated the building of an open source IoT ecosystem. Users come from diverse industries, such as metering, parking, street lamp management, environmental protection, bicycle sharing, and logistics.
- The Huawei LiteOS open source project supports chip architectures such as ARM 64, ARM Cortex-A, ARM Cortex-M0, Cortex-M3, Cortex-M4, and Cortex-M7.
- Huawei LiteOS provides the following features:

**High real-time performance
High stability**

Ultra-compact kernel

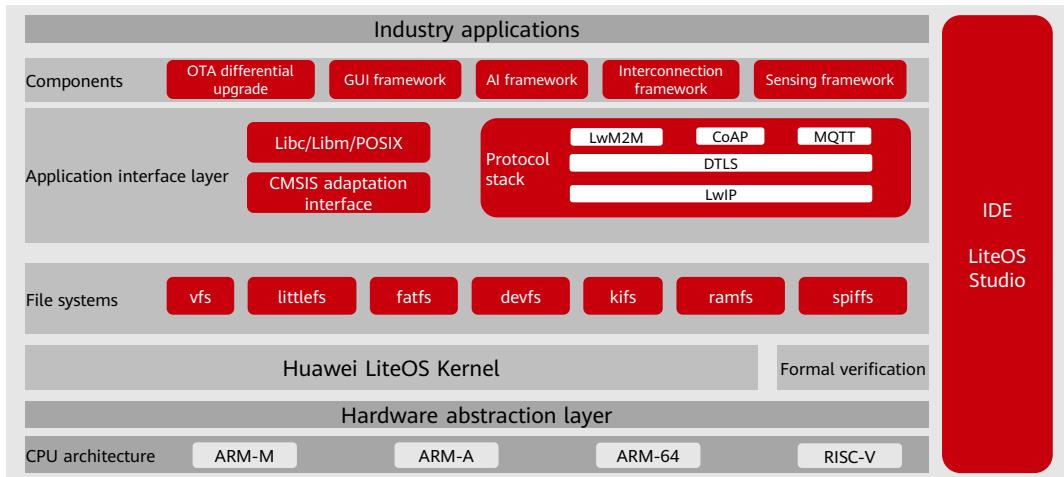
The basic kernel can be tailored to less than 10 KB.

Low power consumption

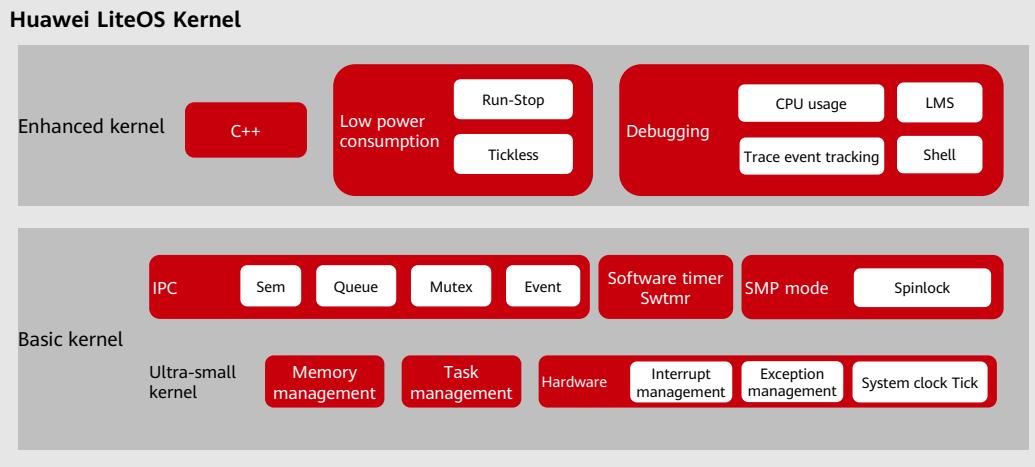
Microampere-level (μA)

Static tailoring

Huawei LiteOS Technical Architecture



Huawei LiteOS Kernel Architecture



Huawei LiteOS Technical Features

-  **Low power consumption framework**
 - LiteOS is lightweight (the kernel can be as small as 6 KB), and it launches fast. It can also run in tickless mode to save power when collecting sensor data.
-  **OpenCPU architecture**
 - It is designed for the LiteOS lightweight kernel and limited hardware resources. OpenCPU integrates a microcontroller unit (MCU) and communication module, so it can significantly reduce the size and costs of terminals such as water meters, gas meters, and vehicle detectors in low power wide area (LPWA) scenarios.
-  **Security design**
 - Secure transmission features low power consumption and supports two-way authentication, firmware over-the-air (FOTA) differential upgrades, and DTLS or DTLS+.
-  **Device-cloud interconnection component**
 - The device-cloud interconnection component of the LiteOS SDK integrates a full set of IoT connectivity protocol stacks such as LwM2M, CoAP, MQTT, mbed TLS, and LwIP. It allows for faster development and enables devices to quickly connect to an IoT platform.
-  **Software over-the-air (SOTA) upgrade**
 - A differential upgrade reduces the package size, so it is a good fit for low-bandwidth and battery-powered environments. Optimized differential combination algorithms reduce the demand on RAM resources, an important requirement for low resource devices.

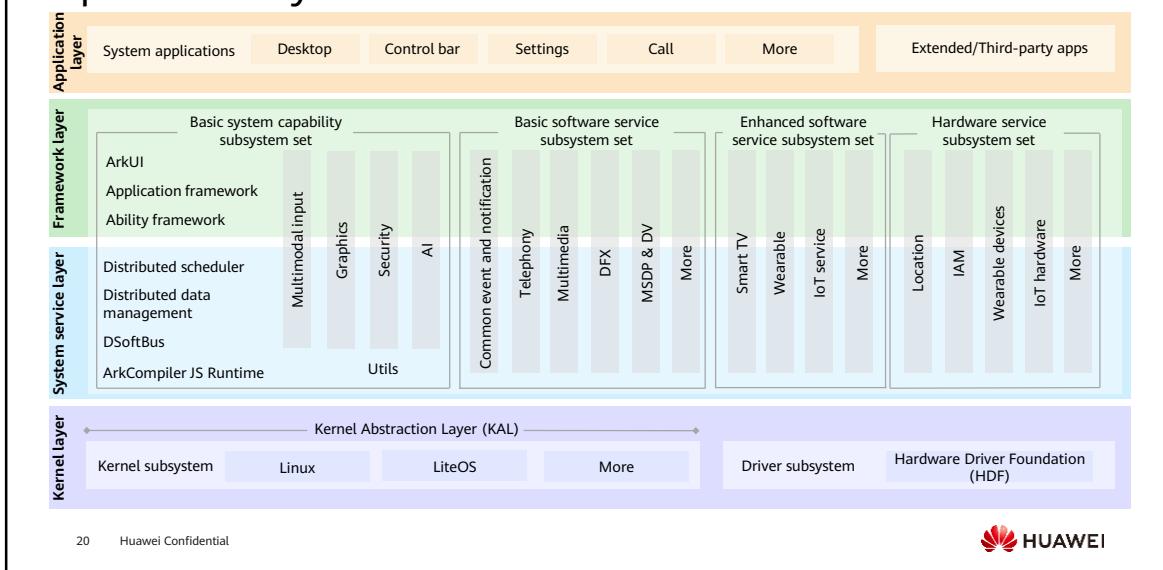
Contents

1. OS
2. Huawei LiteOS
- 3. OpenHarmony**
4. Huawei IoT OS Ecosystem

OpenHarmony Overview

- OpenHarmony is an open source project incubated and managed by the OpenAtom Foundation. The purpose of this project is to build an open-source, distributed OS framework for smart devices in all scenarios of a fully-connected world.
- HarmonyOS is a commercial release based on OpenHarmony.
- HarmonyOS Connect is Huawei's smart hardware ecosystem brand.

OpenHarmony Technical Architecture



- OpenHarmony is designed with a layered architecture, which consists of the kernel layer, system service layer, framework layer, and application layer from the bottom up. System functions are expanded by levels, from system to subsystem, and further to component. In a multi-device deployment scenario, unnecessary components can be excluded from the system as required.

OpenHarmony Technical Features

A unified OS for flexible deployment

- OpenHarmony enables hardware resources to be scaled with its component-based and small-scale designs. It can be deployed on demand for a diverse range of devices, including ARM, RISC-V, and x86 architectures, and providing RAM volumes ranging from hundreds of KB to GB.

Hardware collaboration and resource sharing

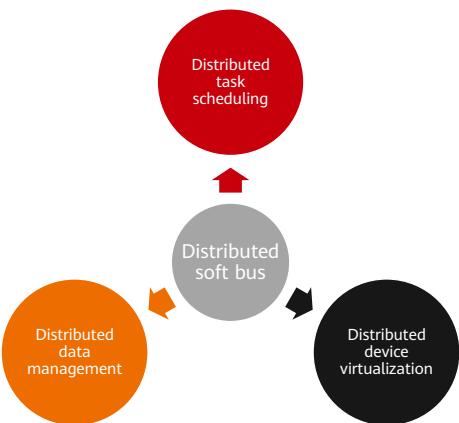
- Devices running HarmonyOS can be connected at the system layer as a Super Device, enabling capability sharing and delivering a seamless experience.

One-time development for multi-device deployment

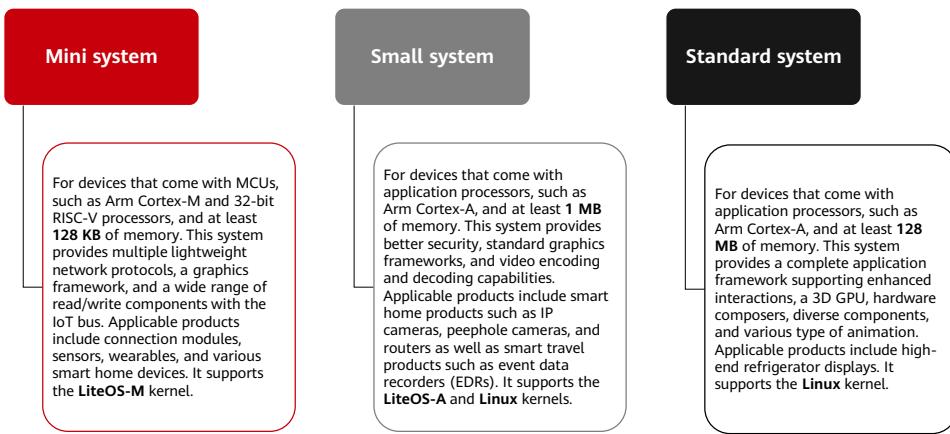
- OpenHarmony provides application, ability, and UI frameworks. With these frameworks, developers can develop applications once, and then flexibly deploy them across a broad range of different devices. Consistent APIs ensure the operational compatibility of applications across devices.

OpenHarmony Distributed Features

- **Distributed soft bus (DSoftBus)** is a unified base for seamless interconnection among devices. It powers OpenHarmony with distributed communications capabilities to quickly discover and connect devices, and efficiently transmit data.
- **Distributed data management** leverages DSoftBus to manage application and user data distributed on different devices.
- **Distributed task scheduling** is based on technical features such as DSoftBus, distributed data management, and distributed profile. It supports remote startup, remote invocation, binding/unbinding, and migration of applications across devices.
- **Distributed device virtualization** enables cross-device resource convergence, device management, and data processing so that virtual peripherals can function as capability extensions of smartphones to form a Super Device.



OpenHarmony Types



Development Boards and Chips for OpenHarmony

- Systems supported by OpenHarmony are classified based on the device memory as standard, small, or mini.
- The following development boards can be used for OpenHarmony-based hardware development:

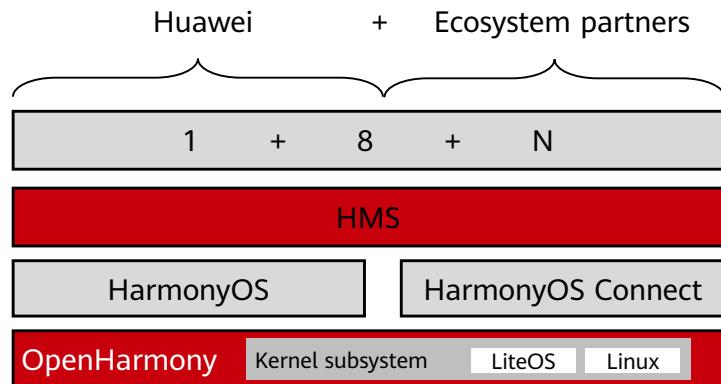
System Type	Board Model	Chip Model	Application Scenario
Standard system	HH-SCDAYU200	RK3568	Entertainment, easy travel, and smart home devices, such as kitchen hoods, ovens, and treadmills.
Small system	Taurus AI Camera	Hi3516DV300	Devices with screens, such as refrigerators and head units.
Small system	BearPi-HM Micro	STM32MP157	Smart home central control panel.
Mini system	Pegasus Wi-Fi IoT	Hi3861	Smart home devices such as white goods and small home appliances.
Mini system	BearPi-HM Nano	Hi3861	Connection devices like smart street lamps, smart logistics sensors, and human body infrared sensors.

- For details, see <https://gitee.com/openharmony>.

Contents

1. OS
2. Huawei LiteOS
3. OpenHarmony
- 4. Huawei IoT OS Ecosystem**

Huawei IoT OS Ecosystem



- The Huawei IoT OS ecosystem includes Huawei and ecosystem partners. HarmonyOS is mainly used for Huawei products. Ecosystem partners can join the all-scenario "1+8+N" IoT ecosystem through HarmonyOS Connect.

Quiz

- (Single-answer question) Which of the following kernels is not part of the OpenHarmony kernel subsystem?
 - LiteOS-M
 - LiteOS-A
 - Linux
 - Unix
- (Multiple-answer question) Which of the following protocol stacks are supported by Huawei LiteOS?
 - Lightweight Machine to Machine (LwM2M)
 - Constrained Application Protocol (CoAP)
 - Message Queuing Telemetry Transport (MQTT)
 - User Datagram Protocol (UDP)

- Answers:

- D
 - ABCD

Summary

- This section described the OS basic concepts and development history, challenges and requirements for OSs in the IoT era, Huawei LiteOS technical architecture and features, OpenHarmony technical architecture, features, and types, as well as details of the IoT OS ecosystem.

Recommendations

- Huawei Cloud IoT Device Access (IoTDA) product documentation:
 - <https://support.huaweicloud.com/intl/en-us/iothub/index.html>
- Huawei LiteOS documentation:
 - <https://support.huaweicloud.com/LiteOS/index.html>
- OpenHarmony project
 - <https://gitee.com/openharmony>

Acronyms or Abbreviations

- CPU: Central Processing Unit
- DOS: Disk Operating System
- DSP: Digital Signal Processing
- FPGA: Field Programmable Gate Array
- IPC: Inter-Process Communication
- RAM: Random Access Memory
- ROM: Read-Only Memory

Thank you.

把数字世界带入每个人、每个家庭、

每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.



IoT Communications Technologies



Foreword

- Communications technologies are the foundation of IoT. If IoT is a logistics system, communications technologies are different modes of transportation. These technologies connect sensing devices to applications on the cloud platform.
- This section describes wired and wireless IoT communications technologies. These technologies have different standards for different scenarios and technical features.

Objectives

- Upon completion of this course, you will have knowledge of:
 - Common wired IoT communications technologies
 - Technical standards and features of different short-range wireless communications technologies
 - Features and development trends of cellular mobile communications technologies
 - Features of low power wide area (LPWA) and different LPWA technologies.

Contents

- 1. Wired Communications Technologies**
2. Wireless Communications Technologies

Ethernet

- Ethernet is a computer local area network (LAN) technology. Technical standard: Institute of Electrical and Electronics Engineers (IEEE) 802.3, which defines cable connections at the physical layer, electronic signals, and protocols at the media access control (MAC) layer.
- A base station has three types of ports: one ETH port, two FE/GE ports, and two SFP ports. SFP ports are gigabit optical ports. ETH is a standard Ethernet network, which transmits data at the rate of 10 Mbit/s. Fast Ethernet (FE) supports a transmission rate of 100 Mbit/s. Gigabit Ethernet (GE) supports a transmission rate of 1,000 Mbit/s. FE/GE are auto-adaptive ports that support 100 Mbit/s and 1,000 Mbit/s based on auto negotiation with the peer switch.
- Ethernet uses carrier sense multiple access with collision detection (CSMA/CD). Carrier sense: The carrier is checked before sending. Multiple access: Data sent by a station is received by other stations. Collision detection: The station detects whether there is a collision when sending data.

- CSMA/CD is a data transmission method used in Ethernet.
- Carrier sense indicates that each station on the network checks whether data is transmitted on the bus before sending data. If the bus is busy, the station will not transmit data. If the bus is idle, the station will transmit prepared data immediately.
- Multiple access indicates that all stations on the network use the same bus to send and receive data, and the data is transmitted in broadcast mode.
- Conflict detection: When a station sends a frame, it listens to the media to detect whether a collision occurs (whether other stations are sending frames.)
- CSMA/CD specifications are standardized by IEEE 802.3 and ISO 8802-3.
- IEEE 802.3 defines the Ethernet physical (PHY) layer, which is a transceiver for transmitting and receiving data at the physical layer. The physical layer defines electrical and optical signals, link status, clock reference, data coding, and circuits required for data transmission and reception, and provides standard interfaces for devices at the data link layer.
- IEEE 802.3 defines the Ethernet MAC, which is a media access controller and implements the data link layer. The data link layer provides functions such as addressing mechanism, data frame construction, data error check, transmission control, and standard data interfaces for the network layer.

RS-232

- RS-232 (or EIA RS-232) is a standard interface between data terminal equipment (DTE) and data communications equipment (DCE) for serial binary data exchange. It was jointly developed in 1970 by the Electronics Industries Association (EIA), Bell System, modem manufacturers, and computer manufacturers.
- Technical features:

Few signal cables

Many baud rates

Negative logic used
for transmission

Short transmission
range

- Few signal cables:
 - An RS-232 bus specifies 25 lines and contains two signal channels: the first (primary) and second (secondary) channels. The RS-232 bus can be used to achieve full-duplex communications. The primary channel is often used, and the secondary channel is used less. Generally, three to nine signal lines can be used to achieve full-duplex communications, and three signal lines (reception line, transmission line, and signal ground) can achieve a simple full-duplex communications process.
- Many baud rates available:
 - RS-232 specifies the following standard transmission rates: 50 bit/s, 75 bit/s, 110 bit/s, 150 bit/s, 300 bit/s, 600 bit/s, 1,200 bit/s, 2,400 bit/s, 4,800 bit/s, 9,600 bit/s, 19,200 bit/s. It can flexibly adapt to devices with different rates. A lower transmission rate can be used for slow peripherals, and vice versa.
- Negative logic used for transmission:
 - The level range of logic 1 is -5V to -15 V, and the level range of logic 0 is 5 V to 15 V. The purpose of selecting this electrical standard is to improve anti-interference and increase the transmission range. The noise tolerance of RS-232 is 2 V, and the receiver will recognize signals as high as 3 V as logic 0 and signals as low as -3 V as logic 1.
- Short transmission range:
 - RS-232 adopts the serial transmission mode and converts the transistor-transistor logic (TTL) level of the microcomputer into the RS-232 level, so the transmission range can reach 30 m.

RS-485

- Originally: Simple process values were output based on analog signals. Later: RS-232 was used as an instrument interface for point-to-point communications, but cannot implement connections at scale. Solution: RS-485 was developed.
- RS-485 is a serial communications protocol without the disadvantages of RS-232. RS-485 uses a differential, half-duplex transmission mode and supports point-to-multipoint communications. An RS-485 bus supports two-wire and four-wire connections. Four-wire connections implement only point-to-point communications thus are seldom used. Two-wire connections support both point-to-point (single-slave) and point-to-multipoint (multi-slave) communications.

- In the early 1980s, MCU technologies were maturing, and smart meters dominated the global meter market. This is driven by enterprise digitalization. Enterprises want meters with network communication interfaces.

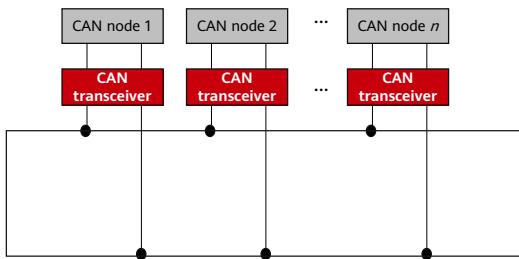
RS-232 vs. RS-485

Item	RS-232	RS-485
Transmission range	Less than 20 m	Theoretical: 1,200 m; actual: 300–500 m
Transmission mode	Unbalanced, single-end	Balanced, differential
Number of transceivers	Point-to-point	Up to 128 transceivers on the bus
Transmission rate	38.4 Kbit/s	10 Mbit/s

- RS-232 is one of the communication interfaces on personal computers and a standard interface developed by EIA for asynchronous transmission.
- The differences between RS-232 and RS-485 are as follows:
 - Transmission mode: RS-232 uses unbalanced, single-end transmission. RS-485 uses balanced, differential transmission.
 - Transmission range: RS-232 is suitable for communications between local devices. The transmission range is usually less than or equal to 20 m. The transmission range of RS-485 can reach tens of meters to thousands of meters.
 - RS-232 supports only point-to-point communications, while RS-485 allows up to 128 transceivers connected on the bus.
- If serial communications are compared to traffic and UART is compared to a station, a frame of data is like a car. Cars on the road must follow traffic rules. The speed limit can be 30 km/h or 40 km/h in a city, and 120 km/h on highways. What road a car runs on and how fast a car runs depend on the protocol. Common serial interface protocols include RS-232, RS-422, and RS-485.

CAN

- Controller Area Network (CAN) is a serial communications protocol recognized by the International Organization for Standardization (ISO). It was developed by Bosch (a German company) in 1983 for data communications between monitoring and execution modules of an automobile internal control system. It is now one of the most widely used fieldbuses in the world.
- A CAN bus uses serial data transmission. When a node on the CAN bus broadcasts data packets, all other nodes (including non-targets) receive the data.
- CAN bus topology:



- Main features of the CAN bus:
 - Long data transmission range (up to 10 km)
 - High data transmission rate (up to 1 Mbit/s)
 - Excellent arbitration mechanism
 - It uses filters to implement multi-address data frame delivery.
 - It uses remote frames for remote data requests.
 - It can detect and process errors.
 - It can automatically resend data.
- A faulty node can be automatically disconnected from the bus without affecting other nodes.

USB

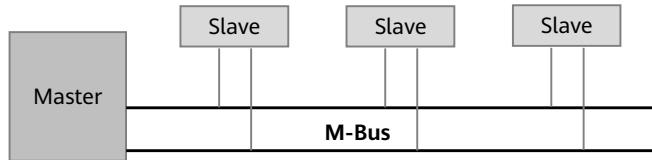
- Universal Serial Bus (USB) is a standard for connecting computer systems and external devices. It is also a technical specification for input/output interfaces widely used on PCs, mobile devices, cameras, set top boxes (STBs), and game consoles.
- USB rates of each version:

Official Version	Original Name	Transmission Rate
LowSpeed	USB 1.0	1.5 Mbit/s
FullSpeed	USB 1.1	12 Mbit/s
HiSpeed	USB 2.0	480 Mbit/s
Gen 1	USB 3.0 Gen1	5 Gbit/s
Gen 2	USB 3.0 Gen2	10 Gbit/s
Gen 2×2	N/A	20 Gbit/s
N/A	USB 4.0	40 Gbit/s

- Before the emergence of USB, there were lots of serial and parallel interfaces. For example, keyboards, mouse devices, modems, printers, and scanners are all connected to different interfaces. One interface can be used to connect to only one device. Computers are unlikely to support so many interfaces, resulting in insufficient scalability and low speed. USB is designed for high speed, scalability, and ease of use.
- USB is much faster than standard buses used by traditional computers, such as parallel interfaces (EPP and LPT) and serial interfaces (RS-232). The maximum rate of USB 1.1 (USB 2.0 FullSpeed) is 12 Mbit/s. The maximum rate of USB 2.0 (USB 2.0 HiSpeed) is 480 Mbit/s. The maximum rate of USB 3.0 (USB 3.2 Gen1) is 5 Gbit/s. The maximum rate of USB 3.1 (USB 3.2 Gen2x1) is 10 Gbit/s. The maximum rate of USB 3.2 (USB 3.2 Gen2x2) is 20 Gbit/s. The maximum rate of USB 4.0 released recently is 40 Gbit/s.

M-Bus

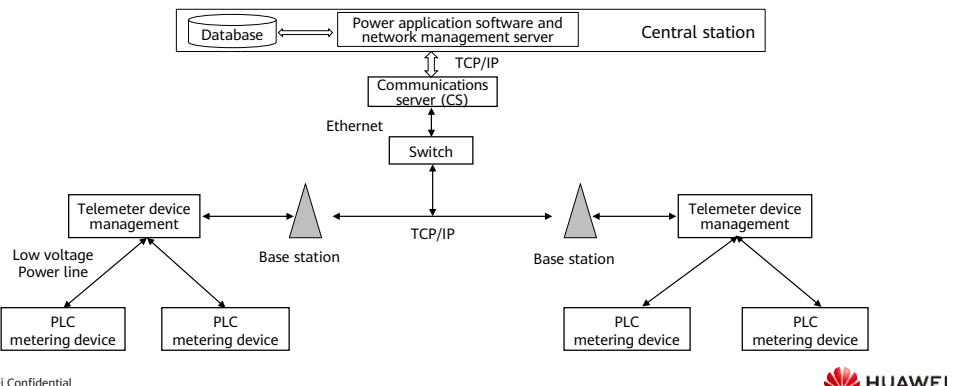
- Meter-Bus (M-Bus) is a standard for data buses of measuring instruments and counters to transmit information. It is widely used for collecting energy consumption data of commercial and industrial buildings.
- Its transmission limit is 1,000 m and can supply power to field devices without auxiliary power cables. A bus allows up to 5 A voltage, and a node allows up to 0.65 mA power.
- M-Bus topology:



- M-Bus was proposed by Dr. Horst Ziegler from the University of Paderborn and Deutschland GmbH and Techem GmbH from the Texas Instruments (TI) company. It is based on the open system interconnection (OSI) reference model but not a real network. M-Bus defines only the functions of the physical layer, data link layer, network layer, and application layer of the OSI network model. Parameters such as the baud rate and address cannot be changed by an upper layer in the OSI model, so M-Bus defines a management layer to manage other layers without complying with the OSI model rules.

PLC

- Power Line Communication (PLC) is a communication technology that transmits data and media signals over power lines. PLC loads high frequency signals containing information onto the current. An adapter receives the information over the cable, separates the high-frequency signals from the current, and sends the signals to a computer or telephone.



- PLC-IoT is a power line communications technology developed by Huawei and uses Huawei HiSilicon chips.
- G3-PLC is an open power line communications protocol designed for global smart grids. It was initiated by the European Regional Development Fund (ERDF) and co-developed by Maxim and Sagem Communications. G3-PLC is a narrowband power line carrier communications standard. It is usually used for low-speed data communications such as automatic metering, energy control, and power grid detection.
- PRIME is an open multi-vendor solution launched by the PRIME Alliance, which consists of more than 30 power supply companies, industry vendors, and university research centers. PRIME uses orthogonal frequency division multiplexing (OFDM), a new communications technology. It contains a large database that stores noise levels, noise rhythms, signal weakening, and impedance modes, and provides accurate statistics models for power grids.
- The parameters of the three PLC technologies are as follows:
 - G3-PLC: subcarrier spacing: 1.5625 kHz; subcarrier width: 3.125 kHz; transmission bandwidth: 54.7 kHz. A total of 36 subcarriers can be deployed.
 - PRIME: subcarrier spacing: 0.488 kHz; subcarrier width: 0.976 kHz; transmission bandwidth: 47 kHz. A total of 96 subcarriers can be deployed.
 - PLC-IoT: subcarrier spacing: 4.6875 kHz; subcarrier width: 9.375 kHz; transmission bandwidth: 2,000–12,000 kHz. A total of 128 subcarriers can be deployed.

Comparison of Wired Communications Technologies

Mode	Feature	Application Scenario
Ethernet	Comprehensive protocols, universal, cost-effective	Intelligent devices, video surveillance
RS-232	Point-to-point communications, cost-effective, short transmission range	A few instruments, industrial control
RS-485	Bus topology, cost-effective, strong anti-interference capability	Industrial instruments, metering
CAN	High real-time performance, long transmission range, vulnerable to electromagnetic interference, and cost-effective	Industrial automation, ships, medical equipment, industrial equipment
USB	Point-to-point communications, universal, fast transmission	Smart home, office, mobile devices
M-Bus	Designed for metering, common twisted-pair cables, strong anti-interference capability	Industrial energy consumption data collection
PLC	For power line communications, wide coverage, easy installation	Power grid transmission, electricity meters

Contents

1. Wired Communications Technologies
2. **Wireless Communications Technologies**
 - Short-Range Wireless Communications Technologies
 - Cellular Mobile Communications Technologies
 - LPWA Communications Technologies
 - Comparison of Wireless Communications Technologies

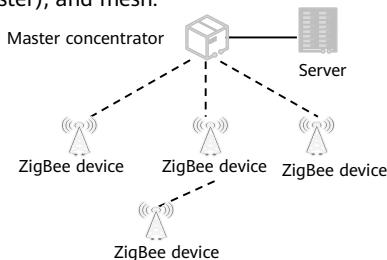
Bluetooth

- Bluetooth is a large-capacity wireless communications technology standard for short distances. The latest Bluetooth 5.0 has a transmission limit of 3 Mbit/s and 300 meters. Two types of Bluetooth: Bluetooth Basic Rate/Enhanced Data Rate (BR/EDR) and Bluetooth Low Energy (BLE). BR/EDR supports only point-to-point (one-to-one) communications. BLE supports point-to-point, broadcast (one-to-many), and mesh (many-to-many) communications. BLE is mainly used in IoT for smart home products to deliver better performance and consume less power.
- Advantages: high rates, high security, and low power
- Disadvantages: EDR has few network nodes thus is not suitable for multi-point deployment.

- Bluetooth is a wireless communications technology standard that implements short-range data exchange between fixed devices, mobile devices, and personal area networks (PDNs). It uses ISM 2.4–2.485 GHz ultra high frequency (UHF) bands. Bluetooth was developed by the telecom giant Ericsson in 1994 as an alternative to RS-232 data lines. It can connect multiple devices, without data synchronization issues.
- Today, Bluetooth is managed by the Bluetooth Special Interest Group (SIG). Bluetooth SIG has more than 25,000 members around the world in different industries such as telecommunications, computers, networks, and consumer electronics.
- Bluetooth 5.0 supports faster and longer transmission with low power consumption. The transmission rate is twice that of Bluetooth 4.2 (up to 2 Mbit/s). The transmission range is four times that of Bluetooth 4.2 (theoretically up to 300 m). The data packet capacity is eight times that of Bluetooth 4.2.
- It supports indoor positioning and navigation and can achieve a positioning precision of less than 1 m by working with Wi-Fi.
- It optimizes IoT foundation layers to enable lower power consumption and higher performance for smart homes.

ZigBee

- ZigBee is a short-range, low-rate wireless communications technology with low power. It is for periodic data, intermittent data, and low response time data transmission. Features: short transmission range, low complexity, self-organization, low power, and low data rate. It is widely used in industrial and smart home fields.
- ZigBee supports three types of network topologies: star, tree (cluster), and mesh.

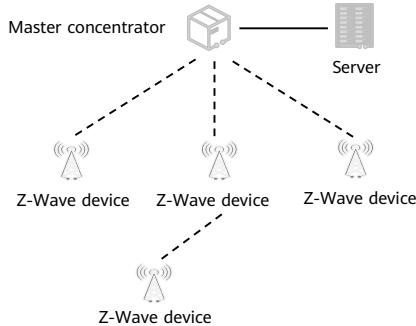


ZigBee	
Low power	Two batteries for 6–24 months
Low cost	No patent fee. Cost of each module: USD2
Low rate	20–250 Kbit/s
Short range	10–100 m
Low latency	15–30 ms
Large capacity	Up to 254 nodes
Strong security	Three-layer security
License-free	915 MHz, 868 MHz (Europe) 2.4 GHz (global)
Easy networking	Mesh networking, self-networking
Incompatibility	Incompatible between different chips
Difficult to maintain	Too flexible to maintain

- ZigBee is a low-power local area network (LAN) protocol based on IEEE 802.15.4 and also a short-range, low-power wireless communications technology. The name is derived from the waggle dance of honey bees after their return to the beehive. Bees exchange the source of pollen location to by zig-zagging and other moves. In other words, the waggle dance helps bees to build their communication network. Features: short transmission range, low complexity, self-organization, low power, and low data rate. ZigBee can be integrated with various devices for automatic control and remote control.

Z-Wave

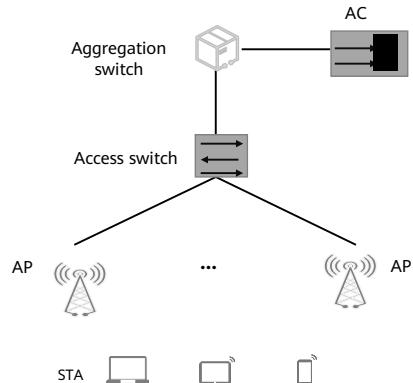
- Z-Wave is an emerging short-range wireless communications technology that uses radio frequency (RF) to transmit data.
- Advantages: simple structure, low rates, low power, low cost, and high reliability
- Disadvantages: Closed standard. The chip can only be obtained from Sigma Designs.



- Z-Wave is an emerging short-range wireless communications technology that uses radio frequency (RF) to transmit data. It features low power consumption, low costs, and high reliability. It operates at the 868.42 MHz (in Europe) to 908.42 MHz (in the US) frequency bands and uses binary or Gaussian frequency-shift keying (FSK) modulation. Z-Wave can transmit data at 9.6 kbit/s. Its transmission range is 30 m indoors and 100 m outdoors. It is perfect for broadband and narrowband scenarios.
- Z-Wave is used for residential and lighting control and status reading scenarios, such as metering, lighting and home appliance control, access control, anti-theft, and fire detection.
- Z-Wave was designed for smart home wireless control. A small data format is used for transmission, so 40 kbit/s is enough. In the early stage, data was even transmitted at 9.6 kbit/s. Compared with other similar wireless communications technologies, it has a relatively low transmission frequency, long transmission range, and price advantages.

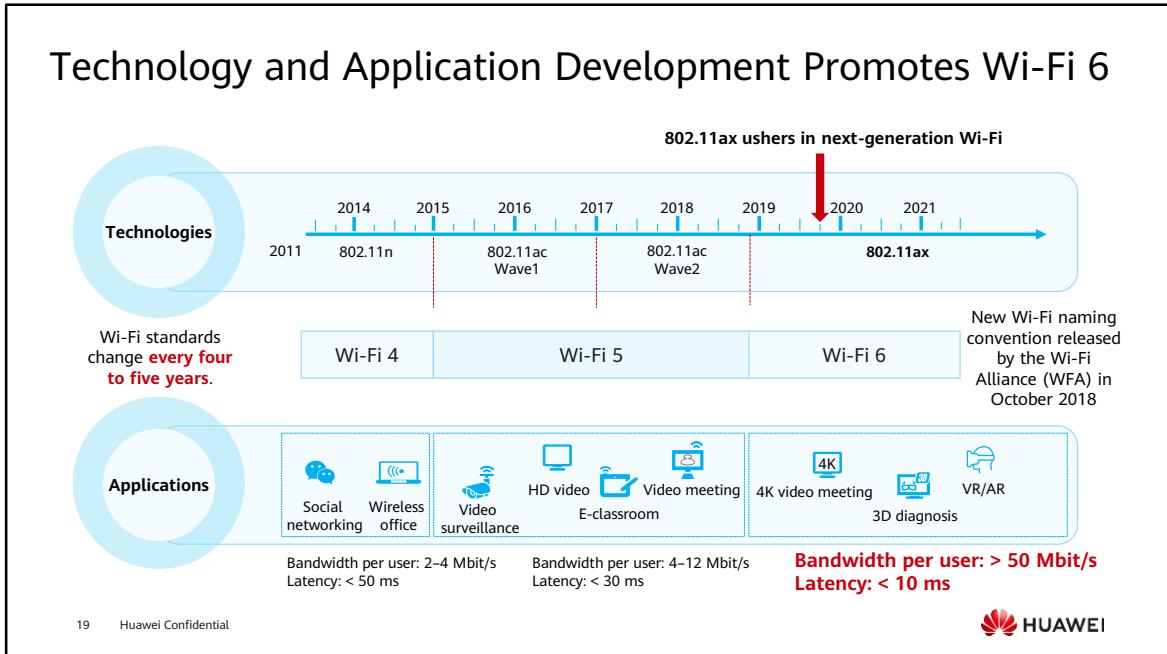
Wi-Fi

- Wi-Fi connects electronic devices to a wireless local area network (WLAN) using the 2.4 GHz ultra high frequency (UHF) or 5 GHz super high frequency (SHF) industrial, scientific, medical (ISM) radio frequency band. The latest Wi-Fi 6 supports a transmission rate of 9.6 Gbit/s and a latency of 20 ms.
- Advantages: wide coverage, fast data transmission
- Disadvantages: low transmission security, low stability, high power consumption, and weak networking
- Access controller (AC)
- Access point (AP)
- Station (STA) client



- 802.11ax supports all ISM frequency bands from 1 GHz to 6 GHz, including the currently used 2.4 GHz and 5 GHz frequency bands. It is compatible with 802.11a, 802.11b, 802.11g, 802.11n, and 802.11ac. It aims to support indoor and outdoor communication scenarios and improve spectral efficiency. Compared with 802.11ac, 802.11ax increases the actual throughput by four times, improves the nominal transmission rate by 37%, and reduces the latency by 75% in dense user environments.
- Main advanced functions:
 - It is compatible with 802.11a, 802.11b, 802.11g, 802.11n, and 802.11ac.
 - Orthogonal Frequency Division Multiple Access (OFDMA): enables channel sharing for uplink and downlink data transmission in strict environments to improve network efficiency and reduce latency.
 - Multi-user multiple-input multiple-output (MU-MIMO): allows more downlink data to be transmitted at a time, so that the access point can transmit data to more devices.
 - 160 MHz channel: The bandwidth is increased to provide higher performance with low latency.
 - 1024-quadrature amplitude modulation (1024-QAM): More data is encoded in the same amount of spectrum to improve the throughput of Wi-Fi devices.
 - Target wake-up time (TWT): significantly prolongs the battery life of Wi-Fi devices, such as IoT devices.
 - Transmit beamforming: supports a higher data rate within a given range, thereby providing a larger network capacity.
 - Four times the OFDM symbol duration.
 - Adaptive clear channel assessment (Adaptive CCA).
 - The security standard is upgraded to Wi-Fi Protected Access (WPA) 3.

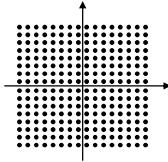
Technology and Application Development Promotes Wi-Fi 6



- Wi-Fi 5 cannot meet the low service latency and high bandwidth requirements of 4K/8K video meeting scenarios.
- Huawei Wi-Fi 6 uses SmartRadio intelligent application acceleration to reduce latency to as low as 10 ms, thereby meeting such requirements.

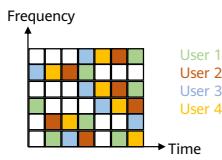
Features of Wi-Fi 6

High bandwidth



1024-QAM
8x8 MU-MIMO

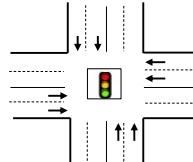
High concurrency



UL/DL OFDMA
UL/DL MU-MIMO

Rate of up to 9.6 Gbit/s
4x higher bandwidth

Low latency



OFDMA
Spatial reuse

20 ms service latency
Average latency reduced
by 30%

Low power



TWT
20 MHz-only

Target wakeup time (TWT)
30% less device power
consumption

- The bandwidth is increased by four times compared with the theoretical rate. Currently, the theoretical rate (in Wave2) of all Wi-Fi 5 products is 2.5 Gbit/s, and the theoretical rate of Wi-Fi 6 products is 9.6 Gbit/s.
- The number of concurrent users supported is increased by four times. In the actual test, Wi-Fi 5 supports 100 concurrent users and Wi-Fi 6 supports 400 concurrent users when the per capita bandwidth is 2 Mbit/s.
- The average latency of Wi-Fi 6 is about 20 ms, while the average latency of Wi-Fi 5 is about 30 ms. After Huawei SmartRadio intelligent application acceleration is used, the service latency can be reduced to 10 ms.
- Wi-Fi 5 does not support TWT.

IEEE 802.11 Standards and Wi-Fi Generations

Wi-Fi	802.11 Standard	Released In	Frequency Band	Theoretical Rate
-	802.11	1997	2.4 GHz	2 Mbit/s
-	802.11 b	1999	2.4 GHz	11 Mbit/s
-	802.11 a	1999	5 GHz	54 Mbit/s
	802.11 g	2003	2.4 GHz	54 Mbit/s
Wi-Fi 4	802.11 n	2009	2.4 GHz or 5 GHz	600 Mbit/s
Wi-Fi 5	802.11 ac Wave1	2013	5 GHz	3.47 Gbit/s
	802.11 ac Wave2	2015	5 GHz	6.9 Gbit/s
Wi-Fi 6	802.11 ax	2018/2019	2.4 GHz or 5 GHz	9.6 Gbit/s

- The first version of the IEEE 802.11 standard was released in 1997 to define the MAC and PHY layers.
- Since then, supplemental standards based on 802.11 have been defined. The most well-known ones influenced Wi-Fi evolution over generations: 802.11b, 802.11a, 802.11g, 802.11n, and 802.11ac.

Comparison of Short-Range Wireless Communications Technologies

	Bluetooth	Wi-Fi	ZigBee	Z-Wave
Frequency band	2.4 GHz	2.4 GHz, 5 GHz	868 MHz, 915 MHz, 2.4 GHz	868.42 MHz (Europe), 908.42 MHz (US)
Transmission rate	1–3 Mbit/s (24 Mbit/s over 802.11 links)	802.11b: 11 Mbit/s 802.11g: 54 Mbit/s 802.11n: 600 Mbit/s 802.11ac: 1 Gbit/s 802.11ax: 9.6 Gbit/s	868 MHz: 20 Kbit/s 915 MHz: 40 Kbit/s 2.4 GHz: 250 Kbit/s	9.6 Kbit/s or 40 Kbit/s
Transmission range	1–300 m	50–100 m	2.4 GHz band: 10–100 m	30 m (indoor) to 100 m (outdoor)
Scenarios	Data exchanged by nearby nodes (mouse, wireless headset, mobile device, and computer)	WLAN, high-speed Internet access at homes and other indoor places	Home automation, building automation, remote control	Smart home, monitoring and control

Contents

1. Wired Communications Technologies
2. **Wireless Communications Technologies**
 - Short-Range Wireless Communications Technologies
 - **Cellular Mobile Communications Technologies**
 - LPWA Communications Technologies
 - Comparison of Wireless Communications Technologies

2G

- Global System for Mobile Communications (GSM) is second-generation (2G) mobile communications technology. It is a standard developed by the European Committee for Standardization in 1992 to unify digital communications technologies into a network standard. This ensures communication quality and enables new services. The data rate of GSM is 9.6 Kbit/s.
- General Packet Radio Service (GPRS) is a mobile data service available to GSM mobile phone users. It is a data transmission technology of 2G mobile communications and an extension of GSM. GPRS provides data rates of 56–114 Kbit/s.

- Main 2G mobile phone communication technical specifications are as follows:
 - GSM: developed based on time division multiple access (TDMA), originated in Europe, and has been globalized.
 - Integrated Digital Enhanced Network (iDEN): developed based on TDMA. It is used by Nextel, a US telecommunications provider.
 - IS-136 (or D-AMPS): developed based on TDMA. It is the simplest TDMA system in the US and is used in the Americas.
 - IS-95 (or CDMA One): developed based on code division multiple access (CDMA). It is the simplest CDMA system in the US and is used in the Americas and some Asian countries.
 - Personal digital cellular (PDC): developed based on TDMA and used only in Japan.

3G

- 3G is third-generation mobile communications technology. It transmits data at high rate, such as voice and data simultaneously at hundreds of Kbit/s. 3G integrates wireless communications and multimedia communications such as the Internet.

Three current standards: code division multiple access 2000 (CDMA2000), wideband code division multiple access (WCDMA), and time division-synchronous code division multiple access (TD-SCDMA). High speed packet access plus (HSPA+), the latest WCDMA technology, supports a downlink rate of up to 42 Mbit/s.

- The standard name of 3G in the International Telecommunication Union (ITU) is International Mobile Telecom System-2000 (IMT-2000). It was first proposed by the ITU in 1985 and was called the future public land mobile telecommunication system (FPLMTS) at that time. Then, its name was changed to IMT-2000 in 1996. The name indicates that it works at the 2,000 MHz frequency band and the maximum rate is 2,000 kbit/s. It was put into commercial use around 2000. The 2G mobile communication system has gained a great success, and the friction between rapid user growth and the limited system capacity and services has been more obvious. The standardization of 3G mobile communications has started since 1997. It was named Universal Mobile Telecommunication System (UMTS) by the European Telecommunications Standards Institute (ETSI).

4G

- 4G is fourth-generation mobile communications technology. It includes two modes: long term evolution (LTE) time division duplexing (TDD) and LTE frequency division duplexing (FDD).
- Integrating 3G and WLAN, 4G can transmit data, high-quality audio, videos, and images at high rates. The download rate of 4G can exceed 100 Mbit/s, which is 25 times ADSL (4 Mbit/s), meeting almost all user requirements on wireless services. In addition, 4G can be deployed in areas not covered by the digital subscriber line (DSL) and cable television modem, and then expanded.

- 4G was officially named International Mobile Telecommunications-Advanced (IMT-Advanced) by the ITU.
- With carrier aggregation (CA), the maximum downlink rate of 4G+ is 330 Mbit/s or higher. Voice over LTE (VoLTE) enables 4G users to avoid falling back to 2G or 3G networks during voice calls and enjoy high-speed communication networks. In addition, the voice quality of VoLTE increases by 40% compared with that of 2/3G, and the delay decreases by 50% compared with that of 3G networks.
- ITU requires a rate of 1 Gbit/s for 4G. Obviously, 100 Mbit/s of time division long term evolution (TD-LTE) or 150 Mbit/s of frequency-division duplex long term evolution (FDD-LTE) do not meet this requirement. After continuous evolution, LTE-Advanced achieved 600 Mbit/s and LTE-Advanced Pro achieved 1 Gbit/s.

LTE Cat

- LTE User Equipment (UE) Category is the UE access capability level, or transmission rate level supported by a UE. For example, LTE Cat 4 indicates that the LTE network access capability level of the UE is 4.

Level	Downlink Rate (Mbit/s)	Downlink MIMO	Uplink Rate (Mbit/s)
1	10	1	5
2	50	2	25
3	100	2	50
4	150	2	50
5	300	4	75
6	300	2 or 4	50
7	300	2 or 4	150
8	1,200	8	600
9	450	2 or 4	50
10	450	2 or 4	100

- User equipment (UE) refers to a device used by a user to access or use a 4G LTE network. It can be a 4G mobile phone, smart watch, or tablet.
- An LTE category indicates the maximum transmission rate supported by UE, and the category must be supported by base stations of the carrier network. LTE Cat 4, Cat 6, and Cat 9 refer to access capability levels 4, 6, and 9 of UE LTE networks, respectively.
- Categories specify the theoretical upper limit of the UE transmission rates. The actual rates depend on network conditions (such as network optimization and the number of UEs), and can only be infinitely close to the upper limit instead of reaching the upper limit.

LTE Cat 1

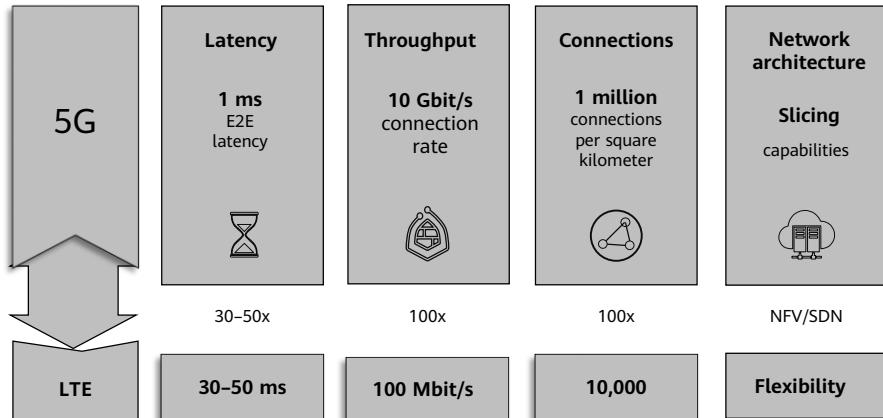
- LTE Cat 1 supports up to 10 Mbit/s downlink, enabling IoT devices with lower power consumption and costs to connect to LTE networks. LTE operators around the world deploy LTE networks based on 3GPP Release 8 or later. As such, operators can simply reconfigure parameters to permit the access of LTE Cat 1 terminals without network upgrade.
- Although LTE Cat 4 or later supports higher rates, the costs are relatively high for the IoT industry, so LTE Cat 1 is the most cost-effective solution.

5G

- 5G is fifth-generation mobile communications technology. Its theoretical transmission rate can reach 10 Gbit/s, which is 100 times 4G. With 5G, a 1 GB movie can be downloaded in eight seconds.
- International Telecommunication Union Radiocommunication Sector (ITU-R) defined three major 5G application scenarios in June 2015: Enhanced Mobile broadband (eMBB), Massive Machine-Type Communications (mMTC), and Ultra-reliable Low-latency Communications (uRLLC). It also defined eight capability specifications: throughput, latency, connection density, and spectral efficiency.
- The Ministry of Industry and Information Technology (MIIT) officially granted 5G licenses for commercial use to China Telecom, China Mobile, China Unicom, and China Broadnet on June 6, 2019, and announced the launch of 5G services on October 31, 2019.

- 5G was officially named IMT-2020 by ITU.
- Currently, eight key capability indicators of 5G have been defined: 20 Gbit/s peak rate, 100 Mbit/s data rate for users, three times higher spectral efficiency than IMT-A, 500 km/h mobility, 1 ms latency, 1 million connections per square kilometer (connection density), 100 times the energy efficiency of IMT-A, and 10 Mbit/s per square meter of the traffic density.

5G: Performance Indicators



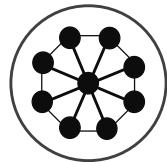
30 Huawei Confidential



- 4G cannot accommodate future applications and requirements, unable to meet their requirements for latency, throughput, and connections.

5G: Three Major Innovations

New architecture



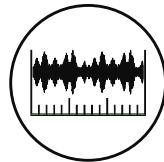
High-band and low-band aggregation
for better experience

New radio



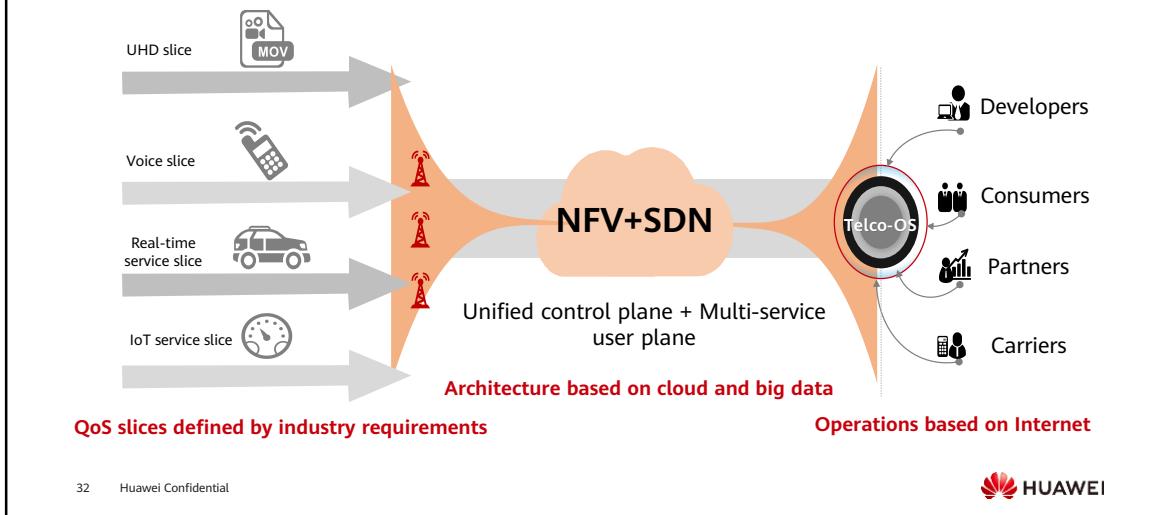
One physical network
for hundreds of industries

Full spectrum



Adaptability to various services
for higher spectral efficiency

5G Architecture: One Network for Hundreds of Industries



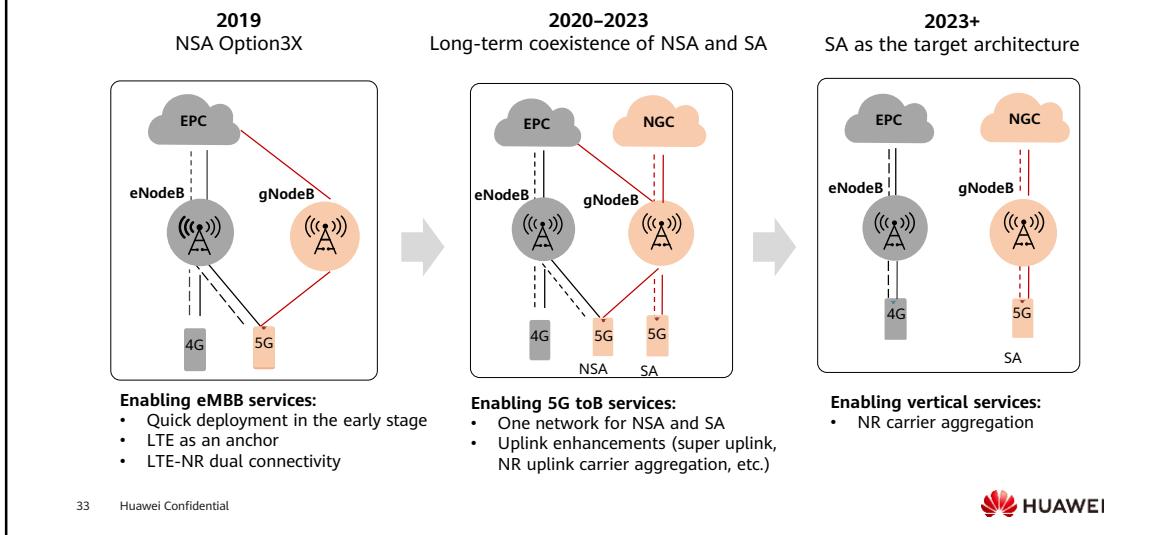
32 Huawei Confidential



- Network function virtualization (NFV) uses universal hardware (such as x86) and virtualization technologies to process software with many functions to reduce network device costs. Using software and hardware decoupling and function abstraction, it enables network device functions to be independent of dedicated hardware. Resources can be flexibly shared, new services can be quickly developed and deployed, and automatic deployment, auto scaling, fault isolation, and self-healing can be performed based on actual service requirements.
- Software-defined networking (SDN): An innovative network architecture proposed by the Clean Slate Program research team from Stanford University. It is an approach to implement network virtualization. Its core technology, OpenFlow, separates the control plane from the data plane of a network device, thereby network traffic can be flexibly controlled. This makes the network pipe more intelligent and provides a good platform for innovation of the core network and applications.
- Network slicing is to build multiple virtual end-to-end networks on a universal hardware using the network slicing technology. Each network has different functions to meet different service requirements. It slices physical resources, containing access, connection, computing, and storage.
- Network slices have three key features: customization, end to end, and isolation.
 - A network slice is an end-to-end network, covering the RAN, transport network, and core network. It requires a cross-domain slice management system.
 - Network slices require isolation of resources, security, and OAM. Different domains can use different technologies to achieve such isolation. For example, the core network uses virtualization technology.
 - Network slices can be customized to provide specific network functions and

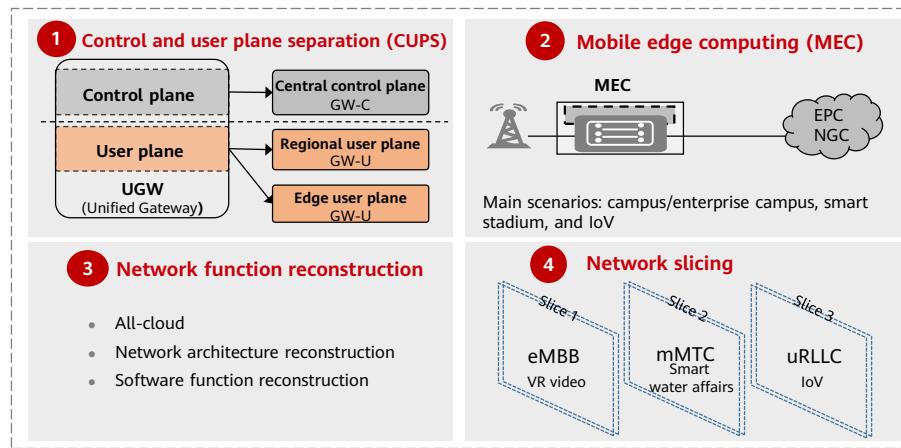
features.

5G Networking



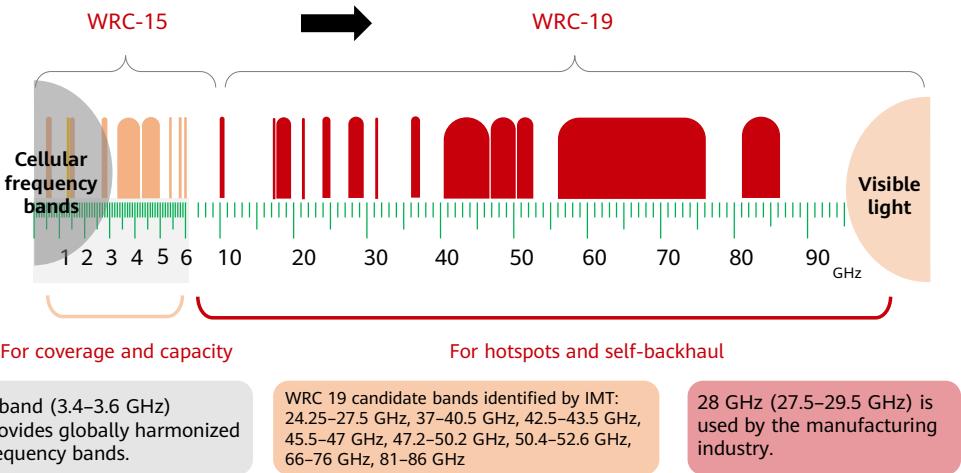
- Multiple stream aggregation (MSA): A terminal can use multiple base stations that use the same or different RATs for data transmission.
- Non-standalone (NSA): 4G core network + 4G/5G base station. There is an independent user plane but no independent control plane over NR.
- Standalone (SA): 5G core network + new 5G base station. There are independent control and user planes.
- eNB: evolved NodeB.
- gNB: next-generation NodeB.

NGC: Four Service-oriented Features



- Next Generation Core (NGC): 5G core network.
- Mobile edge computing (MEC): provides IT architecture-based cloud computing capabilities on the radio access network (RAN) close to mobile users. MEC provides application developers and content providers with extremely low latency, high bandwidth, and real-time information about wireless networks (such as user locations and base station loads), so that they can provide differentiated services for users.

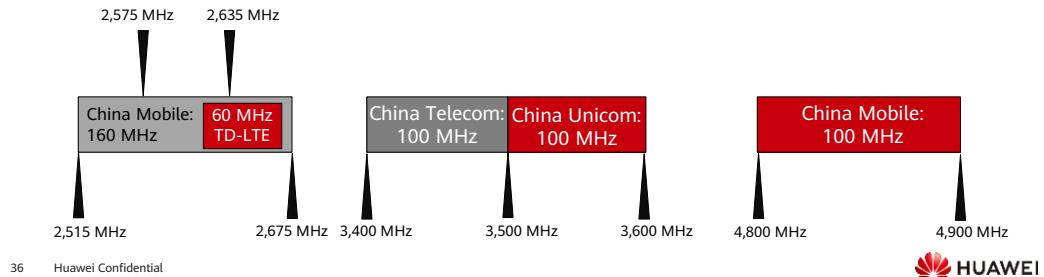
5G: Spectrum Aggregation



- Sub-3 GHz: < 3,000 MHz, mainly used for coverage.
- C-band: 3,000–6,000 MHz, mainly used for capacity expansion.
- mmWave: mainly used for hotspots and self-backhaul.
- Full spectrum: Low frequency bands below 6 GHz are mainly used, and high frequency bands are auxiliary.
 - < 6 GHz: for connectivity, coverage, mobility, and basic capacity.
 - > 6 GHz: for higher rate.

5G Frequency Allocation in China

- MIIT licensed the commercial use of low and mid frequency bands to the three major carriers:
 - China Telecom: 100 MHz (3,400–3,500 MHz)
 - China Telecom: 100 MHz (3,500–3,600 MHz)
 - China Mobile: 2,515–2,675 MHz and 4,800–4,900 MHz
 - 2,515–2,575 MHz, 2,635–2,675 MHz, and 4,800–4,900 MHz are new. 2,575–2,635 MHz are refarmed from 4G.



36 Huawei Confidential

 HUAWEI

eMBB

- 5G provides enhanced mobile Internet services which need faster transmission:
 - Virtual reality (VR), augmented reality (AR), and mixed reality (MR)



Everything you see is virtual.



You can distinguish between the physical and virtual worlds.



You cannot distinguish between the physical and virtual worlds.

Immersion

Interaction

Imagination

Panoramas

- Virtual reality (VR), augmented reality (AR), and mixed or mediated reality (MR)

mMTC

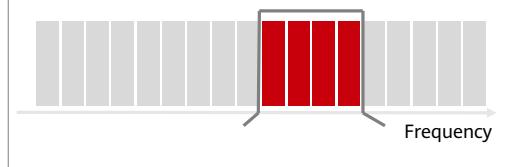
- 5G enables vast quantities of IoT connections.
 - Internet of everything (IoE) extends IoT.



NB-IoT has evolved to 5G NR.

ITU-R Working Party 5D#35e teleconference announces that 3GPP 5G (including NB-IoT) meets the IMT-2020 5G technical standard and is officially accepted as the ITU IMT-2020 5G technical standard.

NB-IoT can be embedded into 5G.

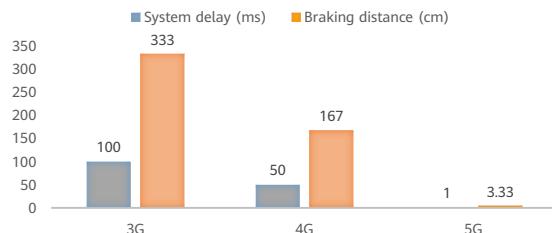


- As NB-IoT is officially incorporated into global 5G standards, the lifecycle and application scenarios of NB-IoT will be greatly expanded.
- NB-IoT has long become a mainstream technology in cellular IoT with its wide coverage, low power, low cost, and massive connection features. According to the latest statistics from Counterpoint, a market research institute, the number of global IoT cellular connections will exceed 5 billion by 2025, among which NB-IoT will contribute nearly half.

uRLLC

- 5G provides ultra-high reliability and ultra-low latency services.
 - For vehicle-to-everything (V2X), services such as assisted driving and autonomous driving have low latency requirements.
 - Smart healthcare and remote surgery have high timeliness requirements.

Relationship between system delay and braking distance



Comparison of Cellular Mobile Communications Technologies

	2G	3G	4G	5G
Licensed frequency band	(mainly 900 MHz)	(mainly 900 MHz and 1,800 MHz)	(1,800–2,600 MHz)	C-band mmWave
Transmission rate	GSM: 9.6 Kbit/s GPRS: 56–114 Kbit/s	TD-SCDMA: 2.8 Mbit/s CDMA2000: 3.1 Mbit/s WCDMA: 14.4 Mbit/s	Downlink Cat 6 and 7: 300 Mbit/s Cat 9 and 10: 450 Mbit/s	10 Gbit/s (Balong 5000 chips: Downlink rate: 4.6 Gbit/s Uplink rate: 2.5 Gbit/s)
Scenarios	POS, smart wearables	Vending machines, smart home appliances	Mobile devices, video surveillance	AR, VR, assisted driving, autonomous driving, and telemedicine

Contents

1. Wired Communications Technologies
2. **Wireless Communications Technologies**
 - Short-Range Wireless Communications Technologies
 - Cellular Mobile Communications Technologies
 - **LPWA Communications Technologies**
 - Comparison of Wireless Communications Technologies

Sigfox

- Sigfox uses Ultra Narrowband (UNB) for stable transmission with low power. Its radio link uses unlicensed ISM radio bands. Frequency usage varies by laws and regulations: 868 MHz in Europe, 915 MHz in the United States.
- Sigfox uses the ultra narrowband modulation technology to enable a base station to transmit messages to devices 50 km away and connect up to 1 million IoT devices.
- Advantages: free bands, low power, leaner network architecture.

- Sigfox is a quickly commercialized LPWA network technology. It uses the ultra narrowband technology to enable network devices to consume 50 or 100 microwatts of power for bidirectional or unidirectional communications. This protocol is owned by the Sigfox company, whose co-founder is Ludovic Le Moan. It aims to build a low-power, low-cost wireless network dedicated to IoT.

LoRa

- Long Range (LoRa) is a physical-layer-based technology that transmits data over networks using chirp spread spectrum (CSS) modulation. Transmission is bidirectional and complies with a series of open source standards. It is maintained and managed by the LoRa Alliance. LoRaWAN defines the specific communications protocol developed by Semtech and supported by IBM. LoRa can be used in automatic metering, smart home appliance, building automation, wireless warning and security systems, industrial monitoring and control, and remote irrigation systems.
- LoRa uses unlicensed spectrum.

- LoRa is one of the LPWA communications technologies. It is an ultra-long-distance wireless transmission solution based on the spread spectrum technology. It is adopted and promoted by Semtech. Instead of balancing between the transmission range and power consumption, this solution provides users with a simple system that can implement long transmission range, long battery life, and large capacity, thereby expanding the sensor network. Currently, LoRa mainly operates on free frequency bands around the world.

eMTC

- eMTC is a wireless IoT solution proposed by Ericsson. It is developed based on LTE for low rate, wide coverage, low power, and massive connection scenarios.
- Compared with NB-IoT, eMTC has higher rates (up to 1 Mbit/s) and power consumption and smaller coverage and capacity. eMTC also supports voice transmission.

NB-IoT

- NB-IoT is built based on cellular networks and consumes only 180 kHz bandwidth. It can be directly deployed on GSM, Universal Mobile Telecommunications System (UMTS), and LTE networks to reduce deployment costs and smoothen upgrading.
- NB-IoT is an emerging technology widely used in LPWA IoT markets worldwide. It features enhanced coverage, wide connections with low rates, costs, power consumption, and an optimal architecture.
- According to 3GPP Release 14, NB-IoT supports base station positioning and mobility within 80 km/h.
- In July 2020, NB-IoT was officially incorporated into 5G standards by ITU.

- NB-IoT features low power consumption, wide coverage, low cost, and large capacity. It can be widely used in different vertical industries, such as remote metering, asset tracking, smart parking, and smart agriculture.

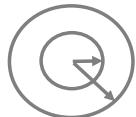
Key NB-IoT Features



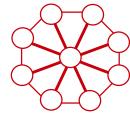
Low cost



Low power



Enhanced coverage

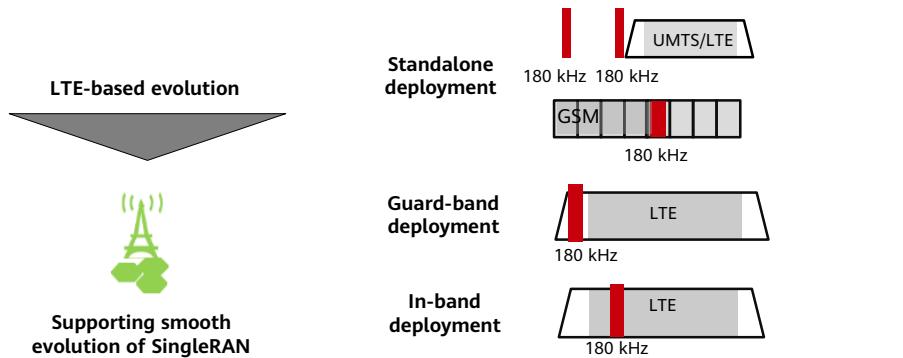


Massive connections

- Low cost: Huawei SingleRAN solution allows upgrade and reconstruction on legacy network devices, thereby reducing network construction and maintenance costs. NB-IoT chips are specifically designed for IoT devices. The chips apply only to narrowband and low rates and support only single-antenna transmission and half-duplex mode in compliance with IoT requirements. In addition, the signaling processing of NB-IoT chips is simplified. These reduce the terminal chip price to only several dollars.
- Low power: NB-IoT uses the power saving mode (PSM) and extended discontinuous reception (eDRX) features for IoT services where small packets are occasionally transmitted. With these features, a device enters the dormancy state immediately after sending data packets and wakes up only when data reporting is required again. An IoT device can be in the dormant state for up to 99% of total time, achieving super-low power consumption.
- Enhanced coverage: NB-IoT is designed for IoT, especially LPWA connections. It uses retransmission over the air interface and ultra-narrow bandwidths to provide an extra gain of over 20 dB compared with GSM. The gain means that fewer stations can cover wider areas with strong signal penetration (down to basements). Devices such as electricity or water meters in hard-to-reach areas can be covered. It can be used in pet tracking and other services that require broad coverage.
- Massive connections: The low prices of NB-IoT terminals allow mass NB-IoT terminal deployment, especially in industries using instruments. For the same eNodeB, NB-IoT can provide 50 to 100 times the number of connections provided by the existing wireless technologies. One sector can support 100 thousands of connections, with the support for low latency sensitivity, ultra-low device costs, low device power consumption, and optimized network architecture.

Low Cost: NB-IoT Deployment

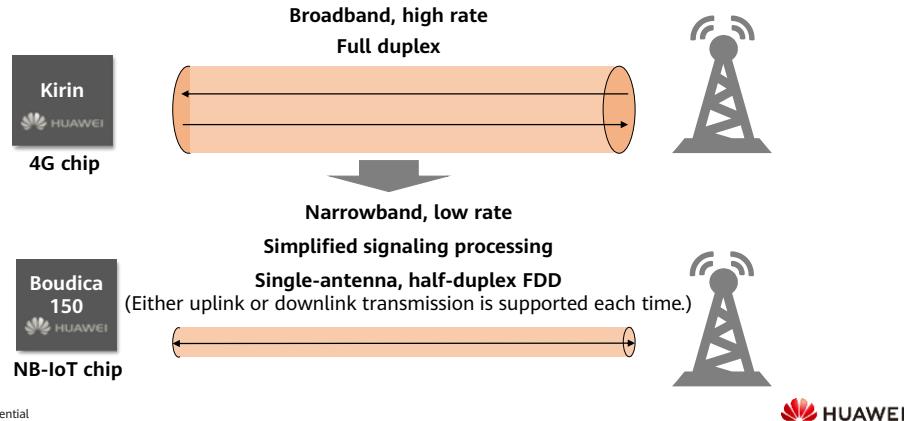
- NB-IoT supports LTE-based smooth evolution and flexible deployment that carriers need, and reduces their network construction and maintenance costs.



- NB-IoT is based on cellular networks and supports standalone deployment, guard-band deployment and in-band deployment to coexist with existing networks. It supports smooth evolution of SingleRAN. NB-IoT occupies a frequency band of about 180 kHz and can be directly deployed on existing GSM, UMTS, and LTE networks to reduce deployment costs and achieve smooth upgrade.
- It supports the following three deployment modes:
 - Standalone deployment: A separate band can be used. This mode can be used for the refarming of GSM frequency bands.
 - Guard-band deployment: Unused LTE edge bands can be used to deploy NB-IoT.
 - In-band deployment: Any resource block in LTE carriers can be used to deploy NB-IoT.
- Related deployment features:
 - The RF bandwidth is 180 kHz (uplink/downlink). (It is also described as 200 kHz considering the guard bands on both sides.)
 - Downlink: OFDMA. The subcarrier spacing is 15 kHz.
 - Uplink: SC-FDMA, single-tone: 3.75 kHz/15 kHz, multi-tone: 15 kHz.
 - Only half-duplex mode is supported.
 - The UE supports the indication of the single-tone and multi-tone capabilities.
 - The processing at the MAC/RLC/PDCP/RRC layer is based on the existing LTE procedures and protocols. The processing at the physical layer is optimized.
 - A separate synchronization signal is designed.

Low Cost: Communications Chips Designed for IoT

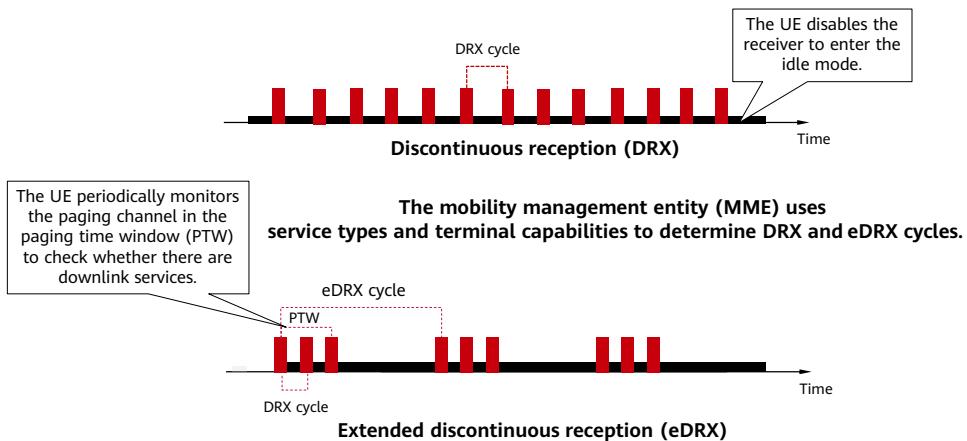
- Unnecessary physical hardware modules are cut off by simplifying functions and algorithms, thereby reducing chip costs.



48 Huawei Confidential

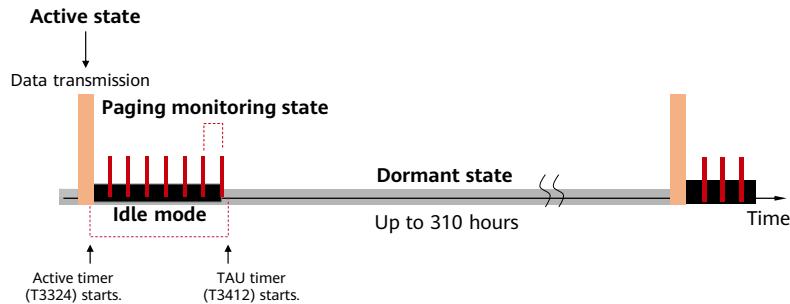
- The RF on the UE side is tailored and the mainstream NB-IoT chip supports one antenna. The antenna complexity is reduced and the algorithm is simplified.
- Full-duplex FDD is tailored to half-duplex, and the number of transceivers is reduced from two to only one in FDD-LTE.
- The low sampling rate and low rate can lower the demand for Flash and RAM.
- Low power consumption means low requirement of RF design and a small PA can achieve this.
- The physical layer and MAC layer of the protocol stack are simplified to reduce the computing capability of the chip and therefore lower the demand for the chip.

Low Power: DRX and eDRX



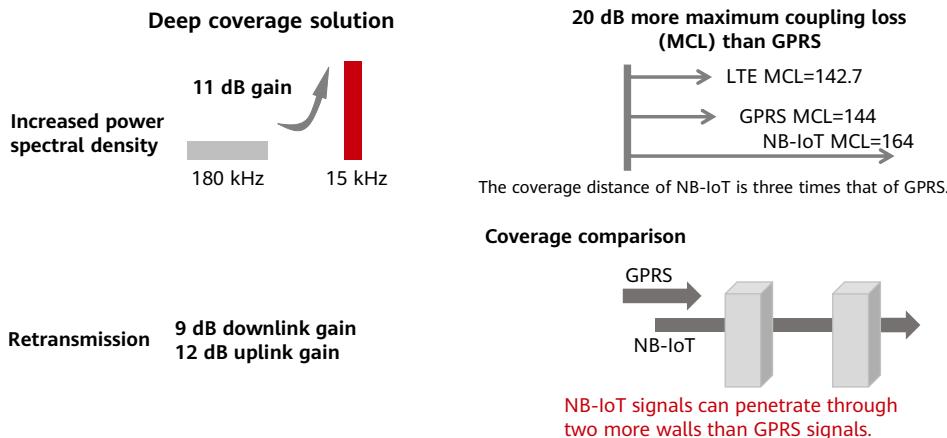
- The energy consumed by a device is related to the data volume and rate. The size of the data packet sent in a unit time determines the power consumption. Decoding the message of the paging channel consumes a large amount of energy.
- PTW: paging time window. The UE detects paging messages in the PTW period.
- DRX: discontinuous reception. In DRX, paging messages are not continuously received. DRX serves as an important solution for wireless communications terminals to save power, and is applicable to services that have high requirements for downlink service latency. A DRX cycle can be 1.28s, 2.56s, 5.12s, or 10.24s. It is determined by the carrier network settings.
- eDRX: extended DRX, an extended version of DRX. It requires capability negotiation between an NB-IoT terminal and the core network. The eDRX power saving technology further extends the sleep period of the terminal in idle mode and reduces unnecessary startup of the receiving unit. The maximum cycle of eDRX is 2.92 hours.

Low Power: PSM



- IoT devices have different communications requirements from mobile phones. In most cases, an IoT device only sends uplink data packets, and the device itself determines whether to send data packets. It does not need to stand by for calls from other devices. By contrast, a mobile phone is ready to respond to network-initiated call requests at any time.
- If IoT communications are designed in a 2G/3G/4G manner, a large amount of power is wasted in monitoring possible requests initiated by the network at any time, which is just the same as what a mobile phone behaves, failing to achieve low power consumption.
- With NB-IoT, an IoT device enters the dormant state immediately after sending data packets and wakes up only when data reporting is required again. An IoT device can be in the dormant state for up to 99% of total time, achieving super-low power consumption.
- NB-IoT reduces unnecessary signaling and prevents terminals from receiving paging messages in the PSM state to save power.
- A terminal in the PSM state is still registered with the network, but the signaling is unreachable. Therefore, the terminal stays in the dormant state for a longer time to reduce battery consumption. Downlink control signaling delivered from the service platform cannot reach the terminal in the PSM state immediately. The signaling is first buffered in the IoT platform.
- Both eDRX and PSM can be enabled on a terminal. When the active timer period is longer than the eDRX period, the terminal can enter the eDRX period to reduce power consumption to the maximum extent.

Enhanced Coverage: Increased Power Spectral Density and Time Domain Retransmission



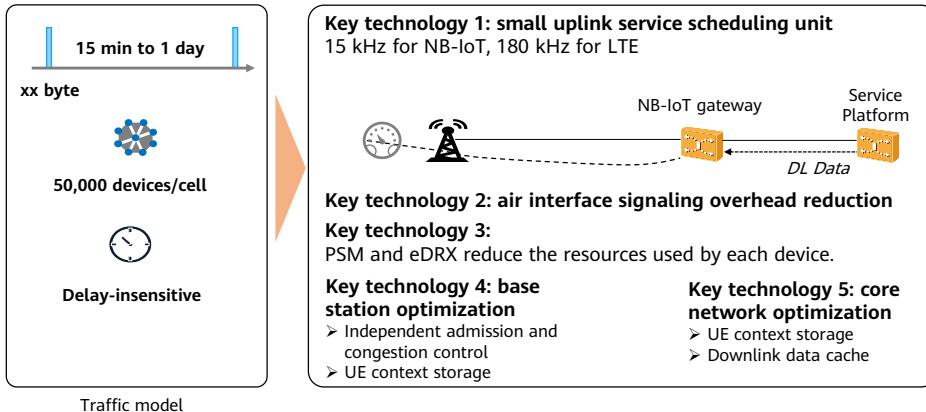
51 Huawei Confidential



- MCL is used to evaluate the extent of coverage (or penetrated range). The larger the value, the larger the extent of coverage (or penetrated range).
- The existing access technologies cannot meet the requirements of deep coverage based on the IoT deployment characteristics. For the deployment characteristics of IoT services such as smart water meters and smart parking, 3GPP TS45.820 proposed that LPWA must meet the requirements of MCL enhanced by 20 dB for GSM/GPRS/LTE networks.
- Compared with LTE and GPRS base stations, NB-IoT increases the gain by 20 dB so that hard-to-reach areas such as underground garages, basements, and underground pipelines can be covered.
- Increased power spectral density (PSD):
 - The uplink and downlink physical channel formats and modulation specifications are redefined so that uplink and downlink control information and service information can be transmitted in narrower bandwidths compared with LTE. The PSD gain increases when the transmit power remains the same, reducing the demodulation requirement on the receiver.
- Retransmission: The retransmission encoding scheme is introduced to improve transmission reliability when the channel condition is extremely unfavorable.

Massive Connections: Less Air Interface Signaling Overhead and Resources

A capacity of over 50,000 users



- First, NB-IoT base stations are designed based on the IoT traffic model. The IoT traffic model is different from the mobile phone traffic model. In the IoT traffic model, there are a large number of terminals, the packet sent by each terminal is small, and the packet transmission is delay-insensitive. The 2G/3G/4G base stations are designed in such way that UEs can run services simultaneously with delay requirements satisfied. The number of connections or the number of admitted UEs is limited to about 1,000. In contrast, NB-IoT services are delay-insensitive, and NB-IoT base stations can be designed in such a way that more terminals can be admitted and more terminal contexts can be saved. An NB-IoT cell can serve about 50,000 terminals at the same time, with a large number of them in the dormant state. The contexts of these terminals are maintained by the base station and core network. Once data needs to be sent, the terminals can quickly enter the active state.
- In addition, the scheduling granularity is large in 2G/3G/4G and is much smaller in NB-IoT because NB-IoT is narrowband-based. When resources are the same, the resource usage of NB-IoT is higher than that of 2G/3G/4G. For the same coverage gain requirements, there are no or less retransmission times and higher spectrum efficiency in NB-IoT scenarios when compared with those in 2G/3G/4G scenarios.

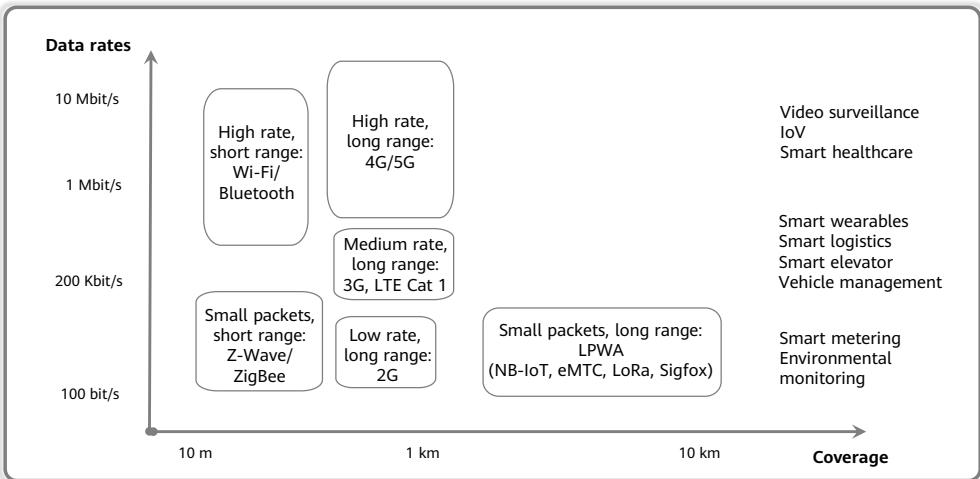
Comparison of LPWA Technologies

	Sigfox	LoRa	NB-IoT	eMTC
Frequency band	Sub-GHz unlicensed frequency bands	Sub-GHz unlicensed frequency bands	Mainly sub-GHz licensed frequency bands	Sub-GHz licensed frequency bands
Transmission rate	100 bit/s	0.3–5 Kbit/s	< 250 Kbit/s	< 1 Mbit/s
Transmission range	1–50 km	1–20 km	1–20 km	2 km
Scenarios	Smart home appliances, smart electricity meters, mobile healthcare, remote monitoring, and retail	Smart agriculture, intelligent buildings, and logistics tracking	Water meters, parking, pet tracking, garbage disposal, smoke alarm, and retail devices	Shared bicycles, pet collars, POS, and smart elevators

Contents

1. Wired Communications Technologies
2. **Wireless Communications Technologies**
 - Short-Range Wireless Communications Technologies
 - Cellular Mobile Communications Technologies
 - LPWA Communications Technologies
 - Comparison of Wireless Communications Technologies

Comparison of Wireless Communications Technologies



55 Huawei Confidential



- Network capacity requirements of the future tens of billions of IoT connections will be different. It is estimated that 10% of nodes require high bandwidth and high-rate transmission technologies, such as 4G and 5G. 30% of nodes require medium-rate transmission technologies, such as Cat 1 and eMTC. 60% of nodes require low-rate transmission technologies, such as NB-IoT and LoRa.
- Compared with other technologies, Cat 1 has the following advantages:
 - Cat 1 has better communication functions than those of NB-IoT. NB-IoT is suitable for scenarios where only a small amount of data is transmitted and the terminals are fixed. Its typical use cases include water, electricity, and gas meters. Cat 1 transmits larger volumes of data and has good mobility and voice functions.
 - Cat 1 has lower costs than eMTC. Cat 1 enables seamless access to existing LTE networks without upgrading the hardware and software of base stations. 4G signals are available in places where Cat 1 is available.
 - Cat 1 has lower costs than Cat 4. Cat 1 with an optimized system is better integrated and has leaner hardware architecture as well as more economic peripheral hardware.

Quiz

1. (True or false) All NB-IoT networks are deployed on licensed sub-GHz bands.
2. (Single-answer question) Which of the following is a wired communications technology?
 - A. 5G
 - B. NB-IoT
 - C. PLC
 - D. ZigBee

- Answers:
 - F
 - C

Summary

- In this section, you have learned the features and applications of common wired and wireless IoT communications technologies. Wireless communications technologies can be further divided into short-range, cellular mobile, and LPWA communications technologies.

Recommendations

- Huawei Cloud IoTDA product documentation:
 - <https://support.huaweicloud.com/intl/en-us/iothub/index.html>

Acronyms and Abbreviations

- FDD: Frequency Division Duplexing
- IEEE: Institute of Electrical and Electronics Engineers
- LPWA: Low Power Wide Area

Thank you.

把数字世界带入每个人、每个家庭、

每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.



IoT Communications Protocols



Foreword

- To connect all things, in addition to communication technologies at the network layer, communications protocols play an essential role.
- Traditional Internet standards and protocols are not a good fit for IoT. Data at the IoT sensing layer is heterogeneous, and interfaces and technical standards vary by device. IoT devices are usually inexpensive and have limited resources and limited power supplies. They require lightweight, low-power, and secure communications protocols.

Objectives

- Upon completion of this course, you will understand:
 - The OSI reference model.
 - The concepts, advantages, and disadvantages of different network topologies.
 - How HTTP works.
 - How AMQP works.
 - How MQTT works.
 - How CoAP works.

Contents

1. Basics of Network Communications

- Communications Protocols
 - Network Topologies

2. Common IoT Protocols

Overview of Communications Protocols

- Communications protocols are rules and conventions that entities must follow to establish communications or provide services. A communications channel connects devices at different or same locations to form a system. In this system, devices must use the same language to share information and resources with each other.
- Information interactions include at least three parts: **transmitters**, **transport channels**, and **receivers**.

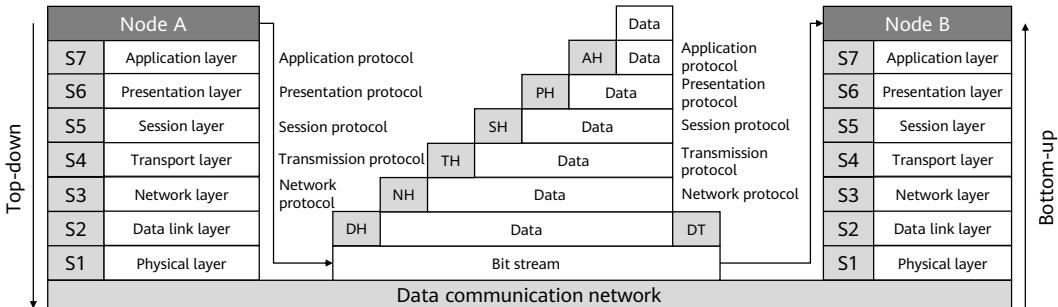


Elements of Communications Protocols

- A communications protocol defines how a message is encapsulated into data packets. Data packets are then converted into bit streams for transmission over the network. A receiver receives and parses the data packets based on the protocol to obtain the message.
- A communications protocol consists of three elements:
 - **Semantics** specifies control information and actions required for communication entities.
 - **Syntax** specifies the format and structure of data or control information exchanged between communication entities.
 - **Timing** specifies the response relationship between communication entities, including speed matching and sequencing.

OSI Reference Model

- The open system interconnection (OSI) reference model is a standard framework proposed by the International Organization Standardization (ISO) to connect different computers around the world in a network.

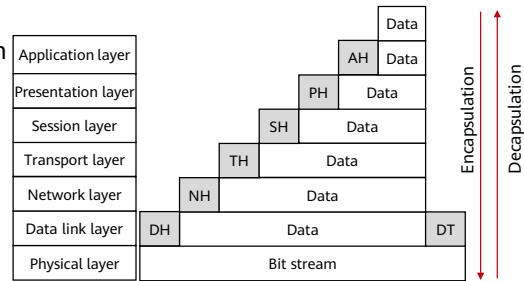


- The physical layer establishes, maintains, or cuts off physical connections between two network nodes for data transmission.
- The data link layer transmits data frames between network nodes. It encapsulates bit streams at the physical layer into frames and transmits the frames to the network layer. It also breaks down frames at the network layer into bit streams and transmits the bit streams to the physical layer.
- The network layer selects the best possible route for packets or segments through the communication subnet based on the routing algorithm.
- The transport layer allows reliable data transmission between network nodes. It isolates the application layer from other data transmission layers, converts data into the format required for network transmission, checks transmission results, and corrects failed transmissions.
- The session layer is responsible for negotiation and connection between applications or processes of network nodes. It not only establishes proper connections, but also authenticates identities of both session parties.
- The presentation layer ensures that commands and data of an application can be understood by other nodes on the network. This simplifies communications and makes communications device independent.
- The application layer provides services for users and completes user tasks on the network.
- OSI data transmission process: To send data to node B, node A encapsulates data, and protocol information is added to the data at each layer. Then, data is converted into bit streams consisting of 0 and 1 and transmitted to network connection media. After node B receives the data, it breaks down the

encapsulated data at each layer to complete a communication process.

Data Encapsulation and Decapsulation

- **Encapsulation** is the process of adding data to a protocol packet header when it is transmitted from the **top to bottom** layers over the network. **Decapsulation** is the process of removing encapsulated information from received data and transmitting the data from the **bottom to top** layers based on the encapsulated information.
- Data is encapsulated and decapsulated:
 - Inside the protocol.
 - During network transmission.



Contents

1. Basics of Network Communications

- Communications Protocols
- Network Topologies

2. Common IoT Protocols

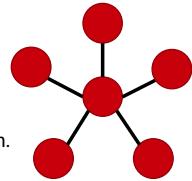
Network Topology Overview

- A network topology refers to the physical (wired) or logical (wireless) layout of various devices connected through transmission media. If two networks have the same connection structure, they have the same network topology, although the two networks may have different physical cable connections and node distances.
- There are different types of network topologies. Each type has its own advantages and disadvantages.



Star Topology

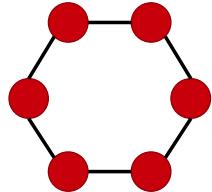
- A star topology is a star-shaped network topology in which each device in the network is individually connected to a central node. It is easy to implement, but is labor intensive to set up and maintain.
- Advantages:
 - Its simple structure makes it easy to manage and debug.
 - It is easy to control, add, or delete a device.
 - Centralized management facilitates service provisioning and network reconfiguration.
 - Each device directly connects to the central node, facilitating fault detection and isolation.
- Disadvantages:
 - The cable utilization is low. Each cable is used only by the central node and a single device.
 - The central node is prone to getting overloaded. If the central node fails, the entire network will fail, so the central node needs to be extremely reliable and includes redundant design features.
 - Installation and maintenance are expensive because a lot of cables are needed.



- A star topology has the following features: simple structure, easy management, simple control, easy network construction, lower network latency, and low transmission error. Its disadvantages include high costs, low reliability, and poor resource sharing capability.

Ring Topology

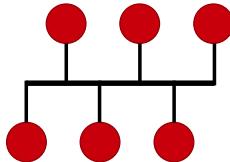
- A ring topology is a type of network topology where each node is connected to two other devices on either side and transmission cables form a circular ring. Data is transmitted in one direction along the ring from one node to another.
- Advantages:
 - Its transmission rate is high.
 - Data flows only in one direction, and there is only one path between two nodes. This simplifies path selection.
 - Each node in the circular ring is bootstrapped, so the control software is simple.
- Disadvantages:
 - Response delay: Too many nodes in the ring will slow down transmission and cause network response delay.
 - Weak scalability: A closed ring topology is less scalable than a star topology. To add or move nodes, the entire network has to shut down.
 - Difficult to maintain: If a node is faulty, the entire network will shut down.



- A ring topology has the following features: Data flows only in one direction, and there is only one path between two nodes. This simplifies path selection. Each node in the circular ring is bootstrapped, so the control software is simple. If there are too many nodes in the ring, the transmission will slow down and network response will be delayed. The ring is closed, so it is difficult to expand it. A ring topology has low reliability. If a node is faulty, the entire network will shut down. It is difficult to maintain and locate faults on nodes.

Bus Topology

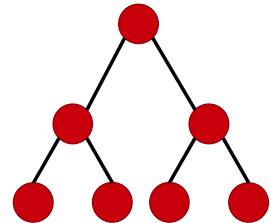
- A bus topology is a type of network topology in which nodes are directly connected to a main half-duplex line, so it is also called linear bus topology. In a bus network, each host receives all network traffic from the main line, and the transmission priority for traffic generated by each is the same.
- Advantages:
 - It is very easy to connect computers or peripherals to a linear bus.
 - It uses fewer cables than a star topology does.
 - It is suitable for small-scale networks.
 - It is inexpensive to implement.
- Disadvantages:
 - If the main cable is disconnected, the whole network fails.
 - If the entire network is interrupted, it will be difficult to locate faults.
 - Too many devices on the network will slow down transmission.



- A bus topology has a simple structure and good scalability. When you want to add a node, you only need to add a branch interface to the bus to connect to the new node. You can also expand a bus when it is overloaded. A bus topology requires fewer cables, is easy to install, and uses simple and reliable devices, but it is difficult to maintain and locate node faults.

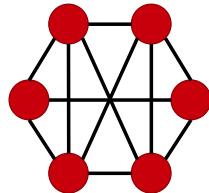
Tree Topology

- A tree topology is used for hierarchical management and control. Compared with a star network, a tree network has shorter communication lines, lower costs, and is easier to replicate, but its structure is more complex than that of a star network. A failure on any node or cable other than the leaf nodes affects all of the branches below it on the network.
- Advantages:
 - Expansion: It is easy to add new nodes and branches to a tree network.
 - Fault isolation: It is easy to isolate a branch with a faulty node or cable from the entire system.
- Disadvantages:
 - High dependency on the root node: If the root node is faulty, the entire network will be affected. This reliability issue is similar to that of a star topology.



Mesh Topology

- A mesh topology is a type of network in which information and control instructions are transmitted between network nodes through dynamic routing. In a mesh network, all nodes are connected to each other. If a node fails, data will just hop to the destination following a new route.
- Advantages:
 - Easy deployment and installation
 - High stability
 - Flexible structure
 - High bandwidth
 - It can be used outdoors.
- Disadvantages:
 - The structure is complex, expensive, and difficult to maintain.



Contents

1. Basics of Network Communications

2. Common IoT Protocols

- HTTP
- AMQP
- MQTT
- CoAP
- Protocol Comparison

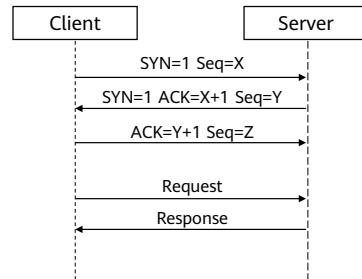
HTTP Overview

- Hypertext Transfer Protocol (HTTP) is an application layer protocol for distributed, collaborative, and hypermedia information systems. It uses a **request/response** model between a client (user) and server (website) based on Transmission Control Protocol (TCP).
- For mobile Internet, HTTP is the most popular application layer protocol. HTTP can also be used for IoT, but it is not suitable for IoT nodes with limited compute and storage resources.



How HTTP Works

- HTTP uses a client/server architecture and is connection-oriented. A typical HTTP transaction processing process is as follows:
 - The client establishes a connection with the server.
 - The client sends a request to the server.
 - The server accepts the request and returns the requested file.
 - The connection between the client and server is closed.



- The HTTP connection between the client and server is a one-time connection. Only one request is processed during each connection. When the server returns a response to a request, the connection is closed immediately and a connection will be re-established for the next request. The WWW server processes requests from thousands of users on the Internet and the number of connections are limited, so it does not keep a connection in the waiting state. Releasing a connection in a timely manner can greatly improve server execution efficiency.
- HTTP is a stateless protocol. The server does not retain any state of completed user transactions. This greatly reduces the server's memory pressure and enables fast responses. HTTP is an object-oriented protocol. Any type of data object can be transmitted. It identifies the content and size of the transmitted data based on the data type and length, and allows data compression for transmission. After a user defines a hypertext link in a Hypertext Markup Language (HTML) document, the browser connects to the specified server using TCP/IP.
- HTTP supports persistent connections. In HTTP/0.9 and HTTP/1.0, a connection is closed after a request/response pair. HTTP/1.1 introduced the keep-alive mechanism that enables connections to be reused for multiple requests. Such persistent connections significantly reduce request latency because the client does not need to renegotiate the TCP three-way handshake connection after sending the first request. Another positive effect is that connections become faster over time due to the slow start mechanism of TCP.

HTTP Request Methods

- HTTP/1.1 defines eight types of request methods (also called actions) to perform different operations on specified resources. It also allows request methods to be extended in other protocols or specifications.

Request Method	Description
GET	Requests a representation of the specified resource.
HEAD	Similar to GET, it requests the specified resource from the server, but the server does not return a response body.
POST	Submits data to the specified resource for the server to process.
PUT	Uploads the latest content to the specified resource.
DELETE	Requests the server to delete the resource identified by the Request-URI.
TRACE	Echoes back the request received by the server. It is used for testing or diagnosis.
OPTIONS	Requests the server to return the HTTP methods supported by the specified resource.
CONNECT	Establishes a tunnel for bidirectional communications between the client and requested resource.

Contents

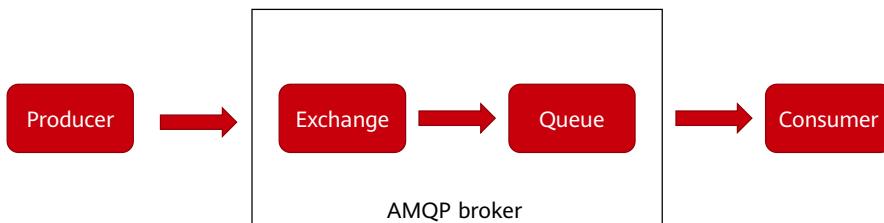
1. Basics of Network Communications

2. Common IoT Protocols

- HTTP
- AMQP
- MQTT
- CoAP
- Protocol Comparison

AMQP Overview

- Advanced Message Queuing Protocol (AMQP) is an open standard application layer protocol for message-oriented middleware. It provides unified messaging services.
- AMQP uses a publish/subscribe model. An exchange in the message broker distributes received messages to different queues based on message topics so that subscribers can receive the messages.
- In IoT applications, AMQP is mainly used for communications between mobile devices and data centers.



- AMQP workflow: A producer publishes a message to an exchange. The exchange distributes the message to the queue bound to the exchange based on routing rules. Finally, the AMQP broker delivers the message to the consumer who subscribes to the queue, or the consumer fetches the message on demand.
- There can be multiple producers, exchanges, queues, and consumers. Because AMQP is a network protocol, producers, consumers, and message brokers in this process can exist on different devices.
- When publishing a message, a producer can specify various message attributes (message meta-data). Some of these attributes may be used by the broker, however, other attributes are completely opaque to the broker and are only used by applications that receive the message.
- When the network is unreliable or a consumer application fails during message processing, messages that fail to be processed are lost. In this case, AMQP provides message acknowledgments. After a message is delivered from a queue to a consumer, the message is not deleted from the queue until it receives an acknowledgment from the consumer.
- In some situations, when a message cannot be routed (cannot be distributed from an exchange to a queue), the message may be returned to the producer and discarded. Or when the message broker implements an extension, a message is placed into a dead letter queue. The producer chooses how to handle these

special situations using certain parameters.

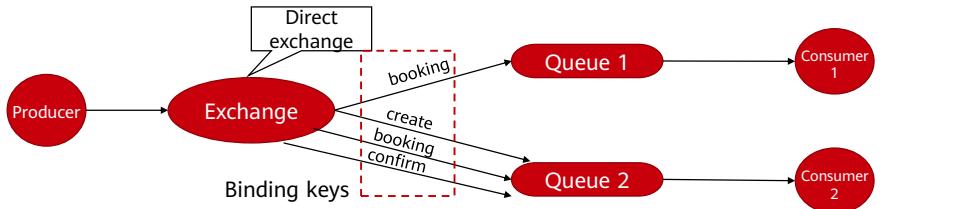
AMQP Core Components (1)

- Producer
 - Producers publish messages to the connected AMQP broker.
- Connection
 - The network connection between the AMQP broker and applications (including producers and consumers) uses TCP/IP three-way handshake and four-way handshake.
 - AMQP connections are usually persistent connections and use TCP to ensure reliable delivery. AMQP uses authentication and provides Transport Layer Security (TLS) protection. When an application no longer needs to be connected to the AMQP broker, it closes the AMQP connection instead of the TCP connection.



AMQP Core Components (2)

- Exchange
 - An exchange receives messages and routes them to the bound queue. It does not store messages.
 - An exchange is an AMQP entity used to send messages. An exchange takes a message and routes it into zero or more queues. The routing algorithm used depends on the exchange type and binding rules.
 - Exchanges can be either durable or transient. Durable exchanges survive broker restarts. Transient exchanges do not (they have to be redeclared after a broker restarts).



23 Huawei Confidential



- A message published by a producer contains the exchange type. The routing rules vary according to the exchange type declared in the message, so messages are routed into queues in different ways.
- Exchange types:
 - Direct exchange
 - Fanout exchange
 - Topic exchange
 - Headers exchange

AMQP Core Components (3)

- Queue
 - Queues in AMQP are similar to queues in other message and task queue system: they store messages to be consumed by applications.
- Consumer
 - Consumers consume messages. In AMQP, consumers can obtain messages in either of the following ways:
 - **Messaging middleware delivers messages to consumers.**
 - **Consumers fetch messages.**
 - When multiple consumers subscribe to the same queue, messages in the queue are still consumed by only one consumer (not by multiple consumers).

Contents

1. Basics of Network Communications

2. Common IoT Protocols

- HTTP
- AMQP
- **MQTT**
- CoAP
- Protocol Comparison

MQTT Overview

- Message Queuing Telemetry Transport (MQTT) is a protocol designed for sensors or controllers that have limited computing capabilities and work on low-bandwidth, unreliable networks.
- It is a **TCP-based** lightweight messaging protocol for IoT. MQTT clients **publish and subscribe** to messages through the broker.



- Due to the particularity of IoT, MQTT complies with the following design principles:
 - Simplify functions.
 - Provide a publish/subscribe model for message transmission between sensors.
 - Allow users to dynamically create topics with no O&M costs.
 - Minimize the transmission volume for efficiency.
 - Take low bandwidth, high latency, and unstable network into consideration.
 - Enable control of continuous sessions.
 - Be suitable for clients with low computing power.
 - Provide quality of service (QoS).
 - Maintain flexible types and formats of transmitted data.

MQTT Core Components

Client	Server
<p>A client is a program or device that uses MQTT. It can:</p> <ul style="list-style-type: none">• Establish a network connection with the server.• Act as a publisher to send messages.• Act as a subscriber to receive messages.• Unsubscribe from a server to delete message requests.• Close a network connection with the server.	<p>A server is a program or device that acts as a broker between publisher and subscriber clients. It can:</p> <ul style="list-style-type: none">• Receive network connection requests from clients.• Receive messages published by a client.• Process subscription and unsubscription requests from clients.• Forward messages that match the topic to which a client subscribes.• Close a network connection with the client.

MQTT Network Connection & Application Message

- Network Connection refers to a construct provided by the underlying transport protocol that is being used by MQTT.
 - It connects a client to a server.
 - It provides the means to send ordered, lossless streams of bytes in both directions.
- Application Message
 - The data carried by the MQTT protocol for an application. An application message transported by MQTT contains the **payload data, QoS, a collection of properties**, and **the topic name**.

MQTT QoS

- MQTT supports three levels of quality of service (QoS), which guarantees the reliability of message delivery in different scenarios.

QoS 0

QoS 0: Messages are delivered at most once based on the capabilities of the underlying network. If the receiver does not send a response, the sender does not perform a retry. Messages arrive at the receiver either once or not at all.

QoS 1

QoS 1: Messages are delivered at least once. This QoS level ensures that messages arrive at the receiver at least once. A QoS 1 PUBLISH packet has a packet identifier in its variable header and is acknowledged by a PUBACK packet.

QoS 2

QoS 2: Messages are delivered exactly once. It is the highest QoS level and is used when neither message loss nor duplication is acceptable. Using QoS 2 will increase overhead. A QoS 2 message has a packet identifier in its variable header. The receiver of a QoS 2 PUBLISH packet acknowledges receipt with a two-step acknowledgement process.

- A client or server can be a sender or receiver. The delivery protocol is involved only when an application message is delivered from a single sender to a single receiver. When a server is delivering an application message to multiple clients, each client is treated independently. The QoS level used to deliver an application message outbound to a client may differ from that of the inbound application message.

MQTT Topic Names and Topic Filters

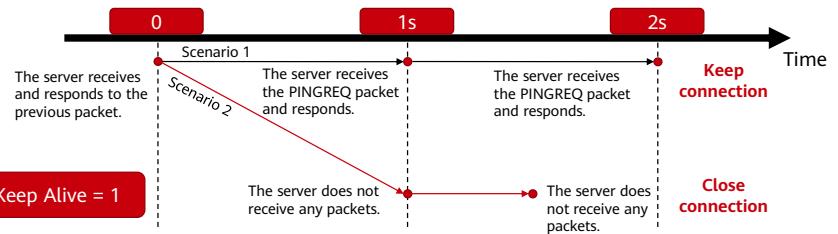
- MQTT application messages are classified by topic name. Topic names do not need to be created. They can be used directly. A topic filter is a topic that contains wildcard characters. It enables a client to subscribe to multiple topics at a time. Different wildcard characters can be used in topic filters:

Topic level separator (/)	Multi-level wildcard (#)
Separates a topic name into multiple levels. Examples: <ul style="list-style-type: none">sport/tennis/player1sport/tennis/player2	Matches any number of levels within a topic. It must be the last character in the topic filter and follow a topic level separator. For example, sport/tennis/player1/# includes: <ul style="list-style-type: none">sport/tennis/player1sport/tennis/player1/rankingsport/tennis/player1/score/wimbledon
Single-level wildcard (+)	Special wildcard (\$)
Matches only one topic level. It can be used at any level in the topic filter, including the first and last levels. It must occupy an entire level of the filter and can be used with a multi-level wildcard. Examples: <ul style="list-style-type: none">+/tennis/#sport+/player1/ranking	Topics starting with a dollar sign (\$) are treated special. They are usually not included in the subscription. A server prevents clients from using such topic names to exchange messages with other clients. These topics are reserved as internal features of the MQTT proxy server. Example: <ul style="list-style-type: none">\$SYS/

- The following rules apply to topic names and topic filters:
 - Topic names and filters must contain at least one character.
 - Topic names and filters are case sensitive.
 - Topic names and filters can contain spaces.
 - A leading or trailing a slash (/) creates a distinct topic name or filter.
 - A topic name or filter consisting only of a slash (/) is valid.
 - Topic names and filters must not include a null character.
 - Topic names and filters are UTF-8 encoded strings. They must not be more than 65,535 bytes after encoding.
- There is no limit to the number of levels in a topic name or filter, other than that imposed by the overall length of a UTF-8 encoded string.

MQTT Keep Alive

- MQTT is based on TCP, which is connection-oriented. In some cases, a TCP **half-open connection** may occur.
- MQTT provides a **Keep Alive** mechanism for a client and server to determine whether a half-open connection occurs and to close the connection if it occurs.
- Keep Alive is a two-byte integer in the CONNECT packet. It indicates the maximum time interval allowed between when a client finishes transmitting an MQTT packet and when it starts sending the next packet. If the Keep Alive value is non-zero and no other MQTT control packets are transmitted, the client must send a PINGREQ packet. If the server does not receive any packets within **1.5** times the length of the interval, it must close the network connection with the client.



- A half-open connection refers to the situation where one end does not establish or has closed a connection while the other end still maintains the connection. In this case, one end of the half-open connection may continuously send data to the peer end, but the data will never be received by the peer end.
- A client can send a PINGREQ packet at any time, irrespective of the Keep Alive value, and check the corresponding PINGRESP packet to determine whether the network and server are available.
- If the client does not receive the PINGRESP packet within a reasonable period of time after sending the PINGREQ packet, the client should close the network connection with the server.
- The Keep Alive value 0 can be used to disable the Keep Alive mechanism. If the Keep Alive value is 0, a client is not obligated to send MQTT control packets based on any particular plan.

MQTT Control Packet

- Structure:
 - MQTT works by exchanging a series of control packets in a defined way. A control packet consists of up to three parts:

A fixed header is contained in all MQTT control packets.
A variable header is contained in some MQTT control packets.
A payload is contained in some MQTT control packets.

- Fixed header

- Each MQTT control packet contains a fixed header as shown below.

Bit	7	6	5	4	3	2	1	0
Byte 1	MQTT control packet type				Flags specific to each MQTT control packet type			
Byte 2	Remaining length							

- The remaining length is a variable byte integer that indicates the number of bytes remaining within the current control packet, including data in the variable header and payload. The remaining length does not include the bytes used to encode the remaining length. The packet size is the total number of bytes in an MQTT control packet, which is equal to the length of the fixed header plus the remaining length.

MQTT Fixed Header: Packet Type

- Packet type data is at the first byte (fixed header) and occupies bits 7–4. It is represented as a 4-bit unsigned value to indicate the control packet type.

Name	Value	Flow Direction	Description
Reserved	0	Forbidden	Reserved
CONNECT	1	Client to server	Connect request
CONNACK	2	Server to client	Connect acknowledgement
PUBLISH	3	Both directions	Publish message
PUBACK	4	Both directions	Publish acknowledgment (QoS 1)
PUBREC	5	Both directions	Publish received (QoS 2 delivery step 1)
PUBREL	6	Both directions	Publish release (QoS 2 delivery step 2)
PUBCOMP	7	Both directions	Publish complete (QoS 2 delivery step 3)
SUBSCRIBE	8	Client to server	Subscribe request
SUBACK	9	Server to client	Subscribe acknowledgement
UNSUBSCRIBE	10	Client to server	Unsubscribe request
UNSUBACK	11	Server to client	Unsubscribe acknowledgement
PINGREQ	12	Client to server	PING request
PINGRESP	13	Server to client	PING response
DISCONNECT	14	Both directions	Disconnect notification
AUTH	15	Both directions	Authentication exchange

MQTT Fixed Header: Flag

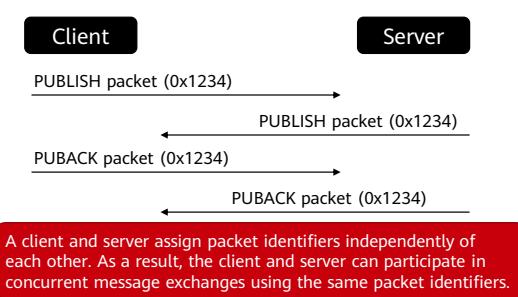
- The remaining four bits (bits 3–0) of the first byte contain the flag of each MQTT control packet type. If the flag is marked as "reserved", it is reserved for future use and must be set to the listed values. If a packet flag is not a listed value, the packet is marked as a malformed packet.

Name	Fixed Header Flag	Bit 3	Bit 2	Bit 1	Bit 0
CONNECT	Reserved	0	0	0	0
CONNACK	Reserved	0	0	0	0
PUBLISH	Used in MQTT v5.0	DUP	QoS		RETAIN
PUBACK	Reserved	0	0	0	0
PUBREC	Reserved	0	0	0	0
PUBREL	Reserved	0	0	1	0
PUBCOMP	Reserved	0	0	0	0
SUBSCRIBE	Reserved	0	0	1	0
SUBACK	Reserved	0	0	0	0
UNSUBSCRIBE	Reserved	0	0	1	0
UNSUBACK	Reserved	0	0	0	0
PINGREQ	Reserved	0	0	0	0
PINGRESP	Reserved	0	0	0	0
DISCONNECT	Reserved	0	0	0	0
AUTH	Reserved	0	0	0	0

- DUP: duplicate delivery of a PUBLISH packet
- QoS: PUBLISH quality of service
- RETAIN: PUBLISH retained message flag

MQTT Variable Header: Packet Identifier

- Some types of packets contain a variable header between the fixed header and payload. The content of a variable header depends on the packet type. The **packet identifier** field is a common part in the variable header.
- A packet identifier is a 2-byte integer. It is used to identify a packet during packet exchange between a client and server.



Packet	Packet Identifier
PUBLISH	Contained (QoS > 0)
PUBACK	Contained
PUBREC	Contained
PUBREL	Contained
PUBCOMP	Contained
SUBSCRIBE	Contained
SUBACK	Contained
UNSUBSCRIBE	Contained
UNSUBACK	Contained

- If the QoS value of a PUBLISH packet is set to 0, the packet must not contain a packet identifier.
- Each time a client sends a new SUBSCRIBE, UNSUBSCRIBE, or PUBLISH packet (QoS > 0), it must assign it a non-zero packet identifier that is currently unused.
- Each time a server sends a new PUBLISH packet (QoS > 0), it must assign it a non-zero packet identifier that is currently unused.
- After a sender processes the corresponding acknowledgment packet, the packet identifier can be reused.
- Packet identifiers used with PUBLISH, SUBSCRIBE, and UNSUBSCRIBE packets form a single, unified set of identifiers for the client and server in a session. A packet identifier cannot be used by multiple commands at any time.
- A PUBACK, PUBREC, PUBREL, or PUBCOMP packet must contain the same packet identifier as the originally sent PUBLISH packet. A SUBACK and UNSUBACK packets must contain the packet identifier used in the corresponding SUBSCRIBE and UNSUSCRIBE packets, respectively.

MQTT Variable Header: Properties

- The last field in the MQTT variable header is a set of properties, which can be contained in all types of packets except **PINGREQ** and **PINGRESP**. A CONNECT packet with a payload may contain a set of optional properties in the will property field.
- A property consists of an Identifier which defines its usage and data type, followed by a value. If a control packet contains an identifier that does not match the packet type, the packet is marked as a malformed packet.

Some property identifiers and their usage

Identifier		Name (Usage)	Type	Packet Type
DEC	HEX			
1	0x01	Payload format indicator	1 byte	PUBLISH
3	0x03	Message expiry interval	4 bytes	PUBLISH
8	0x08	Response topic	UTF-8 encoded string	PUBLISH
11	0x0B	Subscription identifier	Variable byte	PUBLISH, SUBSCRIBE
34	0x22	Maximum topic alias	2 bytes	CONNECT, CONNACK
36	0x24	Maximum QoS	1 byte	CONNACK
39	0x27	Maximum packet size	4 bytes	CONNECT, CONNACK

- The property length is encoded as a variable byte integer. The property length does not include the bytes used to encode itself, but includes the length of the properties. If there are no properties, this must be specified by including a property length of zero.
- Although the property identifier is defined as a variable byte integer, all of the property identifiers are one byte long in MQTT v5.0 specifications.

MQTT Variable Header: PUBLISH Packet Example

Non-standard example of the variable header in a PUBLISH packet

	Description	7	6	5	4	3	2	1	0
Topic name									
Byte 1	Length MSB (0)	0	0	0	0	0	0	0	0
Byte 2	Length LSB (3)	0	0	0	0	0	0	1	1
Byte 3	'a' (0x61)	0	1	1	0	0	0	0	1
Byte 4	'/' (0x2F)	0	0	1	0	1	1	1	1
Byte 5	'b' (0x62)	0	1	1	0	0	0	1	0
Packet identifier									
Byte 6	Packet identifier MSB (0)	0	0	0	0	0	0	0	0
Byte 7	Packet identifier LSB (10)	0	0	0	0	1	0	1	0
Property length									
Byte 8	No property	0	0	0	0	0	0	0	0

MQTT Payload

- Some control packets contain a payload as the final part of the packet. For a PUBLISH packet, its payload is the **application message**.
- Packets that contain a payload include: CONNECT, PUBLISH (optional), SUBSCRIBE, SUBACK, UNSUBSCRIBE, and UNSUBACK.

Contents

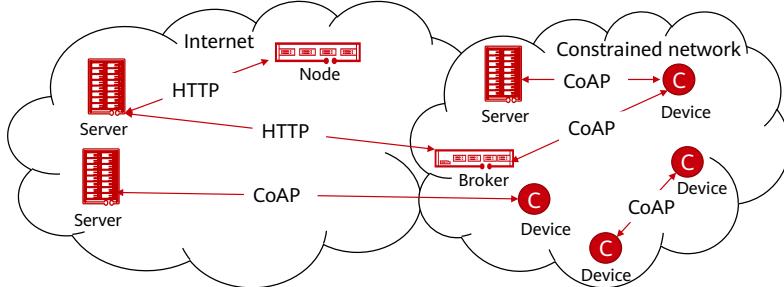
1. Basics of Network Communications

2. Common IoT Protocols

- HTTP
- AMQP
- MQTT
- CoAP
- Protocol Comparison

CoAP Overview

- Constrained Application Protocol (CoAP) is a specialized Internet application protocol for constrained devices (also called nodes) to communicate with the wider Internet.
- CoAP is designed to easily translate to HTTP for integration with the Web while also meeting IoT device communications requirements, such as multicast support, low overhead, and simplicity. CoAP can run on most devices that support **User Datagram Protocol (UDP)**.



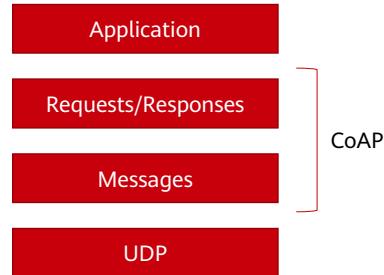
40 Huawei Confidential



- CoAP is designed for use between devices on the same constrained network (for example, low power consumption, lossy network), devices on the Internet, general nodes, and devices connected over the Internet on different constrained networks.
- Features:
 - It is based on REST. Server resource addresses are similar to those of the Internet in URL format. A client also uses the POST, GET, PUT, and DELETE methods to access the server. It simplifies HTTP.
 - CoAP is in binary format, and HTTP is in text format. CoAP is more compact than HTTP.
 - It is lightweight. The minimum length of a CoAP message is only 4 bytes, and an HTTP header contains dozens of bytes.
 - CoAP supports reliable transmission, data retransmission, and block transmission. It ensures reliable data arrival.
 - It supports IP multicast, that is, requests can be sent to multiple devices at the same time.
 - CoAP supports non-persistent connection communications and is applicable to low-power-consumption IoT scenarios.
- CoAP is developed from HTTP, so it uses the request/response model as HTTP does. A client initiates a request, and a server responds to the request.
- CoAP has a compact message format and runs over UDP by default. It can also run over DTLS or other transmission protocols, such as SMS, TCP, and SCTP.

CoAP Logical Layering

- Logically, CoAP uses a two-layer architecture. The **messaging layer** handles UDP and asynchronous interactions. The **request/response layer** interacts with applications using Representational State Transfer (REST) request methods (similar to HTTP) and response codes.



CoAP Messaging Layer: Message Types

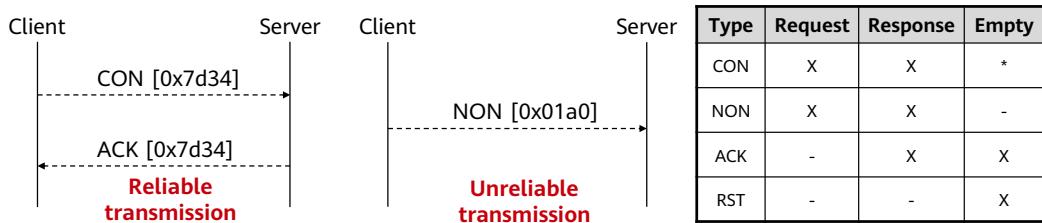
- CoAP defines four types of messages: Confirmable (CON), Non-confirmable (NON), Acknowledgment (ACK), and Reset (RST). The method and response code in a message decide whether the message carries a request or response.
- A message may carry a **request** or **response**, or be **empty**. An empty message has the **Code** field set to **0.00**. The **Token Length** field must be set to **0** and bytes of data must not be present after the **Message ID** field. If there are any bytes, they must be processed as a message format error.

Field Type	Description
CON	CON requests require an acknowledgement.
NON	NON requests do not require an acknowledgement. They are suitable for scenarios where messages are repeatedly sent and packet loss does not affect normal operations.
ACK	Response to CON.
RST	During reliable transmission, if an unrecognizable message is received or if the message has errors or an invalid format, an ACK message cannot be returned. An RST message is returned instead.

- A request can be carried in a CON or NON message, and a response can be carried in a CON, NON, or ACK message.

CoAP Messaging Layer: Transmission Reliability

- CoAP uses CON messages to ensure reliability. A CON message is retransmitted using a default timeout and exponential backoff between retransmissions, until the recipient sends an ACK message with the same message ID. When a receiver cannot process a CON message (cannot provide an appropriate error response), the receiver replies with an RST message instead of an ACK message.
- Messages that do not require reliable transmission are sent as NON messages. These messages are not acknowledged, but still have message IDs for duplicate detection. When a receiver cannot process a NON message, the receiver can reply with an RST message.



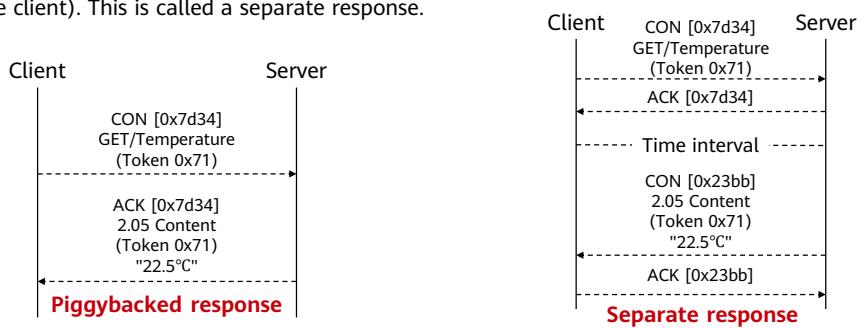
- X indicates that the combination exists. - indicates that the combination does not exist. * indicates that the combination is not used in normal operations but only to elicit an RST message.
- A CON message always carries either a request or response unless it is only used to elicit an RST message, in which case it can be empty. A recipient must either use an ACK message to acknowledge a CON message or reject the message if the recipient lacks context to process the message properly because the message is empty, uses a code with a reserved class, or has a message format error. To reject a CON message, send a matching RST message and ignore it.
- An ACK message must echo the message ID of the CON message and must carry a response or be empty.
- An RST message must echo the message ID of the CON message and must be empty.
- To reject an ACK or RST message, silently ignore it.
- More generally, recipients of ACK and RST messages must not respond with either ACK or RST messages. A sender retransmits the CON message at exponentially increasing intervals, until the sender receives an ACK or RST message or runs out of attempts.

CoAP Request/Response Layer: Response Classification

- CoAP request and response semantics are carried in CoAP messages, which include either a Method Code or Response Code, respectively. Optional details (such as the URI and payload media type) are carried as CoAP options. A Token is used to match a response with a request independent from the underlying messages.
- For CoAP, different types of requests get different types of responses based on how the response is sent:
 - A **piggybacked response** to a CON request
 - A **separate response** to a CON request
 - A response to a NON request

CoAP Request/Response Layer: Responses to CON Messages

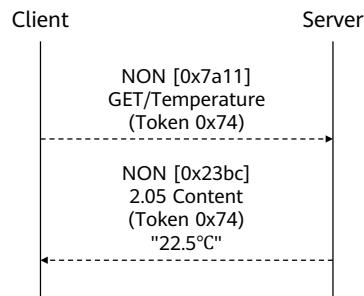
- If a request is carried in a CON message and what is requested is **immediately available**, the response to the CON message is carried in the ACK message. This is called a piggybacked response.
- If the server **cannot respond immediately** to a request carried in a CON message, it simply responds with an **empty ACK message** so that the client can stop retransmitting the request. When the response is ready, the server sends it in a new CON message (which then needs to be acknowledged by the client). This is called a separate response.



- The exact rules for matching a response with a request are as follows:
 - The source endpoint of the response must be the same as the destination endpoint of the original request.
 - In a piggybacked response, the message ID of the CON request and the ACK must match, and the tokens of the response and original request must match. In a separate response, the tokens of the response and original request must match.

CoAP Request/Response Layer: Responses to NON Messages

- If a request is carried in a NON message, a response will be returned in a new NON message. However, the server may respond with a CON message.



CoAP Message Format

- CoAP messages are encoded in a simple binary format starting with a fixed 4-byte header. It is followed by a token in 0-8 bytes. Following the token comes a sequence of zero or more CoAP options in type-length-value (TLV) format, Options may be followed by a payload that takes up the rest of the datagram.

Bit	7	6	5	4	3	2	1	0		
Byte 1	Ver		T		TKL					
Byte 2	Code									
Byte 3	Message ID									
Byte 4	Token (if any)									
Byte 5	Options (if any)									
...	1	1	1	1	1	1	1	1		
...	Payload (if any)									

- **Ver:** CoAP version number, similar to HTTP1.0 and HTTP1.1. The version number occupies two bits.
- **T:** packet type. CoAP defines four types of packets: CON, NON, ACK, and RST.
- **Token Length (TKL):** length of the **Token** field, in 4 bits.
- **Code:** request code and response code. It has different forms in CoAP request and response messages.
- **Message ID:** used to detect message duplication and to match message types. Each CoAP message has an ID. The ID remains unchanged in a session, but is reused after the session ends.
- **Token:** token value. Its length is specified by the **TKL** field. The token value is used to correlate a request with the corresponding response.
- **Options:** packet options, in which CoAP host, CoAP URI, CoAP request parameters, and payload media type can be set.
- **11111111:** separator between the CoAP header and payload. The value is fixed at one byte.
- **Payload:** transmitted data.

CoAP Message Fields: Code (1)

- **Code** is in X.XX format. It occupies one byte and is split into a 3-bit class (bits before the period) and 5-bit details (bits after the period).
- Classes include:
 - Request (0.XX)
 - Success response (2.XX)
 - Client error response (4.XX)
 - Server error response (5.XX)
 - Other classes are reserved.

Code description							
7	6	5	4	3	2	1	0
Class				Details			

CoAP Message Fields: Code (2)

Code	Name	Description	Code	Name	Description
0.00	Empty	Empty message.	4.04	Not Found	Resource not found, similar to HTTP 404.
0.01	GET	Used to obtain a resource.	4.05	Method Not Allowed	Invalid request method, similar to HTTP 405.
0.02	POST	Used to create a resource.	4.06	Not Acceptable	The requested option is inconsistent with the server-generated option, similar to HTTP 406.
0.03	PUT	Used to update a resource.	4.12	Precondition Failed	Insufficient request parameters, similar to HTTP 412.
0.04	DELETE	Used to delete a resource.	4.13	Request Entity Too Large	The request entity is too large, similar to HTTP 413.
2.01	Created	Response to POST and PUT requests only, similar to HTTP 201.	4.15	Unstoppable Content-Format	The media type in the request is not supported, similar to HTTP 415.
2.02	Deleted	Response to DELETE and some POST requests only, similar to HTTP 204.	5.00	Internal Server Error	Internal server error, similar to HTTP 500.
2.03	Valid	ETag in the request is valid, similar to HTTP 304.	5.01	Not Implemented	The server does not support the functionality required to fulfill the request, similar to HTTP 501.
2.04	Changed	Response to POST and PUT requests only, similar to HTTP 204.	5.02	Bad Gateway	The server receives an error response when acting as a gateway, similar to HTTP 502.
2.05	Content	Response to GET requests only, similar to HTTP 200.	5.03	Service Unavailable	The server is overloaded or shut down, similar to HTTP 503.
4.00	Bad Request	Request error, similar to HTTP 400.	5.04	Gateway Timeout	The server does not receive a response in time for processing the request when acting as a gateway, similar to HTTP 504.
4.01	Unauthorized	The client is not authorized to perform the requested action, similar to HTTP 401.	5.05	Proxying Not Supported	The server does not support the proxy function.
4.02	Bad Option	The request contains an invalid option.			
4.03	Forbidden	The server rejects the request, similar to HTTP 403.			

- A request with an unrecognized method code requires a 4.05 (Method Not Allowed) piggybacked response.
- ETag: entity tag. Generally, the ETag is not sent to the client in plain text. Etag values vary in a resource lifecycle to identify the resource status. When a resource changes, if one or more fields in the header change or the message entity changes, ETag changes accordingly.
- An ETag value change indicates that a resource status change. Generally, the ETag header information can be obtained based on the timestamp. The server calculates the ETag value and returns it to the client when requested by the client.

CoAP Message Fields: Options (1)

- CoAP defines a number of options that can be included in a message. Each option instance in a message specifies the option number, option value length, and option value itself.
- Instead of specifying the option number directly, the instances must appear in the order of their option numbers and **delta encoding** is used between them. The option number of each instance is calculated as the sum of its delta and the option number of the preceding instance in the message. For the first instance in a message, a preceding option instance with option number zero is assumed. The delta of multiple instances of the same option is zero.

Bit	7	6	5	4	3	2	1	0				
1 byte	Option delta				Option length							
0-2 bytes	Option delta (extended)											
0-2 bytes	Option length (extended)											
0 or more bytes	Option value											

- Option delta: delta value of option. The current option number is equal to the sum of all option deltas.
- Option length: A value between 0 and 12 indicates the length of the option value, in bytes. **13**: option length from 13 to 268. The extended part is a 1-byte value of the actual length minus 13. **14**: option length from 269 to 65804. The extended part is a 2-byte value of the actual length minus 269. **15**: Reserved for a payload.
- Option value: option content.

CoAP Message Fields: Options (2)

- Requests and responses may include one or more options, which are used to properties, similar to parameters or feature descriptions. Options may specify the destination host port and whether a proxy server is used.
- Options fall into one of two classes:
 - **Critical options** must be recognized by the receiver. Otherwise, the message cannot be processed properly.
 - **Elective options** can be ignored if they cannot be recognized by the receiver. This does not affect message processing.

No.	Name
1	If-Match
3	Uri-Host
4	ETag
5	If-None-Match
7	Uri-Port
8	Location-Path
11	Uri-Path
12	Content-Format
14	Max-Age
15	Uri-Query
17	Accept
20	Location-Query
35	Proxy-Uri
39	Proxy-scheme
60	Size1

CoAP Request/Response Message Examples

Request message								Response message							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
GET = 1								2.05 = 69							
MID = 0xee3d (2 bytes)								MID = 0xee3d (2 bytes)							
Token = 0x3473aab7 (4 bytes)								Token = 0x3473aab7 (4 bytes)							
Options = "temperature" (11 bytes)								Options = "temperature" (11 bytes)							
1								1							
"22.3°C" (6 bytes)								"22.3°C" (6 bytes)							

Contents

1. Basics of Network Communications

2. Common IoT Protocols

- HTTP
- AMQP
- MQTT
- CoAP
- **Protocol Comparison**

Protocol Comparison

Item	HTTP	AMQP	MQTT	CoAP
Work model	Request/Response	Publish/Subscribe	Publish/Subscribe	Request/Response
QoS	Guaranteed by TCP	Three levels	Three levels	CON or NON messages
Transport layer (generally)	TCP	TCP	TCP	UDP
Subscription control	N/A	Queue and message filtering	Layer-matching topic subscription	N/A
Encoding	Plain text	Binary	Binary	Binary
Security	SSL and TLS	SASL authentication and TLS data encryption	Username/Password authentication and SSL data encryption	TLS data encryption

Quiz

1. (True or false) In CoAP, an empty message is neither a request nor a response.
2. (Single-answer question) Which of the following is not a layer of the OSI reference model?
 - A. Physical layer
 - B. Network layer
 - C. Platform layer
 - D. Transport layer

- Answers:

- T
 - C

Quiz

3. (Multiple-answer question) Which of the following are QoS levels of MQTT?
- A. QoS 0
 - B. QoS 1
 - C. QoS 2
 - D. QoS 3

- Answer:
 - ABC

Summary

- Having completed this section, you should now understand the basics of network communications, such as the OSI reference model as well as network topologies and their advantages and disadvantages.
- In addition, you are expected to understand the architectures, features, and performance of HTTP, AMQP, MQTT, and CoAP.

Acronyms or Abbreviations

- TCP: Transmission Control Protocol
- UDP: User Datagram Protocol

Thank you.

把数字世界带入每个人、每个家庭、

每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.



AT Commands for IoT Communication Modules



Foreword

- This section describes Attention (AT) commands used for NB-IoT, Wi-Fi, and Huawei-certified modules.
- AT commands are sent from terminal equipment (TE) or data terminal equipment (DTE) to a terminal adapter (TA) or data circuit terminal equipment (DCE). They are used to control the functions of mobile terminals (MTs) to interact with network services.

Objectives

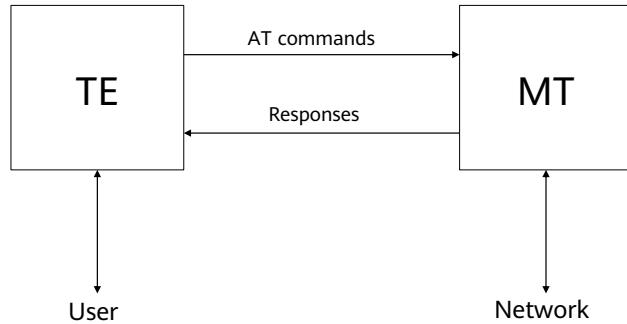
- Upon completion of this course, you will:
 - Know how to distinguish AT commands for NB-IoT, Wi-Fi, and Huawei certified communication modules.
 - Know how to connect NB-IoT, Wi-Fi, and Huawei-certified modules to the IoT platform.

Contents

- 1. AT Command Types**
2. NB-IoT AT Commands
3. Wi-Fi AT Commands
4. AT Commands for Huawei-Certified Modules

AT Commands

- AT commands are used to control the rules of interactions between a TE and an MT.



AT Command Types

- Four types of AT commands:
 - Test command, =?, is used to display valid parameter values set by other AT commands.
 - Read command, ?, is used to query attributes set by other AT commands.
 - Execution command, is used to instruct a module to execute a specific function.
 - Set command, =xx, is used to set the attributes carried in the AT command.

Category	Syntax	Example
Test command	AT+<x>=?	AT+CMEE=?
Read command	AT+<x>?	AT+CMEE?
Set command	AT+<x>=<...>	AT+CMEE=0
Execution command	AT+<x>	AT+NRB

Contents

1. AT Command Types
- 2. NB-IoT AT Commands**
3. Wi-Fi AT Commands
4. AT Commands for Huawei-Certified Modules

NB-IoT AT Commands

- An MCU uses AT commands to control a communication module. Terminal manufacturers must develop software that invokes AT commands to control communication modules in addition to developing corresponding service functions.

Purpose	AT Command
Disabling a function	AT+CFUN=0
Checking the software version	AT+CGMR
Querying the international mobile equipment identity (IMEI)	AT+CGSN=1
Setting the platform address	AT+NCDP=xx.xx.xx.xx,xx
Configuring an access point name (APN)	AT+CGDCONT=1,"IP","xxxx"
Rebooting a module	AT+NRB
Enabling a function	AT+CFUN=1
Querying the international mobile subscriber identity (IMSI) of a SIM card	AT+CIMI
Notifying the terminal of connecting to the base station	AT+CSCON=1

Purpose	AT Command
Notifying the terminal of connecting to the core network	AT+CEREG=2
Notifying the terminal of downlink data transmission	AT+NNMI=1
Notifying the terminal of successful data transmission	AT+NSMI=1
Attaching to a network	AT+CGATT=1
Querying the terminal status	AT+NUESTATS
Querying the IP address assigned by the core network	AT+CGPADDR
Transmitting data	AT+NMGS=1,11
Querying the sending buffer	AT+NQMGSS
Querying the receiving buffer	AT+NQMGR

Connecting an NB-IoT Device to the Huawei Cloud IoT Platform

- Power on the terminal, and run the **AT+NRB** command to reset the terminal. If **OK** is returned, the terminal is running properly.
- Run the **AT+NTSETID=1,*DEVICEID*** command to specify the device ID. The device ID is the terminal IMEI. If the command is executed, **OK** is returned.
- Run the **AT+NCDP=*IP*,*PORT*** command to set the IP address and port for connecting to the IoT platform. The port is 5683. If the command is executed, **OK** is returned.
- Run the **AT+CFUN=1** command to enable the network access function. If the command is executed, **OK** is returned.
- Run the **AT+NBAND=*Frequency_band*** command to specify the frequency band. If the command is executed, **OK** is returned.
- Run the **AT+CGDCONT=1,"IP","APN"** command to set the IoT core APN. If the command is executed, **OK** is returned.
- Run the **AT+CGATT=1** command to connect the terminal to the network. If the command is executed, **OK** is returned.
- Run the **AT+CGPADDR** command to check whether the terminal has obtained the IP address assigned by the IoT core network. If it has, the terminal has accessed the network.
- Run the **AT+NMGS=*DATASIZE*,*DATA*** command to enable the terminal to send upstream data. If the upstream data is sent, **OK** is returned.
- If the IoT platform sends downstream data to the terminal, obtain downstream data by running the **AT+NMGR** command.

Contents

1. AT Command Types
2. NB-IoT AT Commands
- 3. Wi-Fi AT Commands**
4. AT Commands for Huawei-Certified Modules

Universal Wi-Fi AT Commands

Command	Description
AT	Testing the AT function
AT+HELP	Viewing available AT commands
AT+MAC	Managing the MAC address
AT+IPERF	Testing performance
AT+SYSINFO	Viewing system information
AT+PING	Testing the IPv4 network connection
AT+PING6	Testing the IPv6 network connection
AT+DNS	Setting the DNS server address of the board
AT+ARP	Setting the ARP offload
AT+SLP	Setting the system low power function
AT+PS	Setting the Wi-Fi low power function

Instruction	Description
AT+DHCP	Running the DHCP client command
AT+DHCPSS	Running the DHCP server command
AT+IFCFG	Configuring an interface
AT+CC	Setting a country/region code
AT+DUMP	Reading the latest exception information
AT+NETSTAT	Checking the network status
AT+CSV	Querying the software version
AT+RST	Resetting a board
AT+WKGPIO	Setting the GPIO wakeup source in light sleep or deep sleep mode
AT+USLP	Entering the ultra-deep sleep mode
AT+SETUART	Configuring the serial port function

AP-related Wi-Fi AT Commands

- As a non-3GPP short-range wireless communications technology, Wi-Fi involves different AT commands from NB-IoT. Wi-Fi AT commands are mainly used to interact with the gateway and access the network through the gateway. Carrier data related to SIM cards, wireless networks, and core networks is not involved.
- An access point (AP) is the central node of a network. For example, a wireless router is an AP.

Purpose	AT Command
Restarting the module	AT+RST
Querying the version	AT+GMR
Scanning nearby APs	AT+CWLAP
Connecting to an AP	AT+CWJAP
Disconnecting from an AP	AT+CWQAP
Querying connection information	AT+CIPSTATUS
Resolving the domain name	AT+CIPDOMAIN

Purpose	AT Command
Establishing a connection	AT+CIPSTART
Starting transparent transmission	AT+CIPMODE
Transmitting data	AT+CIPSEND
Querying the local IP address	AT+CIFSR
Using the ping operation	AT+PING
Restoring to factory settings	AT+RESTORE
Querying the available memory space of the system	AT+SYSRAM

STA-related Wi-Fi AT Commands

- A station (STA) functions as a client in a wireless local area network (WLAN). It can be fixed or mobile. It can be a computer with a wireless network adapter or a smartphone with a Wi-Fi module.

Purpose	AT Command
Starting the STA	AT+STARTSTA
Stopping the STA	AT+STOPSTA
Configuring the reconnection policy	AT+RECONN
Initiating STA scanning	AT+SCAN
Scanning a specified channel	AT+SCANCHN
Scanning a specified SSID	AT+SCANSSID
Viewing scan results	AT+SCANRESULT

Purpose	AT Command
Initiating a connection to the AP	AT+CONN
Initiating a fast connection to the AP	AT+FCONN
Disconnecting from the AP	AT+DISCONN
viewing the STA status	AT+STASTAT
Using Wi-Fi protected setup (WPS) push-button configuration (PBC) for connection	AT+PBC
Using a WPS pin for connection	AT+PIN
Viewing the generated pin	AT+PINSHOW

Connecting a Wi-Fi Device to the Huawei Cloud IoT Platform

- Power on the terminal, and run the **AT+RST** command to reset the terminal. If **OK** is returned, the Wi-Fi mode has been configured on the terminal.
- Run the **AT+CWJAP="*SSID*","*Password*"** command to connect to the router. If the command is executed, **OK** is returned.
- Run the **AT+CIFSR** command to query the IP address of the local device. If the command is executed, **OK** is returned.
- Run the **AT+CIPSTART="TCP","*IP*","*PORT*"** command to set the IP address for connecting to the IoT platform. The port is 1883 (MQTT port). If the command is executed, **OK** is returned.
- Run the **AT+CIPSEND=<length>** command to send data. After > is returned, input the data. If the command is executed, **SEND OK** is returned.

Contents

1. AT Command Types
2. NB-IoT AT Commands
3. Wi-Fi AT Commands
- 4. AT Commands for Huawei-Certified Modules**

AT Commands for Huawei-Certified Modules

- For modules with Huawei compatibility certification, the AT commands and format specifications are similar to general specifications. Some manufacturers' modules may be implemented slightly differently, due to their AT channel limitations. These differences will be stated in the special description by module manufactures.

Purpose	AT Command
Obtaining the Huawei SDK version	AT+HMVER
Setting MQTT connection parameters	AT+HMCN
Disconnecting from the Huawei Cloud IoT platform	AT+HMDIS
Sending MQTT data to a topic	AT+HMPUB
Transmitting data received by the module to an external MCU	+HMREC
Transmitting the module connection or disconnection status to an external MCU	+HMSTS
Subscribing to a custom topic	AT+HMSUB
Unsubscribing from a custom topic	AT+HMUNS
Setting a server or client certificate	AT+HMPKS

Connecting a Huawei-certified Module to the Huawei Cloud IoT Platform

- Run **AT+HMCON=bsmode,lifetime,"serverip","serverport","deviceID","passwd",codec** to connect to the Huawei Cloud IoT platform. If **+HMCON OK** is returned, the connection is successful.
- Run **AT+HMSUB=qos,topic** to subscribe to a custom topic. If **+HMSUB OK** is returned, the subscription is successful.
- Run **AT+HMPUB=qos,topic,payload_len,payload** to report messages or properties. If **+HMPUB OK** is returned, the reporting is successful.
- Deliver a command on the Huawei Cloud IoT platform. If the device receives **+HMREC:topic,payload_len,payload**, the command is received.
- Run **AT+HMUNS="topic"** to unsubscribe from a custom topic. If **+HMUNS OK** is returned, the unsubscription is successful.
- Run **AT+HMDIS** to disconnect the device from the IoT platform.
- Run **AT+HMPKS=type,para1,[para2],"Certificate"** to set the server or client certificate.

Quiz

1. (True or false) AT commands are used to control the rules of interactions between a TE and an MT.
2. (Single-answer question) Which of the following command types is **AT+CMEE?** classified into?
A. Test command B. Read command C. Set command D. Execution command
3. (Multiple-answer question) Which of the following are types of AT commands?
A. Test command
B. Read command
C. Set command
D. Execution command

- Answers:

- T
 - B
 - ABCD

Summary

- This section describes the definition and classification of AT commands, AT commands for NB-IoT, Wi-Fi, and Huawei-certified modules, and how to connect them to the IoT platform.

Recommendations

- Huawei Cloud IoTDA product documentation:
 - <https://support.huaweicloud.com/intl/en-us/iothub/index.html>

Acronyms and Abbreviations

- MCU: Microcontroller Unit
- NB-IoT: Narrowband Internet of Things

Thank you.

把数字世界带入每个人、每个家庭、

每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.



IoT Platform Overview



Foreword

- IoT is a recent rapid development, with some challenges. The IoT industry requires a reliable, secure platform for decoupling device access and hosting open capabilities.
- The industry-leading Huawei Cloud IoT platform provides complete northbound and southbound APIs. Pre-integrated communication protocol plug-ins help customers quickly launch services.
- Huawei Cloud IoT features full-stack, all-scenario services: simplified access, intelligence, security, and trustworthiness. One-stop tools cover development, integration, hosting, and operations for partners and customers to customize IoT solutions for 5G and AI.

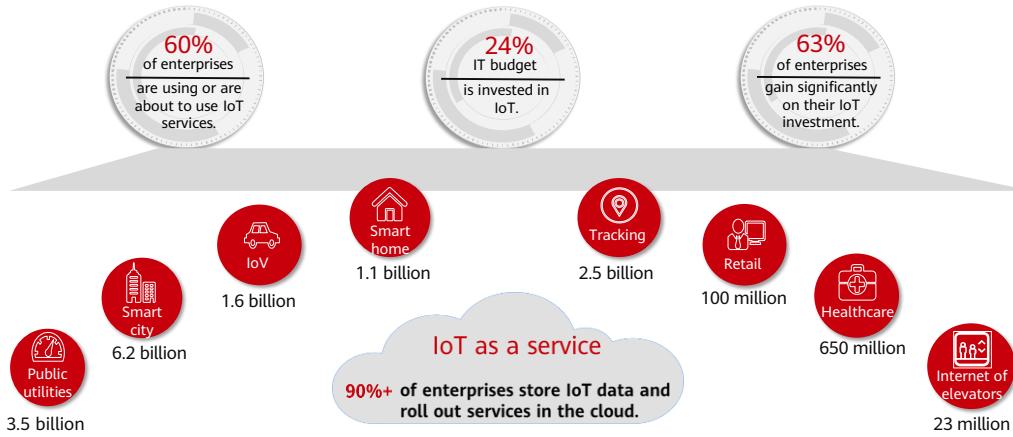
Objectives

- Upon completion of this course, you will:
 - Understand the requirements for IoT platforms.
 - Understand Huawei Cloud IoT full-stack services.
 - Understand how to access the Huawei Cloud IoT platform.

Contents

- 1. Requirements for IoT Platforms**
2. IoT Platform Classification
3. Open Source IoT Platforms
4. Huawei Cloud IoT Full-Stack Services
5. Accessing the Huawei Cloud IoT Platform

IoT Ushers In Industry Innovation and Transformation

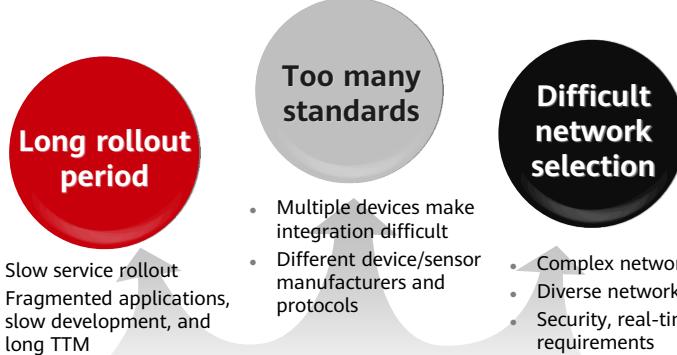


5 Huawei Confidential



- Survey data of 1,096 companies from 11 verticals in 17 countries
- Number of IoT connections by 2025
- Source: Machina, Circle-research, Gartner, IDC, and Huawei Research

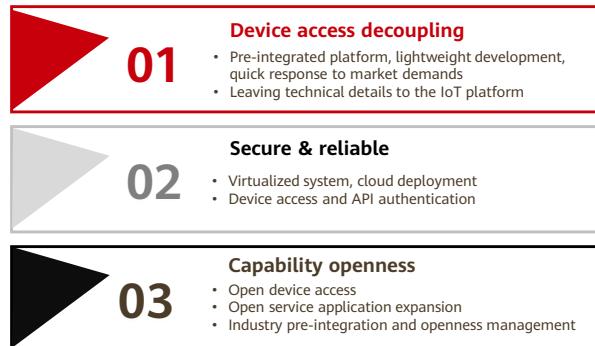
Major Challenges Facing the IoT Industry



How do we address these challenges when developing the IoT industry?

Requirements for IoT Platforms

- Secure and reliable, supporting device access decoupling and capability openness.

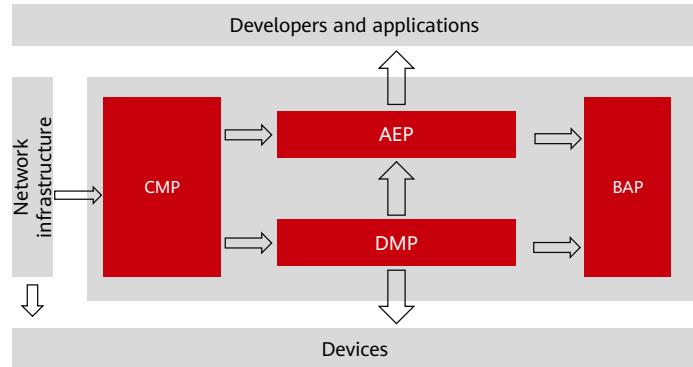


Contents

1. Requirements for IoT Platforms
- 2. IoT Platform Classification**
3. Open Source IoT Platforms
4. Huawei Cloud IoT Full-Stack Services
5. Accessing the Huawei Cloud IoT Platform

IoT Platform Classification

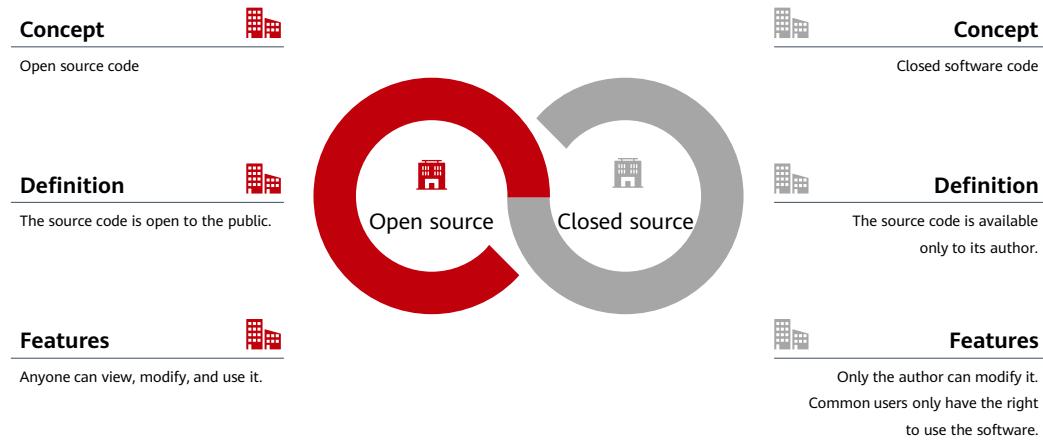
- Four types: connectivity management platform (CMP), device management platform (DMP), application enablement platform (AEP), business analytics platform (BAP)



Contents

1. Requirements for IoT Platforms
2. IoT Platform Classification
- 3. Open Source IoT Platforms**
4. Huawei Cloud IoT Full-Stack Services
5. Accessing the Huawei Cloud IoT Platform

Open vs. Closed Source



11 Huawei Confidential



- It is necessary to consider the features of open source or closed source software when you select between them.
 - Price: Open source software does not have a license or generate fees. The cost of closed source software depends on its size.
 - Customization: Open source software can be customized, but it depends on the open source license. For closed source software, you need to apply to the software company for modifying its source code.
 - Security: Code of open source software is reviewed by its community. This enables bugs to be quickly found and fixed. For closed source software, software distributors are responsible for fixing bugs.

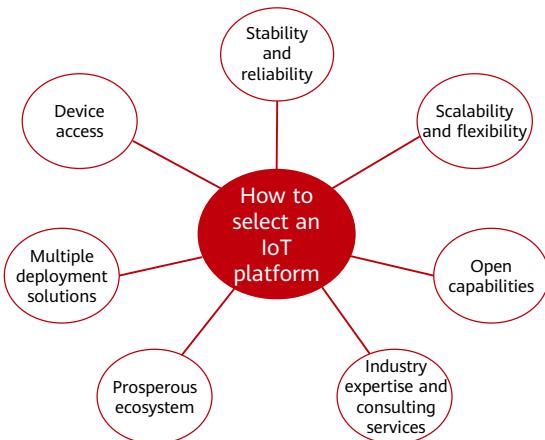
Open Source IoT Platforms

- In wider use than closed source IoT platforms due to: free of charge, low development cost, and open source.
- Popular open source IoT platforms include EMQX, ThingsBoard, and Kaa. Different IoT platforms have differentiated functions to attract users with different requirements.

Comparison of Open Source IoT Platforms

Platform	Product Positioning	Authorization	Pricing	MQTT Broker	Rule Engine	Secondary Development	Language
EMQX	EMQX is an open source IoT MQTT message broker built on the Erlang/OTP platform.	Apache-2.0	Free	✓	-	-	Erlang
ThingsBoard	ThingsBoard quickly develops, manages, and extends IoT projects. It provides off-the-shelf IoT cloud or internal solutions and server-side infrastructure for IoT applications.	Apache-2.0	Free	✓	✓	✓	Java
Kaa	Kaa is a highly flexible and scalable platform for building IoT solutions and managing connected devices.	Apache	Free	✓	✓	✓	Java

How to Select an IoT Platform



14 Huawei Confidential



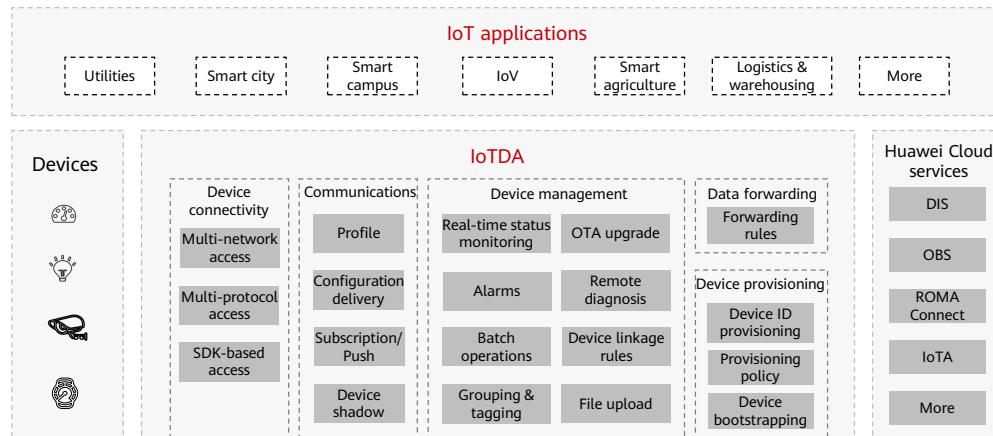
- Stability and reliability
 - IoT solutions are usually oriented to manufacturing, government, and other commercial scenarios. The stability of an IoT platform directly affects the running of thousands to tens of thousands of devices managed by the IoT platform. An unstable IoT platform consumes a large number of resources for system debugging and O&M, resulting in high costs.
- Scalability and flexibility
 - An IoT project evolves from the proof of concept (PoC) to a large-scale solution. An IoT platform needs to support flexible, fast project deployment and implementation at a low cost in the PoC phase. As the project scale increases, the IoT platform needs to expand functions and improve processing performance to support large-scale services.
- Device access
 - Device connectivity is one of the most fundamental functions of IoT platforms. IoT platforms that support multi-protocol, multi-type IoT devices can meet diverse access requirements. For projects of intelligent reconstruction of existing devices, if an IoT platform has strong access capabilities that support existing devices, the project deployment and R&D costs will be significantly reduced.

Contents

1. Requirements for IoT Platforms
2. IoT Platform Classification
3. Open Source IoT Platforms
- 4. Huawei Cloud IoT Full-Stack Services**
 - IoTDA
 - IoT A
 - IoT Stage
 - DRIS
 - IoT Edge
5. Accessing the Huawei Cloud IoT Platform

IoT Device Access (IoTDA)

- IoTDA is a service on the Huawei Cloud IoT platform.



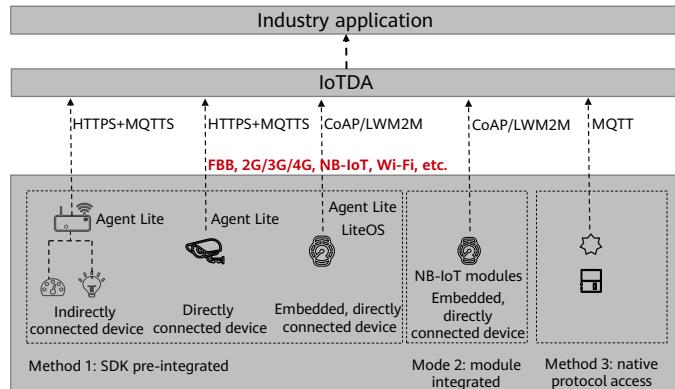
17 Huawei Confidential



- IoTDA provides functions such as device fleet access, bidirectional communications between devices and the cloud, batch device management, remote control and monitoring, OTA upgrade, and device linkage rules. It can also flexibly forward device data to other Huawei Cloud services. Using IoTDA, users can quickly connect devices to the platform and integrate applications.

Device Connectivity

- Devices connect to IoTDA directly or through gateways. They also connect to IoTDA over different protocols and Agents through multiple network types. Protocols are parsed in the cloud for fast device access.



- Multi-Agent access: Agents such as AgentLite and AgentTiny, and languages including C, Java, and Python are supported.
- Multi-protocol access: Access using the combination of HTTPS and MQTT, MQTT, LwM2M, and CoAP is supported.
- Multi-network access: Wired and wireless access modes, such as fixed broadband (FBB), 2G/3G/4G, and NB-IoT, are supported.
- Method 1: SDK pre-integrated
 - Indirectly connected device: The Huawei AgentLite SDK is integrated on a gateway, and devices that do not have the IP capability are connected to IoTDA through the gateway. This method applies to industrial IoT and smart campus scenarios.
 - Directly connected device: Computing and storage devices that have IP capabilities are directly integrated with the Huawei AgentLite SDK and can connect to IoTDA over HTTPS plus MQTT.
 - Embedded, directly connected device: Lightweight embedded devices such as sensors, meters, and controllers are integrated with the Huawei AgentTiny SDK (which can be used together with LiteOS) and can connect to IoTDA using CoAP over LwM2M. This method applies to scenarios that require low power consumption and low real-time performance, such as smart metering.
- Method 2: module integrated
 - Lightweight embedded devices such as sensors, meters, and controllers are integrated with Huawei certified communication modules and can connect to IoTDA using CoAP over LwM2M. This method applies to scenarios that require low power consumption and low real-time performance, such as smart metering.
- Method 3: native protocol access
 - Devices can connect to IoTDA over the native MQTT protocol. This

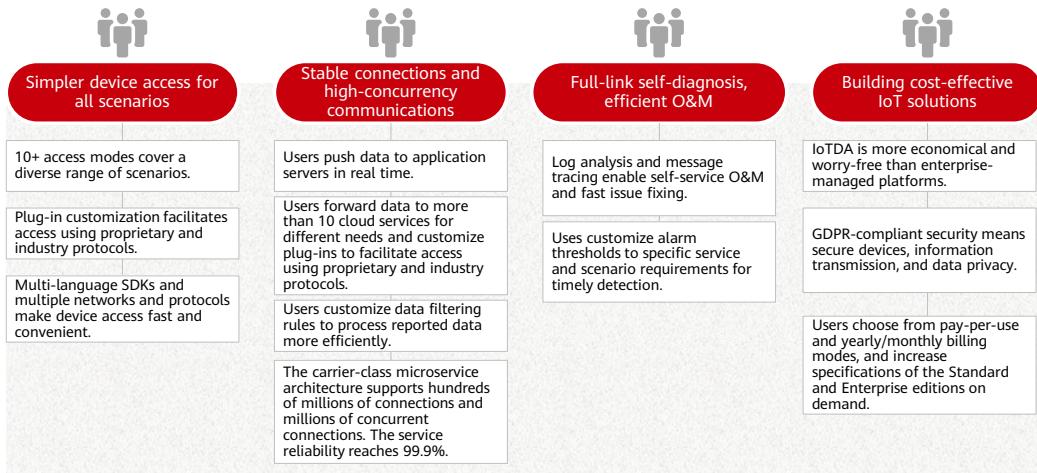
applies to persistent connection scenarios, such as smart street lamps.

Device Management

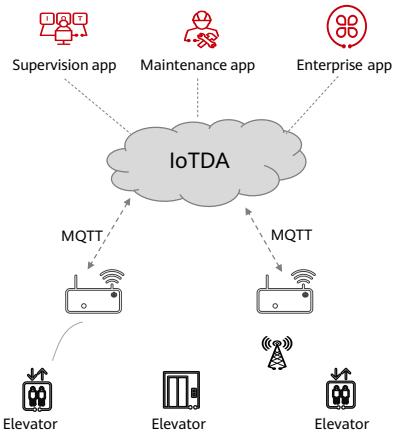
- IoTDA has extensive device management functions on the console or via calling APIs.

Function	Description	Function	Description
Product model definition	Includes device properties (color, size, collected data, identifiable commands, and reported events). Together, the manufacturer, device type, and device model uniquely identify the device type.	Rule engine	It allows users to set their own rules for devices connected to IoTDA. The devices trigger corresponding actions when conditions are met.
Device registration	Users create a device and configure device information on IoTDA.	Command delivery	Users deliver commands to remotely control devices through applications or the IoTDA console.
Access authentication	IoTDA authenticates connecting devices for data integrity and security, ensuring secure access.	Software and firmware upgrades	Users upgrade software and firmware of devices that support LwM2M and MQTT in over the air (OTA) mode.
Device access authorization	IoTDA allows an application to grant management rights of its bound devices to another application for easier management.	Batch operations	Users perform batch operations on devices (device registration, configuration update, command delivery, and software/firmware upgrade).
Device data collection	IoTDA collects device data (service data and alarms) that applications can then subscribe to.	Device log collection	Users remotely maintain devices by checking device logs collected by IoTDA.
Device shadow	This JSON file stores the reported and expected device status for applications.	Alarms	Uses manage devices by viewing alarm details and clearing alarms.
		Real-time status monitoring	IoTDA monitors device statuses in real time and notifies users of status changes.

Advantages of Huawei Cloud IoTDA



IoTDA Application - Smart Elevators



Challenges

- An elevator needs 3-5 sensor types, a significant development workload.
- More connected devices mean more pressure on the platform to maintain performance and scalability.

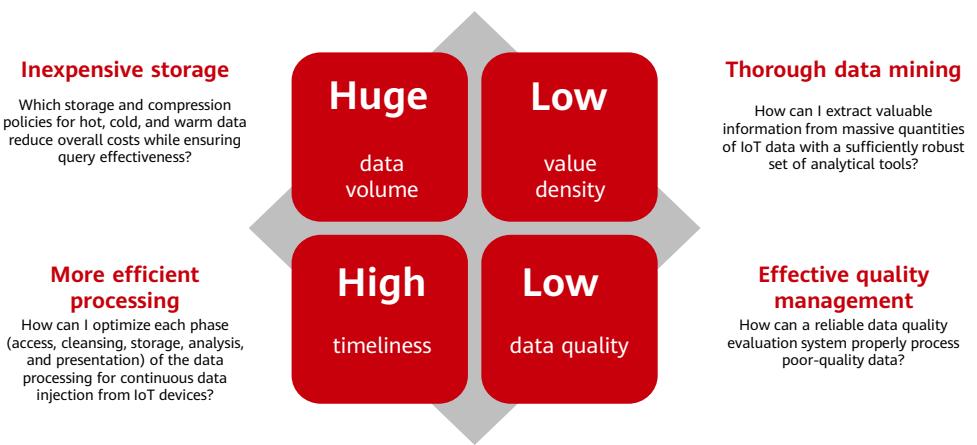
Benefits

- Multiple sensors are easily connected through an edge gateway that is pre-integrated with a device SDK.
- The platform supports hundreds of millions of connections and millions of concurrent connections, ensuring long-term connectivity for countless devices.
- Elevator data is queried easily and visually for timely O&M and unified supervision.
- Elevator control apps and applets for touch-free rides mitigate health risks in public spaces.

Contents

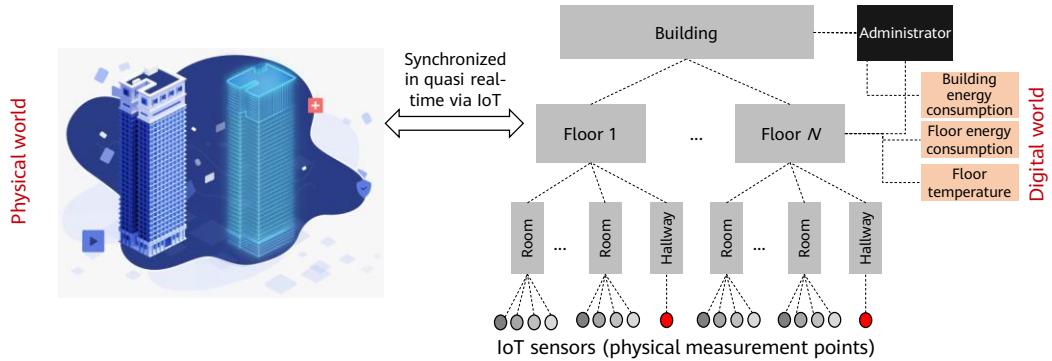
1. Requirements for IoT Platforms
2. IoT Platform Classification
3. Open Source IoT Platforms
- 4. Huawei Cloud IoT Full-Stack Services**
 - IoTDA
 - IoTA
 - IoT Stage
 - DRIS
 - IoT Edge
5. Accessing the Huawei Cloud IoT Platform

Why Is Data Analysis Required?



IoT Analytics: Asset Model

- Establish relationships (between things, between things and space, and between things and people) to understand data in context.
- Use IoT + asset models to build **digital twins** that are synchronized in quasi-real-time with physical things.
- Model-based abstraction is a unified, **service-oriented** foundation for data analysis.



IoT Analytics: Time Series Data Processing Is Key

Write performance

How do we ensure enough concurrency and real-time write for a large number of devices?

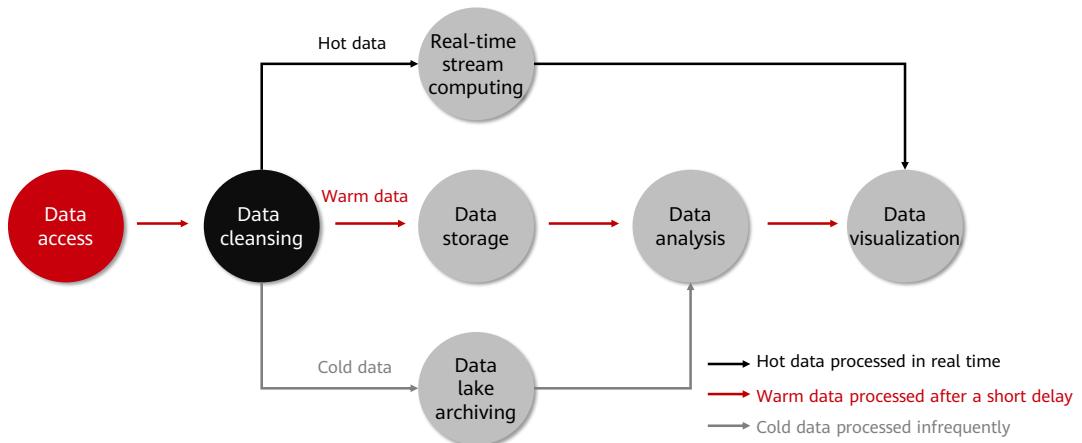
Compression ratio

Some IoT devices generate a lot of data. Does higher compression directly reduce costs?

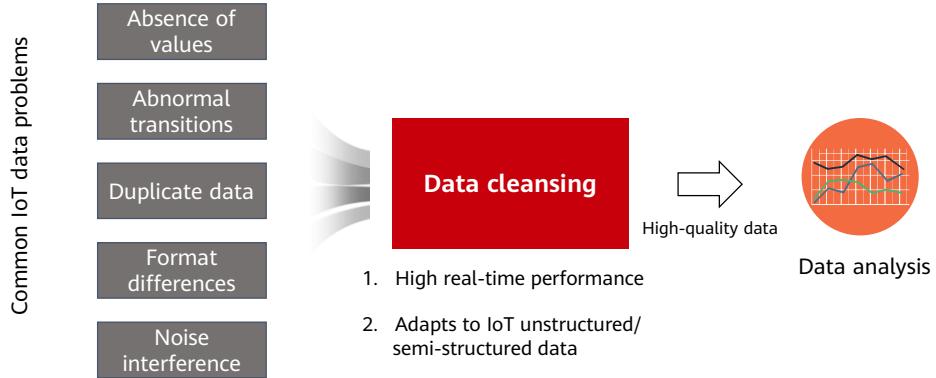
Query efficiency

How do we ensure powerful queries, especially time-based aggregation, for IoT data accumulated over a long period of time?

IoT Analytics: Multi-temperature Data Management Maximizes Processing Efficiency

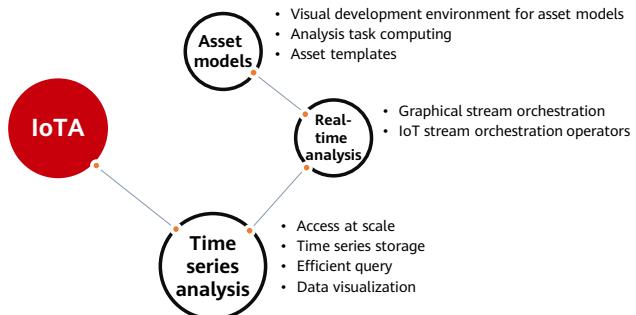


IoT Analytics: Efficient Data Cleansing Provides High-Quality Data for Analysis



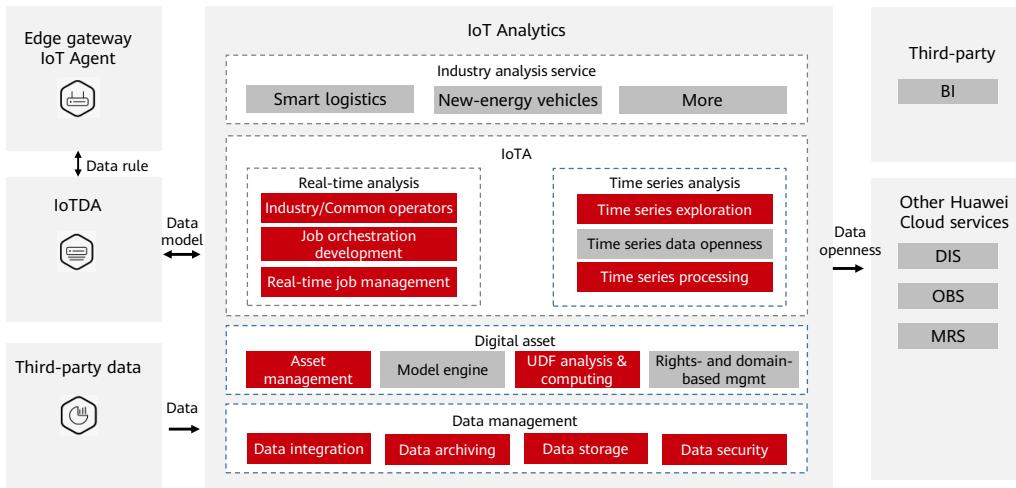
IoT Analytics (IoTA)

- Streamlines data integration, cleansing, storage, analysis, and visualization using asset models into one-stop services for IoT data developers. It makes development and monetization easier and faster.
- IoTA functions:



- Asset models: Physical objects or assets can be digitally projected to become computable and interactive. This greatly improves the connection between service systems and the physical world. IoTA helps developers quickly define complex asset models, perform real-time association computing on IoT data, and implement high-performance data access APIs based on these models.
- Real-time analysis: provided based on the big data stream computing engine. To make stream analysis job development easier, IoTA provides graphical stream orchestration so users can quickly develop and roll out services by dragging and dropping components.
- Time series analysis: IoTA provides enhanced time series data processing functions, for example, time series data storage with high compression ratio, efficient query, and a great number of timelines.

IoTA Architecture



IoTA Advantages

IoT asset models

- IoTA is an analysis service built on asset models. IoTA differs from general-purpose big data cloud services: easy references to IoT model data when defining data analysis jobs to improve job efficiency.

One-stop development

- IoTA uses best practices in big data analysis to provide one-stop data development services. Together with IoTDA, IoTA enables users to develop more efficiently.

Enhanced time series data processing

- IoTA enhances storage and analysis capabilities for time series data. Example: over 100,000 timelines per instance, data compression ratio up to 20:1, and aggregated computing of data in multiple time dimensions.

IoTA Application Scenarios



IoT device operations

- Challenges: Enterprises need multi-dimensional statistics and analysis to understand how devices are running. Such data includes quantity growth, activation, time segments when active, and alarms.
- Solution: IoTA seamlessly integrates with IoTDA. Users will receive data sets commonly used in device operations after connecting a device to Huawei Cloud through IoTDA and authorizing IoTA to access the device data. With IoTA, users no longer spend weeks developing data for IoT device operations. Several minutes are enough.



Enterprise OT data governance

- Challenges: Collected data is poor quality and needs to be cleansed. It costs a lot to store a large amount of OT data or complete data mining. OT data is time-sensitive and degrades quickly.
- Solution: IoTA data pipes and cleansing operators cleanse raw data in drag-and-drop mode. Users also model different types of physical assets and standardize data formats and interactive semantic interfaces. IoTA has a high-performance stream computing engine to process data in milliseconds.



Smart transportation

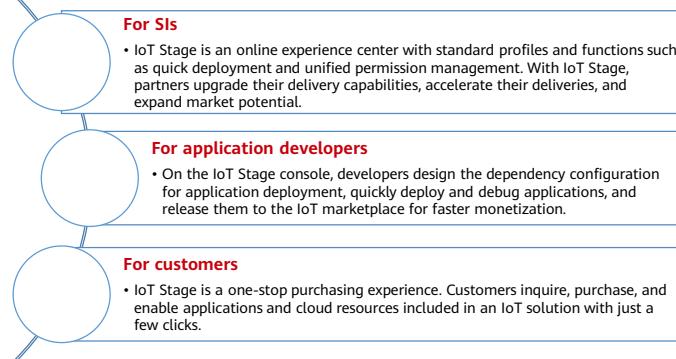
- Challenges: Often involves spatiotemporal analysis based on road models. It is challenging to build high-precision models for analysis.
- Solution: IoTA enables users to quickly build computable road models and develop road twins. With these models, users compute the spatiotemporal data from multiple dimensions.

Contents

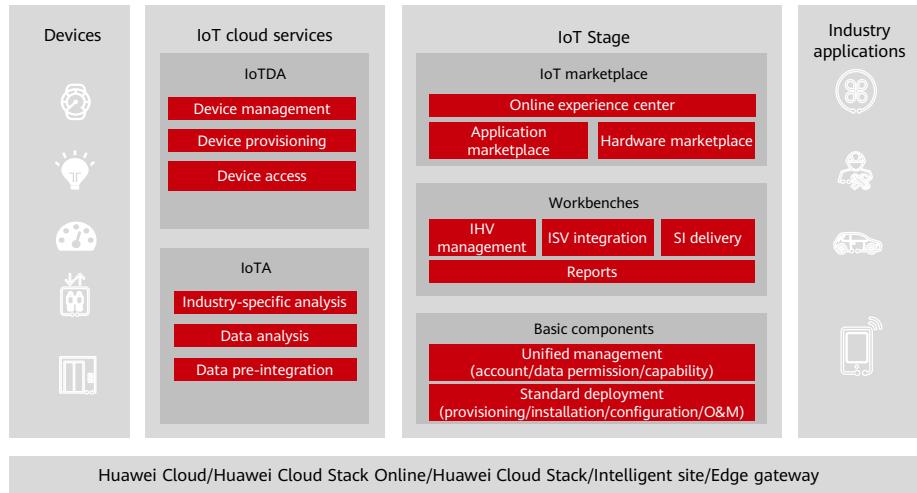
1. Requirements for IoT Platforms
2. IoT Platform Classification
3. Open Source IoT Platforms
- 4. Huawei Cloud IoT Full-Stack Services**
 - IoTDA
 - IoTA
 - IoT Stage**
 - DRIS
 - IoT Edge
5. Accessing the Huawei Cloud IoT Platform

IoT Stage

- This one-stop IoT delivery platform for channel distributors, system integrators, and users facilitates device and application integration, enables industry applications and services, and makes application replication cost-efficient.



IoT Stage Architecture

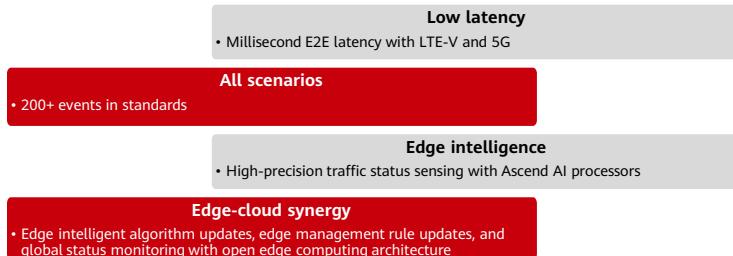


Contents

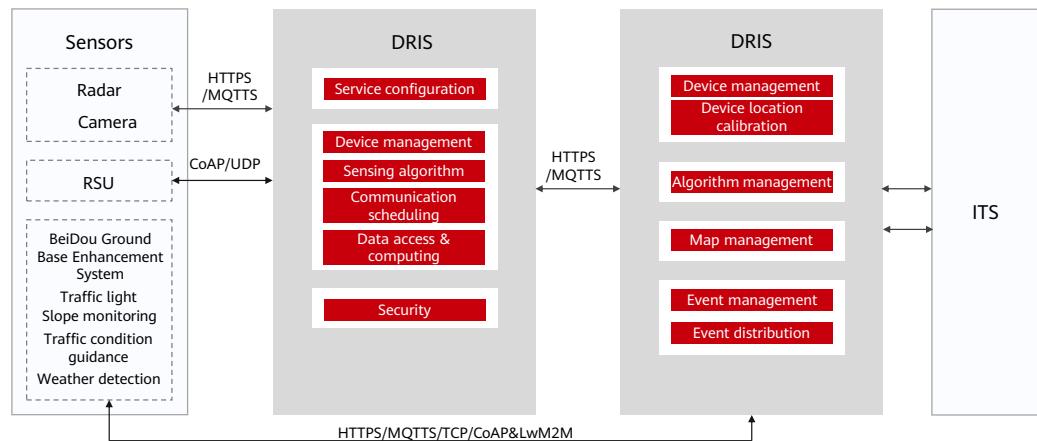
1. Requirements for IoT Platforms
2. IoT Platform Classification
3. Open Source IoT Platforms
- 4. Huawei Cloud IoT Full-Stack Services**
 - IoTDA
 - IoTA
 - IoT Stage
 - DRIS
 - IoT Edge
5. Accessing the Huawei Cloud IoT Platform

Digital Road Infrastructure Service (DRIS)

- V2X, IoT, and other smart tech enable full collaboration of pedestrians, vehicles, roads, and clouds. The service makes transportation and autonomous driving intelligent with safer, more efficient, and more convenient traveling, all-road sensing, all-weather road service, and full-process management and control for city managers.
- DRIS leverages Huawei IoT services, LTE-V/5G, high-precision positioning, and cloud computing technologies to provide information and facilitate operations.



DRIS Architecture



37 Huawei Confidential



- Intelligent Transportation System (ITS) includes IoV technologies such as vehicle-to-vehicle and vehicle-road communications. ITS uses dedicated short-range communications (DSRC) and Long Term Evolution-Vehicle (LTE-V) technologies.
- RSU: road side unit

DRIS V2X Server

- V2X Server is a vehicle-road collaboration platform for smart transportation scheduling.

Function	Description
Data collection	Real-time road events and vehicle/object/traffic light information are aggregated for global data management.
Data openness	Includes third-party platform data and service data (global sensing events, V2X information, and sensor data). Northbound API calling and subscription/push enable other platforms to query and use data.
Device model	Defines roadside device capabilities, with unified device models for devices at the edge.
Device access	Device provisioning, authentication, registration authentication, configuration, data subscription, command delivery, and data storage secure communications between authorized devices and transmitted information.
Precise scheduling	Rules coordinate edge services and long-distance precise push of global events.
Device authentication	Includes one-device-one-secret and two-way certificates, multiple transmission encryption protocols, and EU GDPR compliance.
Event convergence	Analysis of real-time road conditions and events reported by multiple single-point edge computing units generates global events for road sections.
Edge O&M	Edge application versions and configurations are managed together to remotely deploy, upgrade, and monitor applications.
Data storage	Stores system run logs, traffic event information from different channels, and structured data from vehicles.

- Vehicle-to-everything (V2X) Server is a basic application platform and brain of vehicle-road collaboration services. It analyzes multi-dimensional, global traffic information such as traffic conditions, road conditions, and emergencies of road sections, areas, and even the entire city to support smart applications for intelligent connectivity, public mobility, and supervision.

DRIS V2X Edge

- V2X Edge enables edge computing nodes to sense and analyze data for smart transportation.

Function	Description
Scenario-specific capabilities	Supports diverse range of vehicle-road collaboration events: warnings of collision, tunnels, incidents, abnormal vehicles, congestion, lane change in blind spots, emergency braking, wrong-way backing and overtaking, parking violations, rapid acceleration/deceleration, sudden turns, speeding, and trucks.
Edge O&M	Remotely deploys event algorithms for standardized management, event capability sensing, and remote O&M and upgrade.
Device model	Defines roadside device capabilities, with unified device models for devices at the edge.
Device access	Ingests data of devices from different vendors that meet industry standards. Such devices include roadside cameras and millimeter-wave radars.
Event analysis	Uses real-time analysis and required event generation to meet E2E industry latency requirements.
Algorithm deployment	Loads in the cloud and deploys on edge computing units, then upgraded and managed.
Communications & distribution	Broadcasts events locally through communication units and reports them to the V2X Server platform.
Sensing convergence	Analyzes data from roadside sensors such as cameras and millimeter-wave radars, and outputs structured data.
Topology management	Manages RSU network topology.

- V2X Edge is software deployed on intelligent edge computing unit devices on the road side. It extends V2X Server capabilities to the edge side. When cameras and radars connected to intelligent edge computing units detect traffic participants such as vehicles, pedestrians, and obstacles or other objects on the road, the original data of these objects is sent to V2X Edge for further analysis and event identification. Then, V2X Edge will generate traffic events that comply with national standards, send them to RSUs, and broadcast them to nearby vehicles. In addition, V2X Edge sends the events to V2X Server for storage and data openness. If a traffic accident is notified only to vehicles within the broadcast range (300–500 m) of an RSU, heavy trucks may not be able to keep a safe braking distance. Therefore, events need to be notified across measurement points. Roadside devices at an upstream road section need to notify vehicles 500 m to 1,000 m away from the accident point for vehicles to decelerate or change lanes in advance. Behind this, V2X Server pushes rules and algorithms from the cloud, and schedules them to the corresponding edge and RSU to notify drivers from a long distance.

DRIS Application Scenarios: Campus

- DRIS provides traffic light, speed limit, and abnormal vehicle warnings.

Traffic light warning

- When a vehicle approaches an intersection, the traffic light reports real-time light status to DRIS, which then sends the upcoming light status to the vehicle.

Speed limit warning

- DRIS sends an upcoming speed limit warning signal so oncoming vehicles can decelerate or bypass the road for accident prevention.

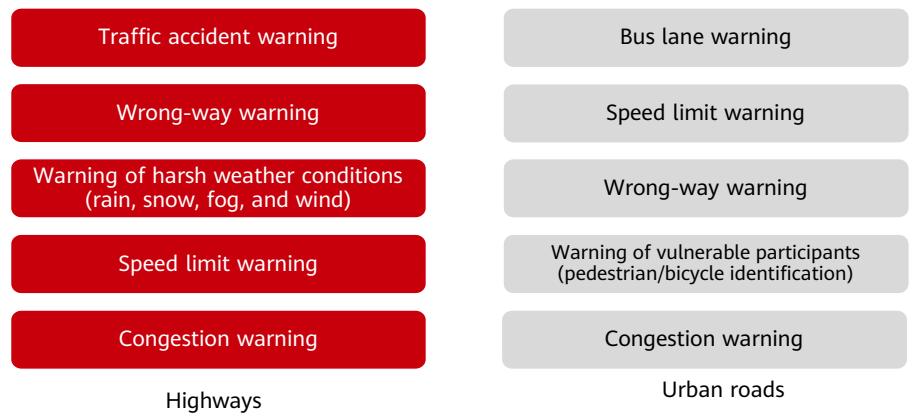
Wrong-way warning

- V2X Edge identifies the vehicle and sends an advance warning signal, so oncoming vehicles can decelerate or bypass the road for accident prevention.

Warning of vulnerable participants (pedestrian/bicycle identification)

- DRIS monitors real-time locations of pedestrians and bicycles at intersections using real-time videos and millimeter-wave radars, and broadcasts the locations to vehicles nearby. This eliminates blind spots for traffic participants and reduces traffic accidents.

DRIS Applications



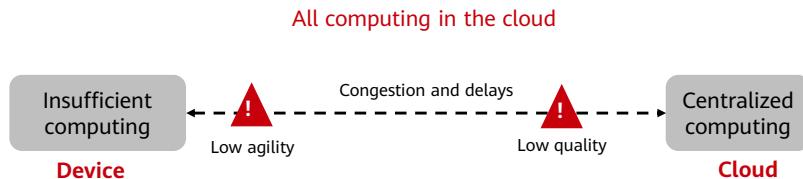
- Warning of vulnerable traffic participants (pedestrian/bicycle identification): DRIS monitors real-time locations of pedestrians and bicycles at intersections based on real-time videos and millimeter-wave radars and broadcasts the locations to vehicles nearby. This eliminates blind spots for traffic participants and reduces traffic accidents.
- Wrong-way driving warning: When an abnormal vehicle (for example, a wrong-way driving vehicle) appears on the road, DRIS V2X Edge identifies the vehicle and sends a warning signal in advance, so oncoming vehicles can decelerate or bypass the road in advance for accident prevention.
- Speed limit warning: DRIS sends an upcoming speed limit warning signal so oncoming vehicles can decelerate or bypass the road for accident prevention.
- Warning of entering bus lanes: When there is a bus lane on the road, DRIS sends bus lane information in advance. A non-bus vehicle that enters the bus lane will be notified to change a lane to prevent violating rules.
- Congestion warning: When a road is congested, DRIS V2X Edge identifies the road and sends a warning signal in advance, so oncoming vehicles can decelerate or replan the route in advance based on the vehicle-mounted sensor information, thereby improving traffic efficiency.
- Warning of harsh weather conditions such as rain, snow, fog, and wind: DRIS broadcasts harsh weather warnings to traffic participants who will be influenced by the weather in real time based on event delivery or weather analysis, thereby reducing the possibility of traffic accidents.

Contents

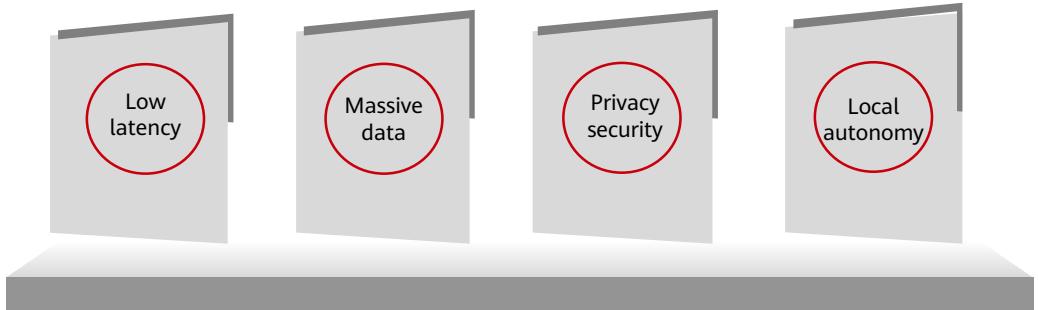
1. Requirements for IoT Platforms
2. IoT Platform Classification
3. Open Source IoT Platforms
- 4. Huawei Cloud IoT Full-Stack Services**
 - IoTDA
 - IoTA
 - IoT Stage
 - DRIS
 - **IoT Edge**
5. Accessing the Huawei Cloud IoT Platform

How is Edge Computing Generated?

- The Internet of Everything (IoE) and 5G mean massive growth in network edge devices and data. IoT devices have insufficient computing capabilities, so their data needs to be aggregated for centralized processing. This challenges service timeliness, network stability, and data security. Solution: better edge-cloud collaborative computing and management of edge device applications.



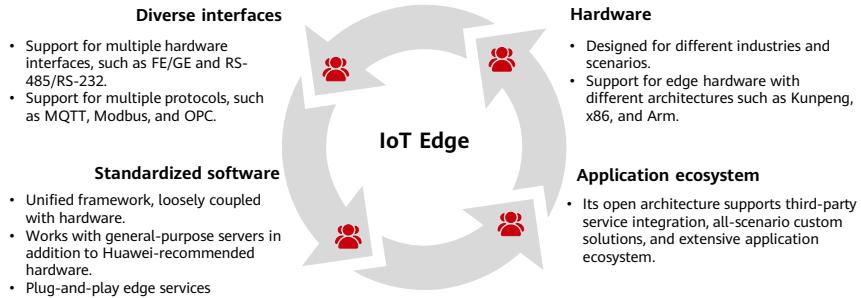
Four Factors for Faster IoT Edge Development



- Low latency: Edge solutions reduce delays as services can be processed closer to where they are needed.
- Massive data: Transmitting heavy data directly to the cloud is expensive, while local data analysis and filtering conserve bandwidth.
- Privacy: Processing enterprise and individual data at the edge ensures enterprise and operations security.
- Local autonomy: Offline processing and self-healing capabilities do not depend on the cloud.

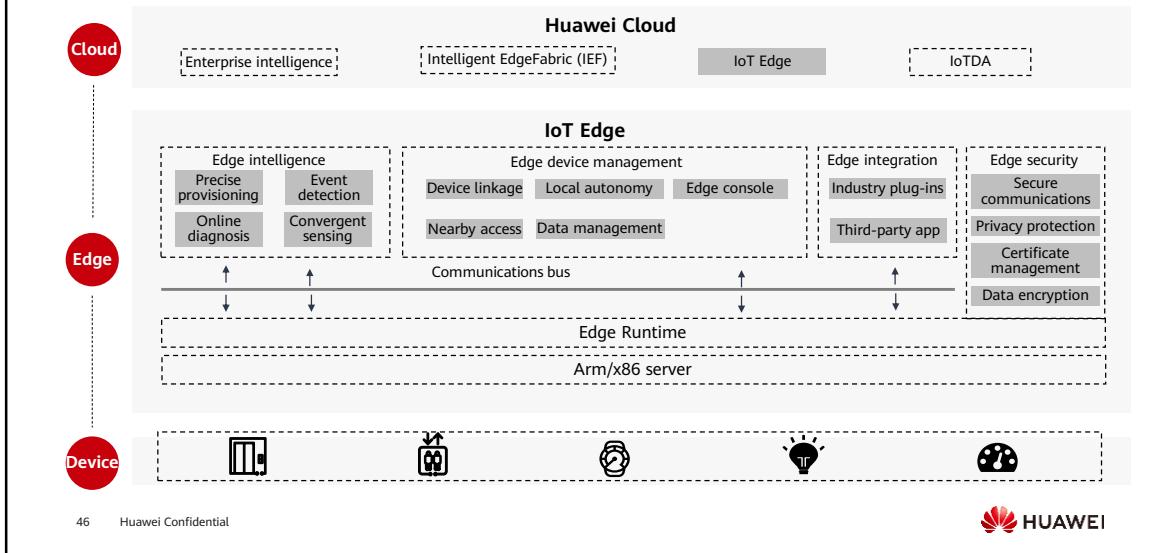
IoT Edge

- IoT Edge is an edge computing application for IoT. It is deployed at the edge close to devices or data sources to provide real-time services. Network, compute, storage, and application capabilities make applications intelligent and data secure.
- IoT Edge advantages:



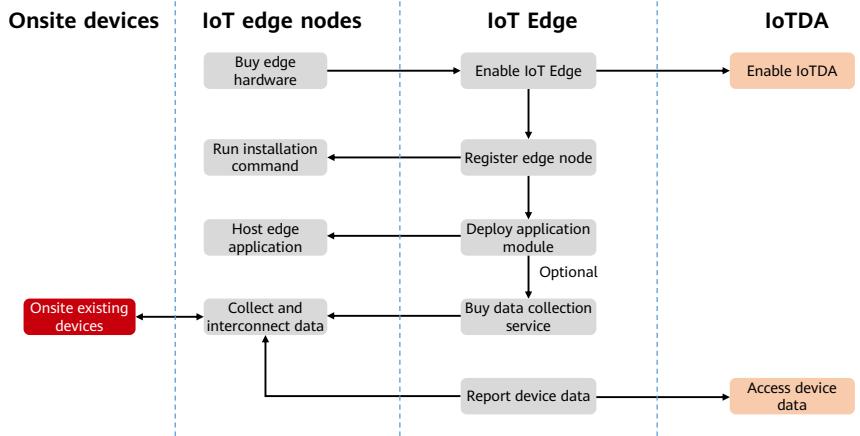
- IoT Edge is a group of software products, including cloud services, edge runtime software, and edge module applications. It quickly extends cloud capabilities to the edge and provides data collection, low-latency autonomy, cloud-edge collaboration, and edge computing. As the data ingestion point, it meets customers' requirements for device cloud migration, local computing, and data preprocessing in campus, city, and manufacturing scenarios.

IoT Edge Architecture



- **Device side:** Devices close to the customer's site can connect to the nearest edge node based on its data collection capability for device management, intelligent control, and data governance.
- **Edge node:** IoT Edge software is deployed on a gateway or server, which is managed as an edge node. The node implements device data collection, preprocessing, and data routing and forwarding. In addition, the edge side provides application hosting and edge computing, facilitating local autonomy and service expansion.
- **Cloud side:** The console is used to monitor and manage edge nodes, deliver configurations through edge-cloud synergy channels, deploy and upgrade applications remotely, and route and forward data to the cloud.

IoT Edge Workflow



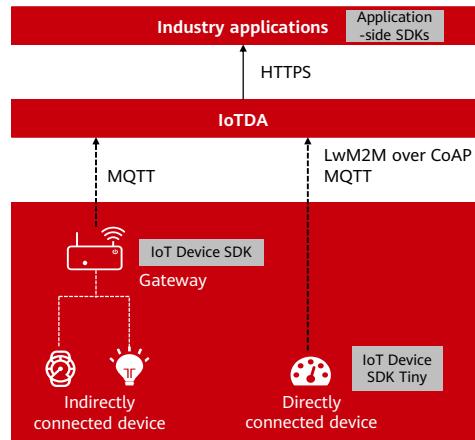
Contents

1. Requirements for IoT Platforms
2. IoT Platform Classification
3. Open Source IoT Platforms
4. Huawei Cloud IoT Full-Stack Services
- 5. Accessing the Huawei Cloud IoT Platform**

Accessing the Huawei Cloud IoT Platform

- Access modes:
 - Using SDKs
 - Device access: IoTDA provides IoT Device SDKs for Java (Linux/Windows), C (Linux), C# (Windows), Go Community Edition (Linux/Windows/Unix-like OS), OpenHarmony, and Android.
 - Application access: IoTDA provides SDKs for Java, Python, .NET, Go, Node.js, and PHP.
 - Using APIs
 - Device access: MQTT, MQTTS, and HTTPS-based APIs
 - Application access: HTTP and HTTPS-based APIs

Access using SDKs



Access using IoT Device SDKs in Different Scenarios

SDK	RAM	Flash	CPU Frequency	Language
IoT Device SDK Tiny	> 32 KB	> 128 KB	> 100 MHz	C

SDK	RAM	Flash	CPU Frequency	Language
IoT Device SDK	> 4 MB	> 2 MB	> 200 MHz	C, Java, C#, Go, and Android

The Device SDK Tiny is for devices with weak computing capabilities.



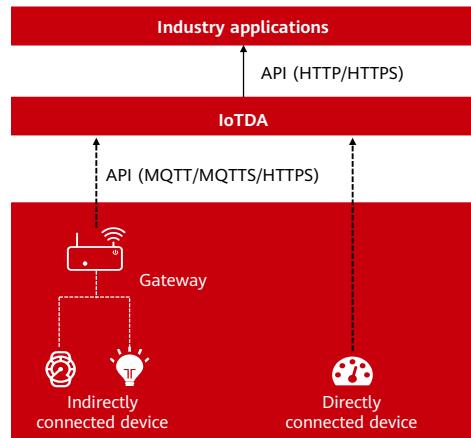
IoT Device SDKs are for intelligent devices and gateways with strong computing capabilities.



- MQTT and LwM2M over CoAP
- DTLS/TLS secure transmission protocols
- FOTA and SOTA
- Bootstrap and patterns client, server, and factory supported
- C supported

- MQTT
- Device binding, login, and data reporting
- Child device addition, deletion, modification, and data reporting
- Command receiving
- Java, C, Android, and C# supported
- Node.js, Python, and iOS supported soon

Access using APIs



Quiz

1. (True or false) Authentication is needed when connecting an IoT application to an IoT platform.
2. (Multiple-answer question) Which of the following are Huawei Cloud IoT full-stack services?
 - A. IoTDA
 - B. IoTA
 - C. IoT Stage
 - D. IoT Edge

- Answers:

- T
 - ABCD

Summary

- This section describes the requirements for IoT platforms, the architectures and use cases of Huawei Cloud IoT full-stack services, and how devices and applications access the Huawei Cloud IoT platform.

Recommendations

- Huawei Cloud IoTDA product documentation:
 - <https://support.huaweicloud.com/intl/en-us/iothub/index.html>

Acronyms or Abbreviations

- API: Application Programming Interface
- SDK: Software Development Kit

Thank you.

把数字世界带入每个人、每个家庭、

每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.



IoT Device-Cloud Connection Development



Foreword

- The Huawei Cloud IoT platform enables southbound and northbound data exchange. Data collected by sensors on southbound devices is reported to the platform, while operation instructions from the platform are sent back to the devices. This device-cloud connection needs to be mastered.
- This course gives developers the detailed steps of developing device-cloud connection using IoTDA.

Objectives

- Upon completion of this course, you will have learned:
 - Overall process of device-cloud connection
 - Product development process
 - Device development process
 - Main application development steps

Contents

- 1. Device-Cloud Connection Overview**
2. Product Development
3. Device Development
4. Application Development
5. Routine Cloud Management

Device-Cloud Connection Overview

- Device-cloud connection: the uploading of data (generated by devices at the sensing layer) to the platform layer (cloud) through multiple network media and delivering of necessary data to devices.
- IoTDA is a service of the Huawei Cloud IoT platform. It provides IoT-based connection for data exchange. Build your own IoT solution:

Product development

Manage products, develop product models and codecs, and perform online debugging on the platform.

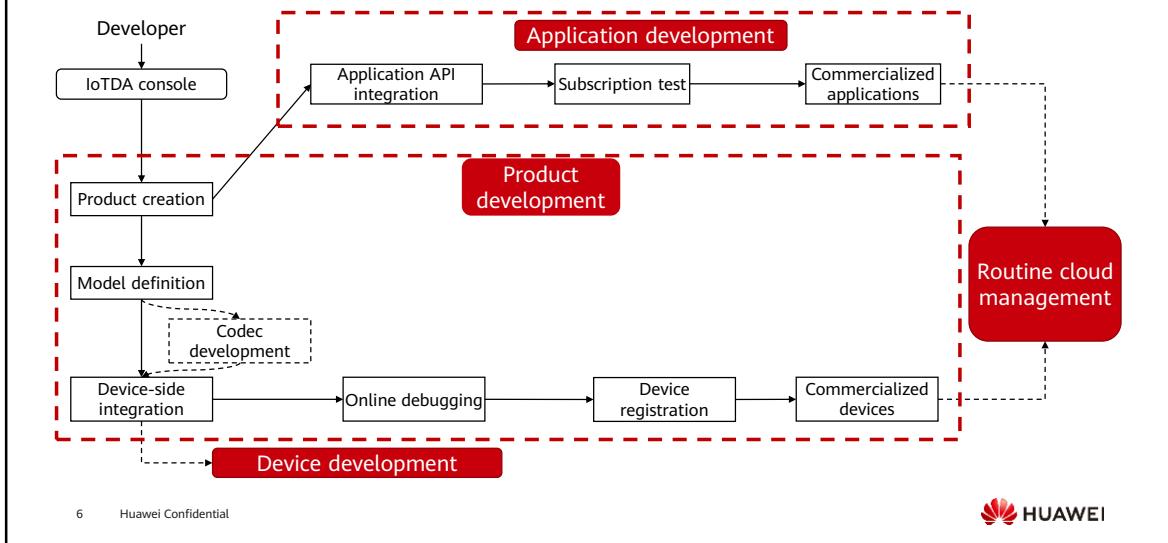
Application development

Carry out development for connection between applications and the platform, including calling APIs, obtaining service data, and managing HTTPS certificates.

Device development

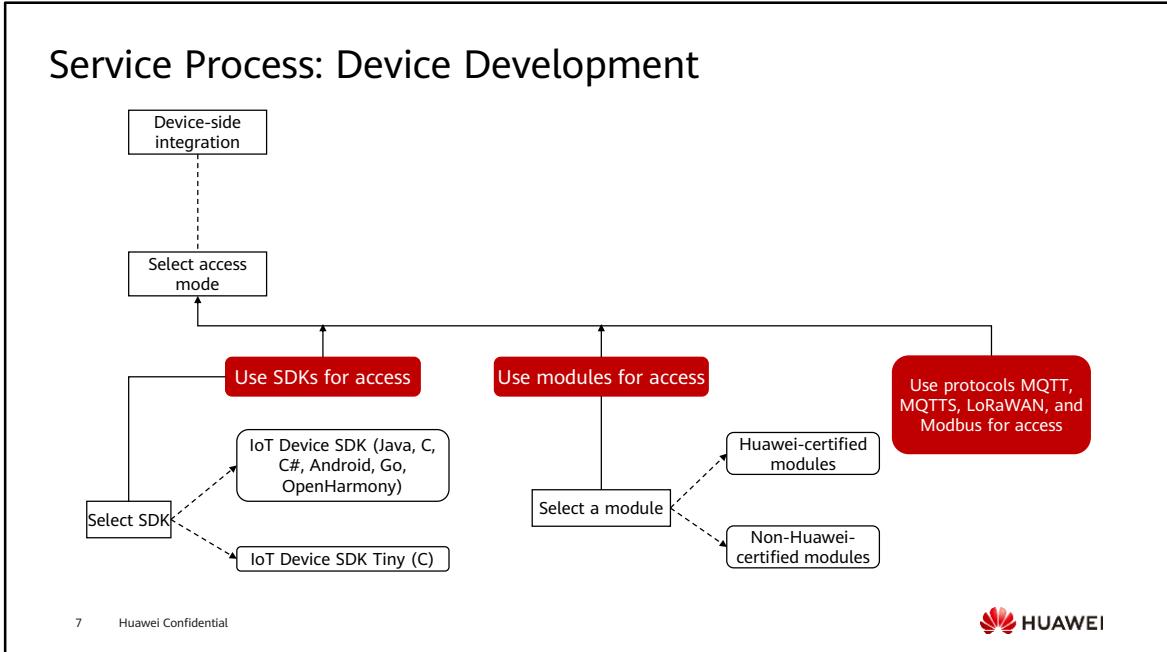
Develop the connection between devices and the platform, including connecting devices to the platform, reporting service data to the platform, and processing commands delivered by the platform.

Service Process: Product and Application Development

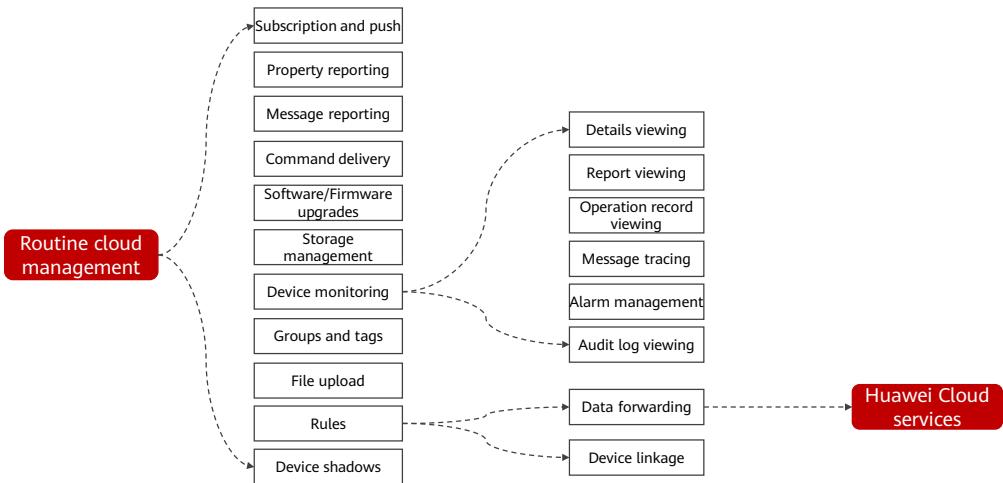


- **Product development:** You can perform development operations on the IoTDA console. For example, you can create a product or device, develop a product model or codec online, perform online debugging, carry out self-service testing, and release products.
- **Application development:** The platform provides robust device management capabilities through APIs. You can develop applications based on the APIs to meet requirements in different industries such as smart city, smart campus, smart industry, and IoV.

Service Process: Device Development



Service Process: Routine Cloud Management



Platform Access Details

- Developers use various protocols to connect devices or applications to IoTDA. The following example uses the basic edition in CN North-Beijing4.

Access Type	Access Protocol (Port)		Access Address
Application access	AMQPS (5671)		015f60c9fb.iot-amqps.cn-north-4.myhuaweicloud.com
	HTTPS (443)		iotda.cn-north-4.myhuaweicloud.com
Device access	CoAP (5683)	CoAPS (5684)	iot-coaps.cn-north-4.myhuaweicloud.com
	MQTT (1883)	MQTT (8883)	iot-mqtts.cn-north-4.myhuaweicloud.com
	HTTPS (443)		iot-https.cn-north-4.myhuaweicloud.com

- In addition, developers also use IoT Edge to connect devices using Modbus, OPC UA or proprietary protocols.

Contents

1. Device-Cloud Connection Overview

2. Product Development

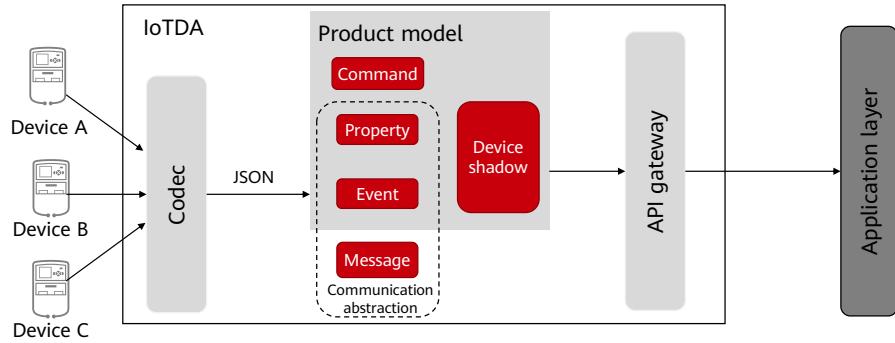
3. Device Development

4. Application Development

5. Routine Cloud Management

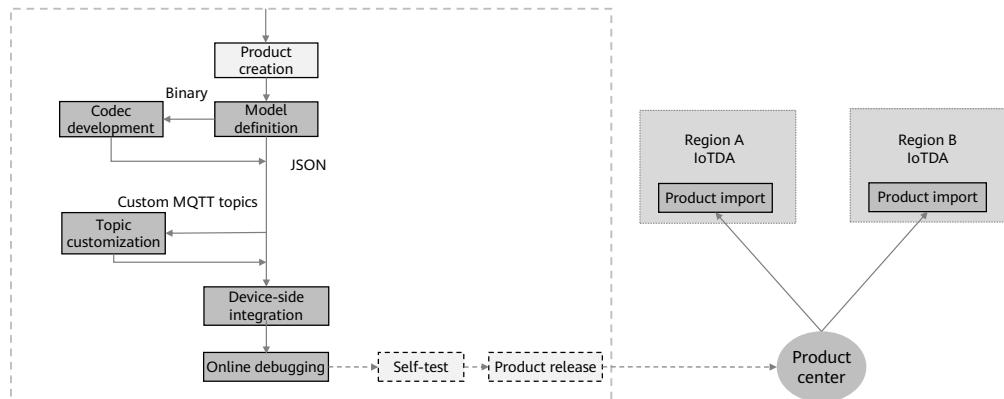
Introduction to Product Development

- In the IoT solution, developers can use the IoT platform to develop application programming interfaces (APIs) for applications to connect with devices that use various protocols. To manage devices, the platform must understand the capabilities of connected devices and the formats of data reported by devices. Therefore, developers need **product models** and **codecs** on the platform.



Product Development Process

- The IoTDA console has a one-stop development toolset for product models and codecs, including self-service testing.



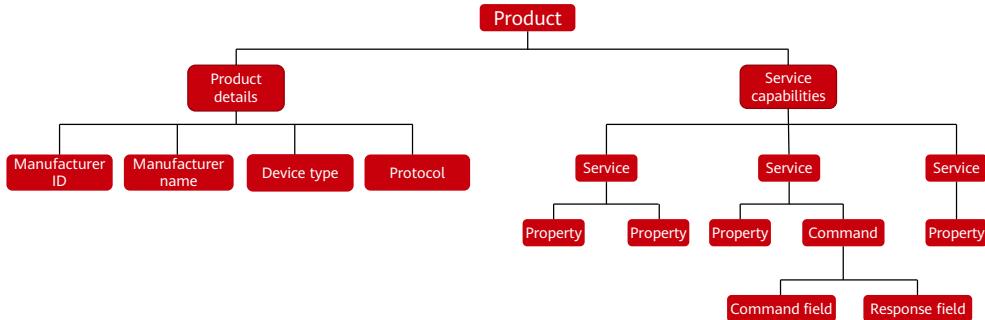
12 Huawei Confidential



- Product creation: A product is a collection of devices with the same capabilities or features. In addition to physical devices, a product includes product information, product models (profiles), and codecs generated during IoT capability building.
- Model definition: Product model development is the most important part of product development. A product model is used to describe the capabilities and features of devices. You can build an abstract model of a device by defining a product model on the platform so that the platform can know what services, properties, and commands are supported by the device.
- Codec development: If a device reports data in binary code stream format, you must develop a codec so that the platform can convert the binary format to the JSON format. If the device reports data in JSON format, you do not need to develop a codec.
- Online commissioning: The IoTDA console provides application and device simulators for you to commission data reporting and command delivery before developing real applications and physical devices. You can also use the application simulator to verify the service flow after the physical device is developed.

Introduction to Product Models

- A product model describes the capabilities and features of a device. Developers build an abstract model of a device by defining a product model on the Huawei Cloud IoT platform so that the platform can understand the services, properties, and commands supported by the device (such as color or any on/off switches).



Product Model Example

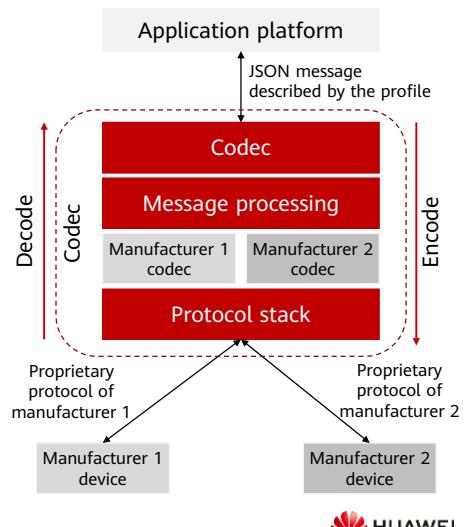
- The following figure is of a smart water meter.

Product	Description Item	Description
Product details	Manufacturer ID	Example parameter: TestUtf8Manuld
	Manufacturer name	Example parameter: HZYB
	Device type	Example parameter: WaterMeter
	Protocol type	Example parameter: CoAP
Service capabilities	Basics (WaterMeterBasic)	Defines parameters reported by the water meter: water flow, temperature, and pressure. If these parameters need to be controlled or modified using commands, you also need to define parameters in the commands.
	Alarm (WaterMeterAlarm)	Defines data reported by the water meter in various alarm scenarios. Commands need to be defined if necessary.
	Battery (Battery)	Defines meter data: voltage and current
	Transmission rule (DeliverySchedule)	Defines transmission rules for the water meter. Commands need to be defined if necessary.
	Connectivity (Connectivity)	Defines meter connection parameters.

- The Huawei Cloud IoT platform provides multiple methods for developing product models. You can select one that suits to your needs.
 - Customize Model** (online development): Build a product model from scratch.
 - Import from Library**: Use a product model preset on the platform to quickly develop a product. The platform provides standard and manufacturer-specific product models. Standard product models comply with industry standards and are suitable for devices of most manufacturers in the industry. Manufacturer-specific product models are suitable for devices provided by a small number of manufacturers. You can select a product model as required.
 - Import from Local** (offline development): Upload a local product model to the platform.
 - Import from Excel**: Define product functions by importing an Excel file. This method can lower the product model development threshold for developers because they only need to fill in parameters based on the Excel file. It also helps high-level developers and integrators improve the development efficiency of complex models in the industry. For example, an auto-control air conditioner model can contain more than 100 service items. Developing the product model by editing the Excel file greatly improves the efficiency, and you can edit and adjust parameters at any time.

Introduction to Codecs

- A codec **decodes** binary data reported by devices into JSON data that can be read by an application and **encodes** downstream command data in JSON format of the application into binary data that can be executed by devices.
- IoTDA provides three methods for developing codecs:
 - Graphical
 - Offline
 - Script-based



15 Huawei Confidential



- In the NB-IoT scenario, a codec can decode binary data reported by a device into the JSON format for the application to read, and encode the commands delivered by the application into the binary format for the device to understand and execute. CoAP is used for communications between NB-IoT devices and the IoT platform. The payload of CoAP messages carries data at the application layer, at which the data type is defined by the devices. As NB-IoT devices require low power consumption, data at the application layer is generally in binary format instead of JSON. However, the IoT platform sends data in JSON format to applications. Therefore, codec development is required for the platform to convert data between binary and JSON formats.
- Graphical development: The codec of a product can be quickly developed in a visualized manner on the IoTDA console.
- Offline development: A codec is developed through secondary development based on the Java codec demo to implement encoding, decoding, packaging, and quality inspection.
- Script-based development: JavaScript scripts are used to implement encoding and decoding.

Codec Development Example: Smart Agriculture (1)

- A developer has a smart agricultural device that uploads **temperature**, **humidity**, and **illuminance** data to the IoT platform, and receives control commands delivered by the IoT platform. Codec design roadmap:

Message list

Message Name	Message Type	messageId
Agriculture	Data reporting	00
Agriculture_Control_Light message	Command delivery	01
	Response	02

Agriculture message

Code Stream Offset	0-1	1-2	2-3	3-5
Field name	messageId	Temperature	Humidity	Luminance
Data type	int8u	int8u	int8u	int16u
Length	1	1	1	2
Example hexadecimal code stream value	00	19	3C	00 64

- messageId** is the address field, which indicates the message type. In the example, the address field **00** indicates a reported data record named **Agriculture**.
- The data type is configured based on the number of data reporting message types. The data type of **messageId** is **int8u** by default.
- The value of offset is automatically filled based on the location and the number of bytes of the field. **messageId** is the first field of the message. The start position is 0, the byte length is 1, and the end position is 1. Therefore, the offset of **messageId** is **0-1**.
- The length is automatically filled based on the data type.

Codec Development Example: Smart Agriculture (2)

Agriculture_Control_Light message

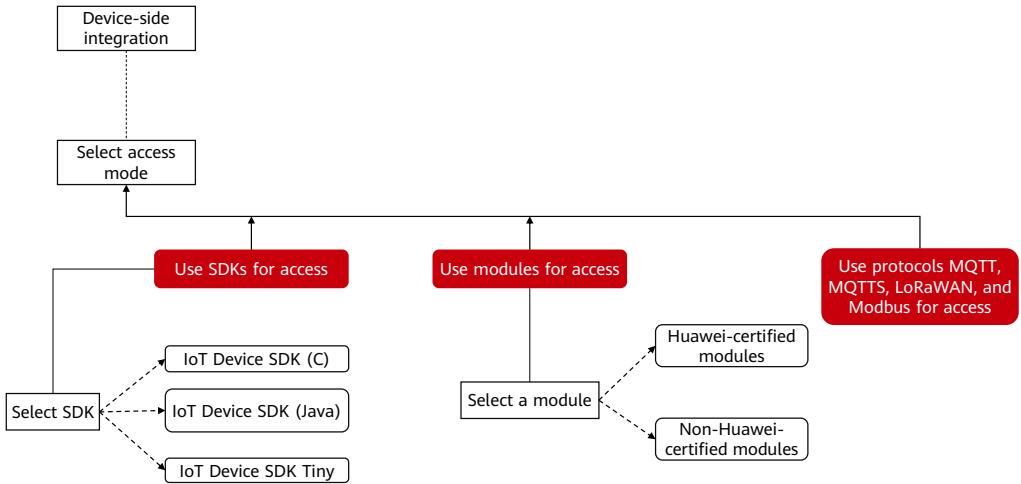
Code Stream Offset	0	1	2	3	4	5
Command field	messageld	mid	Light			
Data type	int8u	int6u	string			
Length	1	2	3			
Hexadecimal code stream	01	00	01	4F	4E	--
Response field	messageld	mid	errcode	Light_State	--	--
Data type	int8u	int6u	int8u	int8u	--	--
Length	1	2	1	1	--	--
Hexadecimal code stream	02	00	01	00/01	00/01	--

- The **mid** field is generated and delivered by the platform and is used to associate the delivered command with the command delivery response. The data type of **mid** is **int16u** by default.
- Add the **errcode** field to indicate the command execution status. **00** indicates success and **01** indicates failure. If this field is not carried in the response, the command is executed successfully by default. The data type of the **errcode** field is **int8u** by default.
- Add the **result** field to indicate the command execution result.

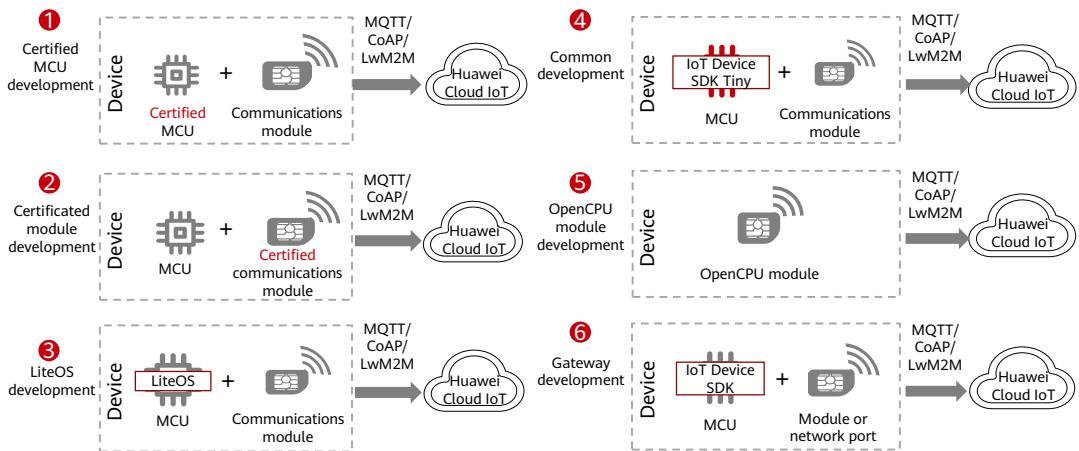
Contents

1. Device-Cloud Connection Overview
2. Product Development
- 3. Device Development**
4. Application Development
5. Routine Cloud Management

Device Development Process



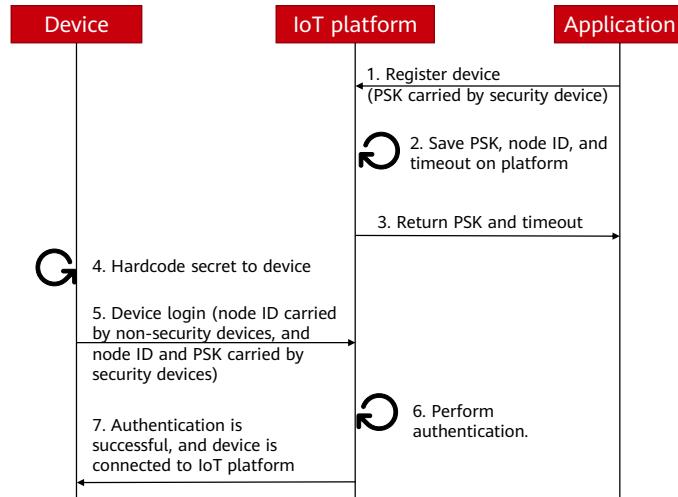
Device Development Modes



- For certified MCU development, the IoT Device SDK Tiny has been pre-integrated into the microcontroller unit (MCU) and can call methods to connect to the Huawei Cloud IoT platform. This mode is applicable to the scenario where devices need to be quickly put into commercial use, with low R&D costs. Devices are connected to the platform directly, without using gateways.
- For certificated module development, the IoT Device SDK Tiny has been pre-integrated into the module and can invoke AT commands to connect to the IoT platform. This mode is applicable to scenarios where there are few MCU resources. Devices are connected to the platform directly, without using gateways.
- For LiteOS development, devices run LiteOS that manages MCU resources. In addition, LiteOS has a built-in IoT Device SDK Tiny that can call functions to connect to the platform. This mode shortens the device development duration and reduces the development difficulty. No OS is required. Devices are connected to the platform directly, without using gateways.
- For common development, the IoT Device SDK Tiny is integrated into the MCU and calls the SDK functions to connect to the platform, which is more convenient than access by calling APIs. This mode is applicable to the scenario where there is sufficient time for devices to put into commercial use, and the flash and RAM resources of the MCU meet the conditions for integrating the IoT Device SDK Tiny.
- For OpenCPU module development, the MCU capability in the common module is used, and device applications are compiled and run on the OpenCPU. This development mode is applicable to devices that have high security requirements, are small in size, and need to be quickly put into commercial use.
- For gateway device development, the IoT Device SDK is pre-integrated into the CPU or MCU and can call functions to connect to the platform. This development

mode is applicable to gateways that manage child devices.

Access Using LwM2M over CoAP

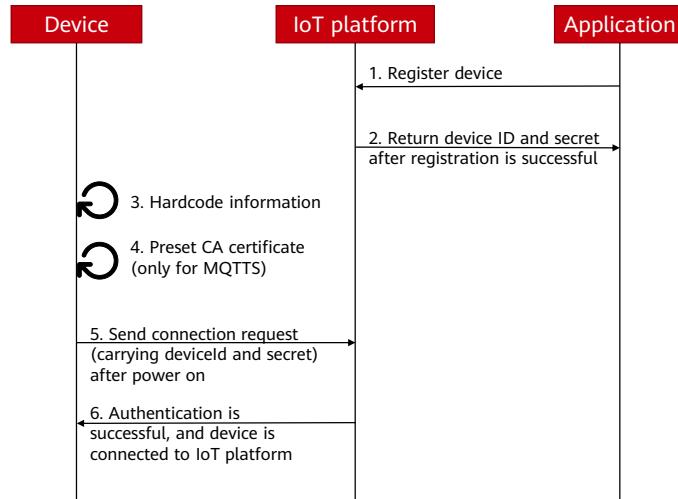


21 Huawei Confidential



- 1. An application calls the API **Creating a Device** to register a device. Alternatively, a user uses the IoTDA console to register a device.
- 2–3. The platform allocates a globally unique PSK to the device and returns **timeout**.
- 4. The user hardcodes PSK into the device hardware, software, or firmware.
- 5. The user powers on the device. The device sends a connection request carrying the node ID (for example, IMEI) and PSK.
- 6–7. If the authentication is successful, the platform returns a success message, and the device is connected to the platform.

Access Using Native MQTT or MQTTS



22 Huawei Confidential



- 1. An application calls the API **Creating a Device** to register a device. Alternatively, a user uses the IoTDA console to register a device.
- 2. The platform allocates a globally unique device ID and secret to the device.
- 3. The user hardcodes the device ID and secret to the device hardware, software, or firmware.
- 4. (Optional) The user presets the CA certificate on the device. This step is required only for devices connected using MQTTS.
- 5. The user powers on the device. The device sends a connection request carrying the device ID and secret.
- 6. If the authentication is successful, the platform returns a success message, and the device is connected to the platform.

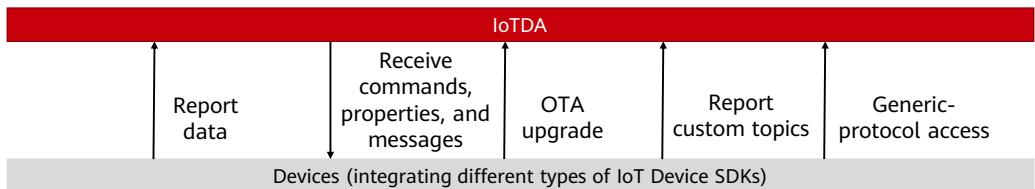
Device Authentication

- The Huawei Cloud IoT platform authenticates devices attempting to access it. The process depends on the access method.

Access Type	Description
Device connected using LwM2M over CoAP	Call the API Creating a Device or use the IoTDA console to register a device with the platform, and set the node ID (for example, the IMEI) as the verification code. The device uses the node ID to get authenticated and connect to the platform. When Datagram Transport Layer Security (DTLS) or DTLS+ is used, the transmission channel between the device and platform is encrypted by using a PSK.
Device using native MQTT or MQTTS	Call the API Creating a Device or use the IoTDA console to register a device with the platform, and hardcode the device ID and secret returned by the platform into the device. A Certification Authority (CA) certificate is preset on MQTTS (not MQTT) devices. The device uses the device ID and secret for authentication and connection to the platform.

Access Using IoT Device SDKs

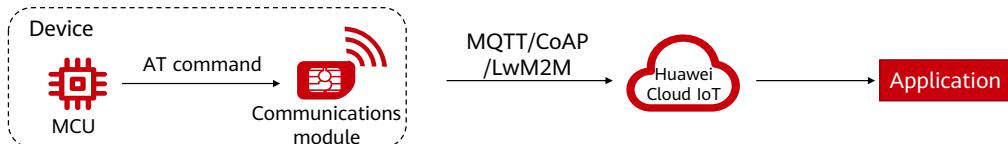
- IoT Device SDKs help you quickly connect devices to the Huawei Cloud IoT platform. After being integrated with an IoT Device SDK, devices that support the TCP/IP protocol stack can **directly communicate with the IoT platform**. Devices that do not support the TCP/IP protocol stack, such as Bluetooth and ZigBee devices, need a **gateway integrated with the IoT Device SDK to communicate with the platform**.



- To use the SDK to connect to the platform, you need to:
 - Create a product on the platform or by calling the API **Creating a Product**.
 - Register a device on the platform or by calling the API **Registering a Device**.
 - Implement the functions demonstrated in the figure, including reporting messages/properties, receiving commands/properties/messages, OTA upgrades, topic customization, and generic-protocol access.

Access Using Communications Modules

- Devices can access the Huawei Cloud IoT platform through communications modules. Non-Huawei-certified modules: devices access the platform using standard AT commands in various network connection modes. Huawei-certified modules: devices access the platform based on Huawei-specified AT command specifications.



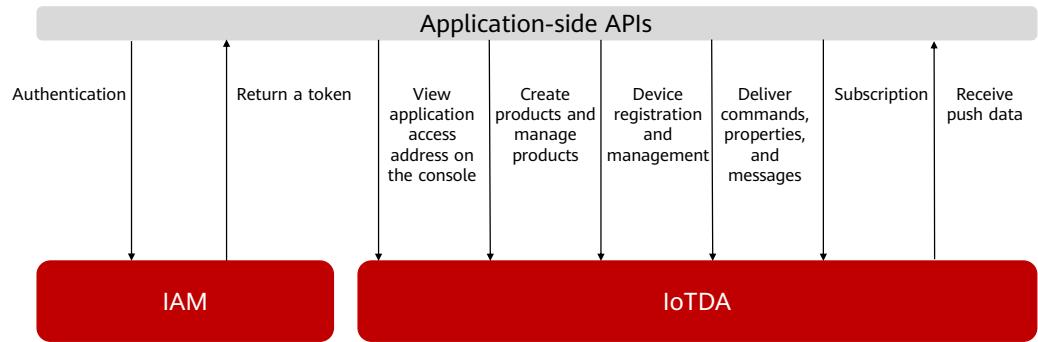
- Huawei certified modules are pre-integrated with the IoT Device SDK Tiny. They have achieved Huawei certification, and comply with Huawei AT command specifications. The following benefits are available for using Huawei certified modules:
 - Device manufacturers do not need to worry about how to connect devices to the Huawei Cloud IoT platform on the MCU (for example, how to set the secret encryption algorithm and clientID composition during MQTT connection setup). To connect their devices to the platform, they only need to invoke AT commands. This accelerates device interconnection and commissioning.
 - The MCU does not need to integrate the MQTT protocol stack or IoT Device SDK Tiny, greatly reducing MCU resource consumption.
 - Huawei releases certified modules on Huawei Cloud Marketplace so that device manufacturers and service providers can purchase these certified modules to quickly connect devices to the Huawei Cloud IoT platform.

Contents

1. Device-Cloud Connection Overview
2. Product Development
3. Device Development
- 4. Application Development**
5. Routine Cloud Management

Application Development Overview

- The Huawei Cloud IoT platform provides APIs for easy and efficient application development. Developers call these open APIs to quickly integrate platform functions, such as product, device, subscription, and rule management, as well as command delivery.



Application-side APIs

- The Huawei Cloud IoT platform provides various northbound RESTful APIs for application developers based on the capabilities provided by the platform.
- Before calling the APIs in the right table, developers need to complete application **authentication**. Either of the following authentication methods call the authentication API:
 - Token:** General requests are authenticated using tokens.
 - AK/SK:** Requests are encrypted using an AK/SK.

Application-side APIs	
Product management	Data transfer
Device management	Device linkage rules
Device messages	Device shadows
Device commands	Device group management
Device properties	Tag management
AMQP queue management	Resource space management
Access credential management	Batch tasks
Data forwarding rule management	Device CA certificate management

Making an API Request

- The IoT platform provides Representational State Transfer (RESTful) APIs, which can be called using HTTPS requests.
- Request URI:** {URI-scheme}:// {Endpoint} / {resource-path}?{query-string}
- Request header**
 - Additional fields in the request header include those required by a specified URI or HTTP method. For example, to request the authentication information, add **Content-Type** to specify the request body type.
- Request body**
 - A request body is usually a structured format. It corresponds to **Content-Type** in the request header and transfers content except the request header. If the request body contains Chinese characters, encode them using UTF-8.

Request Method	Description
GET	Obtains resources from the server.
POST	Creates a resource from the server.
PUT	Updates resources on the server.
DELETE	Deletes resources from the server.
HEAD	Requests a server resource header.
PATCH	Requests a server to update part of specified resources. If the resource does not exist, the PATCH method creates a new resource.

Request method

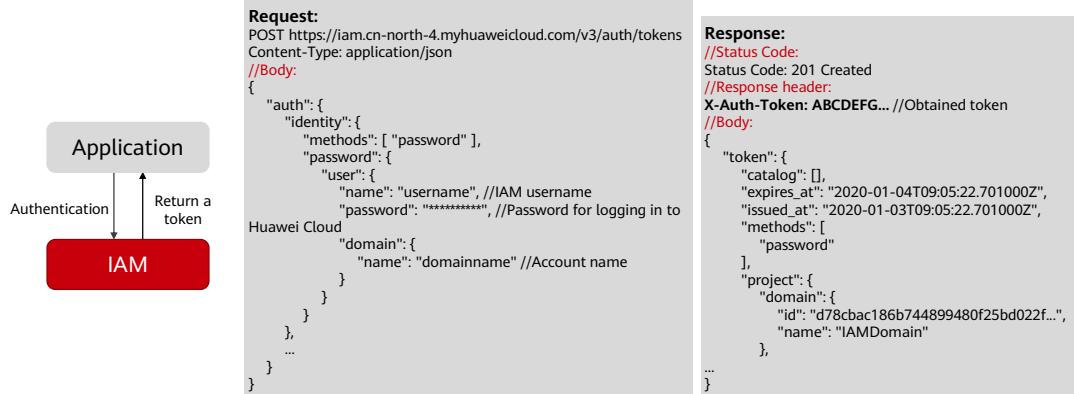
- URI-scheme:** protocol used to transmit requests. Currently, all APIs use HTTPS.
- Endpoint:** domain name or IP address of the server bearing the REST service endpoint. Obtain the value from **Regions and Endpoints**. For example, the endpoint of IoTDA in the CN North-Beijing4 region is **iotda.cn-north-4.myhuaweicloud.com**.
- resource-path:** access path of an API for performing a specific operation. Obtain the path from the URI of an API. For example, **resource-path** of the API for querying products is **/v5/iot/{project_id}/products/{product_id}**.
- query-string:** optional query parameter. Ensure that a question mark (?) is included before each query parameter that is in the format of "*Parameter name=Parameter value*". For example, **? limit=10** indicates that a maximum of 10 data records will be displayed.
- Common request header fields:
 - Content-Type:** request body type or format. This field is mandatory and its default value is **application/json; charset=utf-8**. Other values of this field will be provided for specific APIs if any.
 - X-Auth-Token:** user token. This field is mandatory when token-based authentication is used. You can call the API used to obtain a user token to obtain the value. In the response message header returned by the API, **X-Subject-Token** is the desired user token.

Response

- **Status code**
 - After sending a request, you will receive a response that includes a status code, response header, and response body.
 - A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. If **201** is returned for calling an API, the request is successful.
- **Response header**
 - Similar to a request, a response also has a header.
 - For the API used to create a product, header fields such as **Content-Type** and **Date** will be returned.
- **Response body**
 - A response body is usually a structured format, corresponding to the **Content-Type** in the response header, and is used to transfer content other than the response header.

Token-based Authentication

- A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API.



31 Huawei Confidential



- A token is a character string generated by the server and is used by a client to send a request. After the first login, the server generates a token and returns the token to the client. In subsequent requests, the client needs to carry the token, but not the username and password. The validity period of a token is 24 hours. It starts from the time when the client obtains the token. If the same token needs to be used for authentication, it is recommended that the token be cached to avoid frequent calling. Before the token expires, you must update the token or obtain a new token. Otherwise, the authentication on the server will fail after the token expires.
- If you obtain the token for multiple times, the latest token is used. The previous token will be overwritten and become invalid.

AK/SK-based Authentication

- AK/SK-based authentication uses AK/SK to sign requests, and the signature is then added to request headers for authentication.
 - AK: access key ID for identifying a user. It is a unique identifier used with a secret access key to sign requests cryptographically.
 - SK: secret access key used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents request modification.
- AK/SK usage mechanism
 - After an ECS receives a request, the system generates an authentication character string using the same SK and authentication mechanism and compares the generated string with the one in the request. If the strings are the same, the system decides that the user has the required operation permissions and performs the requested operation. If they are different, the system ignores the operation and returns an error code.

- AK/SK-based authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, use token-based authentication.

Product Creation

- Before connecting a device to the platform, an application must call the API **Creating a Product**. The product created will be used during device registration.

Request:

```
POST  
https://{{IAMEndpoint}}/v5/iot/{{{project_id}}}/products  
Content-Type: application/json  
X-Auth-Token:{{X-Auth-Token}}  
{  
    "name": "Thermometer",  
    "device_type": "Thermometer",  
    "protocol_type": "MQTT",  
    "data_format": "binary",  
    "manufacturer_name": "ABC",  
    "industry": "smartCity",  
    "description": "this is a thermometer produced by  
Huawei",  
    ....  
}
```

Response:

```
Status Code: 201 Created  
Content-Type: application/json  
{  
    "app_id": "46dead3858bb4582a5de3e9ecb06cacc",  
    "app_name": "DefaultApp_HCNAIoT_iot",  
    "product_id": "5e8456df536e0502ec6204d0",  
    "name": "Thermometer",  
    "device_type": "Thermometer",  
    "protocol_type": "MQTT",  
    "data_format": "binary",  
    "manufacturer_name": "ABC",  
    "industry": "smartCity",  
    "description": "this is a thermometer produced by  
Huawei",  
    ....  
}
```

Delivering an Asynchronous Command

- Developers can call this API to deliver a command to an NB-IoT device on the Huawei Cloud IoT platform or an application. The platform delivers commands to the device and asynchronously returns the command execution result to the application.

Request:

```
POST  
https://{{Endpoint}}/v5/iot/{{project_id}}/  
devices/{{device_id}}/async-commands  
Content-Type: application/json  
X-Auth-Token: *****  
Instance-Id: *****  
  
{  
    "service_id": "Switch",  
    "command_name": "ON_OFF",  
    "paras": {  
        "value": "ON"  
    },  
    "expire_time": 0,  
    "send_strategy": "immediately"  
}
```

Response:

```
Status Code: 200 OK  
Content-Type: application/json  
{  
    "device_id": "c1224afb-e9f0-4916-8220-b6bab568e888",  
    "command_id": "b1224afb-e9f0-4916-8220-  
b6bab568e888",  
    "service_id": "Switch",  
    "command_name": "ON_OFF",  
    "send_strategy": "immediately",  
    "paras": {  
        "value": "ON"  
    },  
    "expire_time": 0,  
    "status": "SENT",  
    "created_time": "20151212T121212Z"  
}
```

- This API is used only to deliver commands asynchronously to NB-IoT devices. Synchronous commands can be delivered to devices that use MQTT.

Contents

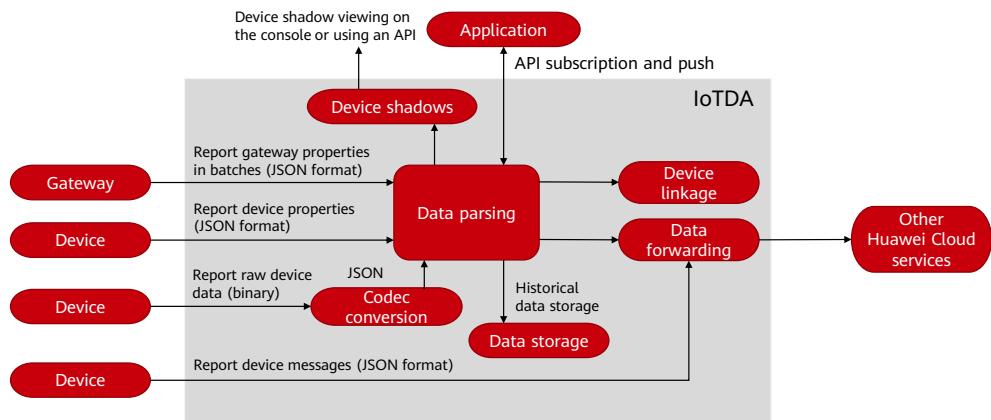
1. Device-Cloud Connection Overview
2. Product Development
3. Device Development
4. Application Development
- 5. Routine Cloud Management**

Communications

- Routine management on the Huawei Cloud IoT platform needs many platform functions: **property reporting** and **command delivery** in message communications.
- The Huawei Cloud IoT platform supports bidirectional device communications. Developers reports data to the platform through device-side APIs, and the platform pushes the data to applications by using the subscription/push mechanism or forwards the data to other Huawei Cloud services. Devices are remotely controlled by means of command delivery (using APIs or the IoTDA console).

Data Type	Message Type	Difference	Similarity
Upstream data	Property reporting	Dependent on product model. The properties reported must match those defined in the product model.	Both properties and messages can be reported to the platform by calling device-side APIs, and forwarded to other Huawei Cloud services using rules.
	Message reporting	Independent from product model. The platform does not verify the message content.	
Downstream data	Command delivery	Dependent on product model. The commands delivered must match those defined in the product model. The command delivery is synchronous. (After a command is delivered, the platform waits for a response. If no response is returned, the command delivery times out.)	Commands, properties, and messages can be delivered by calling application-side APIs.
	Property delivery (for modification purposes)	Dependent on product model. The properties delivered must match those defined in the product model. Property delivery is synchronous.	
	Message delivery	Independent from product model. The platform delivers messages to the device. Message delivery is asynchronous. (After a message is delivered, the platform does not need to wait for a response from the device.)	

Data Reporting



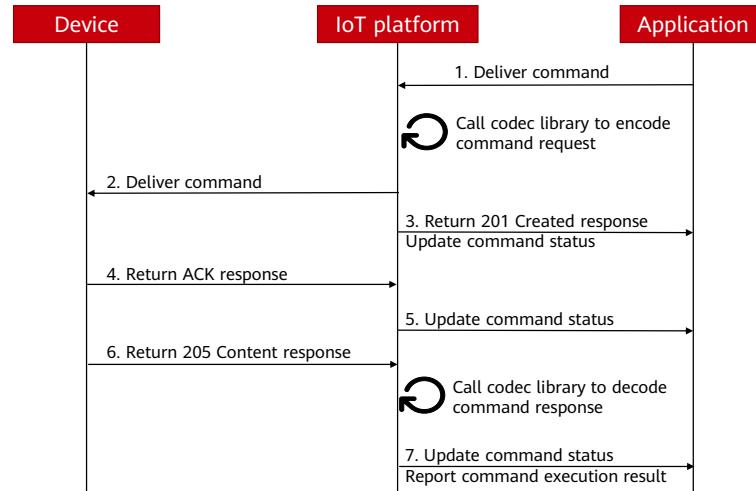
- A device connected to the IoT platform sends data to the platform in the following ways:
 - Reporting device messages: A device reports custom data to the platform through message reporting APIs. The platform does not parse or store the messages reported. Instead, it forwards the messages to other Huawei Cloud services for storage and processing based on data forwarding rules. Then, the data is further processed through the consoles or APIs of the other services.
 - Reporting raw device data (binary): A device reports raw code streams through binary reporting APIs. The platform uses codecs to parse the raw data into JSON data defined in the product model and then performs subsequent processing.
 - Reporting device properties: A device reports property data defined in the product model through property reporting APIs. The platform parses the data and then performs subsequent processing.
 - Report gateway properties in batches: A gateway reports data of a batch of devices to the platform at a time. The platform parses the data and then performs subsequent processing.

Command Delivery

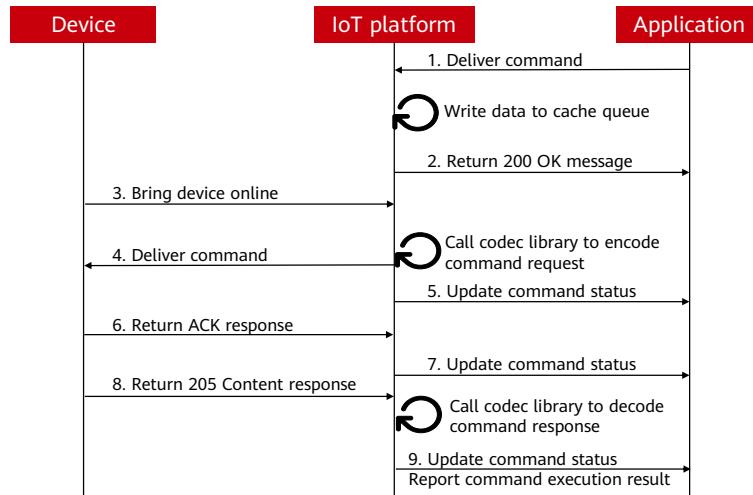
- The Huawei Cloud IoT platform provides **synchronous** and **asynchronous** command delivery.

Mechanism	Definition	Application Scenario	Devices Using LwM2M over CoAP	Devices Using MQTT
Synchronous command delivery	The platform synchronously sends commands to the device and returns the command execution result to the application. If the device does not respond, the platform returns a timeout message to the application.	Applicable to commands that must be executed in real time, for example, turning on a street lamp or closing a gas meter switch. Applications determine the appropriate time to deliver a command.	Unsuitable	Suitable
Asynchronous command delivery	The platform sends the command to the device and asynchronously pushes the command execution result to the application. Asynchronous command delivery is classified into the following two types: <ul style="list-style-type: none">Immediate delivery: The platform delivers commands to a device upon receiving a command regardless of whether the device is online. If the device is offline or the device does not receive the command, the delivery fails.Delayed delivery: After receiving a command, the platform caches a command and delivers it to a device when the device goes online or reports data. If a device has multiple pending commands, the platform delivers the commands in sequence.	<ul style="list-style-type: none">Immediate delivery is for scenarios that demand timeliness.Delayed delivery is for commands that do not need to be executed immediately, for example, configuring water meter parameters.	Suitable	Unsuitable

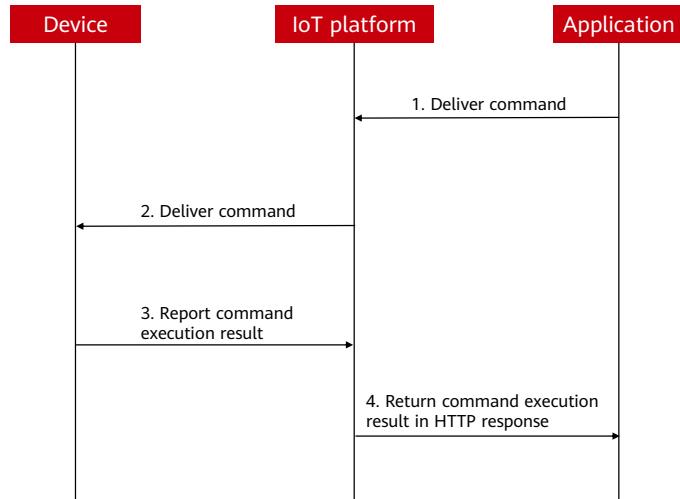
Immediate Delivery for Devices Using LwM2M over CoAP



Delayed Delivery for Devices Using LwM2M over CoAP



Delivery of MQTT Device Commands



Quiz

1. (True or False) An IoT application must be authenticated before being connected to the Huawei Cloud IoT platform.
2. (Single-answer question) In the IoTDA, which of the following data needs to be converted by the codec?
 - A. Use the gateway to report JSON data in batches
 - B. The device reports attributes in JSON format
 - C. The device reports data in binary format
 - D. The device reports a message in JSON format

- Answers:
 - T
 - C

Summary

- This course describes how to develop device-cloud connection on the IoT platform. The process includes product, device and application development, and routine cloud management. Product Development: how to develop product models and codecs. Device Development: multiple device development modes. Application Development: classification of application-side APIs and how to perform authentication. Routine Management in the Cloud: how to report data and deliver commands.

Acronyms and Abbreviations

- ACK: Acknowledge Character
- URI: Uniform Resource Identifier

Thank you.

把数字世界带入每个人、每个家庭、

每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

Copyright©2023 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

