

IoT Explorer

Console Operation Guide



Copyright Notice

©2013–2025 Tencent Cloud. All rights reserved.

The complete copyright of this document, including all text, data, images, and other content, is solely and exclusively owned by Tencent Cloud Computing (Beijing) Co., Ltd. ("Tencent Cloud"); Without prior explicit written permission from Tencent Cloud, no entity shall reproduce, modify, use, plagiarize, or disseminate the entire or partial content of this document in any form. Such actions constitute an infringement of Tencent Cloud's copyright, and Tencent Cloud will take legal measures to pursue liability under the applicable laws.

Trademark Notice



This trademark and its related service trademarks are owned by Tencent Cloud Computing (Beijing) Co., Ltd. and its affiliated companies ("Tencent Cloud"). The trademarks of third parties mentioned in this document are the property of their respective owners under the applicable laws. Without the written permission of Tencent Cloud and the relevant trademark rights owners, no entity shall use, reproduce, modify, disseminate, or copy the trademarks as mentioned above in any way. Any such actions will constitute an infringement of Tencent Cloud's and the relevant owners' trademark rights, and Tencent Cloud will take legal measures to pursue liability under the applicable laws.

Service Notice

This document provides an overview of the as-is details of Tencent Cloud's products and services in their entirety or part. The descriptions of certain products and services may be subject to adjustments from time to time.

The commercial contract concluded by you and Tencent Cloud will provide the specific types of Tencent Cloud products and services you purchase and the service standards. Unless otherwise agreed upon by both parties, Tencent Cloud does not make any explicit or implied commitments or warranties regarding the content of this document.

Contact Us

We are committed to providing personalized pre-sales consultation and technical after-sale support. Don't hesitate to contact us at 4009100100 or 95716 for any inquiries or concerns.

Contents

Console Operation Guide

Instance Management

Instance Overview

Instance Management

Migrating Instance Device Data

Public Instance Device Integration

Product Management

Product Development

Thing Model Definition

Device Development

Interactive Development

Device Debugging

Batch Production

Equipment Mass Production

Mass Production Management

Activation Code Overview

After-Sales Operation and Maintenance

Firmware Upgrade

Device management

Operational Analysis

Enterprise Instance Device Connectivity

Product Management

Product Development

Thing Model Definition

Device Development

Device Debugging

Batch Production

Device management

Firmware Upgrade

Operational Analysis

Application Development

Data Flow

Message Push

Rule Engine

Rule Engine Overview

Data Processing

Rule Function

Data Forwarding to Another Topic

Forwarding Data to a Third-Party Service

Forward Data to TDMQ CKafka

Data Forwarding to Cloud MySQL

Data Forwarding to Cloud Development

Data Forwarding to Cloud Component TDSQL-MySQL

Value-Added Services

Voice Skills

Alexa Voice Skill Service

Cloud Xiaowei Voice Skill Service

Google Voice Skill Service

Xiaodu Voice Skill Service

Location Service

Add Locating Attribute Function

Space Management

Spatial Visualization

Geofencing

Historical Trajectory

Related Thing Model Descriptions

Resource Management

Advanced Functions of the Product

Apply for This Advanced Function

Version Change

Tencent Real – Time Communication (TRTC)

QQ Music Service

Voice Recognition

Voice Assistant

Kugou Music Service

Control Authorization

Video AI Analysis

WeChat Strong Reminder Notification

TWeCall

TWeSee Service

Video Summary

Permission Management

Creating Sub-account

How to Configure CAM Permission for a Sub-Account

Console Operation Guide

Instance Management

Instance Overview

Last updated: 2025-04-27 17:28:07

Instance Overview

An instance is a unit of computing resource on the cloud that can provide device connectivity and management services for user's IoT solutions. The platform is used for device connectivity, device message communication, device management and other business services based on instances. Tencent Cloud IoT Explorer provides two types of instances, public instance and enterprise instance, to meet the access and management of Consumer IoT and industrial IoT solutions respectively.

Public Instance

After the user successfully activates Tencent Cloud IoT Explorer, they can purchase the public instance activation code on demand to use the public instance. If the user's IoT application scenario is full-house intelligent, wearable health, intelligent security (IPC, lock, doorbell, etc.), or consumer electronics intelligent scenarios that require C-end mini program application and device interaction, they can choose a public instance. Such applications often have the characteristics of typical consumer electronics. Public instances are generally billed by purchasing the device activation code in a single transaction.

Public instances provide various types of activation codes for device access and communication in different scenarios:

- **Device activation code:** Refers to the normal whole-house intelligent devices. Directly connected devices or gateway devices usually perform data interaction with the Internet of Things Platform via the MQTT protocol.
- **Activation code for audio and video devices:** Refers to the activation code required for real-time audio and video communication scenarios with the App or mini program for IPC, smart door lock, visual doorbell, etc. Contact sales is required for purchasing this activation code, with a minimum quantity requirement.

Note:

- An account contains only one public instance.
- When users use a public instance to complete device connectivity or application development and need a greater number of device creations and message upstream and downstream TPS, they can purchase an activation code for the public instance. If the upstream and downstream TPS of device messages in the user's actual application scenarios needs to be greater than 1% of the number of activation codes purchased, they can [submit a ticket](#) to contact us.

Enterprise Instance

If the user's IoT application scenarios are Internet of Vehicles in the travel industry, self-service retail, sharing and leasing, park energy management, construction machinery asset management, etc. in the commercial field, such applications often need to connect devices to the IoT platform as needed. Users

usually build a business system to perform unified device management through the IoT. Such industrial IoT solutions often use enterprise instances. Users can select suitable instance specifications based on the number of devices online in their applications, the upstream and downstream TPS of device messages, the message forwarding TPS of the rule engine, and the usage duration.

Note:

Enterprise instances are generated after on-demand purchase. One account can purchase multiple enterprise instances.

Difference between Public Instance and Enterprise Instance

| Project | Public Instance | Enterprise Instance |
|--|---|---|
| Use cases | Consume IoT scenarios, such as whole-house intelligence, wearable health, intelligent security, home furnishing, etc. | Industrial IoT solution scenarios, such as Internet of Vehicles in the travel industry, self-service retail, sharing and leasing, park energy management, construction machinery asset management, etc. |
| Billing Mode | Activation code method | Monthly Subscription |
| SLA | 99.95% | 99.95% |
| Device message upstream and downstream TPS | Number of device activation codes * 1% | 10 messages per second per online device |
| Rule engine message forwarding TPS | User reported Message TPS | Actual reported Message TPS by users |
| LoRaWan service | Not supported. | Supported |
| Tencent Lianlian official mini program | Supported | Not supported. |
| Device authentication method | Key Authentication | Key authentication, certificate authentication |
| Custom Topic | Unsupported. By default, topics starting with \$thing are supported. Develop according to the specification. | Support Thing Model Topics and support user customization of Topics. |
| Message push | Support push to Tencent Lianlian | Not supported. |

| | | |
|---|--------------------------------|----------------|
| | official mini program and App. | |
| Value-added service -> TRTC | Supported | Supported |
| VAS -> location service | Supported | Supported |
| Value-Added Services -> Resource Management | Supported | Supported |
| VAS -> voice skills | Supported | Not supported. |

Instance Management

Last updated: 2025-04-27 17:28:23

Business Introduction

Users can enable public instances through instance management, purchase public instance activation codes, purchase enterprise instances, and renew and upgrade enterprise instances according to business development. They can also perform data migration of devices between instances.

Operation Steps

Enabling Public Instance

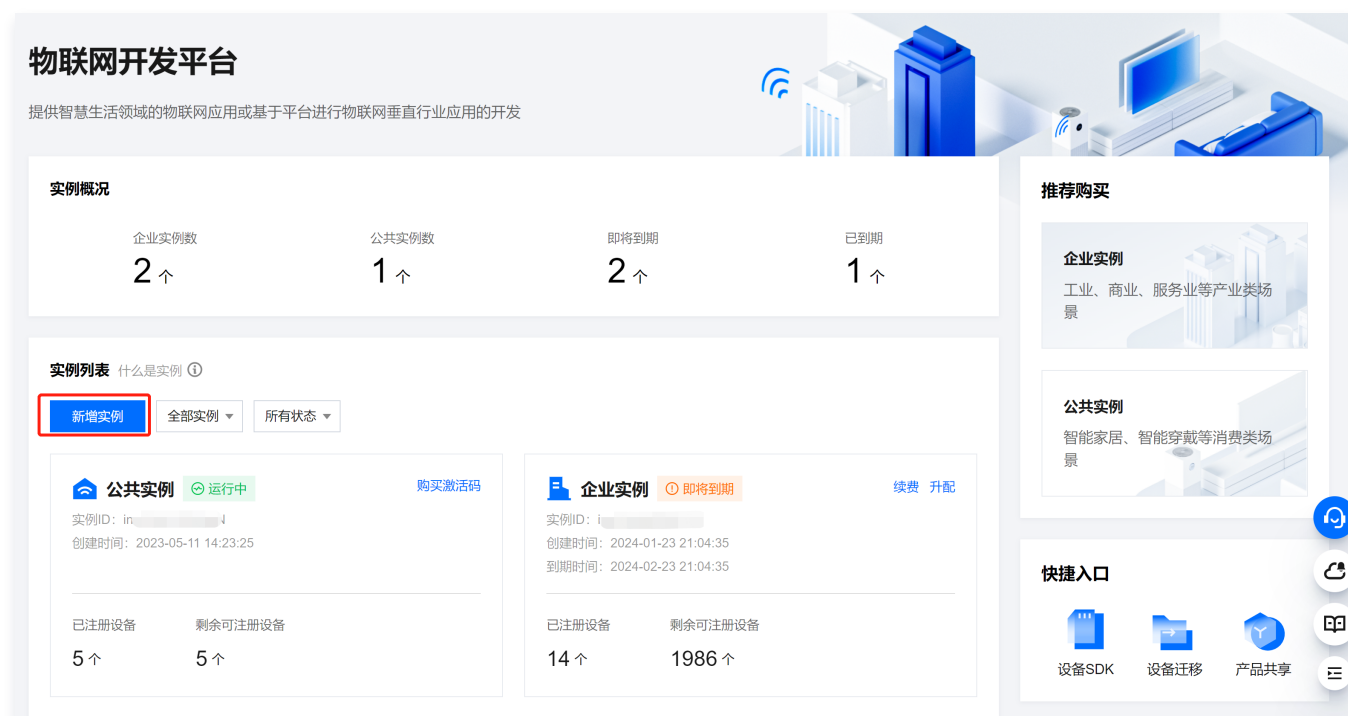
Starting from June 20, 2024 (see [documentation announcement](#)), public instances will no longer provide users with 10 free device activation codes and default activation of public instances. Users need to purchase activation codes for automatic activation of public instances. With a throughput capacity of 1 TPS for uplink and downlink messages, it meets users' needs to connect devices to the platform during the development phase and perform joint debugging with users' business systems.

Purchase Public Instance Activation Code

See purchasing public instance activation codes. Refer to [Billing Overview](#).

Purchase Enterprise Instance

1. Log in to the [IoT Development Platform](#), enter the instance list page, and click **Add Instance** in the figure below, or go to the [Purchase Page of the IoT Development Platform](#).



2. Enter the enterprise instance purchase page in the figure below.

选择配置

实例类型

**企业实例**
适用开发工业、商业、政务等产业方案
✓ 实例包年包月计费
✓ 高设备消息上下行 TPS

**公共实例**
适用开发智能家居、智能穿戴等消费硬件
✓ 设备激活码数量计费
✓ 可接入小程序控制设备

实例单元规格

| | 对比项 | 入门版 小型项目 | 标准版 产业方案首选 | 高性能版 满足更高需求 |
|------|-------------|-------------|---------------|----------------|
| 价格 | 单元价格 | 3元/月 | 3元/月 | 8元/月 |
| 单元资源 | 同时在线设备数 | 500 台 | 10000 台 | 10000 台 |
| | 设备注册数上限 | 5000 台 | 100000 台 | 100000 台 |
| | 设备消息上下行 TPS | 100 条/秒 | 1000 条/秒 | 3000 条/秒 |
| | 每日赠送消息额度 | 100 万条 | 5000 万条 | 1.5 亿条 |

实例单元数

10 个

20 个

-

1

+

个

配置费用 3元

3元

立即购买



联系销售



- On the purchase page, select the resource type as "enterprise instance". For the number of devices, select appropriate device quantity specifications according to the total number of devices required to be registered on the platform for the actual project by the user. And choose appropriate duration. The system will automatically calculate the new price based on the number of devices, duration, etc.
- Click **Buy Now** button, and show the following confirmation page.

©2013–2025 Tencent Cloud. All rights reserved.

Page 9 of 334

腾讯云

Q

备案 | 20227728...

控制台

确认产品信息

返回修改配置

下单说明

请确认产品信息后提交订单，如有优惠券可在支付时选择使用，最终实付金额以支付订单时为准。

产品清单

▼ 预付费产品 (1)

应付合计 300.00元

| 产品名称 | 配置 | 类型 | 单价 | 数量 | 时长 | 订单金额 |
|------------|---------------------|----|-----------|----|-----|---------|
| 实时互动-物联版新购 | 实例版本: 入门版 单元数: 1 | 新购 | 350.00元/月 | x1 | 1个月 | 300.00元 |

选择优惠券

代金券 (1)

☐ 使用代金券抵扣 0.00元 兑换优惠券

您有 1 张代金券，本次有 1 张可用。特权用户最多可用 10 张代金券，其中满减券最多一张。

选择优惠券

实付金额 300.00元

去支付

联系销售

5. Click the **go to pay** button. After payment is successful, when the user returns to the instance list, the enterprise instance of the corresponding specification created for the user by the platform will be queried.

实例列表 什么是实例 ①

新增实例 全部实例 所有状态

公共实例 运行中 购买激活码

实例ID: ins-2dhy1WfAgfN
创建时间: 2023-05-11 14:23:25

已注册设备 25个 剩余可注册设备 7185个

企业实例入门版 即将到期 续费 升配

实例ID: i-
单元个数: 1
创建时间: 2024-03-21 10:41:27
到期时间: 2024-04-21 10:41:43

同时可在线设备数 500个 消息上下行 TPS 峰值 100条/秒

公共实例
智能家居、智能穿戴等消费类场景

快捷入口

设备SDK 设备迁移 产品共享

产品动态

更多

• 控制台改版
• 新增实时音视频增值服务

入门文档

• MQTT.fx 快速接入物联网开发平台
• 云端控制设备入门
• 平台转发消息至用户 HTTP 服务

View Enterprise Instance Information

1. Click the created enterprise instance to enter the Instance Information Page.

- Users can view basic information of the instance, instance status, instance remark information, instance creation time and instance expiration time.
- Users can also view the capacity limit of the number of online devices, the number of registered devices, the upstream and downstream TPS of messages, and the TPS of message forwarding during instance purchasing in the resource status.
- If the user's purchased instance has devices for message upstream and downstream communication and rule engine forwarding, line charts will be shown in "TPS of Message Forwarding in Recent Three Days" and "TPS of Message Upstream and Downstream in Recent Three Days".

实例信息

常用功能 帮助文档

企业实例 入门版

适用于交通出行、工业、能源、服务业等产业物联网应用，资源弹性高可靠，助力企业高效低成本实现物联网应用方案。

常用功能

隐藏常用功能

产品开发

创建产品将设备接入平台

设备管理

创建或管理现有设备

固件升级

远程批量升级设备固件

规则引擎

设备消息转发至用户业务系统

实例信息

实例ID

实例备注

过期时间

2024-04-21 10:41:43

每日赠送消息数

100 万条

实例状态

即将到期

创建时间

2024-03-21 10:41:27

实例单元数

1

产品总数

2 个

资源状态

升配 续费

同时在线设备数上限

500 个

注册设备数 ①

2 / 5000 个

消息上下行 TPS ①

0 / 100 条/秒

消息转发 TPS ①

0 / 150 条/秒

近三日消息转发 TPS ①

近三日消息上下行 TPS ①

Modify Instance Remark

- Click the modify icon of the instance remark to modify the remark name of the instance.

实例信息

常用功能

帮助文档

企业实例

适用于交通出行、工业、能源、服务业等产业物联网应用，资源弹性高可靠，助力企业高效低成本实现物联应用方案。

常用功能

隐藏常用功能

产品开发

设备管理

固件升级

规则引擎

创建产品将设备接入平台

创建或管理现有设备

远程批量升级设备固件

消息转发以二次开发数据

实例信息

实例ID

ins-2qQA9PEtr7G

实例备注

我的实例

过期时间

2024-02-23 21:04:35

实例状态

运行中

创建时间

2024-01-23 21:04:35

资源状态

升配

续费

实例设备数上限

2000

已注册设备数

1

产品总数

1

应用总数

1

2. In the pop-up modify window, input the instance remark name and click **Save** to modify the template successfully.

修改实例备注

×

实例备注

光储充方案

✓

不超过 20 字符

保存

取消

Migrating Instance Device Data

Last updated: 2025-04-27 17:28:39

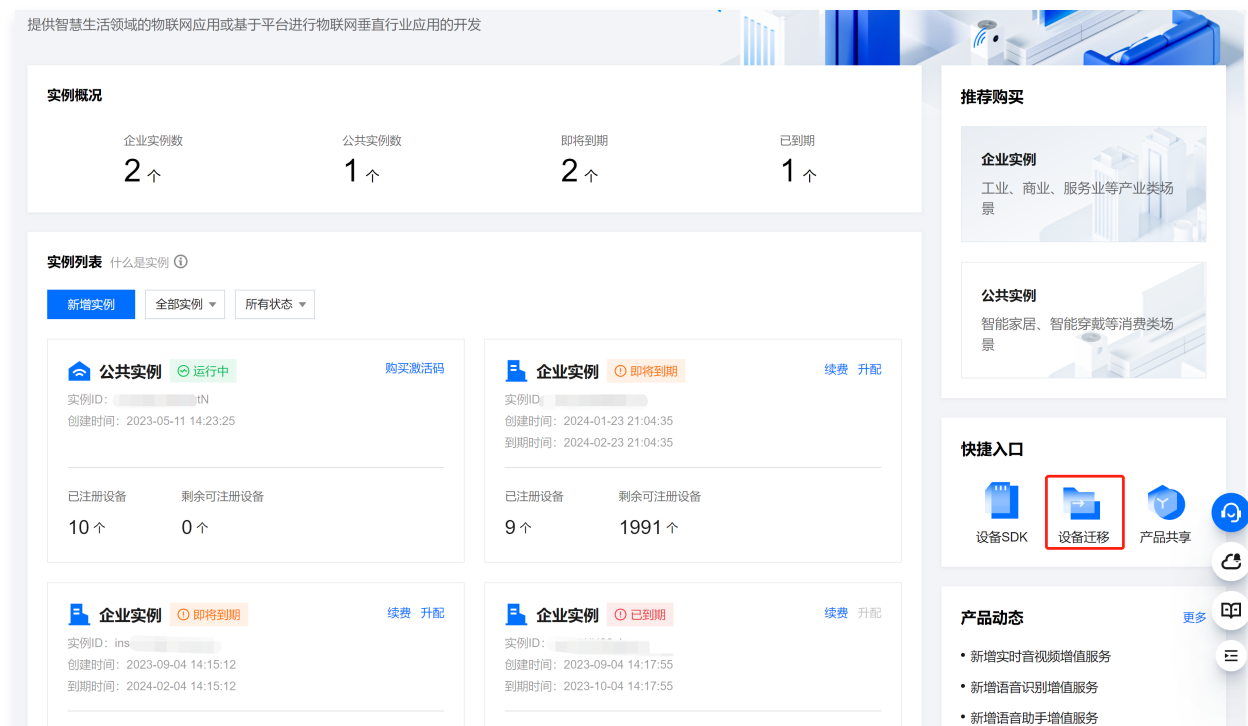
Overview

Instance device migration provides users with the feature of migrating products and device data under a specific instance to another instance. This feature can be applied when users need to migrate a product and its device data from the joint debugging stage of a public instance to another purchased enterprise instance after completion of joint debugging during instance development.

Operation Steps

Equipment Migration

1. Log in to the [IoT Explorer](#) and enter the instance list page.
2. Click "Device Migration" in the shortcut entry area on the right of the instance list.



3. Select the source instance and target instance for migration on the device migration interface, and select the products to be migrated under the source instance.

设备迁移

设备迁移

迁移记录

源实例 *

ins-2

...

N

源项目 *

t

...

目标实例 *

ins

...

G

目标项目 *

...

w

产品 ⓘ *

开关

确定

取消

4. Click **Confirm**. The system will perform the migration operation and generate migration history records.

设备迁移

设备迁移

迁移记录

| 批次号 | 产品名称 | 源实例/项目 | 目标实例/项目 | 状态 | 迁移人 | 迁移时间 | 操作 |
|-----|------|----------------------|--------------------|----|----------------|---------------------|--------------------|
| 314 | 开关 | ins-2 <div>...</div> | ins <div>...</div> | 成功 | <div>...</div> | 2024-01-29 11:24:37 | 删除 |

5. After successful migration, the migrated products and device data under the source instance will not be viewed. They must be viewed under the target instance.
6. The migrated product equipment will not affect the MQTT access of the device side.

Public Instance Device Integration

Product Management

Last updated: 2025-04-27 17:29:03

Before device connectivity, it is required to create a product first. A product is equivalent to the collection of a certain type of devices. Users manage all devices under it through the product.

Creating a Product

1. Log in to the [IoT Explorer](#) and select **public instance**.
2. Click the product development menu, and click **Create Product** to define your product.
3. Select your **product category**.

产品开发 / 新建产品

请选择产品品类

产品品类 *

标 已定义标准物模型 免 包含免开发面板

请输入品类

智慧农业

智慧生活

智能城市

智能制造

全屋智能

其他行业

传感器

光照度传感器

气体检测器

二氧化碳检测器

土壤传感器

酸碱度传感器

无 无

无 无

无 无

无 无

无 无

请选择产品所属的品类

填写产品信息

新建产品

取消

4. After selecting the product category, fill in the basic information of the product.

产品开发 / 新建产品

请选择产品品类

产品品类 *

标 已定义标准物模型 免 包含免开发面板

请输入品类



| | | | | |
|------|-----|---|---|---|
| 智慧农业 | 传感器 | <input checked="" type="radio"/> 光照度传感器 | 无 | 无 |
| 智慧生活 | | <input type="radio"/> 气体检测器 | 无 | 无 |
| 智能城市 | | <input type="radio"/> 二氧化碳检测器 | 无 | 无 |
| 智能制造 | | <input type="radio"/> 土壤传感器 | 无 | 无 |
| 全屋智能 | | <input type="radio"/> 酸碱度传感器 | 无 | 无 |
| 其他行业 | | | | |

已选择品类：智慧农业 / 传感器 / 光照度传感器

填写产品信息

产品名称 *

请输入产品名称

支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

设备类型

设备

网关

子设备

通信方式 *

请选择通信方式

请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

数据协议

物模型

自定义透传



描述

选填

最多不超过80个字符

新建产品

取消

The basic information settings of the product are as follows:

- Product name: A combination of Chinese characters, letters, digits, and underscores, 1 to 20 characters long and cannot be empty.
- Product Category: Select the category to which the product you create belongs. The data templates of attributes, events, etc. of different types of products will vary. For details, see [Data Template](#).
- Device type: Device type is divided into 3 categories: device, gateway, and subdevice. Details as follows:
 - Device: This type of device can directly access IoT Explorer and has no mounted subdevices.

- Gateway: This type of device can directly access IoT Explorer and accept subdevices to join the LAN. It can maintain the topology relationship of subdevices and synchronize the topology relationship with subdevices to the cloud.
- Subdevice: This type of device must rely on a gateway device to communicate with IoT Explorer. For example, Zigbee, Bluetooth, RF433 and other devices. For descriptions of gateways and subdevices, refer to the documentation [gateway subdevice](#).
- Modes of communication: you can choose Wi-Fi, mobile cellular (2G/3G/4G), 5G, BLE, LoRaWAN and other modes of communication.
- Access Gateway Protocol (selectable as subdevice when device type is selected): indicates the communication protocol type for the device under this product to act as a subdevice communicating with the gateway.
- Zigbee: Indicates that the communication protocol between the subdevice and the gateway is ZigBee.
- BLE: Indicates that the communication protocol between the subdevice and the gateway is BLE.
- 433: Indicates that the communication protocol between the subdevice and the gateway is 433.
- Custom: Indicates that the communication protocol between the subdevice and the gateway is another standard or a private protocol.
- Data protocol: Defaults to the data protocol of the Thing Model. You can also customize a protocol for pass-through.
- Description: The number of characters must not exceed 80. You can fill in this field as needed.

Deleting a Product

Note:

To prevent accidental deletion of the product from affecting your business, if there are still devices under the product, the product cannot be deleted.

1. After completing the creation of a new product, you can view the created product on the product list page.
2. When you no longer need the product, you can click **Delete** in the top right corner of the product and confirm.

| 产品名称 | 产品ID | 产品品类 | 设备类型 | 状态 ▼ | 创建时间 | 操作 |
|------|---|-----------------|------|------|---------------------|--------------------|
| 测试产品 |  | 智慧农业-传感器-光照度传感器 | 设备 | 开发中 | 2024-01-22 19:10:48 | 删除 |

确定删除该产品?

删除产品后, 该产品所属设备等数据都将删除且不能恢复, 请确认要删除该产品吗?

[删除](#)[取消](#)

Modifying a Product

1. Click the product name to enter the product detail page.



2. Click **More Information** to view product details. Click **Modify Product Information** to perform product information modification.

修改产品信息

×

产品名称 *

测试产品

支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

产品品类

智慧农业 / 传感器 / 光照度传感器

设备类型

设备

认证方式

密钥认证

通信方式

Wi-Fi

数据协议

物模型

绑定策略 *

抢占绑定 ▼

描述

选填

最多不超过80个字符

保存

取消

Querying Products

1. Click the product name to enter the product detail page.

产品开发 / 测试产品

回到旧版 帮助文档

测试产品 开发中

更多信息

产品 ID

产品密钥

设备类型

设备

设备数量 0

动态注册

1 物模型定义

2 设备开发

3 交互开发

4 设备调试

5 批量投产

导入物模型 查看物模型JSON

物模型定义帮助

标准功能 (0)

自定义功能 (0)

高级功能 (0)

添加标准功能

请输入功能名称或标识符搜索

| 功能类型 | 功能名称 | 标识符 | 数据类型 | 读写类型 | 数据定义 | 操作 |
|--------|------|-----|------|------|------|----|
| 当前列表为空 | | | | | | |

2. Click More Information to view product details.

产品信息

×

产品名称

测试产品

产品 ID

5 

产品品类

智慧农业 / 传感器 / 光照度传感器

设备类型

设备

设备数量

认证方式

密钥认证

通信方式

Wi-Fi

数据协议

物模型

绑定策略

抢占绑定

创建时间

2024-01-22 19:10:48

更改时间

2024-01-22 19:10:48

产品描述

-

修改产品信息

Product Development

Thing Model Definition

Last updated: 2025-04-27 17:29:29

Thing Model Overview

A Thing Model is a digital model defined by Tencent Cloud IoT Developer Platform for the same type of devices, that is, products. It digitally describes products through three feature types: properties, events, and actions.

| Feature Type | Feature Description |
|--------------|---|
| Attribute | Various parameters and status data of the equipment during operation. For example, the temperature and humidity values collected by the Temperature and Humidity Sensor; the switch status values of electrical equipment, the current speed and latitude and longitude values collected by vehicle devices. Attributes support two types: read-write and read-only. Read-write means that the attribute can be reported from the device to the platform and can also be controlled from the platform. Read-only means that the attribute is only reported from the device to the platform. |
| Event | Data initiated by the device during runtime that requires the User Business System's awareness, including three event types: information, alarm, and failure. For example, when a device malfunctions, it is required to send the fault information and related data at the time of failure to the platform. |
| Behavior | <p>The calling method provided by the device for applications, used by the business to initiate requests. It requires the device side to process and then return the processing result. Scenarios where the business system can synchronously or asynchronously obtain the result. For example:</p> <ul style="list-style-type: none">• Parking lot users need to be allowed to pass immediately after paying the parking fee. The business system needs to obtain whether the final status synchronization operation of the barrier is successful.• After a cloud printing scenario user issues a printing action, the input parameters of the action include order ID, print document content, etc., and the output parameters include the encoding of the print result. |

Thing Model Category

The features of the product include standard features, custom functions, and advanced functions.

| Feature Category | Feature Description |
|------------------|---|
| Standard feature | Standard features refer to the Thing Models of some common device categories provided by Tencent Cloud IoT Platform. Users can choose to use or not use these features as needed. |
| Custom function | Custom function refers to the feature provided by Tencent Cloud IoT Platform that allows users to freely define the Thing Model of products. Users can freely create, |

| | |
|-------------------|--|
| | delete and edit Thing Model features according to the specifications and characteristics of the equipment. |
| Advanced Features | Advanced features refer to the value-added service features provided by Tencent Cloud IoT for users. For example, the tencent real-time communication (TRTC) service can be applied to visual intercom, Cloud Broadcasting real-time shouting, one-to-many emergency call scenarios; advanced features generally automatically generate corresponding standard features. |

Operation Steps

Add Standard Features to Thing Model

The standard features are the default Thing Model definitions of preset device categories provided by Tencent Cloud IoT, as well as the default Thing Models corresponding to advanced features.

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **public instance**.
2. Click on the enterprise instance, enter the instance page, click **Product Development** in the left menu, enter the product list, and select a certain "product" to enter **Thing Model Definition**.
3. Click **Add Standard Function** on the **Standard Function** tab.

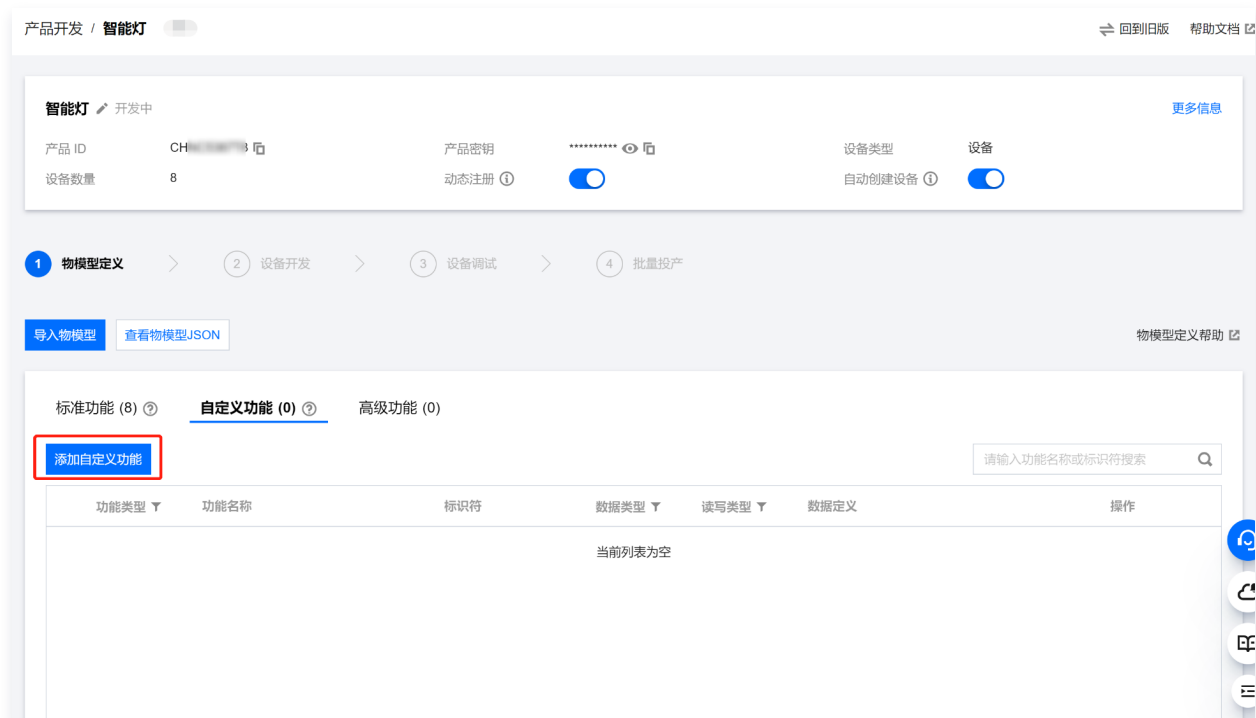
4. The system-defined Thing Models of the same category as the user-selected category for other products can be added as needed by users. Once confirmed, the selected Thing Models will be added to the "custom function".

Custom Addition of Thing Model Attributes

Custom functions allow users to freely define the Thing Model according to device specifications, assigning users adequate freedom to define.

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **public instance**.

- Click the public instance to enter its details page. Click **Product Development** in the left menu to enter the product list, and choose a certain "product" to enter the **Thing Model Definition**.
- Click the **Add Custom Function** button on the **Custom Function** tab page.



- Select the "feature type" as needed in the popup window. For example, if a lighting product has an attribute switch, enter "Switch" as the feature name, and the identifier must be unique among all Thing Models of this product. Select the data type and then save.

新增自定义功能

×

⚠ 注意：添加自定义功能将影响设备通过语音、中控面板控制，建议添加标准功能，若已有标准功能无法满足，您可提交 [意向单](#) 申请新增标准功能。

功能类型

属性

事件

行为

功能名称 *

开关

支持中文、英文、数字、下划线的组合，最多不超过20个字符

标识符 *

power_switch01

第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符

数据类型

布尔型

整数型

字符串

浮点型

枚举整型

枚举字符串

时间型

结构体

数组

读写类型

读写

只读

ⓘ

数据定义

0 关

1 开

支持中文、英文、数字、下划线的组合，最多不超过12个字符

描述

必填

最多不超过80个字符

保存

取消

Custom Addition of Thing Model Events

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **public instance**.
2. Click the public instance to enter its details page. Click **Product Development** in the left menu to enter the product list, and choose a certain "product" to enter the **Thing Model Definition**.
3. Click the **Add Custom Function** button on the **Custom Function** tab page.
4. Select **event** as needed in the popup window. Then select "event type" again as "info". Users can freely define the event information that devices send to the IoT platform.

新增自定义功能

注意：添加自定义功能将影响设备通过语音、中控面板控制，建议添加标准功能，若已有标准功能无法满足，您可提交[意向单](#)申请新增标准功能。

功能类型

属性 事件 行为

功能名称 *

Device_Status

支持中文、英文、数字、下划线的组合，最多不超过20个字符

标识符 *

status_report

第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符

事件类型

告警 故障 信息

事件参数

| 参数名称 | 参数标识符 | 数据类型 | 数据定义 | 操作 |
|---------------|---------|------|--|----|
| running_state | status | 布尔型 | <div>0 normal 1 fail</div> <div>支持中文、英文、数字、下划线的组合，最多不超过12个字符</div> | 删除 |
| Message | message | 字符串 | <div>- 64 + 字节</div> <div>请输入字符串长度限制，最大长度不超过2048个</div> | 删除 |
| 添加参数 | | | | |

描述

选填

最多不超过80个字符

保存

取消

Custom Addition of Thing Model Actions

1. Log in to the [IoT Development Platform](#), enter the instance list page, and select the generated **public instance**.
2. Click the public instance to enter its details page. Click **Product Development** in the left menu to enter the product list, and choose a certain "product" to enter the **Thing Model Definition**.
3. Click the **Add Custom Function** button on the **Custom Function** tab page.
4. Select **action** as needed in the popup window.

In self-service retail, shared payment or parking scenarios, it is often necessary for the business system to immediately perform operations such as opening doors, unlocking or opening barriers through the IoT platform after payment. This requires the business system to receive the processing results from the device end in real time. If it fails to respond promptly, the business system needs to cancel the transaction, such as initiating a refund.

5. Enter the corresponding action name and parameters for the feature name. The request parameters need to be defined by the user, and the response parameters are the parameters reported to the IoT platform after execution on the device side.

For example, the feature name is "Unlock", and the identifier is "Open". The request parameter customizes a string type of order ID, so that the business system can track the processing situation of each order on the device side; the response parameters can also add fields of order ID and the final unlock status result code of the device. If the unlock succeeds, return code 0, otherwise return 1. When the business system receives code 1, it can either retry if permitted by the business, or initiate a refund operation after multiple failures of retries when the payment has been completed.

ⓘ 注意：添加自定义功能将影响设备通过语音、中控面板控制，建议添加标准功能，若已有标准功能无法满足，您可提交 [意向单](#) 申请新增标准功能。

功能类型

属性 事件 **行为**

功能名称 *

开锁

支持中文、英文、数字、下划线的组合，最多不超过20个字符

标识符 *

Open

第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符

调用参数

| 参数名称 | 参数标识符 | 数据类型 | 数据定义 | 操作 |
|------|----------|------|--|----|
| 订单ID | Order_id | 字符串 | <div>2048 字节</div> 请输入字符串长度限制，最大长度不超过2048个 | 删除 |
| 添加参数 | | | | |

返回参数

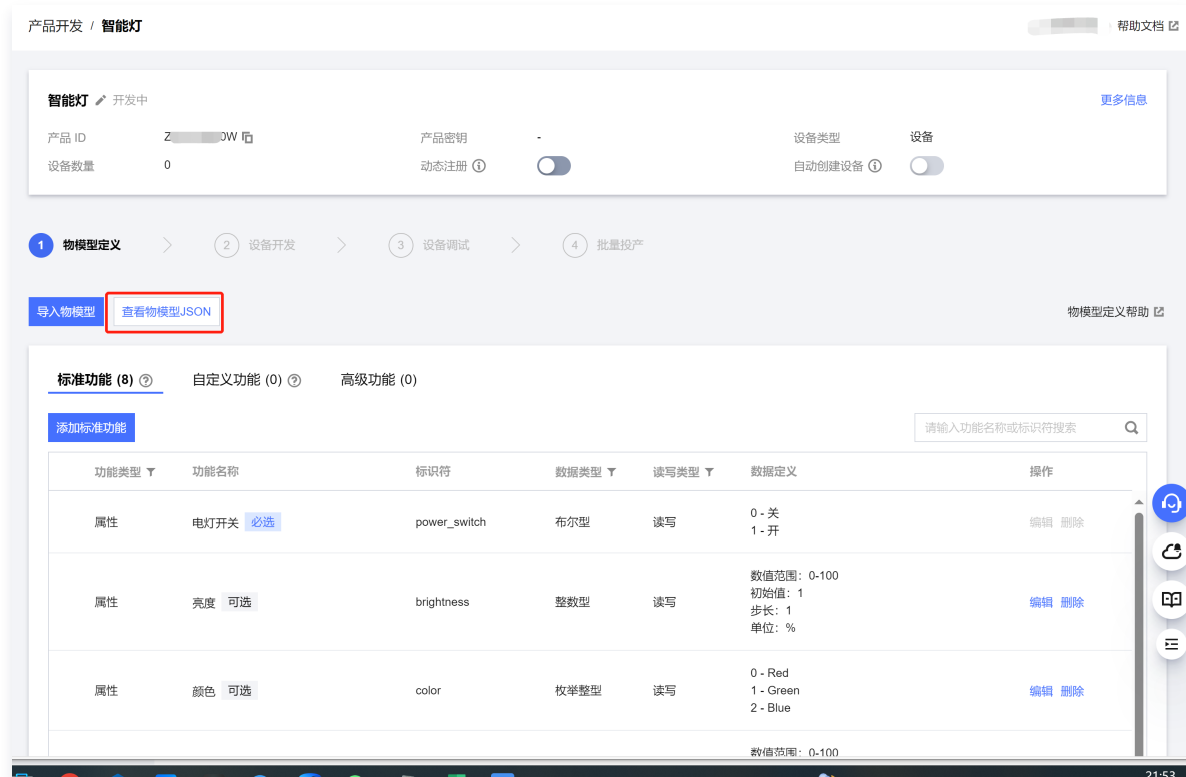
| 参数名称 | 参数标识符 | 数据类型 | 数据定义 | 操作 |
|------|----------|------|--|----|
| 订单ID | Order_id | 字符串 | <div>2048 字节</div> 请输入字符串长度限制，最大长度不超过2048个 | 删除 |
| 状态 | code | 整数型 | <div>数值范围</div> <div>0 100</div> <div>初始值</div> <div>0</div> | 删除 |

Add Advanced Function Thing Model

Refer to the [Advanced Features](#) guide, purchase and activate the corresponding advanced features. Once enabled, the platform will add by default the Thing Model corresponding to the advanced features.

View Thing Model JSON Object

- After adding standard features and custom functions, users can click **View Thing Model JSON** to view the JSON format corresponding to the Thing Model.



2. As shown below, users can copy or download the JSON Thing Model. The copied JSON can be used for copying the Thing Model definition between products through the "Import Thing Model" feature.



Import Thing Model

1. After copying the Thing Model JSON of a specific product, click **Import Thing Model** and paste the copied JSON into the text box.
2. Then click **Import**, which will override the original product's Thing Model. Note that the Thing Model import feature should be used with caution for products in mass production.

导入物模型

注意：导入新的JSON后原产品的物模型将会被覆盖

您可以通过 JSON 对产品的物模型进行定义后导入平台，格式规范请 [查看文档](#)

请将要导入的物模型对应的JSON粘贴到此文本框

导入

取消

Thing Model Format Reference

The following is the JSON description of the Thing Model for the smart light, including various data types and event types. An example JSON is as follows:

```
{
  "version": "1.0",
  "profile": {
    "ProductId": "2300UMK31M",
    "CategoryId": "3"
  },
  "properties": [
    {
      "id": "power_switch",
      "name": "light switch"
      "desc": "Control the on and off of the light"
      "required": true,
      "mode": "rw",
      "define": {
        "type": "bool",
        "mapping": {
          "0": "Off"
          "1": "On"
        }
      }
    }
  ]
}
```

```
},
{
  "id": "color",
  "name": "color"
  "desc": "Light color"
  "mode": "rw",
  "define": {
    "type": "enum",
    "mapping": {
      "0": "Red",
      "1": "Green",
      "2": "Blue"
    }
  }
},
{
  "id": "brightness",
  "name": "brightness"
  "desc": "Light brightness"
  "mode": "rw",
  "define": {
    "type": "int",
    "unit": "%",
    "step": "1",
    "min": "0",
    "max": "100",
    "start": "1"
  }
},
{
  "id": "name",
  "name": "light position name"
  "desc": "Light position name: study, living room, etc."
  "mode": "rw",
  "required": false,
  "define": {
    "type": "string",
    "min": "0",
    "max": "64"
  }
}
],
"events": [
  {
    "id": "status_report",
    "name": "DeviceStatus",
    "desc": "Report the device status",
    "type": "info",
    "required": false,
    "params": [
```

```
{
  "id": "status",
  "name": "running_state",
  "desc": "Report current device running state",
  "define": {
    "type": "bool",
    "mapping": {
      "0": "normal",
      "1": "fault"
    }
  }
},
{
  "id": "message",
  "name": "Message",
  "desc": "Some extra message",
  "define": {
    "type": "string",
    "min": "0",
    "max": "64"
  }
}
],
{
  "id": "low_voltage",
  "name": "LowVoltage",
  "desc": "Alert for device voltage is low",
  "type": "alert",
  "required": false,
  "params": [
    {
      "id": "voltage",
      "name": "Voltage",
      "desc": "Current voltage",
      "define": {
        "type": "float",
        "unit": "V",
        "step": "1",
        "min": "0.0",
        "max": "24.0",
        "start": "1"
      }
    }
  ]
},
{
  "id": "hardware_fault",
  "name": "Hardware_fault",
  "desc": "Report hardware fault",
```

```
"type": "fault",
"required": false,
"params": [
  {
    "id": "name",
    "name": "Name",
    "desc": "Name like: memory,tf card, sensors ...",
    "define": {
      "type": "string",
      "min": "0",
      "max": "64"
    }
  },
  {
    "id": "error_code",
    "name": "Error_Code",
    "desc": "Error code for fault",
    "define": {
      "type": "int",
      "unit": "",
      "step": "1",
      "min": "0",
      "max": "2000",
      "start": "1"
    }
  }
]
},
"actions": [
  {
    "id": "unlock",
    "name": "Behavior detection for turning on the light"
    Action for description of turning on the light
    "input": [
      {
        "id": "open",
        "name": "Switch"
        "define": {
          "type": "bool",
          "mapping": {
            "0": "Off"
            "1": "On"
          }
        }
      }
    ],
    {
      "id": "user",
      "name": "User",
      "define": {
```

```
        "type": "string",
        "min": "0",
        "max": "2048"
      }
    },
    ],
    "output": [
      {
        "id": "user",
        "name": "user"
        "define": {
          "type": "string",
          "min": "0",
          "max": "2048"
        }
      },
      {
        "id": "time",
        "name": "Light on time"
        "define": {
          "type": "timestamp"
        }
      },
    ],
    {
      "id": "state",
      "name": "light status"
      "define": {
        "type": "bool",
        "mapping": {
          "0": "Off"
          "1": "On"
        }
      }
    }
  ],
  "required": false
}
]
```

References

- Learn about Thing Model Protocol. See [Thing Model Protocol](#).
- Viewing device attributes, events, and actions can be found in [Device Debugging](#) content.

Device Development

Last updated: 2025-04-27 17:29:44

Overview

After the user defines the product and Thing Model, the device needs to be connected to the platform as required by the access protocol. This document primarily introduces how to use the development platform for device development.

Operation Steps

Device Development

After the user creates a product and defines its data template, click **Device Development**. Currently, the development platform provides three development methods.

- **Develop based on modules:** Scenarios where MCU meets serial port communication and communicates with the cloud through a communication module.
- **Develop based on SDK:** Access scenarios that meet the direct integration of the C SDK.
- **Develop based on OS:** Access scenarios that meet the integration of C SDK based on the Internet of Things operating system.



Module-Based Development

1. If your device needs to connect to the development platform through a communication module, click **Develop Based on Module**.
2. The system displays the module selection window. You need to choose appropriate communication modules based on your business requirements. This includes the module brand and module

communication type. After choosing appropriate modules, you can click **Confirm**.



3. After selecting a module, you can click **Select Again** to replace the module. You can also click **View Detail** to learn about the detailed parameters of the module. Additionally, you can click **Procurement Consultation** to purchase from the module company.



4. Embedded development.

- For devices connected via modules, if a data template is defined, the platform provides the functionality of automatically generating MCU SDK code, which is used to speed up how the MCU connects to the communication module.
- Click **MCU SDK code**. The development platform will generate a compressed file. After you download it, you can follow the development guide to connect the device to the development platform.

- How to perform MCU development based on the downloaded MCU SDK code, for details, see [device development guide](#) related documentation.

✓ 物模型 > 2 设备开发 > 3 交互开发 > 4 设备调试 > 5 批量投产

设备开发 Topic列表

嵌入式开发



数据模板配置文件生成

[数据模板代码生成指南](#)

[数据模板应用开发指南](#)



OS代码下载

[TencentOS tiny 代码下载](#)

[RT-Thread 代码下载](#)



开发指引

[基于 TencentOS tiny 开发指南](#)

[基于 RT-Thread 开发指南](#)

[基于 Linux 开发指南](#)

[基于 Windows 开发指南](#)

[基于 FreeRTOS 开发指南](#)

[基于其他 OS 开发指南](#)

上一步

下一步

Interactive Development

Last updated: 2025-04-27 17:29:59

Overview

Users can interact with devices through the official Tencent Lianlian Mini Program or Chinese domestic brand mini programs. The Internet of Things development platform provides interactive development configuration services, simplifying the development difficulty of mini programs. You can achieve data communication between the mini program and the platform through simple configuration and quickly possess the ability of the mobile application end.

Prerequisites

Completed the work of the [device development](#) stage.

Operation Steps

Control Product Methods

The IoT development platform provides mini programs and apps in two forms of application sides, and supports the configuration of interaction methods for three application types: official public mini programs, Chinese domestic brand mini programs, and Chinese domestic brand apps. After entering the interaction development page, based on your business needs, you can choose to use the above application types for interaction development configuration through the switch button.

Note:

If your products need to be connected to Tencent Lianlian official mini program, please enable "Connect to Tencent Lianlian official mini program". The Tencent Lianlian platform will review and authenticate the connection.

- Use the official public mini program of Tencent Lianlian to control the product. Then, turn on **Integration with Tencent Lianlian Official Mini Program** and configure accordingly.

物模型定义

设备开发

交互开发

设备调试

批量投产

① 如果您的产品需要接入腾讯连连官方小程序，请开启“接入腾讯连连官方小程序”，接入腾讯连连平台会在批量投产时对交互开发面板配置、产品功能进行审核认证。

接入腾讯连连官方小程序

您选择使用平台的官方小程序控制产品，您可在下方配置产品在腾讯连连小程序中的展示效果。您可以通过 [腾讯连连小程序二维码](#) 前往小程序

产品展示配置

您可以自定义产品在腾讯连连首页-设备列表中展示的产品图标和默认名称

配置

快捷入口配置

您可以自定义产品在设备列表-快捷功能区区域显示的快捷操作

配置

面板配置

您可以自定义产品控制面板的风格、布局、按钮样式等配置

配置

配网引导

您可以自定义产品的配网图文，使产品配网流程引导更明了、有效、符合产品的实际情况

配置

扫一扫产品介绍

您可以自定义用户在使用微信扫一扫添加设备时的产品介绍页

配置

智能联动配置

您可以自定义用户在添加智能时，该产品可作为条件或任务的功能属性

配置

上一步

下一步

- Chinese domestic brand mini program: If you use the Mini Program SDK of IoT Explorer to develop a Chinese domestic brand mini program, turn off **Integration with Tencent Lianlian Official Mini Program** and configure accordingly.

物模型定义

设备开发

交互开发

设备调试

批量投产

① 如果您的产品需要接入腾讯连连官方小程序，请开启“接入腾讯连连官方小程序”，接入腾讯连连平台会在批量投产时对交互开发面板配置、产品功能进行审核认证。

接入腾讯连连官方小程序

您可以使用平台提供的SDK开发自有品牌的小程序，您可在下方自定义小程序内产品图标等信息 [了解自有品牌小程序开发](#)

产品展示配置

您可以自定义产品在腾讯连连首页-设备列表中展示的产品图标和默认名称

配置

智能联动配置

您可以自定义用户在添加智能时，该产品可作为条件或任务的功能属性

配置

上一步

下一步

- Chinese domestic brand App: If you use the open source edition App and App SDK to develop a Chinese domestic brand App, turn off **Integration with Tencent Lianlian Official Mini Program** and configure

accordingly.

✓ 物模型定义

>

✓ 设备开发

>

3 交互开发

>

4 设备调试

>

5 批量投产

① 如果您的产品需要接入腾讯连连官方小程序，请开启“接入腾讯连连官方小程序”，接入腾讯连连平台会在批量投产时对交互开发面板配置、产品功能进行审核认证。

接入腾讯连连官方小程序

ⓧ

您可以使用平台提供的SDK开发自有品牌的小程序，您可在下方自定义小程序内产品图标等信息[了解自有品牌小程序开发](#)

产品展示配置

①

配置

您可以自定义产品在腾讯连连首页-设备列表中展示的产品图标和默认名称

智能联动配置

①

配置

您可以自定义用户在添加智能时，该产品可作为条件或任务的功能属性

上一步

下一步

Device Debugging

Last updated: 2025-04-27 17:30:17

Overview

After device development is completed, it is required to enter the device debugging stage to debug whether the communication between the device and the cloud is normal. Device debugging provides online debugging of real devices and virtual device debugging, and allows querying of the current data reported by the device, historical communication logs, events, and online and offline records via the console. This document primarily introduces how to perform device debugging.

Prerequisites

Completed the work of [device development](#) and [interactive development](#) stages.

Operation Steps

Create New Device

1. Enter the public instance in [IoT Explorer](#), select the target product and then click **Device Debugging** to enter the device debugging process.
2. Click **Create New Device**, enter the device name as shown in the figure below, and click **Save** to create a device.



新建设备

所属产品 智能灯

设备名称 * dev001

支持英文、数字、下划线的组合，最多不超过48个字符

保存 取消

3. After successful creation, you can view the newly created device on the "Device Debugging" list page. The default status of the first-time successfully created device is "inactive".
4. Click **QR code** in the list to quickly bind a real device, which helps developers reduce development difficulty. After mass production, for security reasons, the QR code entry will be closed.



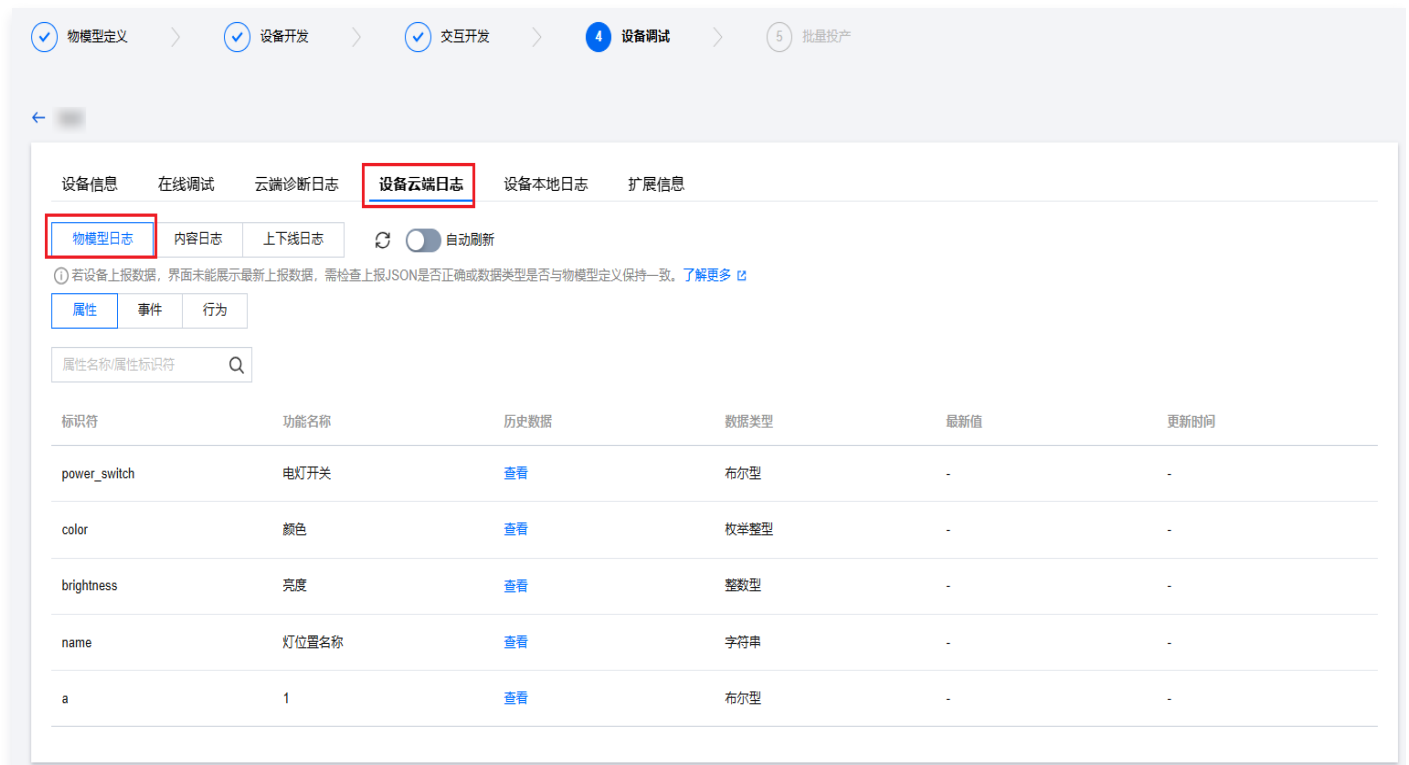
View Device Information

1. Upon successful creation of the device, it is required to query device information, obtain important parameters, and perform device debugging.
2. Click the device name in the device list. By default, enter the **equipment information** page to view the basic device information.
 - Device name: The unique device ID under the product, which generally needs to be burned into the device.
 - Product ID: The product ID associated with the device, which generally needs to be burned into the device.
 - Associated product: The name of the product to which the device belongs.
 - Device key: A randomly generated key for each device by the platform. To authenticate using the key, this information needs to be burned into the device.
 - Device creation time: The time when the device was initially created successfully.
 - Last online time: The time when the device last connected to the connection platform.
 - Activation time: The time when the device successfully connects to the platform for the first time.
 - Device status: The current device status. If the device successfully connects to the platform via MQTT, it displays "online". If the device is offline, it displays "offline". If the device has never connected to the platform, it displays "inactive".



View Thing Model Logs

1. When the device is successfully connected to the platform and publishes messages to the Thing Model Topic, you can view the attributes, events, actions and other data reported by the device in the "Thing Model Log" under "Cloud Log of Equipment".



2. List all the attribute functions of the Thing Model of the device in the list.

- Identifier: The identifier in the Thing Model corresponding to the device.
- Feature name: Correspond to the "feature name" in the Thing Model definition.
- Historical data: Click to view and retrieve the reported data of this feature. Display the historical data reported to the cloud by this feature in chronological order and verify whether the reported data is correct.
- Data type: The "data type" in the functionality defined by the Thing Model.
- Latest value: When a device reports data to the cloud, as long as the latest reported value of a specific feature changes, the latest value column will immediately display the latest value reported by the device.
- Update time: Refers to the change time of the latest value. Generally, it is the occurrence time when the device reports this feature.

! Note:

When a device reports data in the Thing Model Protocol format, but the latest value cannot be viewed in the Thing Model log, you need to confirm whether the format of the reported data is correct. For details, see [Thing Model Protocol](#) and [Common Issues with Thing Model](#).

View Content Logs

1. Content log provides users with the feature to query device uplink and downlink content logs by Topic. Users choose "Content log", and the "Log type" drop-down list will be shown.

设备信息 在线调试 云端诊断日志 **设备云端日志** 设备本地日志 扩展信息

物模型日志 **内容日志** 上下线日志 ☐ 自动刷新

① 日志类型选择"属性", Topic下拉框自动加载属性对应的Topic, 并查询出所选日期范围设备与平台的所有上下行属性内容日志。可以按需选择不同的类型, 即可查询不同的Topic所对应的上下行内容数据用于设备调试与问题定位。 [了解更多](#)

日志类型 topic

近30分钟 近1小时 今天 昨天 近3天 2024-12-26 15:54 ~ 2024-12-26 16:24

| 时间 | 通讯类型 | Topic | 通信内容 |
|--------|------|-------|------|
| 当前列表为空 | | | |

2. Select "Attribute" as the log type. The Topic dropdown list will automatically load the Topics corresponding to the attributes and query all uplink and downlink attribute content logs of the device and the platform within the selected date range. Users can select different types as needed, and then retrieve the uplink and downlink content data corresponding to different Topics for device debugging and problem localization.

设备信息 在线调试 云端诊断日志 **设备云端日志** 设备本地日志 扩展信息

物模型日志 **内容日志** 上下线日志 ☐ 自动刷新

日志类型 **属性** topic

近30分钟 近1小时 今天 昨天 近3天 近7天 近30天 2024-01-26 16:58 ~ 2024-01-26 17:28

| 时间 | 通讯类型 | Topic | 通信内容 |
|-------------------------|------|-------------------------------------|---|
| 2024-01-26 16:58:42.511 | 下行 | \$thing/down/property/R...32/dev001 | {"method":"report_reply","clientToken":"F...2","code":0,"status":"success"} |
| 2024-01-26 16:58:42.401 | 上行 | \$thing/up/property/F...2/dev001 | {"method":"report","clientToken":"R...2","params":{"cid":2,"lac":2,"Ua":0.020000}} |
| 2024-01-26 16:58:39.280 | 下行 | \$thing/down/property/R...2/dev001 | {"method":"report_reply","clientToken":"I...-1","code":0,"status":"success"} |
| 2024-01-26 16:58:39.177 | 上行 | \$thing/up/property/F...2/dev001 | {"method":"report","clientToken":"f...-1","params":{"cid":1,"lac":1}} |
| 2024-01-26 16:58:38.463 | 下行 | \$thing/down/property/R...B2/dev001 | {"method":"get_status_reply","clientToken":"R...0","code":0,"status":"success","data":{"reported":{"Uab":0.1,"Uc":0.1,"lc":0.1,"Ua":0.1,"Ubc":0.1,"Uca":0.1,"la":0.1,"lac":10,"Ub":0.1,"lb":0.1,"cid":10}}} |
| 2024-01-26 16:58:38.358 | 上行 | \$thing/up/property...B2/dev001 | {"method":"get_status","clientToken":"F...-0"} |

View Online and Offline Logs

1. Online and offline logs provide users with the ability to query logs of devices connecting to the platform (going online) as well as devices actively or passively disconnecting from the platform.



2. Device disconnect means the device actively closes the connection with the platform. Device keepalive timeout means the device does not send heartbeat, causing the connection timeout to close.

View Cloud Diagnostic Logs

The cloud diagnostic log feature is used to view the end-to-end trajectory logs of the interaction between devices and the cloud, helping users quickly diagnose exceptional errors that occur with devices during the debugging process, such as no permission to subscribe to a topic, failure to publish upstream messages, failure of rule engine forwarding to third-party services, and other error and exception event location. This document is used to locate the cause of communication content between devices and cloud messages and seek a solution.

Online Debugging

1. After your real device has been successfully connected to the development platform, you can use online debugging to test the data transmission and reception of the real device.
2. Click **Online Debugging** to enter the online debugging functionality. The premise is that the real device has been enabled and successfully connected to the development platform.
3. The control panel on the left side of Online Debugging is automatically generated according to the Thing Model of the product to which the device belongs. After setting the data to be sent, click **Send**. The system will automatically trigger control instructions to the device.
4. After the device receives the command, it will immediately return data to the cloud and display it in the textbox on the right.
5. If users want to issue control commands to debug with real devices through API, they can refer to [Remote Device Control](#) and [Device Pass-through Command Control](#) APIs respectively. One is to issue control commands through the Thing Model Protocol, and the other is to issue commands in the custom Payload mode.

Batch Production

Last updated: 2025-04-27 17:30:33

Overview

Device debugging is completed and after necessary tests, it can enter the production stage. The development platform will set the data template of the product after application release to a non-modifiable status to prevent problems occurring in the actual device running due to modifications made on the development platform. This document primarily introduces how to perform batch production.

Prerequisites

After the device debugging is completed and after testing of a small batch of devices, users can enter the batch production stage.

Operation Steps

1. Log in to the [IoT Explorer console](#), click to enter **Product Development** column, and view the product list.
2. Click the corresponding product name to enter the product detail page, select **batch production**, and enter the batch production phase.
3. After confirming the information is correct, check to agree in the options box. Upload the test report, click **confirm the product and apply for release**, and enter the application release phase.

物模型定义

设备开发

交互开发

设备调试

5 批量投产

产品确认及发布

您的产品选择接入腾讯连连官方小程序，需通过腾讯连连认证流程，待审批通过后再进行量产。如无需接入，可在【交互开发】关闭接入直接发布产品。

产品名称

测试产品

状态

开发中

产品品类

智慧生活 / 电工照明 / 灯

量产二维码



下载二维码 量产二维码文档

接入腾讯连连官方小程序

已启用 [去配置](#) 您选择接入腾讯连连，请下载《[测试报告模板](#)》并参考《[腾讯连连认证厂商测试规范V1.0](#)》编写测试报告提交并邮寄样品审核。

测试报告

上传

请按照上方《[测试报告模板](#)》编写的测试报告，压缩后上传（支持rar、zip、gz，不超过10M）

☐ 我确认该产品将接入腾讯连连官方小程序 ①

☐ 我了解并同意《[授权协议](#)》与《[开发者须知](#)》

确认产品并申请发布

批量投产

您的产品选择接入了连连小程序，需要先完成发布后才可投产

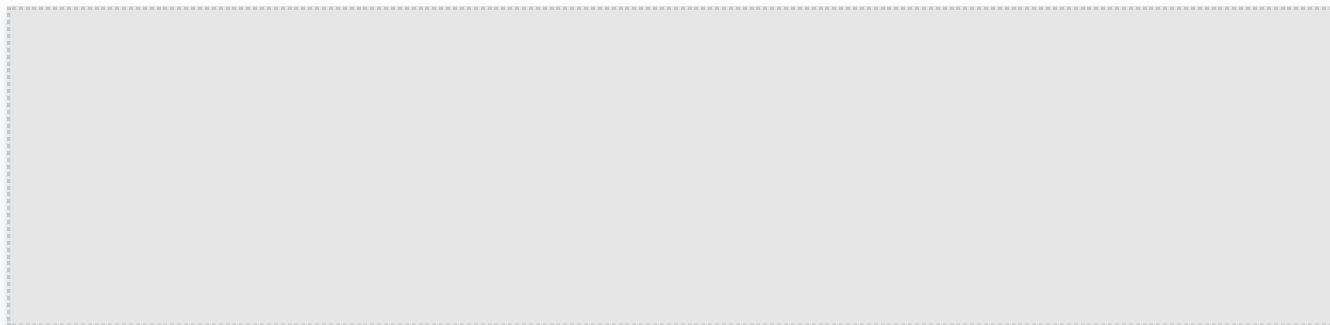
量产管理

上一步

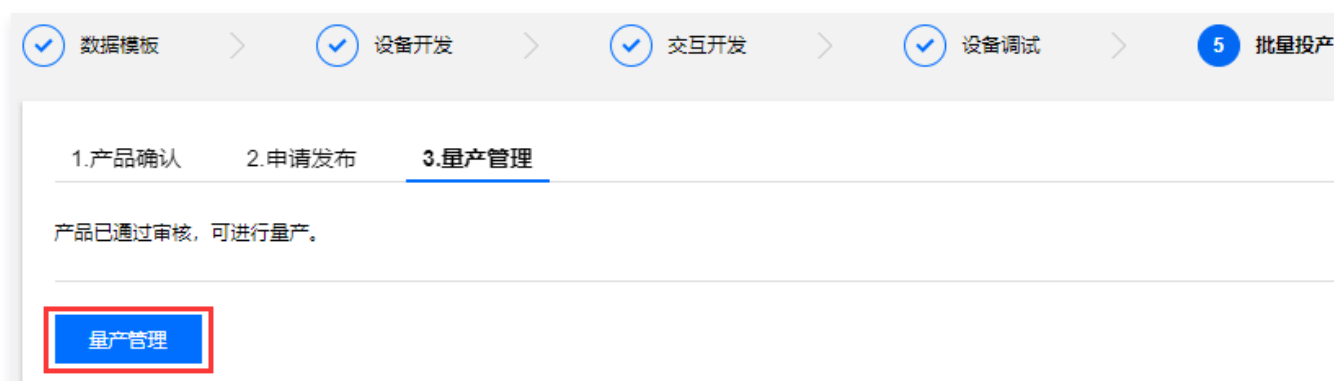
Note:

After the application release, the data template of the product will not be allowed to be modified. If your product needs to add new features due to business needs after obtaining feedback in the market, you can create a new product to solve it.

4. After the application for release is submitted, the product status will be changed to "under review". (If the review fails, it will return to the "under development" status, and you can reapply for release.)



5. If your product has been developed and passed the review, its status will become "released", and you can perform mass production in mass production management.



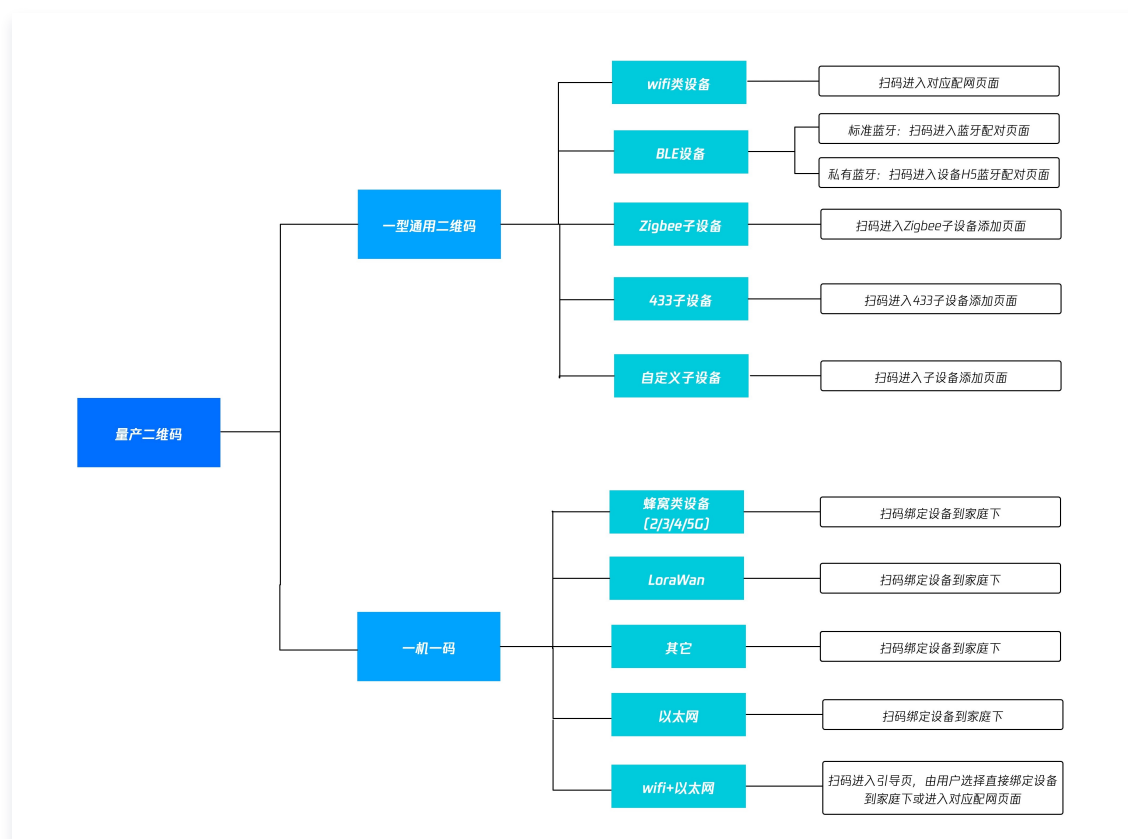
Equipment Mass Production Mass Production Management

Last updated: 2025-04-27 17:31:26

Introduction to Mass Production QR Codes

After the equipment completes development based on the IoT Explorer, it will soon enter the mass production stage. For device manufacturers, whether using the Tencent Lianlian Mini Program, a Chinese domestic brand mini program, or an App, devices can be added by scanning the QR code, thus improving user experience. Therefore, the platform provides mass production QR code solutions for various communication methods and categories. Based on the device type, you can obtain suitable QR codes for batch production.

- **Type I Universal QR Code:** It indicates that the same QR code can be printed on the packaging box or device for all devices within this product category. For example, for a smart light configured via Wi-Fi SmartConfig, the same QR code can be used for all devices of this model to scan and add via the configuration network.
- **One device, one code:** It means that for the devices under this product, during mass production, a QR code corresponding to each device needs to be generated before the device can be bound to a family by scanning the code.



Prerequisites

- The product has completed the preceding development steps and entered the batch production stage.
- The actual production environment of the equipment already meets the qualification requirements for mass production.

A Common Type I QR Code

Only the Tencent Lianlian Mini Program supports the "Common Type I QR Code" method. You can directly add devices by scanning with WeChat scan or scanning within the Tencent Lianlian Mini Program. The specific instructions are as follows.

Device Type

Wi-Fi device, BLE device, Zigbee subdevice, 433 subdevice, custom protocol subdevice.

How to Obtain

The "universal QR code for one model" of such devices can be obtained in [IoT Explorer Console](#) > **Batch Production** > **Product Confirmation and Release**. Meanwhile, you can click **Download QR code** to save it.

物模型定义

设备开发

交互开发

设备调试

批量投产

产品确认及发布

您的产品选择接入腾讯连连官方小程序，需通过腾讯连连认证流程，待审批通过后再进行量产。如无需接入，可在【交互开发】关闭接入直接发布产品。

产品名称

test


状态

开发中

产品品类

其他行业 / 其他品类 / 自定义产品

量产二维码



下载二维码 量产二维码文档

接入腾讯连连官方小程序

已启用 [去配置](#) 您选择接入腾讯连连，请下载《[测试报告模板](#)》并参考《[腾讯连连认证厂商测试规范V1.0](#)》编写测试报告提交并邮寄样品审核。

测试报告

上传

请按照上方《[测试报告模板](#)》编写的测试报告，压缩后上传（支持rar、zip、gz，不超过10M）

☐ 我确认该产品将接入腾讯连连官方小程序 ①

☐ 我了解并同意《[授权协议](#)》与《[开发者须知](#)》

确认产品并申请发布

批量投产

您的产品选择接入了连连小程序，需要先完成发布后才可投产

量产管理

上一步

Notes:

How to configure the page entered after scanning? You can click **Interactive Development > Scan Product Introduction/Network Configuration Guide > Configuration** to perform operations.

One Device, One Code

Tencent Lianlian Mini Program and Chinese domestic brand mini programs/apps both support adding devices via the "one device, one code" method. This document uses Tencent Lianlian Mini Program as an example for illustration. Other devices can refer to these operations. The specific instructions are as follows.

Device Type

Cellular devices (2G/3G/4G/5G), LoraWAN, others.

How to Obtain

During the batch production stage of the product, before the product is put into release, the Tencent Lianlian Mini Program and General Version App need to be reviewed by IoT Explorer. Once approved, they can be released. When you create a device mass production task, download the generated csv file to obtain the QR code content of one device, one code.

Notes:

Chinese domestic brand mini programs/Apps do not need to be reviewed by IoT Explorer. You can perform mass production management after job publishing confirmation.

Operation Steps

Tencent Lianlian Mini Program and General Version App

1. Log in to the [IoT Explorer Console](#). Before mass production, the product needs to be reviewed and approved by IoT Explorer before it can be released. During the product confirmation stage, you need to click **Confirm Product Information and Publish** to enter the application release for mass production.

物模型定义

设备开发

交互开发

设备调试

5 批量投产

产品确认及发布

您的产品选择接入腾讯连连官方小程序，需通过腾讯连连认证流程，待审批通过后再进行量产。如无需接入，可在【交互开发】关闭接入直接发布产品。

产品名称

test


状态

开发中

产品品类

其他行业 / 其他品类 / 自定义产品

量产二维码



[下载二维码](#) [量产二维码文档](#)

接入腾讯连连官方小程序

已启用 [去配置](#) 您选择接入腾讯连连，请下载《[测试报告模板](#)》并参考《[腾讯连连认证厂商测试规范V1.0](#)》编写测试报告提交并邮寄样品审核。

测试报告

上传

请按照上方《[测试报告模板](#)》编写的测试报告，压缩后上传（支持rar、zip、gz，不超过10M）

☐ 我确认该产品将接入腾讯连连官方小程序

☐ 我了解并同意 [《授权协议》](#) 与 [《开发者须知》](#)

确认产品并申请发布

批量投产

您的产品选择接入了连连小程序，需要先完成发布后才可投产

量产管理

上一步

2. Fill in after the publishing is completed and wait for approval.
3. After review passed, select **Device Mass Production** in the left sidebar, click **Mass Production Management > Bulk Device Creation**.
4. After entering the "Create Mass Production" interface, select the mass production product. In the "One Device, One Code QR Code" column, select "Automatically Generate". For details on filling in other

创建量产

量产产品 *

一机一码设备

产品ID

8

烧录方式

一机一密

一型一密

一个产品下每个设备烧录产品ID以及唯一的DeviceName与DeviceSecret

生成方式

自动生成

上传文件

由系统自动生成随机并且唯一的DeviceName与DeviceSecret

量产数量

-

1

+

最多一次性量产10000个设备

一机一密二维码

自动生成

无需生成

确定

取消

| | | | | | |
|------|------|---------|------|---------------------|----|
| 设备管理 | | 量产管理 | | | |
| 创建量产 | | | | | |
| 批次 | 产品ID | 产品名称 | 烧录方式 | 创建时间 | 操作 |
| LI | 34 | 量产一码通用型 | 一机一密 | 2021-04-13 11:03:28 | 查看 |

```
ProductId, DeviceName, ProductSecret, DevicePsk, Status, QRCode
8E4RA7CJ2F,0000000000,PKpb04tDA865x9Ro8rGgKA==,{"ProductId":"","8E4RA7CJ2F,0000000000,"Signature":"","cc6fe7274157...4004a92a16a""}
8E4RA7CJ2F,0000000001,E0y414IoabZVIC8mV6kfw==,{"ProductId":"","8E4RA7CJ2F,0000000001,"Signature":"","8314d242fa...f6b3b3bd939c"
8E4RA7CJ2F,0000000002,ov3kZw9jgQ0icQraccw==,{"ProductId":"","8E4RA7CJ2F,0000000002,"Signature":"","0c6b79da91c...9c9cbde1981""}
8E4RA7CJ2F,0000000003,rj5aSyAGSMybBytp/erX6g==,{"ProductId":"","8E4RA7CJ2F,0000000003,"Signature":"","d7e3afalc1b...0b8dd1b91f1""}
8E4RA7CJ2F,0000000004,sj2zhBRir2laratsyqD/g==,{"ProductId":"","8E4RA7CJ2F,0000000004,"Signature":"","dfb5da95c68...96a6ad1b260""}
```

Page 51 of 334

物模型定义

设备开发

交互开发

设备调试

5 批量投产

产品确认及发布

如您已完成产品开发及调试，可确认下方信息发布产品开始批量投产，发布后会禁用物模型修改以避免误操作影响设备用户，您可随时撤销发布进行修改

产品名称

test

状态

开发中

产品品类

其他行业 / 其他品类 / 自定义产品

接入腾讯连连
官方小程序

未接入 [去配置](#) 您选择不接入连连官方小程序，可使用平台提供的SDK开发自有品牌的小程序

确认信息并发布

批量投产

我们推荐完成设备开发及调试后发布设备再进行投产

量产管理

2. Select **Device Mass Production** in the left sidebar, click **Mass Production Management > Bulk Device Creation**.
3. After entering the "Create Mass Production" interface, select the mass production product. In the "**One Device, One Code QR Code**" column, select "Automatically Generate". For details on filling in other parameters, see [Mass Production Management](#). Click **OK** to complete the creation of mass production.

创建量产

量产产品 *

一机一码设备

产品ID

8

烧录方式

一机一密

一型一密

一个产品下每个设备烧录产品ID以及唯一的DeviceName与DeviceSecret

生成方式

自动生成

上传文件

由系统自动生成随机并且唯一的DeviceName与DeviceSecret

量产数量

-

1

+

最多一次性量产10000个设备

一机一密二维码

自动生成

无需生成

确定

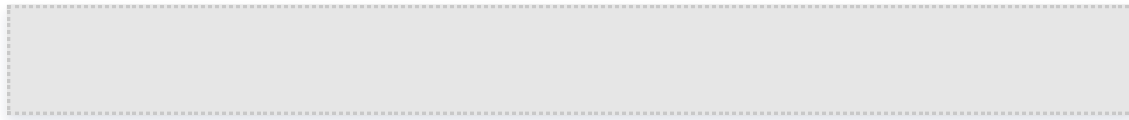
取消

4. After successful creation of mass production, the batch list will appear on the "Mass Production Management" page. Click **View** on the right side of the batch to obtain detailed information about mass production.



| 批次 | 产品ID | 产品名称 | 烧录方式 | 创建时间 | 操作 |
|----|------|---------|------|---------------------|--------------------|
| LI | 34 | 量产一码通用型 | 一机一密 | 2021-04-13 11:03:28 | 查看 |

5. On the mass production details page, click **Download Device Information**. In the generated file, the content after QR-code is the binding QR code content of each device.



Introduction to Mass Production Management

After the device is developed and tested on IoT Explorer, it will enter the mass production stage. After the product is published, users need to generate device certificates and burn them into devices for mass production.

The mass production stage generally goes through at least the following three stages:

1. The device owner batch-generates the key information (DeviceName, DeviceSecret) of devices in IoT Explorer.
2. The device owner distributes the generated key information of the device to the authorized device manufacturer for burning.
3. The device manufacturer performs production testing according to the test requirements of the device owner. The devices that pass the production test will be packaged and delivered. The devices that fail the production test will not be packaged or delivered. Further analysis is required for the causes for the failure of the production test.

Prerequisites

- The product has completed the preceding development steps, the product has been released, and entered the batch production stage.
- The actual production environment of the equipment already meets the qualification requirements for mass production.

Procedure of Mass Production

1. Log in to the [IoT Explorer](#), and select **Public Instance**.
2. Click on the left menu **Mass Production of Devices > Mass Production Management** to view the current mass production product records.

Notes:

The first time you enter Mass Production Management, if there is no data, "No mass production records. Click to create mass production." will be displayed in the list area.

3. 1. Click **bulk device creation**, and the user needs to input specific product information and mass production burning method.
- **Mass-produced products:** Select a published product from the drop-down selection. The product ID will be automatically selected.
 - **Burning method:** You can choose two burning methods: one device one key and one model one key. For details, see [Select a burning method](#).
 - Ways to produce
 - For the one device one key burning method, you can choose system automation to randomly generate a unique DeviceName and DeviceSecret, or upload a file as the DeviceName and generate the corresponding DeviceSecret.
 - For one model one key, you can choose to upload a file as the DeviceName and generate the corresponding DeviceSecret.
 - **Production quantity:** A maximum of 10,000 devices can be mass-produced in a single run.
 - **One Device, One Secret QR code:** Select as required.

创建量产

量产产品 *

文档-非官方一机一码

产品ID

7R

烧录方式

一机一密

一型一密

一个产品下每个设备烧录产品ID以及唯一的DeviceName与DeviceSecret

生成方式

自动生成

上传文件

由系统自动生成随机并且唯一的DeviceName与DeviceSecret

量产数量

-

1

+

最多一次性量产10000个设备

一机一密二维码

自动生成

无需生成

确定

取消

4. After selecting the burning method, click **Confirm**. The backend will process the batch task.
5. When the backend finishes reviewing and processing the mass production task, the mass production information will be displayed, and the batch device information will be provided for download for the manufacturer to burn.

Select a Burning Method

The burning method is divided into local burning and dynamic obtaining. The mass production process of local burning is generally used by enterprises that produce, manufacture, and burn devices by themselves. The key information such as DeviceName and DeviceSecret will not be disclosed to external partners, reducing the risk of key information being transferred during the distribution stage.

| Feature Item | Local Burning | Dynamic Obtaining |
|----------------------------|--|---|
| Device burning information | Device certificate, namely: ProductID, DeviceName, DeviceSecret. | Device certificate, namely: ProductID, ProductSecret, DeviceName (device name, generally the MAC address, SN, etc. of the device itself). |
| Generation Method | Automatically generate and upload files. | Upload file. |
| Production Quantity | 10,000 devices under a single product. | 10,000 devices under a single product. |
| Security | Relatively high. | Moderately low. |
| Enabled by default. | Yes. | Manually enable in the console. |

Local Burning

The process of local burning also comes in two kinds:

- First method: The system automatically generates a batch of DeviceNames and DeviceSecrets.
- Second method: The device owner uploads the product serial number as the DeviceName, and then the system generates a DeviceSecret that has a one-to-one correspondence with it according to the uploaded serial number.

System-Generated DeviceName and DeviceSecret

1. The device owner selects **one device one key** for direct burning.
2. The device owner selects the number of devices to be generated in batches.
3. The backend automatically generates a unique DeviceName and DeviceSecret based on the device quantity.
4. The backend outputs the generated equipment information via downloading a CSV file.
5. The device owner can perform the burning process during the specific generation process after downloading the file.

System Generates Paired DeviceSecret According to the User Imported DeviceName

1. The device owner selects **one device one key** for direct burning.
2. The device owner selects the file upload method and imports the pre-prepared file.
3. The backend uses the data in the first column of the uploaded file as the DeviceName and automatically generates the corresponding DeviceSecret.
4. The backend outputs the generated equipment information via downloading a CSV file.
5. The device owner can perform the burning process during the specific generation process after downloading the file.

Dynamic Obtaining

The purpose of dynamically obtaining the device is not to provide the DeviceSecret during the distribution stage, but only to provide the ProductID, ProductSecret, and DeviceName. During the production testing phase, the device will dynamically obtain the corresponding DeviceSecret from the cloud based on the

ProductID, ProductSecret, and DeviceName. After receiving it, the device will store the DeviceSecret and then initiate the normal login process of the device. It can be applied to scenarios where device key information needs to be distributed multiple times.

Console Operation Process

1. The device owner selects the burning method of dynamic registration with **one model one key**.
2. The device owner selects the file upload method and imports the pre-prepared file.
3. The backend uses the data in the first column of the uploaded file as the DeviceName.
4. The system outputs the product Secret Parameter for the product that selects "dynamically obtain".
5. The device owner distributes the list file of ProductID, ProductSecret and DeviceName to the manufacturer.

Error Prompt

When performing mass production, you may encounter error prompts. See the following table for error codes and reasons.



| Error Code | Error Cause |
|--|--|
| hub create device timeout | Device creation timeout |
| normal device license limit | Insufficient number of ordinary device activation codes |
| video device license limit | Insufficient number of audio/video device activation codes |
| If you encounter other notifications, please Submit a Ticket for consultation, or proceed to purchase. | |

Additional Notes

The process for devices to dynamically obtain keys during the manufacturer's programming and production testing phase when devices go online is as follows:

1. The device manufacturer burns the data obtained from the device owner.
2. Perform production testing after completion of burning.
3. After the device is powered on, if the firmware program checks and finds no DeviceSecret locally, it will send the product ID, product key, and DeviceName through the API encapsulated by the Device SDK. The SDK will retrieve the DeviceSecret from the cloud.
 - 3.1 First, perform signature verification on the validity of the request.
 - 3.2 Secondly, check whether the DeviceName already exists in the cloud.
 - If not exist, registration fails and production test also fails.
 - If the device exists, the cloud will return an encrypted DeviceSecret for the DeviceName.
4. After receiving the DeviceSecret, the device decrypts it and stores it locally.
5. The device side initiates an MQTT log-in request to the cloud with the dynamically obtained DeviceSecret.
6. If the login is successful, it indicates that the first step of the production test has passed.

Activation Code Overview

Last updated: 2025-04-27 17:31:51

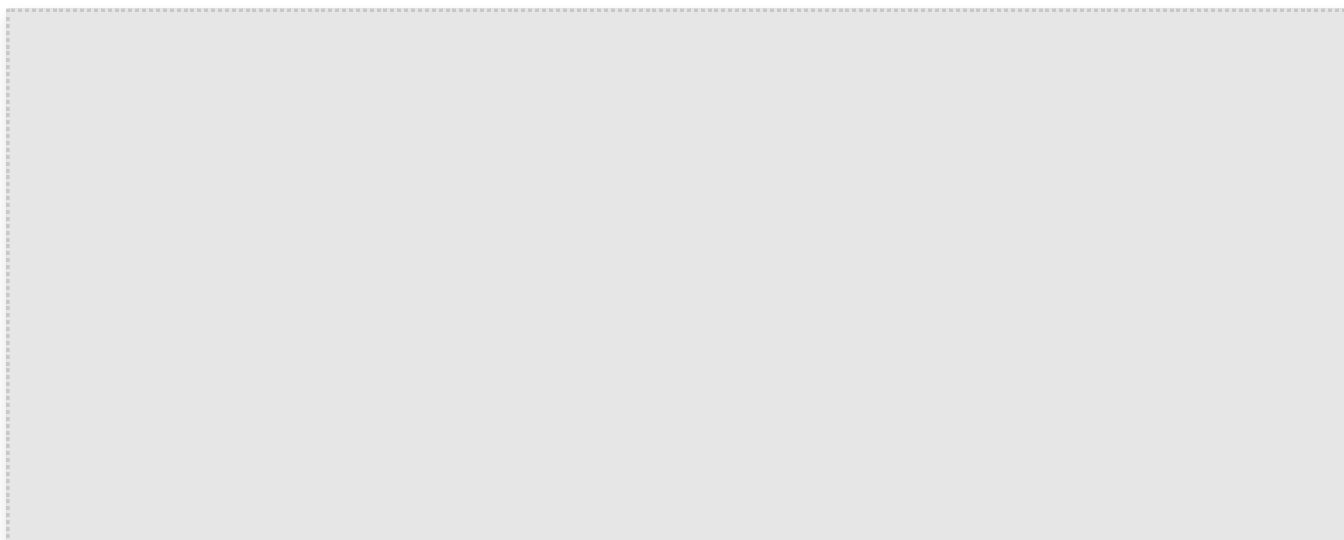
Activation code overview is used to view the total count and remaining quantity of device activation codes of the current account. It mainly includes ordinary connection device activation codes, value-added feature activation codes such as audio and video.

Operation Steps

1. Log in to [IoT Explorer](#), select **Public Instance**. Click to enter the menu **Mass Production > Activation Code Overview**.
2. View the activation code overview, including the number of registrable devices, the number of registered devices, and the remaining number of registrable devices.



3. View the activation code details, including device activation codes, audio and video activation codes, and value-added activation codes.
 - **Device activation code:** Each device of a public instance that accesses a platform needs to use a device activation code. Meanwhile, the quantity of device activation codes impacts the TPS of the public instance. Support viewing the quantity of free test activation codes (pending use/total valid count), the quantity of non-Bluetooth device activation codes (pending use/total valid count), and the quantity of Bluetooth device activation codes (pending use/total valid count).



- **Audio and video activation code:** In addition to the device activation code, devices of the audio and video service category need to consume additional audio and video activation codes for using network resources to transmit audio and video.

设备激活码 ⓘ

音视频激活码 ⓘ

增值激活码 ⓘ

0.5 Mbps （待使用/有效总数）

0 / 0 个

1 Mbps （待使用/有效总数）

8 / 10 个

1.5 Mbps （待使用/有效总数）

0 / 0 个

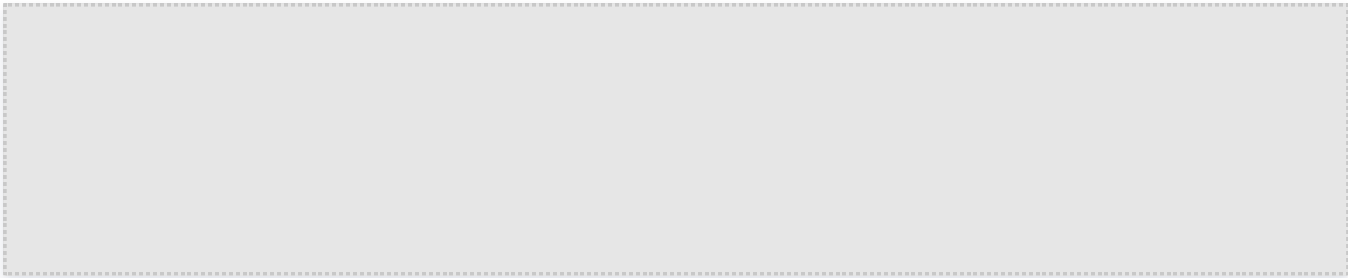
2 Mbps （待使用/有效总数）

0 / 0 个

音视频激活码详情

| 平均传输速率 ⌵ | 使用年限 ⌵ | 激活码总数 | 已使用个数 | 待使用个数 |
|----------|--------|-------|-------|-------|
| 1 Mbps | 5年 | 10 | 2 | 8 |

- **Value-added activation code:** The value-added service activation code is used to enable paid value-added services such as face recognition, Tencent Real-Time Communication (TRTC), voice assistant, and WeChat Call for devices.



After-Sales Operation and Maintenance Firmware Upgrade

Last updated: 2025-04-27 17:32:16

Overview

This document primarily introduces the method of use of firmware upgrade on IoT Explorer, helping you quickly use the firmware upgrade service.

Operation Steps

Add Firmware

1. Log in to the [IoT Explorer console](#) and enter a public instance.
2. Select **After-sales Maintenance > Firmware Upgrade** in the left sidebar menu to enter the firmware list page, where all current firmware can be viewed.
3. Click **Add Firmware** to add new firmware.

添加新固件

×

① 若固件大小限制不能满足您的业务需求, 可 [提交工单](#) 申请增加文件大小限制

固件名称 *

请输入固件名称

支持中文、英文大小写、数字、部分常用符号(下划线, 减号, 括号), 必须以中文、英文或数字开头, 长度不超过32个字符

所属产品 *

请选择产品

固件版本号 *

请输入固件版本号

仅支持英文字母、数字、点、中划线和下划线, 长度限制1~32

选择固件 *

[点击选择固件](#)

仅支持 .bin .dav .tar .gz .zip .gzip .apk .xz .pack 格式的文件, 文件大小不能超过1024MB

自定义信息 ① 新增

通过JSON字符串(utf8)形式下发到设备, 总大小不能超过1536字节, 当前序列化后的大小为0

固件描述

对本次上传的固件进行描述和记录, 请输入0-100个字符

对本次上传的固件进行描述和记录, 请输入0-100个字符

提交

重置

- **Firmware name:** Supports Chinese, English uppercase and lowercase letters, digits, and some commonly used symbols (underscore, minus sign, parenthesis). It must start with Chinese, English, or digits and its length must not exceed 32 characters.
- **Associated product:** Select the product to which the uploaded firmware belongs.

- Firmware version number: Only supports English letters, digits, periods, hyphens, and underscores. Length limited to 1 – 32 characters.
- Select firmware: The uploaded firmware file must be in the specified format: .bin, .dav, .tar, .gz, .zip, .gzip, .apk, .xz, .pack. Individual file size must not exceed 1024 MB.
- Firmware description: Describe and record the uploaded firmware this time. Length limit 0 – 100 characters.
- Custom information: After an upgrade task is created, the custom information will be sent to the device in the upgrade notice.

Note:

A maximum of 100 firmware files can be uploaded under one account. If you continue uploading, deleting old versions of firmware is required.

4. After uploading is complete, the firmware will be displayed in the list. You can perform operations such as upgrading, CRUD operations, and viewing details on the firmware.

Firmware Upgrade

After the firmware is successfully uploaded, select the target firmware version you want to upgrade to and click **Firmware Upgrade** on the right side of the firmware list to initiate an upgrade task. The firmware upgrade methods support two kinds of batch upgrade methods: upgrading by firmware version number and upgrading by device name.

| 添加固件 | | | | |
|------|-------|------|---------|--|
| | | 全部产品 | 请输入固件名称 | |
| 固件名称 | 固件版本号 | 所属产品 | 添加时间 | 操作 |
| | | | | 固件升级 查看详情 删除 |

Upgrade by Firmware Version Number

1. Enter the firmware upgrade page. The page displays information about the target firmware to be upgraded (for example, firmware name, associated product, firmware version number).
2. Select **batch upgrade method** as "Upgrade by Firmware Version".

固件升级

×

固件名称

测试

所属产品

固件版本号

升级模式

静态升级

动态升级

批量升级方式 ①

按固件版本

按设备名称

待升级版本号

请选择版本号

升级范围

全部设备

升级确认

静默升级

超时时长配置 ①

–

15

+

分钟

保存

取消

- Version number to be upgraded: Select the firmware version number in the dropdown list as the firmware to be upgraded. Multiple selections allowed.
- Upgrade scope: Support two kinds of upgrade scopes. All devices under the to-be-upgraded version number or specified devices can be used as target devices for firmware upgrade.

The feature of upgrading specified devices is commonly used for grayscale verification of firmware content. When the upgrade scope is set to specified devices, click the **Select Device** button on the right of the dropdown list to select multiple target upgrade devices from all devices under this product.
- Upgrade confirmation: Support two kinds of firmware upgrade confirmation methods: silent upgrade and user confirmation upgrade. If you use the official Tencent Lianlian application, silent upgrade means that no user confirmation is required. The Tencent Lianlian application side will automatically complete the upgrade. After re-enabling, it will be the upgraded version; user confirmation upgrade means that the user needs to actively enter the device control interface of Tencent Lianlian and check and confirm the firmware upgrade on the device details page. If you use other IoT applications, it is recommended to choose the silent upgrade method.

固件升级

×

| | |
|----------|---|
| 固件名称 | 测试 |
| 所属产品 | 腾讯连连H5小测试 |
| 固件类型 | MCU |
| 固件版本号 | test |
| 批量升级方式 ① | <div><div>按固件版本</div><div>按设备名称</div></div> |
| 待升级版本号 | <div>请选择版本号</div> |
| 升级范围 | <div>全部设备</div> |
| 升级确认 | <div>静默升级</div> |
| 超时时长配置 ① | <div><div>静默升级</div><div>用户确认升级</div></div> |

保存

取消

- Timeout duration configuration: When the cloud does not receive the device firmware upgrade message exceeding the timeout duration, the firmware upgrade task will be redistributed. The default timeout duration for silent upgrade is 15 minutes. The default timeout duration for user confirmation upgrade is 2 minutes. You can also customize the configuration based on actual business needs.

3. Click save. Then, the system will perform an upgrade task and distribute the selected target version firmware to the target devices within the upgrade scope.

Note:

Upgrading by firmware version requires the device to be upgraded to report the currently running firmware version. If it is not uploaded, you can choose to upgrade by device name as introduced below.

Upgrade by Device Name

1. Enter the firmware upgrade page. The page displays information about the target firmware to be upgraded (for example, firmware name, associated product, firmware version number).
2. Select the batch upgrade method as **upgrade by device name**.

固件升级

×

固件名称

测试

所属产品

腾讯连连H5小测试

固件类型

MCU

固件版本号

test

批量升级方式 ⓘ

按固件版本

按设备名称

指定设备

点击选择文件

下载模板

上传文件中请录入准确的DeviceName，一次最多可升级10000个设备，仅支持csv格式。

升级确认

静默升级 ▼

超时时长配置 ⓘ

—

15

+

分钟

保存

取消

- Specify the device: Upload the list of devices that need firmware upgrade. Click **Download Template** to get the template file, enter the accurate DeviceName in the template file, and then click **Upload File** to upload. Up to 10,000 devices can be upgraded at a time. File support only csv format.
- Upgrade confirmation: Identical to upgrading by firmware name, it supports two firmware upgrade confirmation methods: silently upgrade and user confirmation upgrade.
- Timeout duration configuration: When the cloud does not receive the device firmware upgrade message exceeding the timeout duration, the firmware upgrade task will be redistributed. The default timeout duration for silent upgrade is 15 minutes. The default timeout duration for user confirmation upgrade is 2 minutes. You can also customize the configuration based on actual business needs.

3. Click save. Then, the system will perform an upgrade task and send firmware to the target device.

View Firmware Details


1. Click **View Detail** on the right side of the firmware list in the firmware list to view firmware details.

| 添加固件 | | 全部产品 ▼ | | 请输入固件名称 🔍 |
|------|-------|-----------|---------------------|---------------------|
| 固件名称 | 固件版本号 | 所属产品 | 添加时间 | 操作 |
| 测试 | test | 腾讯连连H5小测试 | 2023-11-15 10:00:00 | 固件升级 查看详情 删除 |

2. Enter the firmware details page, where you can view the detailed information of the firmware, firmware upgrade device statistics, and upgrade task management list.

固件信息

编辑

| | | | |
|------|---|------|---------------------|
| 固件名称 | 测试 | 签名算法 | Md5 |
| 所属产品 | 腾讯连连H5小测试 | 添加时间 | 2020-11-17 14:26:45 |
| 固件版本 | MCU test | 固件描述 | |
| 固件签名 |  | | |
| 固件类型 | MCU | | |

固件升级设备统计

| | | | |
|----------|------|------|------|
| 固件升级设备总数 | 升级成功 | 正在升级 | 升级失败 |
| 0 | 0 | 0 | 0 |

任务管理

任务明细

设备明细

请输入任务id

| 任务id | 任务类型 | 任务状态 | 添加时间 | 操作 |
|------|------|------|------|----|
|------|------|------|------|----|

- Firmware information: including firmware name, associated product, firmware version number, firmware signature, signature algorithm, addition time, and firmware description. Click the Edit Button in the upper right corner to modify the firmware name and description.
- Firmware upgrade device statistics: including the total number of devices in all batch upgrade tasks for this firmware, as well as the number of devices corresponding to upgrade tasks with different upgrade statuses.
- Task management list
 - Click **Task Details** to view all upgrade tasks for this firmware. The statuses of upgrade tasks include: not started, creating, creation succeeded, creation failed.



- click **Device Details** to view the record details of device upgrades in all upgrade tasks associated with this firmware. There are 5 device upgrade statuses: Pending, Pushed, Upgrading, Successfully Upgraded, and Upgrade Failure.

任务管理

任务明细 设备明细

请输入完整的任务ID 请输入设备名称

| 设备名称 | 任务ID | 当前版本号 | 升级状态 | 状态更新时间 | 操作 |
|-------|------|-------|------|--------|------|
| light | | - | 已推送 | | 查看详情 |

3. In the **Task Details** or **Device Details** of task management, click **View Details** on the right of a task to go to the task details page, where you can view the device list, upgrade status, and statistics of the number of devices in different upgrade statuses for this task upgrade.

固件升级 / 固件详情 / 任务详情 使用指南

任务信息

任务ID 1

产品名称 智能小灯

目标版本号 3.1.4

升级范围 全部设备

升级时间 2020-11-01 22:50:13

升级方式 批量升级

任务统计

已推送: 100%

● 已推送

设备详情

全部设备(1) 升级成功(0) 待推送(0) 已推送(1) 升级中(0) 升级失败(0)

请输入设备名称

| 设备名称 | 当前版本号 | 最后更新时间 | 升级状态 | 状态详情 | 操作 |
|-------|-------|---------------------|------|------|----|
| light | - | 2020-11-01 22:50:23 | 已推送 | - | 取消 |

In the device detail list, you can view the current upgrade status and status details of all devices in the batch task upgrade of this task.

- Do not display status details when the upgrade status is "Pending" or "Pushed".
- When the upgrade status is "Upgrading", the status details include: downloading, burning, and simultaneously display the percentage progress.
- When the upgrade status is "Upgrade Failure", the status details will feedback error information.

Additionally, on the right side of the device detail list, you can cancel or retry the device upgrade based on the upgrade progress. The upgrade status of a canceled upgrade device will be marked as upgrade failure; for a device with an upgrade failure, you can click **Retry** to perform re-upgrade.

Device management

Last updated: 2025-04-27 17:32:35

Overview

After successfully mass-producing devices for a specific product on IoT Explorer, you can manage and view device information in the console.

Operation Steps

Managing Devices

1. Log in to [IoT Explorer](#).
2. On the instance overview page, find the public instance that needs to be turned on. Click the corresponding instance to navigate to the instance detail page.
3. Click **Equipment Management** in **Aftersales Operations** in the left sidebar to enter the equipment management page. You can perform the following operations:
 - View equipment information under a specific product: Select a product in the pull-down menu at the top of the page. You can view the current device status:
 - Inactive: The device is not integrated with IoT Explorer. You can download the Device SDK for device development and integrate the device with the IoT platform to activate it.
 - Online: The device is activated and has a successful connection with the IoT development platform.
 - Offline: The device has been activated and is disconnected from the IoT development platform.
 - Search for devices: Select the device name or device tag in the search input box on the right to search for specific devices. Fuzzy search is supported.
 - View device details: Find the corresponding device in the list and click **View** to enter the device details page.
 - Delete a certain device: Find the corresponding device in the list and click **Delete** to delete the device. After the device is deleted, the device certificate information will become invalid, and the data records of the device on the IoT platform will also be deleted.

全部产品

已注册设备45个 ① | 已激活设备19个 ①

删除

禁用

启用

设备名称

输入设备名称搜索

| <input type="checkbox"/> 设备名称 | 所属产品 | 设备类型 | 状态 | 禁用/启用 | 激活时间 | 最后上线时间 | 操作 |
|--------------------------------------|-------------|------|-----|---------------------------|------|--------|---------------------------------------|
| <input type="checkbox"/> <div></div> | <div></div> | 设备 | 未激活 | <div><div></div>已启用</div> | - | - | 查看 删除 |
| <input type="checkbox"/> <div></div> | <div></div> | 设备 | 未激活 | <div><div></div>已启用</div> | - | - | 查看 删除 |

View Specific Device Information

In the Device List, click **View** corresponding to the device to enter the device details page. You can perform the following operations.

View Device Information

On the device details page, select **Device Information** to view the basic device information. This includes:

- Device key: For using key authentication, this information needs to be burned into the device side.
- Product ID: unique ID, which needs to be burned into the device side.
- Activation time: the time when the device first connects to the development platform.
- Last online time: the time when the device last connected to the development platform.
- Device status: If the device is online, it displays as "online"; if the device is offline, it displays as "offline"; if the device has never connected to the development platform, it displays as "inactive", etc.

[设备信息](#) [在线调试](#) [云端诊断日志](#) [设备云端日志](#) [设备本地日志](#) [扩展信息](#)

设备信息

| | | | | | |
|------|---|--------|---------------------|--------|----|
| 设备名称 | C | 产品ID | B | 所属产品 | 门禁 |
| 设备密钥 | K | 设备创建时间 | 2023-12-14 10:57:02 | 最后上线时间 | - |
| 激活时间 | - | 设备状态 | 未激活 | 固件版本 | - |

标签信息

设备标签 无标签信息

设备本地日志

调试日志 关闭

日志等级 无

[编辑](#)
[编辑](#)

View Device Properties

On the device details page, select **Device Properties** to view the Thing Model feature items of the device. This includes:

- Variable identifier: the identifier in the Thing Model corresponding to the device.
- Variable name: correspond to the "feature name" in the Thing Model.
- Historical data: Click **View** to retrieve the historical report data of this feature.
- Variable Type: The "data type" in the Thing Model.
- Latest value: When a device reports data to the cloud, as long as the latest reported value of a specific feature changes, the Latest Value column will immediately display the latest value reported by the device.
- Update time: Refers to the change time of the latest value, generally the occurrence time when the device reports the specific feature. View the historical reported data of a specific feature. Display the historical data reported by the specific feature to the cloud by time, and verify whether the reported data is correct.

设备信息

在线调试

云端诊断日志

设备云端日志

设备本地日志

扩展信息

物模型日志

内容日志

上下线日志

 ☒ 自动刷新

属性

事件

行为

属性名称/属性标识符

Q

| 标识符 | 功能名称 | 历史数据 | 数据类型 | 最新值 | 更新时间 |
|--------------|-------|--------------------|------|-----|------|
| power_switch | 电灯开关 | 查看 | 布尔型 | - | - |
| brightness | 亮度 | 查看 | 整数型 | - | - |
| color | 颜色 | 查看 | 枚举型 | - | - |
| color_temp | 色温 | 查看 | 整数型 | - | - |
| name | 灯位置名称 | 查看 | 字符串 | - | - |

View Device Cloud Log

On the device details page, select **Device Cloud Log** to view the information that the device has uploaded to the cloud and received from the cloud.

- View Thing Model Logs
 - Uplink: Uplink refers to the model data reported from the device to the cloud.
 - Downlink: Downlink refers to the model data sent from the cloud to the device.

设备信息

在线调试

云端诊断日志

设备云端日志

设备本地日志

扩展信息

物模型日志

内容日志

上下线日志

 ☒ 自动刷新

30分钟

1小时

今天

昨天

近3天

2024-02-27 15:09 ~ 2024-02-27 15:39

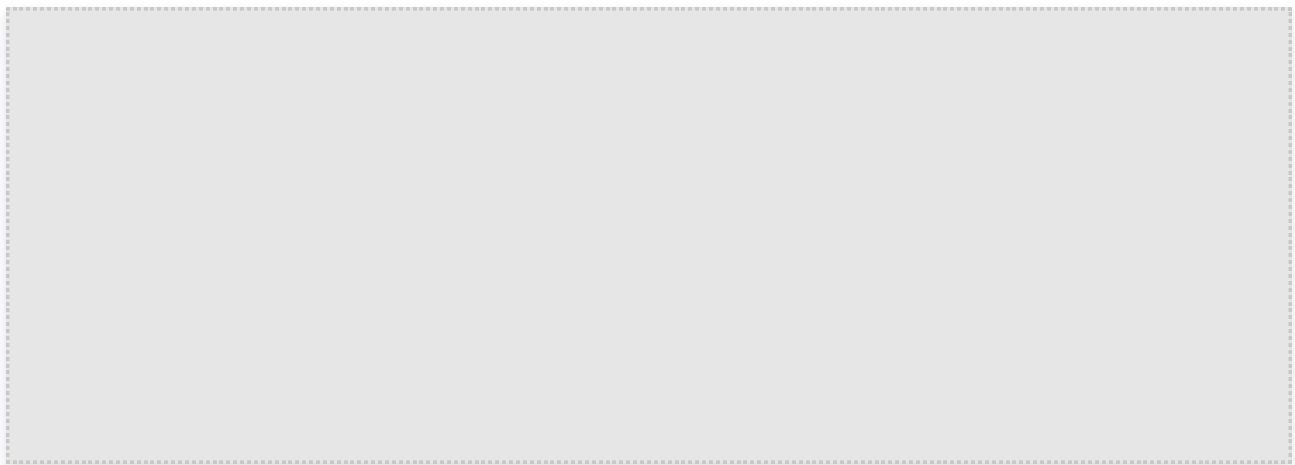


| 时间 | 动作 | 详细信息 |
|-------------------------|----|--|
| 2024-02-27 15:34:45.579 | 下线 | Device keepalive timeout,last active time:2024-02-27 15:28:45 |
| 2024-02-27 15:28:43.810 | 上线 | Device connect,version:4,keepalive:240,cleansession:1,clientip:59.37.125.101 |

- View device events: Select Thing Model logs and click **Event** to view the event information reported by the device to the cloud, This includes:
 - Definition of event: Defined and managed in the Thing Model.
 - Event type: The system divides event types into three types, namely Alarm, Fault, and Info.



- View device behavior: Select Thing Model logs and click **Behavior** to view the behavior information of the device, This includes:
 - Definition of behavior: Defined and managed in the Thing Model.
 - Behavior description: Used to describe complex business logic. Multiple calling parameters and response parameters can be added. It can be used to let devices perform a specific task. For example: The unlocking action needs to know which specific user unlocks at what time, as well as the status of the lock.



- View device online and offline logs: View the log records of the device connecting to and disconnecting from the cloud.

设备信息

在线调试

云端诊断日志

设备云端日志

设备本地日志

扩展信息

物模型日志

内容日志

上下线日志

自动刷新

30分钟

1小时

今天

昨天

近3天

2024-02-27 15:09 ~ 2024-02-27 15:39

| 时间 | 动作 | 详细信息 |
|-------------------------|----|--|
| 2024-02-27 15:34:45.579 | 下线 | Device keepalive timeout,last active time:2024-02-27 15:28:45 |
| 2024-02-27 15:28:43.810 | 上线 | Device connect,version:4,keepalive:240,cleansession:1,clientip:59.37.125.101 |

Operational Analysis

Last updated: 2025-04-27 17:32:55

Overview

Use the operation analysis feature to perform statistical analysis on the activation, active, and online device data of all products within the project; and perform statistical analysis on the geographic distribution of active and activated devices of the product.

Prerequisites

There are activated devices, online devices, and devices connected to the platform.

Device Overview

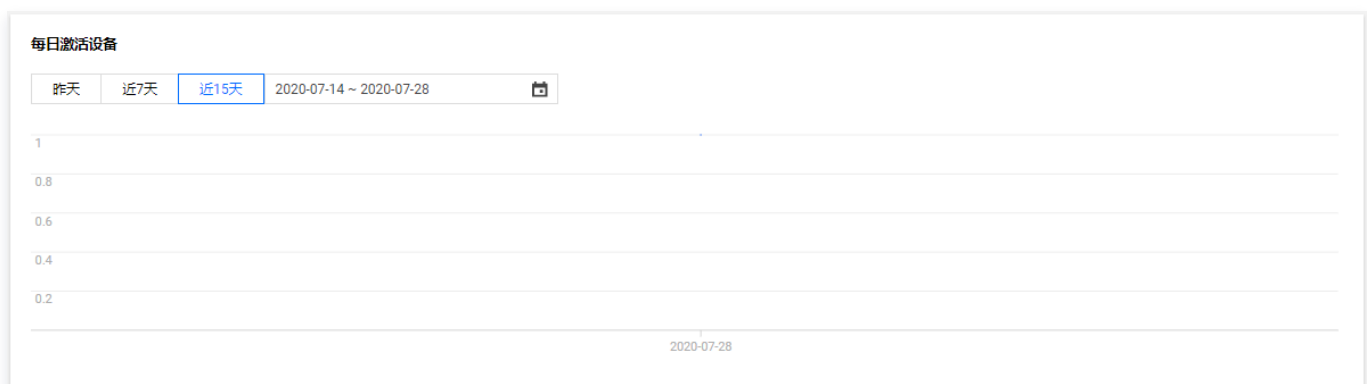
Device overview allows you to view related data of activation, online status, and active status of all products, or view the related data of activation, online status, and active status of a specific product by filtering, as well as information on daily activated devices, daily active devices, and online devices.

Operation Steps

1. Log in to the [IoT Explorer console](#). After entering the public instance, click **Operational Analysis > Device Overview** to enter the overview interface.
2. Select the product you want to view. The page will display the corresponding total number of activated devices, number of online devices, number of activations yesterday, number of active sessions yesterday, number of activations in the last 7 days, number of active sessions in the last 7 days, number of activations in the previous 7 days, and number of active sessions in the previous 7 days.

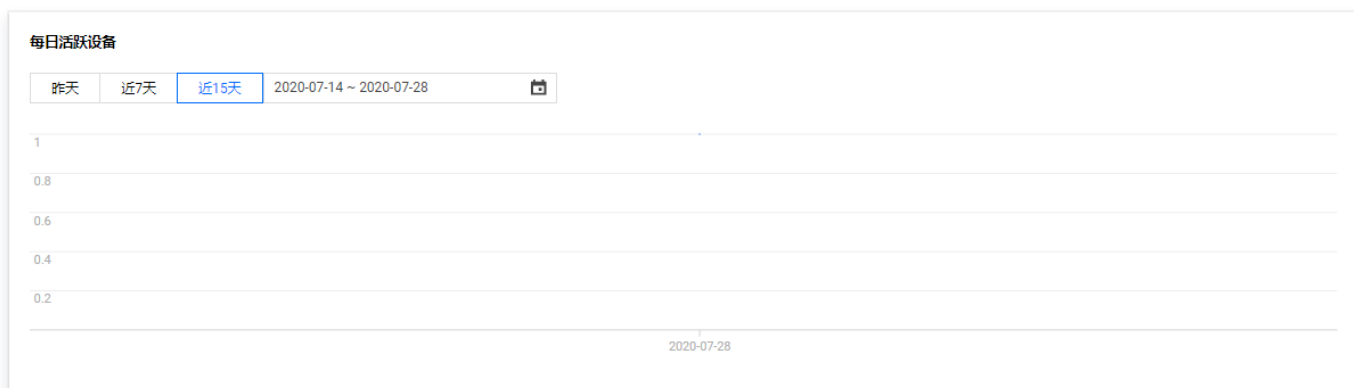
| | | | | | | | |
|------------------------|-------|------|------|-------|-------|-------|-------|
| 当前产品 所有产品 | | | | | | | |
| 激活设备总数 | 在线设备数 | 昨日激活 | 昨日活跃 | 近7日激活 | 近7日活跃 | 上7日激活 | 上7日活跃 |
| 2 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 台 | 台 | 台 | 台 | 台 | 台 | 台 | 台 |

3. Select the time period for viewing the activated device count. The system will show the daily activated device count of the product in the form of a chart within the specified time range of the filter.

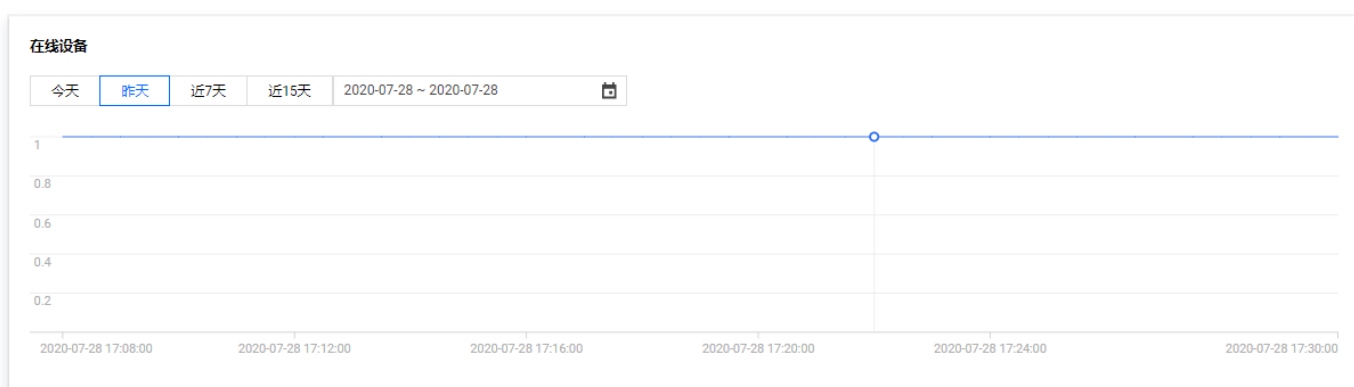


4. Select the time period for viewing active devices (the device must be connected to Tencent Cloud IoT Platform). The system will show the daily number of active devices of the product in the specified time

range of the filter in the form of a chart.



5. Select the time period for viewing online devices. The system will show the daily number of online devices of the product in the specified time range of the filter in the form of a chart.



Device Distribution

Device distribution allows you to view all products, or filter a specific product to view the number of active and activated devices in different regions within the set time. The system provides two display methods: map and table.

Operation Steps

1. Log in to the [IoT Explorer console](#) . After entering the public instance, click **Operational Analysis > Device Distribution** to enter the device distribution interface.
2. Select the product, time period, and device type to view.



3. The system will display the distribution quantity of devices in different regions on the map according to the information such as the product, time period, and device type selected by the user, and intuitively show

the distribution quantity of devices in each region to the user.



4. The system provides a table display method. Users can view detailed information such as the quantity and proportion of each province and city in the table.

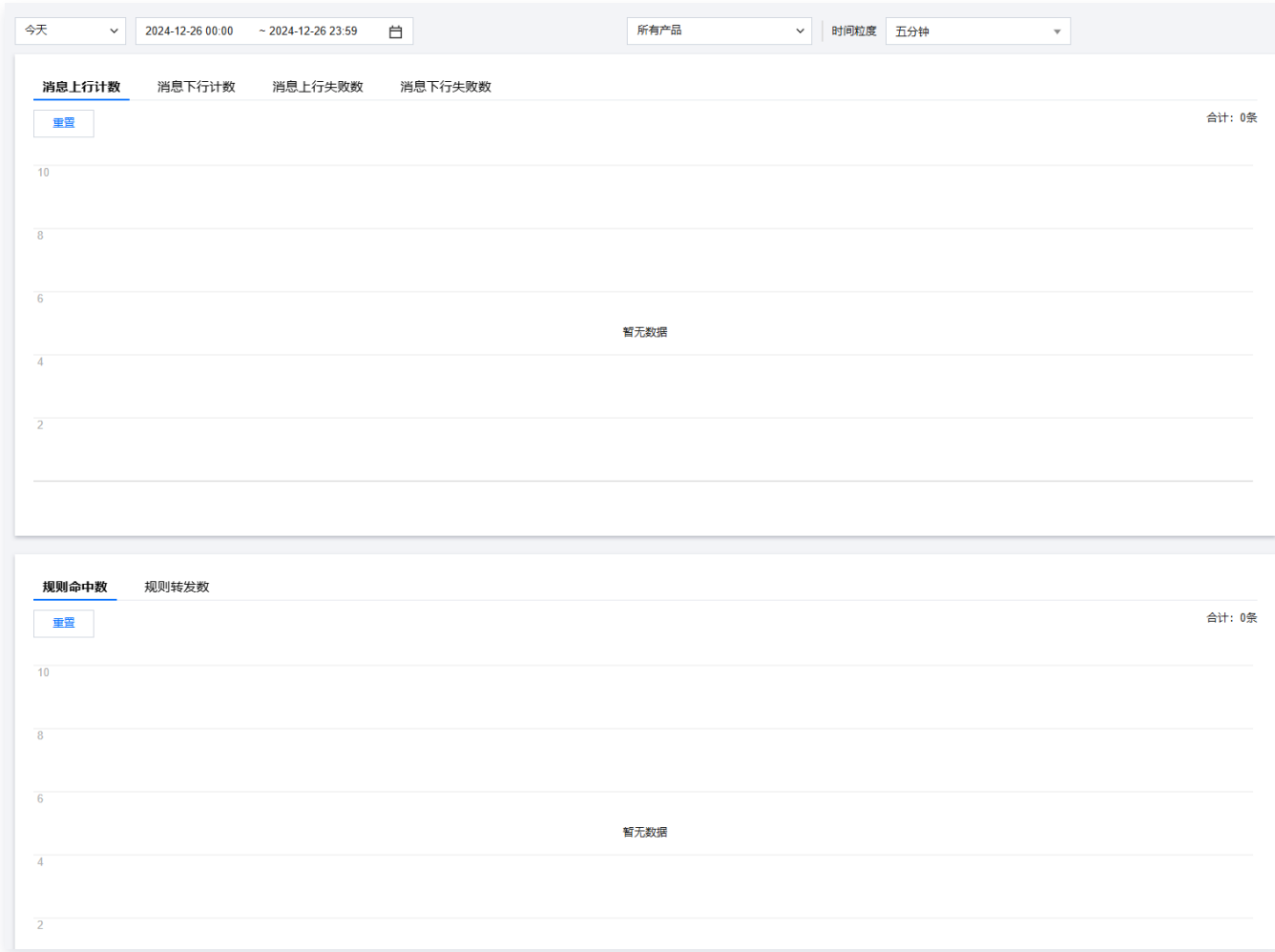
| <div> <div>省份</div> <div>城市</div> </div> | | |
|--|----|---------|
| 地区 | 数量 | 占比 |
| 广东省 | 1 | 100.00% |
| 其它 | 0 | 0.00% |

Message Overview

Message overview allows you to view message upstream count, message downstream count, message upstream failure count, message downstream failure count, rule hits, and rule forwarding count of all products, or view those of a specific product by filtering.

Operation Steps

1. Log in to the [IoT Explorer console](#). After entering the public instance, click **Operational Analysis** > **Message Overview** to enter the device distribution interface.
2. Select the product and time granularity you want to view to view related message data.



Enterprise Instance Device Connectivity Product Management

Last updated: 2025-04-27 17:33:21

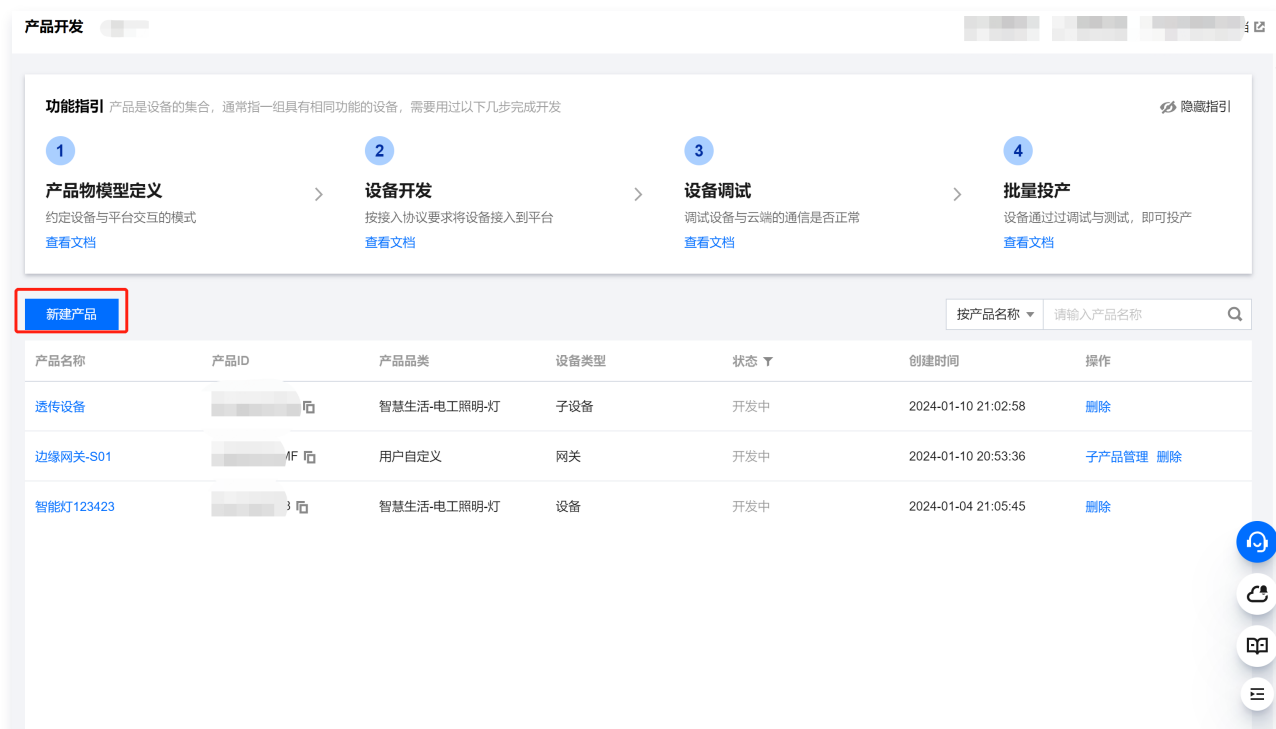
Product Management Overview

User devices need to be connected to the platform. First, you need to create a product. A product refers to a collection of devices of the same type. This document primarily introduces how to use the platform to create, modify products, set identity verification for devices belonging to the product, etc.

Operation Steps

Create a Product

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **enterprise instance**.
2. Click **Enterprise Instance** to enter the instance internal page. Click **Product Development** in the left menu to enter the product list.



3. Click **Create Product**, and the following interface is displayed.

产品开发 / 新建产品 old2 ⇐ 回到旧版 产品开发帮助文档

● 请选择产品品类

产品品类 * 标 已定义标准物模型 免 包含免开发面板 请输入品类

| | | | | |
|------|------|-------------------------------------|---|---|
| 智慧农业 | 公共事业 | <input type="radio"/> 路灯照明 | 标 | 无 |
| 智慧生活 | 环境监测 | <input type="radio"/> 水表 | 无 | 无 |
| 智能城市 | 消防安全 | <input checked="" type="radio"/> 电表 | 无 | 无 |
| 智能制造 | | <input type="radio"/> 燃气表 | 无 | 无 |
| 全屋智能 | | <input type="radio"/> 两轮车 | 标 | 无 |
| 其他行业 | | | | |

请选择产品所属的品类

● 填写产品信息

4. When creating a product, you must select a product category. If there is no suitable category, you can choose **Other Sector > Other Category > Custom Category**. The form page for creating a product will be displayed after selecting a category.

● 填写产品信息

产品名称 *

支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

设备类型

通信方式 *

请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

认证方式

数据协议 ⓘ

描述

最多不超过80个字符

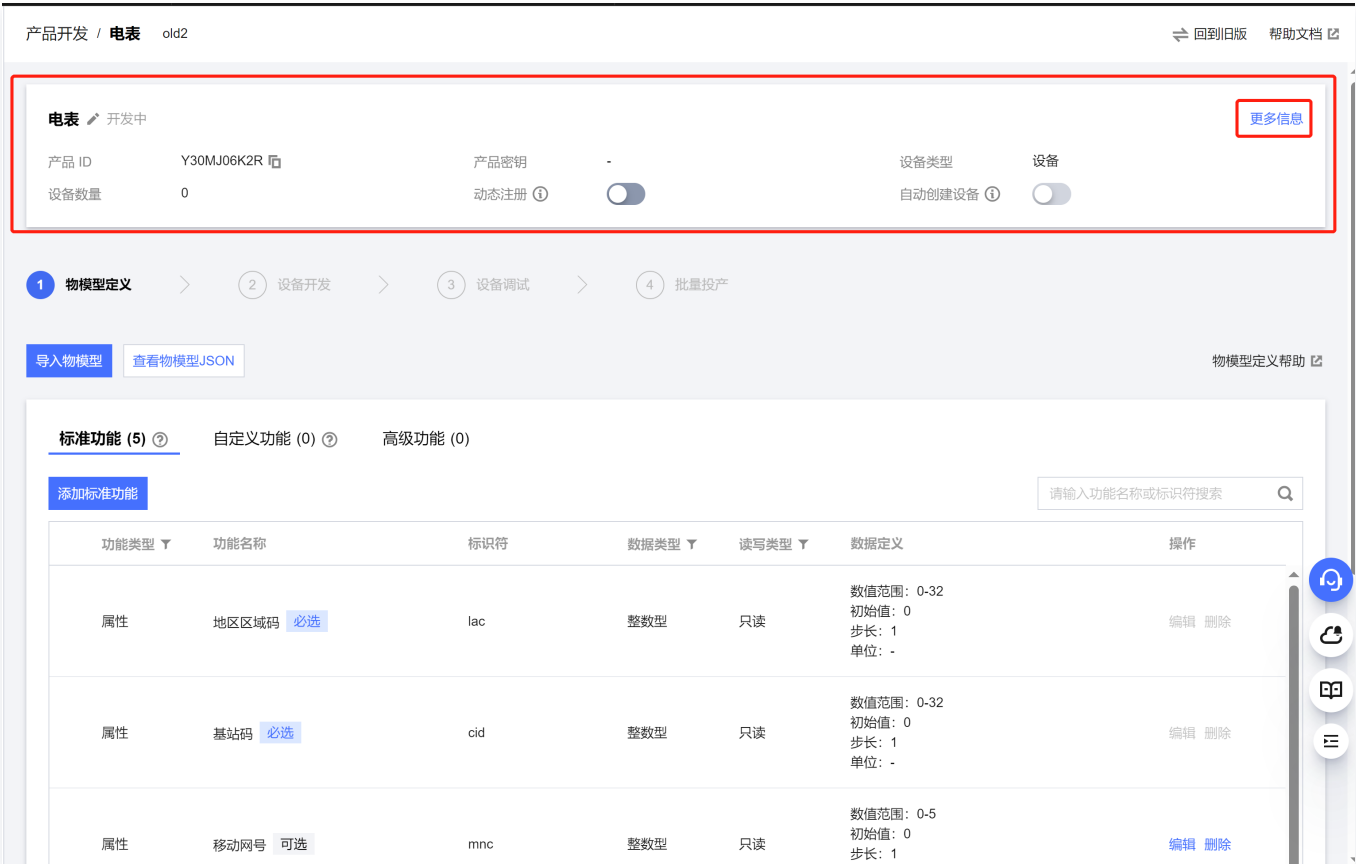
Product information settings are as follows:

- **Product name:** the name of the product. Product names under the same account must be unique. It supports a combination of Chinese characters, letters, digits, and underscores.
- **Device type:** Device type is divided into 3 categories: device, gateway, and subdevice. Details as follows:
 - **Device:** This type of device can access IoT Explorer directly and has no mounted subdevices.
 - **Gateway:** This type of device can directly access the IoT development platform and can accept subdevices to join the LAN. It can maintain the topological relationship of subdevices and synchronize the topological relationship with subdevices to the cloud.

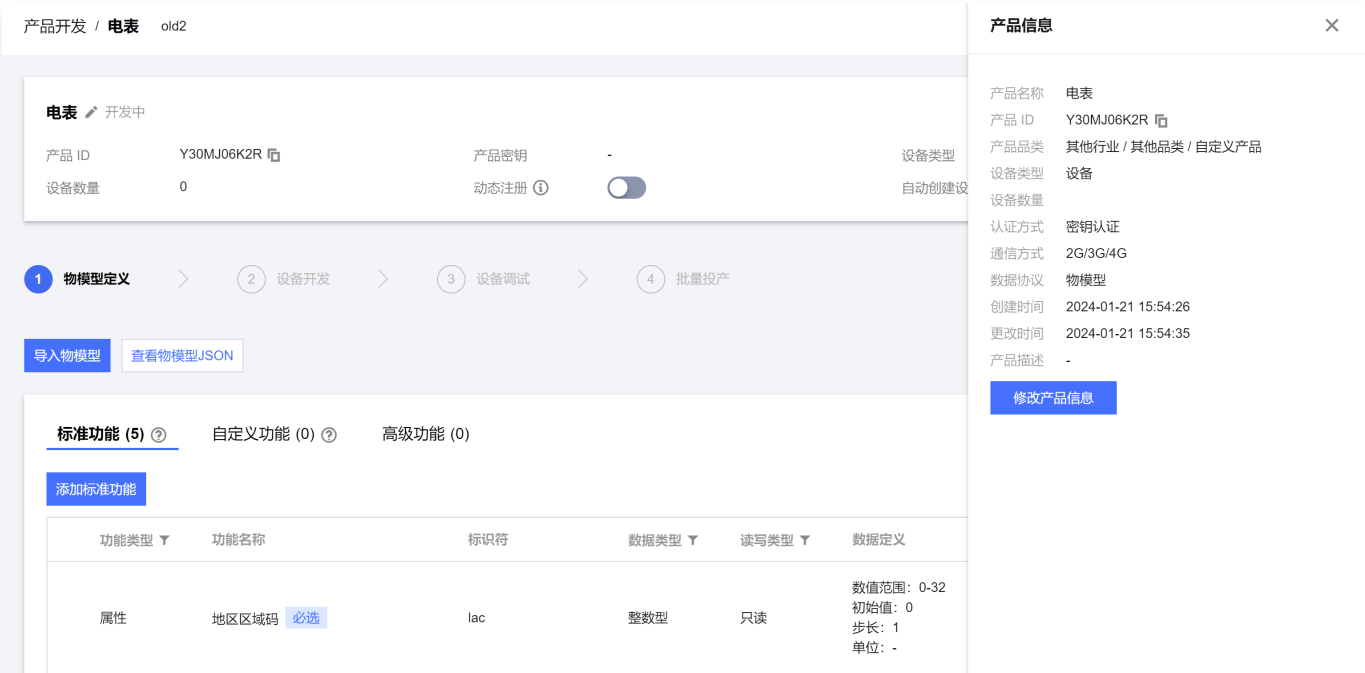
- Subdevice: This type of device must rely on a gateway device to communicate with IoT Explorer. For example, Zigbee, Bluetooth, RF433 and other devices. For the description of gateways and subdevices, refer to the documentation [gateway subdevice](#).
 - Communication method: The communication method is generally a communication method that can be directly connected to the platform for device usage. Including Wi-Fi, mobile cellular (2G/3G/4G), LoRaWAN, Ethernet and other communication methods.
 - Authentication method: IoT Explorer provides two authentication methods for authentication between devices and the platform.
 - Certificate authentication: When creating a device, the platform will generate a certificate file and a private key file for the device to achieve mutual authentication between the device and the cloud.
 - Key authentication: Use the randomly generated device key provided by the platform when creating a device.
 - Access Gateway Protocol (selectable as sub-device when device type is selected): indicates the communication protocol type for the devices under this product to act as sub-devices communicating with the gateway.
 - Zigbee: Indicates that the communication protocol between the subdevice and the gateway is ZigBee.
 - BLE: Indicates that the communication protocol between the subdevice and the gateway is BLE.
 - 433: Indicates that the communication protocol between the subdevice and the gateway is 433.
 - Customization: Indicates that the communication protocol between the subdevice and the gateway is another standard or a private protocol.
 - Data protocol: Defaults to the data protocol of the MQTT Thing Model. Alternatively, you can use a custom MQTT protocol for pass-through.
 - Description: The word count must not exceed 80 characters. You can fill in this field as needed.
5. Click Create Product, and the system will create a product and return to the product list page.

Viewing Products

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **enterprise instance**.
2. Click on the enterprise instance to enter the instance details page. Click "**Product Development**" in the left menu to enter the product list. Select a created specific product and click on the product name to enter the following interface.



3. The most important information of the product will be displayed in the top area during the product development process. Click More Information, and the product information will be displayed on the right side of the screen, and you can initiate modification operations.

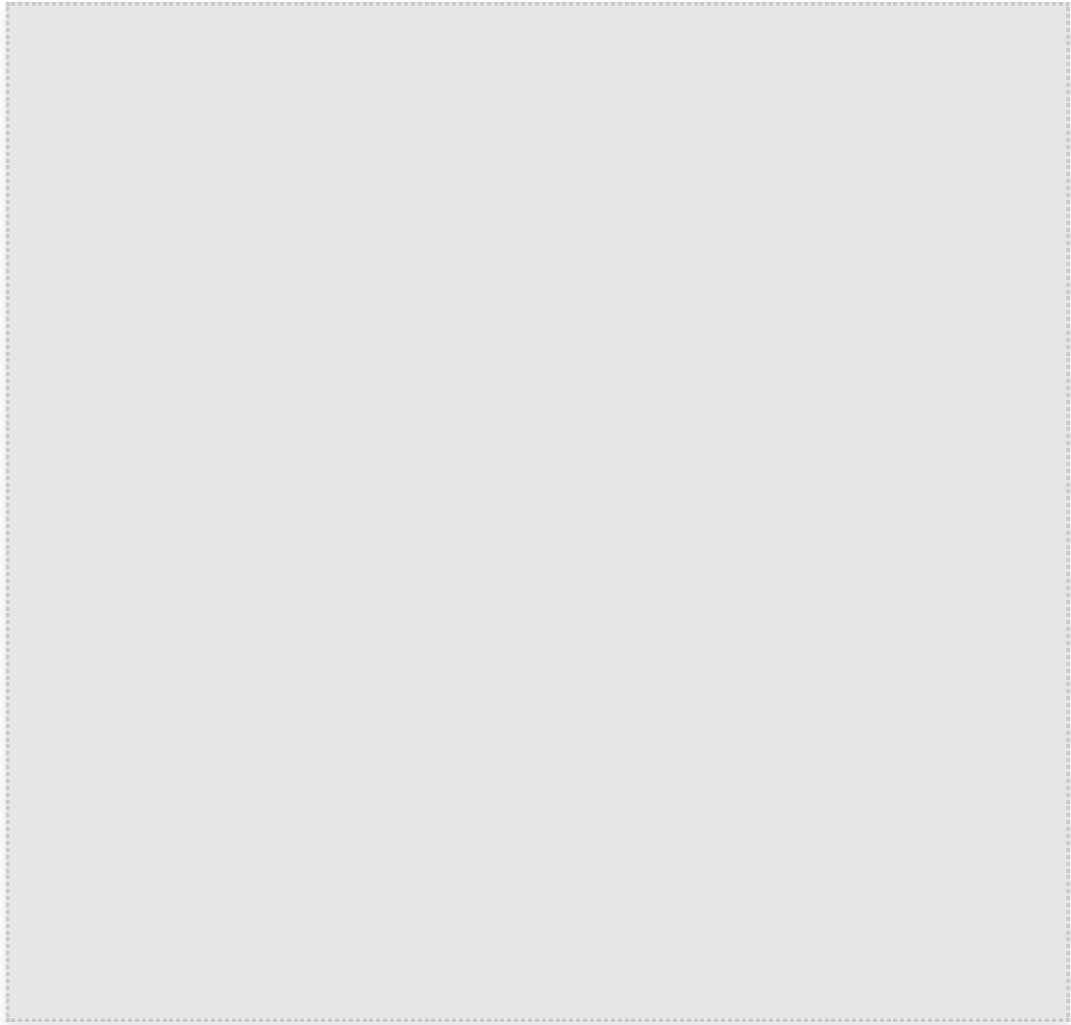


Modifying a Product

1. If a user wants to modify product information, click the "Modify Product Information" button. In the pop-up box, the product and remark can be modified.



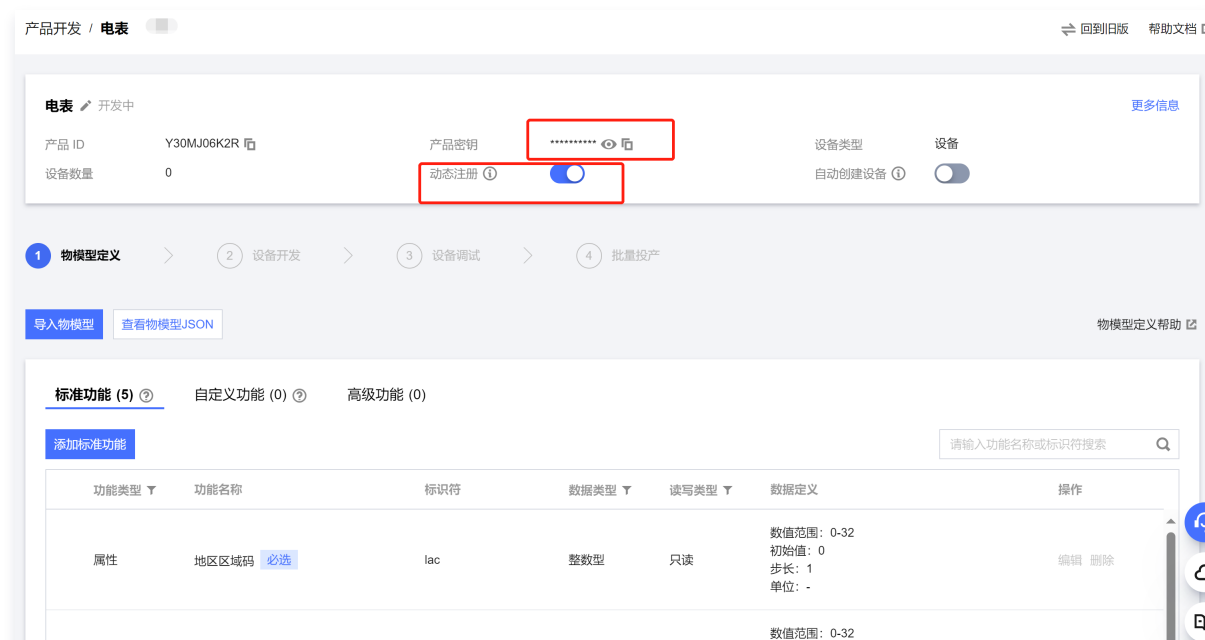
2. Modification of a product only allows modifying the product name and remark information. Other information cannot be modified. If other information must be modified, the product can be deleted and recreated.



Enabling Dynamic Registration

When users' devices are in mass production, they don't want to burn the triplets (i.e., product ID, device name, device key) generated by Tencent Cloud IoT Platform into the devices during the production line programming. Instead, they hope to use unified firmware and burn the same information to reduce mass production costs. Users can enable the "Dynamic Registration" option for the product. Once enabled, it means that the product supports devices dynamically generating device keys, i.e., using "[Product-Level Key Authentication](#)" method. How to enable dynamic registration? The detailed steps are as follows:

1. Select the products that need to enable "dynamic registration", click the "dynamic registration" switch in the figure below. Enabling this option means the product supports product-level key authentication.
2. Once the dynamic registration switch is enabled, the system will automatically generate a "product key". This product key will be burned into the device along with the product ID. For information on how the device calls the dynamic registration API, please refer to "[Dynamic Registration API Description](#)".



3. For some users' devices, when auto-creating devices, if it is hoped that the device side can randomly generate a regular device number for device number registration, then "**Auto-create Device**" can be re-enabled. Enabling automatic device creation means that the IoT platform allows the use of the DeviceName transmitted by the device to the platform to create a device.

产品开发 / 电表

回到旧版 帮助文档

电表 开发中

更多信息

产品 IDY30MJ06K2R

产品密钥

设备数量0

动态注册

设备类型

设备

自动创建设备

1 物模型定义

2 设备开发

3 设备调试

4 批量投产

导入物模型

查看物模型JSON

物模型定义帮助

标准功能 (5)

自定义功能 (0)

高级功能 (0)

添加标准功能

请输入功能名称或标识符搜索

| 功能类型 | 功能名称 | 标识符 | 数据类型 | 读写类型 | 数据定义 | 操作 |
|------|----------|-----|------|------|--|-------|
| 属性 | 地区区域码 必选 | lac | 整数型 | 只读 | 数值范围: 0-32 初始值: 0 步长: 1 单位: - | 编辑 删除 |

Product Development

Thing Model Definition

Last updated: 2025-04-27 17:33:48

Thing Model Overview

A Thing Model is a digital model defined by Tencent Cloud IoT for the same type of devices, i.e., products. It digitally describes products through three feature types: properties, events, and actions.

| Feature Type | Feature Description |
|--------------|---|
| Attribute | Various parameters and status data of devices during operation. For example, the temperature and humidity values collected by the Temperature and Humidity Sensor; the switch status values of electrical devices, the current speed and latitude and longitude values collected by vehicle-mounted devices. Properties support two types: read-write and read-only. Read-write means that the property can be reported to the platform from the device end and can also be controlled by initiating from the platform. Read-only means that the property is only reported from the device to the platform. |
| Event | Data initiated by the device during runtime that requires the User Business System to be aware of, including three event types: information, alarm, and fault. For example, when a device malfunctions during operation, it is required to send the fault information and related data at the time of failure to the platform. |
| Behavior | The calling method provided by the device for applications, used for the business to initiate requests. The device end needs to process and then return the processing result. Scenarios where the business system can synchronously or asynchronously obtain the results. For example: <ul style="list-style-type: none">• Parking lot users need the boom barrier to be opened for passage immediately after paying the parking fee. The business system needs to obtain the final status synchronization of whether the operation of the boom barrier is successful.• After a cloud printing scenario user issues a printing action, the input parameters of the action include the order ID, the content of the printed document, etc. The output parameters include the encoding of the print result. |

Thing Model Category

The features of the product include standard features and custom functions.

| Feature Category | Feature Description |
|------------------|---|
| Standard feature | Standard features refer to the Thing Models of some common device categories provided by Tencent Cloud IoT Platform. They are usable by users. Users can also choose not to use them as needed. |

| | |
|-------------------|--|
| Custom function | Custom function refers to the functionality provided by Tencent Cloud IoT Platform that allows users to freely define the Thing Model of products. Users can freely create, delete and edit Thing Model features according to the specifications and characteristics of the equipment. |
| Advanced Features | Advanced Features refer to the value-added service features provided by Tencent Cloud IoT for users. For example, the tencent real-time communication (TRTC) service can be applied to visual intercom, Cloud Broadcasting real-time shouting, one-to-many emergency call scenarios; Advanced Features generally automatically generate corresponding standard features. |

Operation Steps

Add Standard Features

Standard features of Tencent Cloud IoT, preset category, default Thing Model definition, as well as advanced features corresponding to the default Thing Model.

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **enterprise instance**.
2. Click on the enterprise instance, enter the instance page, click **Product Development** in the left menu, enter the product list, and choose a "product" to enter the **Thing Model Definition**.
3. Click **Add Standard Features** on the **standard feature** tab page.

添加标准功能

选择功能

已选择 (0) [全选删除](#)

灯通用类型其他产品品类

品类名称

开关(属性)
☐ 标识符: _mesh_generic_onoff 数据类型: 布尔型
0 - 关 1 - 开

色温(属性)
☐ 标识符: _mesh_light_color_temperature 数据类型: 整数型
数值范围: 800-20000 初始值: 800 步长: 1 单位: 开尔文

亮度(属性)
☐ 标识符: _mesh_light_lightness 数据类型: 整数型
数值范围: 0-65535 初始值: 0 步长: 1 单位: -

颜色(属性)
☐ 标识符: _mesh_light_hsl 数据类型: 数组

确定

取消

4. Other product categories are system-preset Thing Models at the same level as the category selected by the user. Users can add them as needed. Once confirmed, the selected Thing Model will be added to the "custom function".

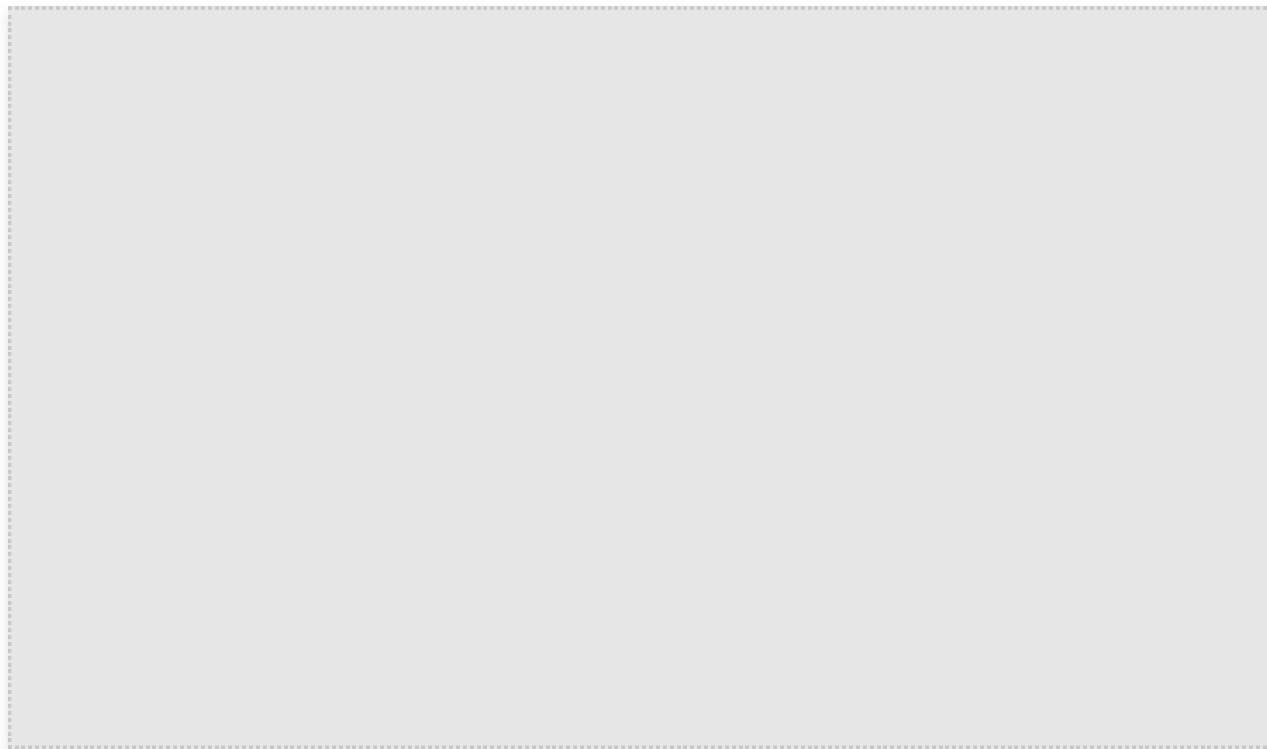
Custom Addition of Thing Model Attributes

Custom function allows users to freely define the Thing Model according to the specifications of the device, giving users full freedom to define.

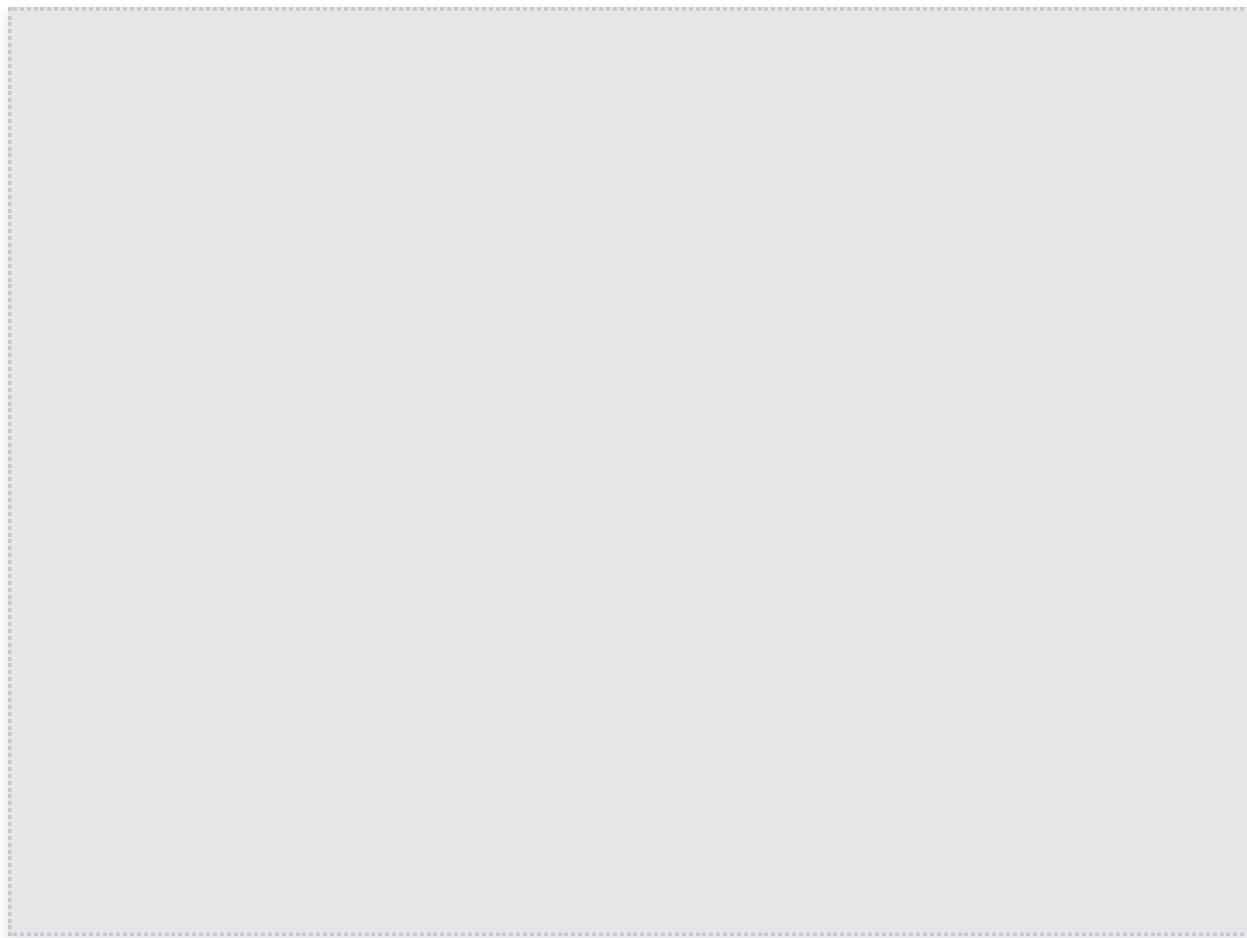
©2013–2025 Tencent Cloud. All rights reserved.

Page 84 of 334

1. Log in to the [IoT Explorer](#) , enter the instance list page, and select the generated **enterprise instance**.
2. Click on the enterprise instance, enter the instance page, click **Product Development** in the left menu, enter the product list, and choose a "product" to enter the **Thing Model Definition**.
3. Click **Add Custom Function** button on the **custom function** tag page.



4. In the popup window, select the "feature type" as needed. For example, if a lighting category product has an attribute switch, enter "Switch" for the feature name. The identifier must be unique across all Thing Models of the product. Select the data type and then save.



Custom Addition of Thing Model Events

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **enterprise instance**.
2. Click on the enterprise instance, enter the instance page, click **Product Development** in the left menu, enter the product list, and choose a "product" to enter the **Thing Model Definition**.
3. Click **Add Custom Function** button on the **custom function** tag page.
4. In the pop-up window, select "event" as needed, and select "event type" as "information" again. Users can freely define the event information that the device sends to the IoT platform.

新增自定义功能

注意：添加自定义功能将影响设备通过语音、中控面板控制，建议添加标准功能，若已有标准功能无法满足，您可提交 [意向单](#) 申请新增标准功能。

功能类型

属性

事件

行为

功能名称 *

Device_Status

支持中文、英文、数字、下划线的组合，最多不超过20个字符

标识符 *

status_report

第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符

事件类型

告警

故障

信息

事件参数

| 参数名称 | 参数标识符 | 数据类型 | 数据定义 | 操作 |
|---------------|---------|------|--|----|
| running_state | status | 布尔型 | <div>0 normal 1 fail</div> <div>支持中文、英文、数字、下划线的组合，最多不超过12个字符</div> | 删除 |
| Message | message | 字符串 | <div>- 64 + 字节</div> <div>请输入字符串长度限制，最大长度不超过2048个</div> | 删除 |
| 添加参数 | | | | |

描述

选填

最多不超过80个字符

保存

取消

Custom Addition of Thing Model Behavior

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **enterprise instance**.
2. Click on the enterprise instance, enter the instance page, click **Product Development** in the left menu, enter the product list, and choose a "product" to enter the **Thing Model Definition**.
3. Click **Add Custom Function** button on the **custom function** tag page.
4. In the pop-up window, select "**Behavior**" as needed. In self-service retail scenarios, shared payment scenarios, or parking scenarios, functions such as door opening, unlocking, and barrier opening are often required to be sent by the business system through the IoT platform immediately after payment. The business system needs to obtain the real-time processing results from the device side in real time. If there is no prompt response, the business system needs to cancel the transaction, for example, by initiating a refund.
5. The above feature scenes require entering the corresponding behavior names in the feature name entry. For example, "unlock", with the identifier "Open". Request parameters need to be defined by users. For example, a string type order ID can be defined so that the business system can track the processing situation of each order on the device side. Response parameters are the parameters reported to the IoT platform by the device side after the unlock operation is completed. Then, the order ID and the field code

of the final unlock status result of the device can also be added. If the unlock succeeds, the return code is 0; if it fails, the return code is 1. When the business system receives a return code of 1, it can either retry if permitted by the business, or initiate a refund operation after multiple retries fail.

❗ 注意：添加自定义功能将影响设备通过语音、中控面板控制，建议添加标准功能，若已有标准功能无法满足，您可提交 [意向单](#) 申请新增标准功能。

功能类型

属性

事件

行为

功能名称 *

开锁

支持中文、英文、数字、下划线的组合，最多不超过20个字符

标识符 *

Open

第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符

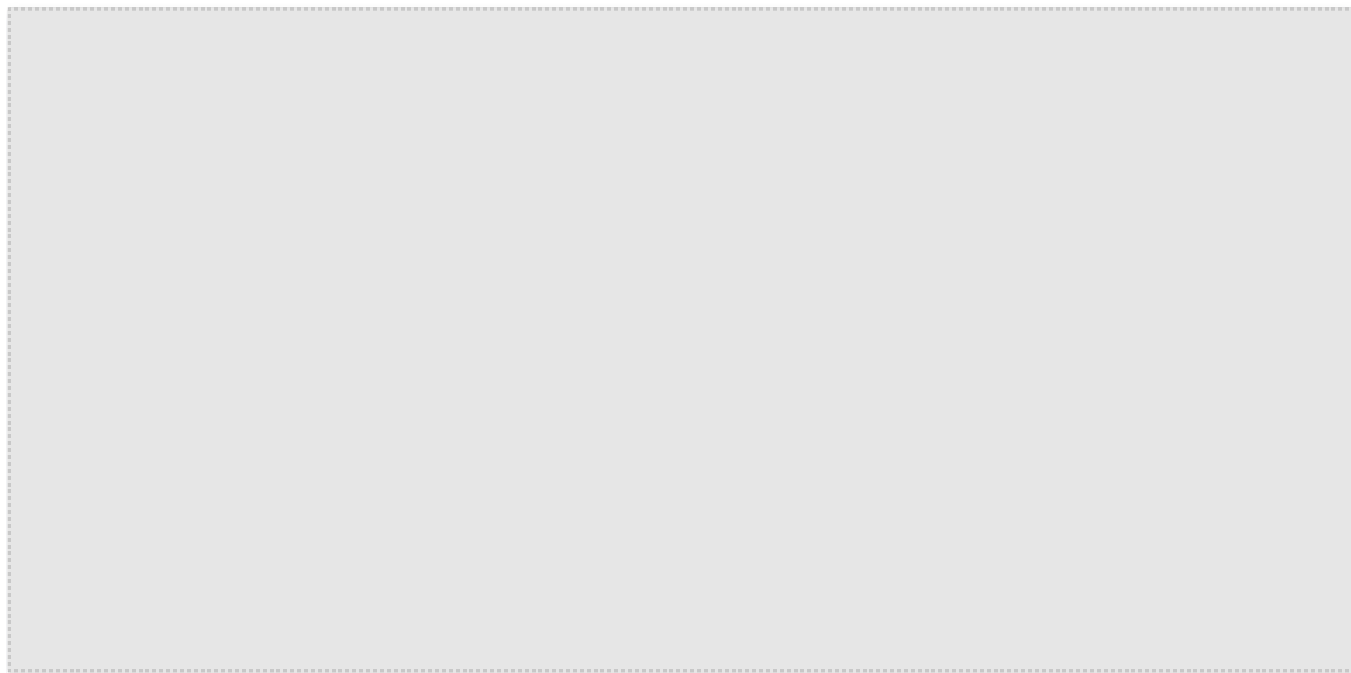
调用参数

| 参数名称 | 参数标识符 | 数据类型 | 数据定义 | 操作 |
|------|----------|------|--|----|
| 订单ID | Order_id | 字符串 | <div><div>-</div><div>2048</div><div>+</div><div>字节</div></div> <div>请输入字符串长度限制，最大长度不超过2048个</div> | 删除 |
| 添加参数 | | | | |

返回参数

| 参数名称 | 参数标识符 | 数据类型 | 数据定义 | 操作 |
|------|----------|------|--|----|
| 订单ID | Order_id | 字符串 | <div><div>-</div><div>2048</div><div>+</div><div>字节</div></div> <div>请输入字符串长度限制，最大长度不超过2048个</div> | 删除 |
| 状态 | code | 整数型 | <div>数值范围</div> <div><div>-</div><div>0</div><div>+</div></div> <div><div>-</div><div>100</div><div>+</div></div> <div>初始值</div> <div><div>-</div><div>0</div><div>+</div></div> | 删除 |

After the behavior function settings are completed, device behavior invocation can be implemented in the developer resources [API Explorer 3.0](#) provided by Tencent Cloud. When making an API call, fill in the corresponding ActionId and input parameters for testing.



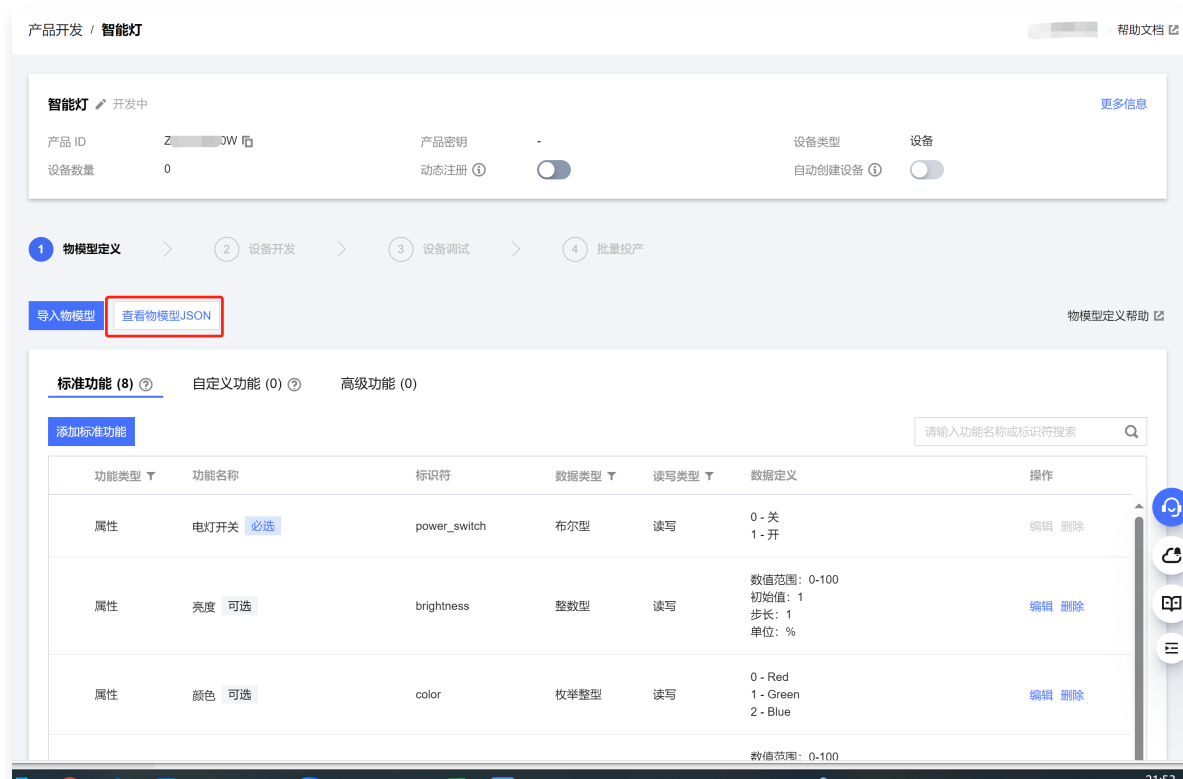
After receiving the Action message, the device cooperates to implement the corresponding Action. [C-SDK](#) provides automatic code generation for data templates and a response framework for attributes, events, and actions.

Advanced Functions of Thing Model

The advanced features of the Thing Model are value-added paid features of Tencent Cloud IoT Explorer. For example, in scenarios where devices and users need tencent real-time communication (TRTC), voice AI, and music content services, such as the tencent real-time communication (TRTC) service can be applied to visual intercom, Cloud Broadcasting real-time shouting, one-to-many emergency call scenarios. Advanced features generally automatically generate corresponding standard features. If you need to use them, please [submit an application](#) for business consultation. The staff will contact you regarding the service purchase matters.

View Thing Model JSON

1. After adding standard features and custom functions, users can click "View Thing Model JSON" to view the JSON format corresponding to the Thing Model.



2. Users can copy or download the JSON Thing Model. The copied JSON can be used with the import Thing Model feature to replicate the Thing Model definition between products, as shown below:



Import Thing Model

1. After copying the JSON of a specific product's Thing Model, you can use the "Import Thing Model" feature to copy the JSON into the textbox below.
2. Click the "Import" button will override the original product's Thing Model. If it is a mass-produced product, the Thing Model import feature needs to be used with caution.

导入物模型

注意：导入新的JSON后原产品的物模型将会被覆盖

您可以通过 JSON 对产品的物模型进行定义后导入平台，格式规范请 [查看文档](#)

请将要导入的物模型对应的JSON粘贴到此文本框

导入

取消

Thing Model Format Reference

The JSON description of the Thing Model for the smart light, including various data types and event types. An example JSON is as follows:

```
{
  "version": "1.0",
  "profile": {
    "ProductId": "2300UMK31M",
    "CategoryId": "3"
  },
  "properties": [
    {
      "id": "power_switch",
      "name": "light switch"
      "desc": "Control the on and off of the light"
      "required": true,
      "mode": "rw",
      "define": {
        "type": "bool",
        "mapping": {
          "0": "Off"
          "1": "On"
        }
      }
    }
  ]
}
```

```
},
{
  "id": "color",
  "name": "color"
  "desc": "Light color"
  "mode": "rw",
  "define": {
    "type": "enum",
    "mapping": {
      "0": "Red",
      "1": "Green",
      "2": "Blue"
    }
  }
},
{
  "id": "brightness",
  "name": "brightness"
  "desc": "Light brightness"
  "mode": "rw",
  "define": {
    "type": "int",
    "unit": "%",
    "step": "1",
    "min": "0",
    "max": "100",
    "start": "1"
  }
},
{
  "id": "name",
  "name": "light position name"
  "desc": "Light position name: study, living room"
  "mode": "rw",
  "required": false,
  "define": {
    "type": "string",
    "min": "0",
    "max": "64"
  }
}
],
"events": [
  {
    "id": "status_report",
    "name": "DeviceStatus",
    "desc": "Report the device status",
    "type": "info",
    "required": false,
    "params": [
```

```
{
  "id": "status",
  "name": "running_state",
  "desc": "Report current device running state",
  "define": {
    "type": "bool",
    "mapping": {
      "0": "normal",
      "1": "fault"
    }
  }
},
{
  "id": "message",
  "name": "Message",
  "desc": "Some extra message",
  "define": {
    "type": "string",
    "min": "0",
    "max": "64"
  }
}
],
{
  "id": "low_voltage",
  "name": "LowVoltage",
  "desc": "Alert for device voltage is low",
  "type": "alert",
  "required": false,
  "params": [
    {
      "id": "voltage",
      "name": "Voltage",
      "desc": "Current voltage",
      "define": {
        "type": "float",
        "unit": "V",
        "step": "1",
        "min": "0.0",
        "max": "24.0",
        "start": "1"
      }
    }
  ]
},
{
  "id": "hardware_fault",
  "name": "Hardware_fault",
  "desc": "Report hardware fault",
```

```
"type": "fault",
"required": false,
"params": [
  {
    "id": "name",
    "name": "Name",
    "desc": "Name like: memory,tf card, sensors ...",
    "define": {
      "type": "string",
      "min": "0",
      "max": "64"
    }
  },
  {
    "id": "error_code",
    "name": "Error_Code",
    "desc": "Error code for fault",
    "define": {
      "type": "int",
      "unit": "",
      "step": "1",
      "min": "0",
      "max": "2000",
      "start": "1"
    }
  }
]
},
"actions": [
  {
    "id": "unlock",
    "name": "Behavior detection for turning on the light"
    "desc": "Action for turning on the light"
    "input": [
      {
        "id": "open",
        "name": "switch"
        "define": {
          "type": "bool",
          "mapping": {
            "0": "Off"
            "1": "Open"
          }
        }
      }
    ],
    {
      "id": "user",
      "name": "User"
      "define": {
```

```
        "type": "string",
        "min": "0",
        "max": "2048"
      }
    },
    ],
    "output": [
      {
        "id": "user",
        User
        "define": {
          "type": "string",
          "min": "0",
          "max": "2048"
        }
      },
      {
        "id": "time",
        "name": "Light on time"
        "define": {
          "type": "timestamp"
        }
      },
    ],
    {
      "id": "state",
      "name": "light status"
      "define": {
        "type": "bool",
        "mapping": {
          "0": "Off"
          "1": "On"
        }
      }
    }
  ],
  "required": false
}
]
```

References

To understand the Thing Model Protocol, refer to [Thing Model Protocol](#).

Device Development

Last updated: 2025-04-27 17:34:05

Device Development Overview

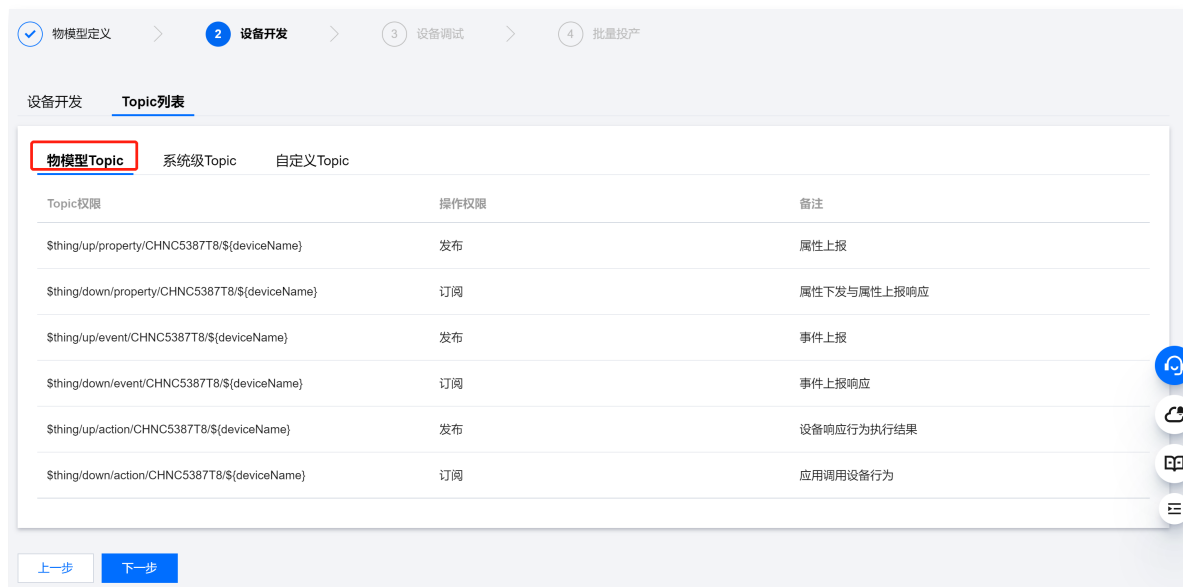
After the user completes the definition of the Thing Model, the device needs to be connected to the platform as required by the access protocol. This document primarily introduces how to use the development platform for device development.

- Device development can also support user management topics and can be freely defined.
- Equipment development supports cloud parsing feature. When using MQTT Custom Transparent Transmission Protocol, if you hope to parse it into Thing Model JSON format in the cloud, you can enable cloud parsing.

Topic Management

Thing Model Topic

The Thing Model Topic is an automatically generated Topic when the platform uses the data protocol for Thing Model products.



The screenshot shows the 'Device Development' page in IoT Explorer. The breadcrumb navigation is: 物模型定义 > 2 设备开发 > 3 设备调试 > 4 批量投产. The '设备开发' (Device Development) section is active, and the 'Topic列表' (Topic List) tab is selected. The table displays the following topics:

| Topic权限 | 操作权限 | 备注 |
|---|------|-------------|
| \$thing/up/property/CHNC5387T8/\${deviceName} | 发布 | 属性上报 |
| \$thing/down/property/CHNC5387T8/\${deviceName} | 订阅 | 属性下发与属性上报响应 |
| \$thing/up/event/CHNC5387T8/\${deviceName} | 发布 | 事件上报 |
| \$thing/down/event/CHNC5387T8/\${deviceName} | 订阅 | 事件上报响应 |
| \$thing/up/action/CHNC5387T8/\${deviceName} | 发布 | 设备响应行为执行结果 |
| \$thing/down/action/CHNC5387T8/\${deviceName} | 订阅 | 应用调用设备行为 |

At the bottom of the page, there are buttons for '上一步' (Previous Step) and '下一步' (Next Step).

System-Level Topic

System-level Topic is an automatically generated Topic for each product by the platform, typically a system standard feature such as OTA.

设备开发 **Topic列表**

| 物模型Topic | 系统级Topic | 自定义Topic |
|---|----------|------------------------------|
| Topic权限 | 操作权限 | 备注 |
| \$ota/report/CHNC5387T8/\$(deviceName) | 发布 | 固件升级消息上行 |
| \$ota/update/CHNC5387T8/\$(deviceName) | 订阅 | 固件升级消息下行 |
| \$broadcast/rxd/CHNC5387T8/\$(deviceName) | 订阅 | 广播消息下行 |
| \$shadow/operation/CHNC5387T8/\$(deviceName) | 发布 | 设备影子消息上行 |
| \$shadow/operation/result/CHNC5387T8/\$(deviceName) | 订阅 | 设备影子消息下行 |
| \$rpc/rxd/CHNC5387T8/\$(deviceName)/\$(MessageId) | 发布 | RRPC消息上行, MessageId为RRPC消息ID |
| \$rpc/rxd/CHNC5387T8/\$(deviceName)/* | 订阅 | RRPC消息下行 |
| \$sys/operation/CHNC5387T8/\$(deviceName) | 发布 | 系统topic: ntp服务消息上行 |
| \$sys/operation/result/CHNC5387T8/\$(deviceName)/* | 订阅 | 系统topic: ntp服务消息下行 |

Custom Topic

Custom Topic is a management feature provided by the platform for users to customize. It is usually applied when users need to freely define the message transmission format and Topic name between devices and the platform.

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **enterprise instance**.
2. Click the enterprise instance, enter it, and select **Product Development** in the left menu. Click the target product in the product list. By default, enter **Thing Model Definition**.
3. Select **device development**, click **Topic List > Custom Topic**.
4. The system has generated 3 custom topics for the products created for the user by default. If users need to define a new topic, click the "Add Custom Topic" button.

设备开发 **Topic列表**

| 物模型Topic | 系统级Topic | 自定义Topic |
|-----------------------------------|----------|---------------------------------------|
| Topic权限 | 操作权限 | 操作 |
| CHNC5387T8/\$(deviceName)/control | 订阅 | 编辑 删除 |
| CHNC5387T8/\$(deviceName)/data | 订阅和发布 | 编辑 删除 |
| CHNC5387T8/\$(deviceName)/event | 发布 | 编辑 删除 |

添加自定义Topic

上一步 下一步

5. On the interface below, input the Topic name, set the permission for the Topic, and click **Confirm** to create the Topic.

添加Topic权限

操作名称 *

请输入操作名称

名称命名支持字母、数字、下划线组合；不同层级之间用 / 分层
+表示一级，使用 /+命名，不能 /+aaa/；长度限制为1-64位

操作权限

发布

确定

取消

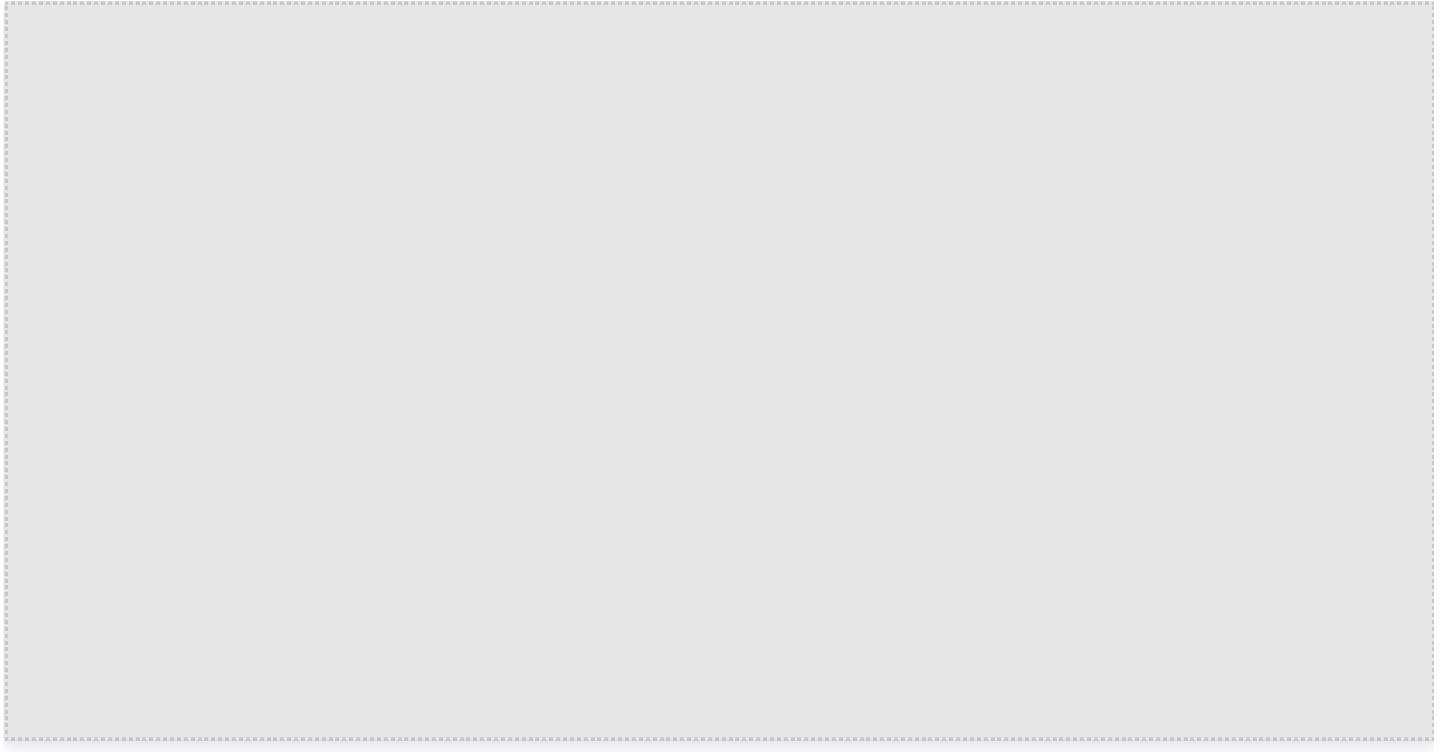
Operation permissions include "publish", "subscribe", and "subscription and publication".

- Publish: It means that the device end of this Topic can only publish and has no permission to subscribe. The Topic with publishing permission is generally for the device end to send data to the platform. If a device subscribes to a Topic with only "publish" permission, it will fail to subscribe, and the cloud diagnostic logs will also prompt "Subscribe no permission (to be refined)".
- Subscribe: It means that the device end of this Topic can subscribe. The Internet of Things Platform will deliver messages to the device through this Topic.
- Publish and subscribe: It means that the Topic simultaneously possesses publish and subscribe permissions.

Device Development

After the user creates a product and defines the Thing Model of the product, click **Device Development**. You can access the platform through the multi-language SDK provided by the platform or modules, DTUs, edge gateways that have been integrated with Tencent Cloud IoT Explorer. Currently, three development methods are provided.

- Develop based on modules: Meet the scenarios where the MCU communicates via the serial port and communicates with the cloud through modules.
- Develop based on SDK: Meet the access scenario of directly integrating the C SDK.
- Develop based on OS: Meet the access scenario of integrating the C SDK based on the Internet of Things operating system.



Developing Based on Modules

1. If your device needs to connect to the development platform through a communication module, click **Module-based Development**.
2. The system displays a module selection window. You need to choose an appropriate communication module based on your business requirements. This includes the module brand and the module

communication type. After choosing an appropriate module, you can click **Confirm**.



3. After selecting a module, you can click **Reselect** to replace the module. You can also click **View Details** to learn about the detailed parameters of the module. Additionally, you can click **Procurement Consultation** to purchase from a module company.



4. Embedded development.

- For devices connected via modules, if a data template is defined, the platform provides the functionality of automatically generating MCU SDK code, which is used to speed up how the MCU connects to the communication module.
- Click **MCU SDK code**. The development platform will generate a compressed file. After you download it, you can just follow the development guide to connect the device to the development platform.

- How to perform MCU development based on the downloaded MCU SDK code. For details, see related document in [Device Development Guide](#).

2. 嵌入式开发

 您可以通过平台自动生成的MCU SDK代码进行开发，也可以通过腾讯云IoT AT指令协议自行开发，开发完后进入设备调试



自动生成MCU SDK代码

MCU SDK代码



腾讯云IoT AT指令协议

腾讯云IoT AT指令集

腾讯云IoT AT指令集-WiFi-ESP8266



开发指引

设备端开发指南

[上一步](#) [下一步](#)

5. Download the AT Command Protocol.

Click [Tencent Cloud IoT AT Command Protocol](#) to learn about the Tencent Cloud IoT AT Command Protocol.

Developing Based on OS


1. If the IoT operating system running on your device is an OS type that has been integrated with the SDK, you can click **Development Based on OS** to view the development guide for integrating with the corresponding IoT operating system access platform.
2. Embedded development.
 - Data template configuration file generation: If the data template and events of the defined product have been created, based on the guide documentation, you can understand how to generate template code from the data template, and how to perform business logic development based on the generated data template configuration file as well as data template examples.
 - OS code download: Provide the download paths of Tencent IoT Terminal Operating System (TencentOS tiny) and RT-Thread.
 - Development guide: Provide development guides for integrating with Tencent Cloud IoT development platform based on different IoT operating systems.

1 物模型 > 2 设备开发 > 3 交互开发 > 4 设备调试 > 5 批量投产

设备开发

Topic列表

嵌入式开发

 根据您选择的OS类型，查看对应的开发指南进行设备端开发，开发完成后您可以进行设备调试



数据模板配置文件生成

[数据模板代码生成指南](#)
[数据模板应用开发指南](#)



OS代码下载

[TencentOS tiny 代码下载](#)
[RT-Thread 代码下载](#)



开发指引

[基于 TencentOS tiny 开发指南](#)
[基于 RT-Thread 开发指南](#)
[基于 Linux 开发指南](#)
[基于 Windows 开发指南](#)
[基于 FreeRTOS 开发指南](#)
[基于其他 OS 开发指南](#)

上一步

下一步

Device Debugging

Last updated: 2025-04-27 17:34:24

Device Debugging Overview

After the device side completes the integration development with the platform or connects to the platform using the MQTT client tool, it can view the device uplink and downlink business data, online and offline status, and device trace logs during the development and mass production stages through the device debugging feature to support users in locating and analyzing and resolving problems. This documentation introduces how to perform device debugging.

Operation Steps

Create New Device

Perform device debugging. First, create a device.

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **enterprise instance**.
2. Click the enterprise instance, enter the instance internal page, click **Product Development** in the left menu, enter the product list, and choose a "product" to enter the third step **Device Debugging**.
3. Click **Create New Device** to enter the create device page.



4. Enter the device name (the device name under each product is unique. It is recommended to use English or numbers. Chinese characters are not allowed). Click save to create the device.

新建设备

所属产品 电表

设备名称 *

支持英文、数字、下划线的组合，最多不超过48个字符

保存

取消

5. After successful creation, the successfully created device will be viewed on the **Device Debugging** list page. The default status of the first created device is **"Inactive"**.

产品开发 / 电表 帮助文档

电表 开发中 更多信息

产品 ID

产品密钥

设备类型 设备

设备数量 1

动态注册 ☒

自动创建设备 ☐

物模型定义

设备开发

3 设备调试

4 批量投产

设备调试提供真实调试功能，便于测试设备上报、接收数据是否正常，可创建测试设备后进行调试

新建设备

设备名称

输入设备名称搜索

| 设备名称 | 状态 | 激活时间 | 最后上线时间 | 操作 |
|--------|-----|------|--------|---------------------------------------|
| dev001 | 未激活 | - | - | 调试 删除 |

🔊

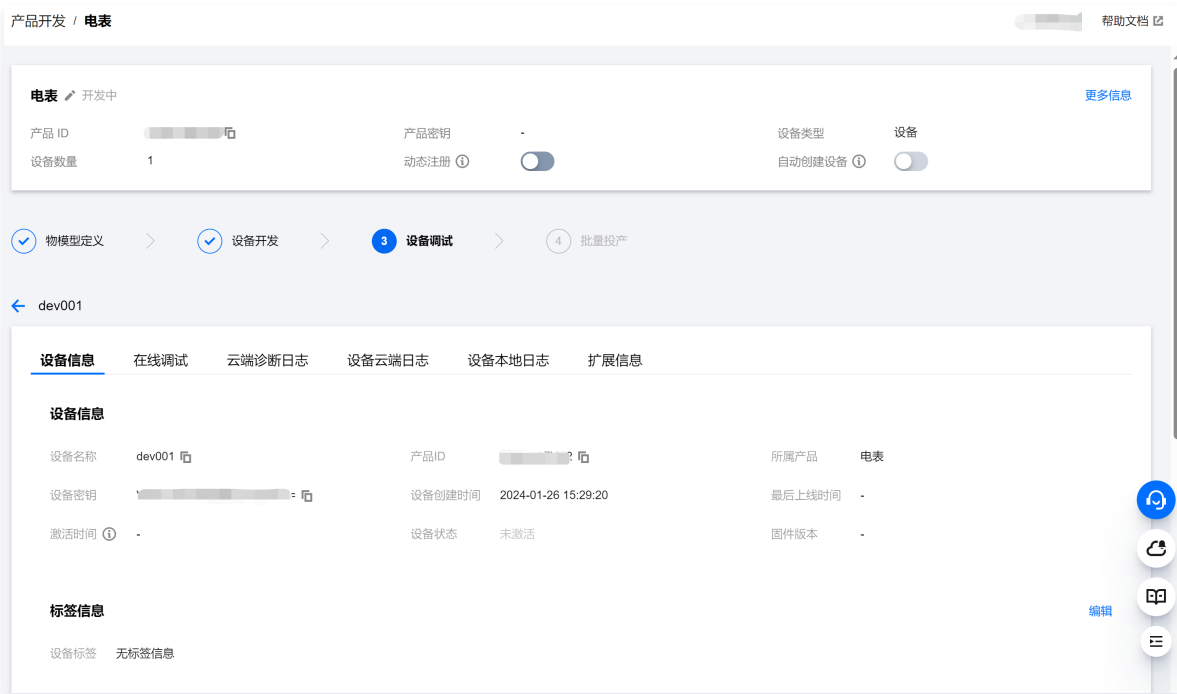
🔗

📄

☰

View Device Information

1. Upon success of creating a new device, it is required to query device information, obtain important parameters and perform device debugging.
2. Click the device name in the device list to view the basic device information.



3. Once the device view is opened, the basic device information can be viewed.

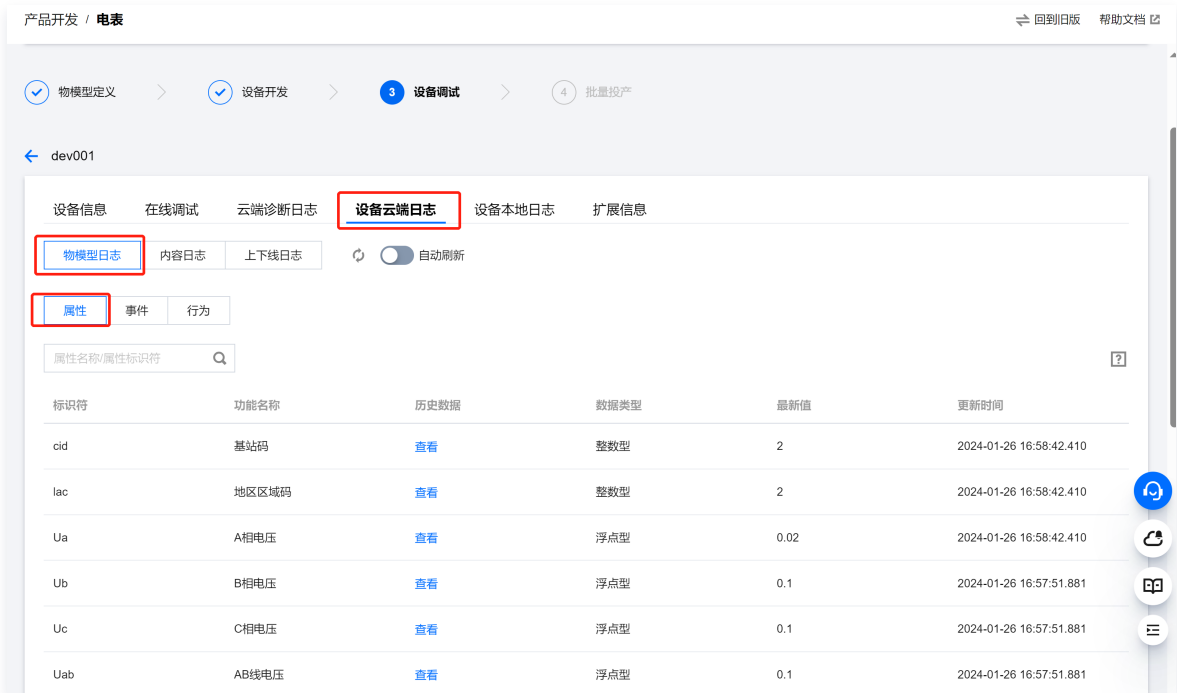
- Device name: A unique device ID under the product, which generally needs to be burned into the device.
- Product ID: the product ID to which the device belongs, which generally needs to be burned into the device.
- Associated product: the name of the product to which the device belongs.
- Device key: a randomly generated key for each device by the platform. To authenticate using the key, this information needs to be burned into the device.
- Creation time of the device: the time when the device was initially created successfully.
- Last online time: the time when the device last connected to the connection platform.
- Activation time: the time when the device successfully connects to the platform for the first time.
- Device status: the current device status. If the device successfully connects to the platform via MQTT, it displays "online". If the device is offline, it displays "offline". If the device has never connected to the platform, it displays "inactive".

4. Below the basic device information, tags can be set, and whether to enable the local log of the device can also be configured. The automatically generated "device connection parameters" can be copied, and ClientId, UserName, and Password can be directly copied into MQTT client tools such as MQTT.fx for quick simulation of device access to the platform.



View Thing Model Logs

1. When the device is successfully connected to the platform and publishes messages to the Thing Model Topic, the submitted attributes, events, acts and other data of the device can be viewed under "Cloud Log of the Device" > "Thing Model Log".

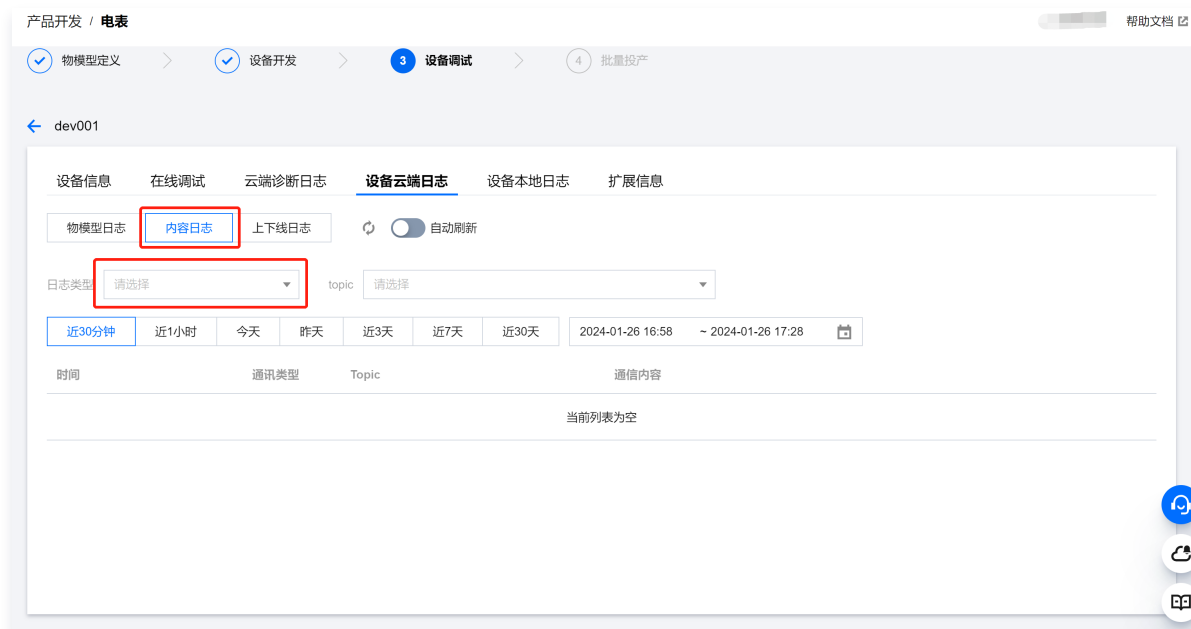


2. List all the attribute functions of the Thing Model of this device in the list.
- Identifier: the identifier in the Thing Model corresponding to the device.
 - Feature name: the "feature name" in the corresponding data template.
 - Historical data: click **View** to retrieve the historical data reporting of this feature item.

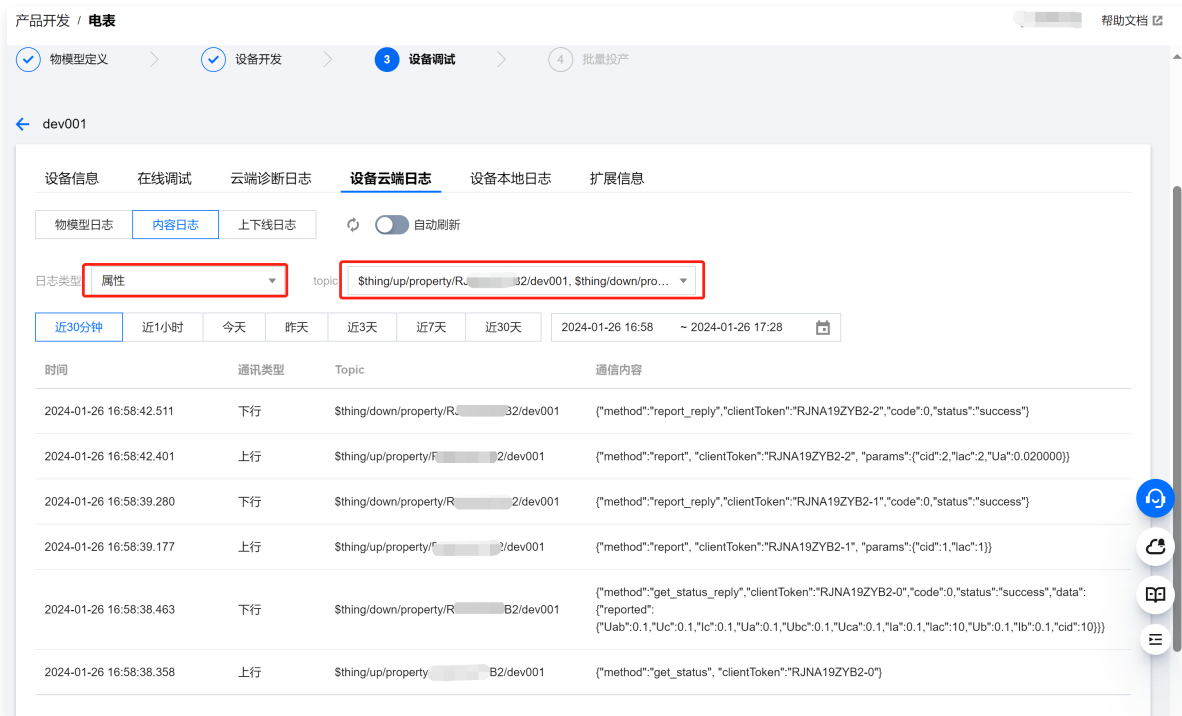
- Data type: the "data type" of the functionality in the Thing Model.
 - Latest value: When a device reports data to the cloud, as long as the latest reported value of a specific feature changes, the Latest Value column will immediately display the latest value reported by the device.
 - Update time: refers to the time when the latest value changes. It is generally the occurrence time of the device reporting the feature.
3. View the historical report data of a specific feature. Display the historical data reported by the feature to the cloud in chronological order and verify whether the reported data is correct.
 4. When the device reports data in the format of the Thing Model Protocol, if the latest value cannot be viewed in the Thing Model Log, you need to confirm whether the format of the reported data is correct. For details, see [Thing Model Protocol](#) and [Common Issues of Thing Model](#).

View Content Log

1. Content logs provide users with the feature to query uplink and downlink content logs of devices by Topic. Users select "Content Log", and the "Log Type" drop-down list will be shown.

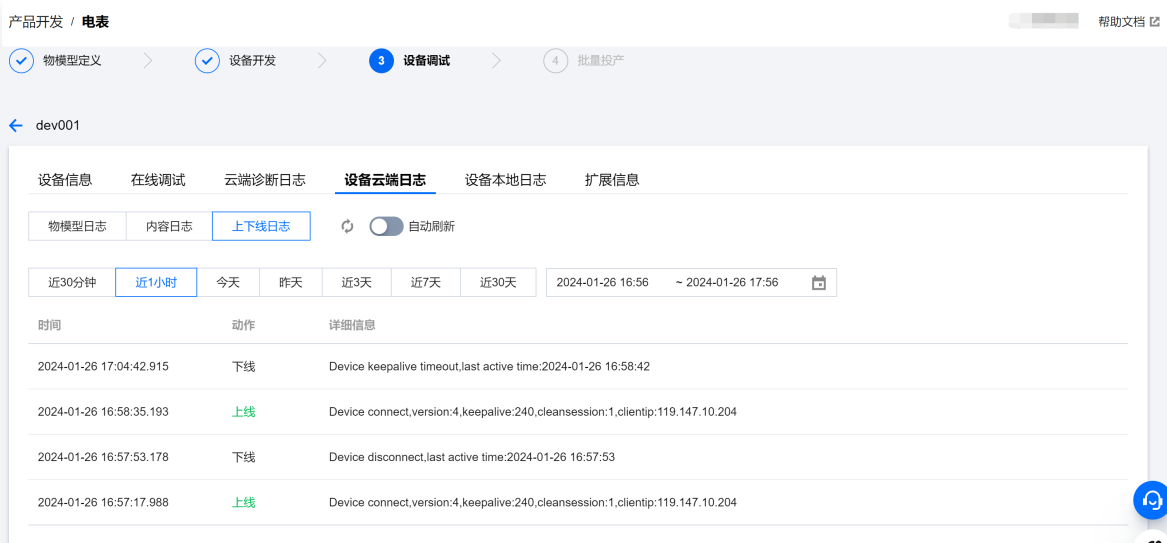


2. Select "Attribute" as the log type. The drop-down list of Topic will automatically load the Topic corresponding to the attribute, and retrieve all upstream and downstream attribute content logs of the device and platform within the selected date range. Users can select different types as needed, and retrieve the upstream and downstream content data corresponding to different Topics for device debugging and problem localization.



View Online and Offline Logs

1. Online and offline logs provide users with the feature to query logs of devices connecting to the platform (going online) and devices actively or passively disconnecting from the platform.



2. Device disconnect means the device actively closes the connection with the platform. Device keepalive timeout indicates that the device does not send heartbeat, causing the connection to close due to timeout.

View Cloud Diagnostic Logs

The cloud diagnostic logs feature is used to view the end-to-end trace logs of interactions between devices and the cloud, helping users quickly diagnose exceptional errors that occur with devices during the debugging process, such as permission denied for subscribing to topics, failed to release uplink messages, release failure of rule engine forwarding to third-party services, etc., for locating exceptional events. This

document is used to locate the cause of communication content between devices and cloud messages and seek a solution.

Device Online and Offline Related

| content | result | errcode | Description |
|----------------|--------------------|---------------------|--|
| Device connect | FAIL, system error | Dev_Conn_System_Err | Device going online fails. System error. |

Publishing and Subscribing to Topics Related

| content | result | errcode | Description |
|------------------------------|------------------------------|------------------------------|---|
| Device subscribe topic: {} | FAIL, unauthorized operation | Subscribe_Topic_Unauthorized | Failed to subscribe. No permission to subscribe to the topic. |
| Device subscribe topic: {} | FAIL, system error | Subscribe_System_Err | Failed to subscribe. System error. |
| Device unsubscribe topic: {} | FAIL, system error | UnSubscribe_System_Err | Unsubscribe failed. System error. |

Device Message Related

| content | result | errcode | errcode Definition |
|---|---------------------------------|-------------------------------------|--|
| Device publish message to topic:{}, QOS:{} | FAIL, unauthorized operation | Dev_Pub_Unauthorized | Publication failed. No permission to release to the Topic. |
| | FAIL, reach max limit with {} | Dev_Pub_Reach_Max_Limit | Publication failed. publish exceeds the frequency limit. |
| | FAIL, payload too long({} > {}) | Dev_Pub_Payload_TooLong | Publication failed. payload exceeds the length limit. |
| | FAIL, system error | Dev_Pub_System_Err | Publication failed. System error. |
| Publish message to device: topic: {}, QOS:{} {Error description of the business} | FAIL,no subscriber | Pub_To_Dev_No_Subscriber | Sending failure. No subscribers. |
| | FAIL, too many offline message | Pub_To_Dev_Offline_Msg_Exceed_Limit | Sending failure. Offline message storage is full. |

| | | | |
|--|--|------------------------------|--|
| | FAIL, offline message payload exceed limit | Pub_To_Dev_Payload_Too_Large | Sending failure. payload size exceeds the limit. |
| | FAIL, system error | Pub_To_Dev_System_Err | Sending failure. System error. |

Rule Engine Related

| content | result | errcode | Definition of errcode |
|---|---|--------------------------------|--|
| Send message to RuleEngine, topic:{} | FAIL, system error | Msg_Send_To_Rule_System_Err | Forwarding failure. System error. |
| MQ:forward CMQ, type: {}, CMQ{queue: {}, region: {}} | FAIL, queue name is not existed, or deleted | MQ_Queue_NotExist_Or_Deleted | Forwarding failure. Queue does not exist, or the queue has been deleted. |
| | FAIL, exceed maximum message size | MQ_Exceed_Max_Msg_Size | Forwarding failure. At least one message has reached the maximum message size limit. |
| | FAIL, reach maximum retention number of message | MQ_Reach_Max_Retention_Number | Forwarding failure. The maximum message backlog count of the queue has been reached. |
| | FAIL, unexpected error: {} | MQ_Forward_CMQ_Unexpected_Err | Forwarding failure. Public error of Tencent Cloud API. |
| | FAIL, system error | MQ_Forward_CMQ_System_Err | Forwarding failure. System error. |
| MQ forward CKafka, type:{}, Ckafka{instance: {}, topic: {}, region: {}} | FAIL, unexpected error: {} | MQ_Foward_Kafka_Unexpected_Err | Forwarding failure. Kafka error information. |
| | FAIL, system error | MQ_Foward_Kafka_System_Err | Forwarding failure. System |

| | | | |
|--|------------------------------------|--|--|
| | | | error. |
| RuleEngine republish message, source topic: {}, destination topic: {} | FAIL, no such field({}) in payload | Payload_No_Field | Forwarding failure. The payload has no corresponding field. |
| | FAIL, system error | Rule_Repub_System_Err | Forwarding failure. System error. |
| RuleEngine forward third-party server,topic: {}, url: {} | FAIL, url server timeout | Forward_Third_Not_Responding | Forwarding failure. Third-party server has no response. |
| | FAIL, system error | Forward_Third_System_Err | Forwarding failure. System error. |
| RuleEngine forward CKafka, topic:{}, Ckafka.instance:{}, topic: {}, region: {},retry times:{}" | FAIL,unexpected error: {} | Rule_Forward_Kafka_Unexpected_Err | Forwarding failure. Kafka error information. |
| RuleEngine forward CMQ Topic, IOT topic:{}, CMQ.topic:{}, region: {} | FAIL,system error: {} | MQ_Queue_NotExist_Or_Deleted | Forwarding failure. Queue does not exist, or it has been deleted. |
| | | MQ_Exceed_Max_Msg_Size | Forwarding failure. Message exceeds the maximum limit. |
| | | MQ_Reach_Max_Retention_Number | Forwarding failure. The maximum retention number has been reached. |
| | | Rule_Forward_CMQ_Topic_No_Subscription | Forwarding failure. This topic has no subscribers. |
| | | Rule_Forward_CMQ_Topic_Unexpected_Err | Forwarding failure. Unknown error. |

| | | | |
|--|-----------------------------|--|--|
| RuleEngine forward CMQ Queue, IOT topic: {}, CMQ.topic:{}, region:{} | FAIL,system error:{} | MQ_Queue_NotExist_Or_Deleted | Forwarding failure. Queue does not exist, or it has been deleted. |
| | | MQ_Exceed_Max_Msg_Size | Forwarding failure. Message exceeds the maximum limit. |
| | | MQ_Reach_Max_Retention_Number | Forwarding failure. The maximum retention number has been reached. |
| | | Rule_Forward_CMQ_Topic_No_Subscription | Forwarding failure. This topic has no subscribers. |
| | | Rule_Forward_CMQ_Topic_Unexpected_Err | Forwarding failure. Unknown error. |
| RuleEngine forward CTSDB. topic:{}, CTSDB.instanceid:{}, region:{}, metric:{} | FAIL,system error:{} | Rule_Forward_Ctsdb_System_Err | Forwarding failure. System error. |
| RuleEngine forward Mongo, topic:{}, Mongo.instanceid:{}, database:{}, collection:{} | FAIL,system error:{} | Rule_Forward_MongoDB_System_Err | Forwarding failure. System error. |
| RuleEngine forward TCB Func,get role error topic: {}, TCB.envID:{}, functionName:{} | FAIL,system error:{} | Rule_Forward_TCB_Func_System_err | Forwarding failure. System error. |
| RuleEngine forward Mysql.IOT topic:{}, info:{} | FAIL,system error:{} | Rule_Forward_Mysql_System_Err | Forwarding failure. System error. |
| RuleEngine forward TDMQ.IOT topic:{}, info: {} | FAIL,system error:{} | Rule_Forward_TDMQ_System_Err | Forwarding failure. System error. |
| Process sql: {} | FAIL,unexpected error:{} | Rule_Process_SQL_Err | Execution failure. |
| Payload is not JSON fmt | FAIL | Payload_Not_JSON | The message payload is not in |

JSON format.

Online Debugging

1. After your real device has been successfully connected to the development platform, you can use online debugging to test the data transmission and reception of the real device.
2. Click **Online Debugging** to enter the online debugging feature. The premise is that the real device is enabled and successfully connected to the development platform.
3. The control panel on the left-side of Online Debugging is automatically generated according to the Thing Model of the product to which the device belongs. After setting the data to be sent, click **Send**, and the system will automatically trigger control instructions to the device.
4. After the device receives the command, it will immediately return data to the cloud and display it in the textbox on the right.
5. If users want to debug with real devices by issuing control commands through APIs, they can refer to [Remote Control of Equipment](#) and [Equipment Transparent Transmission Instruction Control](#). One is to issue control commands via the Thing Model Protocol, and the other is to issue commands in a customized Payload method.

Batch Production

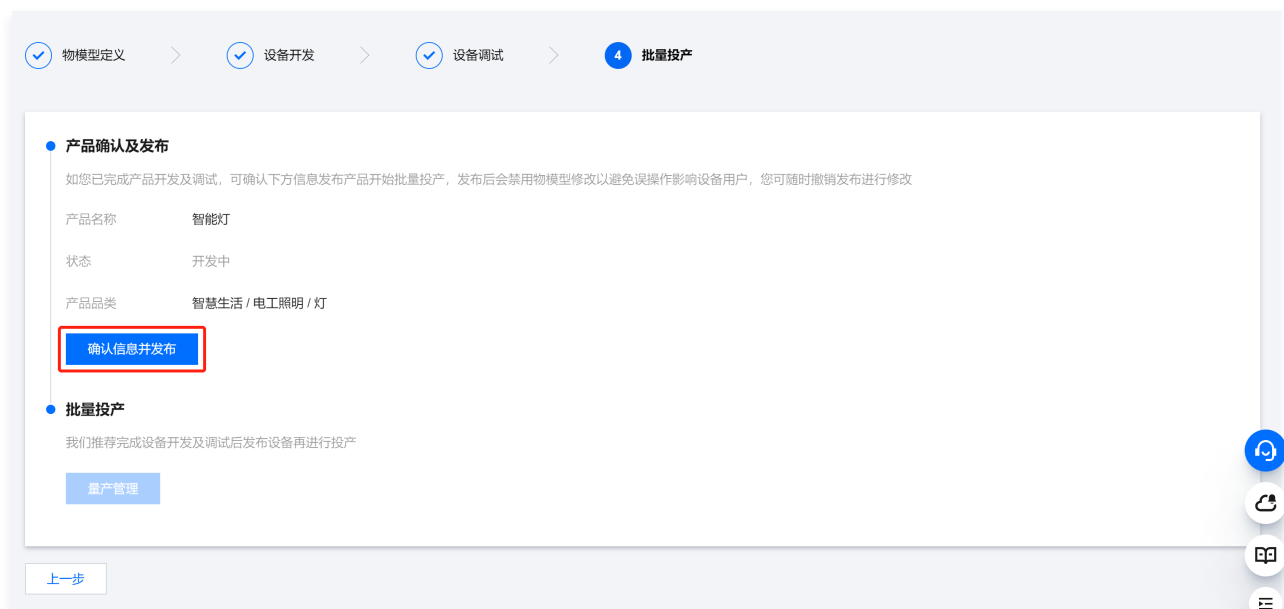
Last updated: 2025-04-27 17:34:38

Introduction to Batch Production

Normally, IoT solutions include device-side development as well as business system development. The development stage is also divided into at least the development and debugging stage and the mass production stage. More rigorous users may conduct multiple rounds of small-batch user verification before mass production to ensure the reliability of the overall solution. After completing the development work of device-side integration with the platform and the corresponding development of the business-side with the platform and completing the overall testing process, the corresponding products will enter the batch production stage. At this point, the platform can be used to change the status of the products to "released". For products in the "released" status, it is not allowed to modify the Thing Model or related configuration details of the product, to prevent some important data from being modified after mass production, which may cause exceptions in mass-produced devices.

Operation Steps

1. Log in to the [IoT Explorer](#), enter the instance list page, and select the generated **enterprise instance**.
2. Click the enterprise instance to enter its internal page. Click **Product Development** in the left menu to enter the product list.
3. Select developed products and enter step 4 **batch production**.



4. Click the **Confirm Information and Publish** button. After secondary confirmation, the product status will be modified to "Released".

产品开发 / 智能灯 回到旧版 帮助文档

智能灯 已发布 更多信息

产品 ID

CI-*****

产品密钥

设备类型

设备

设备数量

8

动态注册 ①

自动创建设备 ①

物模型定义

设备开发

设备调试

4 批量投产

产品确认及发布

如您已完成产品开发及调试，可确认下方信息发布产品开始批量投产，发布后会禁用物模型修改以避免误操作影响设备用户，您可随时撤销发布进行修改

产品名称

智能灯

状态

已发布

产品品类

智慧生活 / 电工照明 / 灯

撤销发布

批量投产

产品已发布，可进行量产

量产管理

5. Once released, the Thing Model of the product cannot be modified. You can click "Cancel Publishing". Once the product status is "Under Development", you can continue to modify the Thing Model.

Note:

Released products cannot modify the Thing Model, set dynamic registration, or modify product information of the product. If necessary to modify the Thing Model, you can cancel publishing, and the product status will be changed to "under development".

Attention

Is batch production mandatory? Can we skip the release and keep the product status as under development without affecting the business? However, when multiple accounts manage a product simultaneously, users may modify the Thing Model or product-related configurations, causing exceptions in data sending and receiving for mass-produced devices.

Device management

Last updated: 2025-04-27 17:34:54

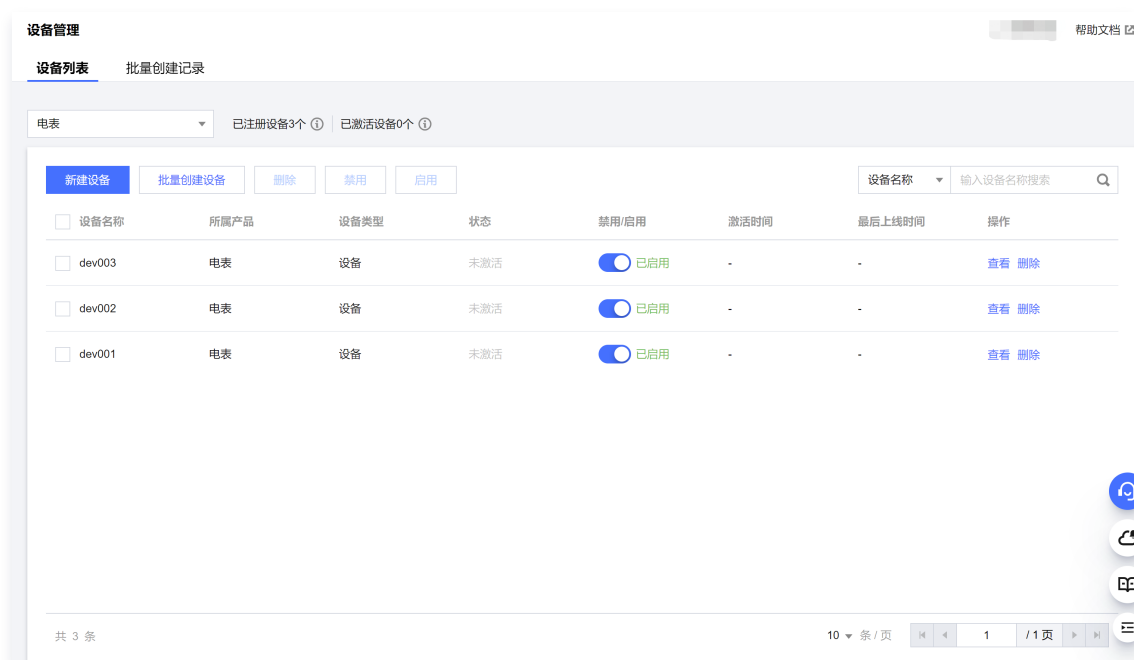
Equipment Management Overview

Device management functionality provides users with the ability to individually or bulk create devices, and supports the deletion, disabling, and enabling of devices. It also allows viewing of basic device information and various business log query features for devices and the platform, to support users' business systems in managing devices more conveniently through the IoT platform.

Operation Steps

View Device List

1. Log in to the [IoT Development Platform](#), enter the instance list page, and select the generated **enterprise instance**.
2. click **Enterprise Instance**, enter the instance inner page, click **Equipment Management** in the left menu, and enter the device list. You can switch products to query devices under the product, or perform a name fuzzy search for devices.



3. Enter the device management page. You can perform the following operations:
 - View device information under a specific product: Select a specific product in the pull-down menu at the top of the page, and load devices under the product by page.
 - Inactive: The device is not integrated with IoT Explorer. You can download the Device SDK for device development and integrate the device with the IoT platform to activate it.
 - Online: The device is activated and successfully connected to the IoT development platform.
 - Offline: The device has been activated but its connection with IoT Explorer is interrupted.
 - Search for devices: Select the device name or device Tag in the search box on the right to search for specific devices. Fuzzy search is supported.

- View device details: Find the corresponding device in the list and click **View** to enter the device details page.
- Delete a device: Find the corresponding device in the list and click **Delete** to delete the device. After the device is deleted, the device certificate information will become invalid, and the data records of the device on the IoT platform will also be deleted.

Create New Device

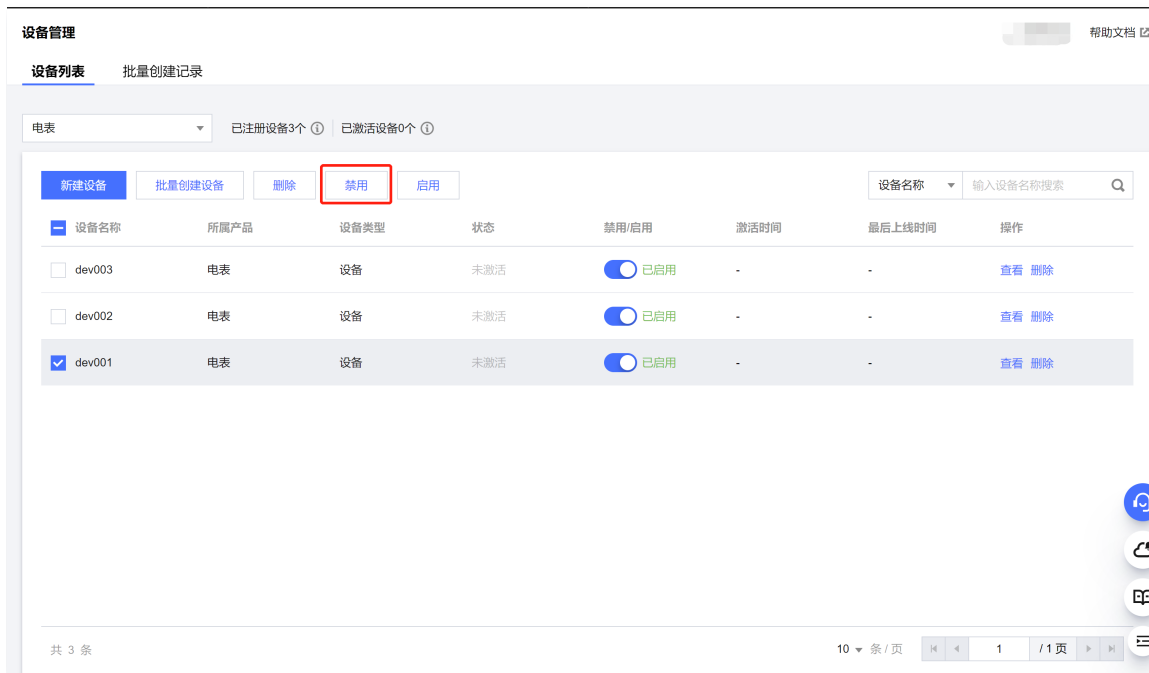
1. Click the "**Create New Device**" button above the device list to enter the device creation page.



2. Enter the device name (the device name under each product is unique. It is advisable to use English or numbers. Chinese characters are not allowed). Click save to create a device.
3. After successful creation, the newly created device will be seen on the device list page. The default status of the first created device is "**Inactive**".

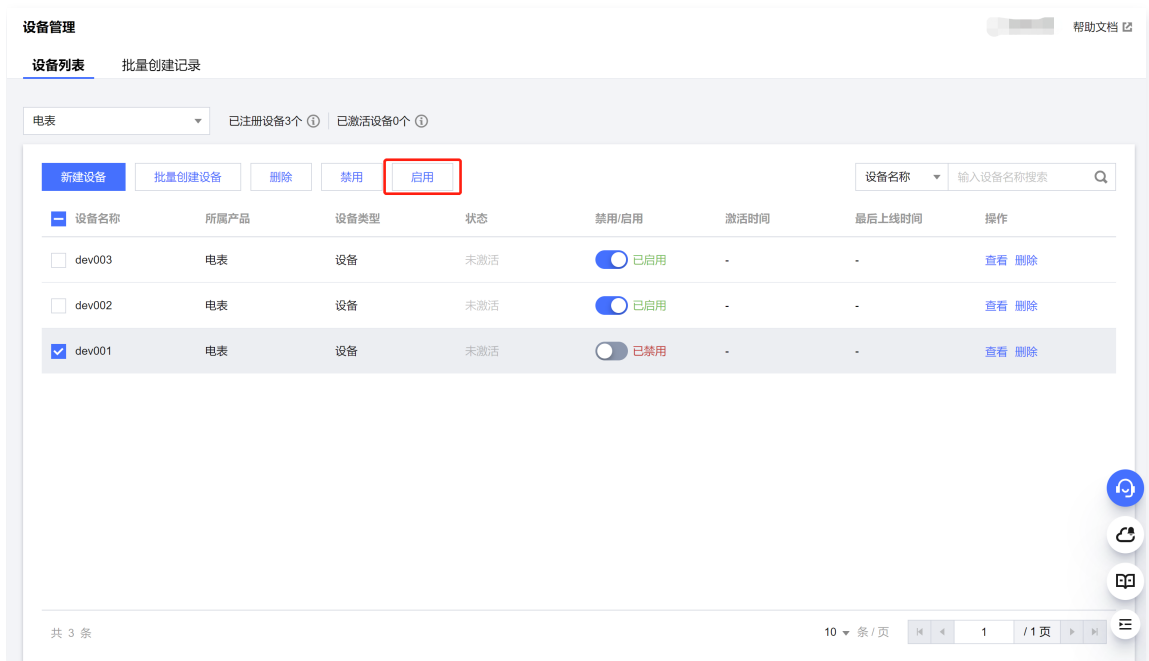
Disable the Device

1. Select a certain device that needs to be disabled, click the **Disable** button. After passing the secondary confirmation, the system will modify the status of the selected device to the **Disabled** status.
2. Devices in the disabled status will not be able to connect to the platform, nor will they be able to communicate with the platform. Users need to confirm whether to disable before disabling the device, to prevent affecting the normal operation of the device and the user's business.



Enable Device

1. Select a certain device that needs to be enabled, click the **Enable** button. After passing the secondary confirmation, the system will modify the status of the selected device to the **Enabled** status.
2. The platform allows devices in the enabled status to connect to the platform normally and communicate with the platform normally via message. If a user misoperates a running device or disables a device due to urgent business needs, when recovery is needed, business can be resumed by enabling the device.



Bulk Device Creation

1. When users engage in mass production, they need to create the equipment information for mass production as needed, burn the triplet information of the equipment into the equipment. The platform

provides the feature to bulk create devices on the console, and users can also create a large number of devices based on the create device API.

2. Click the **"Bulk Device Creation"** button on the device list page. The following window is displayed.
3. Select the product for mass production, set the burning method and generation method. After clicking OK, devices can be automatically generated or imported in batch according to the file uploaded by the user.

批量创建设备

×

量产产品 *

电表

产品ID

RJNA19ZYB2

烧录方式

一机一密

一型一密

一个产品下每个设备烧录产品ID以及唯一的DeviceName与DeviceSecret

生成方式

自动生成

上传文件

由系统自动生成随机并且唯一的DeviceName与DeviceSecret

量产数量

—

5

+

最多一次性量产10000个设备

一机一密二维码

自动生成

无需生成

确定

取消

Product information settings are as follows:

- **Mass-produced products:** Select the burning method as "one device one secret", which means burning different device secrets (DeviceSecret) for each device, and select automatic generation.
- **Burning method**
 - **One device one secret:** It means that each device burns a different device secret (DeviceSecret).
 - **One model one secret:** Usually refers to when users mass-produce equipment on the production line, they unify the burning of a set of firmware and equipment information (product ID, product key), and reduce the burning cost by obtaining the device key via dynamic registration.
- **Generation method:** Automatically generated means that the device name is randomly automatically generated by the platform. Upload file means that users can freely customize the device name according to the uploaded file template.

View Device Information

To view device information, please refer to [Device Debugging](#) in Product Development.

Firmware Upgrade

Last updated: 2025-04-27 17:35:11

Application Scenario

This document primarily introduces the method of use of firmware upgrade on IoT Explorer, helping you quickly use the firmware upgrade service.

Directions

Add Firmware

1. Log in to the [IoT Explorer console](#) and click an enterprise instance to enter the Instance Information Page.
2. Select **Firmware Upgrade** in the left navigation to enter the firmware list page. All firmware in the current project is viewable.
3. Click **Add Firmware** to add new firmware.

添加新固件

① 若固件大小限制不能满足您的业务需求, 可 [提交工单](#) 申请增加文件大小限制

固件名称 *

请输入固件名称

支持中文、英文大小写、数字、部分常用符号（下划线，减号，括弧），必须以中文、英文或数字开头，长度不超过32个字符

所属产品 *

暂无产品

固件版本号 *

请输入固件版本号

仅支持英文字母、数字、点、中划线和下划线，长度限制1~32

选择固件 *

[点击选择固件](#)

仅支持 .bin .dav .tar .gz .zip .gzip .apk .xz .pack 格式的文件，文件大小不能超过1024MB

自定义信息 ①

新增

通过JSON字符串(utf8)形式下发到设备，总大小不能超过1536字节，当前序列化后的大小为0

固件描述

对本次上传的固件进行描述和记录，请输入0-100个字符

对本次上传的固件进行描述和记录，请输入0-100个字符

提交

重置

- Firmware name: Supports Chinese, English uppercase and lowercase letters, digits, and some commonly used symbols (underscore, minus sign, parenthesis). It must start with Chinese, English, or a digit, and its length must not exceed 32 characters.
- Associated product: Select the product to which the uploaded firmware belongs.

- Firmware version number: Only supports English letters, digits, periods, hyphens, and underscores. Length limited to 1 – 32 characters.
- Select firmware: The uploaded firmware Files in the specified format must be in .bin,.dav,.tar,.gz,.zip,.gzip,.apk,.xz,.pack format, and the individual file size must not exceed 1024 MB.
- Custom information: Optional. After creating an upgrade task, the custom information will be sent to the device in the upgrade notice.
- Firmware description: Describe and record the uploaded firmware this time. Length limit 0 – 100 characters.

Note:

A maximum of 100 firmware files can be uploaded under one account. If you continue uploading, deleting old versions of firmware is required.

4. After uploading is complete, the firmware will be displayed in the list. You can perform operations such as upgrading the firmware, CRUD operations, and viewing details.

Firmware Upgrade

After the firmware is successfully uploaded, select the target firmware version you want to upgrade to and click **Firmware Upgrade** on the right side of the firmware list to initiate an upgrade task. The firmware upgrade methods support two kinds of batch upgrade methods: upgrading by firmware version number and upgrading by device name.

| 添加固件 | | | | | 全部产品 | 请输入固件名称 | Q |
|------|-------|------|------|--------------|------|---------|---|
| 固件名称 | 固件版本号 | 所属产品 | 添加时间 | 操作 | | | |
| | | | | 固件升级 查看详情 删除 | | | |

Upgrade by Firmware Version Number

1. Enter the firmware upgrade page. The page displays information about the target upgrade firmware (for example, firmware name, associated product, firmware version number).
2. Select the "batch upgrade method" as **upgrade by firmware version**.

固件升级

×

固件名称

测试

所属产品

固件版本号

升级模式

静态升级

动态升级

批量升级方式 ⓘ

按固件版本

按设备名称

待升级版本号

请选择版本号

升级范围

全部设备

升级确认

静默升级

超时时长配置 ⓘ

−

15

+

分钟

保存

取消

- To-be-upgraded version number: Select the firmware version number in the dropdown list as the firmware to be upgraded. Multiple selections are allowed.
- Upgrade mode
 - Static upgrade: It refers to upgrading only the devices that currently meet the upgrade conditions. Once the upgrade starts, the system will only process these eligible devices. For newly emerging eligible devices subsequently, they will not be automatically included in the scope of this upgrade.
 - Dynamic upgrade: It will not only upgrade the devices that currently meet the upgrade conditions, but also continuously monitor the device status and upgrade the devices that meet the upgrade conditions subsequently. Cover the following situations:
 - Newly activated devices subsequently meeting the upgrade conditions: If after the start of dynamic upgrade, a new device is activated and its status meets the upgrade conditions, the system will automatically include the device in the upgrade scope and upgrade it.
 - The firmware version number currently reported by the device does not meet the upgrade conditions, and a firmware version number that meets the upgrade conditions was reported subsequently: Some devices reported firmware version numbers that do not meet the upgrade requirements when the dynamic upgrade started, but during the subsequent operation, the devices reported firmware version numbers that meet the upgrade conditions. At this point, the system will also perform upgrade operations on these devices.
- Upgrade scope: Support two kinds of upgrade scopes. All devices or specified devices under the version number to be upgraded can be used as firmware upgrade target devices.

The feature to upgrade specified devices is commonly used for grayscale verification of firmware content. When the upgrade scope is set to specified devices, click the **Select Device** button on the right of the dropdown list to select multiple target upgrade devices from all devices under this product.

- **Upgrade confirmation:** Support two firmware upgrade confirmation methods: silent upgrade and user confirmation upgrade. If you use the official Tencent Lianlian application, silent upgrade means that no user confirmation is required, and the Tencent Lianlian application side will automatically complete the upgrade. After re-enabling, it will be the upgraded version; user confirmation upgrade means that the user needs to actively enter the device control interface of Tencent Lianlian and check and confirm the firmware upgrade on the device details page. If you use other IoT applications, it is recommended to choose the silent upgrade method.
 - **Timeout duration configuration:** When the cloud fails to receive the firmware upgrade message from the device exceeding the timeout duration, it will redispach the firmware upgrade task. The default timeout duration for silent upgrade is 15 minutes. The default timeout duration for user confirmation upgrade is 2 minutes. You can also customize the configuration based on actual business needs.
3. Click **Save**. Then, the system will perform the upgrade task and send the selected target version firmware to the target devices within the upgrade scope.

Note:

Upgrading by firmware version requires the upgradable device to report the currently running firmware version. If it is not uploaded, you can choose to upgrade by device name as introduced below.

Upgrade by Device Name

1. Enter the firmware upgrade page. The page displays information about the target upgrade firmware (for example, firmware name, associated product, firmware version number).
2. Select the batch upgrade method as **upgrade by device name**.

固件升级

×

固件名称

测试

所属产品

腾讯连连H5小测试

固件类型

MCU

固件版本号

test

批量升级方式 ⓘ

按固件版本

按设备名称

指定设备

点击选择文件

下载模板

上传文件中请录入准确的DeviceName，一次最多可升级10000个设备，仅支持csv格式。

升级确认

静默升级

超时时长配置 ⓘ

—

15

+

分钟

保存

取消

- **Specify the device:** Upload the device list that needs firmware upgrade. Click **Download Template** to get the template file, enter the accurate DeviceName in the template file, and then click **Upload File** to

upload. Up to 10,000 devices can be upgraded at a time. The file only supports csv format.

- Upgrade confirmation: Consistent with upgrading by firmware name, it supports two firmware upgrade confirmation methods: silently upgrade and confirm upgrade.
- Timeout duration configuration: When the cloud fails to receive the firmware upgrade message from the device exceeding the timeout duration, it will redispach the firmware upgrade task. The default timeout duration for silent upgrade is 15 minutes. The default timeout duration for user confirmation upgrade is 2 minutes. You can also customize the configuration based on actual business needs.

3. Click save. Then, the system will perform an upgrade task and send firmware to the target device.

View Firmware Details

1. In the firmware list, click **View Detail** on the right side of the list to view firmware details.

| 添加固件 | | | | |
|---|----------|-----------|---------------------|--|
| 全部产品 ▼ <input type="text" value="请输入固件名称"/> Q | | | | |
| 固件名称 | 固件版本号 | 所属产品 | 添加时间 | 操作 |
| 测试 | MCU test | 腾讯连连H5小测试 | 2020-11-17 14:26:45 | 固件升级 查看详情 删除 |

2. Enter the firmware details page, where you can view the detailed information of the firmware, firmware upgrade device statistics, and upgrade task management list.

固件信息

编辑

固件名称

测试

所属产品

腾讯连连H5小测试

固件版本

MCU test

固件签名

签名算法

Md5

添加时间

2020-11-17 14:26:45

固件描述

固件类型

MCU

固件升级设备统计

固件升级设备总数

0

升级成功

0

正在升级

0

升级失败

0

任务管理

任务明细

设备明细

请输入任务id

任务id

任务类型

任务状态

添加时间

操作

- Firmware Information: including firmware name, associated product, firmware version number, firmware signature, signature algorithm, addition time, and firmware description. Click the Edit Button in the upper right corner to modify the firmware name and description.

- Firmware upgrade device statistics: including the total number of devices in all batch upgrade tasks for this firmware, as well as the device quantity of firmware upgrade tasks in different upgrade statuses.
- Task management list
 - Click **Task Details** to view all upgrade tasks for this firmware. The statuses of the upgrade tasks include: not started, creating, creation succeeded, creation failed.



- Click **device details** to view the record details of device upgrades in all upgrade tasks associated with this firmware. There are 5 device upgrade statuses: pending, pushed, upgrading, successfully upgraded, and upgrade failed.



3. In the **task details** or **device details** of task management, click **View Details** on the right of a task to go to the task details page, where you can view the device list, upgrade status, and statistics of device quantities in different upgrade statuses for this task upgrade.

固件升级 / 固件详情 / 任务详情

使用指南

任务信息

任务ID 1

产品名称 智能小灯

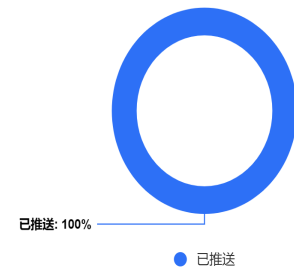
目标版本号 3.1.4

升级范围 全部设备

升级时间 2020-11-01 22:50:13

升级方式 批量升级

任务统计



设备详情

全部设备(1)

升级成功(0)

待推送(0)

已推送(1)

升级中(0)

升级失败(0)

请输入设备名称



| 设备名称 | 当前版本号 | 最后更新时间 | 升级状态 | 状态详情 | 操作 |
|-------|-------|---------------------|------|------|----|
| light | - | 2020-11-01 22:50:23 | 已推送 | - | 取消 |

In the device detail list, you can view the current upgrade status and status details of all devices in the batch task upgrade of this task.

- Do not display status details when the upgrade status is "pending" or "pushed".
- When the upgrade status is "upgrading", the status details include: downloading, burning, and simultaneously display the percentage progress.
- When the upgrade status is "upgrade failure", the status details will feedback error information.

Additionally, on the right side of the device detail list, you can cancel or retry the device upgrade based on the upgrade progress. The upgrade status of a canceled upgrade will be marked as upgrade failure; for a device with an upgrade failure, you can click **Retry** to perform a re-upgrade.

Operational Analysis

Last updated: 2025-04-27 17:35:27

Overview

Use the operational analysis feature to perform statistical analysis on the activation, active, and online device data of all products within the project; and to perform statistical analysis on the distribution of active and activated devices by region for the product.

Prerequisites

There are activated devices, online devices, and devices connected to the platform.

Device Overview

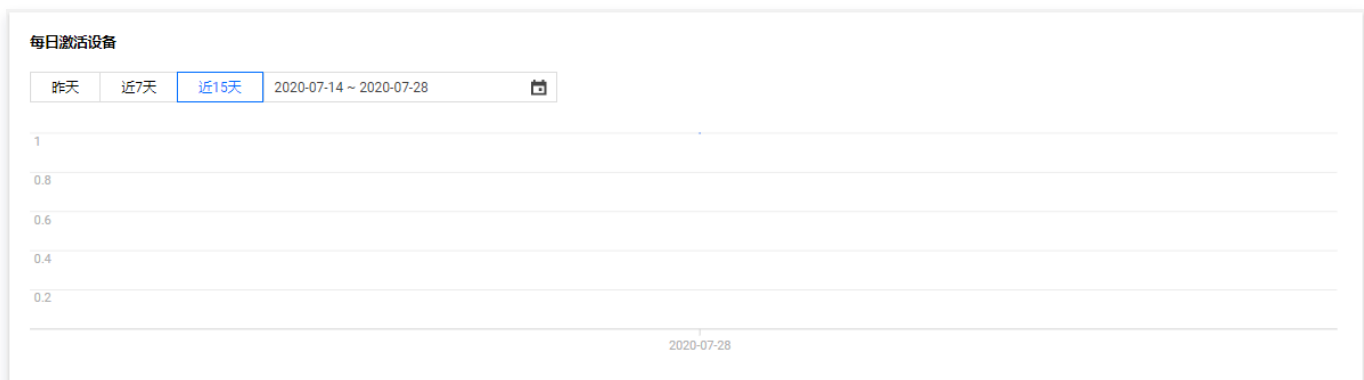
Device overview allows you to view related data of activation, online status, and activity of all products, or view the related data of a specific product by filtering, as well as daily activated devices, daily active devices, and number of online devices info.

Operation Steps

1. Log in to the [IoT Explorer console](#), enter the enterprise instance and click **Operational analysis > Device overview** to enter the device overview interface.
2. Select the product you want to view. The page will display the corresponding total number of activated devices, number of online devices, number of activations yesterday, daily active count yesterday, number of activations in the last 7 days, daily active count in the last 7 days, number of activations in the previous 7 days, and daily active count in the previous 7 days.

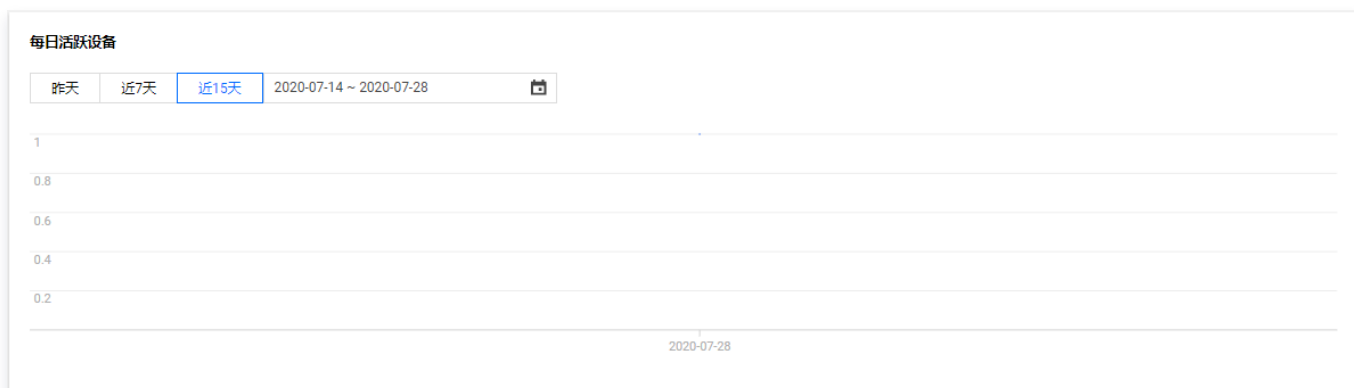
| | | | | | | | |
|------------------------|-------|------|------|-------|-------|-------|-------|
| 当前产品 所有产品 | | | | | | | |
| 激活设备总数 | 在线设备数 | 昨日激活 | 昨日活跃 | 近7日激活 | 近7日活跃 | 上7日激活 | 上7日活跃 |
| 2台 | 0台 | 1台 | 1台 | 1台 | 1台 | 0台 | 0台 |

3. Select the time period for viewing the activated device count. The system will show the daily activated device quantity of the product in the selected time period in the form of a chart.

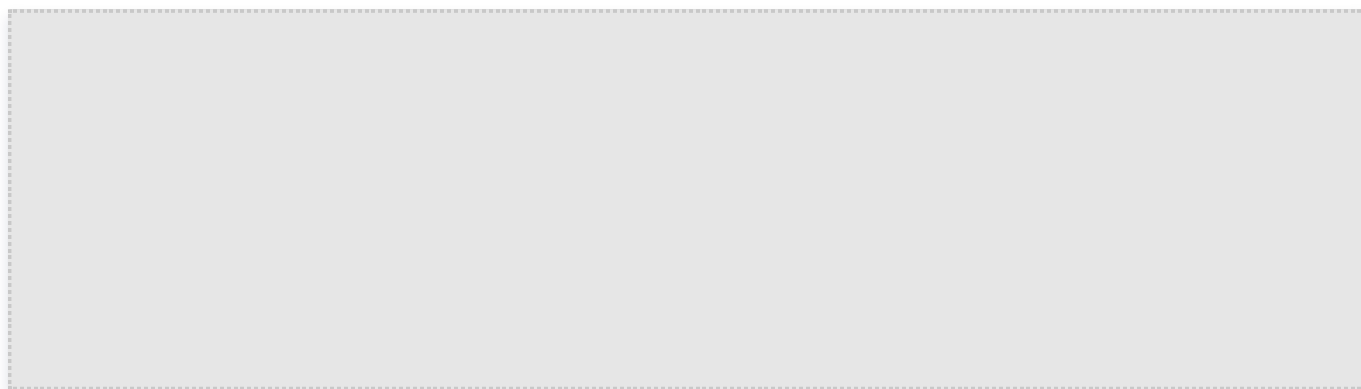


4. Select the time period for viewing the number of active devices (the device must be connected to the Tencent Cloud IoT Platform). The system will show the daily number of active devices of the product in the

specified time range of the filter in the form of a chart.



5. Select the time period for viewing the number of online devices. The system will show the daily number of online devices of the product in the specified time range of the filter in the form of a chart.



Device Distribution

Device distribution allows you to view all products, or filter a specific product to view the number of active and activated devices in different regions within a set time. The system provides two display methods: map and table.

Operation Steps

1. Log in to the [IoT Explorer console](#), enter the enterprise instance and click **Operational analysis > Device distribution** to enter the device distribution interface.
2. Select the product, time period, and device type to view.

3. The system will display the number of device distributions in different regions on the map according to the information such as the product, time period, and device type selected by the user, and intuitively show

the number of device distributions in each region to the user.



4. The system provides a table display method. Users can view detailed information such as the detailed quantity and proportion of each province and city through the table.

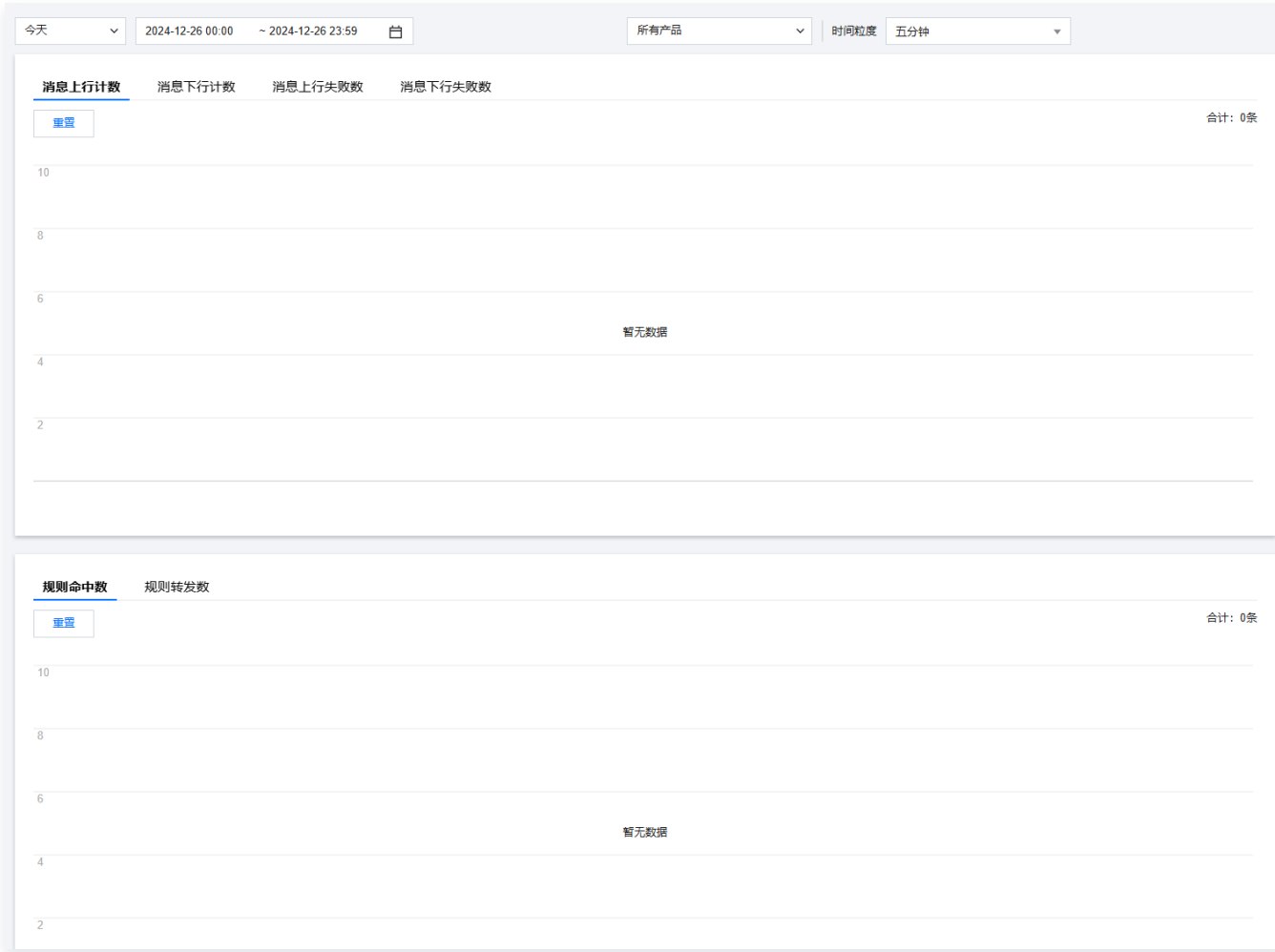
| <div> <div>省份</div> <div>城市</div> </div> | | |
|--|----|---------|
| 地区 | 数量 | 占比 |
| 广东省 | 1 | 100.00% |
| 其它 | 0 | 0.00% |

Message Overview

Message overview allows you to view the message upstream count, message downstream count, message upstream failure count, message downstream failure count, rule hit count, and rule forwarding count for all products, or view those for a specific product by filtering.

Operation Steps

- Log in to the [IoT Explorer console](#), enter the enterprise instance, and click **Operational analysis** > **Message overview** to enter the device distribution interface.
- Select the product and time granularity you want to view to see related message data.



规则命中数

规则转发数

重置

合计: 0条

暂无数据

Application Development

Last updated: 2025-04-27 17:35:44

Application development is to meet users' needs in the business scenario of creating their own brand mini programs, Apps and performing two-way communication with user devices. Users can integrate the application-side SDK provided by the integrated platform on demand to reduce the development costs of C-end applications.

Operation Steps

Creating Application

1. Log in to [IoT Explorer](#), select an instance, then choose **Application Development** from the left menu, and click **Create Application**.
2. Enter the Create Application page and fill in relevant information.
 - **Application name:** Input an easy-to-identify application name according to actual business operations.
 - **Description:** Input remark information of the application.

新建应用

×

您可新建自主品牌小程序应用或App应用，点击[查看文档](#)

应用名称 *

支持中文、英文、数字、下划线的组合，最多不超过50个字符

备注

选填

最多不超过1024个字符

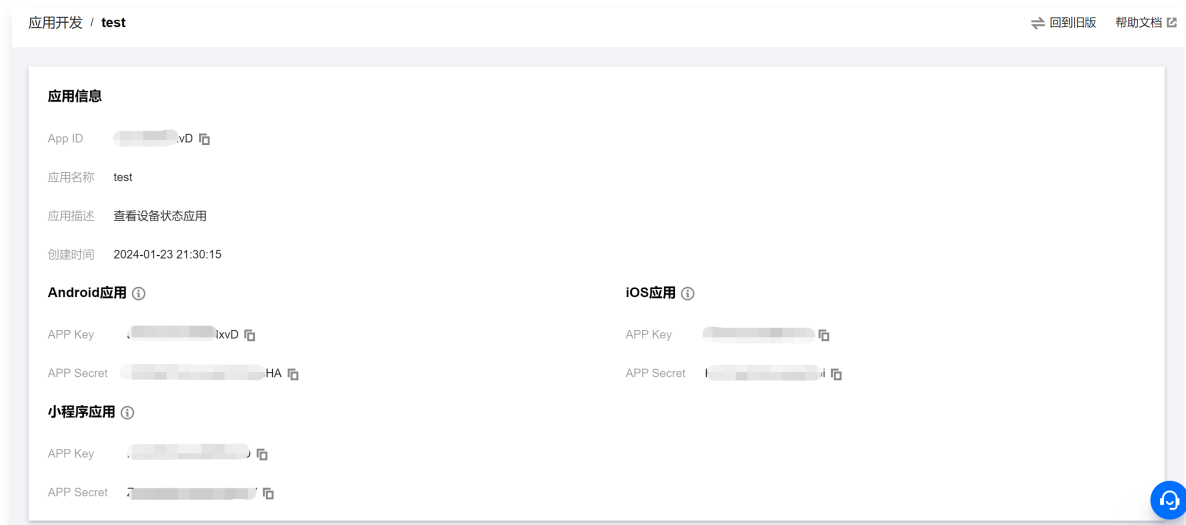
保存

取消

3. Click **Save** to create a new application.

Viewing Applications

1. Select an existing application, after clicking, enter the application information interface, and can view the basic information of the application.



2. The platform automatically assigns three AppKeys and App Secrets to each application by default. You can access the application-side API provided by the platform based on the assigned AppKey and App Secret.
3. To access the devices under a specific product, the application needs to select the corresponding product in the associated products and perform the "Associate" operation.



Deleting Application

Note:

To prevent accidental deletion of data from affecting your business, you need to confirm whether the application data is in use. If it is in use, your application will encounter errors when accessing the platform's API after deletion.

1. When you no longer need this application, you can click **Delete** in the operation area of the application list.
2. Confirm deletion. The system will delete this application.

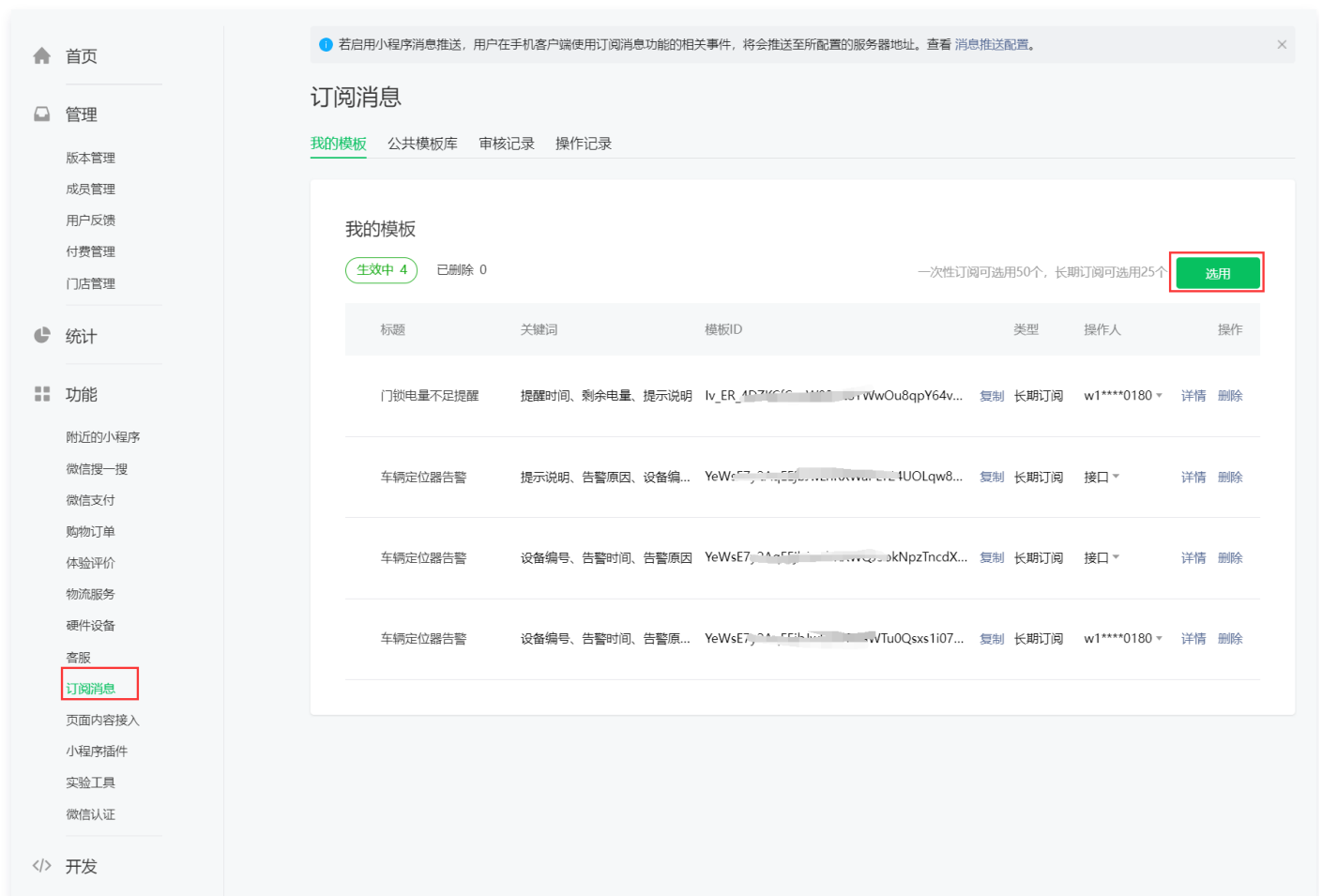
WeChat Strong Reminder Notification

If device messages need to be sent through WeChat service notifications to notify the user of the device, the WeChat strong reminder feature can be configured.

1. You need to first apply for a WeChat mini program from [WeChat public platform](#) and apply for **hardware equipment**.



- After a successful application for hardware devices, then **apply for device capacity**, and in device capacity, select device notifications.
- Once the application for hardware capacity is successful, apply for a message template on the WeChat public platform.



- Apply for enabling the WeChat strong reminder feature on IoT Explorer.

5. After approval, open the WeChat strong reminder configuration switch and perform WeChat model configuration.

应用开发 / 测试应用

应用名称

测试应用

应用描述

-

创建时间

2022-08-04 14:16:40

Android应用

APP Key

aS

APP Secret

tO

Bdhv

iOS应用

APP Key

iwhh

APP Secret

ZyTu

ILIN

小程序应用

APP Key

mGf

APP Secret

SrO

微信强提醒

开通微信强提醒

关联产品

6. In IoT Explorer, use the WeChat account of the Mini Program Administrator to scan code for authorization of **Hardware Service** permissions.
7. Once authorized, you can add the service notification templates applied on the WeChat public platform to your WeChat mini program.

应用开发 / 测试应用

微信强提醒

开通微信强提醒

新增模版

再次授权小程序

已绑定的微信小程序APPID: wx5201edc27c631209

| 产品名称 | 产品id | modelId | 模版名称 | 模版ID | 操作 |
|------|------|---------|------|------|----|
| 暂无数据 | | | | | |

©2013–2025 Tencent Cloud. All rights reserved.

Page 134 of 334

新增模版

产品 *

强提醒测试

ModelId *

3vo0U Ydg

模版id *

Iv_ER_ HoOQf4

模版名称 *

门锁电量不足提醒

触发条件 *

DeviceStatus

消息模版内容 *

time1.DATA

模版取值，输入"{关联事件参数}"

phone_number2.DATA

{{.EventParams.message}}

enum_string3.DATA

{{.EventParams.status}}

新增

跳转小程序页面 *

pages

跳转小程序类型 *

正式版

提交

8. Once the template is added successfully, it will be added to and displayed in the template list.

应用开发 / 测试应用

回到旧版 帮助文档

微信强提醒

开通微信强提醒 ☒

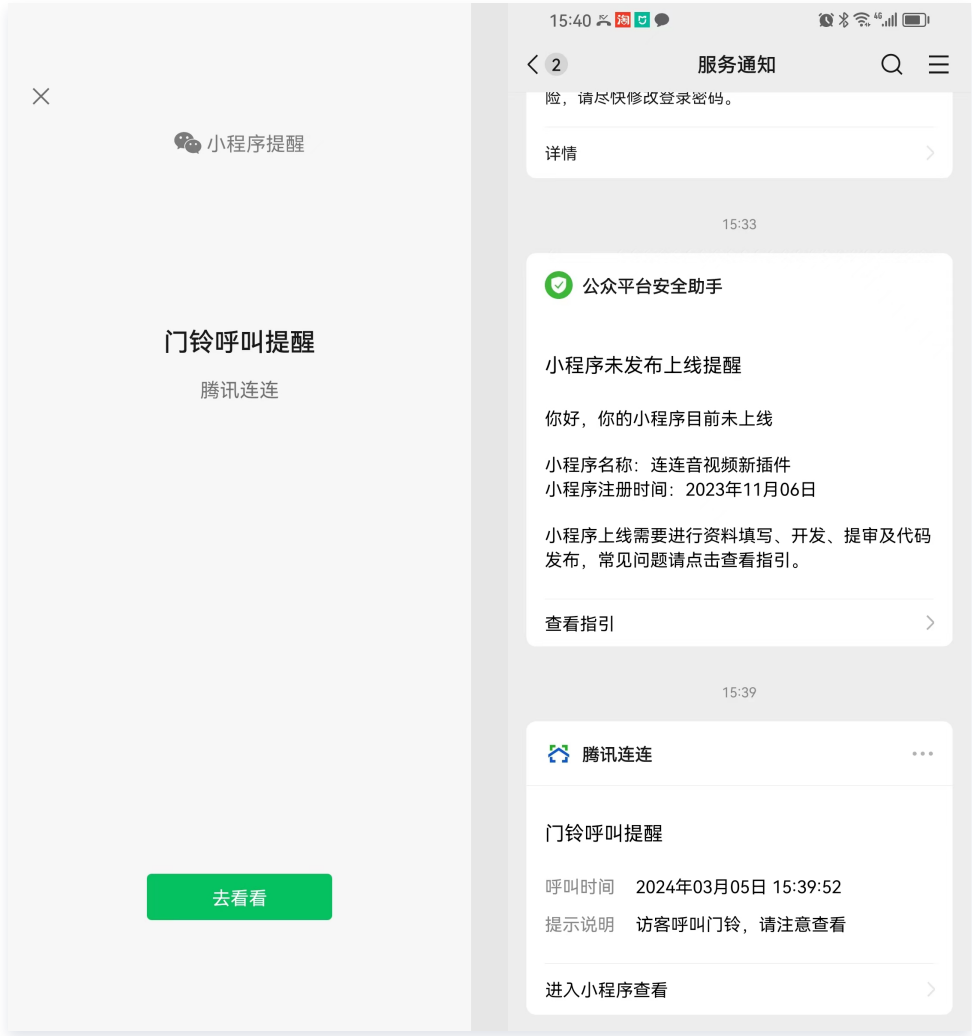
新增模版

再次授权小程序

已绑定的微信小程序APPID: wx5

| 产品名称 | 产品id | modelId | 模版名称 | 模版ID | 操作 |
|------|------|---------|--------|------|-------|
| 门锁 | S | 3v bC | 门铃呼叫提醒 | 2 | 修改 删除 |

9. Once the template is added successfully, WeChat strong reminder notifications can be triggered through device messages.



Data Flow

Message Push

Last updated: 2025-04-27 17:36:13

Overview

To meet developers' needs to process device-reported data and status, by defining rules in the cloud, notifications such as Alarms and messages can be pushed in real time to Tencent Lianlian Official Account or via App Push, reducing the cost for developers to handle device-reported data.

Prerequisites

[Project and product created](#), and the [Thing Model](#) of the product defined.

Create an Alarm Task

1. Log in to the [IoT Explorer console](#), select **Public Instance** or the **standard enterprise instance** you have purchased to enter the project list page.
2. Click on the project name to enter the project details page, click on the left menu **Data Development**.
3. Enter the data development list page and click **Create New Data Stream**.

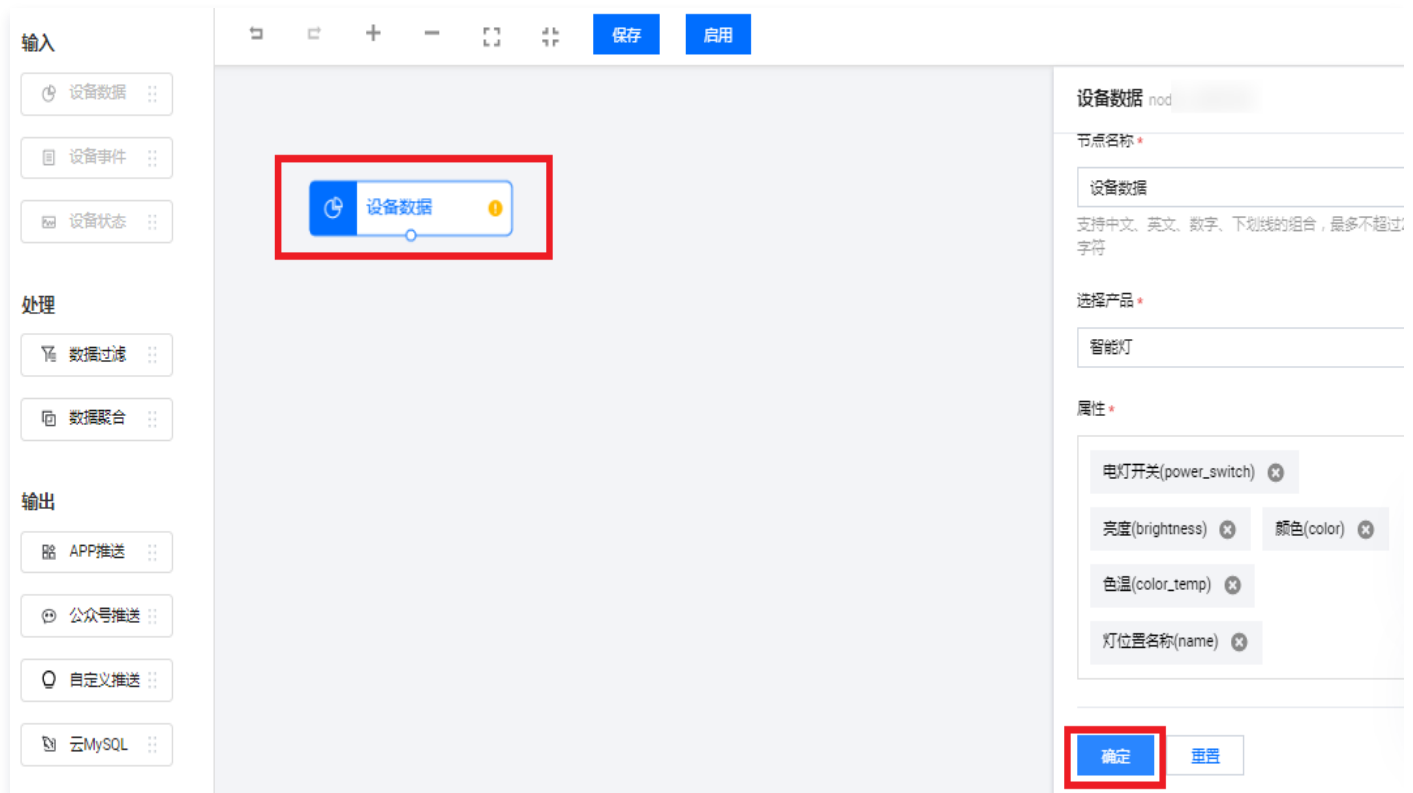


- **Data Stream Name:** Enter the name identifier for data stream processing.
 - **Description:** Input the remark information for this data stream processing.
4. Click **Save**. After saving is successful, the data stream list page will be displayed.

Alert Task Configuration

Device Property Orchestration

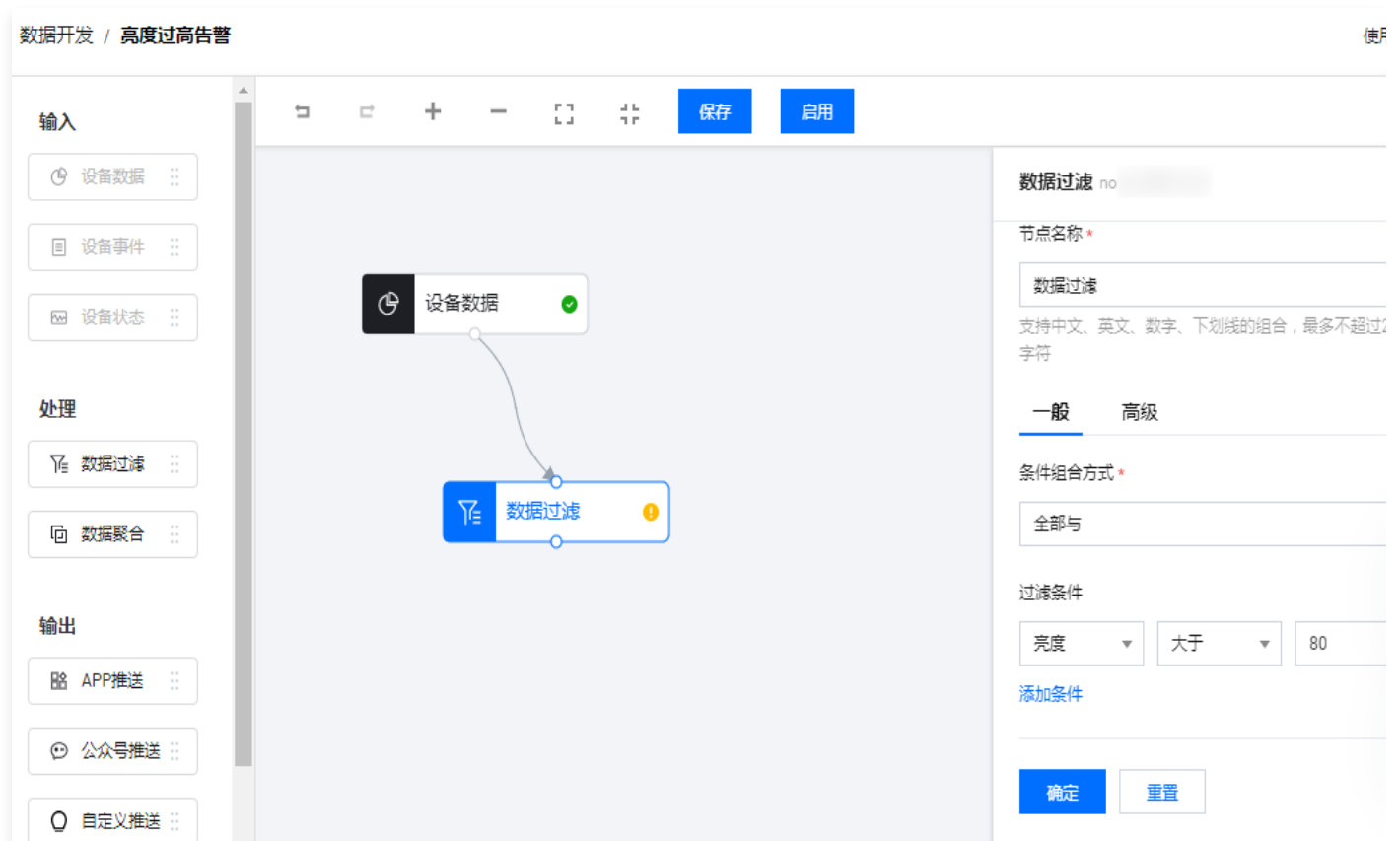
1. Click on the **stream name** in the stream list page to enter the orchestration page.
2. Mouse long press and drag the **device data** node in the left input area and place it in the canvas area. The device data node corresponds to the attributes of device reporting in the Thing Model.
3. Click the **dragged and dropped device data node**. The configuration content of the node will be displayed on the right side of the interface.



- **Node name:** The node name of this **device data**, defaults to "Device Data", modifiable.
 - **Select product:** After dropping down to select a specific product under the project, the platform will perform real-time processing on the reported data of all devices under the product.
 - **Attribute:** Users can select which attributes of the product to use as input as needed.
4. After selecting the product and attribute, you must **confirm** for the system to save the configuration data of the node.
 5. After the **device data** node is saved successfully, the right-side icon of the **device data** node in the canvas will become green.

Data Filtration Orchestration

1. Mouse long press and drag the **Data Filtering** node in the left processing zone and place it in the canvas area.
2. Click the dragged and dropped **Data Filtering** node. The configuration content of the node is displayed on the right side of the interface.
3. Before configuring "data filtering", you must specify a data source, that is, you need to connect the "device data" node with the "data filtering" node.

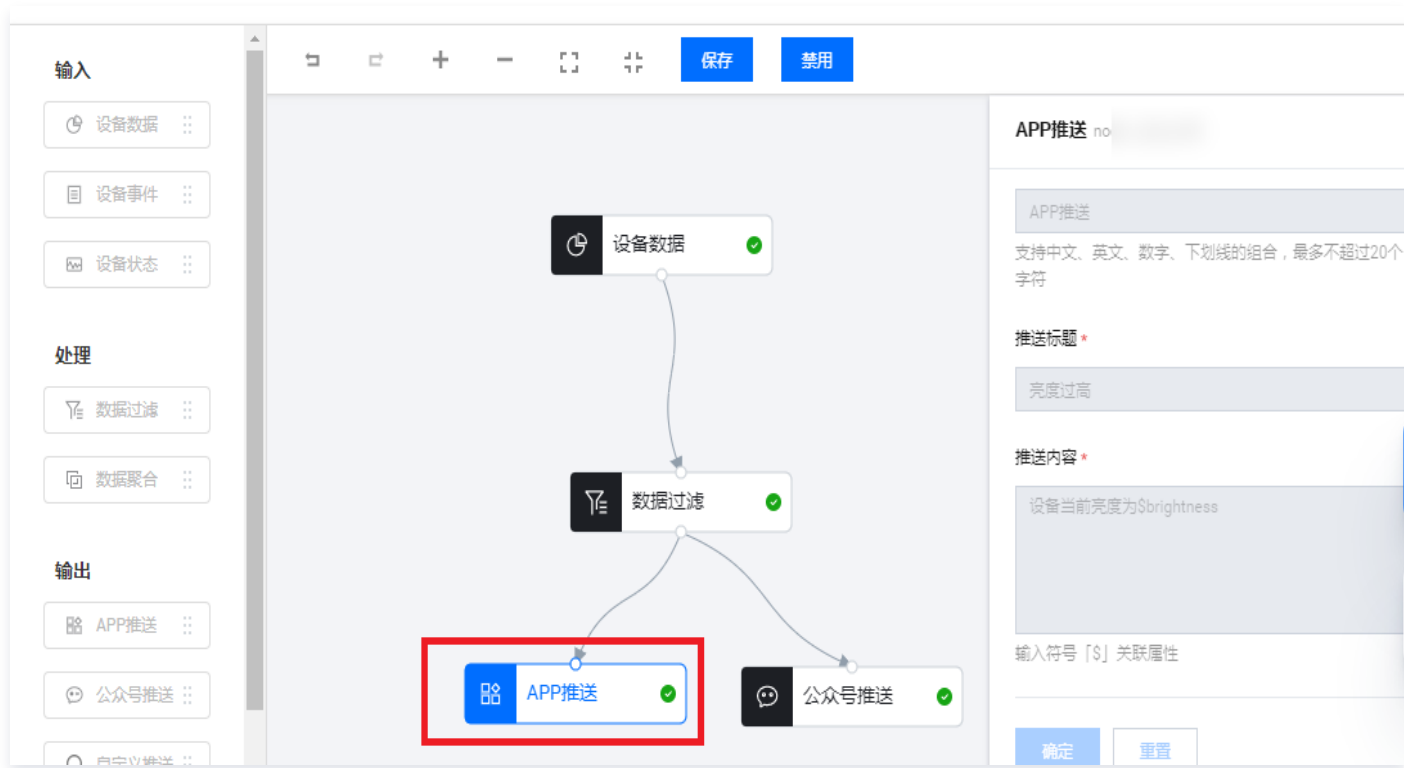


4. The data filtering criteria currently support all AND, all OR logical combinations. Filter conditions can be combined in any way based on the output of the previous node. The selection criteria combination method in this document is "all AND".
5. Add "filter conditions", select the "brightness" attribute, set the selection condition to "larger than", and enter the value 80; this means that only when the brightness value reported by the device is larger than 80, the data will be output to the next processing node.

Tencent Lianlian App Push

If the developer's alert scenario is to push alarm information to the Tencent Lianlian App, the "App Push" node can be placed on the canvas.

1. Mouse long press and drag the **App Push** node in the left output region and place it in the canvas area.
2. **Click the dragged and dropped App Push node.** The configuration content of the node will be displayed on the right side of the interface.
3. Before configuring "App Push", you must specify a data source, that is, you need to connect the "App Push" node with the "data filtering" node.



4. Input the push title.

5. The push content can be defined by the developer, and the output data of the previous node can be obtained through input \$.

6. Click Confirm to save the configuration of the node.

Tencent Lianlian Official Account Push

If the developer's alert scenario is to push alarm information to the Tencent Lianlian Official Account, the "official account push" node can be dragged onto the canvas.

1. Mouse long press and drag the **Official Account Push** node in the left output region and place it in the canvas area.
2. Click the dragged and dropped **Official Account Push** node. The configuration content of the node will be displayed on the right side of the interface.
3. Before configuring "Official Account Push", you must specify a data source, that is, you need to connect the "Official Account Push" node with the "data filtering" node.

The screenshot shows the IoT Explorer interface with a workflow diagram on the left and a configuration panel for the '公众号推送' (Public Account Push) node on the right. The workflow starts with '设备数据' (Device Data), followed by '数据过滤' (Data Filter), and ends with '公众号推送' (Public Account Push). The configuration panel for the '公众号推送' node is open, showing fields for '节点名称' (Node Name), '通知类型' (Notification Type), '推送标题' (Push Title), '推送内容' (Push Content), '跳转路径' (Jump Path), and '推送范围' (Push Scope). The '通知类型' and '推送内容' fields are highlighted with red boxes.

4. Select the notification type: device alarm or notification message. The difference between notification types lies in the different template titles of official account messages.
5. Input the push title.
6. The push content can be defined by the developer, and the output data of the previous node can be obtained through input \$. It includes the attribute data reported by the device and system-level data, such as product ID, device ID, and device alias.
7. Jump path: This jump path is the page that users will be navigated to after they click to view the official account message. It can be selected from the drop-down items.
8. Push scope: You can select which roles to push to from the drop-down items.

Note:

Push content must follow WeChat official account template message regulations. Each intermediate main content should not exceed 20 characters and does not support line breaks.

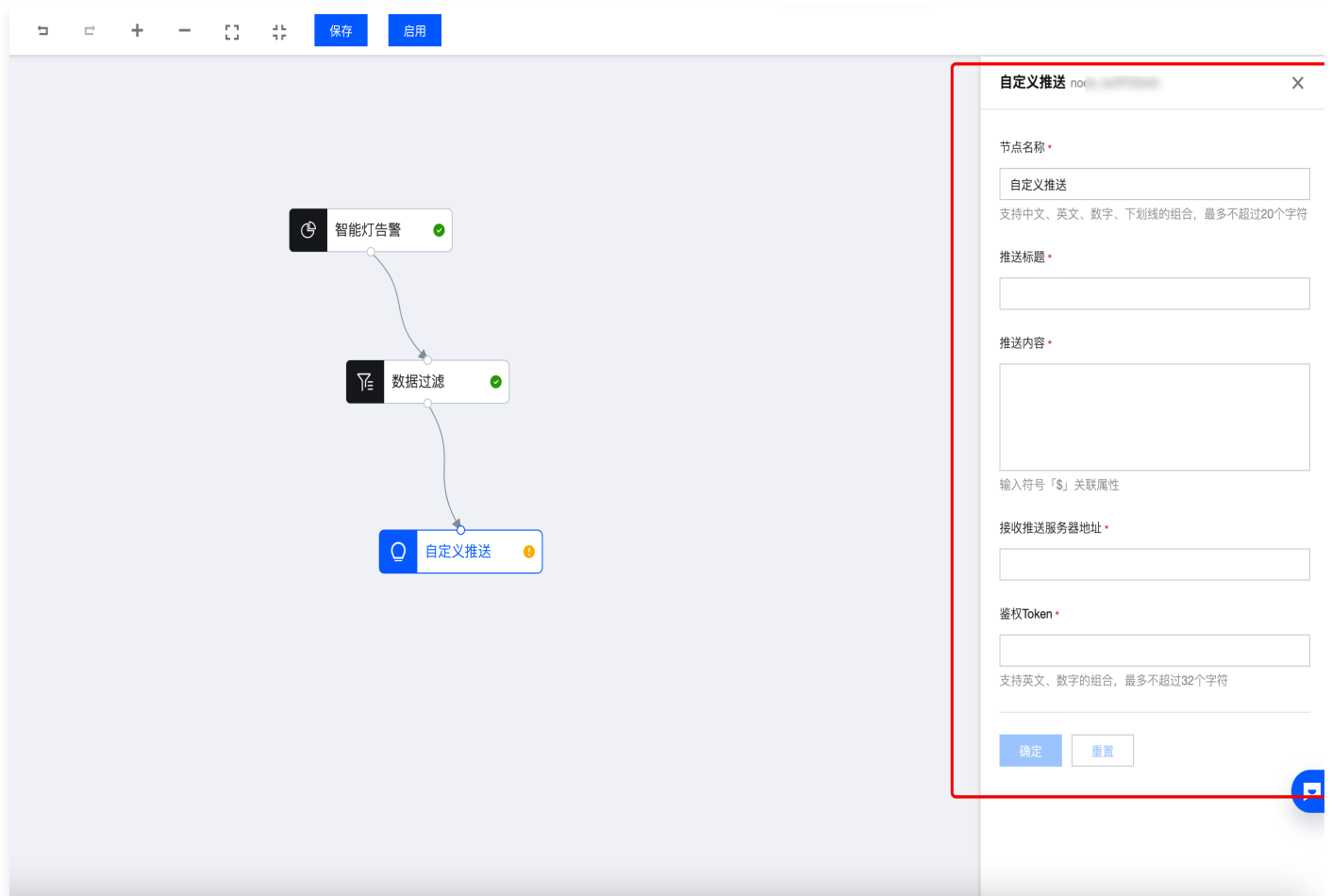
9. Click **Confirm** to save the configuration of the node.

Custom Push

If developers need to forward filtered and aggregated data to their own servers, they can use custom push.

1. Mouse long press and drag the **Custom Push** node in the left output region and place it in the canvas area.

2. Click the dragged and dropped Custom Push node. The configuration content of the node will be displayed on the right side of the interface.
3. Before configuring "Custom Push", you must specify a data source, that is, you need to connect the "Custom Push" node with the "data filtering" node.



4. Input the push title and push content.
5. Input the server address and authentication Token for accepting alarm push notifications and click **Confirm**.

Note:

For your backend's consistent usage, please fill in the authentication Token. You can fill in any Token, used as generating signature (this Token will be compared with the Token contained in the API URL, thereby verifying security).

Push Message Content Format

1. The message template for alarm content pushed from the platform side to the server is as follows:

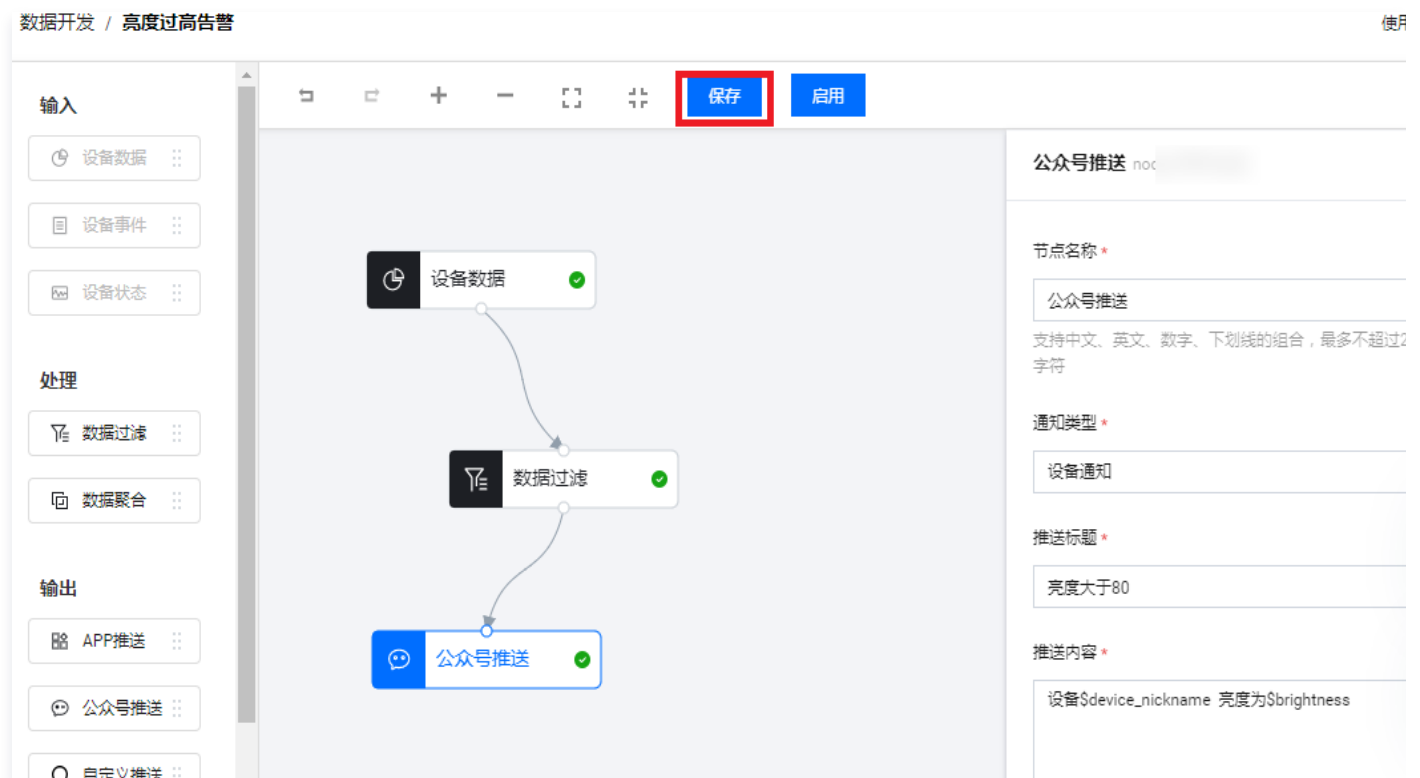
```
{
  "RequestId": "Push request ID"
  "ProductId": "Product ID"
```

```
"DeviceName": "device name"
"MsgTitle": "push title"
"MsgContent": "push content"
}
```

2. If you need to push to your own brand official account, please see the WeChat official account official documentation [Push via Template Message](#).
3. If alarm messages need to be pushed to Chinese domestic brand mini programs or apps, developers can implement according to business logic on their own.

Saving and Enabling Data Stream

1. After the user sets the rules for input, processing, and output nodes, click **Save** > **Enable** at the top of the page.
2. After the data stream is enabled, as long as the reported data of the device meets the defined rules, an official account push will be triggered.



Experiencing Official Account Push

After the alarm rule is configured and enabled as above, you can follow the following two steps to experience the virtual device alarm push official account message.

Bind a Virtual Device to the Tencent Lianlian Mini Program

1. Enter the product corresponding to the above alarm rule, and click **Device Debugging** > **Virtual Device Debugging**.

数据模板 > 设备开发 > 交互开发 > **4 设备调试** > 5 批量投产

设备调试提供真实、虚拟设备调试功能，便于测试设备上报、接收数据是否正常，可创建测试设备后进行测试

新建设备

虚拟设备调试

设备名称

设备名称

Q

| 设备名称 | 状态 | 激活时间 | 最后上线时间 | 操作 | 绑定网关 |
|-----------------------|----|------|--------|----|------|
| 当前产品下设备列表为空，您可以点击新建设备 | | | | | |

智能灯 开发中 [编辑](#)

产品ID

01

产品品类

智慧生活-电工照明-灯

设备类型

设备

认证方式

密钥认证

通信方式

Wi-Fi

数据协议

数据模板

创建时间

2021-06-28 15:53:25

更改时间

2021-07-17 11:15:34

产品描述

-

功能定义

标准功能

8个

自定义功能

0个

2. Open WeChat App and directly use WeChat "Scan" to scan the Device QR Code of the virtual device.

虚拟设备调试

虚拟设备调试

虚拟设备用于模拟真实设备上报设备数据，并模拟接收应用端下发控制指令，您可使用微信或「腾讯连连小程序」扫码体验



3. After successful binding, it will be displayed in the device list as follows:



4. First-time binding requires attention to the "Tencent Lianlian" official account. Click the red box in the figure below to follow the Tencent Lianlian official account, and then you can receive official account

messages pushed from the cloud.



Virtual Device Simulated Reporting Data

1. On the device control panel of the virtual device, fill in the brightness value as 91, fill in other values as the corresponding values, and click **Submit**.

虚拟设备操控面板

属性调试 事件触发

| <input checked="" type="checkbox"/> 功能名称/标识符 | 期望值 | 实时数据 |
|--|---|-------|
| <input checked="" type="checkbox"/> 电灯开关(power_switch) | <input type="checkbox"/> | 关 |
| <input checked="" type="checkbox"/> 亮度(brightness) | <input type="text" value="91"/> % | 91 |
| <input checked="" type="checkbox"/> 颜色(color) | <input type="text" value="Green"/> | Green |
| <input checked="" type="checkbox"/> 色温(color_temp) | <input type="text" value="30"/> % | 30 |
| <input checked="" type="checkbox"/> 灯位置名称(name) | <input type="text" value="0/64"/> 支持英文字母、数字、常见半角符号组合 | - |

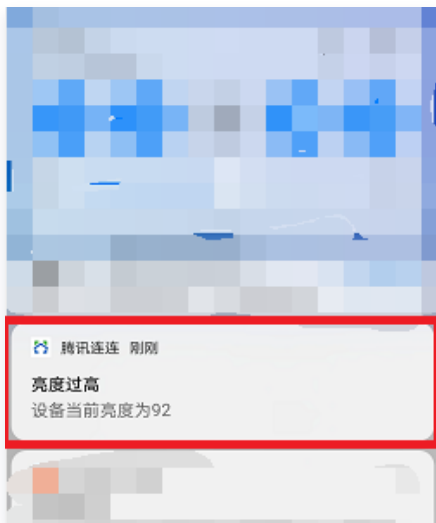
上报 **重置**

2. Enter the Tencent Lianlian Official Account, and a push "message" will be displayed.



Experiencing App Push

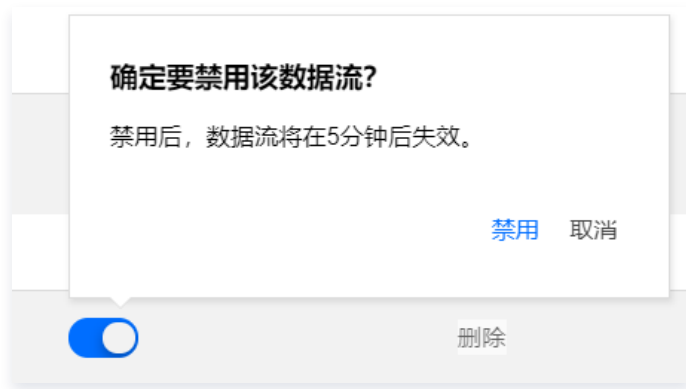
1. Refer to the download instructions of the [General App](#) and download the corresponding app.
2. [Experience official account push](#) After the virtual device simulation reports data, you can view the following App push messages.



Disable Data Stream

1. When a user needs to stop a certain data stream service, enter the data development page to check the corresponding stream list.

2. Click **Off** under the effective status column. A system pop-up confirmation prompt will appear. Click **Disable** to stop the data stream service. Even if the device reports data, the system will no longer process it.



Rule Engine

Rule Engine Overview

Last updated: 2025-04-27 17:36:42

Purpose

When communicating based on a Topic, you can use the rule engine to process the data in the Topic and then forward it to other services on Tencent Cloud or the user's business backend service. You do not need to purchase servers to deploy a distributed architecture. You only need to configure through the rule engine in the console to achieve full-stack services such as collection, calculation, and storage. The following are the types supported for forwarding:

- Forward data to another Topic.
- Forward data to a third-party service.
- Forward data to CKafka, Message Queue.
- Data forwarding to Time Series Database TSDB.
- Forward data to TencentDB for MySQL.
- Forward data to TencentDB for MongoDB.

Create Rules

1. Log in to [IoT Explorer Console](#). After entering the target instance, select **Rule Engine** in the left menu.
2. On the rule engine page, click **Create Rule**, enter the rule name, and click **Confirm**.
 - Rule name: A combination of English letters, digits, and underscores, with a maximum of 32 characters. (The name cannot be modified after creation. Fill in carefully.)
 - Rule Description: 0-256 characters. This is editable.

创建规则

×

规则名称 *

支持英文、数字、下划线的组合，最多不超过32个字符

规则描述

选填

最多不超过256个字符

确定

取消

3. Once created successfully, you can automatically proceed to the Rule Detail Page.

基本信息

编辑

规则名称

规则状态

已禁用

规则描述

测试用，可删除

规则标签

无标签信息

筛选数据

编辑

SQL调试

字段

Topic

\$(productId)/\$(devicename)/event

条件

当前SQL

SELECT FROM \$(productId)/\$(devicename)/event

行为操作

添加行为操作

转发错误行为操作

添加行为操作

You can start writing different forwarding rules.

Last updated: 2025-04-27 17:36:57

Extract the three fields of `action targetDevice count` from the JSON message in the Thing Model attribute reporting Topic of all devices of the intelligent gateway, and filter the data via `count <=3` to obtain the finally processed data for the next data forwarding. The rule expressed in the example in the figure below:

Action

- Forward data to another Topic.
- Forward data to a third-party service.
- Forward data to CKafka, Message Queue.
- Data forwarding to Time Series Database CTSDb.
- Forward data to TencentDB for MySQL.
- Forward data to TencentDB for MongoDB.

When triggering the forwarding behavior, the rule engine will perform JSON encapsulation on the payload reported by the device. The format example is as follows:

1. JSON example forwarded to Ckafka:

```
{
  "MsgType": "Publish",
  "Topic": "AD4GVS5549/device/data",
  "Seq": 13107192,
  "PayloadLen": 17,
  "Payload": "dGhpcyBpcyBhIGV4YW1wbGU=",
  "ProductId": "AD4GVS5549",
  "DeviceName": "device",
  "Time": "2018-08-14 15:12:05"
}
```

The meanings of each field are as follows:

- **MsgType:** Valid values are Publish (for configuring message queue forwarding), Forward (for rule engine forwarding), and StatusChange (for status changes).
- **PayloadLen:** Length of the device-reported message payload, in bytes.
- **Payload:** The payload of the raw message. By default, the content is encoded in Base64.
- **Event:** Only available for StatusChange messages. Currently, the values are Online and Offline, representing online and offline operations.
- **DeviceName:** The device name defined by the equipment in the IoT communication platform.
- **Time:** The timestamp when the forwarding behavior is triggered.

2. JSON example forwarded to a third-party service (http forward):

```
{
  "devicename": "device",
  "payload": {
    "params": {
      "power_switch": 1,
      "color": 1,
      "brightness": 32
    }
  },
  "productid": "AD4GVS5549",
  "seq": 2,
  "timestamp": 1587109346,
  "topic": "AD4GVS5549/device/data"
}
```

The meanings of each field are as follows:

- **devicename:** The device name defined by the equipment in the IoT communication platform.
- **Payload:** The payload of the original message. If the device's original reporting format is JSON, it will be forwarded as-is; if it is in binary format, the content will be encoded in Base64.
- **seq:** A unique message identifier that auto-increments internally, int type.

- timestamp: The Unix timestamp when the forwarding behavior is triggered.

Supplemental Description

Definition of a Field

- The field only supports '*', ',', '.', space, letters and digits. It cannot be empty and has a maximum of 300 characters.
- The field represents the Key-value in JSON. Field filtering is unavailable when the data format is binary. Use '*' to forward all binary data.
- The reported JSON data format can be nested JSON. For example: {"device_status":{"switch":"on"}}, the value of switch can be obtained through device_status.switch.
- Sub-SQL and JSON array are not currently supported.

Definition of Topic Wildcard

If you want to listen to multiple topics, you can use # and + wildcards to define multiple topics.

- # represents zero or more arbitrary Topic segments and can only be placed at the end of a Topic.
- + represents one arbitrary Topic segment, which can be placed in the middle of a Topic.

For example, house_monitor/+/get :

- Can listen to topics such as house_monitor/thermometer/get and house_monitor/door/get .
- But cannot listen to house_monitor/door/switch/get , because + can only represent one Topic segment.

For example, house_monitor/# :

- Can listen to topics such as house_monitor/thermometer and house_monitor/door/switch/get .
- But house/#/get is illegal because # can only be placed at the end of a Topic.

Definition of a Condition

The [condition] expression is used for filtering messages in the Topic. Only when the message satisfies the [condition] expression will it be extracted and undergo subsequent handling. The supported expressions are shown in the table below.

| Operator | Description | For Example |
|----------|-----------------------------|---|
| = | Equal | color = 'red' |
| <> | Not equal to | color <> 'red' |
| AND | Logical AND | color = 'red' AND siren = 'on' |
| OR | Logical OR | color = 'red' OR siren = 'on' |
| () | Bracket represents a whole. | color = 'red' AND (siren = 'on' OR siren ='isTest') |
| + | Arithmetic addition | age = 4 + 5 |
| - | Arithmetic subtraction | age = 5 - 4 |

| | | |
|----|--------------------------|--------------|
| / | Divide | age = 20 / 4 |
| * | Multiply | age = 5 * 4 |
| % | Get the remainder | age = 0 % 6 |
| < | Less than | 5 < 6 |
| <= | Less than or equal to | 5 <= 6 |
| > | Greater than | 6 > 5 |
| >= | Greater than or equal to | 6 >= 5 |

Rule Function

Last updated: 2025-04-27 17:37:12

The rule engine provides multiple functions. You can use these functions in the fields of the rule engine, conditions, and corresponding values of database fields to achieve diversified data processing.

Supported Functions

| Function Name | Usage Description |
|---|--|
| productId() | Return the product ID of the message source. |
| deviceName() | Return the device name of the message source. |
| timestamp() | Return the current Unix system timestamp in seconds. |
| topic() | Return the original Topic of the message source. |
| topic(n) | The nth segment of the original Topic of the message source, separated by <code>/</code> . |
| payloadLen() | Return the byte length of the payload. |
| bin_to_dec() | Convert the binary number data to a decimal integer. |
| to_hex () | Convert the input raw message to a hexadecimal string. |
| randint(min,max) | Return a random integer between min and max. |
| upper(string) | Return uppercase string (The input message format should be in JSON format, and the function object corresponds to the key value. For example, if the input message is <code>"tencent":"iot"</code> , then <code>upper(tencent)=IOT</code>). |
| lower(string) | Return a lowercase string (The input message format should be in JSON format, and the function object should be the corresponding key value). |
| crypto(field,String) | Encrypt the value of the field, with the second parameter String being the algorithm string. Options: MD5, SHA1, SHA256, SHA384, SHA512. (The input message format should be in JSON format, and the function object should be the corresponding key value). |
| concat(string1, string2) | String concatenation, for example <code>concat(deviceid, 'a')</code> or <code>concat(field1, field2)</code> . |
| requestId() | Return the message ID generated by the Internet of Things (IoT) Hub. |
| newuuid() | Return a random uuld string. |
| replace(source, substring, replacement) | Replace the substring in the source. |

substring(source, start, end)

String extraction, return the string from start (including) to end (excluding).

Usage Example

The content of messages sent by a home temperature and humidity device dev00 to the cloud:

```
{"room1":{"temperature":31,"humidity":"63%"},
"room2":{"temperature":26,"humidity":"63%"}}
```

The temperature and humidity products have three devices, dev00, dev01, and dev02, which monitor the temperature and humidity of six rooms, room1, room2, room3,..., room6. Data need to be transferred to the MySQL database for processing only when the temperature of room1 exceeds 30 Celsius. The settings of the rule engine are as follows:

筛选数据 ?

字段 room1.temperature as temp,room1.humidity as hum

Topic /+/data

条件 topic(2) = 'device00' AND room1.temperature > 30

当前SQL SELECT room1.temperature as temp,room1.humidity as hum FROM /+/data' WHERE topic(2) = 'device00' AND room1.temperature > 30

添加规则



① 行为将数据插入到云数据库 (MySQL) 中, [查看文档](#)

行为类型

数据转发到云数据库 (MySQL)

地域 *

请选择地域

实例 *

请选择实例

实例不能为空

Mysql数据库 *

请选择数据库

数据库不能为空

数据表 *

请选择数据表

数据表不能为空

实例登录账户 ① *

允许使用大小写英文字母、数字的组合, 不为空

登录密码 ① *

密码长度为8-30个字符, 至少包含英文、数字和符号_-=+@!%*#?^&.()等三种字符。

数据字段

| 字段名称 ① | 值 ① | | |
|-------------------|--------------|---|---|
| table_temperature | \${temp} | + | - |
| table_humidity | \${hum} | + | - |
| productId | productId() | + | - |
| deviceName | deviceName() | + | - |

数据字段不能为空

☐ 使用批量设置 ①

保存

取消

Data Forwarding to Another Topic

Last updated: 2025-04-27 17:37:26

Overview

Forward the necessary message fields of the business requirement to another Topic to achieve M2M communication between different devices. The completion of Topic supports the following methods:

- Fill in a Topic Name

For example `${productId}/house_monitor/thermometer` to forward messages that meet the rules to this Topic.

- Fill in a Topic Name with variables

For example `${productId}/${house}/device`, where `house` enclosed in `${}` represents a variable name, and this variable name is the field content selected in the SELECT statement.

Example Illustration

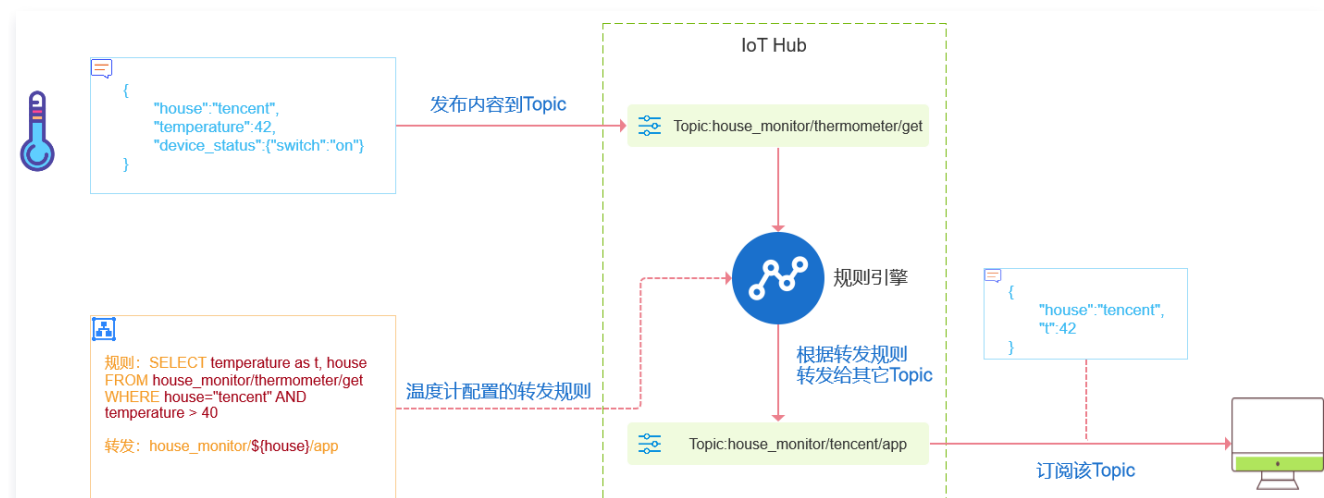
This example mainly explains how the forwarding Topic with variables takes effect. Assume one rule is defined. An example is as follows:

```
SELECT temperature as t, house
FROM house_monitor/thermometer/get
WHERE house="tencent" AND temperature > 40
```

This rule extracts the values of `t` and `house` from the message. Suppose the content of the `house` field is `tencent`.

At this point, if forwarding to the Topic `house_monitor/${house}/app` is defined, then the rule engine will replace the `${house}` variable in this Topic with `"tencent"`, thereby sending the field contents of `t` and `house` to the Topic `house_monitor/tencent/app`.

The entire forwarding process is shown in the figure below:



Configuration

1. Log in to [IoT Explorer Console](#), click the target instance, select **Data Flow > Rule Engine** from the left menu bar, and click the configuration rules that need to be configured.

2. On the Rule Detail Page, click **Add behavior operation**.
3. In the pop-up "Add Rule" window, fill in relevant information. Click **Save** on IoT Explorer, and the device-reported data can be sent to the Topic.
 - Select the behavior type "Data Forwarding to Another Topic (Republish)".
 - Select the corresponding project, product and device.
 - Select another Topic Type and names to forward to.

添加规则

×

① 将筛选后的数据转发到另外一个Topic中[查看文档](#)

行为类型

数据转发到另一个Topic (Republish)

项目 *

产品 *

请选择产品

产品不能为空

设备 *

请选择设备

☒ 手动填写

不能为空, 支持字母数字_()的组合和\${}形式的变量, 长度1~48

Topic类型 *

自定义

Topic *

请选择Topic

名称命名支持字母、数字、下划线、“(”、“)”、“\$”、“{”、“}”、“.”组合；不同层级之间用 / 分层。+表示一级，使用/+命名，不能/+aaa/；长度限制为1-64位。

☒ 手动填写

保存

取消

Forwarding Message Service Quality Grade

The service quality grade of a message will not change when it is forwarded from the source Topic to other Topics.

- When the service quality grade of the message published by the device is QOS0, the rule engine will forward it according to QOS0 messages. When the service quality grade of the published message is QOS1, it will be forwarded according to QOS1.
- When the QoS level of forwarded messages is 0, the messages will be discarded if forwarding fails. When the QoS level of forwarded messages is 1, the system will retry forwarding the messages if forwarding fails. The retries will be performed sequentially at intervals of 3s, 6s, and 9s for three times. If all three retries fail, the messages will be saved in the offline message queue.

Forwarding Data to a Third-Party Service

Last updated: 2025-04-27 17:37:42

Overview

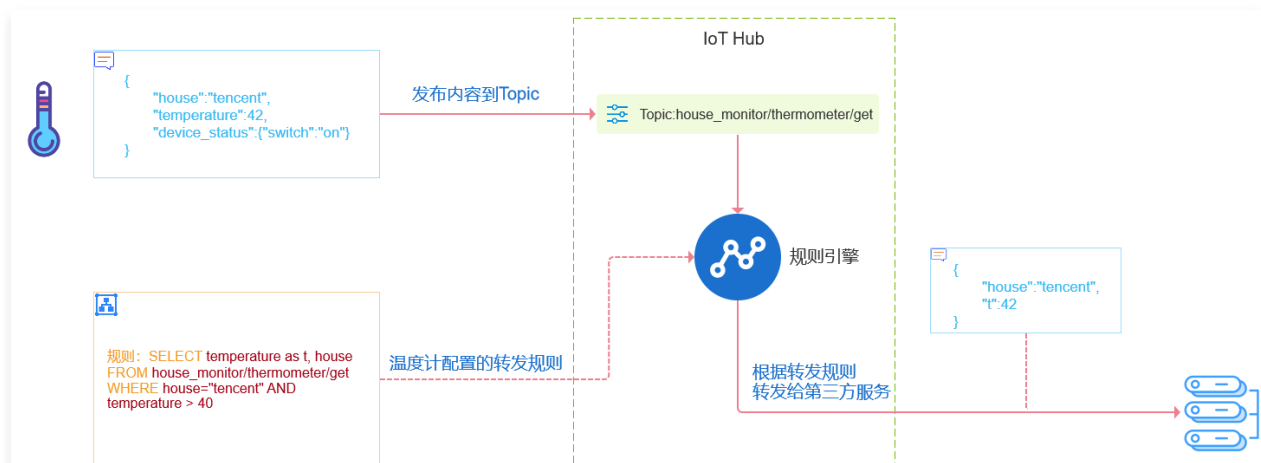
When forwarding device data extracted by rules to the third-party HTTPS service deployed by users, you can customize how to handle these data. This approach is the highest one in flexibility provided to users for message processing.

Note:

Third-party services must provide services via HTTP or HTTPS. To configure forwarding for third-party services, you need to provide the website URL and port that support HTTP or HTTPS. After successful rule engine forwarding, the third-party service will receive data packets from

42.193.134.62, 106.52.211.220, 157.255.11.202, 14.215.166.14, and 120.233.106.237.

The following diagram shows the entire process of forwarding data to a third-party service:



For the data content and format forwarded, please see [Data Processing](#) document.

Filling in Server Configuration

1. Log in to the [IoT Explorer console](#), click the target instance and project name, and select **Data Flow > Rule Engine** in the left menu bar.
2. Click the rule you want to configure. Enter the Rule Detail Page and click **Add Behavior Operation**.
3. In the pop-up "Add Rule" window, fill in relevant information. Click **Save**.
 - Select the behavior type "Forward data to third-party service".
 - Select the API address type. You can option "use existing HTTP service address" or "use IoT Enable service address (recommended)".
 - Fill in your HTTP or HTTPS service address. IoT Explorer will forward the reported data from devices to the HTTP or HTTPS service address.
 - Check "Add authentication Token" and fill in the Token corresponding to your service; you can add any Token, used as generating signature (this Token will be compared with the Token contained in the API URL, thereby verifying security).

添加规则

将筛选后的数据转发到第三方服务中。您可使用物联使能部署您的服务，提供 服务端模板支持低代码、免备案获取服务地址，转发更快速稳定、资源消耗更低，点击[查看文档](#)

行为类型

数据转发到第三方服务 (Forward)

API地址: *

☒ 使用已有HTTP服务地址 ☐ 使用物联使能服务地址 (推荐)

http://act.com

☒ 增加鉴权token

Token: *

thread

保存

取消

Verification Message Comes From the IoT Development Platform

Note:

For your consistent backend usage, please select to add an authentication Token.

Request ID

If the user has selected "Add authentication Token" during forwarding to a third-party service (Forward), i.e., HTTP forwarding, the IoT development platform will add the following fields in the header of the HTTP or HTTPS request:

| Parameter | Description |
|-----------|---|
| Signature | Signature combines the Token parameter filled in Add Rule and the Timestamp parameter and Nonce parameter in the request. |
| Timestamp | Timestamp. |
| Nonce | Random number. |

1. Perform lexicographical order on the three parameters: Token, Timestamp, and Nonce.
2. Perform string concatenation on the three parameter strings to form a string and then perform sha1 encryption.

- The developer obtains the encrypted string and compares it with the Signature to identify that the request comes from IoT Explorer.

Sample PHP code for inspecting the Signature:

```
private function checkSignature()
{
    $signature = $_GET["signature"];
    $timestamp = $_GET["timestamp"];
    $nonce = $_GET["nonce"];

    $token = TOKEN;
    $tmpArr = array($token, $timestamp, $nonce);
    sort($tmpArr, SORT_STRING);
    $tmpStr = implode( $tmpArr );
    $tmpStr = sha1( $tmpStr );

    if( $tmpStr == $signature ){
        return true;
    }else{
        return false;
    }
}
```

For example, for a certain request, the relevant parameters are as follows. The user sets the Token to aaa.

```
Nonce: IkOaKMDalrAzUTxC
Signature: c259ed29ec13ba7c649fe0893007401a36e70453
Timestamp: 1604458421
```

The sorted string is `1604458421IkOaKMDalrAzUTxCaaa` . The final sha1 calculation result is `c259ed29ec13ba7c649fe0893007401a36e70453` .

Service address validation

- When enabling the rule engine, IoT Explorer will send a GET request once to the filled server address URL, and add the following fields to the GET request header:

| Parameter | Description |
|-----------|---|
| Signature | Signature combines the Token parameter filled in Add Rule and the Timestamp parameter and Nonce parameter in the request. |
| Timestamp | Timestamp. |
| Nonce | Random number. |
| Echostr | Random string. |

IoT development platform sends a message example to a third-party service:

```
GET / HTTP/1.1
Host: ...:4443
User-Agent: Go-http-client/1.1
Content-Type: application/json
Echostr: UPWIAFASvDUFcTEE
Nonce: testrance
Signature: abb6c316a8134596d825c5a1295bfa6f7657664d
Timestamp: 1623149590
Accept-Encoding: gzip
```

2. If the third-party service confirms that this GET request comes from IoT Explorer, please return the content of the Echostr parameter in the body without modification.

Message example of the third-party service replying to IoT Explorer:

```
HTTP/1.1 200 OK
Date: Tue, 08 Jun 2021 10:53:10 GMT
Content-Length: 16
Content-Type: text/plain; charset=utf-8

UPWIAFASvDUFcTEE
```

3. IoT Explorer verifies the content of the returned Echostr parameter to confirm whether the server address URL is valid.

3.1 Node.js service address validation example:

```
const sha1 = require('js-sha1');

// Function for authentication information verification
function checkSignature(token, signature, timestamp, nonce) {
  // Concatenate token, timestamp, and nonce into a string after lexicographical
  sorting
  let tmpStr = [token, timestamp, nonce].sort().join('');
  // sha1 encryption
  tmpStr = sha1(tmpStr);
  return tmpStr === signature;
}

app.get('/test', (req, res) => {
  const token = 'the token you configured in the rule engine here';
  const { signature, timestamp, nonce, echostr } = req.headers;
  if (checkSignature(token, signature, timestamp, nonce)) {
    // If confirm that the GET request comes from IoT Explorer, respond with
    echostr as the body
    res.set('Content-Type', 'text/plain; charset=utf-8');
    res.send(echostr);
  } else {
    res.status(403).send('Forbidden');
  }
}
```

}})

Deploying Based on Cloud Lightweight Server

You can choose to use the SCF platform to quickly deploy a Web service, or choose to self-build a Web service. The data processed by the rule engine can be forwarded by calling a third-party API and received on the server side, and you can decide how to handle it.

Note:

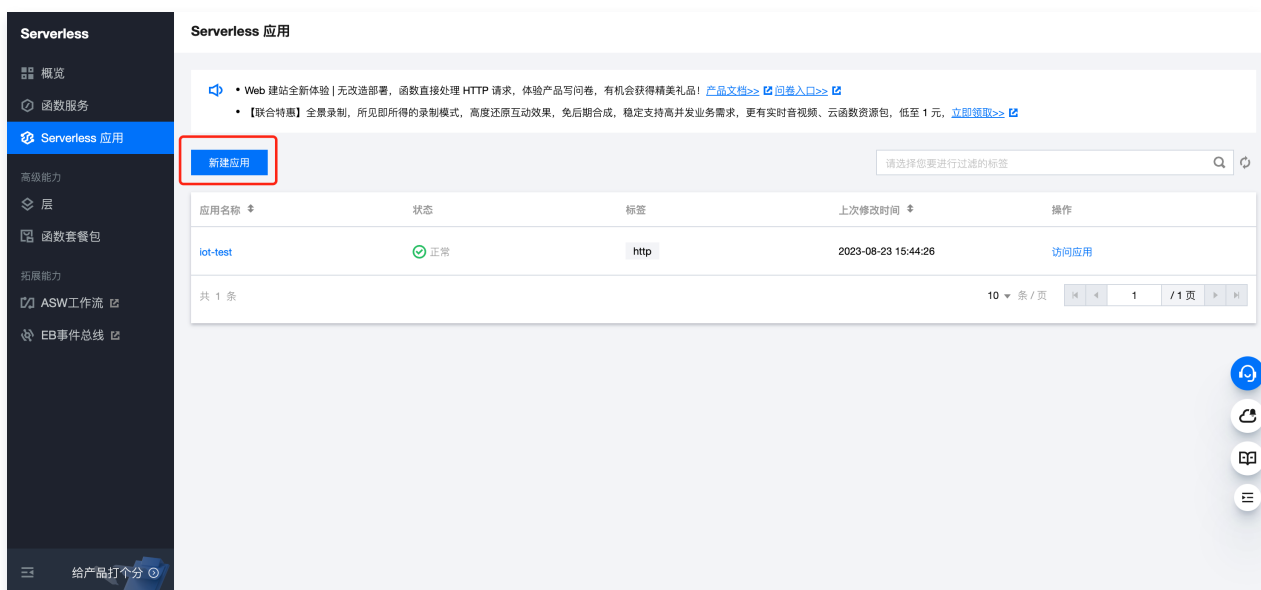
The rule engine will make a GET request when validating the authentication information for the first time, and subsequent forwarding will be a POST request.

The data fields forwarded and received by the third-party service API are as follows:

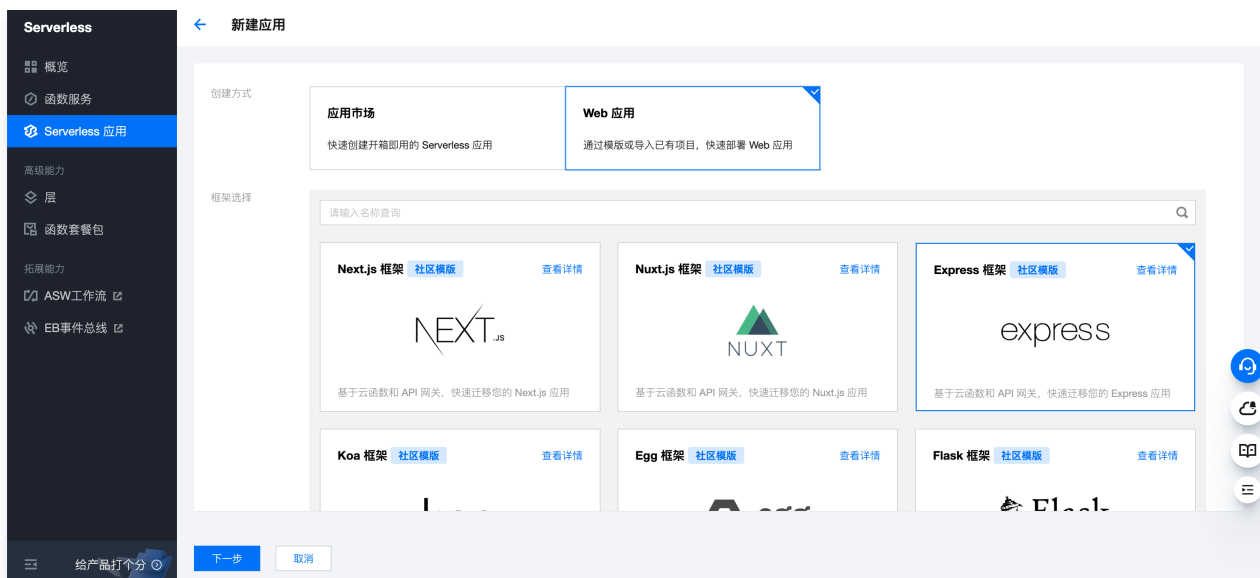
```
{
  payload: {
    clientToken: '7d9b4f5-22bd-4455-6a4c-6dce7d95b3',
    method: 'report',
    params: { body_temperature: 20 }, //Forwarded parameters
    timestamp: 1692787444
  },
  timemills: 1692787444271,
  seq: 1141064,
  timestamp: 1692787444,
  topic: '$thing/up/property/Y6ONBYP3U5/test',
  devicename: 'test',
  productid: 'Y6ONGYP3U1'
}
```

1. Example of quickly building a Node service based on Express on the SCF platform

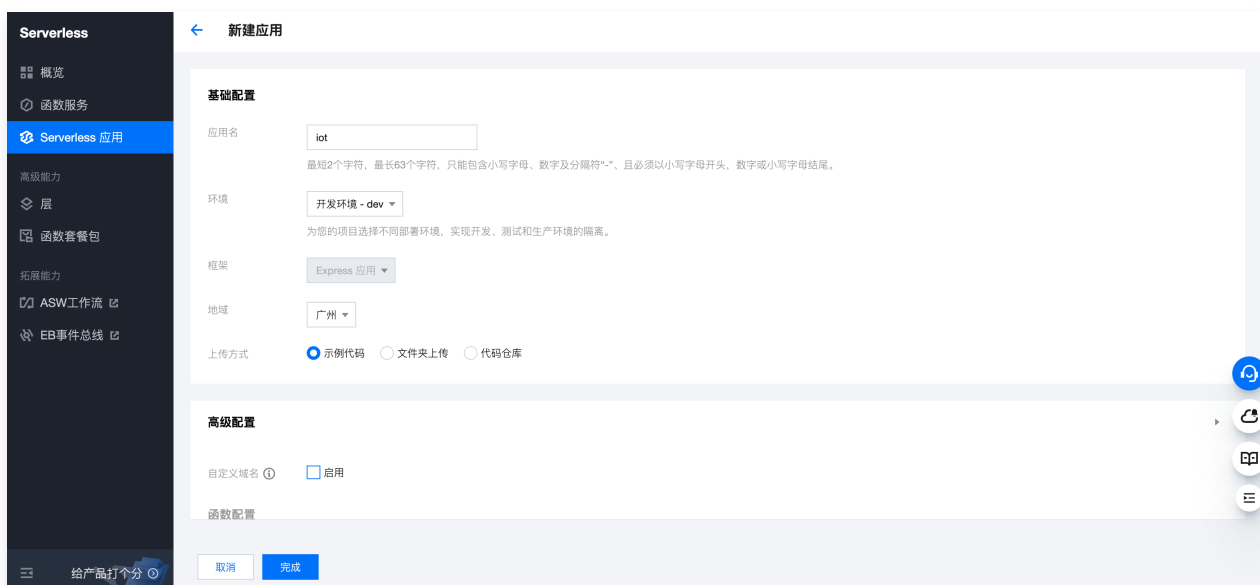
1.1 Log in to the SCF console, select a Serverless application, and create a new application.



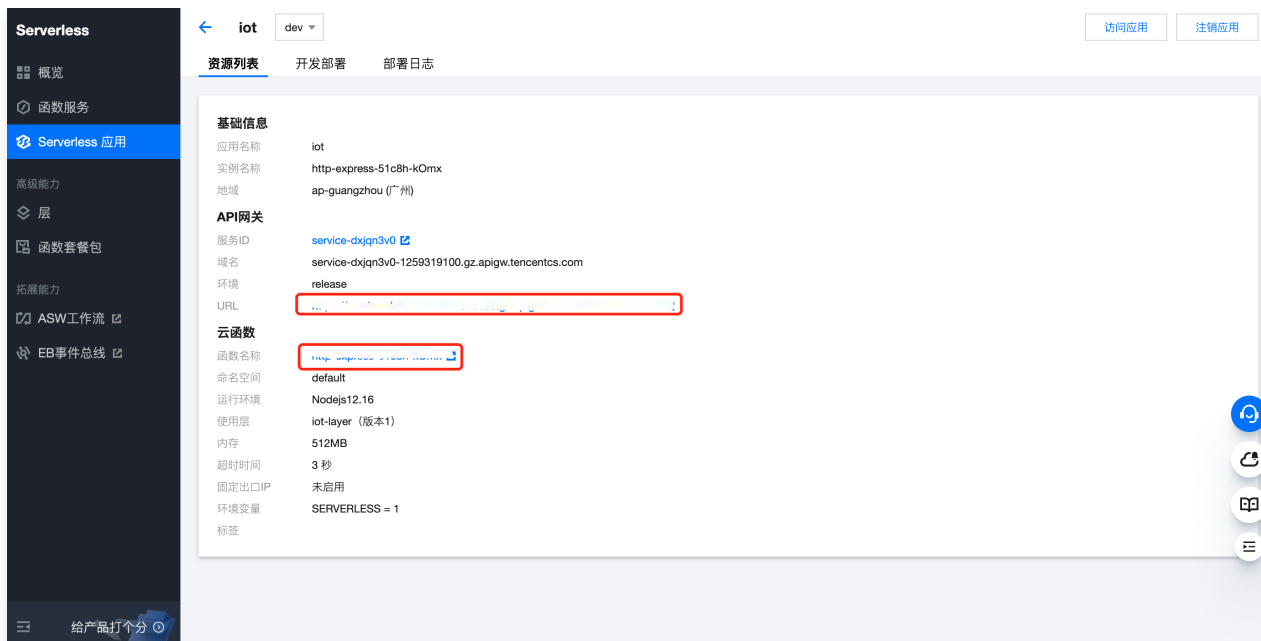
1.2 Select the Express framework template under the Web application, and click next.



1.3 Configure according to your own needs. Click finish after configuration.



1.4 Wait until the deployment is completed and then enter the application. You can see the URL in the API gateway. This URL is the API address you need to fill in the rule engine. Assume that this URL address is: `https://iot-api/`. Then in this example, your API address is the URL plus the interface route, that is: `https://iot-api/test`. Next, click on the function name to enter the function service.



1.5 Below are example codes for authentication information, checksum, receipt of rule engine forwarding parameters:

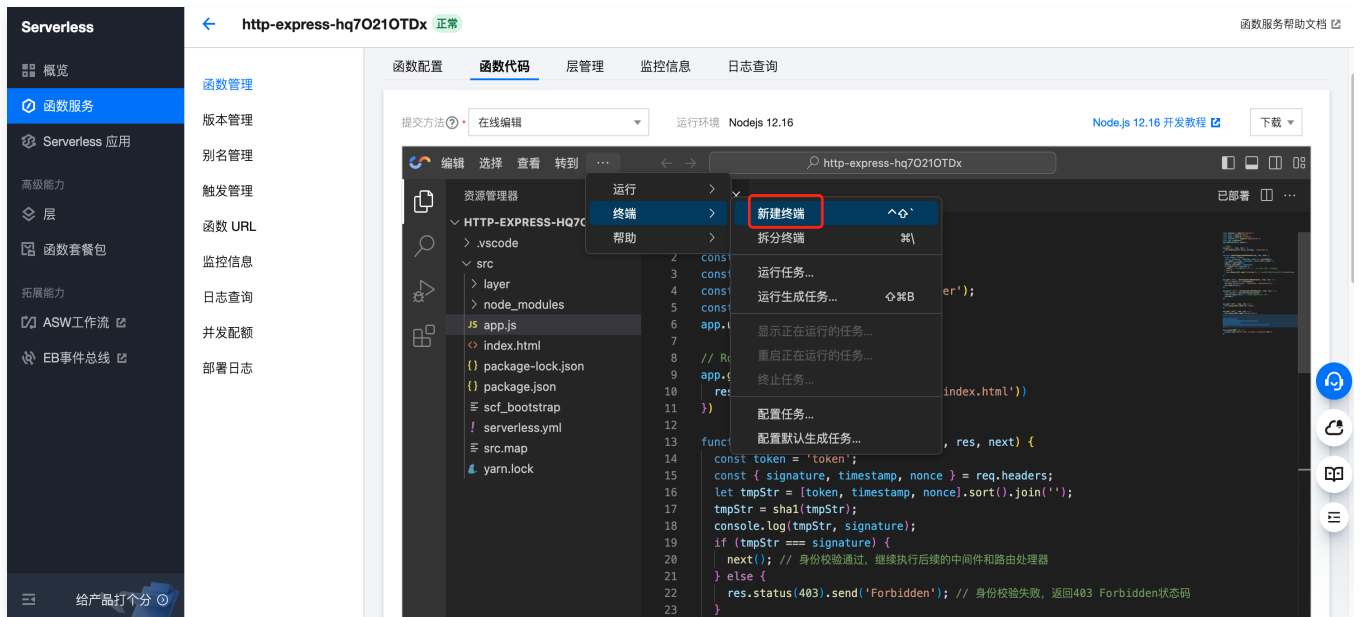
```
const express = require('express')
const bodyParser = require('body-parser');
const sha1 = require('js-sha1');
const app = express();
app.use(bodyParser.json());

// The code for authentication information verification has been explained
// above. Here, Middleware is used to simplify the code.
function checkSignatureMiddleware(req, res, next) {
  const token = 'the token you configured in the rule engine here';
  const { signature, timestamp, nonce } = req.headers;
  let tmpStr = [token, timestamp, nonce].sort().join('');
  tmpStr = sha1(tmpStr);
  if (tmpStr === signature) {
    next(); // Identity verification passed. Continue execution of subsequent
    // middleware and route handlers.
  } else {
    res.status(403).send('Forbidden'); // Identity verification fails, return
    // 403 Forbidden status code
  }
}

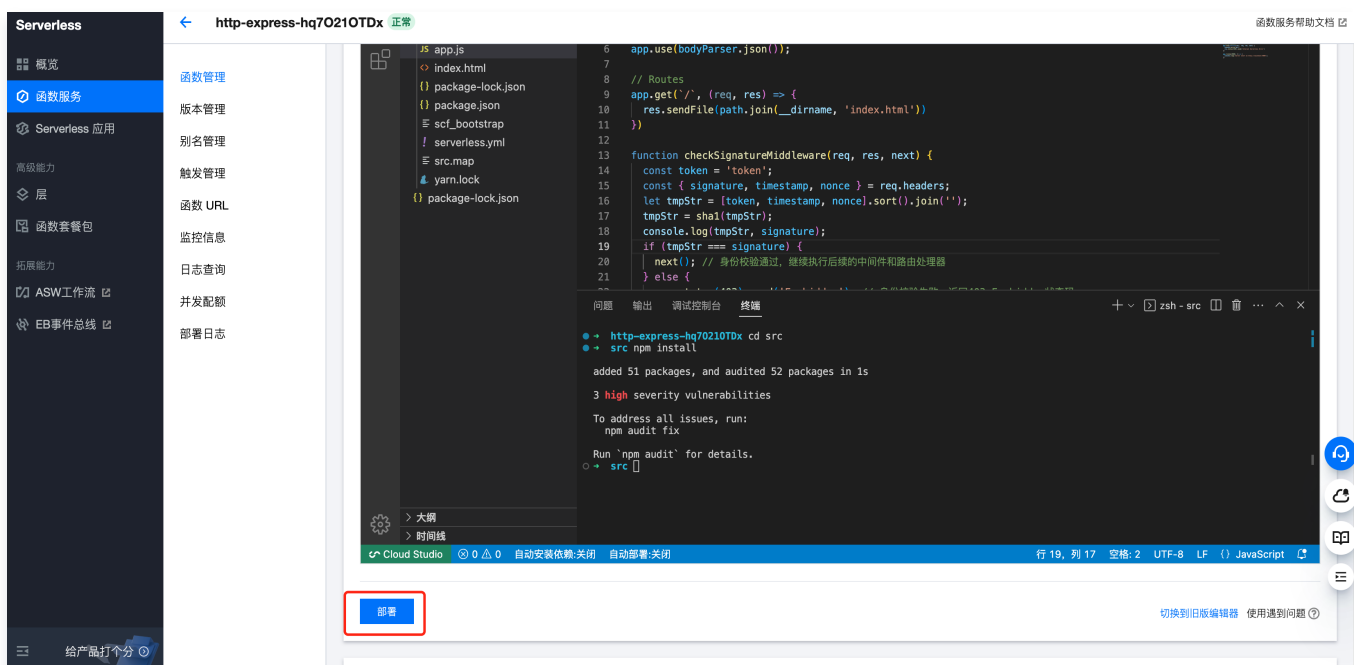
app.get('/test', checkSignatureMiddleware, (req, res) => {
  const { echostr } = req.headers;
  res.set('Content-Type', 'text/plain; charset=utf-8');
  res.send(echostr);
});
```

```
// Note that the data interface received by the rule engine forwarding is of POST type.
app.post('/test', checkSignatureMiddleware, (req, res) => {
  const params = req.body.payload.params;
  console.log(params); // { body_temperature: 36 }
  res.end();
});
```

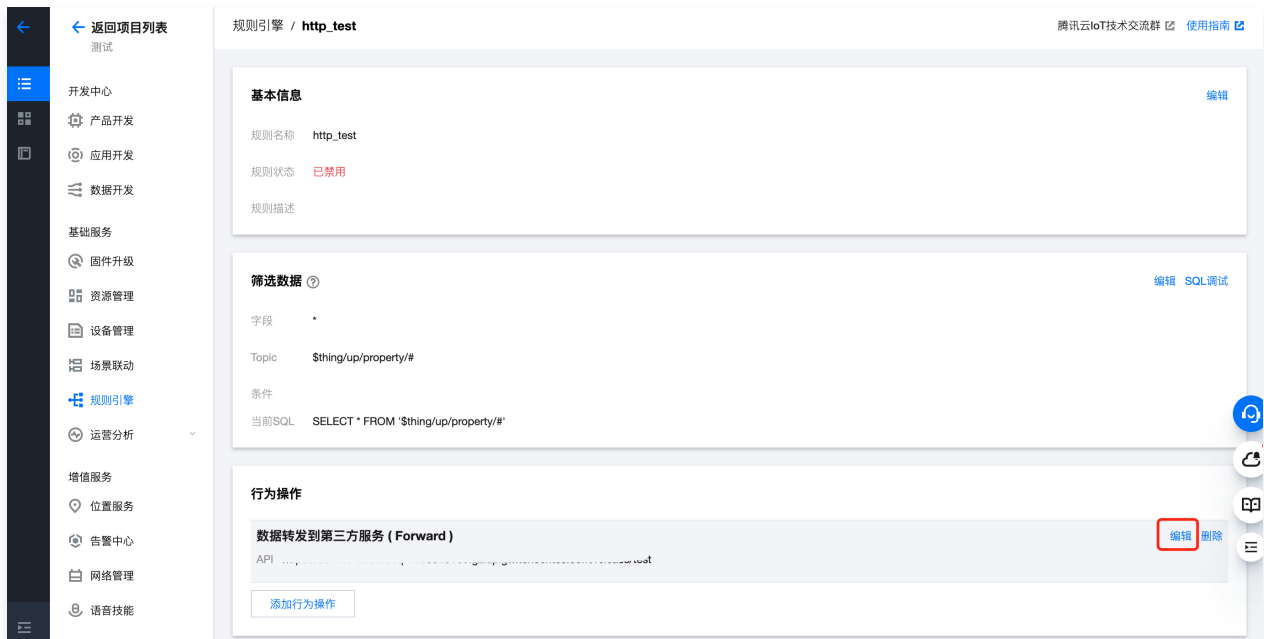
1.6 In Function Service, you can see the corresponding code template. Paste the previous step's code into the editor. Select terminal above the code editor, create a new terminal, then input command `cd src` in the bottom terminal to enter the src directory, next input command `npm install` for dependency installation.



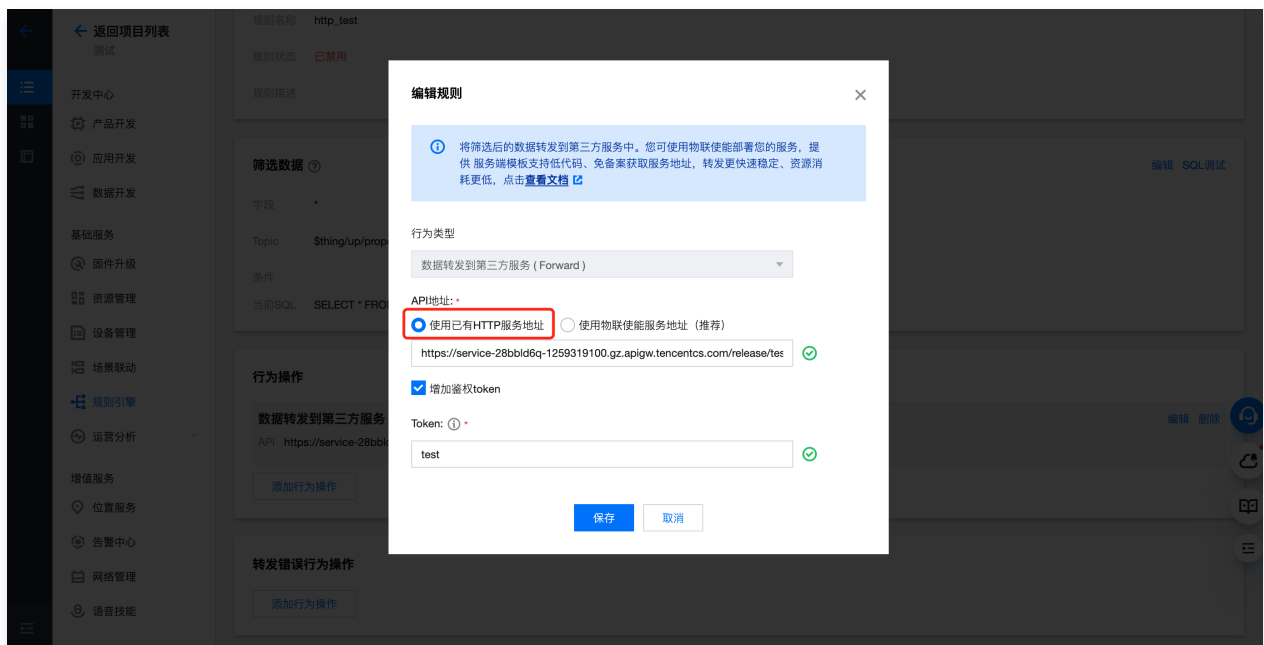
1.7 Wait for the dependency installation to complete. Then click Deploy below to successfully deploy the Web service.



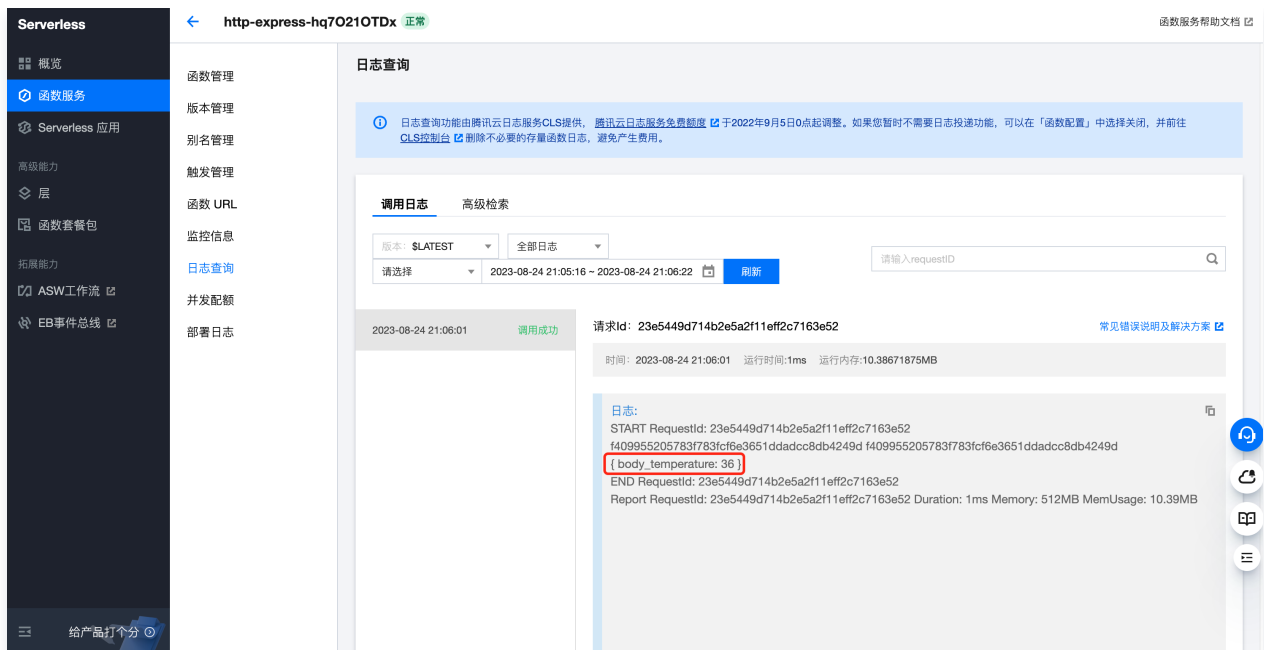
1.8 In the rule engine of the `iot-explorer` console, select `Edit` and fill in the API address deployed in the previous steps.



1.9 In `Edit Rule`, select `"Utilize existing HTTP service address"`, next fill in the API address and Token, then select `Save`. Finally, enable the rule status.



1.10 If there is data forwarded by the rule engine, a preconfigured API will be called. You can see the printed information in the log module of the SCF platform.



Resend Mechanism

The resend mechanism is used to resend again in case of a failure during the message forwarding process to achieve the purpose of receiving messages. as follows:

- If message forwarding fails, the system will perform forwarding retries. The retries will be performed sequentially with time intervals of 1 s, 3 s, and 10 s. If all three retries fail, the message will be discarded.
- If the user has configured "forwarding error behavior operation", after three retry failures, one more message forwarding will be proceeded with according to the configuration of "forwarding error behavior operation". If it still fails, the message will be discarded.

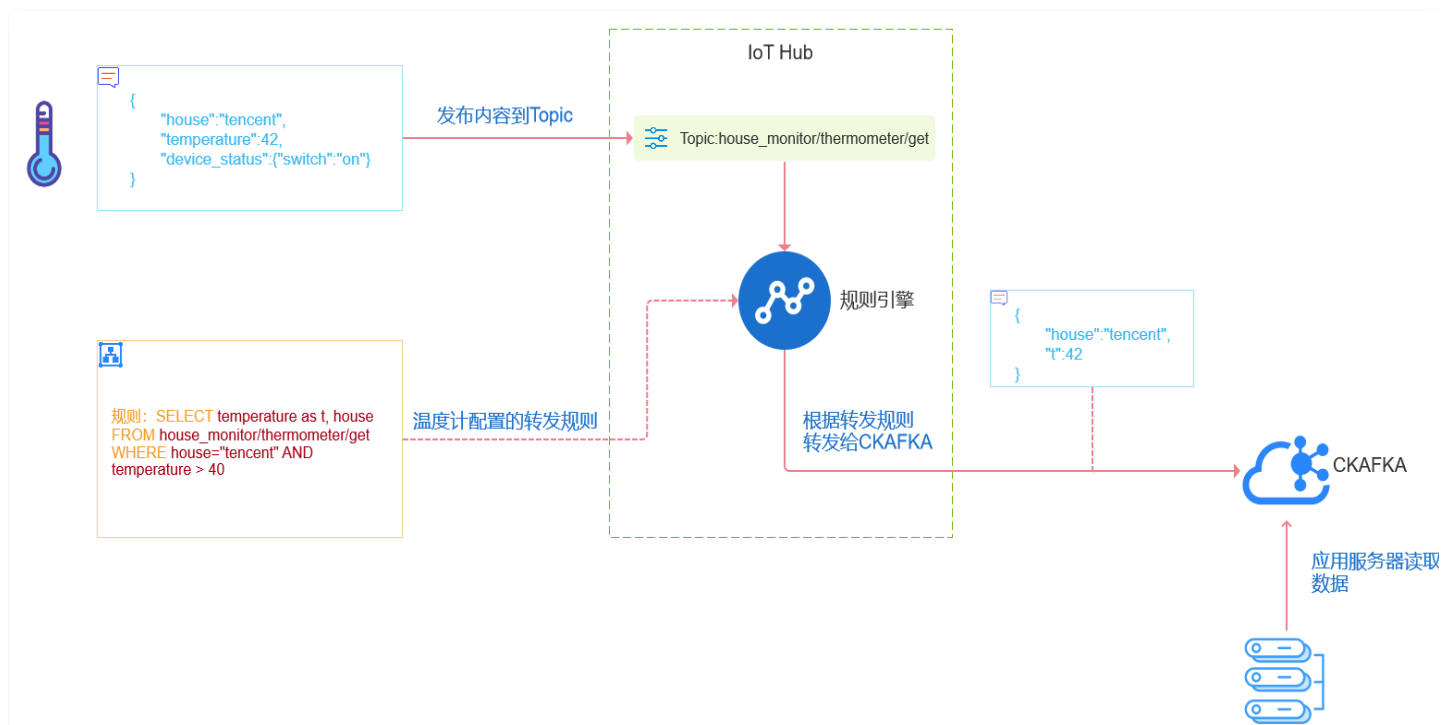
Forward Data to TDMQ CKafka

Last updated: 2025-04-27 17:37:57

Overview

The rule engine supports user configuration rules to forward device-reported data that meets the conditions to [Message Queue CKAFKA](#) (hereinafter referred to as CKAFKA). The user's application server then reads the data content from CKAFKA for processing. This leverages the high I/O throughput advantage of CKAFKA to create a high-availability message link for users.

The following diagram shows the entire process of the rule engine forwarding data to CKAFKA:

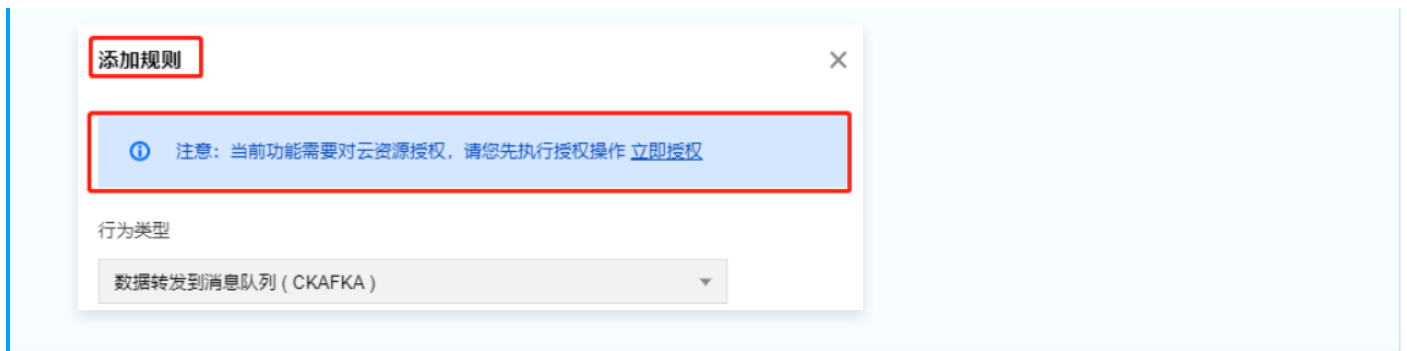


Configuration

1. Log in to the [IoT Explorer Console](#), click the target instance and project name, and select **Data Flow > Rule Engine** from the left menu bar.
2. Open the Rule Engine page and click the rule that needs to be configured.
3. In the Rule Details page, click **Add Behavior Operation**.

Note:

For first-time use, the user will be prompted to authorize access to CKAFKA. You need to click **Authorize access to CKAFKA** to continue creating.



4. In the pop-up "Add Rule" window, select the behavior "Forward Data to Message Queue (CKAFKA)"; sequentially select the CKAFKA instance and Topic, and click **Save**.



5. After completing the above configurations, IoT Explorer will forward the device-reported data that complies with the Rule Condition to the user-configured CKAFKA. You can refer to the [Create Instances and Topic](#) document to read data on the application server and process it.

Resend Mechanism

The resend mechanism is used to resend again in the event of a failure during the message forwarding process to achieve the purpose of receiving messages. Specific instructions are as follows:

- If message forwarding fails, the system will perform forwarding retries. The retries are performed sequentially at intervals of 1 s, 3 s, and 10 s. If all three retries fail, the message will be discarded.
- If the user has configured "forwarding error behavior operation", after three retry failures, one more message forwarding will be proceeded with according to the configuration of "forwarding error behavior operation". If it still fails, the message will be discarded.

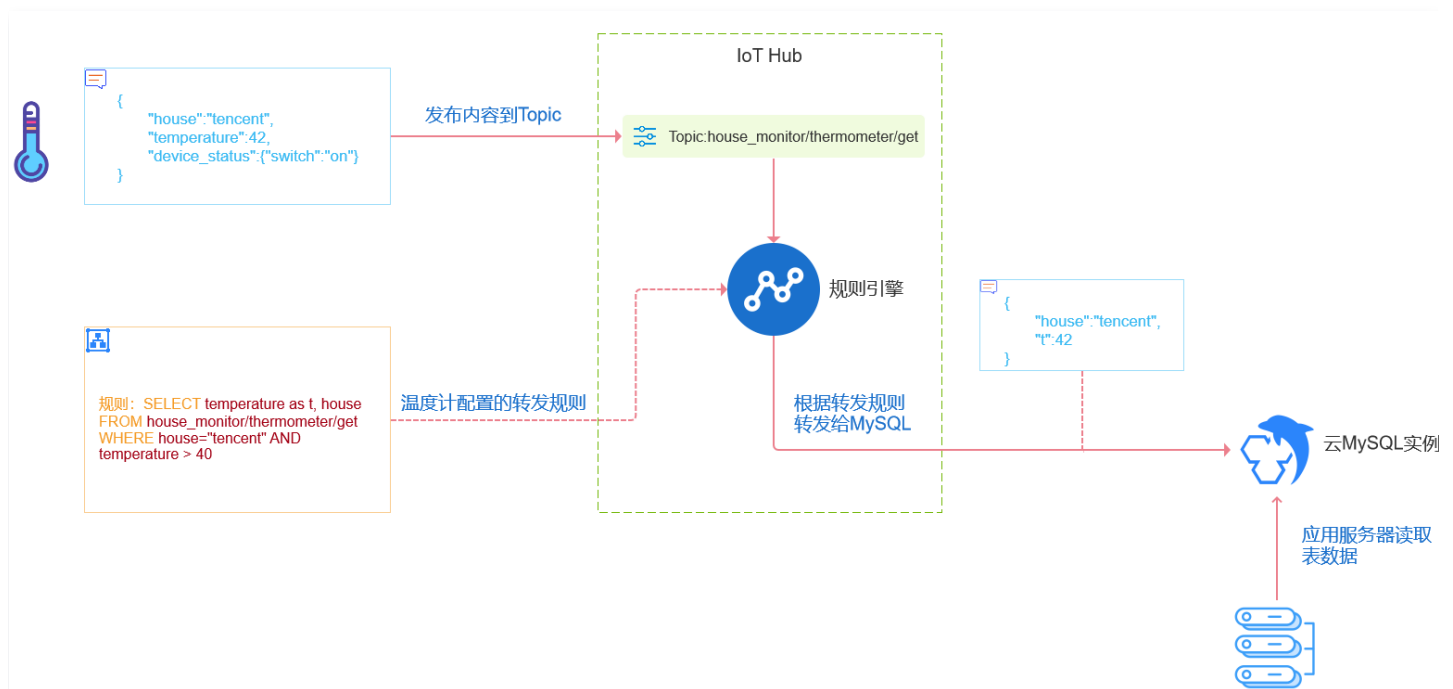
Data Forwarding to Cloud MySQL

Last updated: 2025-04-27 17:38:11

Overview

The rule engine supports user configuration of forwarding rules to forward eligible device-reported data to the cloud component MySQL. You can create a MySQL instance and table in the [MySQL console](#) or by using the TencentCloud API. Then, you can write the specified fields in the device messages into the corresponding MySQL table.

The following diagram shows the entire process of the rule engine forwarding data to MySQL:

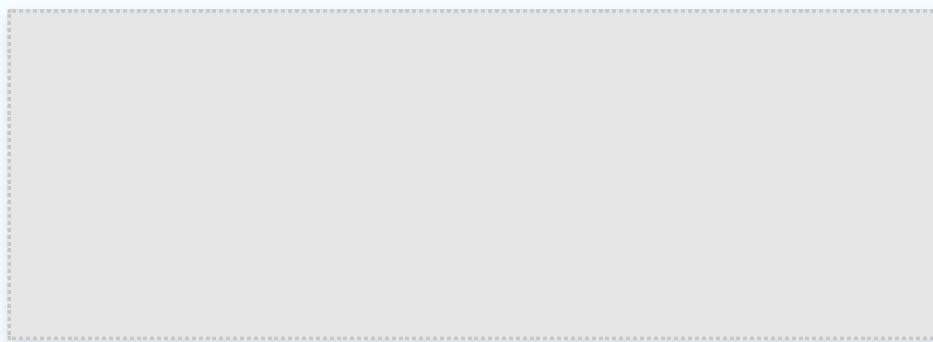


Configuration

1. Log in to the [IoT Explorer console](#), click the target instance and project name, and select **Data Flow > Rule Engine** in the left menu bar.
2. Open the Rule Engine page and click the rule that needs to be configured.
3. On the Rule Detail Page, click **Add Behavior Operation**.

Note:

You will be prompted to authorize access to MySQL if this is your first time using it. Click **Authorize Now** to continue creating.



4. In the pop-up "Add Rule" window, select the "Forward Data to Cloud Database (MySQL)" option. After successful authorization, you need to configure the MySQL instance information and the field information to be written, as shown below. Once configured, click **Save**.

添加规则

行为将数据插入到云数据库 (MySQL) 中, 查看文档

行为类型

数据转发到云数据库 (MySQL)

地域 *

广州

实例 *

cdb- / iothub-test

MySQL数据库 *

sokol

数据表 *

hub_data

实例登录账户 *

root

登录密码 *

.....

数据字段

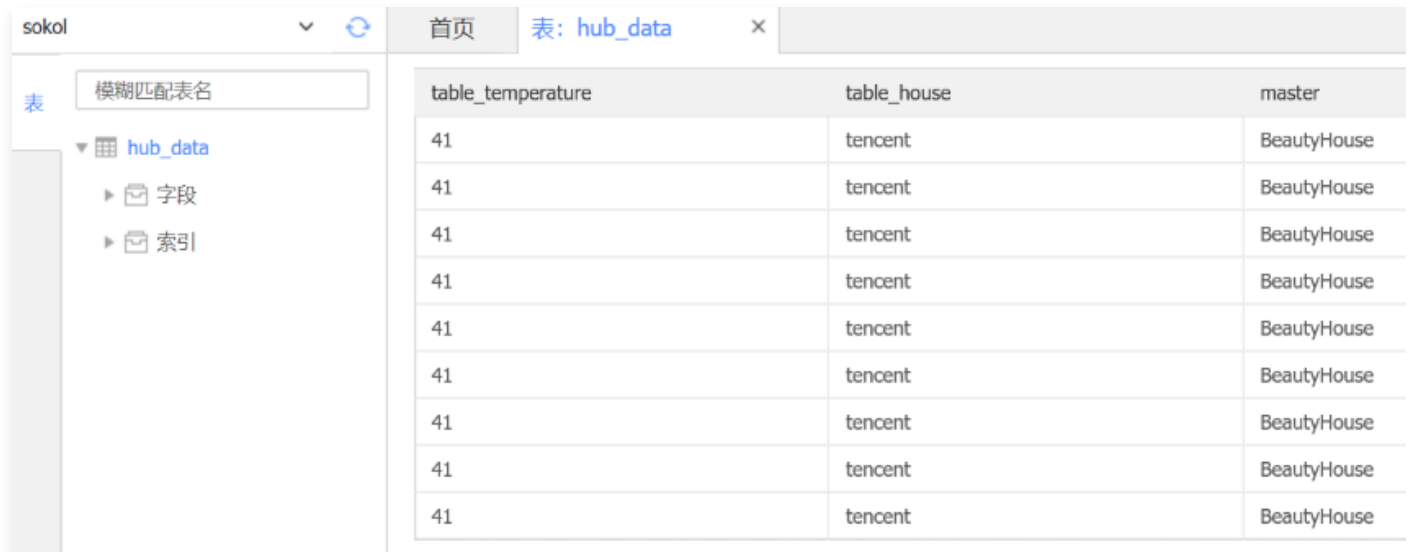
| 字段名称 | 值 | |
|-------------------|-------------|-----|
| table_temperature | \${t} | + - |
| table_house | \${house} | + - |
| master | BeautyHouse | + - |

☐ 使用批量设置

保存

取消

After successful forwarding, the information displayed in MySQL is as shown below:



| table_temperature | table_house | master |
|-------------------|-------------|-------------|
| 41 | tencent | BeautyHouse |
| 41 | tencent | BeautyHouse |
| 41 | tencent | BeautyHouse |
| 41 | tencent | BeautyHouse |
| 41 | tencent | BeautyHouse |
| 41 | tencent | BeautyHouse |
| 41 | tencent | BeautyHouse |
| 41 | tencent | BeautyHouse |
| 41 | tencent | BeautyHouse |

Configuration Instructions

The configuration is divided into the following steps:

1. Select a region and a MySQL instance.
2. Input the username of the newly created MySQL instance.
3. Enter the instance login password.
4. Select the name of the database where you want to write data. If no database has been created under the created MySQL instance, please go to the MySQL console to create a new database. For details, see [Creating Databases and Tables](#).
5. Select the table to write to. If no table has been created in the database, go to the MySQL console to create a new table.
6. Configure the fields to be written. There are two columns here: "Field Name" and "Value". The "Field Name" corresponds to the field in the database table, indicating the field to be written. "Value" indicates the value to be written to the corresponding field. The source of the value can be the message body (note that the message body must be in Json format to support value extraction), or a constant filled in here.

Note:

- If the source is the message body, use "\${}" to reference the fields in the message body. If you want to specify a constant, just fill in the corresponding value, such as a number or a string literal like 5 or hello.
- You need to create the database, tables, and field names in TencentDB for MySQL first before you can successfully write data to the database.

For more details, see [Create database and tables](#).

Resend Mechanism

The resend mechanism is used to resend again in case of a failure during the message forwarding process to achieve the purpose of receiving messages. The specific instructions are as follows:

- If message forwarding fails, the system will perform forwarding retry. Retries will be performed sequentially with time intervals of 1s, 3s, and 10s. If all three retries fail, the message will be discarded.
- If the user has configured "forwarding error behavior operation", after three retry failures, one more message forwarding will be carried out according to the configuration of "forwarding error behavior operation". If it still fails, the message will be discarded.

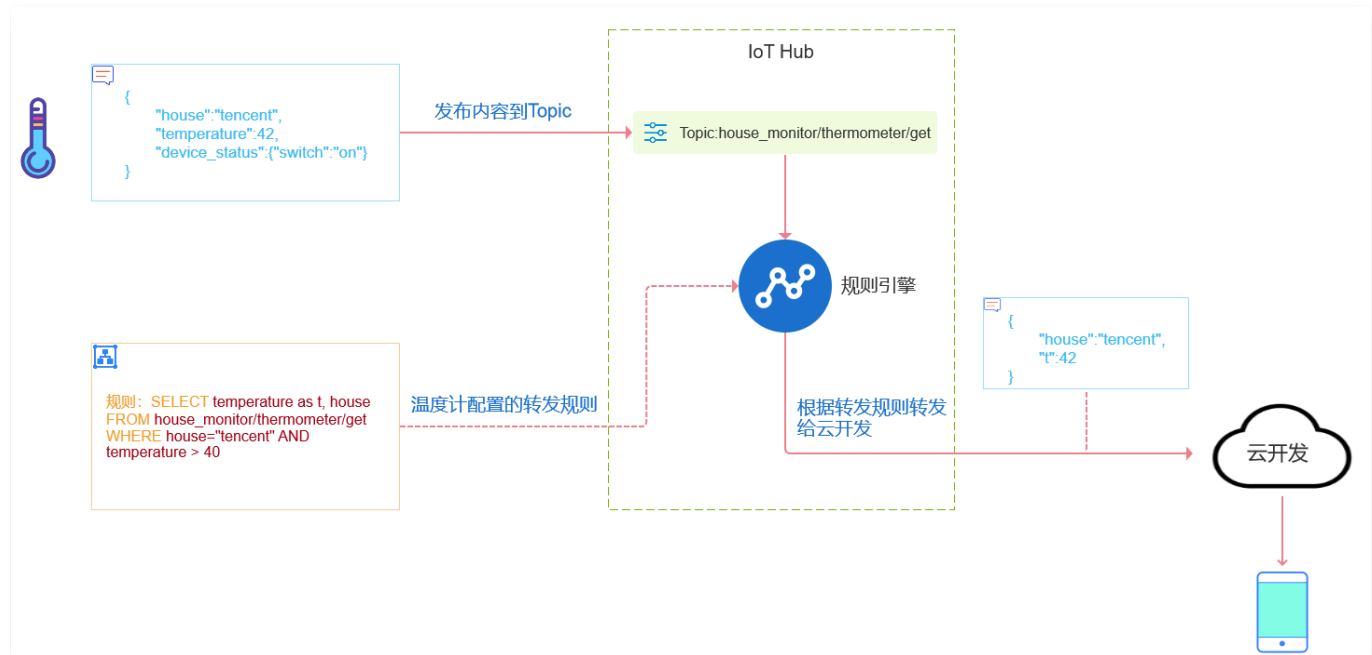
Data Forwarding to Cloud Development

Last updated: 2025-04-27 17:38:25

Overview

The rule engine supports users in configuring forwarding rules to forward eligible device-reported data to cloud development components. You can complete the activation of the TCB environment in [Tencent CloudBase Console](#). For specific operations, see [Activate Environment](#).

The following diagram shows the entire process of the rule engine forwarding data to cloud development:

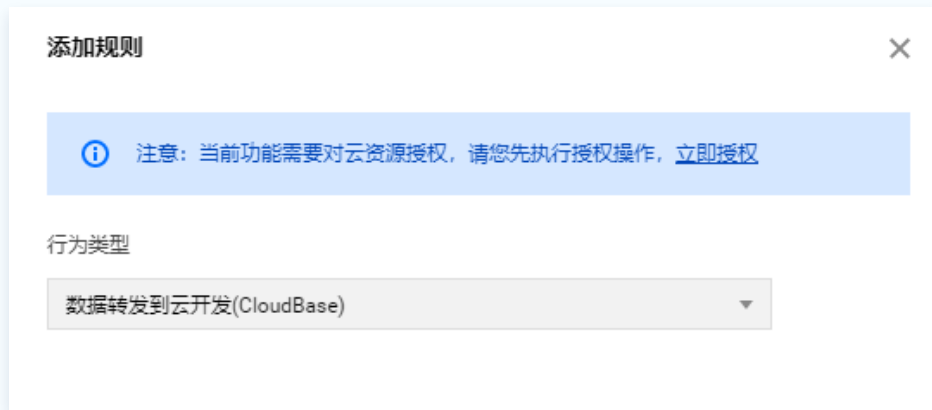


Configuration

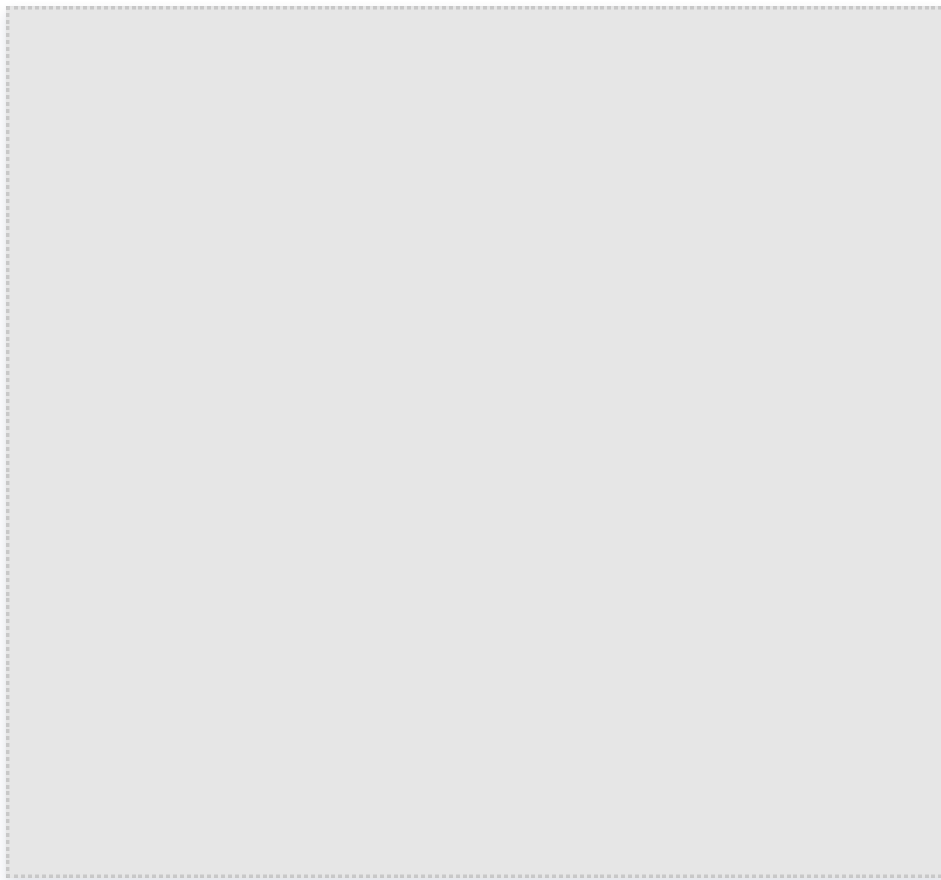
1. Log in to the [IoT Explorer console](#), click the target instance and project name, and select **Data Flow > Rule Engine** in the left menu bar.
2. Open the Rule Engine page and click the rule that needs to be configured.
3. In the Rule Details page, click **Add behavior operation**.

Note:

You will be prompted to authorize access to cloud development for the first-time use. You need to click **Authorize Now** to continue creating.



4. In the pop-up "Add Rule" window, select the "Forward Data to Cloud Development (CloudBase)" option. After selecting the established environment and function, click **Save**.



Note:

Currently, only forwarding data to SCF in cloud development is supported. You need to create a development environment and SCF in cloud development first before you can select the environment and SCF.

Resend Mechanism

The resend mechanism is used to resend again in case of a failure during the message forwarding process, so as to achieve the purpose of receiving messages. The specific instructions are as follows:

- If message forwarding fails, the system will perform forwarding retries. The retries are performed sequentially at intervals of 1 s, 3 s, and 10 s. If all three retries fail, the message will be discarded.
- If the user has configured the "forwarding error behavior operation", after three retry failures, one more message forwarding will be proceeded with according to the configuration of the "forwarding error behavior operation". If it still fails, the message will be discarded.

Data Forwarding to Cloud Component

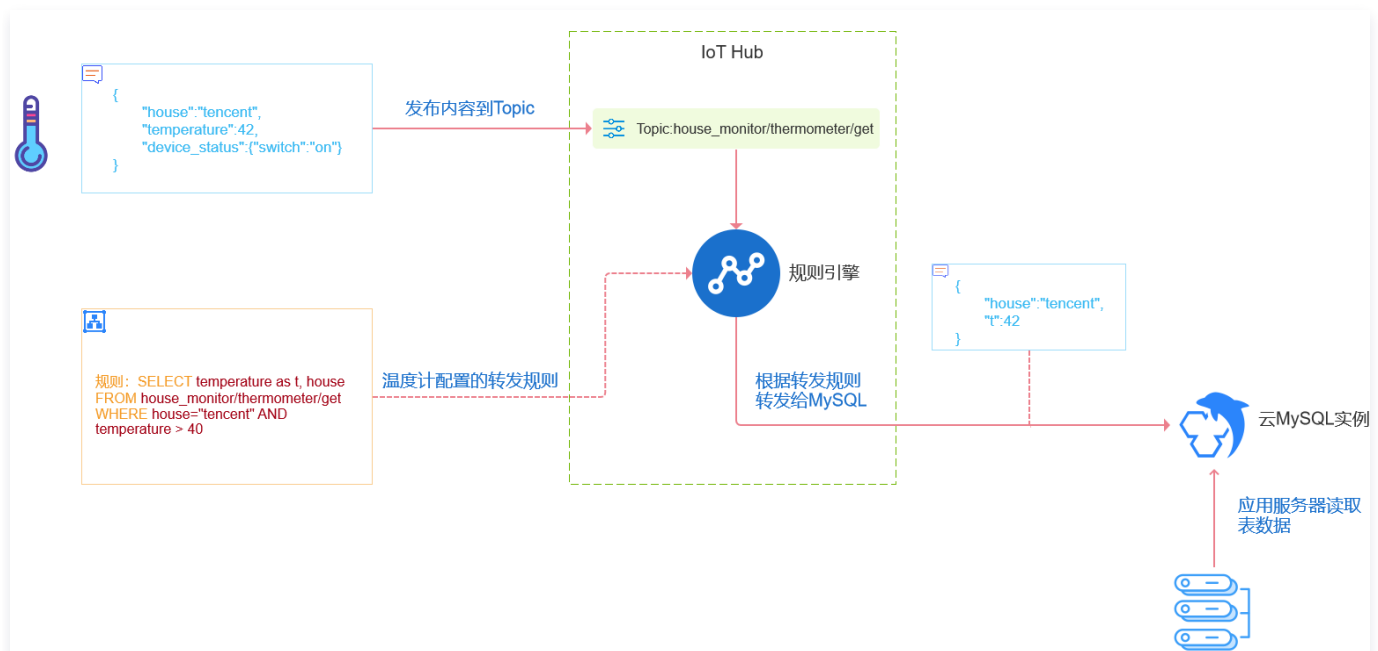
TDSQL-MySQL

Last updated: 2025-04-27 17:38:40

Overview

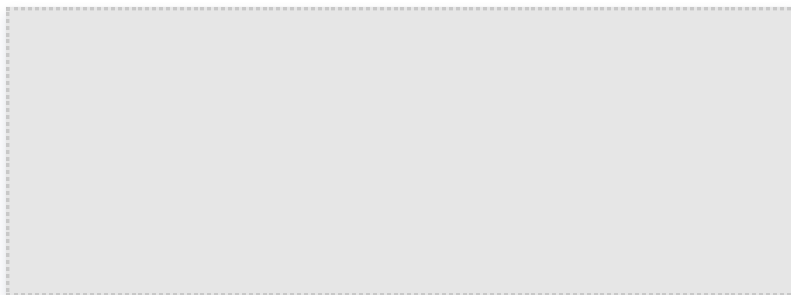
The rule engine supports users in configuring forwarding rules to forward eligible device-reported data to the cloud component TDSQL-MySQL. You can create a TDSQL instance and tables using the [TDSQL console](#) or TencentCloud API. Then, you can write specified fields from device messages into the corresponding TDSQL tables.

The following diagram shows the entire process of the rule engine forwarding data to TDSQL:



Configuration

1. Log in to the [IoT Explorer console](#), click the target instance and project name, and select **Data Flow > Rule Engine** in the left menu bar.
2. Open the Rule Engine page, click the **rule name** that needs to be configured to enter the Rule Detail Page.
3. click **Add Behavior Operation**. If it is the first time you use it, you will be prompted to authorize user access to TDSQL. You need to click **Authorize Now** to continue creating.



4. After successful authorization, in the pop-up "Add Rule" window, select "Forward Data to CloudDB TDSQL for MySQL". You need to configure the TDSQL-MySQL instance information and the field information to be written, as shown below:

添加规则

✕

行为将数据插入分布式数据库TDSQL-MySQL, [查看文档](#)

行为类型

数据转发到分布式数据库TDSQL-MySQL

地域 * 实例 *

广州 请选择实例

Mysql数据库 * 数据表 *

请选择数据库 请选择数据表

实例登录账户 ⓘ * 登录密码 ⓘ *

输入登录账户 输入登录密码

数据字段

| 字段名称 ⓘ | 值 ⓘ |
|--------|-----|
| | |

☐ 使用批量设置 ⓘ

保存 取消

5. Once configured, click **Save** just.

Configuration Instructions

The configuration is divided into the following steps:

1. Select a region and a TDSQL-MySQL instance.
2. Enter the userName of the TDSQL for MySQL instance you just created.
3. Enter the instance login password.
4. Select the name of the database to be written. If no database has been created under the created TDSQL-MySQL instance, please go to the TDSQL-MySQL console to create a new database.
5. Select the table to write to. If no tables have been created in the database, go to the TDSQL-MySQL console to create a new table.
6. Configure the fields to be written. There are two columns here: "Field Name" and "Value". The "Field Name" corresponds to the field in the database table, indicating the field to be written. The "Value" indicates the value to be written to the corresponding field. The source of the value can be the message body (warning: the message body must be in Json format to support value extraction), or a constant filled in here.

Note:

- If the source is the message body, use "\${}" to reference fields in the message body. If you want to specify a constant, directly enter the corresponding value, such as a number or a string literal like 5 or hello.
- First, create a database, tables, and field names in the cloud component TDSQL-MySQL before you can successfully write data to the database.

Resend Mechanism

The resend mechanism is used to resend again in case of failure during the message forwarding process to achieve the purpose of receiving messages. Details are as follows:

- If message forwarding fails, the system will perform forwarding retry. Retries are performed sequentially with time intervals of 1s, 3s, and 10s. If all three retries fail, the message will be discarded.
- If the user has configured "forwarding error behavior operation", after three retry failures, one more message forwarding will be proceeded with according to the configuration of "forwarding error behavior operation". If it still fails, the message will be discarded.

Value-Added Services

Voice Skills

Alexa Voice Skill Service

Last updated: 2025-04-27 17:39:18

This instance uses the third-party platform Amazon Alexa to implement intelligent voice services. For detailed official documentation about Alexa, for details, see [Amazon Alexa Documentation](#).

Prerequisites

Activate Device Integration with Amazon Alexa Only Supports the Following Regions

United States

Voice Skills Only Support the Following Languages

English

Console Activation of Third-Party Voice Skill Service

Step 1: Confirm the Product Scope and Features

When creating a new product, the product category must be selected from the categories supported by the selected platform. If other categories are selected or customized, the Voice Skill Service will be unavailable.

请选择产品品类

产品品类 * 标 已定义标准物模型 免 包含免开发面板

| | | | | |
|------|------|-----------------------------|---|---|
| 智慧生活 | 电工照明 | <input type="radio"/> 洗衣机 | 标 | 无 |
| 智能城市 | 报警传感 | <input type="radio"/> 空调 | 标 | 免 |
| 智慧农业 | 户外出行 | <input type="radio"/> 燃气热水器 | 标 | 无 |
| 智能制造 | 家用电器 | <input type="radio"/> 空气净化器 | 标 | 免 |
| 其他行业 | 网关中控 | <input type="radio"/> 电热水器 | 标 | 无 |
| | 厨房电器 | <input type="radio"/> 冰箱 | 标 | 无 |
| | 运动健康 | <input type="radio"/> 扫地机 | 标 | 无 |
| | 影音娱乐 | <input type="radio"/> 取暖器 | 标 | 无 |
| | 视频服务 | <input type="radio"/> 新风机 | 标 | 无 |
| | | <input type="radio"/> 净水器 | 标 | 无 |

已选择品类: 智慧生活 / 电工照明 / 灯

填写产品信息

产品名称 *

支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

设备类型

通信方式 *

请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

数据协议

描述

最多不超过80个字符

You can view through the following table whether the supported categories and features meet your product development.

| Supported Categories | Functions Supported |
|---|--|
| Smart Life – Electrical Lighting – Lamp | Switch, brightness adjustment, color adjustment, color temperature adjustment. |
| Smart Life – Electrical Lighting – Switch Panel | Switch |
| Smart Life – Electrical Lighting – Socket | Switch |
| Smart Life – Electrical Lighting – Curtain | Switch, percentage adjustment. |
| Smart Life – Household Appliances – Aromatherapy Diffuser | Switch |
| Smart Life – Household Appliances – Sweeping Robot | Switch |

Smart Life – Household Appliances – Air Purifier

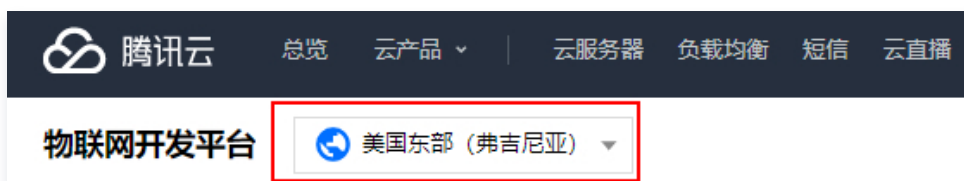
Switch, wind speed (increase or decrease), mode.

Note:

Tencent Cloud IoT Explorer's supported categories or features for Amazon Alexa. More will be supported subsequently. If you have an integration requirement, you can contact [online customer service](#) on Tencent Cloud official website, describe your product requirements and submit an apply for this feature. We will arrange related staff to connect with you.

Step 2: Apply for Enabling Amazon Alexa Service

1. Log in to the [IoT Development Console](#), select East US as the region, create a product. For details, see [Product Definition](#).



2. Click on an item to enter the project detail interface. Click **Voice Skills > Amazon Alexa > Apply for enabling** to enter the application page.



3. Select the product you need to enable, fill in the application information. You also need to check "I understand and agree to the Developer Notice and Authorization". Click **Submit Application**. We will arrange related staff to dock with you.

- **Select product:** Currently created products.
- **Other requirement descriptions:** It can contain up to 250 characters.

申请开通Amazon Alexa服务

选择产品 *

请选择

请选择产品

其他需求描述

最多不超过250个字符

服务说明

- 1.提交申请后，腾讯云商务经理会与您沟通服务费用；
- 2.可用语音控制功能：[点此查看该平台支持的功能和语言列表](#)

☐ 我了解并同意《[开发者须知与授权协议](#)》

提交申请

取消

4. After approval, you can also add new products under the project at **Select Product**.

Amazon Alexa ⓘ

Google Assistant ⓘ

腾讯云小微

小度

Amazon Alexa接入

申请开通

与Amazon Alexa语音平台进行对接，支持用户通过Amazon Alexa音箱等语音设备，对联网设备进行语音控制



使用腾讯连连小程序（免开发），平台根据标准数据格式，提供了一个标准的Amazon Alexa Skill。接入腾讯连连小程序的产品可以直接使用该技能，实现Amazon Alexa音箱等语音设备进行语音控制，支持的设备范围以及可识别的语音指令，可查看[详细介绍](#)

已开通产品

选择产品

支持品类



灯



开关面板



插座



窗帘



香薰机



空气净化器



扫地机器人

5. Select product to add, click **Confirm**.



6. Voice skills will take effect after the product is added and reviewed.

Note:

Use Tencent Lianlian Mini Program to perform device debugging. After network configuration binding your device, you can bind Amazon Alexa based on the following [consumer usage](#) steps to achieve the feature of controlling the device via the speaker.

Consumer Use

Prerequisites

1. Have one Alexa device as well as an Amazon Alexa account that can be used normally.
2. Have one or more smart devices released by IoT Explorer and bind the device using WeChat mini program "Tencent Lianlian".
3. Have a Wi-Fi network that can smoothly access Amazon services.

Operation Steps

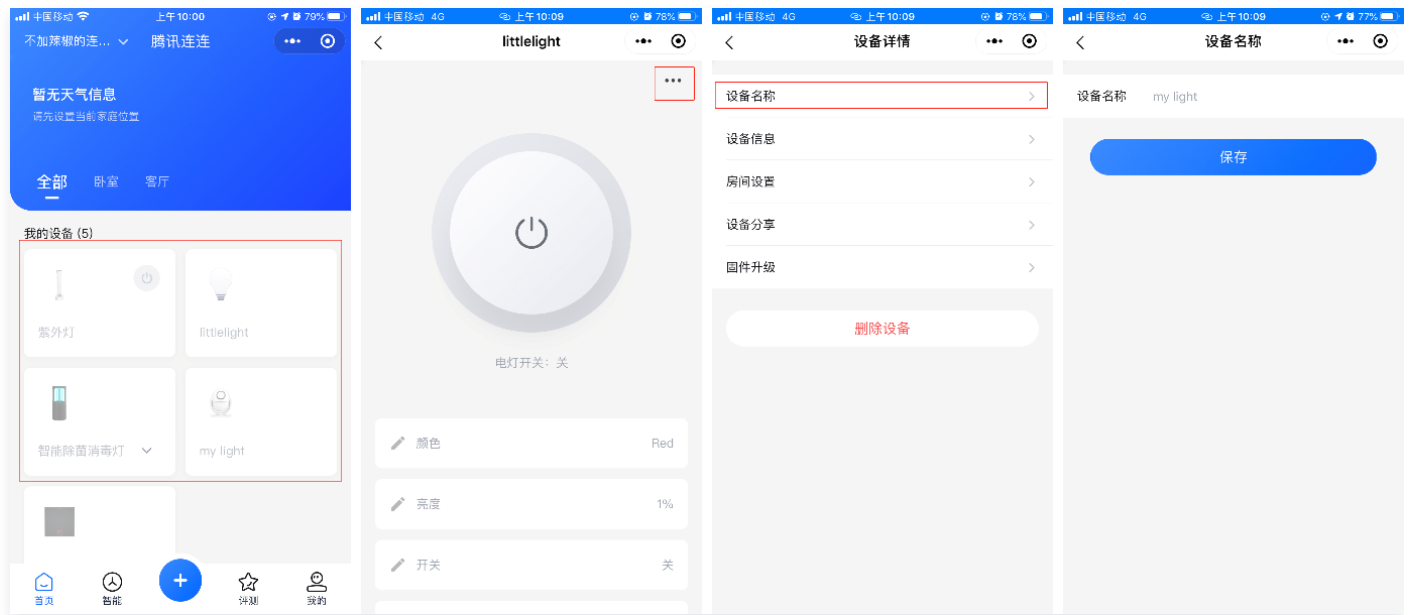
1. When users use WeChat mini program "Tencent Lianlian" to bind smart device products released by IoT Explorer.

Note:

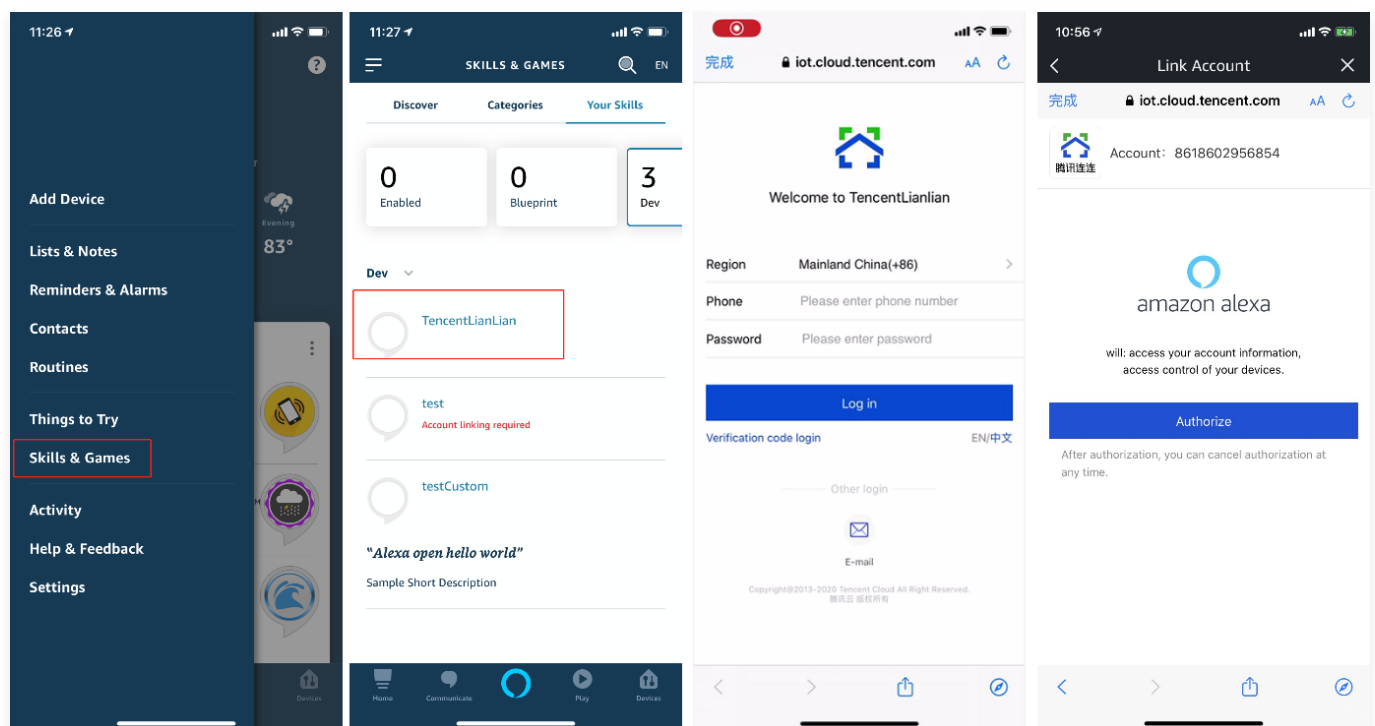
Users who log in using the WeChat mini program "Tencent Lianlian" **need to go to the personal center to bind their mobile number or email number and set the password**. The path is to select **My > Personal Information**, and enter the account and security page to bind the mobile number or email number.

2. **Change the bound device to an English name**, for example: my light. The modified name should avoid using symbols. **The path to modify the device is:** mini program homepage > select the specified product

name > open device detail > click **device name**, after modification, click **save** just.



3. Possess one Amazon Alexa smart speaker, download the Amazon Alexa App and bind the speaker.
4. Log in to the Tencent Lianlian account with the Amazon Alexa App and authorize the control authority of the equipment. Open the menu, select **Skill&Games**, detect Tencent Lianlian, and perform account binding after selecting it.



5. Before controlling the device, the Amazon Alexa speaker needs to discover devices first. You can say to the Amazon Alexa speaker: "Alexa, discover devices".

Note:

If the product name is modified in Tencent Lianlian, the device needs to be rebound in Amazon Alexa speaker.

Use Amazon Alexa speaker to control the product. Functions supported can be found in the following table.

| Category | Function | Speech Example Sentence (Based on Real-World Usage Scenarios) |
|-----------------------|---|---|
| Light | Switch | <ul style="list-style-type: none">• Alexa, turn on the light.• Alexa, turn off the light. |
| Switch panel | Switch | <ul style="list-style-type: none">• Alexa, turn on the switch.• Alexa, turn off the switch. |
| Socket | Switch | <ul style="list-style-type: none">• Alexa, turn on the socket.• Alexa, turn off the socket. |
| Curtain | Switch, percentage adjustment | <ul style="list-style-type: none">• Alexa, turn on curtain.• Alexa, turn off curtain.• Alexa, open close the curtain 50%. |
| Aromatherapy diffuser | Switch | <ul style="list-style-type: none">• Alexa, turn on the Aroma.• Alexa, turn off the Aroma. |
| Sweeping Robot | Switch | <ul style="list-style-type: none">• Alexa, turn on the worker.• Alexa, turn off the worker. |
| Air purifier | Switch, wind speed (increase or decrease), mode | <ul style="list-style-type: none">• Alexa, turn on the purifier.• Alexa, turn off the purifier.• Alexa, set the purifier FanSpeed to 3.• Alexa, set the purifier to 3.• Alexa, setFanSpeed. |

Cloud Xiaowei Voice Skill Service

Last updated: 2025-04-27 17:39:34

this document introduces how to use Tencent Cloud Xiaowei service to control smart devices.

Prerequisites

Only Support Activating Devices in the Following Regions to Interconnect with Tencent Cloud Xiaowei

Chinese mainland (exclude Hong Kong, Macao, and Taiwan region)

Voice Skills Support the Following Languages Only

Chinese

Console Activation of Third-Party Voice Skill Service

Step 1: Confirm the Product Scope and Features

When creating a product, the product category must be a supported category assigned by the selected platform. If you select another category or customize it, the Voice Skill Service will not be available.

请选择产品品类

产品品类

标

已定义标准物模型

免

包含免开发面板

请输入品类

Q

智慧生活

智能城市

智慧农业

智能制造

其他行业

电工照明

报警传感

户外出行

家用电器

网关中控

厨房电器

运动健康

影音娱乐

视频服务

洗衣机

空调

燃气热水器

空气净化器

电热水器

冰箱

扫地机

取暖器

新风机

净水器

标

无

标

免

标

无

标

免

标

无

标

无

标

无

标

无

标

无

已选择品类: 智慧生活 / 电工照明 / 灯

填写产品信息

产品名称

请输入产品名称

支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

设备类型

设备

网关

子设备

通信方式

请选择通信方式

请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

数据协议

物模型

自定义透传

①

描述

选填

最多不超过80个字符

新建产品

取消

You can view through the following table whether the supported categories and features meet your product development.

| Supported Categories | Functions Supported |
|---|--|
| Smart Life – Electrical Lighting – Lamp | Switch, brightness adjustment, color adjustment, color temperature adjustment. |
| Smart Life – Electrical Lighting – switch panel | Switch. |
| Smart Life – Electrical Lighting – Socket | Switch. |
| Smart Life – Electrical Lighting – Curtain | Switch. |
| Smart Life – Household Appliances – Aromatherapy Diffuser | Switch. |

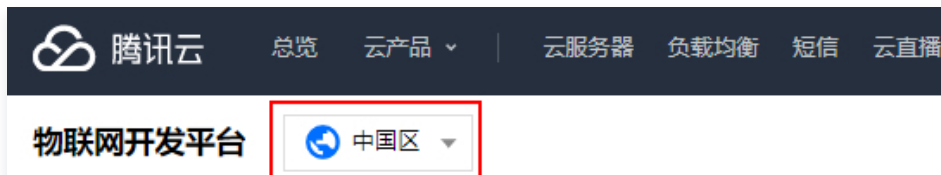
| | |
|--|---|
| Smart Life – Household Appliances – Air Purifier | Switch, wind speed (increase or decrease), mode. |
| Smart Life – Household Appliances – air conditioning | Switch, mode, wind speed (increase or decrease), temperature (increase or decrease, specify temperature). |
| Smart Life – Household Appliances – fan | Switch, wind speed (increase or decrease). |
| Smart Life – Household Appliances – Sweeping Robot | Switch. |
| Smart Life – Household Appliances – TV | Switch, channel adjustment, audio volume adjustment. |

Note:

Tencent Cloud IoT Explorer will gradually add categories or features supported by Tencent Cloud Xiaowei. If you have an integration need, you can visit Tencent Cloud official website and contact [online customer service](#) to describe your product requirements and submit an apply for this feature. Our staff will dock with you.

Step 2: Apply for Activating Tencent Cloud Xiaowei Service

1. Log in to the [IoT Development Console](#), select China as the region, create a product. For details, refer to [Product Definition](#).



2. Click on the item to enter the project detail interface. Click **Voice Skills > Tencent Cloud Xiaowei > Apply for Enabling** to enter the application page.

Amazon Alexa

Google Assistant

腾讯云小微 ⓘ

小度 ⓘ

腾讯云小微接入

申请开通

与腾讯云小微语音平台进行对接，支持用户通过腾讯云小微音箱等语音设备，对联网设备进行语音控制

 **腾讯云小微**

使用腾讯连连小程序（免开发），平台根据标准数据格式，提供了一个标准的腾讯云小微 Skill。接入腾讯连连小程序的产品可以直接使用该技能，实现腾讯云小微音箱等语音设备进行语音控制，支持的设备范围以及可识别的语音指令，可查看[详细介绍](#)

已开通产品

选择产品

支持品类

 灯

 开关面板

 插座

 窗帘

 香薰机

 空气净化器

 空调

 风扇

 扫地机器人

 电视

3. Select the product you need to activate, fill in the application information, and also check "I understand and agree to the Developer Notice and Authorization". Click **Submit Application**. We will arrange for related staff to dock with you.

- **Select product:** Currently created products.
- **Other requirement descriptions:** with a maximum of 250 characters.

申请开通腾讯云小微服务

选择产品 *

请选择

请选择产品

其他需求描述

最多不超过250个字符

服务说明

1.提交申请后，腾讯云商务经理会与您沟通服务费用；
2.可用语音控制功能：[点此查看该平台支持的功能和语言列表](#)

☐ 我了解并同意《[开发者须知与授权协议](#)》

提交申请

取消

4. After approval, you can also add new products under the project in **Select Product**.

5. Select product to add, click **Confirm**.

选择产品

选择产品 *

请选择产品

所有产品 电视机, 智能窗帘, 智能开关, 空调, 风扇, 智能台灯, 扫地机器人,

确定 **取消**

6. Voice skills will take effect after the product addition is approved.

Note:

Use Tencent Lianlian Mini Program to perform device debugging. After Network Configuration Binding your device, you can bind Tencent Cloud Xiaowei based on the following [consumer usage](#) steps to achieve the feature of controlling devices via the speaker.

Consumer Use

Prerequisite

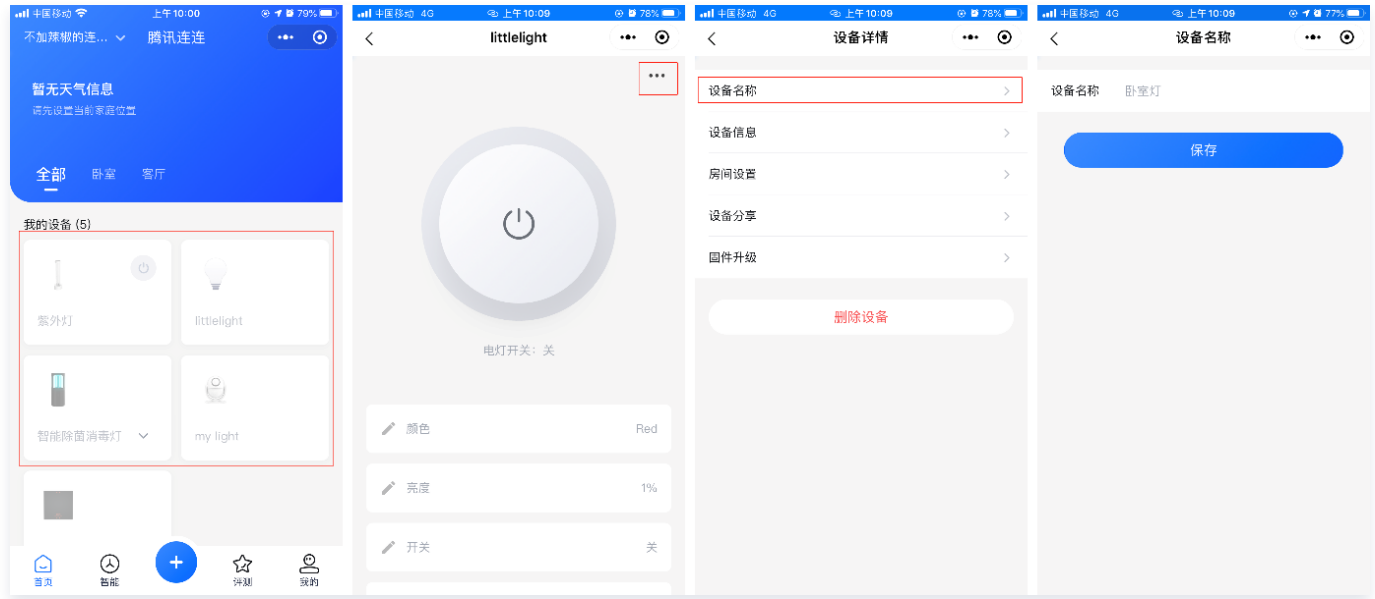
1. Possess a device with Cloud Xiaowei voice skills, such as: smart speaker, voice smart light, etc.
2. Possess one or more smart devices released by IoT Explorer and bind the device using WeChat mini program "Tencent Lianlian".
3. Use the third-party product to bind the Tencent Lianlian Account and authorize device usage. Perform the specific binding process according to the notifications of the third-party platform product.

Note:

For specific consumer tutorial, please follow the guidance of third-party vendors.

Operation Steps

1. After applying for a Tencent Lianlian Account, bind a mobile number or email number and set a password.
2. Voice skill products added in Tencent Lianlian Mini Program need to modify the device name. For example: bedroom light. The modified name should avoid using symbols.



Use Tencent Cloud Xiaowei to control smart devices. Functions supported can be found in the following table.

| Category | Functionality | Speech Examples (Based on Real-World Usage Scenarios) |
|------------------------|--|---|
| Light | Switch, brightness adjustment, color temperature adjustment, color adjustment. | <ul style="list-style-type: none">• Turn on the light, turn off the light.• Turn on the bedroom light, turn off the bedroom light.• Brighten the light a bit.• Adjust the light color temperature to be warmer.• Set the light to blue. |
| Switch panel | Switch | <ul style="list-style-type: none">• Turn on the switch.• Turn off the switch. |
| Socket | Switch | <ul style="list-style-type: none">• Turn on the socket.• Turn off the socket. |
| Curtain | Switch | <ul style="list-style-type: none">• Open the curtain.• Close the curtain.• Open the curtain to 50%. |
| Aromatherapy diffuser | Switch | <ul style="list-style-type: none">• Turn on the aromatherapy diffuser.• Turn off the aromatherapy diffuser. |
| Floor – cleaning robot | Switch | <ul style="list-style-type: none">• Turn on the floor – cleaning robot.• Turn off the floor – cleaning robot. |
| Air purifier | Switch, wind speed (increase or decrease), mode. | <ul style="list-style-type: none">• Turn on the air purifier.• Turn off the air purifier.• Increase the wind speed. |

| | | |
|------------------|---|---|
| | | <ul style="list-style-type: none">• Decrease the wind speed.• Turn on sleep mode. |
| Air conditioning | Switch, mode, wind speed (increase or decrease), temperature (increase or decrease, specify temperature). | <ul style="list-style-type: none">• Turn on air conditioning, turn off air conditioning.• Turn on cooling mode.• Increase the temperature, decrease the temperature, set the temperature to 26 degrees. |
| TV | Switch, channel adjustment, audio volume adjustment. | <ul style="list-style-type: none">• Turn on TV, turn off TV.• Next channel, previous channel. I want to watch CCTV-1.• Increase the volume, decrease the volume, mute the TV. |
| fan | Switch, wind speed (increase or decrease). | <ul style="list-style-type: none">• Turn on the fan, turn off the fan.• Increase the wind speed, decrease the wind speed. |

Google Voice Skill Service

Last updated: 2025-04-27 17:39:49

This document introduces how to open the Google Assistant voice intelligent service to control smart devices through the Google Home speaker.

Prerequisites

Activate Device Integration with Google Assistant Is Only Supported in the Following Regions

United States

Voice Skill Supported Languages

- English
- Chinese (partially semantic)

Console Activation of Third-Party Voice Skill Service

Step 1: Confirm the Product Range and Features

When creating a product, the product category must be selected from the platform-specified supported categories. If other categories are selected or customized, the Voice Skill Service will be unavailable.

请选择产品品类

产品品类 *

标

已定义标准物模型

免

包含免开发面板

请输入品类

Q

智慧生活

智能城市

智慧农业

智能制造

其他行业

电工照明

报警传感

户外出行

家用电器

网关中控

厨房电器

运动健康

影音娱乐

视频服务

精选-筒灯

双模筒射灯电源

精选-窗帘

灯

五路灯

一路开关

精选-射灯

智能无线开关（一键）

RGB灯带

智能无线开关（双键）

标

免

标

免

标

免

标

无

标

免

标

免

标

无

标

免

已选择品类：智慧生活 / 电工照明 / 灯

填写产品信息

产品名称 *

请输入产品名称

支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

设备类型

设备

网关

子设备

通信方式 *

请选择通信方式

请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

数据协议

物模型

自定义透传

?

描述

选填

最多不超过80个字符

新建产品

取消

You can view through the following table whether the supported categories and features meet your product development.

| Supported Categories | Functions Supported (Get Status & Command) |
|---|--|
| Smart Life – Electrical Lighting – Lamp | Switch, color adjustment, brightness adjustment. |
| Smart Life – Electrical Lighting – Switch Panel | Switch. |
| Smart Life – Electrical Lighting – Socket | Switch. |
| Smart Life – Electrical Lighting – Curtain | Switch, percentage adjustment. |
| Smart Life – Household Appliances – Aromatherapy Diffuser | Switch. |

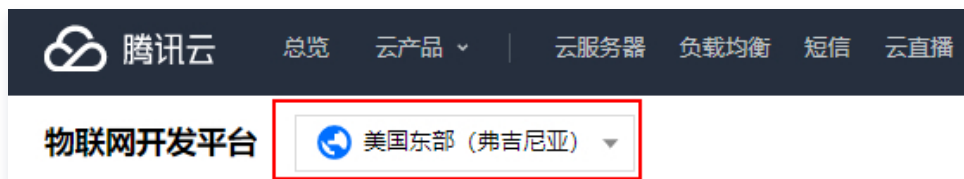
| | |
|--|---------|
| Smart Life – Household Appliances – Sweeping Robot | Switch. |
| Smart Life – Household Appliances – Fan | Switch |

Note:

Tencent Cloud IoT Platform's supported categories or features regarding Google Assistant will support more in the future. If you have an integration need, you can describe your product requirements and submit a ticket through [online customer service](#) on Tencent Cloud official website. We will arrange related staff to dock with you.

Step 2: Apply for Enabling Google Assistant Service

1. Log in to the [IoT Explorer](#), select the United States as the region, create a product. For details, see [Product Definition](#).



2. Click on an item to enter the project detail interface. Click **Voice Skills > Google Assistant > Apply** to enter the application page.



3. Select the product you need to activate, fill in the application information. You also need to check "I understand and agree to the Developer Notice and Authorization", click Submit Application. We will arrange related staff to docking with you.
 - **Select product:** Currently created products.

- Other requirement descriptions: It can contain up to 250 characters.

申请开通Google Assistant服务

选择产品 *

请选择

请选择产品

其他需求描述

最多不超过250个字符

服务说明

1.提交申请后，腾讯云商务经理会与您沟通服务费用；

2.可用语音控制功能：[点此查看该平台支持的功能和语言列表](#)

☐ 我了解并同意 [《开发者须知与授权协议》](#)

提交申请

取消

4. After approval, you can also add new products under your project in **Select Product**.

Amazon Alexa ⓘ

Google Assistant ⓘ

腾讯云小微

小度

Amazon Alexa接入

申请开通

与Amazon Alexa语音平台进行对接，支持用户通过Amazon Alexa音箱等语音设备，对联网设备进行语音控制

 amazon alexa

使用腾讯连连小程序（免开发），平台根据标准数据格式，提供了一个标准的Amazon Alexa Skill。接入腾讯连连小程序的产品可以直接使用该技能，实现Amazon Alexa音箱等语音设备进行语音控制，支持的设备范围以及可识别的语音指令，可查看[详细介绍](#)

已开通产品

选择产品

支持品类

灯

开关面板

插座

窗帘

香薰机

空气净化器

扫地机器人

5. Select product to add, click **Confirm**.



6. Voice skills will take effect after adding product is approved.

Note:

Use Tencent Lianlian Mini Program for device debugging. After performing Network Configuration Binding on your device, you can bind Google Home based on the following [consumer usage](#) steps to achieve the feature of controlling devices via the speaker.

Consumer Use

Prerequisites

1. Have one Google Home device as well as a login account for the Google Home App.
2. Have one or more smart devices released by IoT Explorer and bind the device using WeChat mini program "Tencent Lianlian".
3. Wi-Fi network that can access Google Services.

Usage Steps

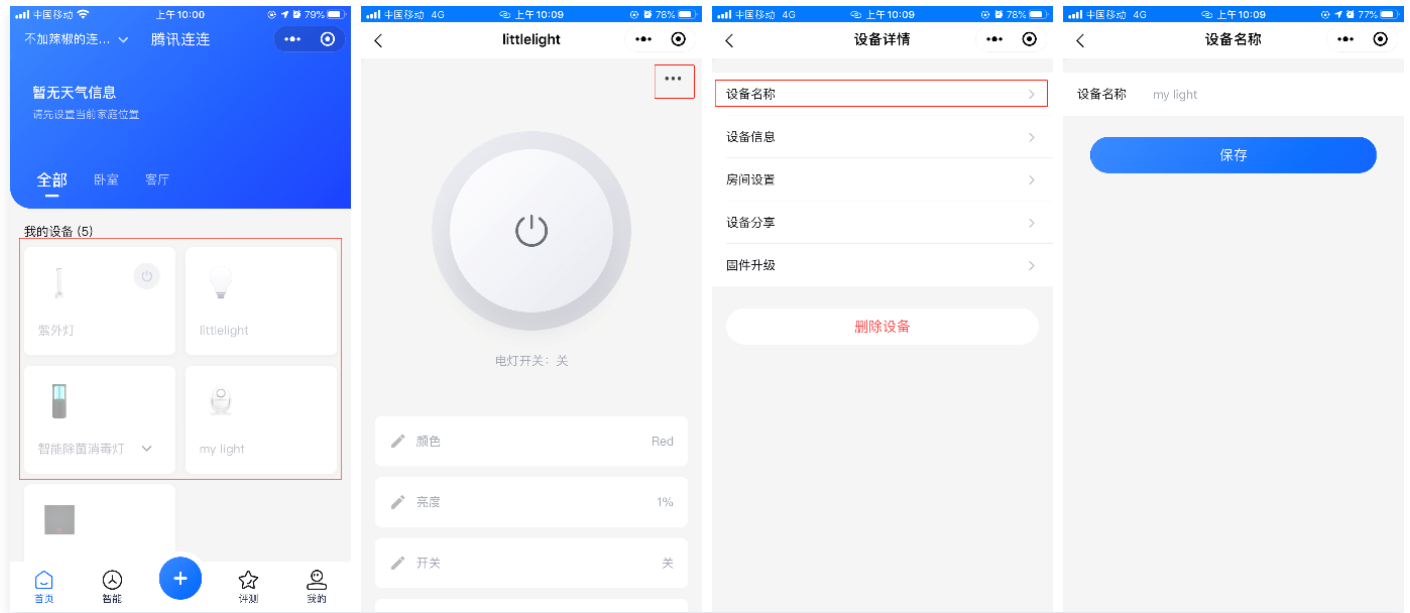
1. When users use WeChat mini program "Tencent Lianlian" to bind smart device products released by IoT Explorer.

Note:

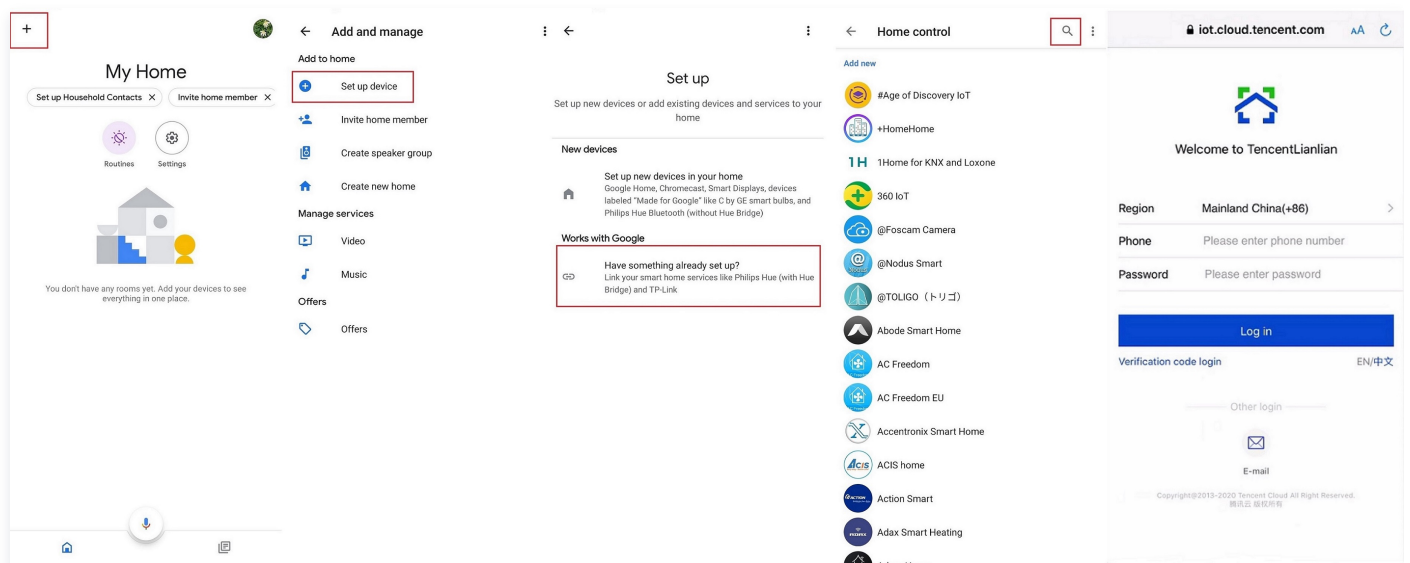
Users who log in using the WeChat mini program "Tencent Lianlian" need to go to the personal center to bind their mobile number or email number and set a password. The path is to select **My > Personal Information**, enter the account and security page, and then bind the mobile number or email number.

2. **Change the bound device to an English name**, for example: my light. The modified name should avoid using symbols. **The device modification path is:** Mini program homepage > Select the specified product

name > Open device detail > Click **Device name**. After modification, click **Save**.



3. Download and install Google Home or Google Assistant App and bind Google Home smart speaker.
4. On the home page of Google Home App, click the "+" button to add a device. Select **Works with Google** in the **Set up devices** list. Search for and select "tencentlianlian". Log in to your Tencent Lianlian account to perform account binding and grant control authority of the device. After successful binding, your device will be displayed in the Devices list of Home Control.



Use Google Home speaker control to use the product. Functions supported can be found in the following table.

| Category | Function | Speech Example Sentences (Based on Real-World Usage Scenarios) |
|----------|--|---|
| Light | Switch, color adjustment, brightness adjustment. | <ul style="list-style-type: none"> • Hey/Ok Google, turn on bedroom light. • Hey/Ok Google, turn off bedroom light. • Hey/Ok Google, set bedroom light to blue. • Hey/Ok Google, brighten bedroom light |

| | | |
|-----------------------|--------------------------------|--|
| Switch panel | Switch. | <ul style="list-style-type: none">• Hey/Ok Google,turn on the switch.• Hey/Ok Google,turn off the switch. |
| Socket | Switch. | <ul style="list-style-type: none">• Hey/Ok Google,turn on the socket.• Hey/Ok Google,turn off the socket. |
| Curtain | Switch, percentage adjustment. | <ul style="list-style-type: none">• Hey/Ok Google,turn on the curtain.• Hey/Ok Google,turn off the curtain.• Hey/Ok Google,open/close the curtain 50%. |
| Aromatherapy diffuser | Switch. | <ul style="list-style-type: none">• Hey/Ok Google,turn on the Aroma.• Hey/Ok Google,turn off the Aroma. |
| Sweeping Robot | Switch. | <ul style="list-style-type: none">• Hey/Ok Google,turn on the worker.• Hey/Ok Google,turn off the worker. |
| fan | Switch. | <ul style="list-style-type: none">• Hey/Ok Google,turn on the fan.• Hey/Ok Google,turn off the fan. |

Xiaodu Voice Skill Service

Last updated: 2025-04-27 17:40:06

This document describes how to open Baidu Xiaodu's third-party voice service to achieve control of smart devices through Xiaodu speakers.

Prerequisites

Activate Device Integration with Xiaodu Is Only Supported in the Following Regions

Chinese mainland (exclude Hong Kong, Macao, and Taiwan region)

Voice Skill Supported Languages

Chinese

Console Activation of Third-Party Voice Skill Service

Step 1: Confirm the Product Scope and Features

When creating a product, the product category must be selected from the categories supported by the designated platform. If you select another category or customize it, the Voice Skill Service will be unavailable.

请选择产品品类

产品品类

标

已定义标准物模型

免

包含免开发面板

请输入品类

Q

智慧生活

智能城市

智慧农业

智能制造

其他行业

电工照明

报警传感

户外出行

家用电器

网关中控

厨房电器

运动健康

影音娱乐

视频服务

精选-筒灯

双模筒射灯电源

精选-窗帘

灯

五路灯

一路开关

精选-射灯

智能无线开关（一键）

RGB灯带

智能无线开关（双键）

标

免

标

免

标

免

标

无

标

免

标

免

标

无

标

免

已选择品类: 智慧生活 / 电工照明 / 灯

填写产品信息

产品名称

请输入产品名称

支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

设备类型

设备

网关

子设备

通信方式

请选择通信方式

请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

数据协议

物模型

自定义透传

①

描述

选填

最多不超过80个字符

新建产品

取消

You can view through the following table whether the supported categories and features meet your product development.

| Supported Categories | Functions Supported (Get Status & Command) |
|---|--|
| Smart Life – Electrical Lighting – Lamp | Switch, color adjustment, brightness adjustment, color temperature adjustment. |
| Smart Life – Electrical Lighting – Switch Panel | Switch |
| Smart Life – Electrical Lighting – Socket | Switch |
| Smart Life – Electrical Lighting – Curtain | Switch |
| Smart Life – Household Appliances – | Switch |

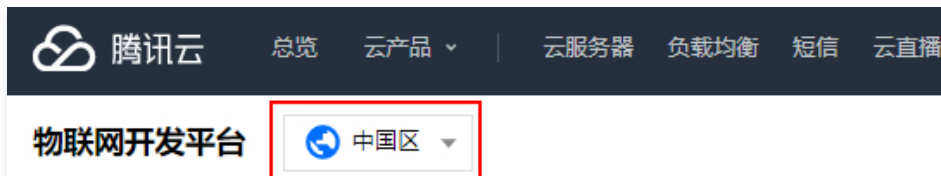
| | |
|--|--|
| Aromatherapy Diffuser | |
| Smart Life – Household Appliances – Sweeping Robot | Switch |
| Smart Life – Household Appliances – Air Purifier | Switch, wind speed (increase or decrease), mode. |
| Smart Life – Household Appliances – air conditioning | Switch, mode, wind speed (increase or decrease), temperature (increase or decrease, specify temperature), swing. |
| Smart Life – Household Appliances – TV | Switch, suspend, resume, audio volume adjustment, channel adjustment. |
| Smart Life – Household Appliances – fan | Switch, wind speed (increase or decrease), mode. |

Note:

Tencent Cloud IoT Development Platform currently supports certain categories and features of Baidu Xiaodu. More will be supported in the future. If you have an integration need, you can visit Tencent Cloud official website and contact Online Customer Service to describe your product requirements and submit a ticket. We will arrange related staff to dock with you.

Step 2: Apply for Enabling Baidu Xiaodu Service

1. Log in to the [IoT Explorer](#), select China as the region, create a product. For details, refer to [product definition](#).



2. Click on a project to enter the project details interface. Click **Voice Skills > Xiaodu > Apply for Enabling** to enter the application page.

Amazon Alexa ⓘ

Google Assistant ⓘ

腾讯云小微

小度

Amazon Alexa接入

申请开通

与Amazon Alexa语音平台进行对接，支持用户通过Amazon Alexa音箱等语音设备，对联网设备进行语音控制

 amazon alexa

使用腾讯连连小程序（免开发），平台根据标准数据格式，提供了一个标准的Amazon Alexa Skill。接入腾讯连连小程序的产品可以直接使用该技能，实现Amazon Alexa音箱等语音设备进行语音控制，支持的设备范围以及可识别的语音指令，可查看[详细介绍](#)

已开通产品

选择产品


支持品类

 灯

 开关面板

 插座

 窗帘

 香薰机

 空气净化器

 扫地机器人

3. Select the products requiring activation. After filling in the application information, you also need to check "I understand and agree to the Developer Notice and Authorization". Click **submit the application**. We will arrange for related staff to dock with you.

- **Select product:** Currently created products.
- **Other requirement descriptions:** It can contain up to 250 characters.

申请开通小度服务

选择产品 *

请选择

请选择产品

其他需求描述

最多不超过250个字符

服务说明

1.提交申请后，腾讯云商务经理会与您沟通服务费用；

2.可用语音控制功能：[点此查看该平台支持的功能和语言列表](#)

☐ 我了解并同意 [《开发者须知与授权协议》](#)

提交申请

取消

4. After approval, you can also add new products under the project at "Select Product".



5. Select product to add, click **Confirm**.



6. After the product addition is approved, the voice skills will take effect.

Note:

Use Tencent Lianlian Mini Program to perform device debugging. After Network Configuration Binding your device, you can bind Xiaodu speaker based on the following [consumer usage](#) procedure to achieve the feature of controlling the device via the speaker.

Consumer Use

Prerequisites

1. Have one speaker device equipped with Xiaodu voice service and a login account for the Xiaodu Speaker App.
2. Have one or more smart devices released by IoT Explorer and bind the device using WeChat mini program "Tencent Lianlian";

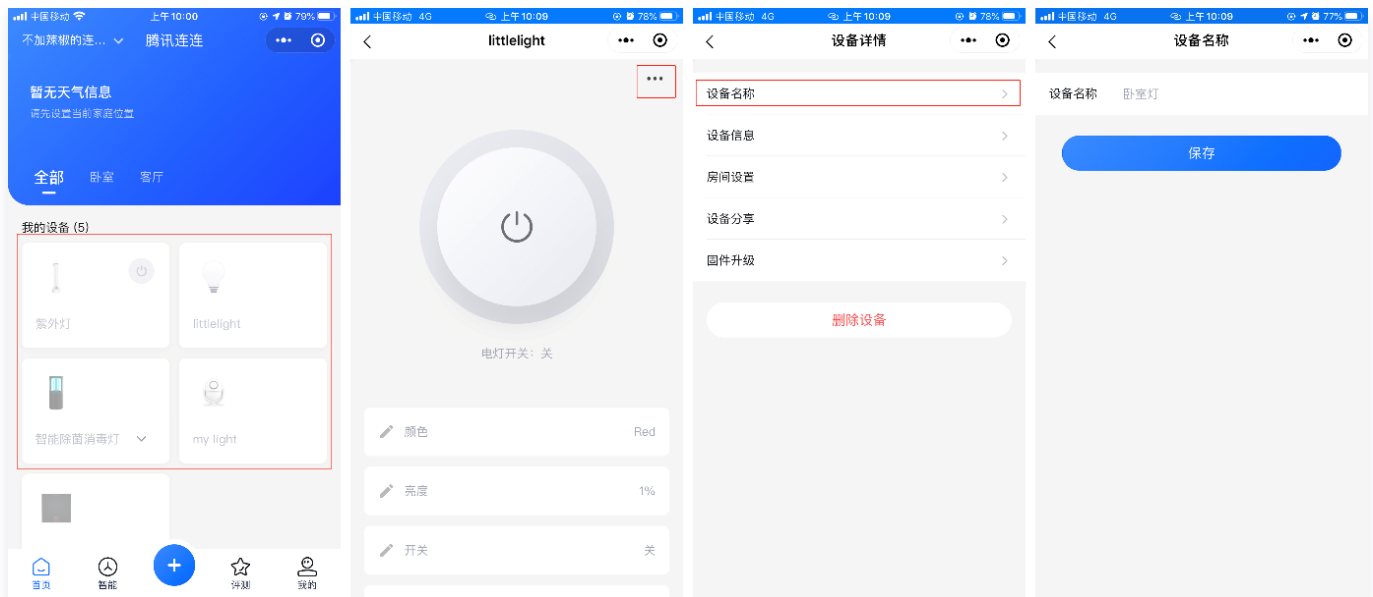
Operation Steps

1. When users use WeChat mini program "Tencent Lianlian" to bind smart device products released by IoT Explorer.

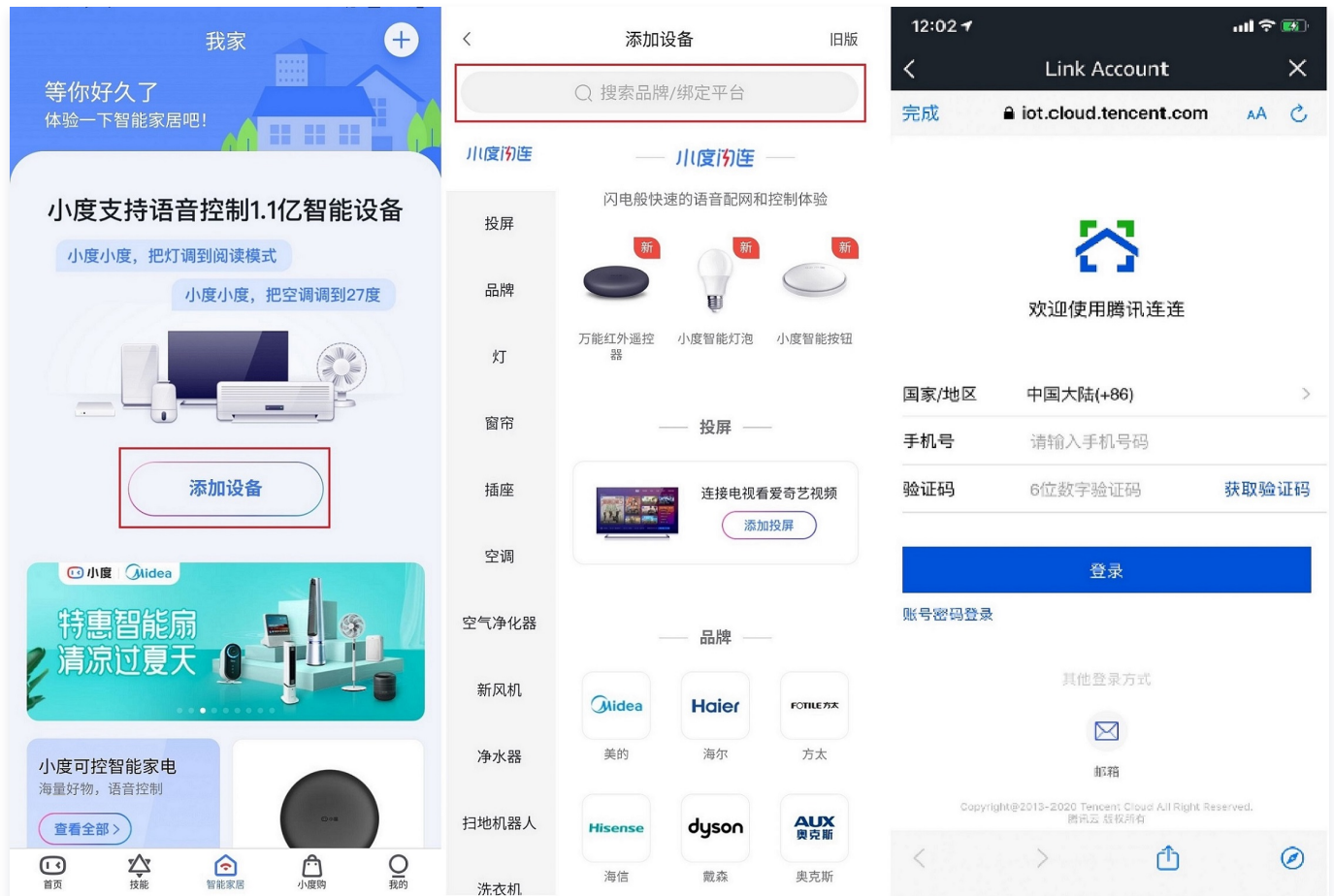
Note:

Users who log in using the WeChat mini program "Tencent Lianlian" **need to go to the personal center to bind their mobile number or email number and set the password.** The path is to select **My > Personal Information**, enter the account and security page, and then you can bind the mobile number or email number.

2. The voice skill product added in Tencent Lianlian Mini Program needs to modify the device name. For example, bedroom light. Avoid using symbols in the modified name.



3. Have one Baidu Xiaodu Smart Speaker, download the Xiaodu Speaker App and bind the speaker.
4. On the homepage of Xiaodu Speaker App, enter **Smart Home**, click **Add Device**, search and select **"Tencent Lianlian"**, and log in to Tencent Lianlian Account, authorize the control authority of the device.



Use Xiaodu Speaker to control the product. Functions supported can be found in the following table.

| Category | Functionality / Feature / Functional / Function (Selected According to Context) | Speech Example Sentences (Based on Real-World Usage Scenarios) |
|--------------|---|---|
| Light | Switch, color adjustment, brightness adjustment, color temperature adjustment. | <ul style="list-style-type: none"> Turn on the light, turn off the light. Turn on the bedroom light, turn off the bedroom light. Set the light to blue. Brighten the light a bit. Adjust the light color temperature to be warmer. |
| Switch panel | Switch | <ul style="list-style-type: none"> Turn on the switch. Turn off the switch. |
| socket | Switch | <ul style="list-style-type: none"> Turn on the socket. Turn off the socket. |
| Curtain | Switch | <ul style="list-style-type: none"> Open the curtain. Close the curtain. |

| | | |
|------------------------|--|---|
| Aromatherapy diffuser | Switch | <ul style="list-style-type: none"> • Turn on the aromatherapy diffuser. • Turn off the aromatherapy diffuser. |
| Floor – cleaning robot | Switch | <ul style="list-style-type: none"> • Turn on the floor – cleaning robot. • Turn off the floor – cleaning robot. |
| Air purifier | Switch, wind speed (increase or decrease), mode. | <ul style="list-style-type: none"> • Turn on the air purifier. • Turn off the air purifier. • Increase the wind speed. • Decrease the wind speed. • Turn on sleep mode. |
| Air conditioning | Switch, mode, wind speed (increase or decrease), temperature (increase or decrease, specify temperature), swing. | <ul style="list-style-type: none"> • Turn on air conditioning, turn off air conditioning. • Turn on cooling mode. • Increase the temperature, decrease the temperature, set the temperature to 26 degrees. • Set the air conditioner to swing left and right. |
| TV | Switch, suspend, resume, channel adjustment, audio volume adjustment. | <ul style="list-style-type: none"> • Turn on the TV, turn off the TV. • Pause TV playback. Continue TV playback. Next channel. Previous channel. I want to watch CCTV-1. • Increase the volume, decrease the volume, mute the TV. |
| fan | Switch, wind speed (increase or decrease), mode. | <ul style="list-style-type: none"> • Turn on the fan, turn off the fan. • Increase the wind speed, decrease the wind speed. • Turn on sleep mode. |

Location Service

Add Locating Attribute Function

Last updated: 2025-04-27 17:40:33

Overview

Currently, the location service of IoT Explorer supports various device positioning attributes, including GPS positioning (GPS_Info, GPS_ExtInfo), cellular positioning (Cell_Info), Wi-Fi positioning (Wifi_Info), and LoRa Edge positioning (Wifi_Info, GNSS_NAV), thereby determining the specific location of the device.

- **GPS positioning:** Your device can report longitude and latitude directly. The device can be located using location service attributes (GPS_Info, GPS_ExtInfo).
- **Cellular positioning:** If the device is a 2G/4G device, it can be located using location service attributes (Cell_Info) by submitting Base Station Information.
- **Wi-Fi positioning:** If the device is a Wi-Fi device, it can be located using the location service feature attribute (Wifi_Info) by submitting the MAC addresses of nearby Wi-Fi routers.

Prerequisites

Product has been created on the console .

GPS Positioning

1. Log in to the [IoT Explorer Console](#).
2. Enter the left-side menu **Product Development** page, select a created product to enter the product **Thing Model Definition** page, and click **Add Standard Features**.



3. In the pop-up for "Add Standard Features", click **General Type > Positioning Function** and check "GPS Positioning".

Note:

When a real device uses the location service for positioning, it needs to report data following the definition of the attribute's Thing Model. For details, see [related Thing Model descriptions](#).



4. 1. Click **Confirm** to complete the addition of the standard feature for geographic location.
5. The device will report the position to the cloud via the Data Template Protocol. For details, see [Thing Model Protocol](#).
 - Device reporting protocols are as follows:
 - Uplink request Topic: `$thing/up/property/{ProductID}/{DeviceName}` .
 - Downlink response Topic: `$thing/down/property/{ProductID}/{DeviceName}` .
 - Request example is as follows:

```
{
  "clientToken": "123",
  "method": "report",
  "params": {
    "GPS_Info": {"longitude": 112.59014, "latitude": 22.28014}
  }
}
```

Cellular Positioning

1. Log in to the [IoT Explorer Console](#), and select a created project to enter the project details page.
2. By default, enter the left-side menu **Product Development** page, select a created product to enter the product **Thing Model Definition** page, and click **Add Standard Features**.



3. In the pop-up for "Add Standard Features", click **General Type > Positioning Function** and check "Cellular Positioning".



4. Click **Confirm** to complete the addition of the Cellular Positioning standard feature.

5. The device will report the position to the cloud via the Data Template Protocol. For more details, see [Thing Model Protocol](#).

○ Device reporting protocols are as follows:

○ Uplink request Topic: `$thing/up/property/{ProductID}/{DeviceName}` .

○ Downlink response Topic: `$thing/down/property/{ProductID}/{DeviceName}` .

○ Request example is as follows:

```
{
  "clientToken": "123",
  "method": "report",
}
```

```
"params": {  
  "Cell_Info":  
  {"rss":-85,"networkType":1,"mcc":460,"mnc":13824,"lac":3,"cid":33}  
}
```

Wi-Fi Positioning

1. Log in to the [IoT Explorer Console](#), and select a created project to enter the project details page.
2. By default, enter the left-side menu **Product Development** page, select a created product to enter the product **Thing Model Definition** page, and click **Add Standard Features**.



3. In the pop-up for "Add Standard Features", click **General Type > Positioning Function** and check "wifi Positioning".



4. Click **Confirm** to complete the addition of the Wi-Fi Positioning standard feature.

5. The device can report the position to the cloud via the Data Template Protocol. For more details, see [Thing Model Protocol](#).

- Device reporting protocols are as follows:

- Uplink request Topic: `$thing/up/property/{ProductID}/{DeviceName}` .

- Downlink response Topic: `$thing/down/property/{ProductID}/{DeviceName}` .

- An example of a request is as follows:

```
{
  "clientToken": "123",
  "method": "report",
  "params": {
    "Wifi_Info": [{"Mac": "78a106a5b166", "Rssi": -81},
{"Mac": "608f5c66b3f7", "Rssi": -82}, {"Mac": "282cb293a5d6", "Rssi": -83}]
  }
}
```

Space Management

Last updated: 2025-04-27 17:40:48

This document introduces related operations for creating, editing, and deleting spaces in the console.

Prerequisites

Completed [device positioning](#).

Create New Space

1. Log in to the [IoT Explorer Console](#).
2. Click on the left-side menu **Location Service** to enter the Location Service interface, click **Create New Space**, and fill in the corresponding information.

新建空间

空间名称 *

共享单车设备管理

支持中文、英文、数字、下划线的组合，最多不超过20个字符

关联产品 *

搜索产品

☒ 蜂窝产品

☒ 地理位置1

蜂窝产品

地理位置1

授权形式

☒ 只读（展示）

备注 *

用于管理深圳市投放的共享单车

保存

取消

- Space name: Name the space. The space name cannot be repeated with other spaces. A combination of Chinese characters, English letters, digits, and underscores is supported, with a maximum of 20 characters.

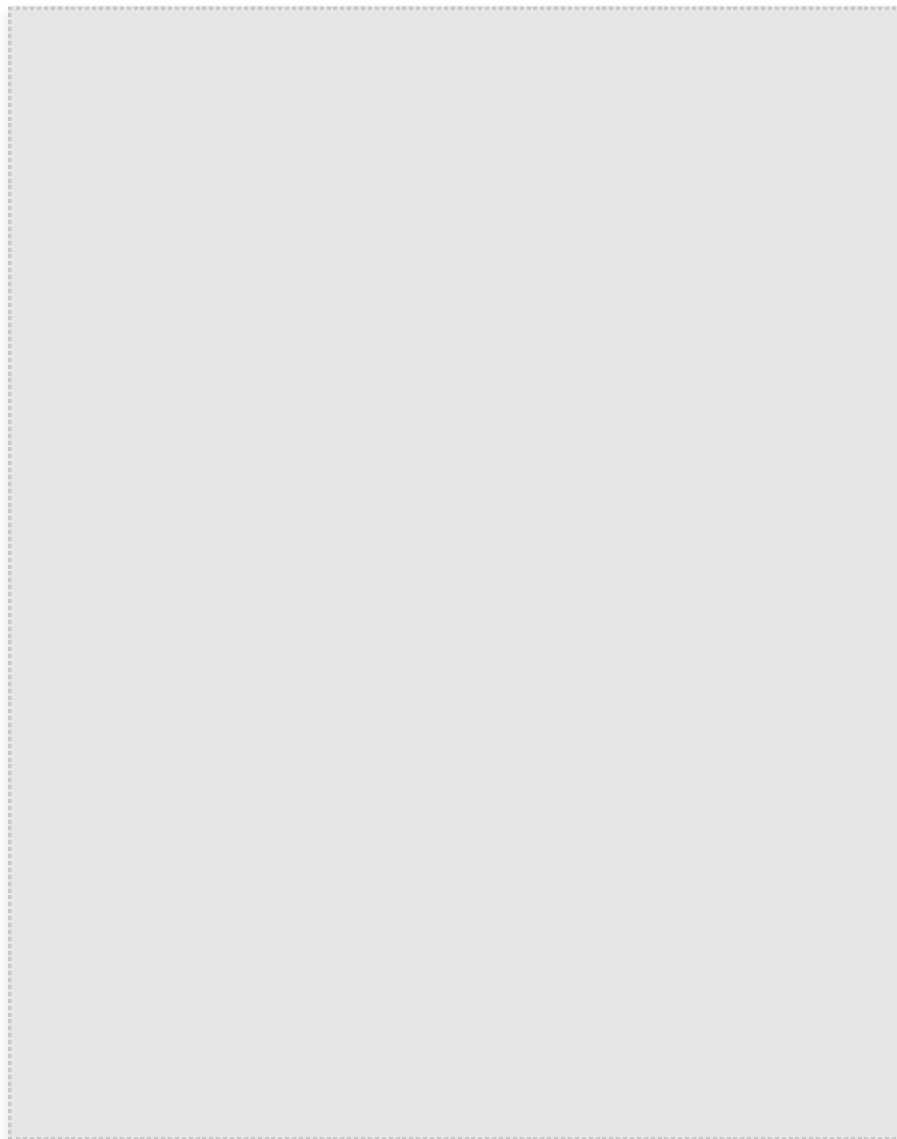
- **Associated products:** Associate the selected products with this space project. Only after the association can the space project obtain relevant status information of the devices under the product.
 - **Authorization form:** Select the permissions of the space project for the associated product equipment. Currently, the space project only has the right to view and display the devices.
 - **Remarks:** You can type text to describe the space project. Limit on number of characters: 25.
3. Click **Save**, and the newly created space will be updated to the location service list page.
 4. Click on the "space name" of a certain location space in the list to enter the visual operation interface of this space project.

Edit Space

1. On the location service list page, click **Edit** in the right-side menu of a certain space to enter the space details page.

| 序号 | 名称 | 备注 | 创建时间 | 操作 |
|----|-------------------------|------------------|---------------------|---------------------------------------|
| 1 | 蜂窝设备空间 | 关于蜂窝类设备的围栏空间管理项目 | 2020-11-20 16:13:46 | 编辑 删除 |
| 2 | 位置空间444 | for some project | 2020-11-19 22:11:14 | 编辑 删除 |
| 3 | 位置空间333 | 位置空间管理深圳南山区的相关设备 | 2020-11-19 18:55:13 | 编辑 删除 |

2. You can modify the relevant information of the space.



3. Click **save** to update the edited and saved spatial information.

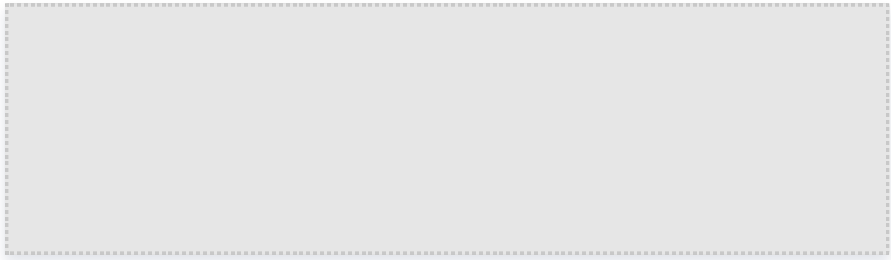
Delete Space

1. On the location service list page, click **Delete** in the right-side menu of a certain space.

| 序号 | 名称 | 备注 | 创建时间 | 操作 |
|----|--------|------------------|---------------------|---------------------------------------|
| 1 | 蜂巢设备空间 | 关于蜂巢类设备的围栏空间管理项目 | 2020-11-20 16:13:46 | 编辑 删除 |
| 2 | 位置空间 | 位置空间管理 | 2020-11-19 18:55:13 | 编辑 删除 |

2. Click **OK** in the pop-up.

- If there is no **geofencing** under the space project, a notification "Deletion successful" will be displayed, and it will be removed from the location list.
- If there are still **geofencings** under the space project, it cannot be deleted and a prompt "Unable to delete this space project" will appear. All geofencings under the space project need to be manually deleted before the deletion operation can be performed.



Spatial Visualization

Last updated: 2025-04-27 17:41:05

Tencent Cloud IoT develops location space visualization services, allowing you to display device status information in real time on the map. It supports viewing attribute data of devices, the specific location of devices, heat maps (points), heat maps (areas) and other features, as well as map zoom and full-screen viewing, making it convenient for you to view and manage devices.

Prerequisites

Completed [Space Creation](#).

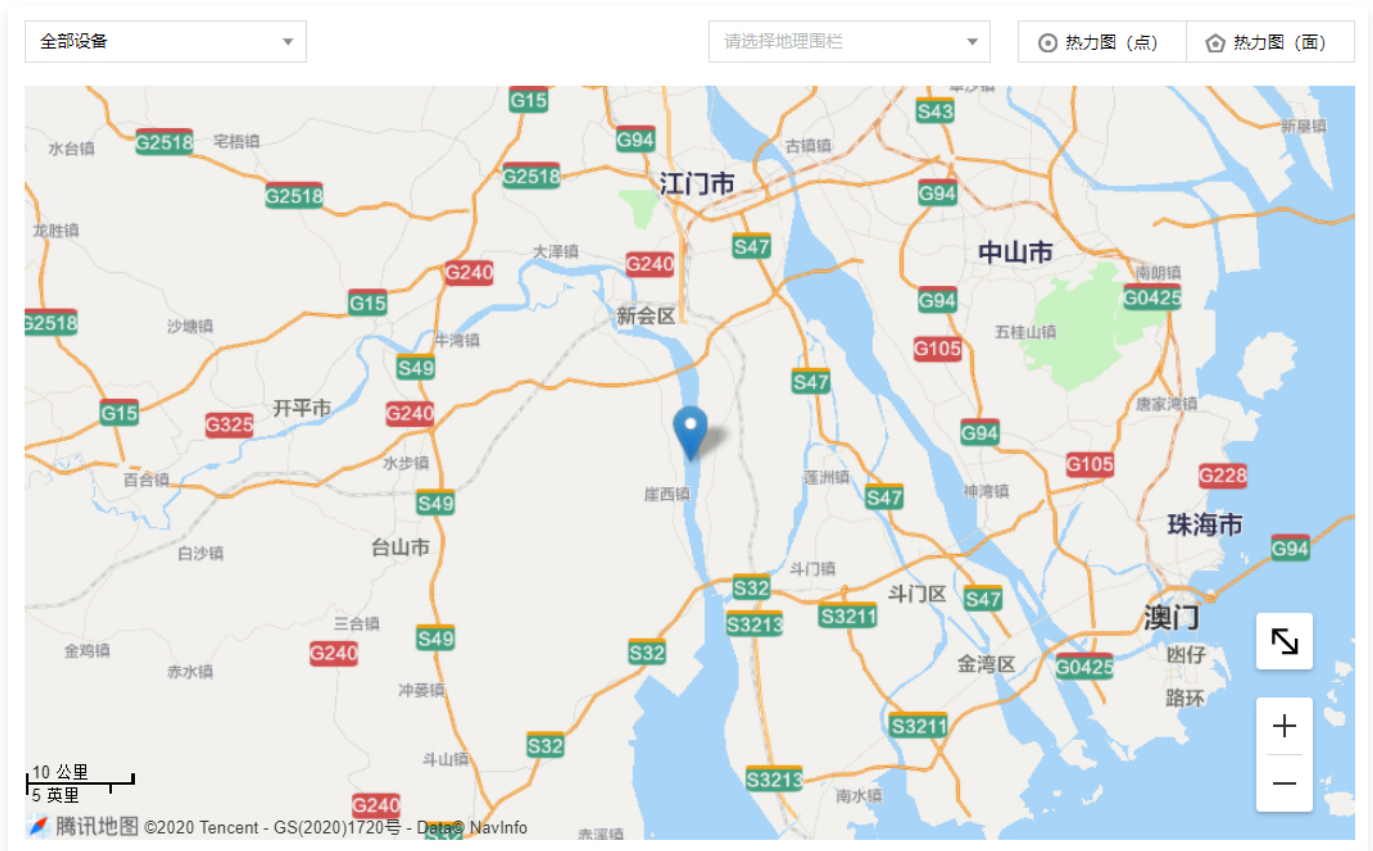
Device Location Visualization

1. Log in to the [IoT Explorer Console](#).
2. Click on the left menu **Location Service** to enter the location service page.
3. On the location service page, click on a certain already created "space name".

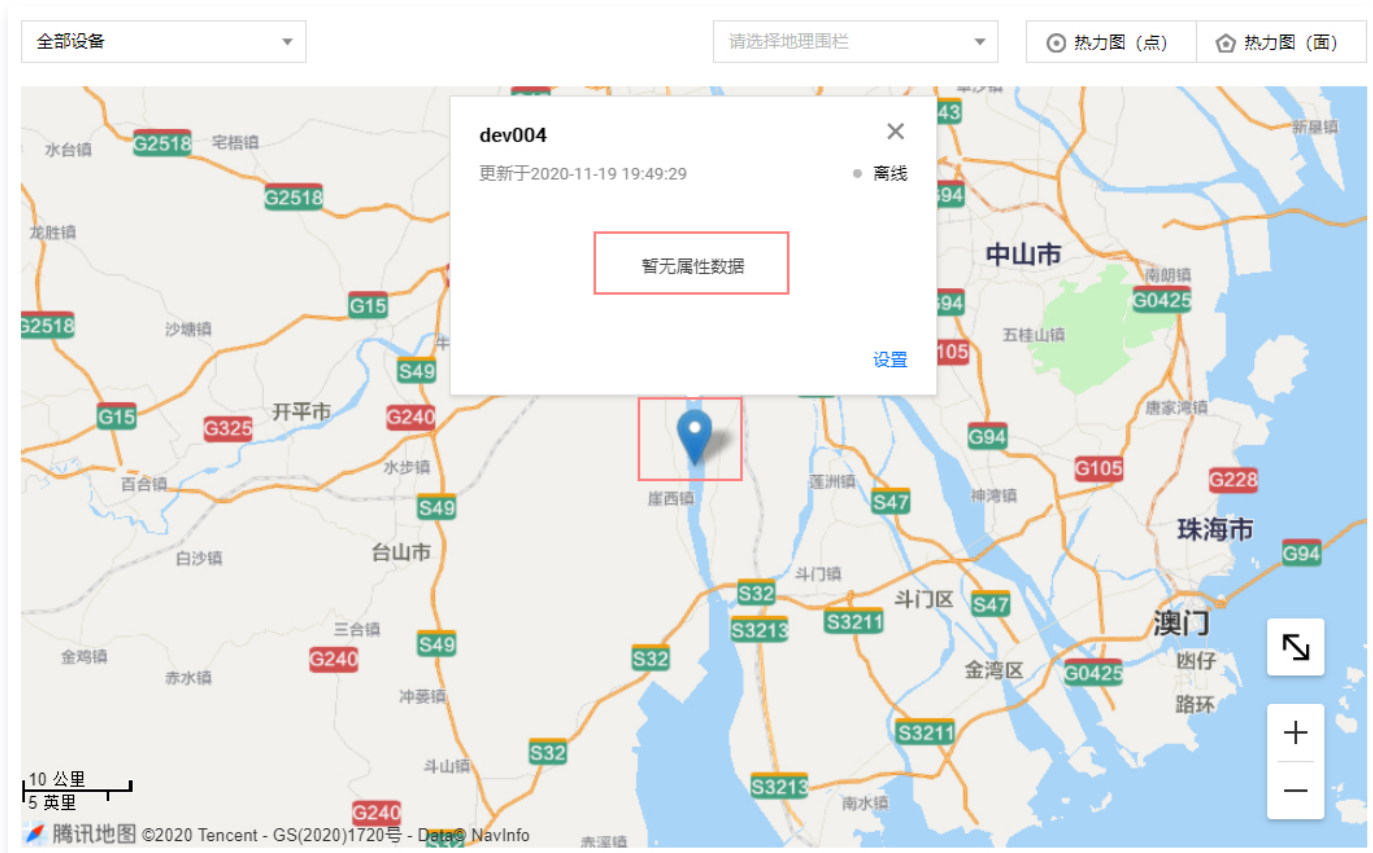


| 新建空间 | | | | | 请输入空间名称 |
|---|------------------------|------------------|---------------------|---------------------------------------|---------|
| ① 仅支持数据模板（物模型）定义了定位属性的产品使用此功能。点击 查看文档 | | | | | Q |
| 序号 | 名称 | 备注 | 创建时间 | 操作 | |
| 1 | 蜂窝设备空间 | 关于蜂窝类设备的围栏空间管理项目 | 2020-11-20 16:13:46 | 编辑 删除 | |
| 2 | 位置空间 | 位置空间管理深圳南山区的相关设备 | 2020-11-19 18:55:13 | 编辑 删除 | |

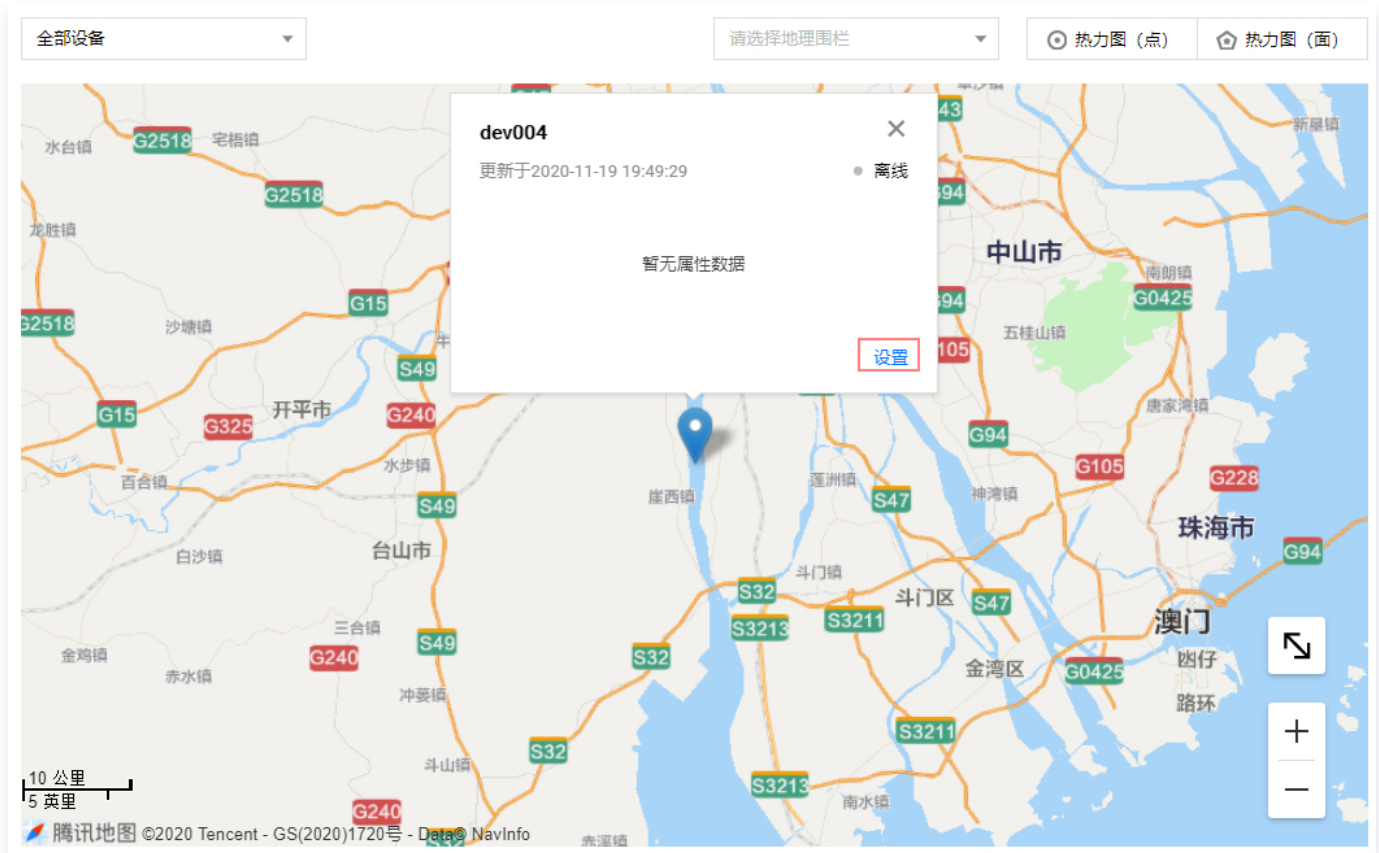
4. Enter the spatial visualization interface. Devices reporting location information will be displayed on the map in the form of icons.



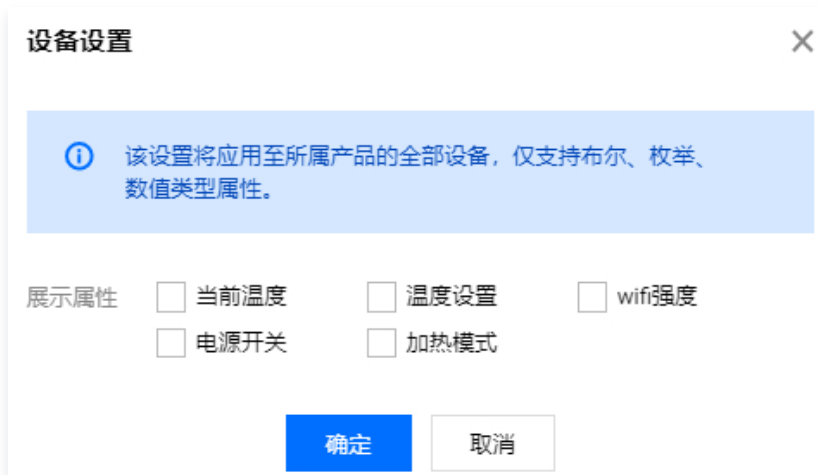
5. When entering this location space for the first time, click the "device icon" on the map. "No attribute data" will be displayed in the device pop-up.



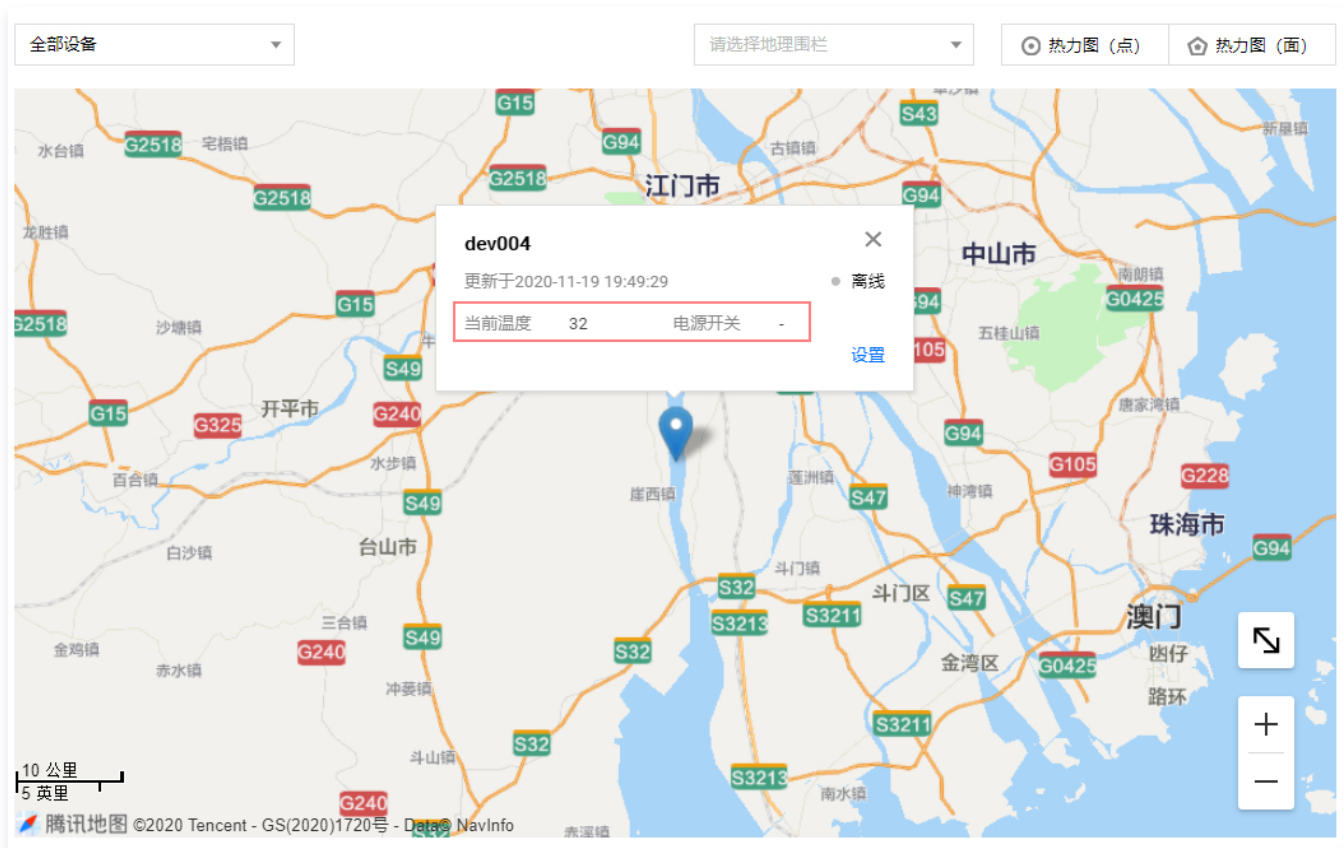
6. At this point, click **settings** in the bottom-right corner of the device pop-up.



7. Select the attribute data you need to display in the device popup in the "device settings" popup.

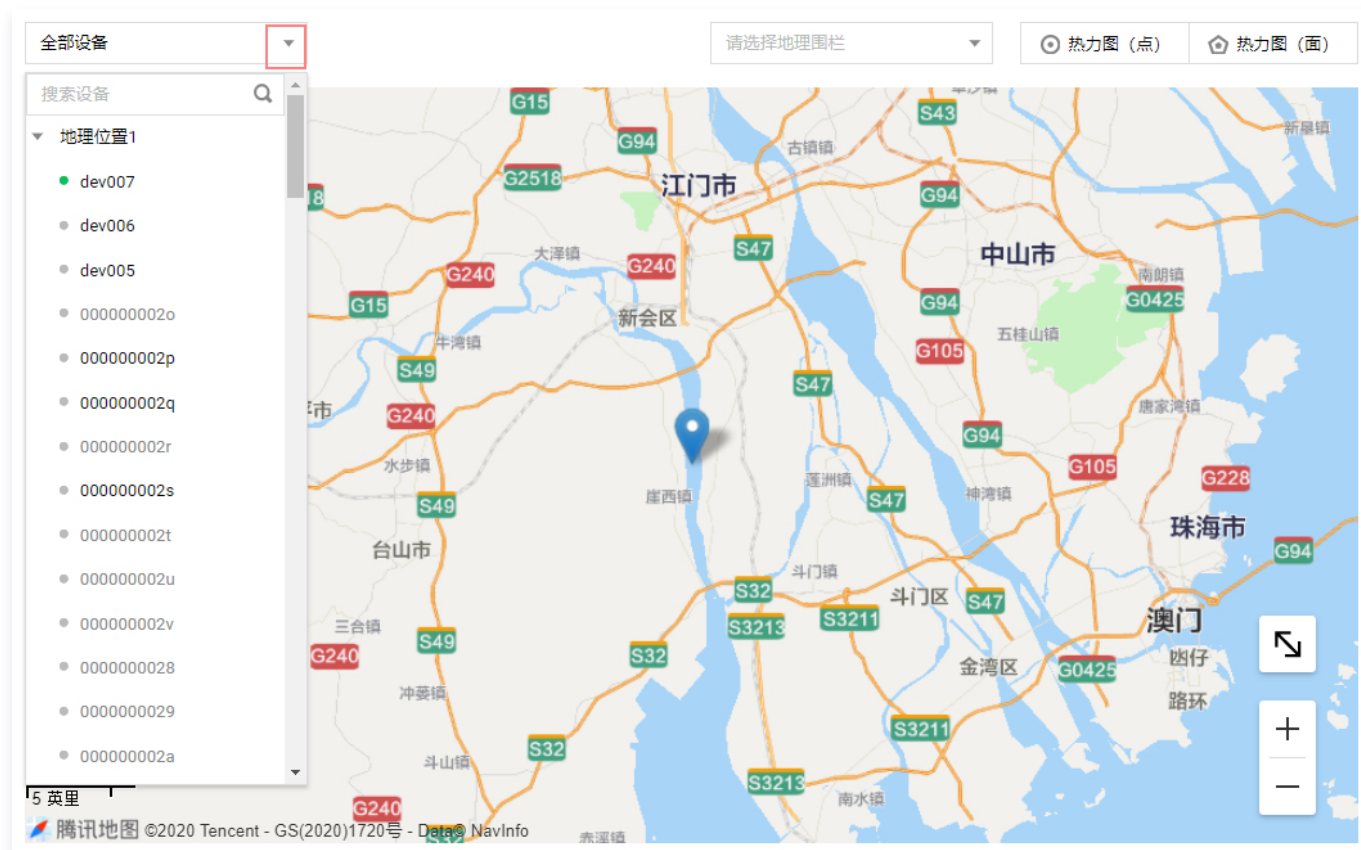


8. Click **save** after the settings are completed. Enter again to view the data display of the device icon, and you can view the attribute data that has just been set to be displayed.

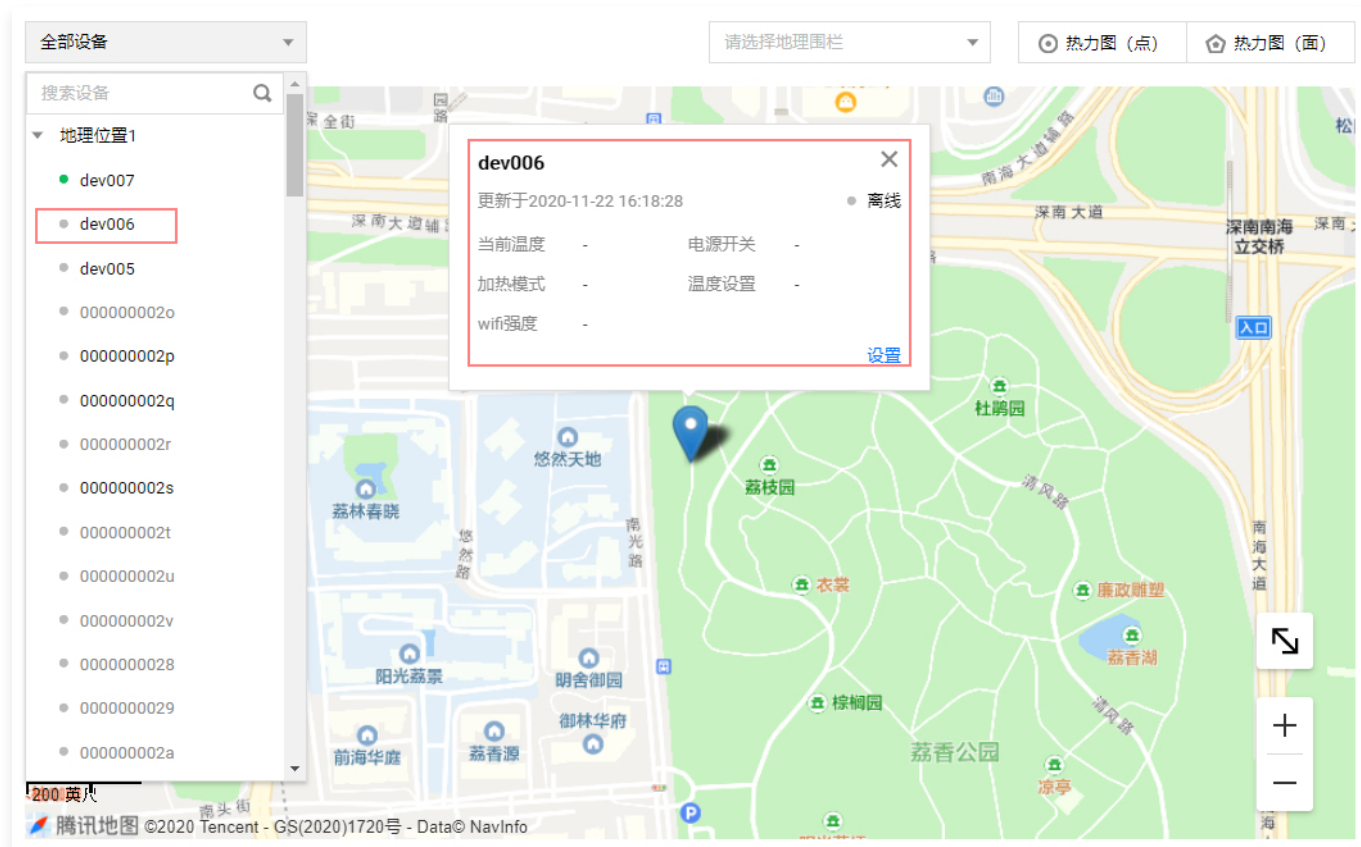


View Specified Device Location

1. Enter the page of [step 4](#) of device visualization to proceed with the operation.
2. Unfold the drop-down list in the upper left corner of the map to view all devices under the space project-associated product.



3. Click to select a specified device, navigate to its location, and show the device property status data.



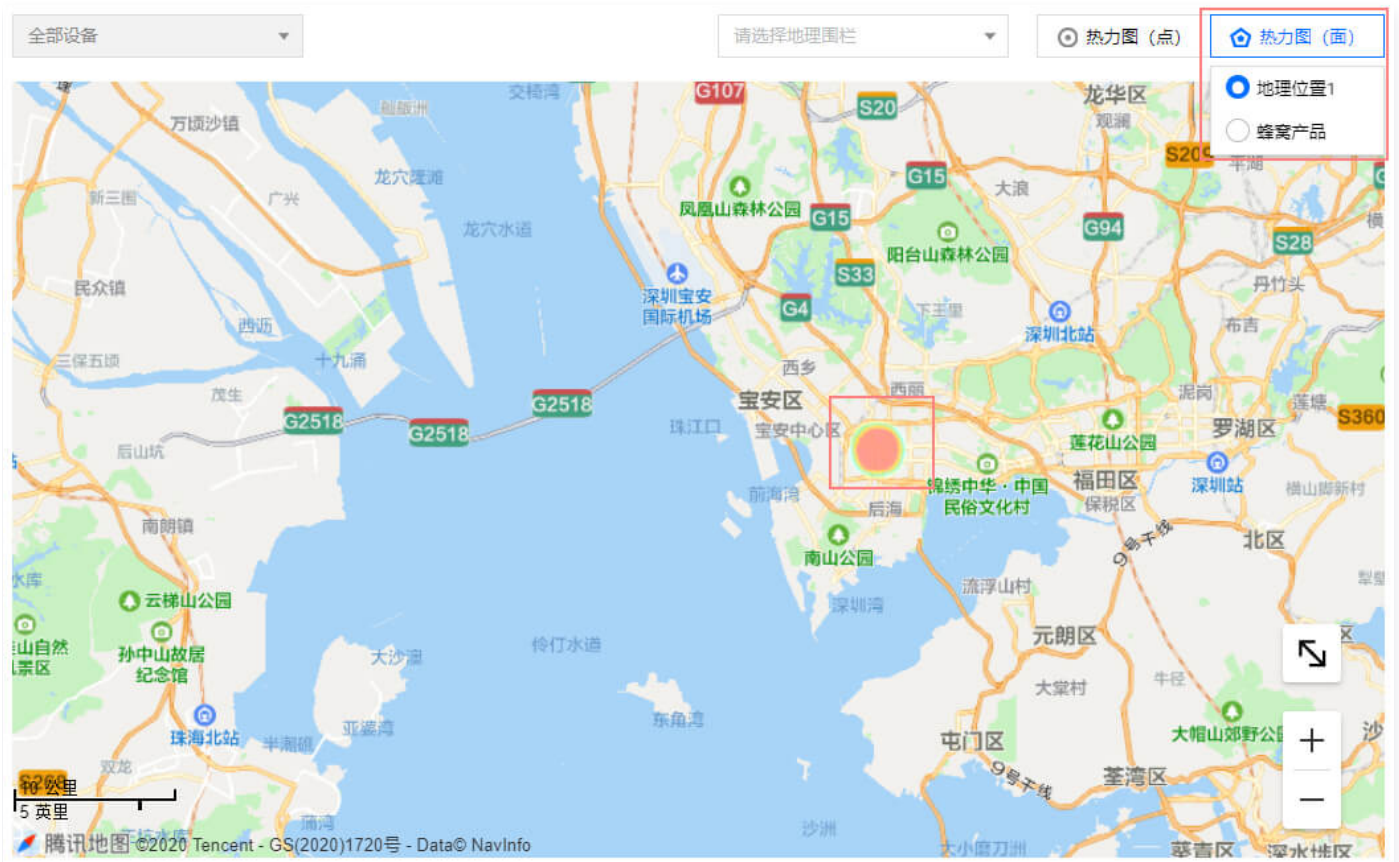
Heat Map (Point)

Click the "Heat Map (Points)" feature on the map, select a specific product to be displayed, and enter the aggregated view status of the heat map (points).



Heat Map (Area)

Click the "Heat Map (Areas)" feature on the map, select a specific product to be displayed, and enter the aggregated view status of the heat map (areas).



Geofencing

Last updated: 2025-04-27 17:41:23

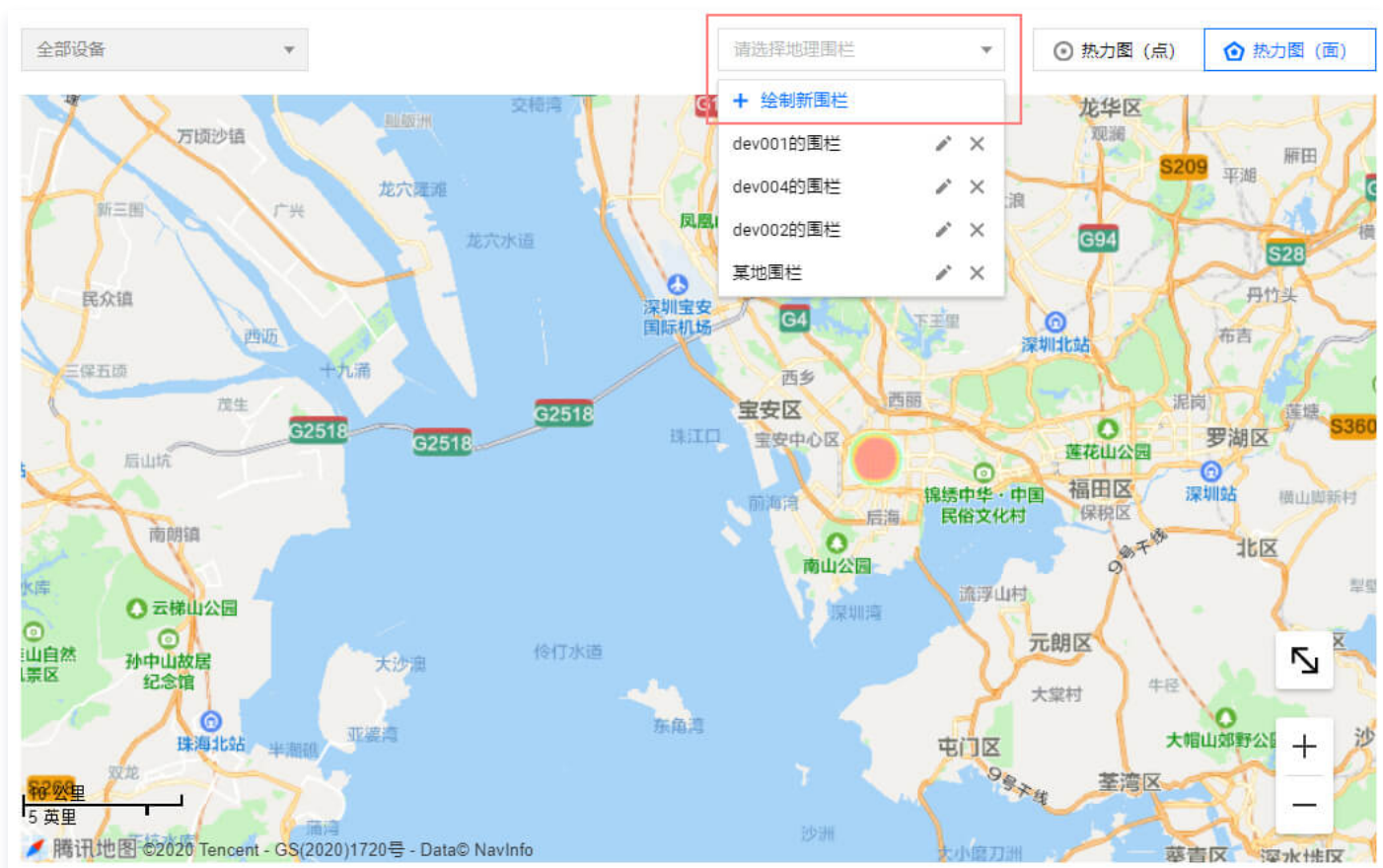
Tencent Cloud IoT Explorer's location service feature provides users with geofencing capabilities. It supports multiple fence types such as circular, polygon, and administrative district. It can be widely used in scenarios where locators, smart watches, smart farms, etc. need geographic range – limited alarms. This document primarily introduces the method of use of the geofencing feature.

Prerequisites

The creation of [location space](#) has been completed, and the associated product equipment must be able to report relevant position attribute information. For specific location reporting, please refer to [feature introduction](#) and [Thing Model description](#).

Create a Fence

1. Log in to the [IoT Explorer Console](#).
2. Click on the left menu **Location Service** to enter the location service page.
3. In the location service Space List, create or enter a certain space and enter the visual interface of the space.
4. click **Geofencing** > **Draw a new fence**.



5. Enter the geofence information entry interface and fill in relevant information.

新建围栏

围栏名称 * 请输入围栏名称
支持中文、英文、数字、下划线的组合，最多不超过10个字符

围栏类型 ☒ 圆形 ☐ 多边形 ☐ 行政区

关联设备 * 搜索设备

▼ 地理位置1

- ☒ dev007
- ☒ dev006
- ☒ dev005
- ☐ 000000002o
- ☐ 000000002p
- ☐ 000000002q
- ☐ 000000002r
- ☐ 000000002s
- ☐ 000000002t
- ☐ 000000002u
- ☐ 000000002v

dev007 x

dev006 x

dev005 x

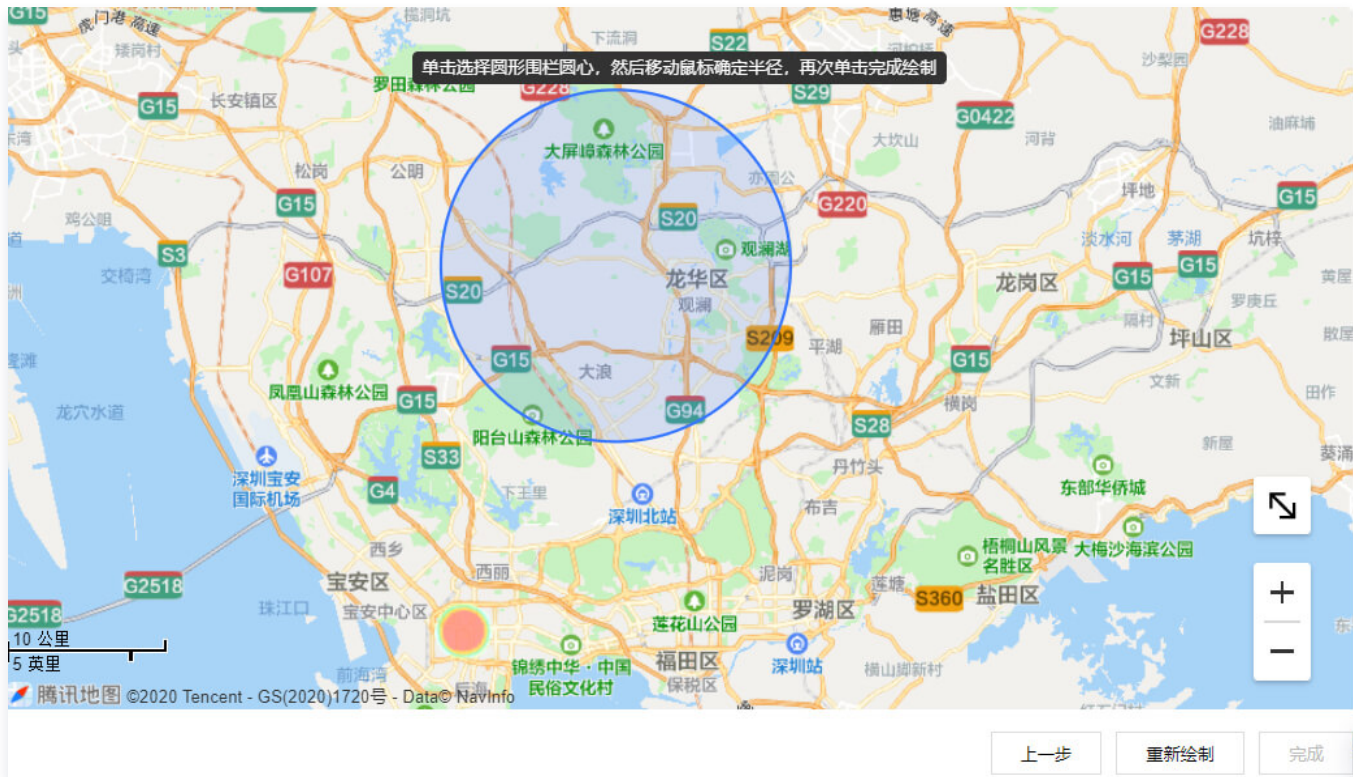
触发条件 * 进入围栏时 ▼

下一步 取消

- Fence name: Set the fence name. It supports a combination of Chinese, English, digits, and underscores, with a maximum of 10 characters.
- Fence type: Support three types of fences, circular, polygon, and administrative district fences, which can be selected based on actual business needs.
- Associated equipment: Select the equipment that needs to be associated with the created fence. Only the equipment associated with the fence can trigger the alarm of this fence.
- Trigger conditions: Set the conditions for triggering the fence alarm, when entering or exiting the fence.

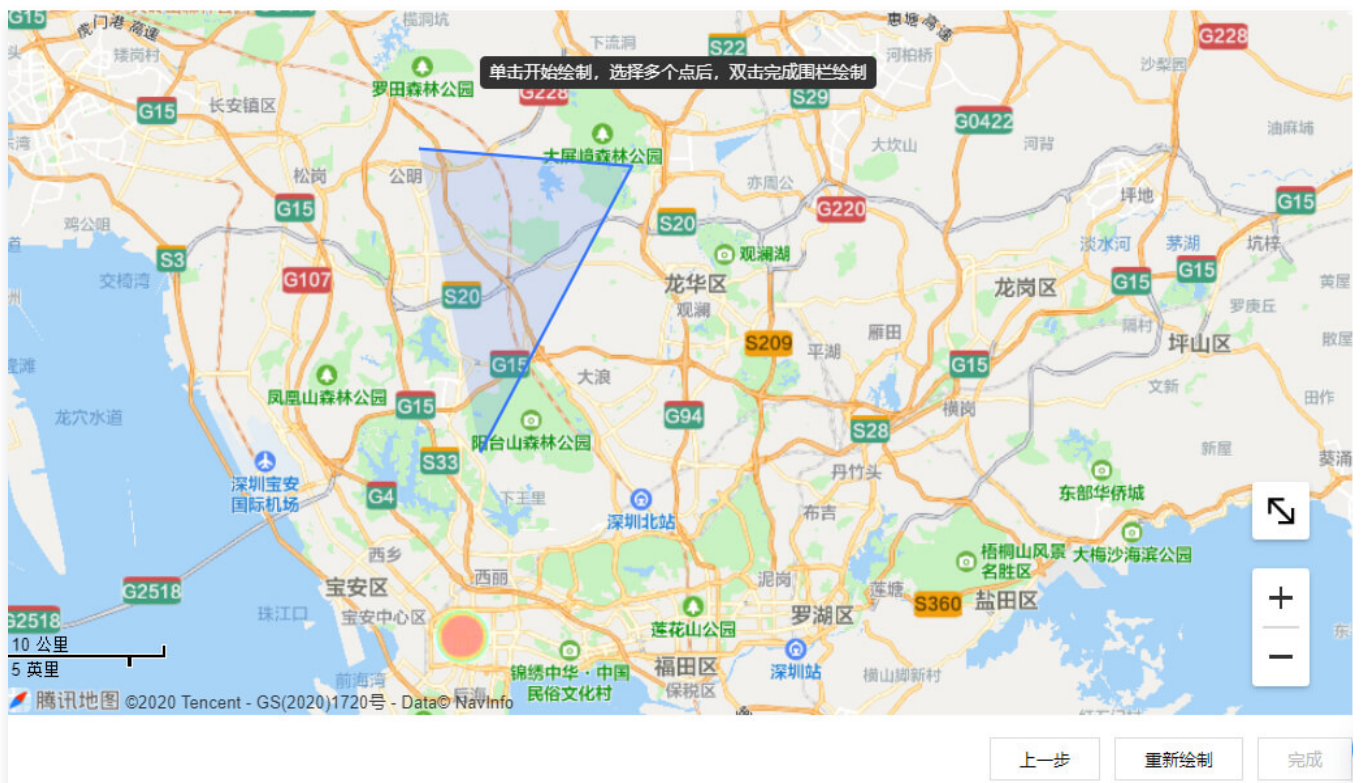
6. click **Next** to start drawing a fence. Different fence types have different drawing methods. Once the fence drawing is completed, if modifications are needed for the drawn fence, click **Redraw** to redraw.

- **Circular Fence**
Click to select the circle center of the circular fence, then move the mouse to determine the radius, and click again to complete the drawing.



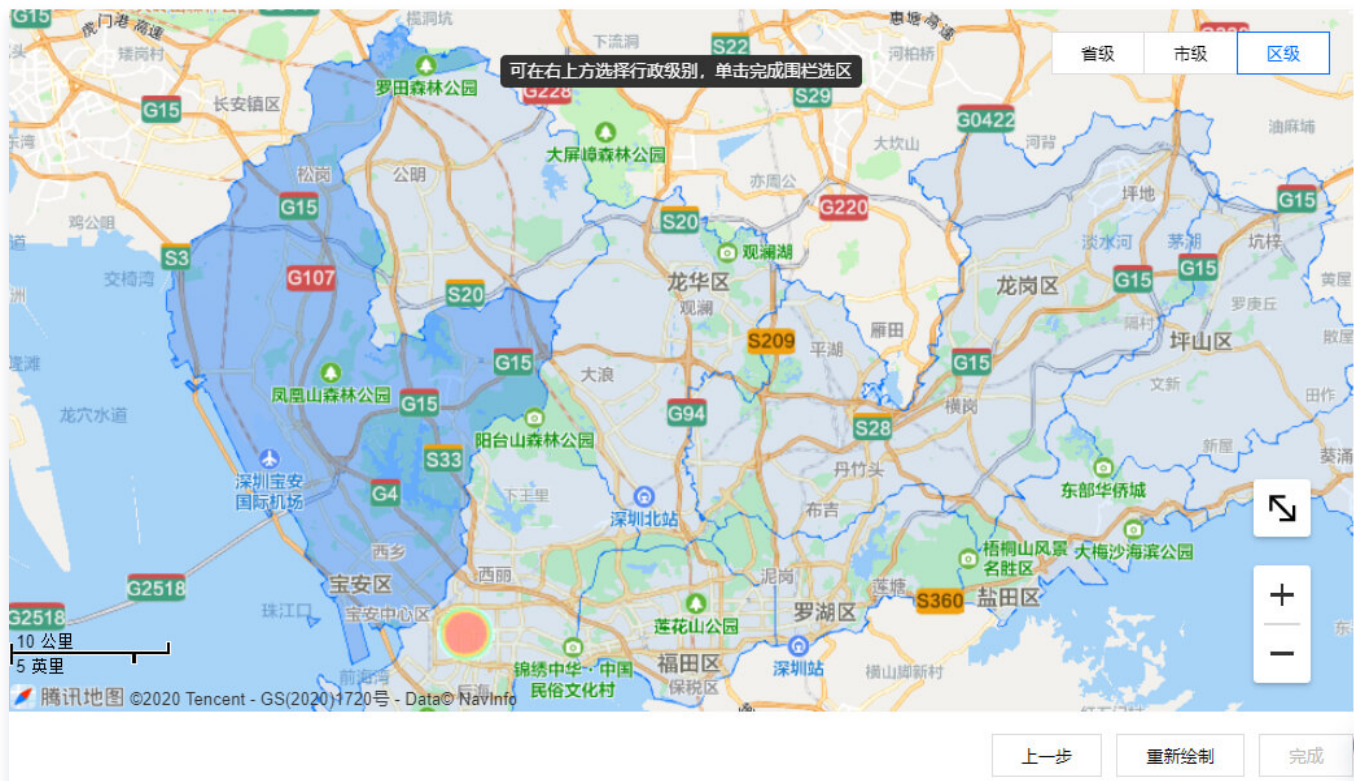
○ Polygon Fence

Click to start drawing, select multiple points, and double-click to complete the fence drawing.



○ Administrative District Fence

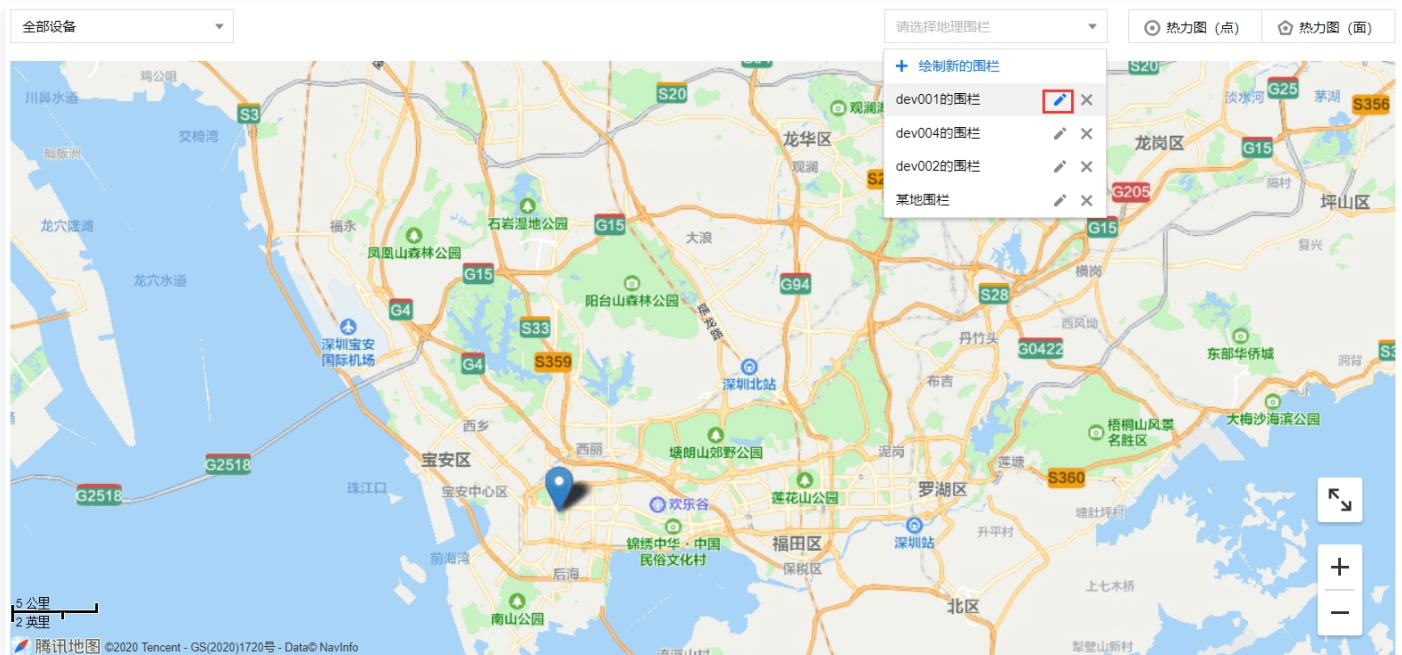
You can select the administrative level (province-city-district) in the upper right corner, and click to complete the fence selection.



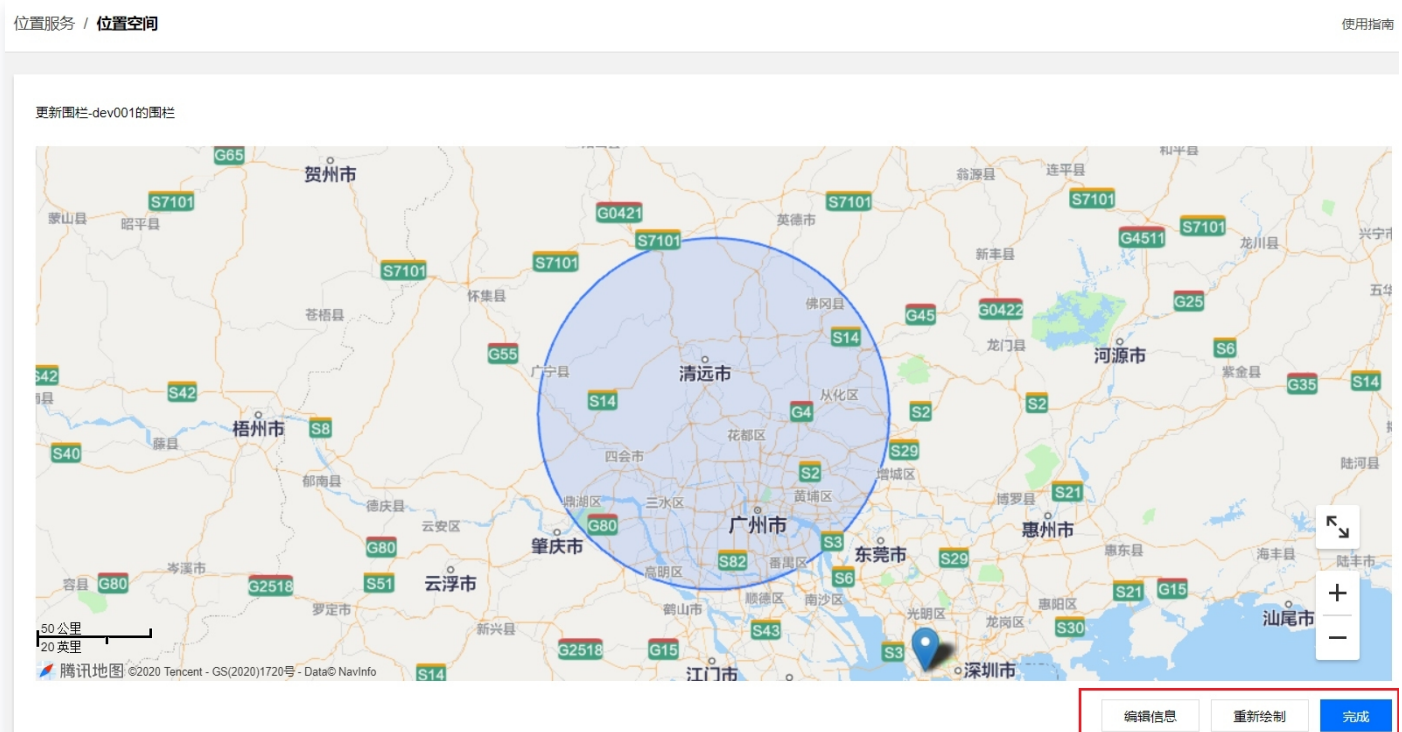
7. After drawing the fence, click **Done**. A notification “Fence added successfully” will prompt, and the fence creation will be completed.

Refreshing a Fence

1. After completing [geofence creation](#), in the dropdown list of the fence, click the "edit" icon of a certain geofence.



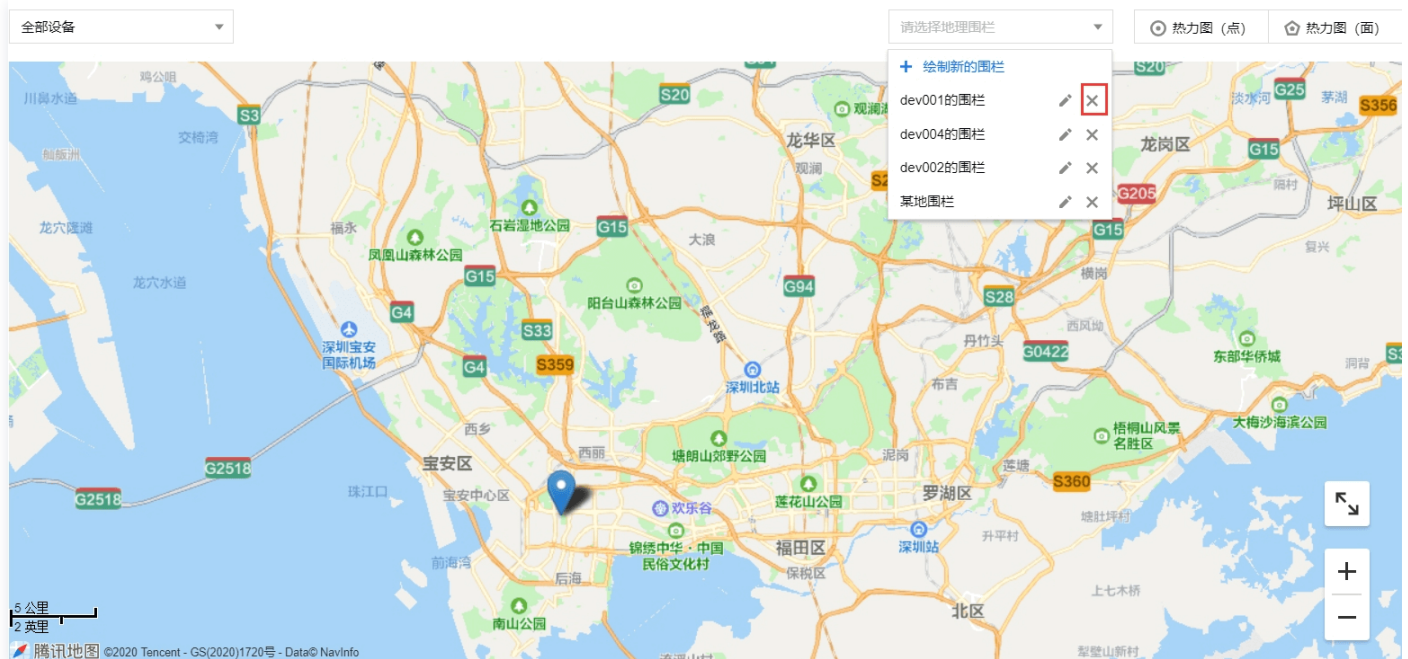
2. Enter the fence update interface. You can click **Edit Information** to modify the fence configuration-related information; after clicking **Redraw**, the existing fence will be deleted and a new fence will be drawn.



3. After the update operation is completed, click **Done** to complete the update of the fence-related information.

Delete a Fence

1. After completing [the fence creation](#), in the fence dropdown, click the "Delete" icon of a certain geofence.



2. A "Confirm Deletion of Fence" prompt dialog box pops up. Click **Confirm** to delete this geofence.



Fence Alarm Event Query

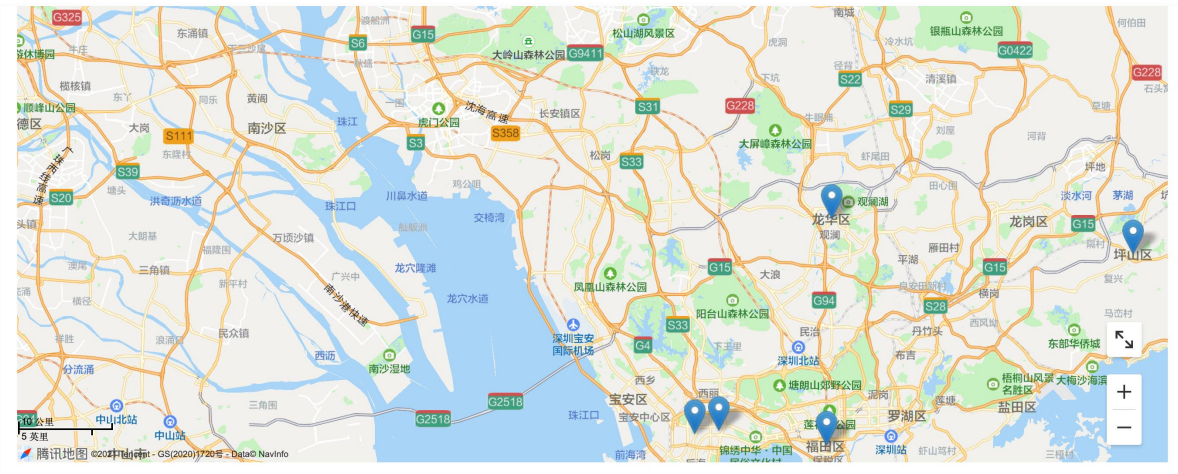
1. When a device associated with the created geofence enters or exits the fence, the platform will push a fence alarm. You can query the device's fence alarm events in the fence alarm list at the bottom of the interface.

开发中心

- 产品开发生成
- 应用开发生成
- 数据开发生成
- AI开发生成

服务中心

- 固件升级
- 设备量产
- 网络管理
- 数据同步
- 运营分析
- 语音技能
- 物联使能
- 位置服务

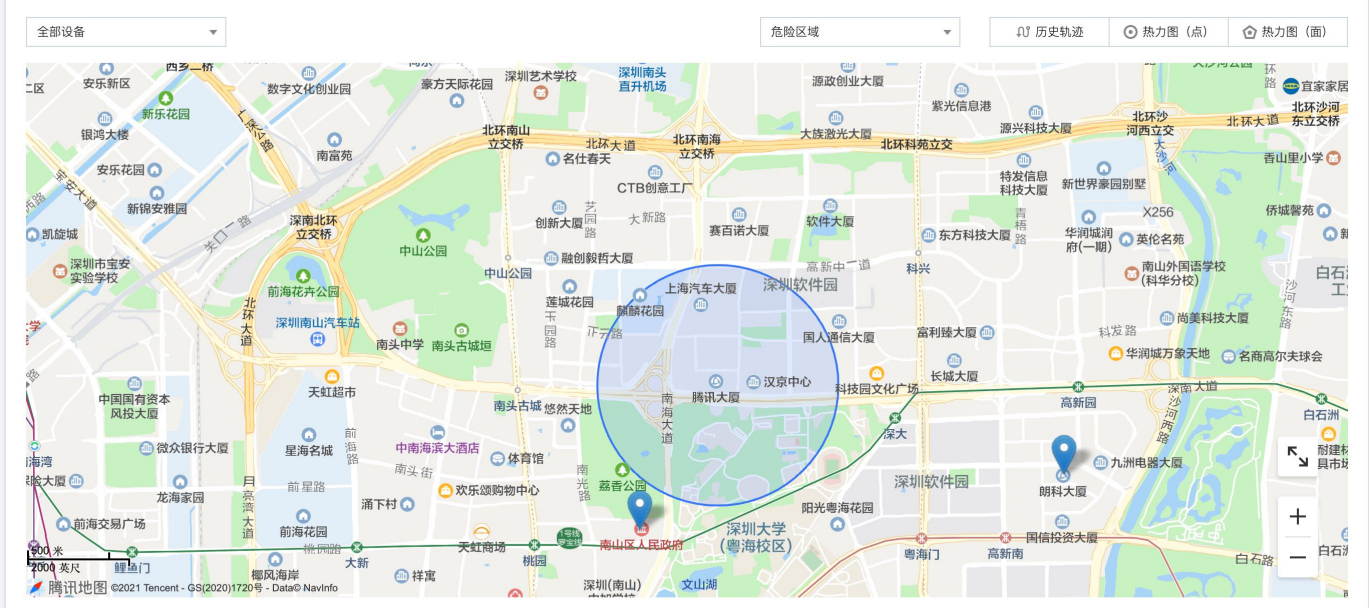


围栏告警 自动刷新 ☐ 选择地理围栏

| 所属产品 | 时间 | 设备名称 | 告警类型 | 所属围栏 |
|------|---------------------|----------|------|------|
| 定位器 | 2021-01-22 18:30:00 | fence001 | 进入 | 危险区域 |
| 定位器 | 2021-01-22 17:07:17 | fence002 | 进入 | 危险区域 |
| 定位器 | 2021-01-22 16:44:50 | fence001 | 进入 | 危险区域 |

2. In the fence alarm list, new fence alarm events can be received through manual refresh or automatic refresh. Click the "Manual Refresh" icon to refresh the fence alarm list once and load the latest alarm events. If **automatic refresh** is enabled, the list will refresh the alarm events every 5 seconds.

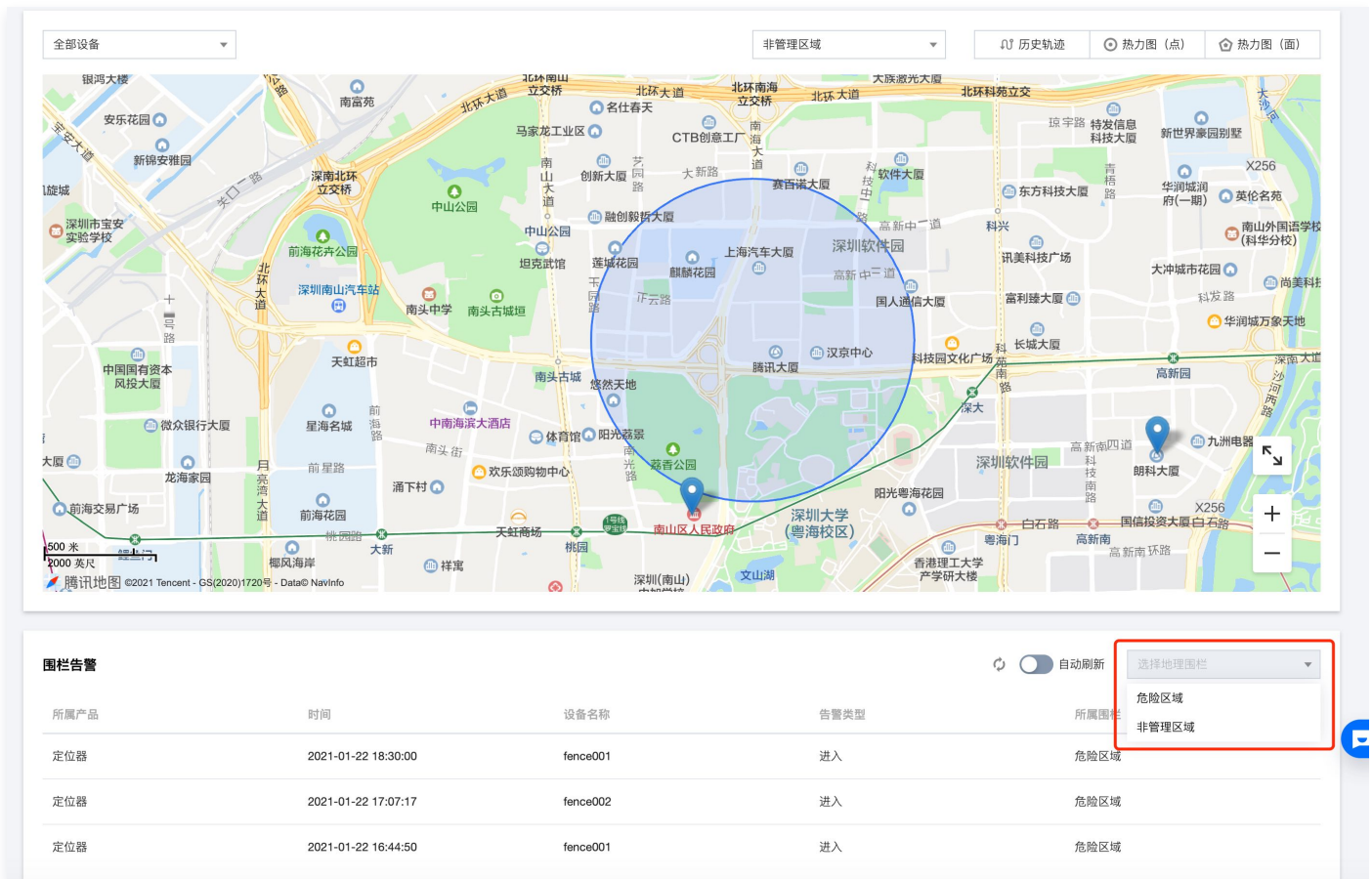
全部设备



围栏告警 自动刷新 ☒ 选择地理围栏

| 所属产品 | 时间 | 设备名称 | 告警类型 | 所属围栏 |
|------|---------------------|----------|------|------|
| 定位器 | 2021-01-22 18:30:00 | fence001 | 进入 | 危险区域 |
| 定位器 | 2021-01-22 17:07:17 | fence002 | 进入 | 危险区域 |
| 定位器 | 2021-01-22 16:44:50 | fence001 | 进入 | 危险区域 |

3. Meanwhile, to make user management of event alarms under each fence convenient, you can also select a specific fence on the right to view the alarm events under a certain specific fence.



The screenshot displays the IoT Explorer interface. The top section features a map of Shenzhen with a blue circular fence area. The bottom section contains a table titled "围栏告警" (Fence Alarm) and a dropdown menu for selecting a geographical fence.

围栏告警

| 所属产品 | 时间 | 设备名称 | 告警类型 | 所属围栏 |
|------|---------------------|----------|------|------|
| 定位器 | 2021-01-22 18:30:00 | fence001 | 进入 | 危险区域 |
| 定位器 | 2021-01-22 17:07:17 | fence002 | 进入 | 危险区域 |
| 定位器 | 2021-01-22 16:44:50 | fence001 | 进入 | 危险区域 |

选择地理围栏

- 危险区域
- 非管理区域

Historical Trajectory

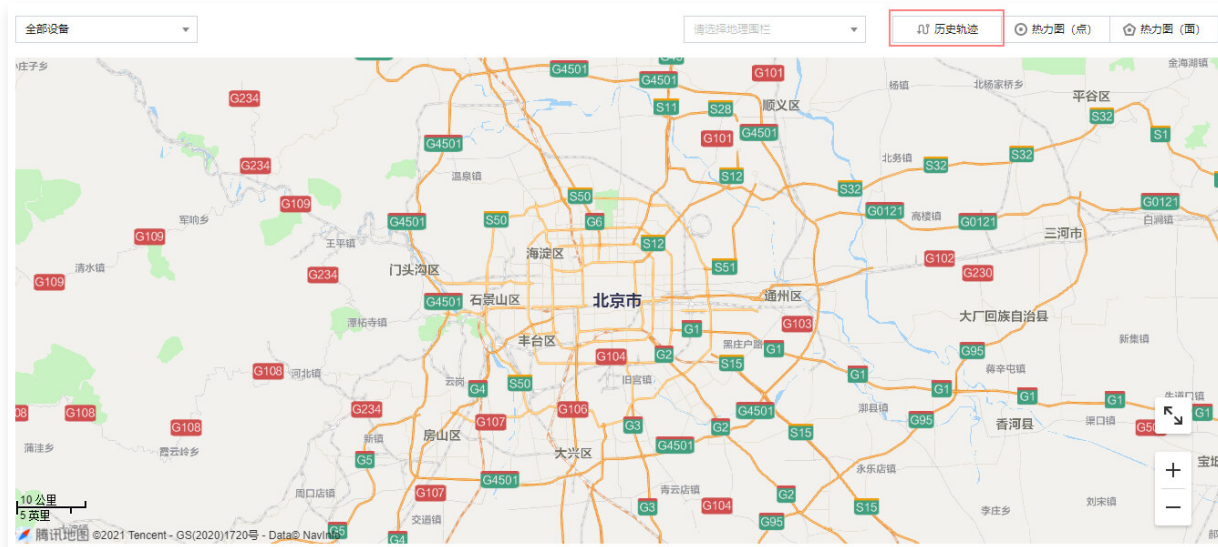
Last updated: 2025-04-27 17:41:40

Application Scenarios

Tencent Cloud IoT Explorer's location service feature supports saving the location of historical trajectory points of user devices and historical trajectory visualization query.

View Historical Trajectory

1. Log in to the [IoT Explorer console](#) and enter the target instance.
2. Select **VAS > Location Service** in the left menu to enter the location service page.
3. On the location service page, select the target space name and enter. Then click the upper right **historical trajectory** to enter the historical trajectory selection interface.



4. In the historical trajectory selection interface, select the historical trajectory to be queried.

历史轨迹

轨迹时间段2023-06-14 00:00:00 ~ 2023-06-15 23:59:59

选择设备0000000001

定位精度60

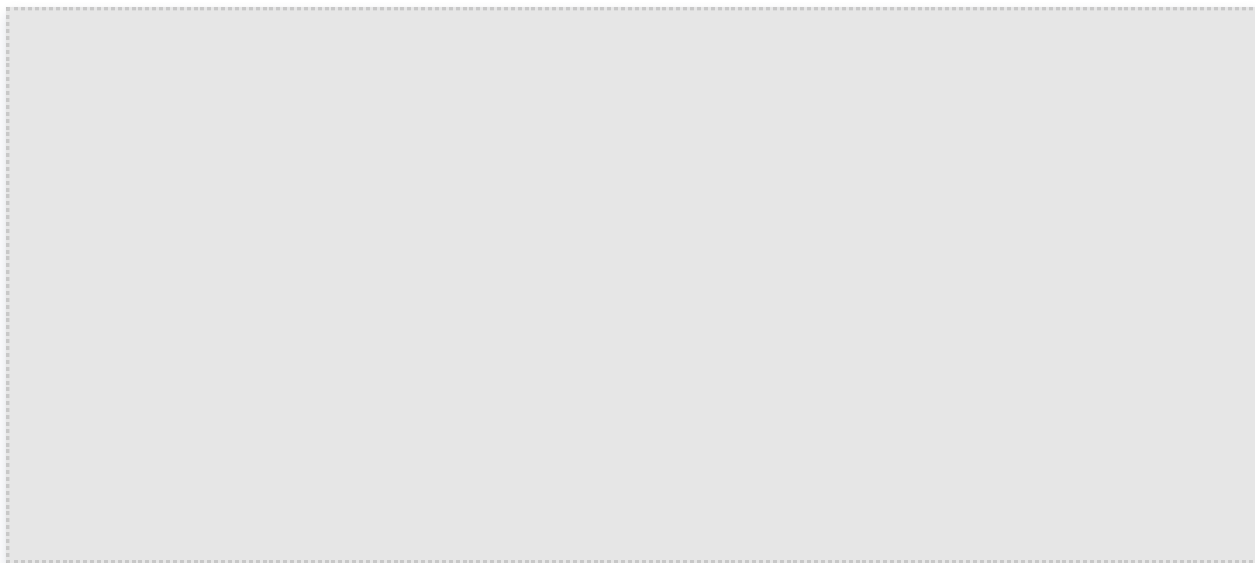
确定重置

- **Trace time period:** you can choose to view a specific time period within one month.
- **Select device:** Currently, the console can simultaneously support trajectory queries for up to three devices.


- **Positioning accuracy:** Input the positioning accuracy of the historical trajectory. The default is 60 meters. It is modifiable by the user. If the estimated deviation of the positioning accuracy exceeds the threshold, it will not be shown in the trajectory.
5. After selection, click **Confirm** to generate the historical trajectory. Meanwhile, the playback progress bar of the historical trajectory will pop up in the upper right corner.



6. Click the "play" icon on the right of the playback progress bar to start playback of the historical trajectory. The device will also start moving according to the trajectory points.



Note:

If you want to view the historical trajectory of other time periods or other devices, you can click  on the right of the playback progress bar to reselect the historical trajectory.

Related Thing Model Descriptions

Last updated: 2025-04-27 17:41:55

GPS Positioning Thing Model

- attribute identifier: GPS_Info
- Type: structure
- parameter description

| Field Name | Description | Required or Optional |
|------------|--|----------------------|
| latitude | GPS latitude; value range: -90 – 90; unit: degrees; 6 decimal places. | Yes |
| longitude | GPS longitude; value range: -180 – 180; unit: degrees; 6 decimal places. | Yes |

- Sample code

```
{
  "clientToken": "****",
  "method": "report",
  "params": {"GPS_Info":{"longitude":112.59014,"latitude":22.28014}}
}
```

GPS Positioning – Extended Object Model

- attribute identifier: GPS_ExtInfo
- Type: structure
- parameter description

| Field Name | Description | Required or Optional |
|------------|---|----------------------|
| latitude | GPS latitude; value range: -90 – 90; unit: degrees; 6 decimal places. | Yes |
| longitude | GPS longitude; value range: -180 – 180; unit: degrees; 6 decimal places. | Yes |
| altitude | <ul style="list-style-type: none">• Altitude, numeric value.• value range: -5000 – 99999.• Initial value: 0.• Unit: m. | No |
| gps_speed | <ul style="list-style-type: none">• GPS speed, integer.• value range: 0 – 1000.• Initial value: 0. | No |

| | | |
|----------------|---|----|
| | <ul style="list-style-type: none"> Unit: km/h. | |
| direction | <ul style="list-style-type: none"> Azimuth angle. value range: 0 – 360. Initial value: 0. Unit: degree. | No |
| location_state | Locating status, integer. <ul style="list-style-type: none"> 0: invalid. 1: valid. | No |
| satellites | Satellite number, integer. | No |
| gps_time | GPS time, time-based; timestamp precision to seconds, the time collected from satellites. | No |
| collect_time | Collection time, time-based; timestamp precision to seconds, the time when device data is collected. | No |

• Sample code

```
{
  "clientToken": "****",
  "method": "report",
  "params": { "GPS_ExtInfo":
{"longitude":112.59014,"latitude":22.28014,"altitude":200,"gps_speed":80,"direction":30}}
}
```

ⓘ Note:

If you need to customize the above parameters, latitude (lat) and longitude (lon) must be defined, and other parameters can be added or deleted depending on the actual situation.

Cellular Positioning Thing Model

- **Objective:** For devices such as 2G and 4G, individual Base Station Information can be reported to the cloud for locating.
- attribute identifier: Cell_Info
- Type: structure
- **parameter description**

| Field Name | Description | Required or Optional |
|------------|--|----------------------|
| mcc | Base station country code (460). | Yes |
| mnc | Base Station Network Code (00). | Yes |
| lac | Base station cell number (5-digit decimal number). | Yes |

| | | |
|--------------|--|-----|
| cid | Base station ID (5-digit decimal number). | Yes |
| rss | Signal strength of base station, unit dbm. | Yes |
| networktype | <ul style="list-style-type: none"> 1:GSM 2:CDMA 3:WCDMA 4:TD_CDMA 5:LTE | Yes |
| collect_time | The time when device acquisition gets Base Station Information. | No |

• Sample code

```
{
  "clientToken": "***",
  "method": "report",
  "params": {"LBS_BS": "mcc:460;mnc:13824;lac:3;cid:33;rss:-85;networktype:1"}
}
```

Wi-Fi Positioning Thing Model

- **Objective:** For Wi-Fi devices, multiple MAC addresses of nearby Wi-Fi router devices can be reported to the cloud to complete the locating.
- attribute identifier: Wifi_Info
- **Type:** struct
- **parameter description**

| Field Name | Description | Required or Optional |
|------------|---|----------------------|
| Mac | String type, MAC of the Wi-Fi router. | Yes |
| Rssi | Signal strength (ASU) of Wi-Fi router (int) | Yes |

• example code

```
{
  "clientToken": "***",
  "method": "report",
  "params": {"Wifi_Info": [{"Mac": "78a106a5b166", "Rssi": -81},
    {"Mac": "608f5c66b3f7", "Rssi": -82}, {"Mac": "282cb293a5d6", "Rssi": -83}]}
}
```

Resource Management

Last updated: 2025-04-27 17:42:12

Use Cases

The resource management feature is mainly used by developers to send face recognition libraries, image libraries, music libraries and other standard device resources to the device side, realizing the upload and download of resource content between the platform and the device.

This document introduces the method of use of the resource management feature, helping you quickly use the resource management feature to send resources to the device or upload device resources to IoT Explorer.

Operation Steps

1. Log in to the [IoT Development Platform console](#), select the corresponding instance, and you will be directed to the **Product Development** page by default.
2. Select **Value-added Service** in the left sidebar, then select **Resource Management**, and click **Add Resource**.



3. Input the resource name, select the associated product of the resource, upload the resource and its thumbnail, and meanwhile, you can edit the resource description.

添加资源

资源名称 *

请输入资源名称

支持中文、英文大小写、数字、部分常用符号（下划线，减号，括弧），必须以中文、英文或数字开头，长度不超过32个字符

所属产品 *

选择资源 *

点击选择资源

文件大小不能超过1024MB

资源缩略图

点击上传缩略图

上传资源缩略图，最多同时选择6张，支持JPG、JPEG、PNG格式

资源描述

对本次上传的资源进行描述和记录，请输入0-100个字符

对本次上传的资源进行描述和记录，请输入0-100个字符

保存

取消

4. Click the **Save** button. Then enter the resource management page. You can click **Task issuance** to issue the resource download task to the device.

资源管理

当前账号已上传资源大小: 20.07 MB / 1GB

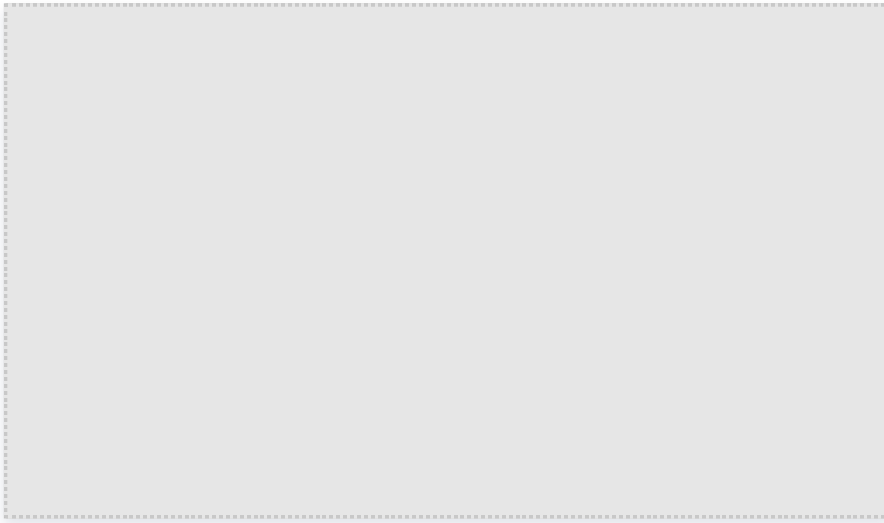
添加资源

网关1

请输入资源名称

| 资源名称 | 资源大小 | 所属产品 | 添加时间 | 操作 |
|-------|-----------|------|---------------------|--------------|
| 人脸识别库 | 660.80 KB | 网关1 | 2021-09-07 16:53:55 | 资源下发 查看详情 删除 |

5. Enter the resource issuance pop-up page. The issuance method is divided into single-device issuance and batch deployment. Here, select single-device issuance and click **Save**.

**Note:**

- Send to a single device: Send resource download tasks only to the selected device.
- Batch dispatch: Dispatch resource download tasks to all devices corresponding to DeviceName in the file by uploading a csv-formatted file. Up to 10,000 devices can be upgraded at a time.

6. After the resource distribution task is successfully created, click **View Task Details** to enter the task details page:



7. The task details page includes three parts: task information, task statistics, and device details.

资源管理 / 资源详情 / 任务详情

任务信息
任务ID: [redacted]
产品名称: 网关1
目标资源: 人脸识别库
下发方式: 单个下发
下发时间: 2021-09-07 17:02:10

任务统计

已推送: 100%

设备详情

全部设备(1) | 下发成功(0) | 待推送(0) | 已推送(1) | 下发中(0) | 下发失败(0) | [刷新](#) |

| 设备名称 | 最后更新时间 | 下发状态 | 状态详情 | 操作 |
|------------|---------------------|------|------|--------------------|
| gateway001 | 2021-09-07 17:02:13 | 已推送 | - | 取消 |

共 1 条

10 条 / 页 | [1](#) / 1 页

- task ID: The task ID of this resource distribution task.

- Product name: The product to which the file of this resource belongs.
 - Target resource: The resource name defined by the user when adding a resource.
 - Issuance method: The issuance method of resource tasks, which can be divided into individual dispatch and batch dispatch.
 - Release time: The time when the resource task is successfully created for distribution.
 - Distribution status: include delivery fail, distributing, pushed, pending, delivery successful.
8. Start the download of this resource task after the device is online. After the download is successful, the status changes to delivery success.



| 设备详情 | | | | |
|---|---------------------|------|------|----|
| <div>全部设备(1) 下发成功(1) 待推送(0) 已推送(0) 下发中(0) 下发失败(0) 刷新 <input type="text" value="请输入设备名称"/> Q</div> | | | | |
| 设备名称 | 最后更新时间 | 下发状态 | 状态详情 | 操作 |
| gateway001 | 2021-09-07 17:12:09 | 成功 | - | |
| <div>共 1 条 10 条 / 页 1 / 1 页</div> | | | | |

For the specific communication protocol between the device and the cloud, see [Resource Management Protocol](#) document.

Advanced Functions of the Product

Apply for This Advanced Function

Last updated: 2025-04-27 17:43:11

The advanced features of Tencent Cloud IoT Explorer are paid value-added services. If you need to use them, please [submit an application](#) for business consultation. The staff will contact you regarding the service purchase matters. Please see the following application process.

Application Steps

1. Log in to the [IoT Explorer console](#), select the region "China", and create a project and product. For details, see [Product Definition](#).
2. Click **Product Development** on the left sidebar to enter the product list page.
3. Select a created product to enter the product detail page, click **Thing Model Definition**, and select the required advanced functions in **Advanced Functions**.
4. When you hover over the enable button for the corresponding function, a notification will pop up on the page. You can click **Ticket** in the pop-up dialog box to enter the contact page for prospective customers of Real-time Interaction – IoT Edition.

| 标准功能 (8) 自定义功能 (0) 高级功能 (0) | | |
|-----------------------------|-------|---|
| 功能名称 | 收费方式 | 功能描述 |
| 实时音视频 | 按设备收费 | 专为物联网场景打造的音视频通话解决方案，支持智能终端与小程序、App 一对一音视频通话，支持一键呼叫、分机呼叫，提供应用端、设备端 SDK。 |
| 语音识别 | 按设备收费 | 腾讯连连语音识别服务是针对录音笔、翻译笔、会议办公等物联网设备场景提供物联网+AI语音识别的服务，包括文件识别、一句话识别、实时语音识别等各种物联网场景AI产品服务。 |
| 语音助手 | 按设备收费 | 腾讯连连物联网平台打通了业内主流的三方语音技能平台，提供快速接入三方平台并支持通过语音控制腾讯连连生态智能设备的能力。目前已经打通的语音技能平台包括Amazon Alexa、Google Assistant、百度小度和云小微。 |
| 酷狗音乐服务 | 按设备收费 | 腾讯连连整合酷狗音乐内容资源，面向消费物联网和产业互联网用音乐赋能智能硬件。 |

5. On the Contact Page for Prospective Customers of Real-time Interaction – IoT Version, fill in your relevant information and click **Submit**.

实时互动-物联版意向客户联络

实时互动-物联版为各行业的设备制造商、方案商及应用开发商提供一站式设备智能化服务。平台提供海量设备连接与管理能力及小程序应用开发能力，并打通腾讯云基础产品及 AI 能力，提升传统行业设备智能化的效率，降低用户的开发运维成本，助力用户业务发展。

您是*

☐ 个人 ☐ 企业

您所在企业名称*

请填写您现在所处的企业名称

企业类型*

☐ 芯片模组商 ☐ 硬件终端商 ☐ 应用开发商 ☐ 解决方案商 ☐ 其他

行业分类*

☐ 公共服务 ☐ 商业场所 ☐ 生活场所 ☐ 工业制造 ☐ 农林牧渔 ☐ 建筑工地 ☐ 其他

企业业务场景*

请填写您的企业业务场景描述

业务阶段*

☐ 未开发 ☐ 开发中 ☐ 测试中 ☐ 已上线 ☐ 大规模应用中

联系人*

请填写您的真实姓名

联系电话*

请填写可以联系到您的电话

联系邮箱*

请填写可以联系到您的邮箱

您需要咨询的内容*

请填写您的业务需求、问题描述或使用反馈，我们将尽快联系您

☐ 我已阅读和同意 [《腾讯云隐私保护声明》](#) 内容

提交

6. After the request is submitted successfully, the staff will contact you about the service purchase matter within 7 working days.

提交成功

感谢您的宝贵时间和对我们的支持。

完成

Version Change

Last updated: 2025-04-27 17:43:24

Tencent Real-Time Communication (TRTC)

Device-Side SDK

Version V3.3.0

- Release date: January 14, 2021
- System platform: Android
- Development language: Java
- Development environment: Android Studio
- Update content
 - Add RTC call SDK and demo for scenarios in explorer.
 - Fix several issues.

Version V1.1.1

- Release date: January 14, 2021
- Development language: C language
- Development environment: Linux/Windows, Cmake/GNU Make
- Update content
 - Implement audio and video calls with the application side.
 - Fix several issues.

Application-Side SDK

Version V1.3.0

- Release date: January 14, 2021
- System platform: iOS / Android
- Development language: Objective-C / Java
- Development environment: Mac, Xcode / Android Studio
- Update content: Optimize the signaling logic of TRTC and enhance user experience.

Version V1.2.1

- Release date: December 10, 2020
- System platform: iOS / Android
- Development language: Objective-C / Java
- Development environment: Mac, Xcode / Android Studio
- Update content: Fix the SDK configuration of V1.2.0 version, delete the i386 architecture.

Version V1.2.0

- Release date: Dec 2, 2020

- System platform: iOS / Android
- Development language: Objective-C / Java
- Development environment: Mac, Xcode / Android Studio
- Update content: Provide the capacity to meet the requirements of integrating audio and video call scenarios.

Automatic Speech Recognition (ASR)

Device-Side SDK sample

Version V3.1.5

- Release date: December 2, 2020
- Development language: C language
- Development environment: Linux/Windows, Cmake/GNU Make
- The content is as follows:
 - Newly-added resource management features and examples.
 - Newly-added ASR features and examples.
 - Add file operation for HAL layer adaptation api.
 - Optimize multithreaded operations.
 - MQTT sample code implements data interaction based on the Data Template Protocol.
 - Change the version number to V3.1.5.

Tencent Real – Time Communication (TRTC)

Last updated: 2025-04-27 17:43:38

This document introduces how to open and use the tencent real-time communication (TRTC) service on Tencent Cloud IoT development platform.

Prerequisites

Activating devices to enable the tencent real-time communication (TRTC) service is only supported in the following regions:

Chinese mainland (exclude Hong Kong, Macao, and Taiwan region)

Applying for Enabling Tencent Real-Time Communication (TRTC) Service

Tencent Real-Time Communication (TRTC) service is a paid value-added feature and does not offer a free trial. You can [Online Consultation](#) for help. The staff will connect with you on matters related to service purchase.

Operation Steps

Step 1: Enabling Tencent Real-Time Communication (TRTC) Service

1. Log in to the [IoT Development Platform Console](#), select the region "China" and create a product. For details, see [Product Definition](#).
2. Click **Product Development** on the left sidebar to enter the product list page.
3. Select a created product to enter the product detail page, click **Thing Model Definition**, and in **Advanced Features**, click the button at "TRTC" to activate successfully.

| 标准功能 (8) 自定义功能 (0) 高级功能 (0) | | | | | |
|-----------------------------|-------|---|-------------------------------------|----------------------|--|
| 功能名称 | 收费方式 | 功能描述 | 启用 | 操作 | |
| 实时音视频 | 按设备收费 | 专为物联网场景打造的音视频通话解决方案，支持智能终端与小程序、App 一对一音视频通话，支持一键呼叫、分組呼叫，提供应用端、设备端 SDK。 | <input checked="" type="checkbox"/> | 查看文档 | |
| 语音识别 | 按设备收费 | 腾讯连连语音识别服务是针对录音笔、翻译笔、会议办公等物联网设备场景提供物联网+AI语音识别的服务，包括文件识别、一句话识别、实时语音识别等各种物联网场景AI产品服务。 | <input type="checkbox"/> | 查看文档 | |
| 语音助手 | 按设备收费 | 腾讯连连物联网平台打通了业内主流的三方语音技能平台，提供快速接入三方平台并支持通过语音控制腾讯连连生态智能设备的能力。目前已经打通的语音技能平台包括Amazon Alexa、Google Assistant、百度小度和云小微。 | <input type="checkbox"/> | 查看文档 | |
| 酷狗音乐服务 | 按设备收费 | 腾讯连连整合酷狗音乐内容资源，面向消费物联网和产业互联网用音乐赋能智能硬件。 | <input type="checkbox"/> | 查看文档 | |

Step Two: Create a Device

Click the **Product Development** in the left sidebar of the console, select a product to enter the Product Details Page, click **Device Debugging** > **Create New Device**, fill in relevant information and click **Save** just.

新建设备

×

所属产品

设备名称 *

支持英文、数字、下划线的组合，最多不超过48个字符

保存

取消

Note:

- When there are devices under the product, the status of the value-added service switch cannot be changed.
- If you delete a device with activated VAS, the purchase License quantity cannot be recovered.

Device Access Guide

Tencent Cloud IoT development platform's tencent real-time communication (TRTC) service provides Mini Program Plugins and iOS, Android App SDKs on the application side. On the equipment side, Android platform and Hisilicon DV300 platform are now supported.

Mini Program Plugin

- Mini Program Plugin operation guide. For details, see [Mini Program Plugin operation guide](#).
- Glossary and interface usage instructions of the plug-in. For details, see [Tencent Lianlian Mini Program Plug-in Instructions](#).

SDK Acquisition

- The application-side SDK is hosted on Github. You can access Github to download the latest version of [Android SDK](#) and [iOS SDK](#).
- Device-side SDK is hosted on Github. You can access Github to download the latest version of [Android SDK](#) and [Linux C SDK \(submit an inquiry form for consultation\)](#).

Development Guide

Please refer to the following guide:

- [Guide for Android SDK Integration on the Application Side](#)
- [Guide for iOS SDK Integration on the Application Side](#)
- [Guide for Android SDK Integration on the Device Side](#)
- [Integration Guide for Device-side C SDK \(Please submit an inquiry form for consultation\)](#)

QQ Music Service

Last updated: 2025-04-27 17:43:54

This document shows you how to open and use Tencent Cloud IoT Explorer IoT+QQ music service.

Prerequisites

QQ music service can only be enabled by activating devices in the following regions:

Chinese mainland (exclude Hong Kong, Macao, and Taiwan region)

Applying for Enabling QQ Music Service

QQ music service is a paid value-added feature. There is no free trial. You can [Online Consultation](#) for help. The staff will connect with you on service purchase matters.

Operation Steps

Step 1: Enabling QQ Music Service

1. Log in to the [IoT Explorer console](#) , select the region "China" and create a product. For details, see [Product Management](#) .
2. Click **Product Development** on the left sidebar to enter the product list page.
3. Select a created product to enter the product detail page, click **Thing Model Definition**, and in **Advanced Function**, click the switch button at "QQ Music Service". Turn on the switch to activate successfully.

产品开发 / 自定义品类 新项目

回到旧版 帮助文档

导入物模型 查看物模型JSON

物模型定义帮助

标准功能 (0) 自定义功能 (0) 高级功能 (1)

| 功能名称 | 收费方式 | 功能描述 | 启用 | 操作 |
|-------------|-------|---|-------------------------------------|--|
| 语音助手 | 按设备收费 | 腾讯连连物联网平台打通了业内主流的三方语音技能平台，提供快速接入三方平台并支持通过语音控制腾讯连连生态智能设备的能力。目前已经打通的语音技能平台包括Amazon Alexa、Google Assistant、百度小度和云小微。 | <input type="checkbox"/> | 查看文档 |
| 酷狗音乐服务 | 按设备收费 | 腾讯连连整合酷狗音乐内容资源，面向消费物联网和产业互联网用音乐赋能智能硬件。 | <input type="checkbox"/> | 查看文档 |
| IoT Video服务 | 按设备收费 | 为客户提供视频连接、存储和智能应用服务，安全高效。客户可简单快速地实现设备接入、宽带传输、云端存储、远程观看等一站式视频场景能力，并提供丰富的 AI 算法模型实现具体场景的智能解析及应用，实现云边协同智能应用。 | <input type="checkbox"/> | 查看文档 |
| QQ 音乐服务 | 按设备收费 | 腾讯连连整合 QQ 音乐内容资源，面向消费物联网和产业互联网用音乐赋能智能硬件。 | <input checked="" type="checkbox"/> | 查看文档 收起物模型 |

| 功能类型 | 功能名称 | 标识符 | 数据类型 | 读写类型 | 数据定义 |
|------|----------|--------------------|------|------|--|
| 属性 | 播放模式 | _sys_play_mode | 枚举整型 | 读写 | 0 - 顺序播放 1 - 单曲循环 2 - 随机播放 |
| 属性 | 当前曲目播放进度 | _sys_play_position | 整数值 | 读写 | 数值范围: 0-7200 初始值: 0 步长: 1 单位: S |
| | | | | | 0 - 标准 |

4. After enabling the advanced function switch of "QQ Music Service", click to expand the Thing Model. You can view the functional design of the Thing Model that interacts with the cloud, which collaborates with the configuration on the device-side SDK.

Note:

Due to the involvement in the interaction protocol with the platform and application side, the system Thing Model cannot be modified or deleted.

Step Two: Create a Device

Click the **Product Development** in the left sidebar of the console, select a product to enter the product detail page, click **Device Debugging > Create New Device**, fill in relevant information and click **Save** just.

新建设备

所属产品

设备名称 *

支持英文、数字、下划线的组合，最多不超过48个字符

保存

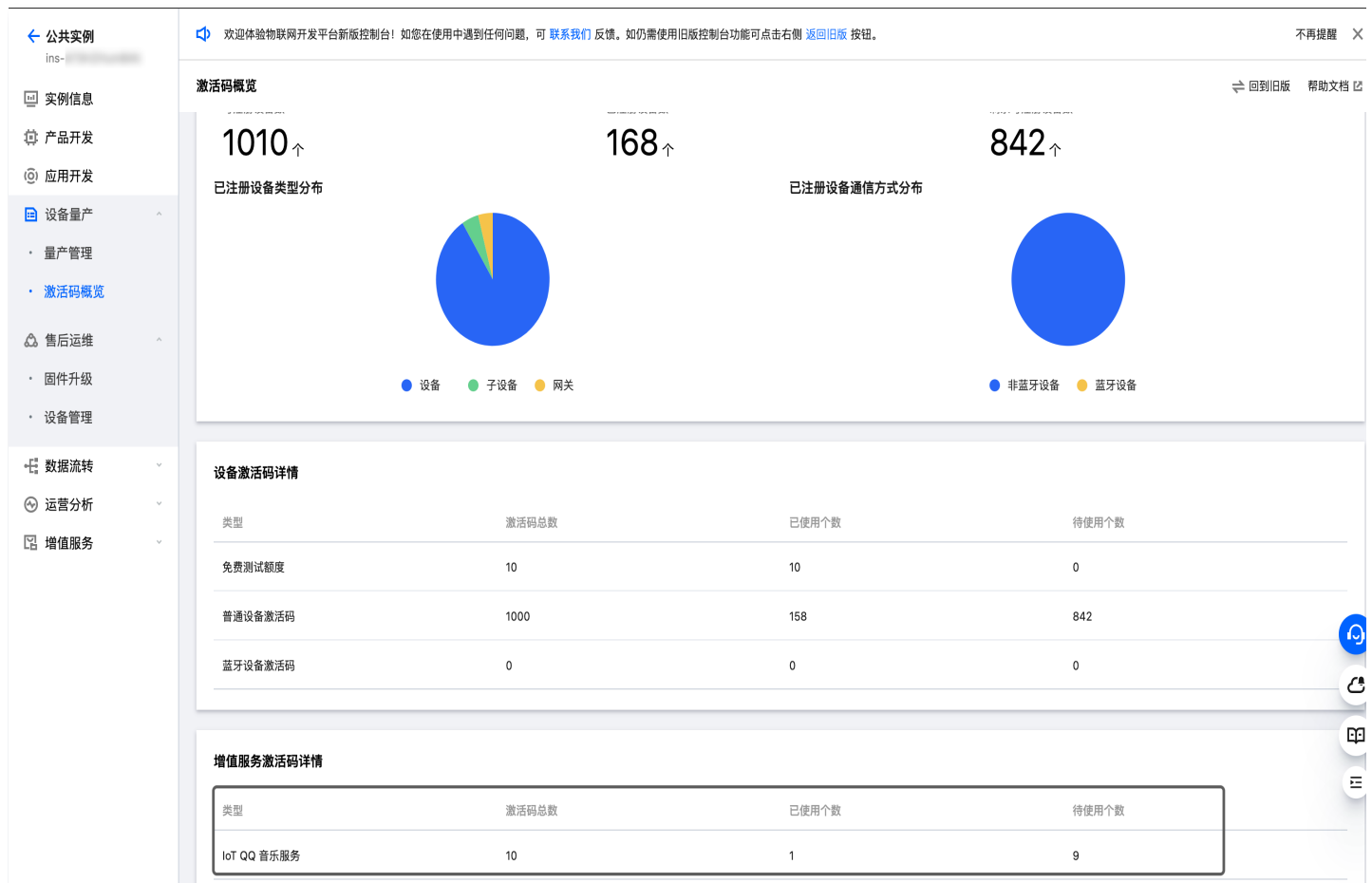
取消

Note:

- When there are devices under the product, the switch status of the value-added service cannot be changed.
- If you delete a device with activated VAS, the purchased License quantity cannot be recovered.

Step 3: Managing Activation Codes

For the product with the advanced function of QQ Music enabled, one IoT QQ Music service activation code is consumed for each created device. You can view the usage of value-added service activation codes of the current account in **Mass Device Production > Activation Code Overview**.



Step Four: Application Development Guide

Tencent Cloud IoT Explorer QQ Music access service provides two integration solutions for application development: official mini program custom panel development H5 SDK and application-side API on the application side. You can choose an appropriate application development method based on your business scenario.

Custom Panel Development for Mini Programs H5 SDK

Complete the panel development of the device that integrates the QQ Music service based on the H5 SDK within the official mini program. You can also quickly verify the related features through the QQ Music-related H5 Demo panel. Bind the device through the "Tencent Lianlian" mini program searched in WeChat and then it can be used.

For the operation guide, see the following:

- [H5 custom development SDK – music service](#)
- [H5 panel Demo example code](#)

Application Side API

Self-create a third-party brand mini program based on the application-side API. Call and query the playback of QQ Music music content service resources through the application-side API form to implement relevant features.

Operation guide: For details, see [Application-side API Documentation – Music Service](#).

Voice Recognition

Last updated: 2025-04-27 17:44:16

this document introduces how to open and use the speech recognition service of Tencent Cloud IoT Explorer.

Prerequisites

Activating devices to enable the speech recognition service is only supported in the following regions:

Chinese mainland (exclude Hong Kong, Macao, and Taiwan region)

Applying for Enabling ASR Service

The speech recognition service is a paid value-added service and does not provide a free trial. You can [Online Consultation](#) to ask for help. The staff will connect with you on service purchase matters.

ASR Service Access Guide

Step 1: Enable the ASR Service

1. Log in to the [IoT Explorer console](#), select the region "China", and create a project and product. For details, see [product definition](#).
2. Click **Product Development** on the left sidebar to enter the product list page.
3. Select a created product to enter the product detail page, click **Thing Model Definition**, and click the button at "Speech Recognition" in **Advanced Function** to activate successfully.

| 标准功能 (8) 自定义功能 (0) 高级功能 (0) | | | | |
|-----------------------------|-------|--|-------------------------------------|----------------------|
| 功能名称 | 收费方式 | 功能描述 | 启用 | 操作 |
| 实时音视频 | 按设备收费 | 专为物联网场景打造的音视频通话解决方案, 支持智能终端与小程序、App 一对一音视频通话, 支持一键呼叫、分组呼叫, 提供应用端、设备端 SDK。 | <input type="checkbox"/> | 查看文档 |
| 语音识别 | 按设备收费 | 腾讯连连语音识别服务是针对录音笔、翻译笔、会议办公等物联网设备场景提供物联网+AI语音识别的服务, 包括文件识别、一句话识别、实时语音识别等各种物联网场景AI产品服务。 | <input checked="" type="checkbox"/> | 查看文档 |
| 语音助手 | 按设备收费 | 腾讯连连物联网平台打通了业内主流的三方语音技能平台, 提供快速接入三方平台并支持通过语音控制腾讯连连生态智能设备的能力。目前已经打通的语音技能平台包括Amazon Alexa、Google Assistant、百度小度和云小微。 | <input type="checkbox"/> | 查看文档 |
| 酷狗音乐服务 | 按设备收费 | 腾讯连连整合酷狗音乐内容资源, 面向消费物联网和产业互联网用音乐赋能智能硬件。 | <input type="checkbox"/> | 查看文档 |

Step Two: Create a Device

Click the **Product Development** in the left sidebar of the console, select a product to enter the product detail page, click **Device Debugging** > **Create New Device**, fill in relevant information and click **Save**.

新建设备

所属产品

设备名称 *

支持英文、数字、下划线的组合，最多不超过48个字符

保存 取消

Note:

- When there are devices under the product, the status of the value-added service switch cannot be changed.
- If you delete a device with activated value-added services, the purchased License quantity is irrecoverable.

Device Connectivity Guide

Tencent Cloud IoT Explorer uses the Automatic Speech Recognition (ASR) product feature and combines with the [ASR](#) sample `asr_data_template_sample` of C SDK to quickly experience the ASR functionality.

Configure Device Information

Modify the device information configuration `device_info.json`, fill in the created ASR product and device information correspondingly. To get device information, refer to [Device Debugging](#).

数据模板 > 设备开发 > 交互开发 > 4 设备调试 > 5 批量投产

← asr

设备信息 设备属性 设备日志 设备事件 设备行为 设备上下线日志 在线调试 扩展信息 设备调试日志

设备信息

设备名称 asr

设备密钥

激活时间 -

所属产品 语音识别产品

产品ID

最后上线时间 -

设备创建时间 2020-12-08 16:42:31

设备状态 未激活

固件版本 -

Modify the data template attributes that the ASR result return depends on to system properties and decouple from the user's data template. Sample code is as follows:

```
{
  "auth_mode": "KEY",
  "productId": "PRODUCT_ID",
  //ASR product ID
```



```

"productSecret": "YOUR_PRODUCT_SECRET",
"deviceName": "YOUR_DEV_NAME", //ASR device name
"key_deviceinfo": {
    "deviceSecret": "YOUR_IOT_PSK" //ASR device key
},
"cert_deviceinfo": {
    "devCertFile": "YOUR_DEVICE_CERT_FILE_NAME",
    "devPrivateKeyFile": "YOUR_DEVICE_PRIVATE_KEY_FILE_NAME"
},
"region": "china"
}

```

Modify Compilation Options

Modify CMakeLists.txt (taking **key authentication device** as an example) to enable ASR and resource management features (the ASR feature depends on the resource management feature). There are two ways:

- Use cmake to compile

```

set (BUILD_TYPE "release")
set (COMPILE_TOOLS "gcc")
set (PLATFORM "linux")
set (FEATURE_RESOURCE_UPDATE_ENABLED ON)
set (FEATURE_ASR_ENABLED ON)
set (FEATURE_AUTH_MODE "KEY")

```

Execute script compilation.

```
./cmake_build.sh
```

- Use makefile to compile

```

PLATFORM_CC          = gcc
PLATFORM_AR           = ar
PLATFORM_OS           = linux
FEATURE_RESOURCE_UPDATE_ENABLED = y
FEATURE_ASR_ENABLED  = y

```

Execute make compilation:

```
make
```

asr_data_template_sample example output is located in the output/release/bin folder.

Example

- Use the ASR feature. Perform the following operations:

1.1 Call the api `IOT_Asr_Init` to initialize `asr_client`.

1.2 If it is a file or single-sentence recognition, call `IOT_Asr_RecordFile_Request` , input the request parameter, file name and callback; if it is real-time speech, call `IOT_Asr_Realttime_Request` , input real-time audio data, request parameter and callback. Meanwhile, the encoding of audio data needs to be completed before initiating the request. The return values of the two APIs are `request_id` when the calls are successful. The corresponding results will be returned in the callback. The returned result will carry the corresponding `request_id` . That is, the result return of ASR only supports asynchronously.

- `asr_data_template_sample` shows how to use the above API in three use cases of ASR: **file, one sentence, and real-time speech**. By changing the macro definition `DEMO_ASR` in the sample, you can select the corresponding sample scenario. The data source for all three scenarios uses the test file `tools/test_file/test.wav` . After compilation, this file will be copied to `output/release/bin/test_file/test.wav` . If you need to use your own test file, you can replace this test file.

```
#define DEMO_ASR_FILE           0
#define DEMO_ASR_REALTIEM      1
#define DEMO_ASR_SENTENCE      2
#define DEMO_ASR                DEMO_ASR_SENTENCE
```

Usage Scenario Description

Three use cases and request parameters of ASR, including **file, one sentence, real-time speech**. For details, please refer to [ASR official website document](#).

Operation Instructions

- Configure `DEMO_ASR` as `DEMO_ASR_FILE` , corresponding to the usage scenario of example ASR **file** recognition:

```
./asr_data_template_sample
INF|2020-11-10 16:24:19|qcloud_iot_device.c|iot_device_info_set(55): SDK_Ver:
3.1.4, Product_ID: W0MHQCSFN5, Device_Name: dev002
INF|2020-11-10 16:24:19|mqtt_client.c|IOT_MQTT_Construct(125): mqtt connect with
id: q4ZhF success
INF|2020-11-10 16:24:19|asr_data_template_sample.c|event_handler(88): subscribe
success, packet-id=2717
INF|2020-11-10 16:24:19|data_template_client.c|IOT_Template_Construct(936): Sync
device data successfully
INF|2020-11-10 16:24:19|asr_data_template_sample.c|main(379): Cloud Device
Construct Success
INF|2020-11-10
16:24:19|asr_data_template_sample.c|_register_data_template_property(227): data
template property=asr_response registered.
INF|2020-11-10 16:24:19|asr_data_template_sample.c|main(394): Register data
template propertys Success
INF|2020-11-10 16:24:20|asr_data_template_sample.c|event_handler(88): subscribe
success, packet-id=2718
INF|2020-11-10 16:24:22|asr_data_template_sample.c|main(492): record file
test.wav's request_id 1
```

```

INF|2020-11-10 16:24:24|asr_data_template_sample.c|main(492): record file
test.wav's request_id 2
INF|2020-11-10 16:24:26|asr_data_template_sample.c|main(492): record file
test.wav's request_id 3
INF|2020-11-10 16:24:27|asr_data_template_sample.c|main(492): record file
test.wav's request_id 4
INF|2020-11-10 16:24:29|asr_data_template_sample.c|main(492): record file
test.wav's request_id 5
INF|2020-11-10 16:24:31|asr_data_template_sample.c|main(492): record file
test.wav's request_id 6
INF|2020-11-10 16:24:32|asr_data_template_sample.c|asr_result_cb(340):
request_id:3: 1/1 text:[0:0.000,0:2.800] Beijing Science and Technology Museum.
INF|2020-11-10 16:24:32|asr_data_template_sample.c|asr_result_cb(340):
request_id:2: 1/1 text:[0:1.040,0:3.100] Beijing Science and Technology Museum.
INF|2020-11-10 16:24:33|asr_data_template_sample.c|main(492): record file
test.wav's request_id 7
INF|2020-11-10 16:24:35|asr_data_template_sample.c|main(492): record file
test.wav's request_id 8
INF|2020-11-10 16:24:36|asr_data_template_sample.c|asr_result_cb(340):
request_id:1: 1/1 text:[0:0.000,0:2.800] Beijing Science and Technology Museum.
INF|2020-11-10 16:24:36|asr_data_template_sample.c|asr_result_cb(340):
request_id:4: 1/1 text:[0:0.000,0:2.800] Beijing Science and Technology Museum.

```

- **Configure** DEMO_ASR as DEMO_ASR_REALTIME . The corresponding usage scenario for the ASR one-sentence recognition sample code:

```

./asr_data_template_sample
INF|2020-11-10 16:21:00|qcloud_iot_device.c|iot_device_info_set(55): SDK_Ver:
3.1.4, Product_ID: WOMHQCSFN5, Device_Name: dev002
INF|2020-11-10 16:21:00|mqtt_client.c|IOT_MQTT_Construct(125): mqtt connect with
id: oLbdC success
INF|2020-11-10 16:21:00|asr_data_template_sample.c|event_handler(88): subscribe
success, packet-id=53209
INF|2020-11-10 16:21:00|data_template_client.c|IOT_Template_Construct(936): Sync
device data successfully
INF|2020-11-10 16:21:00|asr_data_template_sample.c|main(379): Cloud Device
Construct Success
INF|2020-11-10
16:21:00|asr_data_template_sample.c|_register_data_template_property(227): data
template property=asr_response registered.
INF|2020-11-10 16:21:00|asr_data_template_sample.c|main(394): Register data
template propertys Success
INF|2020-11-10 16:21:01|asr_data_template_sample.c|event_handler(88): subscribe
success, packet-id=53210
INF|2020-11-10 16:21:03|asr_data_template_sample.c|main(508): record file
test.wav's request_id 1
INF|2020-11-10 16:21:04|asr_data_template_sample.c|main(508): record file
test.wav's request_id 2
INF|2020-11-10 16:21:05|asr_data_template_sample.c|main(508): record file
test.wav's request_id 3

```

```

INF|2020-11-10 16:21:06|asr_data_template_sample.c|main(508): record file
test.wav's request_id 4
INF|2020-11-10 16:21:06|asr_data_template_sample.c|asr_result_cb(340):
request_id:1: 1/1 text:Beijing Science and Technology Museum.
INF|2020-11-10 16:21:07|asr_data_template_sample.c|main(508): record file
test.wav's request_id 5
INF|2020-11-10 16:21:07|asr_data_template_sample.c|asr_result_cb(340):
request_id:2: 1/1 text:Beijing Science and Technology Museum.
INF|2020-11-10 16:21:07|asr_data_template_sample.c|asr_result_cb(340):
request_id:3: 1/1 text:Beijing Science and Technology Museum.
INF|2020-11-10 16:21:08|asr_data_template_sample.c|main(508): record file
test.wav's request_id 6
INF|2020-11-10 16:21:08|asr_data_template_sample.c|asr_result_cb(340):
request_id:4: 1/1 text:Beijing Science and Technology Museum.

```

- **Configure** DEMO_ASR as DEMO_ASR_SENTENCE , corresponding to the usage scenario of example ASR real-time speech recognition:

```

./asr_data_template_sample
INF|2020-11-10 16:25:39|qcloud_iot_device.c|iot_device_info_set(55): SDK_Ver:
3.1.4, Product_ID: WOMHQCSFN5, Device_Name: dev002
INF|2020-11-10 16:25:39|mqtt_client.c|IOT_MQTT_Construct(125): mqtt connect with
id: 1tBx2 success
INF|2020-11-10 16:25:39|asr_data_template_sample.c|event_handler(88): subscribe
success, packet-id=45977
INF|2020-11-10 16:25:39|data_template_client.c|IOT_Template_Construct(936): Sync
device data successfully
INF|2020-11-10 16:25:39|asr_data_template_sample.c|main(379): Cloud Device
Construct Success
INF|2020-11-10
16:25:39|asr_data_template_sample.c|_register_data_template_property(227): data
template property=asr_response registered.
INF|2020-11-10 16:25:39|asr_data_template_sample.c|main(394): Register data
template propertys Success
INF|2020-11-10 16:25:40|asr_data_template_sample.c|event_handler(88): subscribe
success, packet-id=45978
INF|2020-11-10 16:25:42|asr_data_template_sample.c|main(556): realtime request_id 1
INF|2020-11-10 16:25:43|asr_data_template_sample.c|main(556): realtime request_id 2
INF|2020-11-10 16:25:44|asr_data_template_sample.c|main(556): realtime request_id 3
INF|2020-11-10 16:25:44|asr_data_template_sample.c|asr_result_cb(340):
request_id:1: 0/0 text:NULL
INF|2020-11-10 16:25:45|asr_data_template_sample.c|main(556): realtime request_id 4
INF|2020-11-10 16:25:45|asr_data_template_sample.c|asr_result_cb(340):
request_id:2: 1/0 text:NULL
INF|2020-11-10 16:25:46|asr_data_template_sample.c|main(556): realtime request_id 5
INF|2020-11-10 16:25:46|asr_data_template_sample.c|asr_result_cb(340):
request_id:3: 2/0 text:NULL
INF|2020-11-10 16:25:46|asr_data_template_sample.c|main(556): realtime request_id 6
INF|2020-11-10 16:25:46|asr_data_template_sample.c|asr_result_cb(340):
request_id:4: 3/0 text:NULL

```

```
INF|2020-11-10 16:25:47|asr_data_template_sample.c|main(556): realtime request_id 7
INF|2020-11-10 16:25:47|asr_data_template_sample.c|asr_result_cb(340):
request_id:5: 4/0 text:NULL
INF|2020-11-10 16:25:48|asr_data_template_sample.c|main(556): realtime request_id 8
INF|2020-11-10 16:25:48|asr_data_template_sample.c|asr_result_cb(340):
request_id:6: 5/0 text:NULL
INF|2020-11-10 16:25:49|asr_data_template_sample.c|main(556): realtime request_id 9
INF|2020-11-10 16:25:49|asr_data_template_sample.c|asr_result_cb(340):
request_id:7: 6/0 text:NULL
INF|2020-11-10 16:25:50|asr_data_template_sample.c|main(556): realtime request_id
10
INF|2020-11-10 16:25:50|asr_data_template_sample.c|asr_result_cb(340):
request_id:8: 7/0 text:NULL
INF|2020-11-10 16:25:51|asr_data_template_sample.c|main(556): realtime request_id
11
INF|2020-11-10 16:25:51|asr_data_template_sample.c|asr_result_cb(340):
request_id:9: 8/0 text:NULL
INF|2020-11-10 16:25:52|asr_data_template_sample.c|main(556): realtime request_id
12
INF|2020-11-10 16:25:52|asr_data_template_sample.c|asr_result_cb(340):
request_id:10: 9/0 text:NULL
INF|2020-11-10 16:25:53|asr_data_template_sample.c|main(556): realtime request_id
13
INF|2020-11-10 16:25:53|asr_data_template_sample.c|asr_result_cb(340):
request_id:11: 10/0 text:NULL
INF|2020-11-10 16:25:53|asr_data_template_sample.c|main(556): realtime request_id
14
INF|2020-11-10 16:25:53|asr_data_template_sample.c|asr_result_cb(340):
request_id:12: 11/0 text:NULL
INF|2020-11-10 16:25:54|asr_data_template_sample.c|main(556): realtime request_id
15
INF|2020-11-10 16:25:54|asr_data_template_sample.c|asr_result_cb(340):
request_id:13: 12/0 text:Beijing Technology.
INF|2020-11-10 16:25:55|asr_data_template_sample.c|main(556): realtime request_id
16
INF|2020-11-10 16:25:55|asr_data_template_sample.c|asr_result_cb(340):
request_id:14: 13/0 text:Beijing Technology.
INF|2020-11-10 16:25:56|asr_data_template_sample.c|main(556): realtime request_id
17
INF|2020-11-10 16:25:56|asr_data_template_sample.c|asr_result_cb(340):
request_id:15: 14/0 text:Beijing Science and Technology Museum.
INF|2020-11-10 16:25:57|asr_data_template_sample.c|main(556): realtime request_id
18
INF|2020-11-10 16:25:57|asr_data_template_sample.c|asr_result_cb(340):
request_id:16: 15/0 text:NULL
INF|2020-11-10 16:25:58|asr_data_template_sample.c|main(556): realtime request_id
19
INF|2020-11-10 16:25:58|asr_data_template_sample.c|asr_result_cb(340):
request_id:17: 16/0 text:NULL
```

```
INF|2020-11-10 16:25:59|asr_data_template_sample.c|main(556): realtime request_id
20
INF|2020-11-10 16:25:59|asr_data_template_sample.c|asr_result_cb(340):
request_id:18: 17/0 text:NULL
INF|2020-11-10 16:26:00|asr_data_template_sample.c|main(556): realtime request_id
21
```

Retrieve the Device-Side SDK sample

The device-side SDK sample is hosted on Github and can be accessed on Github to download the latest version of [device-side SDK sample](#).

Custom H5 Panel Development

The device for speech recognition supports custom development of the H5 control panel. Device control can be performed in the Tencent Lianlian Mini Program. For details, see H5 customization development [ASR](#).

Voice Assistant

Last updated: 2025-04-27 17:44:34

this document introduces how to open and use the voice assistant service of Tencent Cloud IoT Explorer.

Prerequisites

Activating devices to enable the voice assistant service is only supported in the following regions:

Chinese mainland (exclude Hong Kong, Macao, and Taiwan region)

Apply for Enabling Voice Assistant Service

The voice assistant service is a paid value-added service and does not offer a free trial. You can [online consultation](#) to ask for help. The staff will connect with you on service purchase matters.

Voice Assistant Service Access Guide

Step 1: Enabling Voice Assistant Service

1. Log in to the [IoT Explorer console](#), select the region "China", and create a project and product. For details, see [Product Definition](#).

ⓘ Note:

When creating a product, it is recommended to select the product category: Intelligent Life/Audio-Visual Office/Smart Speaker. Developers can also choose other categories for custom development.

2. Click **Product Development** on the left sidebar to enter the product list page.
3. Select a created product to enter the product detail page, click **Thing Model Definition**, and click the button at "voice assistant" in **Advanced Function** to activate successfully.

| 标准功能 (8) ⓘ | | 自定义功能 (0) ⓘ | 高级功能 (0) | | |
|------------|-------|--|-------------------------------------|----------------------|--|
| 功能名称 | 收费方式 | 功能描述 | 启用 | 操作 | |
| 实时音视频 | 按设备收费 | 专为物联网场景打造的音视频通话解决方案, 支持智能终端与小程序、App 一对一音视频通话, 支持一键呼叫、分组呼叫, 提供应用端、设备端 SDK。 | <input type="checkbox"/> | 查看文档 | |
| 语音识别 | 按设备收费 | 腾讯连连语音识别服务是针对录音笔、翻译笔、会议办公等物联网设备场景提供物联网+AI语音识别的服务, 包括文件识别、一句话识别、实时语音识别等各种物联网场景AI产品服务。 | <input type="checkbox"/> | 查看文档 | |
| 语音助手 | 按设备收费 | 腾讯连连物联网平台打通了业内主流的三方语音技能平台, 提供快速接入三方平台并支持通过语音控制腾讯连连生态智能设备的能力。目前已经打通的语音技能平台包括Amazon Alexa、Google Assistant、百度小度和云小微。 | <input checked="" type="checkbox"/> | 查看文档 | |
| 酷狗音乐服务 | 按设备收费 | 腾讯连连整合酷狗音乐内容资源, 面向消费物联网和产业互联网用音乐赋能智能硬件。 | <input type="checkbox"/> | 查看文档 | |

Step Two: Bind Application Information of Cloud Xiaowei Open Platform

ⓘ Note:

Ensure that the App Key and App Secret are not leaked.

1. Log in to the open platform of [Tencent Cloud Xiaowei](#), click **Device Platform > Create Application** to enter the application creation page and fill in relevant information.

The screenshot shows the 'Application Scenario' (应用场景) step of the application creation process. At the top, there are four steps: 1. Application Scenario (selected), 2. Application Name, 3. Skill Configuration, and 4. Application Release. Below the steps, there are radio buttons for 'Device System' (设备系统): Android, Linux, RTOS, and Others. The main area is titled 'Application Scenario' (应用场景) and contains a grid of 11 icons representing different use cases: Mobile Application (手机应用), Screen-free Audio Device (无屏音箱), Car Unit (车机), TV (电视), Smartwatch (手表), Headphones (耳机), WeChat Public Account (微信公众号), Mini Program (小程序), Screen-free Robot (无屏机器人), Screened Robot (有屏机器人), and Others (其他). At the bottom, there are radio buttons for 'Application Mode' (应用模式): Standard Mode (标准模式) and Child Mode (儿童模式). A blue 'Next Step' (下一步) button is located at the bottom right.

- Device system: including Android, Linux, RTOS and others.
 - Use cases: including mobile applications, screen-free audio devices, in-vehicle entertainment systems, etc. This time, select "screen-free audio device".
 - Application mode: Includes "standard mode" and "Child Mode".
2. Click **Next**, enter the application name and application description.

The screenshot shows the 'Application Name' (应用名称) step of the application creation process. At the top, there are four steps: 1. Application Scenario, 2. Application Name (selected), 3. Skill Configuration, and 4. Application Release. Below the steps, there is a text input field for 'Application Name' (应用名称) with a placeholder suggestion: '建议以公司加产品命名，如 优必选_悟空机器人'. Below this is a larger text area for 'Application Description' (应用描述). At the bottom right, there are two buttons: 'Previous Step' (上一步) and 'Next Step' (下一步).

3. Click **Next**, select **customization > import from custom skill library**, check **Tencent Lianlian** and click **confirm**.



4. Enter the version number in the format of x.x.x.x, for example, 1.0.0.0. The version number cannot be changed after it takes effect. The version number is primarily used to cooperate with the update of the terminal version, and can also be used to distinguish the created applications. Click **Next**.



5. Fill in the release note, click **Complete**, and then click **Publish** to successfully create the application.

The screenshot shows the '应用发布' (Application Release) step in the Tencent Cloud IoT Explorer console. At the top, a progress bar indicates four steps: 1. 应用场景 (Application Scenario), 2. 应用名称 (Application Name), 3. 技能配置 (Skill Configuration), and 4. 应用发布 (Application Release), with the fourth step being the active one. The form contains the following fields:

- 应用名称 (Application Name): ffhhh
- 应用描述 (Application Description): [blurred]
- 应用类型 (Application Type): 无屏音箱 (Screenless Speaker)
- 操作系统 (Operating System): OTHER
- AppKey: [blurred]
- AccessToken: [blurred]
- Product ID: [blurred]
- 发布说明 (Release Note): 填写“发布说明”后才能发布应用 (Fill in the "Release Note" before you can release the application)

At the bottom right, there are three buttons: '上一步' (Previous Step), '完成' (Complete), and '发布' (Publish). The '发布' button is highlighted in blue.

6. Click **Application Overview** to see the newly created application, thereby obtaining the corresponding **App Key**, **App Secret**, and **Product ID**. Ensure that the App Key and App Secret are not leaked, and contact

Tencent Cloud IoT development engineers offline.



The screenshot shows a web form for creating a new IoT application. The form includes the following fields and options:

- 创建者** (Creator): A blurred text field.
- 应用名称** (Application Name): A text field containing "ffhhh".
- 应用描述** (Application Description): A blurred text field.
- 屏幕类型** (Screen Type): Two radio buttons. "有屏设备" (Device with screen) is unselected, and "无屏设备" (Device without screen) is selected.
- 应用类型** (Application Type): A dropdown menu showing "无屏音箱" (Device without screen - speaker).
- APP KEY**: A blurred text field.
- AccessToken**: A blurred text field.
- Product ID**: A blurred text field.
- 应用图标** (Application Icon): A dashed box with a blue "+" icon. To its right, text specifies: "支持PNG/JPG格式文件" (Supports PNG/JPG format files), "图片大小不超过 100KB" (Image size not exceeding 100KB), and "建议固定 512*512像素" (Suggest fixed 512*512 pixels).
- 删除该应用** (Delete this application): A button at the bottom left.

Step 3: Integration of the Master Control Device

1. Develop based on the fusion version SDK of Tencent Cloud IoT and Cloud Xiaowei. For the SDK, please contact Tencent Cloud colleagues or agents offline to obtain.
2. Perform device-side integration development based on Tencent Cloud triplet information (device name, device key, and product ID), Cloud Xiaowei tvs_pid, and DSN.

Note:

- Triplet information retrieval. For more information, see [equipment information](#).
- DSN consists of "product ID_device name".
- The controlled device supports Cloud Xiaowei skills. You need to apply in advance for enabling [Cloud Xiaowei Voice Skill Service](#). Once the service is activated, you need to contact Tencent Cloud IoT development engineers offline.

Step Four: Mass Production of Devices

Log in to the [IoT Explorer Console](#). On products where "language assistant" has been activated, submit a mass production application. After review passed, mass production can be started. For more details, see [batch production](#).

Mini Program Operation Guide

1. Perform master control device network configuration binding.

Use the Tencent Lianlian Mini Program to scan and bind the master control device network configuration QR code to perform network configuration binding.

2. Activate the master control device.



3. Associate the master control device with the controlled device.

After the master control device is activated, it can be associated with the controlled device. Upon

successful association, you can proceed with voice control.



Note:

Before adding a controlled device, you need to rename the controlled device before you can control the smart device.

4. Authorize the use of QQ Music.

If a developer integrates music skills in [step two](#) and clicks "QQ Music" on the master control device panel in the mini program and authorizes it, then music skills can be used.

Note:

- The standard control panel for mini program music on demand is under development and will be published subsequently; developers can also choose [custom H5 panel development](#).
- Music skill access guide: On the Cloud Xiaowei Skill Platform, select QQ Music skill in the version management section. For details, see [Music Service](#).

Custom H5 Development

Developers can custom develop the control panel of the master control device or the controlled device. For details, see [custom H5 development](#).

Kugou Music Service

Last updated: 2025-04-27 17:44:50

this document introduces how to open and use Tencent Cloud IoT Explorer KuGou Music value-added service.

Prerequisites

Activating devices to enable KuGou Music service is only supported in the following regions:
Chinese mainland (excluding Hong Kong, Macao, and Taiwan region)

Apply for Enabling KuGou Music Service

KuGou Music service is a paid value-added service and does not offer a free trial. You can [submit a ticket](#) for business consultation. The staff will contact you for service purchase matters.

Operation Steps

Step 1: Enable KuGou Music Service

1. Log in to the [IoT Development Platform Console](#), select the region "China", and create a project and a product. For details, see [Product Definition](#).
2. Click **Product Development** on the left sidebar to enter the product list page.
3. Select a created product to enter the product detail page, click **Thing Model Definition**, and click the button at "KuGou Music Service" in **Advanced Function** to activate successfully.
4. After the successful activation of the KuGou Music service, the Thing Model of this product will automatically add system properties related to the KuGou Music service.

| 功能类型 | 功能名称 | 标识符 | 数据类型 | 读写类型 | 数据定义 | 操作 |
|------|---------------------|---------------------------|------|------|--|-------|
| 属性 | tvS授权下行命令 必选 | _sys_kg_tvs_auth_cmd | 字符串 | 读写 | 字符串长度: 0 - 2048个字符 | 编辑 删除 |
| 属性 | tvS授权上行回复 必选 | _sys_kg_tvs_auth_reply | 字符串 | 读写 | 字符串长度: 0 - 2048个字符 | 编辑 删除 |
| 属性 | 开关 必选 | _sys_kg_powers_witch | 布尔型 | 读写 | 0 - 关 1 - 开 | 编辑 删除 |
| 属性 | 播放暂停 必选 | _sys_kg_pause_play | 布尔型 | 读写 | 0 - 暂停 1 - 播放 | 编辑 删除 |
| 属性 | 当前播放列表 必选 | _sys_kg_cur_play_list | 字符串 | 读写 | 字符串长度: 0 - 2048个字符 | 编辑 删除 |
| 属性 | 前一首后一首 必选 | _sys_kg_pre_next | 枚举型 | 读写 | 0 - 不变 1 - 上一首 2 - 下一首 | 编辑 删除 |
| 属性 | 播放模式 必选 | _sys_kg_play_mode | 枚举型 | 读写 | 0 - 顺序播放 1 - 单曲循环 2 - 随机播放 | 编辑 删除 |
| 属性 | 播放的歌曲列表 必选 | _sys_kg_song_list | 字符串 | 读写 | 字符串长度: 0 - 2048个字符 | 编辑 删除 |
| 属性 | 音量 必选 | _sys_kg_volume | 整型 | 读写 | 数值范围: 0-100 初始值: 0 步长: 1 单位: | 编辑 删除 |
| 属性 | 播放进度 必选 | _sys_kg_play_position | 整型 | 读写 | 数值范围: 0-7200 初始值: 0 步长: 1 单位: | 编辑 删除 |
| 属性 | 当前曲目 必选 | _sys_kg_cur_song_id | 字符串 | 读写 | 字符串长度: 0 - 2048个字符 | 编辑 删除 |
| 属性 | 下发标志 必选 | _sys_kg_control_seq | 整型 | 读写 | 数值范围: 0-100000 初始值: 0 步长: 1 单位: | 编辑 删除 |
| 属性 | 推荐音质 必选 | _sys_kg_recommend_quality | 枚举型 | 读写 | 0 - 标准 1 - 高清 2 - 无损 | 编辑 删除 |

产品品类 智慧生活-影音办公-智能音箱

设备类型 设备

认证方式 密钥认证

通信方式 Wi-Fi

数据协议 数据模板

创建时间 2021-06-16 19:46:30

更新时间 2021-06-16 19:46:30

产品描述 -

功能定义

标准功能 7个

自定义功能 0个

增值服务

人脸识别 ☐

实时音视频 ☐

语音识别 ☐

语音助手 ☐

酷狗音乐服务 ☒

动态注册配置 ①

动态注册 ☐

Step Two: Create a Device

Click **Device Debugging** > **Create New Device**, fill in relevant information and click **Save**.

新建设备

×

所属产品

设备名称 *

支持英文、数字、下划线的组合，最多不超过48个字符

保存

取消

Note:

- When there are devices under the product, the status of the value-added service switch cannot be changed.
- If you delete a device with activated value-added services, the purchased License quantity cannot be recovered.

Step Three: Device-Side Integration Guides

Device-side integration is divided into [Android SDK Integration](#) for screen-equipped devices and [C-SDK Integration](#) for low-power IoT devices according to different system resources and security requirements of different IoT terminals.

Android SDK Integration Guide

Android SDK integration for IoT+ music content service for Android terminal equipment such as vehicle systems, intelligent screens, and TVs. After the device ports the SDK and binds to Tencent Lianlian Mini Program, it can authorize the device to access KuGou Music in Tencent Lianlian Mini Program; through KuGou Music Mini Program, the device can be controlled to play audio resources of KuGou Music.

Seamless Integration with SDKs

Depend on the local sdk source code to build and modify the [build.gradle](#) of the application module, so that the application module depends on the [explorer-device-tme](#) source code. An example is as follows:

```
dependencies {  
    implementation project(':explorer:explorer-device-tme')  
}
```

! Note:

If you need to integrate this SDK, please contact us offline. The Demo sample project uses the SDK built from the source code that depends on the local explorer-device-tme.

Points to Note When Integrating the SDK

- Add the following configuration in the build.gradle file:

Currently, the sdk provides only two kinds of so files: armeabi-v7a and x86. Add abiFilters in case the so library cannot load in some cases.

```
android {  
    ndk {  
        abiFilters 'armeabi-v7a', 'x86'  
    }  
    if (findProject(':explorer:explorer-device-tme') != null) {  
        api project(':explorer:explorer-device-tme')  
    }  
  
    implementation 'io.reactivex.rxjava2:rxjava:2.2.10'  
    implementation 'io.reactivex.rxjava2:rxandroid:2.1.1'  
    implementation 'com.squareup.retrofit2:retrofit:2.6.0'  
    implementation 'com.squareup.retrofit2:adapter-rxjava2:2.6.0'  
    def room_version = "2.2.5"  
    implementation "androidx.room:room-runtime:$room_version"  
    annotationProcessor "androidx.room:room-compiler:$room_version"  
    implementation "androidx.room:room-rxjava2:$room_version"  
}
```


- For systems above Android 6.0, the following permissions are required to be dynamically requested. Please ensure they are authorized.

```
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

- Call `UltimateTv#onApplicationCreate()` in `Application#onCreate`.

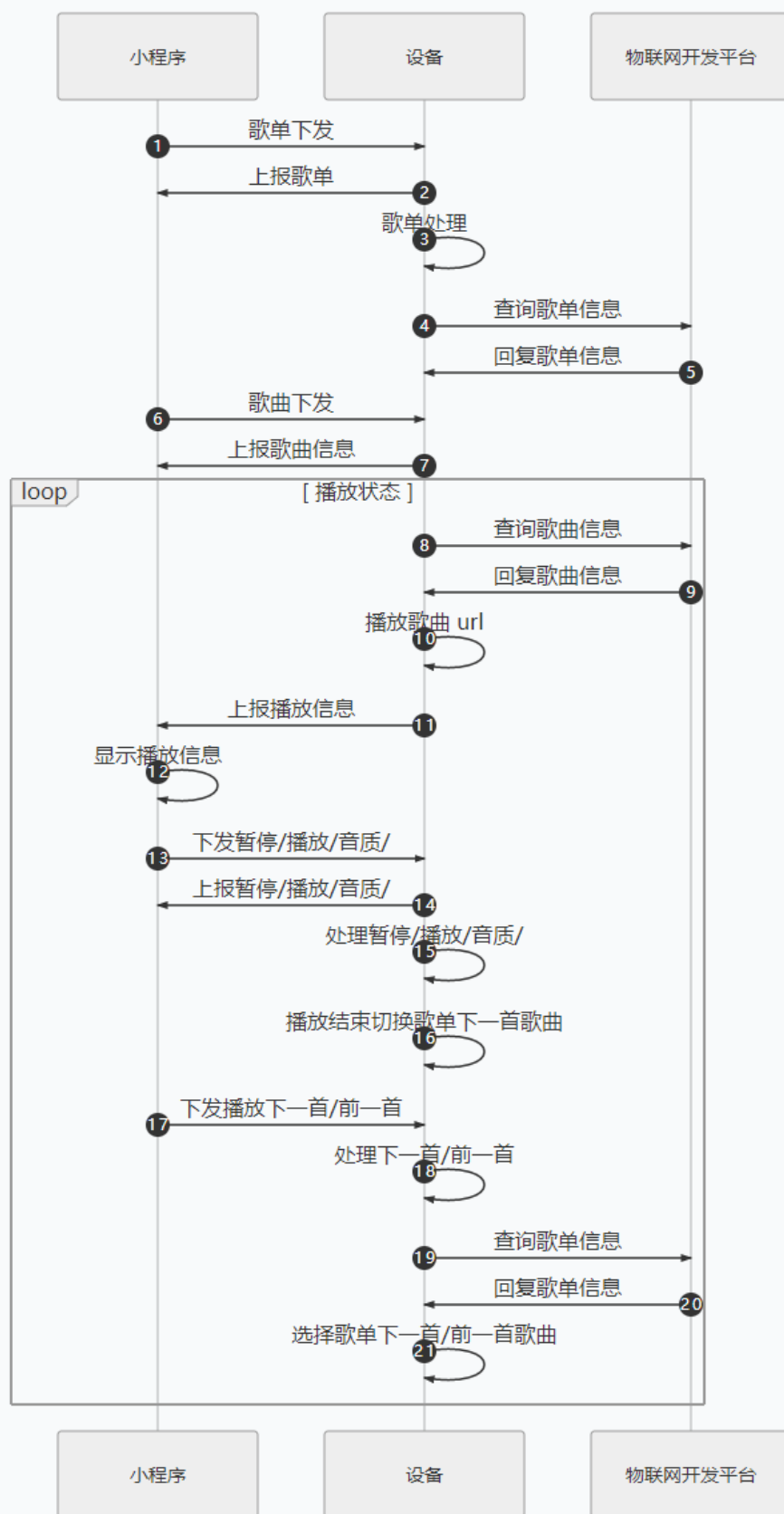
```
public class MyApplication extends Application {
    @Override
    public void onCreate() {
        super.onCreate();
        // Call the onApplicationCreate method of the sdk
        UltimateTv.getInstance().onAppcationCreate(this);
        ...
    }
    ...
}
```

C-SDK Integration Guide

Overview

C-SDK integration for IoT+ KuGou Music content service for Linux or embedded operating system terminal equipment such as wearables, small household appliances, and children's educational products. After the device ports the SDK and binds to Tencent Lianlian Mini Program, it can authorize the device to access KuGou Music in Tencent Lianlian Mini Program; through KuGou Music Mini Program, the device can be controlled to play audio resources of KuGou Music.

Interaction Process



Data Template

- The mini program sends a song list to the data template, with the IDs of the songs to be played.
- Control play/pause/next/previous.

- Set playback progress/sound quality, etc.
- The device reports playback information through the data template, such as the current song ID/sound quality/playback progress, etc.

topic

```
$thing/up/property/{product_id}/{device_name}
$thing/down/property/{product_id}/{device_name}
```

Song Inquiry and Playlist Inquiry

topic

```
$thing/up/service/{product_id}/{device_name}
$thing/down/service/{product_id}/{device_name}
```

Data Format

```
song information query
{
  "method": "kugou_query_song",
  "clientToken": "a string that is not duplicate earlier"
  "params": {
    "song_id": "xxxxx",
  }
}

Song list query
{
  "method": "kugou_user_command",
  "clientToken": "a string that is not duplicate earlier"
  "timestamp": 1594108563000
  "params": {"album_id/playlist_id/top_id": "Song list ID", "page": Which page of
songs in the song list is requested, starting from 1, "size": Number of songs per page,
"kugou_command": "Song list query command"}
}

{
  "method": "kugou_user_command_reply",
  "clientToken": "20a4ccfd-d308-11e9-86c6-5254008a4f10",
  "code": 0,
  "status": "",
  "data": {
    "Song list information"
```

```
How many songs are there in total in the song list?  
"songs": [{"song_id information"}, {"song_id information"}, {"song_id information"}]  
}  
}
```

Step 4: Develop and Integrate on the Application Side: a Guide

Integration of Tencent Lianlian and KuGou Music Based on Development-Free Panel

- Select the **standard category** when creating a product in the [console](#), and specify the **Smart Life > audio-visual office > background music** category.

新建产品

产品名称 *

支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

产品品类

标准品类自定义品类

请选择产品品类

设备类型

设备网关子设备

通信方式 *

请选择通信方式

请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

认证方式

密钥认证证书认证

数据协议

物模型自定义透传

?

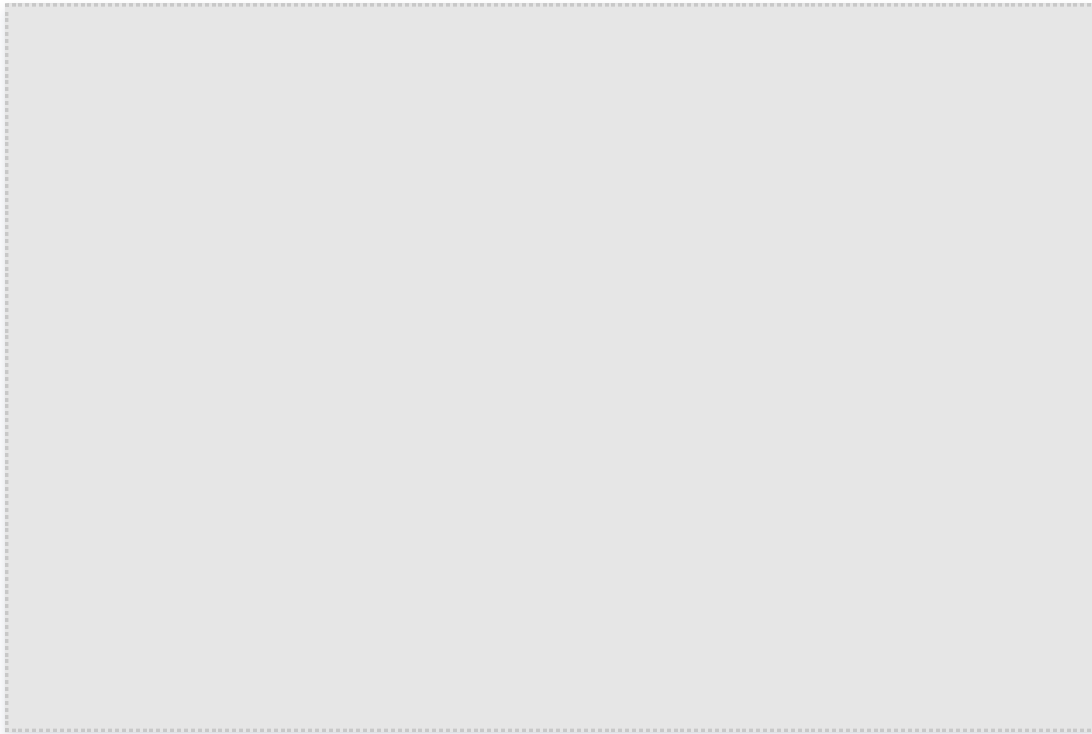
描述

选填

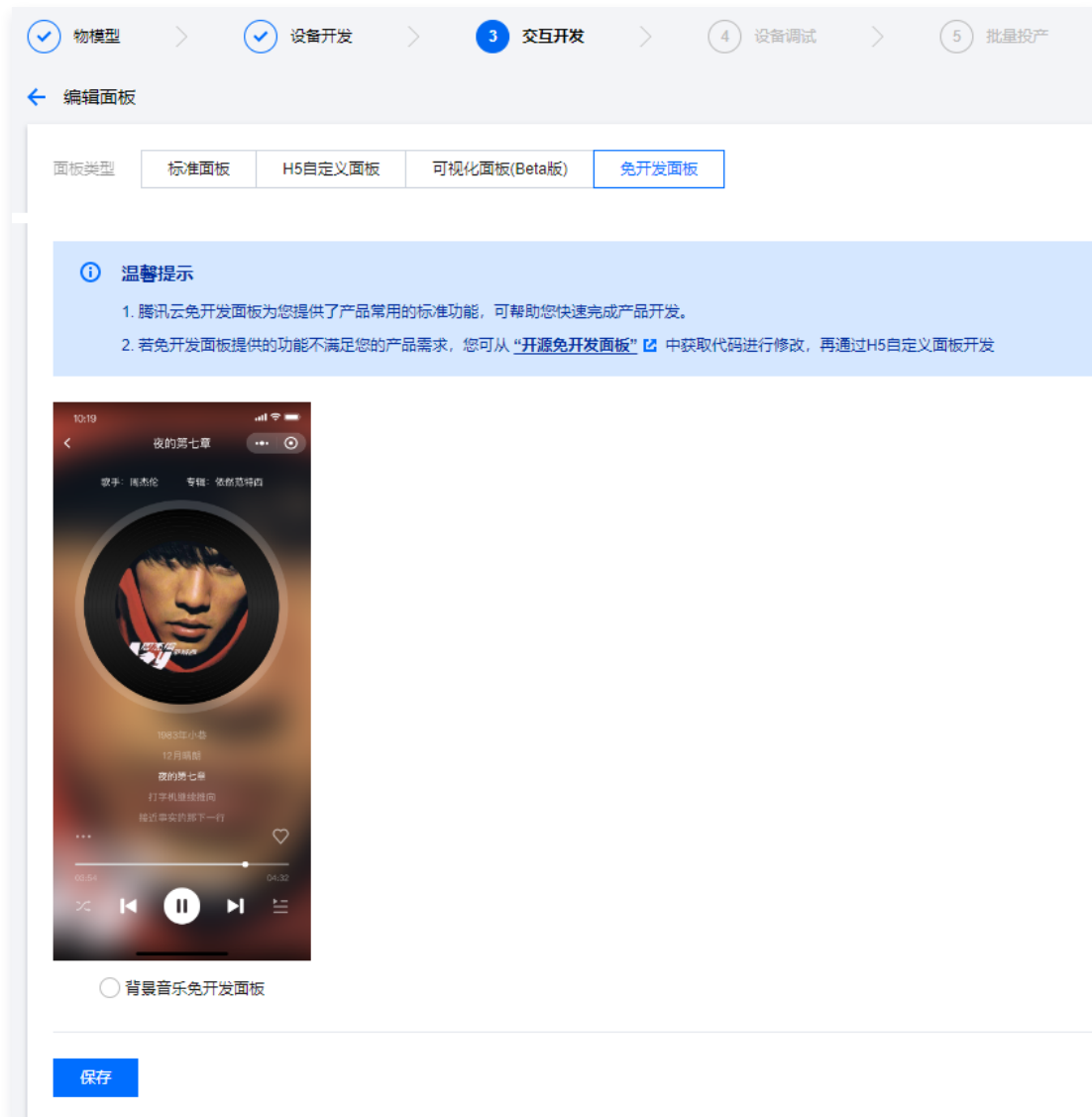
最多不超过80个字符

确定

取消



- After enabling the KuGou Music Service, select the **dashboard type** as "development-free dashboard" in **interactive development** to use the commonly used standard features of the product and quickly complete product development.



- After selecting the development-free mini program dashboard, you can obtain massive music resources for playback control through the device.



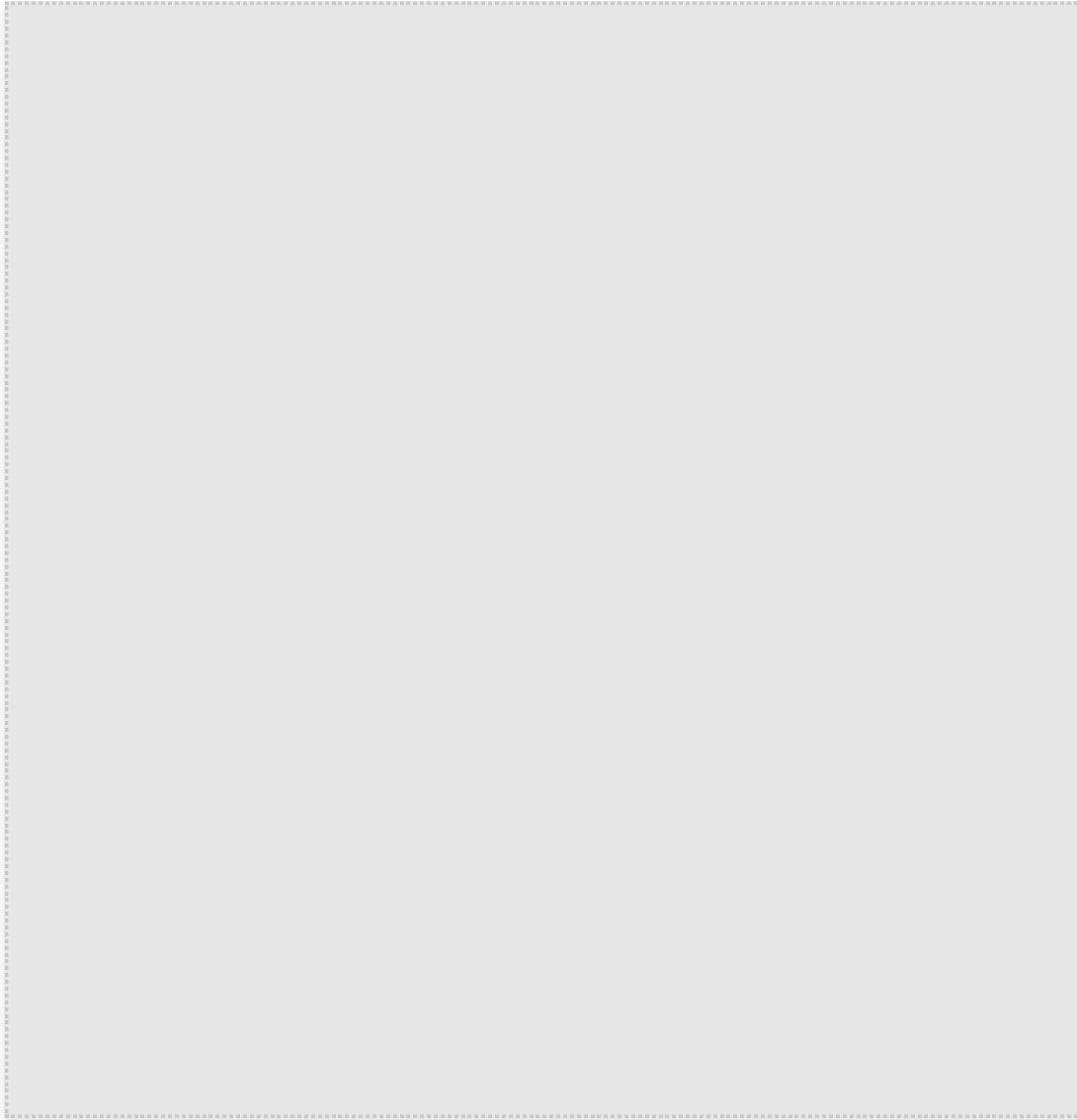
Integration Based on Music Service SDK

For this value-added service, a music service SDK is provided for custom H5 development and Chinese domestic brand mini-program development. See [Music Service SDK](#) for details.

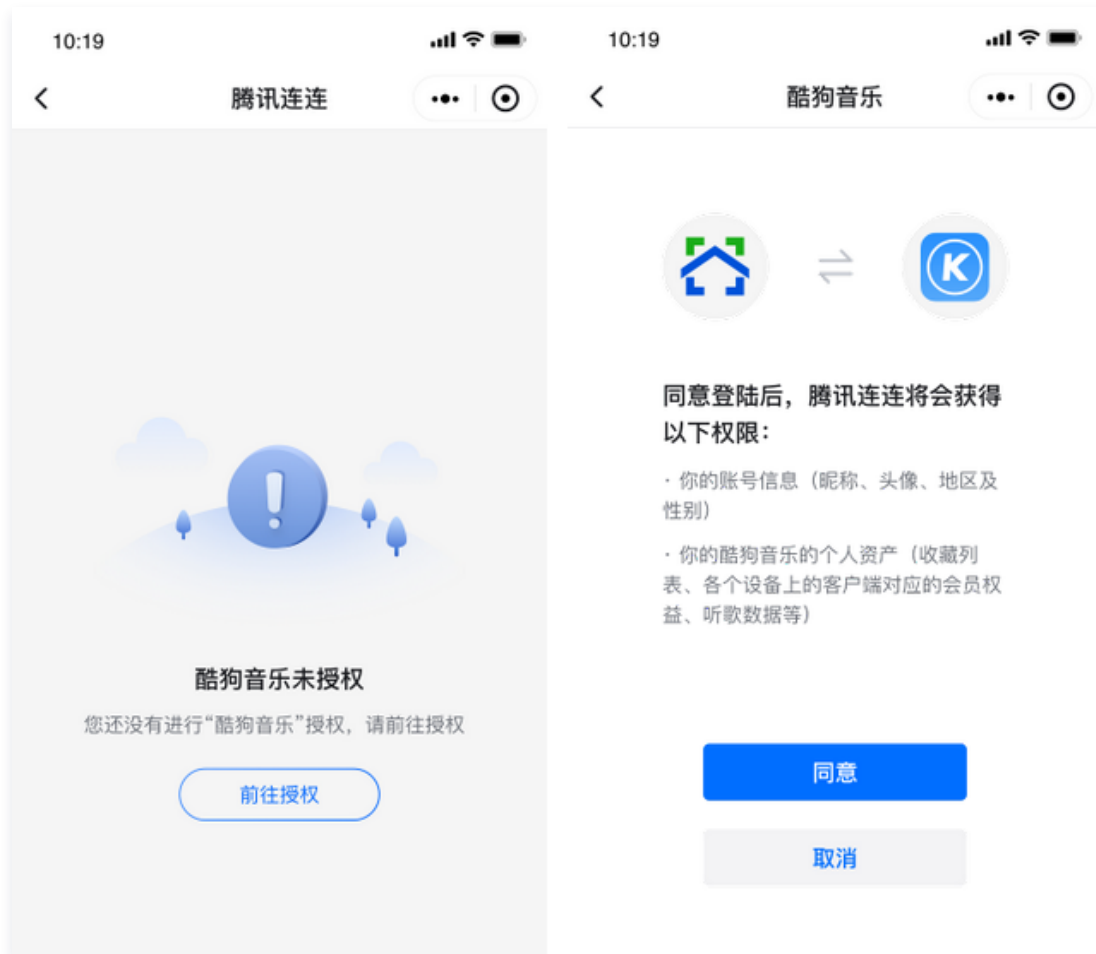
Lianlian Mini Program User Operation Guide

1. Device binding: Use the Tencent Lianlian Mini Program to scan the Device QR Code to bind the device.
(Select different binding methods according to the communication type of the device. For Wi-Fi devices,

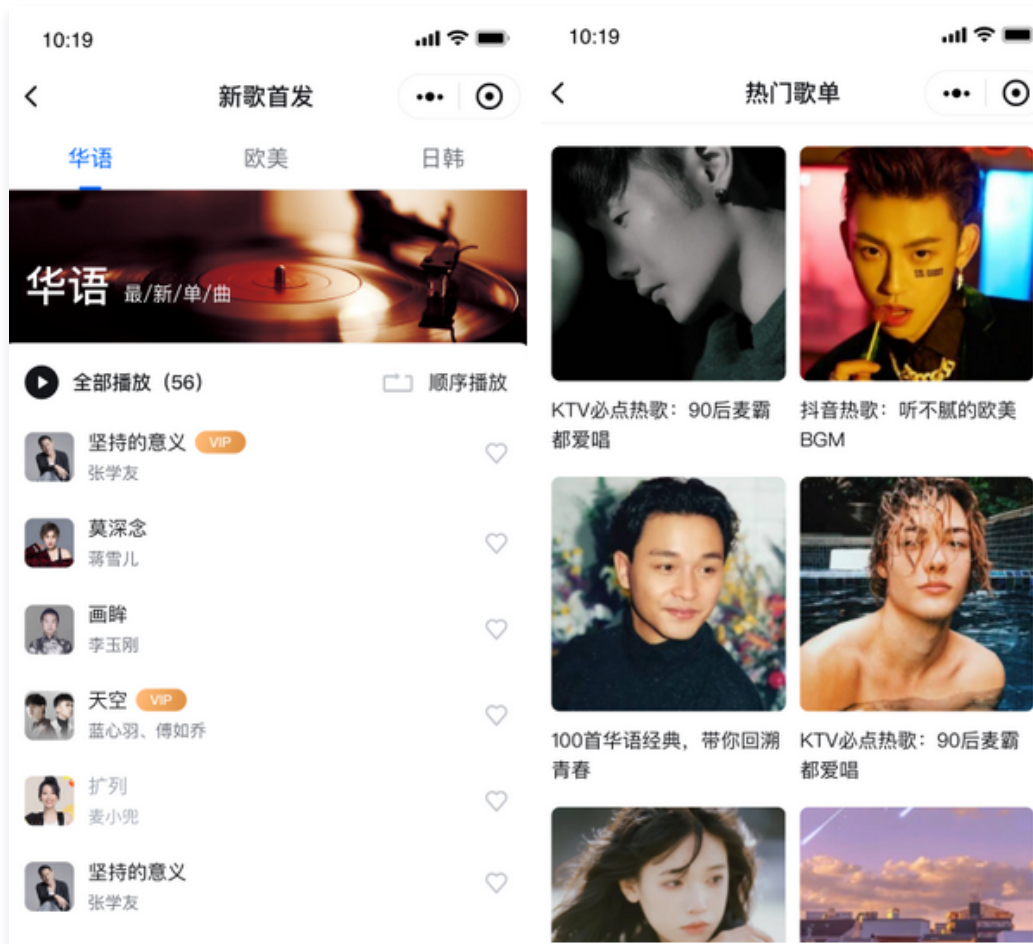
perform Network Configuration Binding. For cellular devices, perform scanning and binding.)



2. Bind the device through the Lianlian Mini Program. After selecting the device, enter the main interface portal to authorize the KuGou Music service.



3. After authorizing the KuGou Music service, you can obtain the playback control of mass music resources.



SDK Integration Guide

- Use the application-side SDK and access the official website for the latest version of [H5 custom development SDK](#).
- The device-side SDK is hosted on Github. You can access Github to download the latest version of [Android SDK integration guide](#) and [device-side C-SDK integration guide](#).

Control Authorization

Last updated: 2025-04-27 17:45:05

This document shows you how to open and use the product capabilities of the gateway central control in Tencent Cloud IoT Explorer.

Prerequisites

Only supported in the following product categories:

Smart Life – Electrical Lighting – Intelligent Control Screen, Smart Life – Gateway Central Control – Central Control Panel.

Apply for Enabling the Capacity of the Central Control Gateway Product

The central control gateway is a paid value-added feature and does not offer a free trial. You can [ask for online consultation](#) for help. The staff will contact you regarding the service purchase matters.

Activating Central Gateway Service

1. Log in to the [IoT Explorer console](#), select the region "China" and create a product. For details, see [Product Definition](#).
2. Click **Product Development** on the left sidebar to enter the product list page.
3. Select a created product to enter the product detail page, click **Thing Model Definition**, and click the button at "Control Authorization" in **Advanced Function** to activate successfully.

1 物模型定义 > 2 设备开发 > 3 交互开发 > 4 设备调试 > 5 批量投产

导入物模型 查看物模型JSON 物模

标准功能 (13) 自定义功能 (0) 高级功能 (0)

| 功能名称 | 收费方式 | 功能描述 | 启用 | 操作 |
|-------------|-------|---|--------------------------|------------------------|
| 实时音视频 | 按设备收费 | 专为物联网场景打造的音视频通话解决方案，支持智能终端与小程序、App 一对一音视频通话，支持一键呼叫、分組呼叫，提供应用端、设备端 SDK。 | <input type="checkbox"/> | 查看文档 |
| 语音识别 | 按设备收费 | 腾讯连连语音识别服务是针对录音笔、翻译笔、会议办公等物联网设备场景提供物联网+AI语音识别的服务，包括文件识别、一句话识别、实时语音识别等各种物联网场景AI产品服务。 | <input type="checkbox"/> | 查看文档 |
| 语音助手 | 按设备收费 | 腾讯连连物联网平台打通了业内主流的三方语音技能平台，提供快速接入三方平台并支持通过语音控制腾讯连连生态智能设备的能力。目前已经打通的语音技能平台包括Amazon Alexa、Google Assistant、百度小度和云小微。 | <input type="checkbox"/> | 查看文档 |
| 酷狗音乐服务 | 按设备收费 | 腾讯连连整合酷狗音乐内容资源，面向消费物联网和产业互联网用音乐赋能智能硬件。 | <input type="checkbox"/> | 查看文档 |
| IoT Video服务 | 按设备收费 | 为客户提供视频连接、存储和智能应用服务，安全高效。客户可简单快速实现设备接入、宽带传输、云端存储、远程观看等一站式视频场景能力，并提供丰富的 AI 算法模型实现具体场景的智能解析及应用，实现云边协同智能应用。 | <input type="checkbox"/> | 产品补充文档 |
| 主控授权 | 按设备收费 | 产品补充描述 | <input type="checkbox"/> | 产品补充文档 |

Creating a Device

Click **Product Development** in the left sidebar of the console, select a product to enter the product detail page, click **Device Debugging** > **Create new device**, fill in relevant information and click **Save** just.

新建设备

×

所属产品

设备名称 *

支持英文、数字、下划线的组合，最多不超过48个字符

保存取消

Note:

- When there are devices under the product, the switch status of the advanced function cannot be changed.
- If you delete a device with activated value-added services, the purchased License quantity cannot be recovered.

Video AI Analysis

Last updated: 2025-04-27 17:45:20

Prerequisite: Device Connectivity to Event Cloud Storage

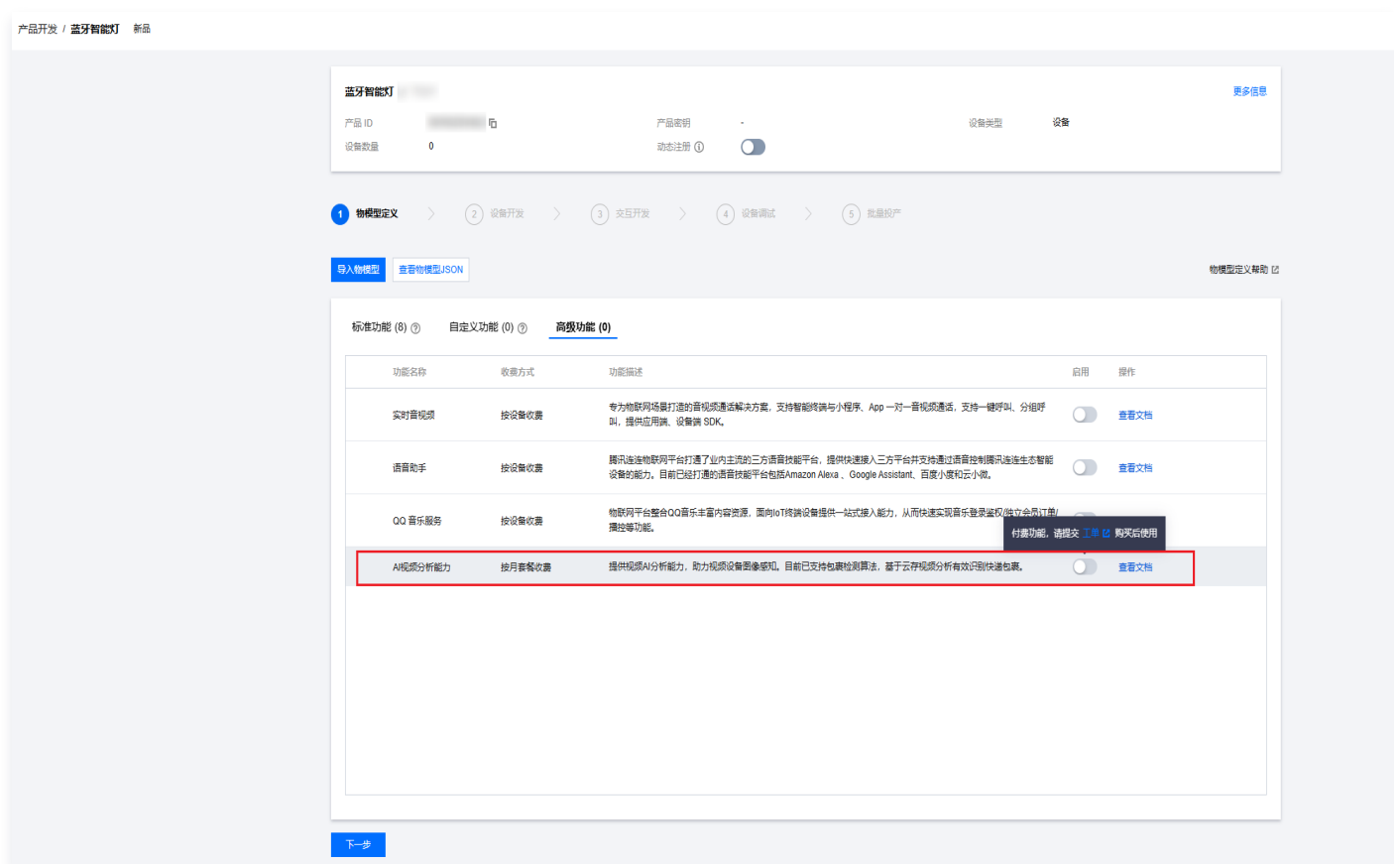
The IoT intelligent video service (Consumer Edition) platform's AI analytical capability requires devices to upload cloud-stored event recordings. The access process is as follows:

- Enable the Event Cloud Storage Package for equipment through TencentCloud API. See [Enable Cloud Storage Package](#) for details.
- Enable the cloud storage module on the device, report cloud storage events, and push audio and video data to the cloud. See [Cloud Storage Module](#) in the device connectivity manual for details.

Enable Video AI Analytical Capability for Devices

Applying for Video AI Analysis Value-Added Service Capability

1. Open [console](#), enter **product development** page, click on the product name of the feature to be enabled, and enter the product development Thing Model definition page.
2. Switch to the advanced function of the product **Advanced Function**, and submit a request to activate the advanced function of video AI analysis of the product.



Enable Device Cloud Storage AI Analytical Capability

You can call TencentCloud API to modify Device Cloud Storage AI Analysis Service to enable video AI analytical capability for a specified device. After a device with video AI analytical capability enabled reports a Cloud Storage Event, the platform will create a video AI analysis task for the uploaded video and return analysis results.

The IoT intelligent video service (Consumer Edition) platform currently supports the following video AI analytical capabilities:

| Video AI Analytical Capability | ServiceType Parameter Value |
|--------------------------------|-----------------------------|
| Package detection | PackageDetect |

When enabling video AI analysis capability for a device, you need to input the corresponding `ServiceType` parameter. If you expect to enable multiple video AI analysis capabilities for one device, you need to call the [Modify Device Cloud Storage AI Analysis Service](#) TencentCloud API separately for each capability and input the corresponding `ServiceType` parameter.

Examples of wrap detection input sources and recognition results:



Get Video AI Analysis Result

You can query video AI analysis results through TencentCloud API, or receive video AI analysis result callbacks through HTTP API.

Querying Analysis Results Via TencentCloud API

You can call TencentCloud API [to query the list of AI analysis tasks in Device Cloud Storage](#) to query the video AI analysis results of a specified device.

Meaning of Field

- **TaskId:** The field value of TaskId is the unique identifier of the video AI analysis task.
- **Status:** The Status field indicates the task running status of the video AI analysis task as well as coarse result information.

| Field Value | Meaning |
|-------------|--|
| 1 | Analysis failed |
| 2 | Analysis succeeded, but no targets were identified. |
| 3 | Analysis succeeded and at least one target was identified. |
| 4 | Analytical task is in progress |

- **Result:** The Result field provides detailed information on the video AI analysis results. The output format may vary for different video AI analytical capabilities.

| Video AI Analytical Capability | Result Format | Example Output |
|--------------------------------|---|-----------------------------------|
| Package detection | <p>Possible values:</p> <ul style="list-style-type: none">• plastic-bags• file-bags• paper-bags <p>When detecting multiple different types of packages, use space as a delimiter to concatenate them into a string.</p> | <pre>plastic-bags file-bags</pre> |

Receiving Analysis Result Callbacks Via HTTP Api

You can set the HTTP callback configuration for video AI analysis of the specified product on the console. After completing the HTTP callback configuration, the platform will forward the video AI analysis results of the devices under the product to the specified HTTP callback URL.

消息接收设置

 您可以在此处设置AI视频分析结果推送目的端，支持推送至第三方服务器，[点击查看文档](#)了解如何开发HTTP服务接受物联网平台数据

数据转发到第三方服务 (Forward)

API地址: *

☒ 使用已有HTTP服务地址



☐ 增加鉴权token

保存

取消

You need to deploy an http service that complies with the definition of the callback API and make it accessible from the public network to complete the HTTP callback configuration and receive the result callback of video AI analysis. The callback API is defined as follows:

- GET /** Connectivity check: You need to fetch the value of the HTTP request header `Echostr`, use it as the content of the HTTP response, and return a 200 OK status code.
- POST /** Analysis result callback: You can receive the video AI analysis results pushed by the platform from this API. The video AI analysis results are in JSON format and are located in the HTTP request body. After successful receipt of this request, a 200 OK status code should be returned.

Note:

When configuring the HTTP callback URL in the console, the platform will initiate a connectivity check request to the specified URL. If the connectivity check fails, the HTTP callback configuration will not be saved to take effect.

Example: Connectivity Check – Request

When you configure the HTTP callback URL in the console, the platform initiates a connectivity check request to the URL.

```
GET / HTTP/1.1
User-Agent: IotVideo/1.0
Echostr: 3A4dsItMWYmL7pkL
```

Example: Connectivity Check – Response

When the http service you deployed receives a connectivity check request, it should return a response with a status code of 200 OK, and the response content should be the value of the HTTP request header `Echostr`.

```
200 OK
```



```
Content-Type: text/plain
Content-Length: 16

3A4dsItMWYmL7pkL
```

Example: Analysis Result Callback – Request

Upon completion of the AI analysis task for video, the platform will carry the analysis result to request the specified HTTP callback URL. The analysis result is in JSON format and is located in the HTTP request body.

```
POST / HTTP/1.1
User-Agent: IotVideo/1.0
Content-Type: application/json

{
  "MsgType": "ai_analytics_result",
  "ProductId": "MVTYMD8YCD",
  "DeviceName": "dev001",
  "ServiceType": "PackageDetect",
  "TaskId": "95990598-aba1-465e-9993-3f90c071e5d9",
  "Status": 3,
  "Result": "paper-bags file-bags plastic-bags",
  "Time": "2024-04-10 18:12:05"
}
```

Example: Analysis Result Callback – Response

When the http service you deployed receives an analysis result callback request, it should return a response with a status code of 200 OK to indicate successful reception. If the request fails, times out without receiving a response, or the returned HTTP status code is not 200, the platform will retry the request after a delay of 5 seconds, up to a maximum of 2 retries.

```
200 OK
Content-Length: 0
```

Additionally, you can configure an authentication Token to verify that requests come from the IoT Video (Consumer Edition) platform. When you configure the authentication Token, the platform's request will include the following HTTP request headers. Among them, `Nonce` and `Timestamp` are used to prevent replay attacks, and `Signature` can be used for confirmation that the request comes from the platform.

| Request Header | Meaning | Example Value |
|----------------|-----------------------------|--|
| Nonce | Random string | vLeD3GvsUx5O1kLP |
| Timestamp | Second-level UNIX timestamp | 1712744577 |
| Signature | Signature value | 2194e42c5e15e34ef9463de4c3f27619edf748d7 |

You can calculate the signature using the same algorithm based on the configured authentication Token and the `Nonce` and `Timestamp` values in the HTTP request header. If the signature value matches, the request can be considered as coming from the platform. The calculation method for the signature value `Signature` is:

1. Sort the three strings `Nonce`, `Timestamp`, and authentication Token lexicographically, and then concatenate them sequentially to get a string (in the above example, assuming the authentication Token is `example`, the concatenated string is `1712744577examplevLeD3GvsUx501kLP`).
2. Calculate the SHA1 hash value of the concatenated string.
3. Use the lowercase hexadecimal representation of the hash value as the value of `Signature` .

Example: Connectivity Check (Including Authentication Information) – Request

This example assumes that the authentication Token is `example` .

```
GET / HTTP/1.1
User-Agent: IotVideo/1.0
Echostr: 3A4dsItMWYmL7pkL
Nonce: vLeD3GvsUx501kLP
Signature: 2194e42c5e15e34ef9463de4c3f27619edf748d7
Timestamp: 1712744577
```

WeChat Strong Reminder Notification

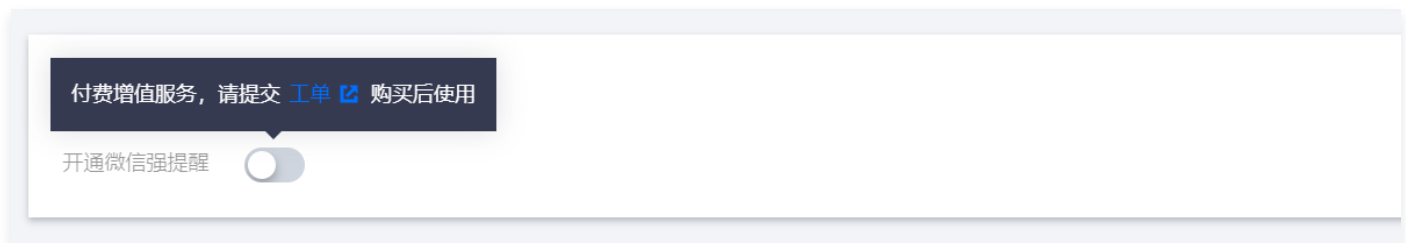
Last updated: 2025-04-27 17:45:36

Application Scenarios

WeChat strong reminder notifications are mainly used for device alarms, notifying users through the WeChat strong reminder effect with a high trigger rate to ensure no alarms are missed.

Operation Steps

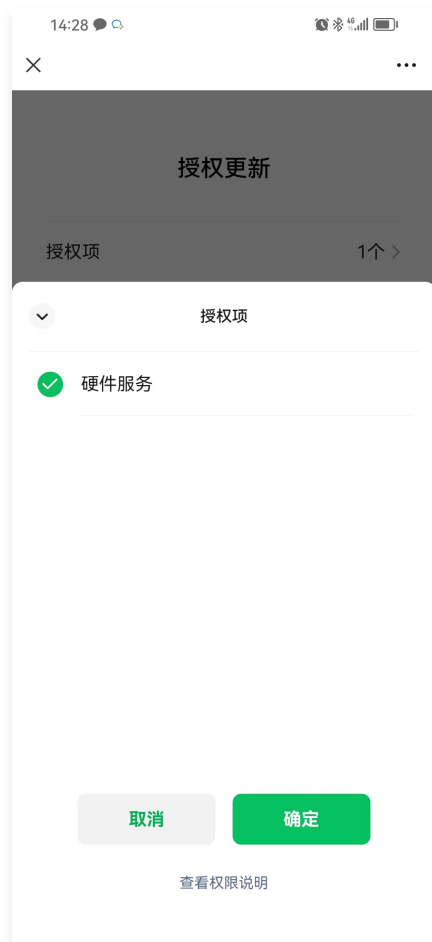
1. Log in to the [WeChat public platform](#), and apply for a mini program, device type (model id), device message capacity, and message template.
2. Log in to the [IoT Explorer console](#), select the corresponding instance, and enter the **Application Development** page.
3. Create an application. Click on the application name to enter the application details, and **apply** to enable **WeChat strong reminder**.



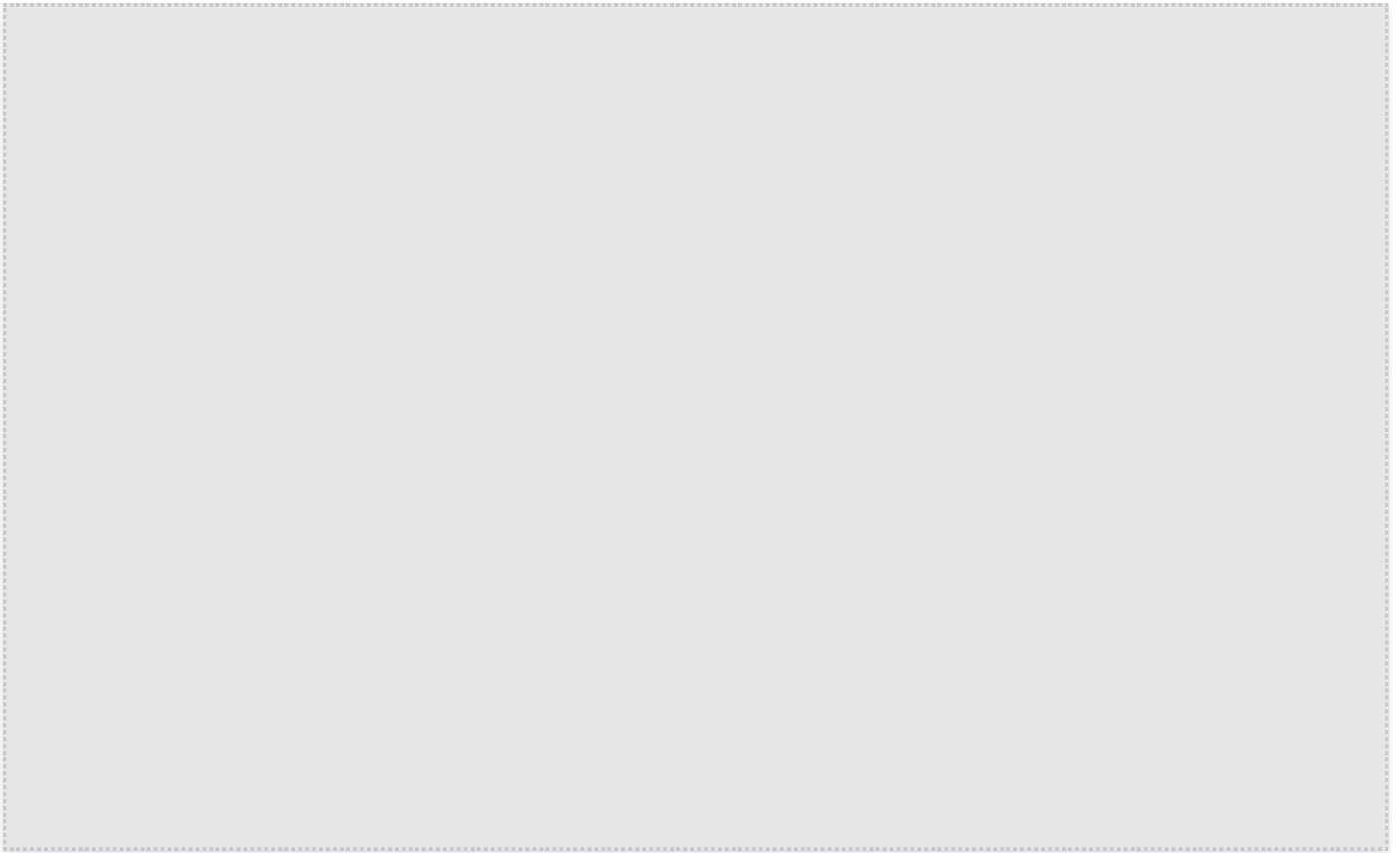
4. After applying for enabling, click **Authorize Mini Program**. Please ask your WeChat mini program administrator to scan the code for authorization.



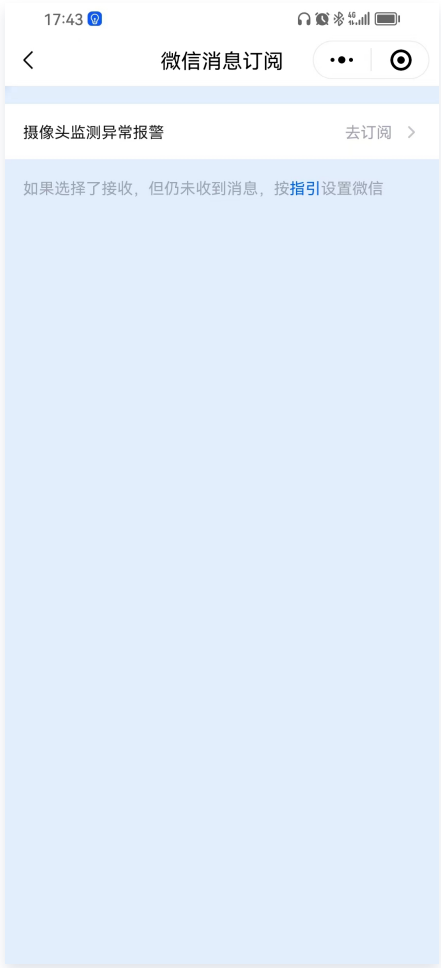
5. Only need to select **Hardware Service** permission and authorize.



6. Click **Add New** template, and configure the conditions for triggering strong reminders and the alarm template according to the console configuration.



7. When users use the mini program to bind a device and subscribe to the strong reminder template of this device, the device reports data. After the trigger conditions are met, WeChat will push a strong reminder to notify users.



8. The effect of WeChat strong reminder when the mobile phone is in sleep mode.



TWeCall

Last updated: 2025-04-27 17:45:53

IoT Explorer provides users with the TWeCall access capability based on the integration plugin of WeChat mini program and the integration SDK of the device side. This document introduces relevant content about TWeCall operations.

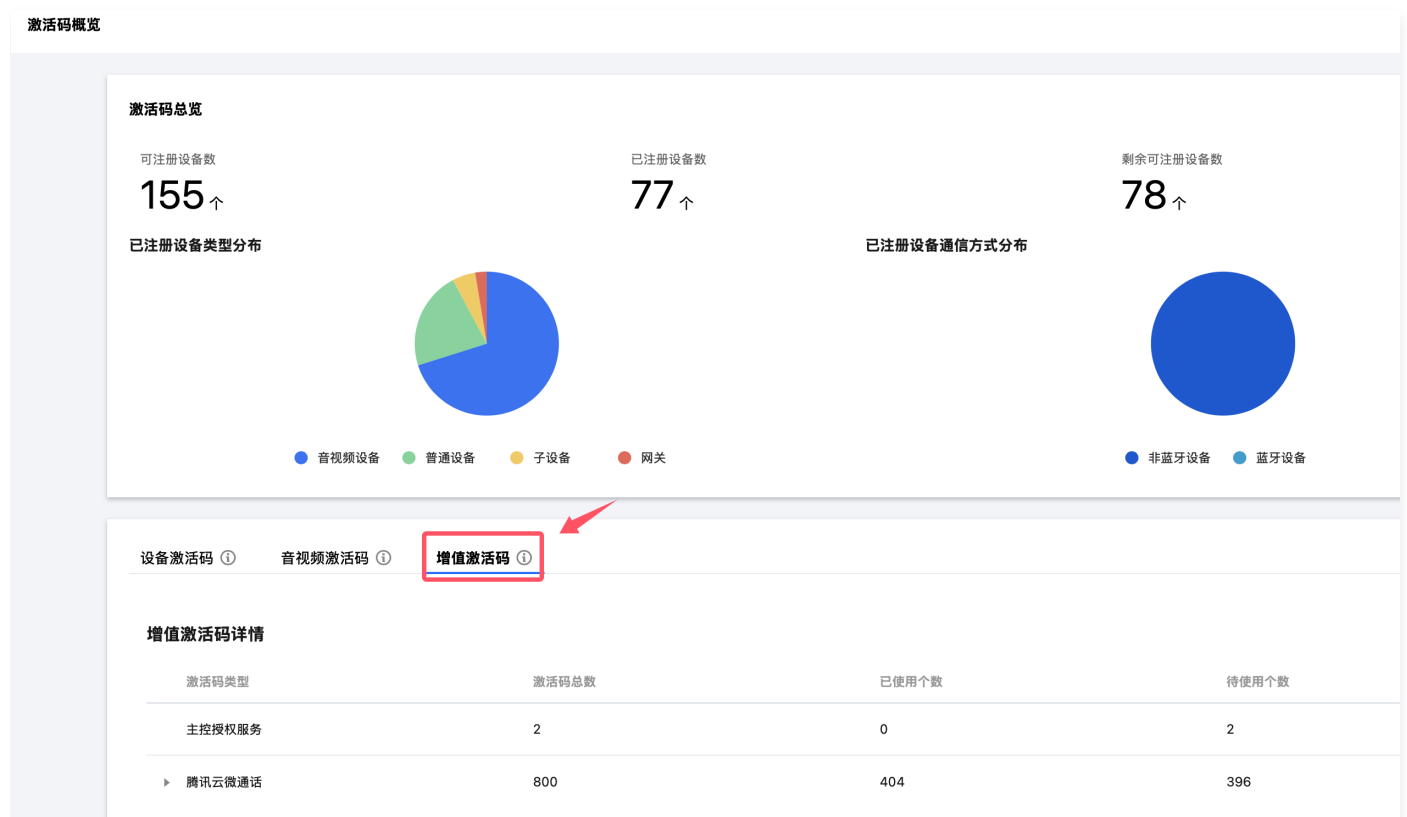
Feature Overview

The device and mini program can make two-way calls, achieving the native experience of WeChat audio and video calls; the device makes a one-click call, and the WeChat keeps ringing as a reminder.

Operation Steps

View the Activation Code Limit of TWeCall

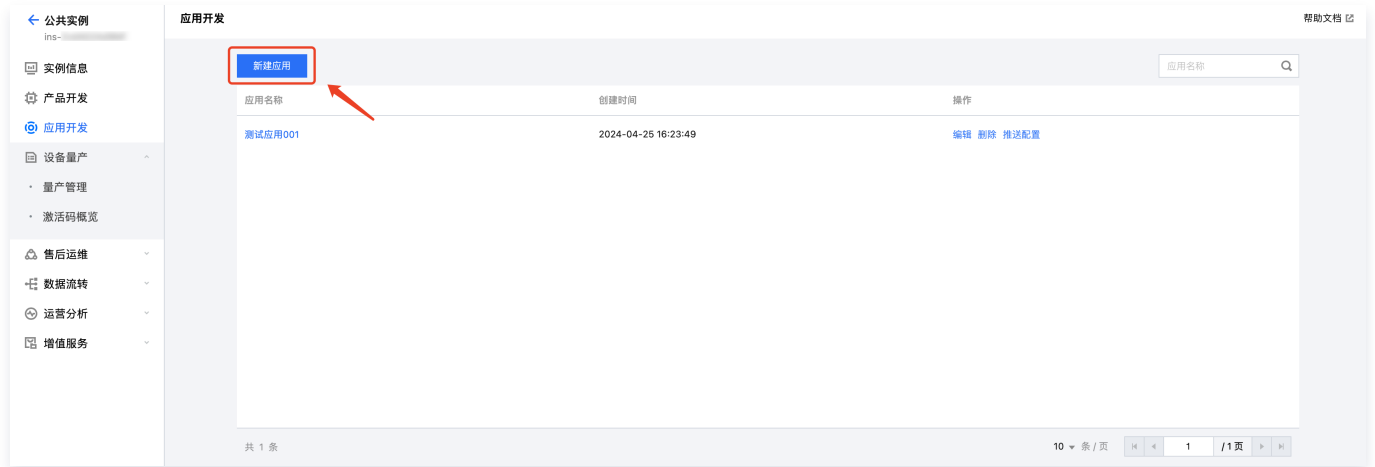
Log in to [IoT Explorer](#). After selecting an instance, enter the menu **Mass Production > Activation Code Overview > Value-added Activation Code**. On this page, you can view the total number of TWeCall activation codes, the number of used activation codes and the number of unused activation codes of the current account.



Mini Program Authorization

Mini programs need to be associated with applications of IoT Explorer before activation code assignment can be carried out.

1. After selecting an instance, enter the menu **Application Development**, click **Create Application** to create an application and associate it with a mini program.



2. Enter the menu **Application Development > Details**, click to enter the application name to open the application details page of **Tencent Cloud Micro Call**, and click **proceed to authorize**.

Note:

If there is no assigned entry for allocation, contact as soon as possible to purchase a value-added activation code.



3. In the mini program authorization popup, select the application that needs authorization. Click **Start Authorization**, and the system will automatically generate an authorization QR code. Please ask the administrator of the mini program to scan the code for authorization to complete.





TWeSee Service Video Summary

Last updated: 2025-04-27 17:47:32

This document mainly introduces the enabling, resource creation, API call, algorithm configuration steps of the video summary service as well as related usage scenarios.

Activate Service

Before using this capacity, you need to contact the product manager to enable the "AI Video Analysis Capacity" allowlist for your Tencent Cloud account and the target product on IoT Explorer. The following information needs to be submitted when enabling.

- Tencent Cloud account ID. You can log in to Tencent Cloud official website, click in the upper right corner on your avatar to view and copy the required account ID.

Note:

If you are a sub-account type, you also need to submit the main account ID to the product manager to enable allowlist.

- The target product ID of the Internet of Things development platform (the product must be a video service category). You can log in to the [console](#), enter the corresponding instance, and copy the product ID on the **product development** page. If you have not created a product yet, you can refer to the following "create resource" to create a video category product.

The screenshot shows the Tencent Cloud IoT Explorer console. The top navigation bar includes a search bar, a 'Quick Start' button, and various menu items like 'Group Account', 'Reservations', 'Tools', 'Customer Support', 'Fees', 'Language', and a user profile icon. The main content area is titled 'Product Development' and features a 'Smart Light' category. A sidebar on the right displays account information, including 'Sub-account ID' and 'Main account ID', both highlighted with red boxes. The main content area shows a workflow for creating a product: 1. Product Model Definition, 2. Device Development, 3. Interaction Development, 4. Device Debugging. Below the workflow, there is a 'New Product' button and a table of products. The 'Product ID' column in the table is highlighted with a red box.

Creating resources

1. Take creating a product under a public instance as an example. Enter the [IoT Explorer console](#), click the public instance, select **Product Development > Self-owned Products**, and then click **Create Product**.



2. You can select the product category by clicking the option in the red box in the figure below, enter the product name, and choose "WiFi" as the communication type.



3. Click **Create Product**. Saving succeeded.

产品开发 / 新建产品 ota测试 广州 回到旧版

填写产品信息

产品名称 支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

设备类型 设备 网关 子设备

平均传输速率 0.5 Mbps 1.0 Mbps 1.5 Mbps 2 Mbps 适合最高 200万像素 (1920*1080) 的 H.264 编码设备或最高 300万像素 (2304*1296) 的 H.265 编码设备

服务期限 设备激活后 5 年可用，如您有其它服务期限需求，可[咨询商务](#)

通信方式 Wi-Fi 请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

数据协议 物模型 自定义透传 ①

描述 最多不超过80个字符

新建产品 取消

4. Enable the advanced function of AI video analysis for products in IoT Explorer.

Note:

- This operation requires you to provide your Tencent Cloud account ID and product ID to the product manager to activate the service.
- If you encounter a service failure when enabling the service, please troubleshoot whether there are devices under the current product. Ensure that there are no devices under this product before you can successfully enable the service.



Click the product that was successfully created, enter **Thing Model Definition > Advanced Function**, and enable "AI Video Analysis Capability".

产品开发 / IPC01 式

广州 回到旧版 帮助文档

1 物模型定义 2 设备开发 3 交互开发 4 设备调试 5 批量投产

导入物模型 查看物模型JSON

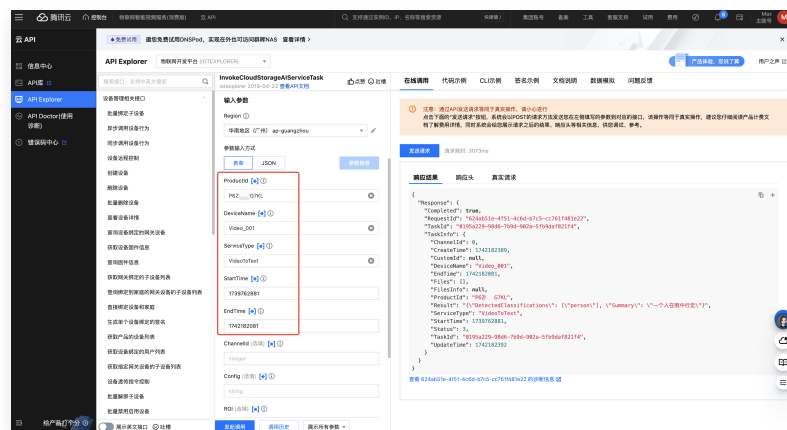
物模型定义帮助

标准功能 (8) 自定义功能 (2) 高级功能 (1)

| 功能名称 | 收费方式 | 功能描述 | 启用 | 操作 |
|----------|--------|--|-------------------------------------|---|
| 实时音视频 | 按设备收费 | 专为物联网场景打造的音视频通话解决方案, 支持智能终端与小程序、App 一对一音视频通话, 支持一键呼叫、分组呼叫, 提供应用端、设备端 SDK。 | <input type="checkbox"/> | 查看文档 |
| 语音助手 | 按设备收费 | 腾讯连连物联网平台打通了业内主流的三方语音技能平台, 提供快速接入三方平台并支持通过语音控制腾讯连连生态智能设备的能力。目前已经打通的语音技能平台包括Amazon Alexa、Google Assistant、百度小度和云小微。 | <input type="checkbox"/> | 查看文档 |
| 腾讯云微通话 | 按设备收费 | 设备音视频呼叫微信用户, 微信提供原生音视频呼叫提醒和音视频通话 | <input type="checkbox"/> | 查看文档 |
| QQ 音乐服务 | 按设备收费 | 物联网平台整合QQ音乐丰富内容资源, 面向IoT终端设备提供一站式接入能力, 从而快速实现音乐登录鉴权/独立会员订单/单播等功能。 | <input type="checkbox"/> | 查看文档 |
| AI视频分析能力 | 按月套餐收费 | 提供视频AI分析能力, 助力视频设备图像感知。目前已支持包裹检测算法, 基于云存视频分析有效识别快速包裹。 | <input checked="" type="checkbox"/> | 查看文档 消息接收设置 |

Calling API Trial Summary Feature Synchronization Request

1. Enter [cloud API console](#), request TencentCloud API [InvokeCloudStorageAIServiceTask](#) to execute video semantic understanding.
2. Region, ProductId, DeviceName, ServiceType, StartTime, EndTime, VideoURLs.N are required items, and others are optional.

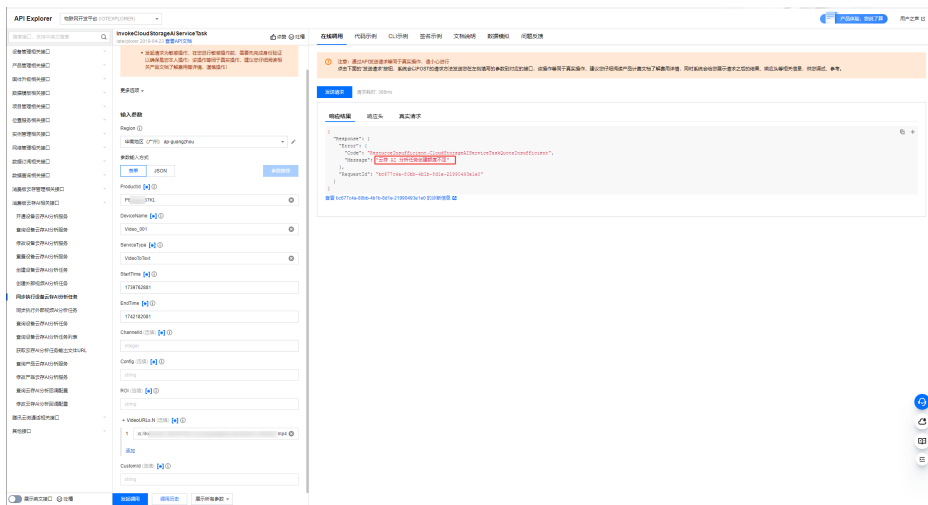


| Parameter Name | Parameter Type | Description | Value Description |
|----------------|----------------|--------------|----------------------------|
| Product Id | string | product ID. | Import actual information. |
| Device Name | string | Device name. | Import actual information. |

| | | | |
|--------------|----------|---|---|
| Service Type | string | Video AI Analysis Service Type. It can be set to RealtimeObjectDetect (target detection), Highlight (video highlights), and VideoToText (video semantic understanding). | Fixed to send "VideoToText". |
| StartTime | int64 | Start time of cloud storage to be analyzed (second-level UNIX timestamp, no earlier than 30 days before the current time). | Input according to the actual situation. You can use a timestamp tool: Online Unix Timestamp Conversion Tool . |
| EndTime | int64 | End time (in UNIX timestamp format, not later than 1 hour after the current time) of the cloud storage to be analyzed. | Input according to the actual situation. You can use a timestamp tool: Online Unix Timestamp Conversion Tool . |
| VideoURLs | string[] | Analyze the list of images/videos URLs from external input. <ul style="list-style-type: none"> Video: Support HLS video-on-demand (m3u8) and common video formats (mp4, etc.). Image: Support common image formats (jpg, png, etc.). If an input image is provided, "IsImage": true needs to be configured in Config. | Currently only 1 URL is supported. If multiple URLs are input, only the first one will be analyzed. |
| Config | string | Algorithm configuration parameters (JSON string). | Optional parameter, value example: " {\"MultiCameraLayout\": \"Single\"} Field description: see below Field description of algorithm configuration parameters (Config) . |

- (Return results within 20 seconds) Return Completed = true, and return the task result simultaneously. The task is complete.
- (Timeout within 20 seconds) Return Completed = false, indicating that the task is not completed within the timeout period (backlogged/incomplete tasks will continue to be queued and executed). The execution results of asynchronous tasks need to be retrieved asynchronously.
 - Method one: Query task results asynchronously via TencentCloud API.
 - Method two: Receive task result push through HTTP callback.

If the prompt "Credit insufficient for creating Tencent Cloud AI Analysis Task" appears, it means the account ID is not added to the allowlist. Contact the product manager to activate the service.



Request Example: Analyze Video – Successfully

```
POST / HTTP/1.1
Host: iotexplorer.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: InvokeCloudStorageAIServiceTask

{
  "ProductId": "AABBCCDDEE",
  "DeviceName": "dev123",
  "ServiceType": "VideoToText",
  "StartTime": 1714240800,
  "EndTime": 1714240802,
  "VideoURLs": ["https://example.com/video.mp4"],
  "Config": "{\"MultiCameraLayout\":\"Single\"}"
}
```

Response:

```
{
  "Response": {
    "RequestId": "8b490930-d119-4ee2-963c-c58973a1ebe6",
    "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",
    "Completed": true,
    "TaskInfo": {
      "ChannelId": 0,
      "CreateTime": 1714240812,
      "DeviceName": "dev123",
      "EndTime": 1714240802,
      "Files": [],
      "ProductId": "AABBCCDDEE",
      "Result": "{\"DetectedClassifications\": [\"person\"], \"Summary\": \"A person in white walks on a wet road surface\"}"
    }
  }
}
```



```

    "ServiceType": "VideoToText",
    "StartTime": 1714240800,
    "Status": 3,
    "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",
    "UpdateTime": 1714240814
  }
}
}

```

Request Example: Analyze Video – Timeout Incomplete

```

POST / HTTP/1.1
Host: iotexplorer.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: InvokeCloudStorageAIServiceTask

{
  "ProductId": "AABBCCDDEE",
  "DeviceName": "dev123"
  "ServiceType": "VideoToText",
  "StartTime": 1714240800,
  "EndTime": 1714240802,
  "VideoURLs": ["https://example.com/video.mp4"],
  "Config": "{\"MultiCameraLayout\":\"Single\"}"
}

```

Response:

```

{
  "Response": {
    "RequestId": "8b490930-d119-4ee2-963c-c58973a1ebe6",
    "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",
    "Completed": false,
    "TaskInfo": {
      "ChannelId": 0,
      "CreateTime": 1714240812,
      "DeviceName": "dev123",
      "EndTime": 1714240802,
      "Files": [],
      "ProductId": "AABBCCDDEE",
      "Result": "",
      "ServiceType": "VideoToText",
      "StartTime": 1714240800,
      "Status": 4,
      "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",

```

```
    "UpdateTime": 1714240812
  }
}
```

Request Example: Analyze Video – Video Download/Read Fail

```
POST / HTTP/1.1
Host: iotexplorer.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: InvokeCloudStorageAIServiceTask

{
  "ProductId": "AABBCCDDEE",
  "DeviceName": "dev123"
  "ServiceType": "VideoToText",
  "StartTime": 1714240800,
  "EndTime": 1714240802,
  "VideoURLs": ["https://example.com/video.mp4"],
  "Config": "{ \"MultiCameraLayout\": \"Single\" }"
}
```

Response:

```
{
  "Response": {
    "RequestId": "8b490930-d119-4ee2-963c-c58973a1ebe6",
    "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",
    "Completed": true,
    "TaskInfo": {
      "ChannelId": 0,
      "CreateTime": 1714240812,
      "DeviceName": "dev123",
      "EndTime": 1714240802,
      "Files": [],
      "ProductId": "AABBCCDDEE",
      "Result": "",
      "ServiceType": "VideoToText",
      "StartTime": 1714240800,
      "Status": 2,
      "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",
      "UpdateTime": 1714240813
    }
  }
}
```

Request Example: Analyze Image – Successful

```
POST / HTTP/1.1
Host: iotexplorer.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: InvokeCloudStorageAIServiceTask

{
  "ProductId": "AABBCCDDEE",
  "DeviceName": "dev123"
  "ServiceType": "VideoToText",
  "StartTime": 1714240800,
  "EndTime": 1714240802,
  "VideoURLs": ["https://example.com/image.jpg"],
  "Config": "{\\"IsImage\\":true}"
}
```

Response:

```
{
  "Response": {
    "RequestId": "8b490930-d119-4ee2-963c-c58973a1ebe6",
    "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",
    "Completed": true,
    "TaskInfo": {
      "ChannelId": 0,
      "CreateTime": 1714240812,
      "DeviceName": "dev123",
      "EndTime": 1714240802,
      "Files": [],
      "ProductId": "AABBCCDDEE",
      "Result": "{\\"DetectedClassifications\\": [\\"person\\"], \\"Summary\\": \\"A person in white walks on a wet road surface\\"}",
      "ServiceType": "VideoToText",
      "StartTime": 1714240800,
      "Status": 3,
      "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",
      "UpdateTime": 1714240813
    }
  }
}
```

Routine: Request TencentCloud API Using Python SDK

```
pip install tencentcloud-sdk-python
```

```
# -*- coding: utf-8 -*-
from tencentcloud.common.common_client import CommonClient
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.common.profile.client_profile import ClientProfile
from tencentcloud.common.profile.http_profile import HttpProfile

try:
    cred = credential.Credential(
        "", # SecretId
        "", # SecreyKey
    )

    httpProfile = HttpProfile()
    httpProfile.endpoint = "iotexplorer.tencentcloudapi.com"
    clientProfile = ClientProfile()
    clientProfile.httpProfile = httpProfile

    params = {
        "ProductId": "AABBCCDDEE",
        "DeviceName": "dev123",
        "ServiceType": "VideoToText",
        "StartTime": 1714240800,
        "EndTime": 1714240802,
        "VideoURLs": [
            "https://example.com/video.mp4",
        ],
        "Config": "{\"MultiCameraLayout\":\"Single\"}"
    }
    common_client = CommonClient("iotexplorer", "2019-04-23", cred, "ap-guangzhou",
profile=clientProfile)
    resp = common_client.call_json("InvokeCloudStorageAIServiceTask", params)
    print(resp)
except TencentCloudSDKException as err:
    print(err)
```

Asynchronous Request

1. Request the TencentCloud API [CreateCloudStorageAIServiceTask](#) to create a video semantic understanding task.

| Parameter Name | Parameter Type | Description | Value Description |
|----------------|----------------|-------------|-------------------|
|----------------|----------------|-------------|-------------------|

| | | | |
|--------------|----------|---|--|
| Product Id | string | product ID. | Import actual information. |
| Device Name | string | Device name. | Import actual information. |
| Service Type | string | Video AI Analysis Service Type. It can be set to RealtimeObjectDetect (target detection), Highlight (video highlights), and VideoToText (video semantic understanding). | Fixed to send "VideoToText". |
| StartTime | int64 | Start time of cloud storage to be analyzed (second-level UNIX timestamp, not earlier than 30 days before the current time). | Input according to the actual situation. You can use a timestamp tool: Online Unix Timestamp Conversion Tool . |
| EndTime | int64 | End time (in UNIX timestamp format, not later than 1 hour after the current time) of the cloud storage to be analyzed. | Input according to the actual situation. You can use a timestamp tool: Online Unix Timestamp Conversion Tool . |
| VideoURLs | string[] | Analyze the list of images/videos URLs from external input. <ul style="list-style-type: none"> Video: Support HLS video-on-demand (m3u8) and common video formats (mp4, etc.). Image: Support common image formats (jpg, png, etc.). If an input image is provided, "IsImage": true needs to be configured in Config. | Currently only 1 URL is supported. If multiple URLs are input, only the first one will be analyzed. |
| Config | string | Algorithm configuration parameters (JSON string). | Optional parameter, value example: " {\"MultiCameraLayout\":\"Single\"} Field descriptions are provided below. Field descriptions for algorithm configuration parameters (Config) . |

2. Retrieve the video semantic understanding task result.

- Method one: Asynchronously query task results via Cloud API.
- Method two: Receive task result push through HTTP callback reception.

Request Example: Analyze Video

```
POST / HTTP/1.1
Host: iotexplorer.tencentcloudapi.com
Content-Type: application/json
```

```
X-TC-Action: CreateCloudStorageAIServiceTask

{
  "ProductId": "AABBCCDDEE",
  "DeviceName": "dev123"
  "ServiceType": "VideoToText",
  "StartTime": 1714240800,
  "EndTime": 1714240802,
  "VideoURLs": ["https://example.com/video.mp4"]
}
```

Response:

```
{
  "Response": {
    "RequestId": "8b490930-d119-4ee2-963c-c58973a1ebe6",
    "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8"
  }
}
```

Request Example: Analyze Video (With Algorithm Parameters)

```
POST / HTTP/1.1
Host: iotexplorer.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateCloudStorageAIServiceTask

{
  "ProductId": "AABBCCDDEE",
  "DeviceName": "dev123"
  "ServiceType": "VideoToText",
  "StartTime": 1714240800,
  "EndTime": 1714240802,
  "VideoURLs": ["https://example.com/video.mp4"],
  "Config": "{\"MultiCameraLayout\":\"Single\"}"
}
```

Response:

```
{
  "Response": {
    "RequestId": "8b490930-d119-4ee2-963c-c58973a1ebe6",
    "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8"
  }
}
```

Request Example: Analyze an Image

```
POST / HTTP/1.1
Host: iotexplorer.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateCloudStorageAIServiceTask

{
  "ProductId": "AABBCCDDEE",
  "DeviceName": "dev123"
  "ServiceType": "VideoToText",
  "StartTime": 1714240800,
  "EndTime": 1714240802,
  "VideoURLs": ["https://example.com/image.jpg"],
  "Config": "{\"IsImage\":true}"
}
```

Response:

```
{
  "Response": {
    "RequestId": "8b490930-d119-4ee2-963c-c58973a1ebe6",
    "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8"
  }
}
```

Routine: Request TencentCloud API Using Python SDK

```
pip install tencentcloud-sdk-python
```

```
# -*- coding: utf-8 -*-
from tencentcloud.common.common_client import CommonClient
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.common.profile.client_profile import ClientProfile
from tencentcloud.common.profile.http_profile import HttpProfile

try:
    cred = credential.Credential(
        "", # SecretId
        "", # SecreyKey
    )

    httpProfile = HttpProfile()
    httpProfile.endpoint = "iotexplorer.tencentcloudapi.com"
    clientProfile = ClientProfile()
    clientProfile.httpProfile = httpProfile
    params = {
```

```
"ProductId": "AABBCCDDEE",
"DeviceName": "dev123",
"ServiceType": "VideoToText",
"StartTime": 1714240800,
"EndTime": 1714240802,
"VideoURLs": [
    "https://example.com/video.mp4",
],
"Config": "{\"MultiCameraLayout\":\"Single\"}"
}
common_client = CommonClient("iotexplorer", "2019-04-23", cred, "ap-guangzhou",
profile=clientProfile)
resp = common_client.call_json("CreateCloudStorageAIServiceTask", params)
print(resp)
except TencentCloudSDKException as err:
    print(err)
```

Algorithm Configuration Parameter (Config) Field Description

| Parameter Name | Parameter Type | Description | Value Description |
|-------------------|----------------|---|---|
| MultiCameraLayout | string | (Optional) Multi-camera video recognition. If the imported video is a tiled video footage from multiple cameras, you can input this parameter to obtain more accurate video semantic understanding results. | See below for MultiCameraLayout description. |
| EnableFlameDetect | boolean | (Optional) Enable flame detection and description. | <ul style="list-style-type: none">• true: Enable• false: Not enabled (default value). |
| EnableSmokeDetect | boolean | (Optional) Enable smoke detection and description. If you enable smoke detection for videos that are relatively blurry or contain mosaics, there may be false detections. | <ul style="list-style-type: none">• true: Enable• false: Not enabled (default value) |
| IsImage | boolean | (Optional) Whether it is image input. | <ul style="list-style-type: none">• true: Image input• false: Video input (default value) |
| IsNightVision | boolean | (Optional) Whether it is a night vision frame. | <ul style="list-style-type: none">• Night vision view• false: Non-night vision footage (default value) |

MultiCameraLayout Description

Multi-camera format: Vertical, Num = number of screens, Index = screen serial number

If analyzing multi-screen displays at the same time, you can fill in more than one screen number, separated by a semicolon (;).

Selectable values:

| Layout | MultiCameraLayout Parameter Value | Description |
|---|-----------------------------------|--|
| Single Camera | Single | Deem the video as one complete picture (default value). |
| Dual Camera (Resolution, Tile in Vertical Direction) | Vertical,Num=2,Index=0 | Specify the analysis of the first frame from top to bottom. |
| | Vertical,Num=2,Index=1 | Specify to analyze the second visual from top to bottom. |
| | Vertical,Num=2,Index=0;1 | Analyze all two visuals. |
| | Vertical | Analyze all two visuals (compatible with previous parameter values). |
| Three Cameras (Equal Resolution, Tiled Vertically) | Vertical,Num=3,Index=0 | Specify to analyze the first visual from top to bottom. |
| | Vertical,Num=3,Index=1 | Specify to analyze the second visual from top to bottom. |
| | Vertical,Num=3,Index=2 | Specify to analyze the third visual from top to bottom. |
| | Vertical,Num=3,Index=0;1 | Specify to analyze the first and second visuals from top to bottom. |
| | Vertical,Num=3,Index=0;2 | Specify to analyze the first and third visuals from top to bottom. |
| | Vertical,Num=3,Index=1;2 | Specify to analyze the second and third visuals from top to bottom. |
| | Vertical,Num=3,Index=0;1;2 | Analyze all three visuals. |

Querying Video AI Analysis Task Results Via TencentCloud API

Query the Specified Video AI Analysis Task by TaskId

Prerequisites

The Task ID (TaskId) of the video AI analysis task has been obtained through TencentCloud API or HTTP callback method.

Operation Steps

Request TencentCloud API [Query Device Cloud Storage AI Analysis Task](#) to query the details of the specified video AI analysis task by TaskId.

Request sample:

```
POST / HTTP/1.1
Host: iotexplorer.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeCloudStorageAIServiceTask

{
  "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8"
}
```

Sample response:

```
{
  "Response": {
    "RequestId": "8b490930-d119-4ee2-963c-c58973a1ebe6",
    "TaskInfo": {
      "ChannelId": 0,
      "CreateTime": 1714240802,
      "DeviceName": "dev123",
      "EndTime": 1714240802,
      "Files": [],
      "ProductId": "TSLFHRWDS",
      "Result": "{ \"Summary\": \"Abstract text\", \"DetectedClassifications\": [ \"car\" ] }",
      "ServiceType": "VideoToText",
      "StartTime": 1714240800,
      "Status": 3,
      "TaskId": "c31aa4f2-08c9-4088-9603-186d7311fdd8",
      "UpdateTime": 1714240802
    }
  }
}
```

Receiving Video AI Analysis Results Via HTTP Callback

The platform supports receiving video AI analysis results via HTTP callback. You need to deploy an http service that complies with the callback API definition and make it accessible from the public network to achieve the reception of video AI analysis results.

Callback API Definition

(Refer to the following HTTP service routine provided)

- GET / Connectivity check: Fetch the value of the HTTP request header Echostr, use it as the content of the HTTP response, and return a 200 OK status code.
- POST / Analysis result callback: Receive video AI analysis results pushed by the platform from this API. The video AI analysis results are in JSON format and located in the HTTP request body. After successfully receiving this request, a 200 OK status code should be returned.

Configure HTTP Callback URL

Prerequisites

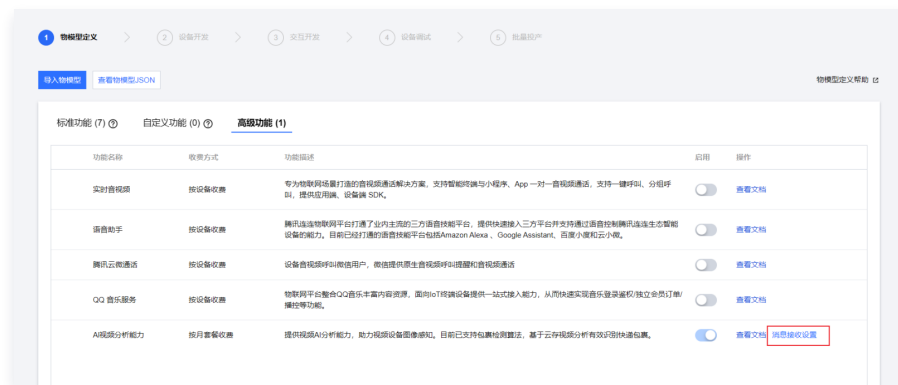
1. The product has activated the AI video analysis service.
2. A deployed HTTP callback service that is external network accessible.

Note:

When configuring the HTTP callback URL on the console or through TencentCloud API, the platform will initiate a connectivity check request to the specified URL. If the connectivity check fails, the HTTP callback configuration will not be saved and take effect. Meanwhile, it will return error code `InvalidParameterValue.InvalidCallbackUrl`.

Operation Steps

1. Log in to the console of [IoT Explorer](#).
2. Select an instance and a product sequentially, and enter the product detail page.
3. Select the **Advanced Function** tab, and click **Receive Settings** in the operation column of AI Video Analysis capability.



4. Fill in the HTTP callback address (and authentication Token, if necessary).

消息接收设置

您可以在此处设置 AI 视频分析结果推送目的端，支持推送至第三方服务器，[点击查看文档](#) 了解如何开发 HTTP 服务接受物联网平台数据

数据转发到第三方服务 (Forward)

API 地址: *

使用已有 HTTP 服务地址

增加鉴权 token

保存

取消

5. Click **Save** to save the configuration.

Possible Reasons for Configuration Failure

- The specified callback URL is in incorrect format.
- The specified callback URL is unable to be accessed via the public network.
- The port number in the specified callback URL is not 80 or 443.
- The platform triggers a connectivity check to the specified callback URL, but no HTTP 200 OK status code is returned.
- When the platform initiates a connectivity check to the specified callback URL, it does not return the Echostr in the request header as the response content.

Callback Request Authentication

The authentication method is the same as that of [Message Queue](#) HTTP forwarding. You can configure an authentication Token to verify that requests come from the IoT intelligent video service (Consumer Edition) platform. If an authentication Token is configured, requests initiated by the platform (including connectivity check requests and analysis result callback requests) will include the following HTTP request headers. Among them, Nonce and Timestamp are used for replay attack prevention, and Signature can be used for confirmation that the request comes from the platform.

| Request Header | Meaning | Example Value |
|----------------|-----------------------------|--|
| Nonce | random string | pWttcBmqRpOfdlph |
| Timestamp | Second-level UNIX timestamp | 1714285922 |
| Signature | signature value | 5d5a0ac2dd41f5b5e553f95e0f657247281aea0b |

Based on the configured authentication Token, as well as the Nonce and Timestamp values in the HTTP request header, calculate the signature using the same algorithm. If the signature value matches, the request can be considered to come from the platform. The steps for calculating the signature value Signature are as follows:

1. Sort the three strings Nonce, Timestamp and authentication Token lexicographically, and then concatenate them sequentially to get a string (in the above example, assume that the authentication Token is example, and the concatenated string is 1714285922examplepWttcBmqRpOfdlph).
2. Calculate the SHA1 hash value of the concatenated string.
3. Use the lowercase hexadecimal representation of the hash value as the value of Signature.

Example: Connectivity Check

Request

When configuring the HTTP callback URL on the console or through TencentCloud API, the platform initiates a connectivity check request to the URL.

```
GET / HTTP/1.1
User-Agent: IotVideo/1.0
```

Echostr: 3A4dsItMWYmL7pkL

Response

When the http service receives a connectivity check request, it should return a response with a status code of 200 OK, and the response content should be the value of the HTTP request header Echostr.

200 OK
Content-Type: text/plain
Content-Length: 16
3A4dsItMWYmL7pkL

Example: Connectivity Check (Including Authentication Information)

Request

This example assumes the authentication Token is example.

GET / HTTP/1.1
User-Agent: IotVideo/1.0
Echostr: 3A4dsItMWYmL7pkL
Nonce: pWttcBmqRpOfdlph
Signature: 5d5a0ac2dd41f5b5e553f95e0f657247281aea0b
Timestamp: 1714285922

Response

200 OK
Content-Type: text/plain
Content-Length: 16
3A4dsItMWYmL7pkL

Analysis Result Data Structure

| Field Name | Field description | Example Value |
|-------------|--|-----------------------|
| MsgType | Fixed value, indicates video AI analysis result. | "ai_analytics_result" |
| ProductId | product ID. | "AABBCCDDEE" |
| DeviceName | Device name. | "dev123" |
| ServiceType | Video AI analysis service type, Parameter Value: VideoToText for video semantic understanding. | "VideoToText" |
| Status | Video AI analysis task status. Possible values: <ul style="list-style-type: none">Failure. | 3 |

| | | |
|-----------|---|---|
| | <ul style="list-style-type: none">2: Succeed but result is empty.3: Succeed and result is not empty. | |
| Result | Analysis result details (JSON string). | Summary: Abstract text DetectedClassifications: car |
| Files | File list output by video AI analysis. | [] |
| Time | Analysis result callback time. | "2024-04-10 18:12:05" |
| StartTime | Corresponding cloud storage segment start time (second-level UNIX timestamp). | 1714240800 |
| EndTime | The segment end time (in UNIX timestamp format) of the corresponding cloud storage clip. | 1714240802 |

Example: Video Semantic Understanding Result Callback

Request

Upon completion of the video AI analysis task, the platform will request the specified HTTP callback URL carrying the analysis result in JSON format, which is located in the HTTP request body.

```
POST / HTTP/1.1
User-Agent: IotVideo/1.0
Content-Type: application/json

{
  "MsgType": "ai_analytics_result",
  "ProductId": "AABBCCDDEE",
  "DeviceName": "dev123",
  "ServiceType": "VideoToText",
  "TaskId": "95990598-aba1-465e-9993-3f90c071e5d9",
  "Status": 3,
  "Result": "{\"Summary\": \"Abstract text\", \"DetectedClassifications\": [\"car\"]}"
  "Files": [],
  "Time": "2024-04-10 18:12:05",
  "StartTime": 1714240800,
  "EndTime": 1714240802
}
```

Routine: Receiving callback HTTP service routine source code (Python)

```
from flask import Flask, request, make_response
import hashlib
import traceback

LISTEN_HOST = "0.0.0.0"
```

```
LISTEN_PORT = 80
AUTH_TOKEN = "example" # Authentication token, if empty, indicating that
authentication is not required

app = Flask(__name__)

# Verify the validity of the request according to the authentication Token
def is_signature_valid(nonce, timestamp, signature):
    if nonce is None or timestamp is None or signature is None:
        return False

    plain_text = "".join(sorted([nonce, timestamp, AUTH_TOKEN]))

    sha1_hash = hashlib.sha1(plain_text.encode("utf-8")).hexdigest()

    expected_signature = sha1_hash.lower()

    return expected_signature == signature.lower()

# Connectivity check
def handle_connectivity_check(request):
    if AUTH_TOKEN and not is_signature_valid(
        nonce=request.headers.get("Nonce"),
        timestamp=request.headers.get("Timestamp"),
        signature=request.headers.get("Signature"),
    ):
        return "Forbidden", 403
    echostr = request.headers.get("Echostr")
    if echostr:
        return make_response(echostr, 200, {"Content-Type": "text/plain"})
    else:
        return "Bad Request", 400

# Analytical result callback
def handle_analytic_result(request):
    if AUTH_TOKEN and not is_signature_valid(
        nonce=request.headers.get("Nonce"),
        timestamp=request.headers.get("Timestamp"),
        signature=request.headers.get("Signature"),
    ):
        return "Forbidden", 403

    post_data = request.get_data(as_text=True)
    # TODO: Handle the analysis result

    return "OK", 200, {"Content-Type": "text/plain"}
@app.route("/", methods=["GET", "POST"])
def handle_request():
    try:
        # Print request
```

```
print("[Request]", request.method, request.path)
for key, value in request.headers:
    print(f"{key}: {value}")
print()

if request.method == "POST":
    print(request.get_data(as_text=True))
    print()

Process a request
if request.method == "GET":
    return handle_connectivity_check(request)
elif request.method == "POST":
    return handle_analytic_result(request)
else:
    return "Method Not Allowed", 405
except Exception as e:
    print("[Exception]")
    traceback.print_exc()
    return "Internal Server Error", 500

if __name__ == "__main__":
    app.run(port=LISTEN_PORT, host=LISTEN_HOST)
```

Semantic Understanding Output Result

The semantic understanding result is in the form of status value + specific result.

- **status value**

The status value is in the Status field of the video AI analysis task. It can be queried through TencentCloud API or received through HTTP callback. Possible values:

- 1: Fail
- 3: Successful and result is not empty.

- **Detailed Result**

The specific result is in the Result field of the video AI analysis task. It can be queried through TencentCloud API or received through HTTP callback. Result is a JSON Object string. The data structure of the Result object is as follows:

| Field | Type | Description |
|-------------------------|----------|---|
| Summary | string | Video summary text in the form of natural language. |
| DetectedClassifications | string[] | List of target types identified in the video. Possible parameter values of the target type in the list <ul style="list-style-type: none">• person• vehicle• dog• cat• fire• smoke |

| | | |
|--|--|---|
| | | <ul style="list-style-type: none">• package• license plate |
|--|--|---|

Example value:

```
{"Summary": "Video summary text", "DetectedClassifications": ["person", "vehicle"]}
```

Permission Management

Creating Sub-account

Last updated: 2025-04-27 17:47:57

Overview

this document mainly introduces how to add a sub-account to a root account and assign a certain level of resource management permissions to the sub-account, taking "Sub-user" as an example.

Operation Steps

1. Log in to the [CAM console](#) with the Tencent Cloud root account. Select the left menu bar **User > List of users**.
2. Enter the user list page and click **Create User**.
3. A user type selection interface pops up. Select **Custom creation** for the sub-user.
4. Select "access resources and receive messages" and click **Next**.
5. Fill in the required information for the Sub-user. You can customize the "username", corresponding phone number, and select "console access" as the access method. After selecting console access, you need to set the password for this Sub-user. If you plan to access through API, it is recommended to choose "programmatic access", as shown in the figure below.

6. After setting the password, click **Next** to perform identity verification using the mobile verification code bound to the root account.
7. Enter the authorization interface, enter the predefined "policy" in the text box, check it, and click **Next** to enter the final confirmation interface.

8. Click **Complete**, indicating that the root account has assigned a certain "policy" to the "sub-user".
9. After the root account creates a Sub-user, the Sub-user can access the Tencent Cloud console through the assigned sub-account. Generally, view the user details page of the Sub-user under the root account. The console login link is shown on the right of the page (as shown in the red box in the figure below).



How to Configure CAM Permission for a Sub-Account

Last updated: 2025-04-27 17:48:15

Overview

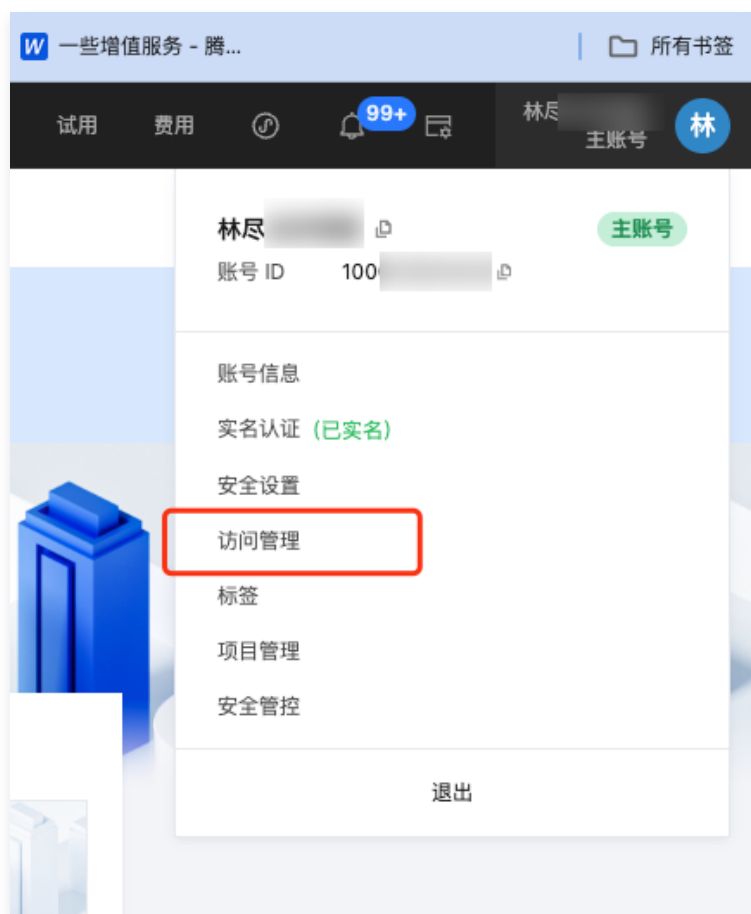
This document mainly introduces How to grant sub-accounts product-level access control permissions. Product-level access control permissions can let sub-accounts have access control capabilities for products they create themselves or products created by the root account for them. The root account has created one or more products and devices, which are assigned to different partners in two ways. As shown below:

| 新建产品 | | | | | | | 按产品名称 ▾ | 请输入产品名称 |
|--------|-------|----------------------|------|------|---------------------|----|---------|---------|
| 产品名称 | 产品ID | 产品品类 | 设备类型 | 状态 了 | 创建时间 | 操作 | | |
| 三位智能开关 | ATINV | 全屋智能-电工精品-三位智能开关（零火） | 设备 | 开发中 | 2024-06-06 17:33:58 | 删除 | | |
| 两轮车 | FOS2 | 智能城市-公共事业-两轮车 | 设备 | 开发中 | 2024-04-11 11:42:59 | 删除 | | |
| 智能情景台灯 | G4IQ | 全屋智能-灯光照明-智能情景台灯 | 设备 | 开发中 | 2024-04-09 11:50:07 | 删除 | | |
| 气体检测器 | IYF7 | 智慧农业-传感器-气体检测器 | 设备 | 开发中 | 2024-04-08 11:56:57 | 删除 | | |
| 三键无线开关 | WZ98 | 全屋智能-电工精品-智能无线开关（三键） | 设备 | 开发中 | 2024-03-11 17:52:21 | 删除 | | |

Create by Policy Generator

Creating a Policy

1. Log in to the [CAM Console](#) with your Tencent Cloud Main Account, click on the account in the upper right corner, and turn on access management.

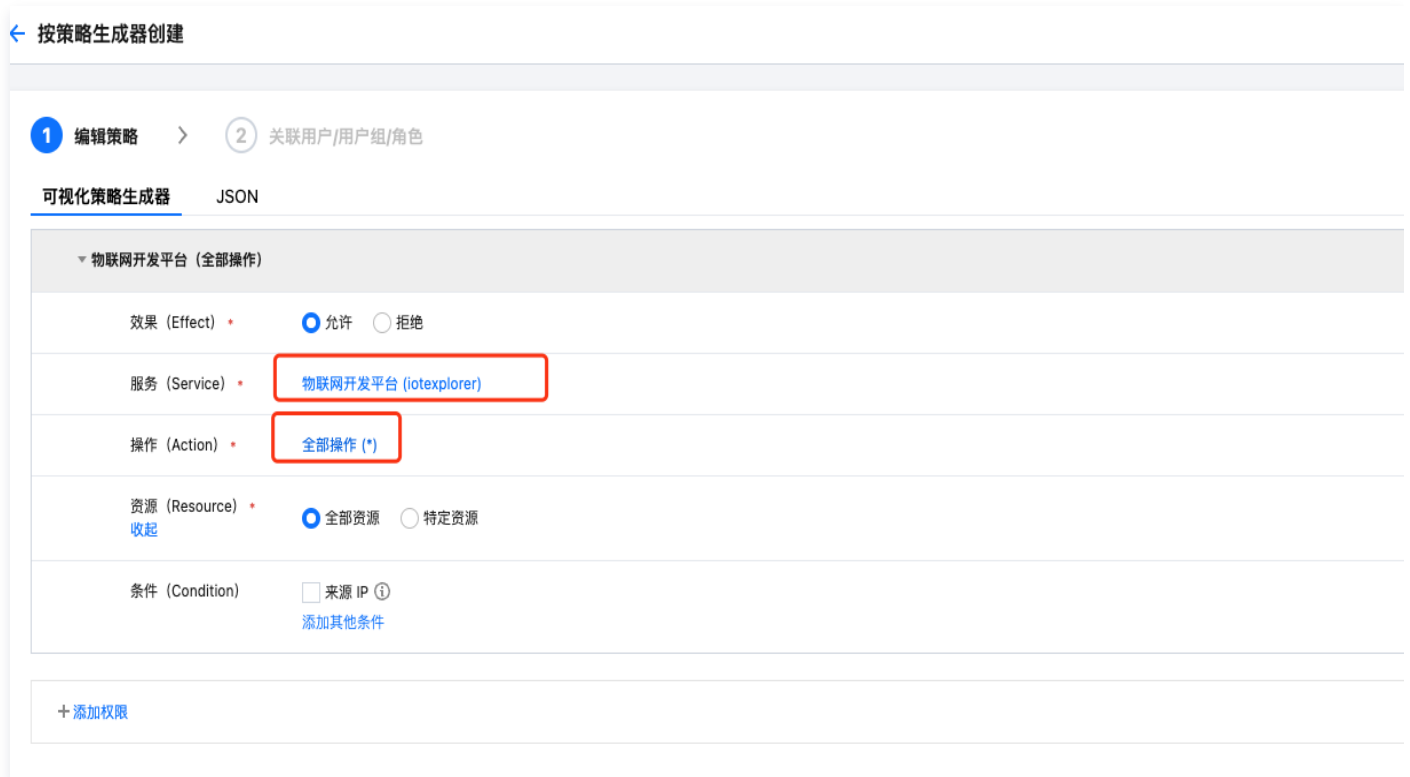


2. Click **Policies > Create Definition Policy**. Select **"Create by Policy Generator"** as the policy method.





3. Select the service `iotexplorer`. If necessary, to grant all APIs permissions to the sub-user, just select all operations.



If necessary, specify API access permissions for the sub-user and select the corresponding API as needed. For example, select API: `GetCategoryPanelsByCategoryKey`.

[← 按策略生成器创建](#)**1 编辑策略** > **2 关联用户/用户组/角色****可视化策略生成器** JSON

▼ 物联网开发平台 (1 个操作)

效果 (Effect) * ☒ 允许 ☐ 拒绝

服务 (Service) * 物联网开发平台 (iotexplorer)

操作 (Action) * 读操作 编辑

GetCategoryPanelsByCategor...

拉取指定CategoryKey分类面板

资源 (Resource) * ☒ 全部资源 ☐ 特定资源
收起条件 (Condition) ☐ 来源 IP ⓘ
添加其他条件

+ 添加权限

下一步

字符数: 197 (最多6144)

4. Resources can be selected as all resources or specific resources as needed. After completion, click **Next** to associate a policy.

Associated Policy

1. Click "Select User" to grant permissions to the specified sub-user.

[← 按策略生成器创建](#)

1 编辑策略 > 2 关联用户/用户组/角色

基本信息

策略名称 *

policygen-20240612094920

策略创建后，策略名称不支持修改

描述

请输入策略描述

关联用户/用户组/角色

将此权限授权给用户

[选择用户](#)

将此权限授权给用户组

[选择用户组](#)

将此权限授权给角色

[选择角色](#)[上一步](#)[完成](#)

2. After selecting a user, click **Confirm** > **Complete** to take effect.

关联用户

选择添加的用户（共 2 个）

支持多关键词(间隔为空格)搜索用户名/ID/SecretId/手机/邮箱/密

用户

类型

☐ leo_sub002

用户

☒ leosp_sub01

用户

已选择 (1) 个

名称

类型

leosp_sub01

用户

支持按住 shift 键进行多选

确定

取消

Create by Policy Syntax

Creating a Policy

1. Log in to the [CAM console](#) with the Tencent Cloud root account and click **Policies** in the left menu.
2. Enter the Policy page and click **Create Custom Policy**.
3. Select **Create by policy syntax**.
4. Select template type, check **Blank Template**, and click **Next**.
5. Fill in the Custom policy name and edit the policy content according to the policy template.

©2013–2025 Tencent Cloud. All rights reserved.

Page 331 of 334

← 按策略语法创建

✓ 选择策略模板 > 2 编辑策略

策略名称 * policygen-20240621175405

策略创建后，策略名称不支持修改

描述 编辑智能家居项目的子账号策略

策略内容 [使用旧版](#)

```
1 {  
2   "version": "2.0",  
3   "statement": []  
4 }
```

Below is a sample code:

- Example of allocating all permissions to a sub-account

```
{  
  "version": "2.0",  
  "statement": [  
    {  
      "action": [  
        "*"   
      ],  
      "resource": [  
        "qcs::iotcloud:gz:uin/your_uid:*",  
        "qcs::iotexplorer:gz:uin/your_uid:project/your_project_id",  
        "qcs::iotexplorer:gz:uin/your_uid:project/your_project_id/",  
        "qcs::iotexplorer:gz:uin/your_uid:project/your_project_id/product/your_product_id",  
        "qcs::iotexplorer:gz:uin/your_uid:project/your_project_id/product/your_product_id/*"  
      ],  
      "effect": "allow"  
    }  
  ]  
}
```

Policy description as follows:

- Resource corresponds to projects and products. If you want to authorize a certain product ID of a certain project ID of the root account to a sub-user, you need to add the following four items in the resource part. The red marking indicates the parts to be replaced: `your_uid` is the user account ID, `your_project_id` is the console project ID, and `your_product_id` is the product ID within the project.

```
"qcs::iotexplorer:gz:uin/your_uid:project/your_project_id",
"qcs::iotexplorer:gz:uin/your_uid:project/your_project_id/",
"qcs::iotexplorer:gz:uin/your_uid:project/your_project_id/product/your_product_id",
"qcs::iotexplorer:gz:uin/your_uid:project/your_project_id/product/your_product_id/*"
```

- * indicates all operations.
- Effect: allow means permission, deny means not allowed.
- Instructions for using project policy syntax, please see [policy syntax description](#).
- Disable partial permissions of a sub-account

Example code (here, the example disables sub-accounts from deleting products and devices):

```
{
  "version": "2.0",
  "statement": [
    {
      "action": [
        "*"
      ],
      "resource": [
        "qcs::iotcloud:gz:uin/your_uid:*",
        "qcs::iotexplorer:gz:uin/your_uid:project/your-project-id",
        "qcs::iotexplorer:gz:uin/your_uid:project/your-project-id/",
        "qcs::iotexplorer:gz:uin/your_uid:project/your-project-id/product/your-product-id",
        "qcs::iotexplorer:gz:uin/your_uid:project/your-project-id/product/your-product-id/*"
      ],
      "effect": "allow"
    },
    {
      "action": [
        "iotexplorer:DeleteStudioProduct",
        "iotexplorer:DeleteDevice"
      ],
      "resource": [
        "qcs::iotcloud:gz:uin/your_uid:*",
        "qcs::iotexplorer:gz:uin/your_uid:project/your-project-id",
        "qcs::iotexplorer:gz:uin/your_uid:project/your-project-id/",
        "qcs::iotexplorer:gz:uin/your_uid:project/your-project-id/product/your-product-id",
        "qcs::iotexplorer:gz:uin/your_uid:project/your-project-id/product/your-product-id/*"
      ],
      "effect": "deny"
    }
  ]
}
```

```
"effect": "deny"
  }
]
}
```

When a sub-account logs in to the console and deletes a project or product, a pop-up window will prompt no permission:



- Enter the relevant API name. For example: DeleteStudioProduct (delete product), DeleteDevice (delete device). For other specific API names, see the relevant APIs in [API Overview](#).
- Effect: allow means permission, deny means not allowed.

Associated Policy

1. After the custom policy is created, enter the **User > List of Users** page and select the sub-account to which you want to grant permissions.
2. Click the user name, enter the user details page, and in the "Permission" column, click **Associated Policy**.
3. Search for the policy name you just created, select it, then click **Confirm** to complete granting the permissions defined in the policy.