

# 阿里云 生活物联网平台

## 用户指南

文档版本：20190911

## 法律声明

---

阿里云提醒您 在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的”现状“、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含”阿里云”、Aliyun”、”万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

## 通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 <b>禁止：</b> 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 <b>警告：</b> 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 <b>说明：</b> 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
<b>粗体</b>	表示按键、菜单、页面名称等UI元素。	单击 <b>确定</b> 。
<code>courier</code> 字体	命令。	执行 <code>cd /d C:/windows</code> 命令，进入Windows系统文件夹。
<code>##</code>	表示参数、变量。	<code>bae log list --instanceid</code> <code>Instance_ID</code>
<code>[ ]</code> 或者 <code>[a b]</code>	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
<code>{ }</code> 或者 <code>{a b}</code>	表示必选项，至多选择一个。	<code>swich {stand   slave}</code>

# 目录

---

法律声明.....	I
通用约定.....	I
1 平台介绍.....	1
2 账号管理.....	2
3 站点管理.....	3
4 项目管理.....	4
4.1 创建项目.....	4
4.2 授权项目.....	5
4.3 重命名项目.....	8
4.4 删除项目.....	9
4.5 设置设备数据同步.....	10
4.6 设置用户数据同步.....	12
4.7 查看API接口.....	13
5 产品管理.....	16
5.1 新建产品.....	16
5.2 关联产品.....	18
5.3 复制产品.....	21
5.4 修改产品.....	23
5.5 删除产品.....	25
6 开发中心.....	27
6.1 定义产品功能.....	27
6.1.1 功能概述.....	27
6.1.2 新增标准功能.....	27
6.1.3 修改标准功能.....	30
6.1.4 删除标准功能.....	30
6.1.5 新增自定义功能.....	31
6.1.6 修改自定义功能.....	41
6.1.7 删除自定义功能.....	42
6.1.8 查看并导出物模型.....	42
6.1.9 导入物模型.....	57
6.2 调试设备.....	60
6.2.1 选择认证模组/芯片.....	60
6.2.2 透传/编辑脚本.....	62
6.2.3 新增测试设备.....	75
6.2.4 添加设备标签.....	76
6.2.5 在线调试设备.....	78
6.3 人机交互.....	84
6.3.1 功能概述.....	84
6.3.2 配置App交付终端.....	86

6.3.3 配置App分享方式.....	88
6.3.4 配置App多语言管理.....	89
6.3.5 配置App配网方案和引导文案.....	91
6.3.6 配置App设备告警.....	95
6.3.7 配置App界面.....	98
6.3.8 配置App自动化和定时.....	99
6.3.9 配置接入第三方平台.....	101
6.4 批量投产.....	102
6.5 创建自有App.....	103
<b>7 账单中心.....</b>	<b>112</b>
7.1 购买激活码.....	112
7.2 分配激活码.....	113
7.3 查看消费记录.....	114
<b>8 量产中心.....</b>	<b>115</b>
8.1 配置激活码烧录方式.....	115
8.2 查看量产记录.....	123
<b>9 服务中心.....</b>	<b>124</b>
<b>10 运营中心.....</b>	<b>126</b>
10.1 能力介绍.....	126
10.2 设置运营中心样式.....	130
10.3 固件升级.....	131



# 1 平台介绍

阿里云IoT提供的一款针对生活领域的物联网平台，平台针对家电智能化的设备连接、移动端控制、设备管理、数据统计等问题，提供了一整套配置化方案，大幅减低“设备-云端-App”的开发成本。

## 生活物联网平台

生活物联网平台是一款针对智能生活领域的物联网云平台。在阿里云的IaaS和PaaS层云产品的基础上，搭建了一套完整的、更贴近智能家电领域的公有云平台。助力生活领域的开发者、方案商，提供从功能设计、嵌入式开发调试、设备安全、云端开发、App开发、运营管理、数据统计等，覆盖前期开发到后期运营，全生命周期的服务。

## 功能特性

 <p><b>零门槛</b> 极速化客户</p>	<p><b>周期短</b> 1分钟完成面板配置 5小时完成智能化，10天完成量产</p> <p><b>成本低</b> 提供认证模组，天然连接云端 提供Demo APP源码，快速上手</p> <p><b>免开发</b> 提供公版APP，免开发App界面 提供标准运营后台，包括数据图表</p>	 <p><b>高灵活</b> 深定制客户</p>	<p><b>高可用性</b> 一机一密，保障个人隐私安全 亿级设备接入，保障连接质量</p> <p><b>全球化</b> 设备可在全球任意地区激活，助力中国智造全球化</p> <p><b>品牌自有</b> 提供云端 API，支持灵活自主开发业务逻辑</p>
---	---	--	--

## 与物联网平台的区别

[生活物联网平台](#)和[物联网平台](#)均为阿里云IoT提供的云服务平台，两个平台各自优势和使用场景如下。

- 物联网平台，提供原子化的设备接入能力，适用于云开发能力较强的用户，可以在各个行业领域使用。单击[什么是物联网平台](#)了解更多详情。
- 生活物联网平台，提供了设备接入能力、移动端的SDK以及免开发的公版App和界面，更适用于消费级的智能设备开发者，开发门槛较低，可以快速实现消费级设备的智能化，如智能家电、穿戴、家装领域等。

使用同一个阿里云账号登录的用户，在生活物联网平台创建的所有产品和设备，将自动同步到物联网平台中。而在物联网平台中创建的产品，也可以通过手动切换收费模式，将产品转移到生活物联网平台中。

## 2 账号管理

---

使用生活物联网平台之前，您需要注册阿里云账号，并进行实名认证。

### 注册流程

1. 进入 [阿里云官网](#)，单击页面右上角免费注册按钮。
2. 填写注册信息，阅读服务条款，单击同意条款并注册。

- 设置会员名：
  - 5-25 个字符，不能包含标点等特殊字符。
  - 尽量避免使用姓名、手机号、身份证、银行卡等隐私信息。
  - 推荐使用中文。



说明：

注册成功后，会员名不能修改。

- 设置登录密码：
    - 6-20 个字符。
    - 只能包含字母、数字以及标点符号（除空格）。
    - 字母、数字和标点符号三者中至少包含 2 种。
  - 输入手机号：请使用中国大陆的手机号。
3. 输入您的手机收到的验证码，完成手机验证。

完成验证后，账号注册成功。

为了您更好地使用阿里云服务，请补充以下信息：

- 单击注册成功页面绑定邮箱，绑定您的常用邮箱，以便及时收取阿里云产品和服务相关信息，包括产品开通、到期、故障、安全、关停等重要通知。
- 选择您所在的行业，以便我们更好地为您提供服务。

### 实名认证

如果要购买阿里云产品、提现账号中的现金余额等，需您完成阿里云账号的实名认证。

实名认证方法，请参见：

- [#unique\\_5](#)
- [#unique\\_6](#)

## 3 站点管理

阿里生活物联网平台支持全球化部署，在选择项目前，请先确认当前站点。默认为中国站。

### 中国站与国际站的区别

- 中国站：如果您的设备打算在国内售卖和使用，请选择在“中国站”创建项目、产品和App。
- 国际站：如果您的设备将售往海外，如美洲、澳洲、东南亚等地区，请选择在“国际站”创建项目、产品和App。



#### 说明：

由于欧盟GDPR和海外合规的要求，不能创建一个产品同时在中国站和国际站使用。如果您希望将一个App同时发布到中国站和国际站，需要在平台上的国内站和国际站分别创建App。

### 站点切换

将鼠标移至当前站点位置，单击待选择的站点，即可进行站点切换。



## 4 项目管理

### 4.1 创建项目

项目是用来对数据进行隔离，以确保客户的数据安全。建议以客户的名称来创建。

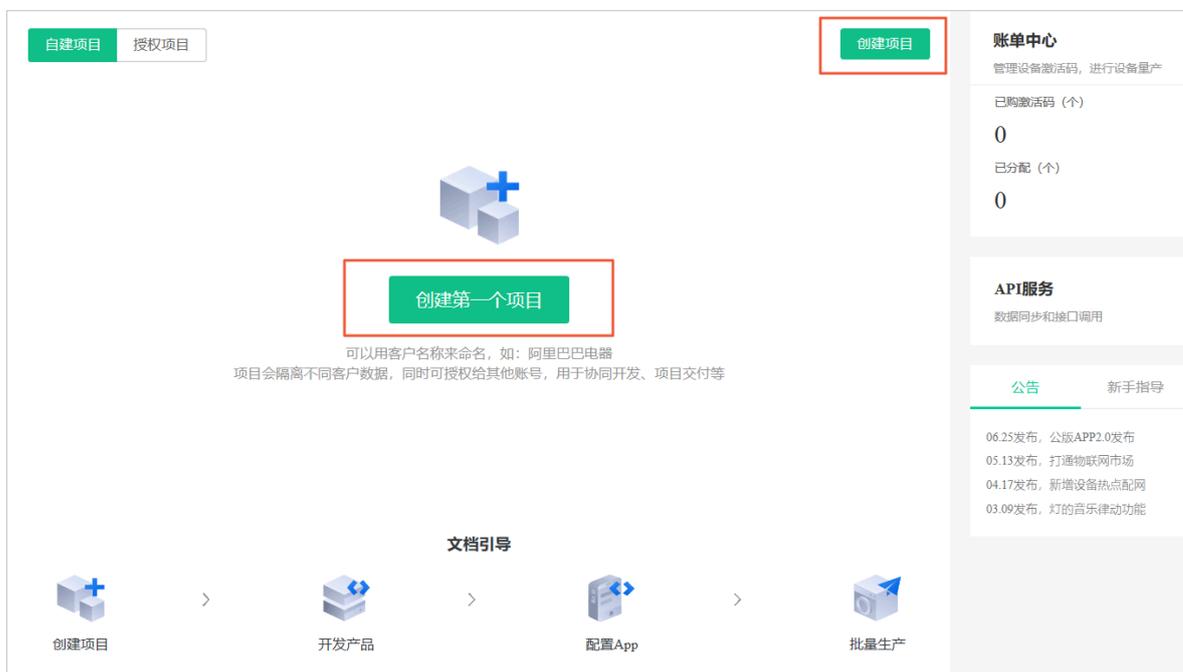
#### 背景信息

每个项目具有唯一ID，可以独立授权给其他阿里云账号，可以用于项目的多方开发协作、和运营中心的交付。

登录生活物联网平台后，可以创建新项目，或进入被授权的项目。

#### 操作步骤

1. 登录生活物联网平台的控制台。
2. 选择对应的工作站点。默认为中国站。
3. 选择自建项目，单击创建项目或创建第一个项目。



说明:

创建第一个项目仅在还未创建项目的时候出现。当已存在项目时，只出现创建项目。

#### 4. 配置项目名称。

新建项目 ×

---

**名称**

某客户的项目 6/20

**数据节点**

设备可在中国大陆任意地区使用，自动连接最近的节点：华东2

---

确定 取消

#### 5. 单击确定完成。

#### 预期结果

项目成功创建后，可在项目中创建产品，和创建自有品牌App，并可以实现App对设备的智能控制和联动。

**某客户的项目**

ID: a124QJAJNJBcC6OR

设备产品	App	成员数
0	0	1

创建时间：2019-07-10

## 4.2 授权项目

当您的项目需要其他成员共同管理时，需要用到项目授权功能。

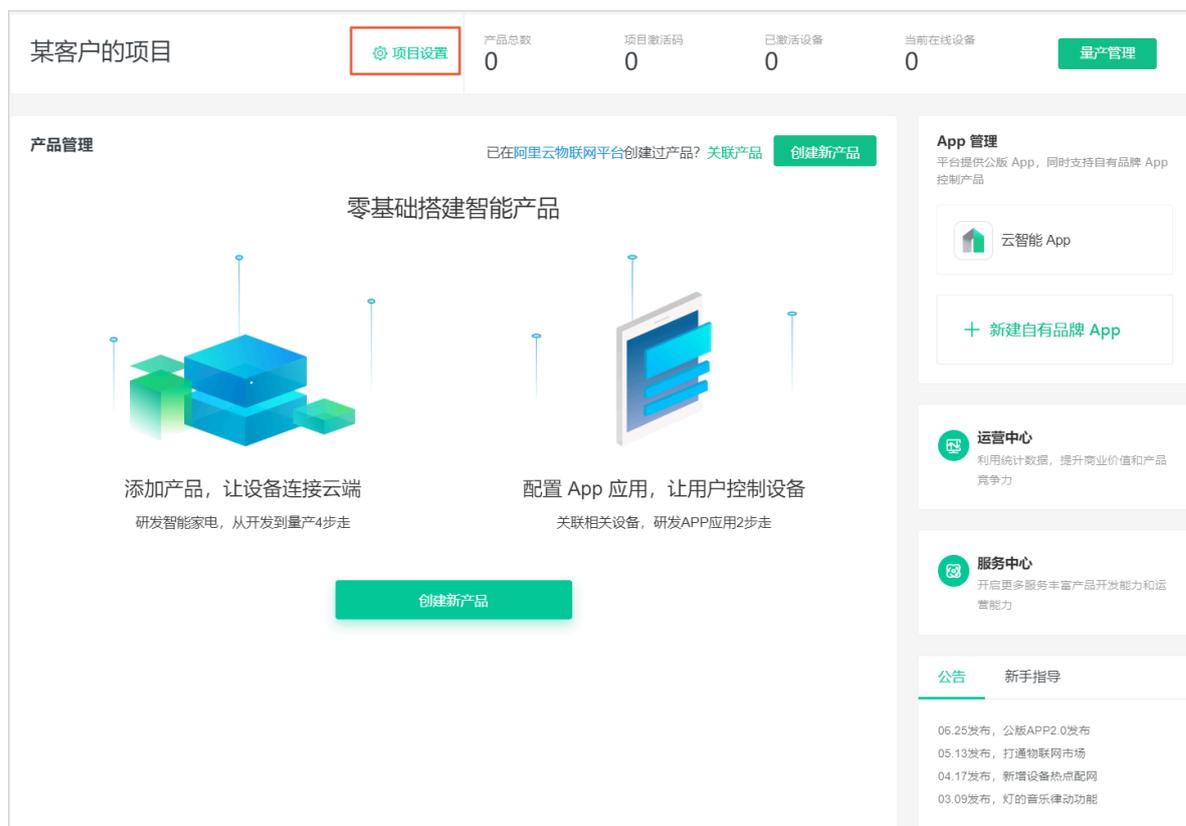
#### 背景信息

项目以下部分可以独立授权给其他角色的阿里云账号，具体介绍如下。

独立授权功能	适合授权情景	授权功能
开发中心	适合多方协作开发，可以分享开发中心给多个成员	开发中心主要包括了产品开发、App 开发、云端管理和量产管理。
运营中心	适合直接交付给客户，如设备厂商，也可以更改运营中心的标题、logo等定制化样式	运营中心提供了一个完整的SaaS服务，包括设备运维、用户运营需要的基本功能，如设备列表和日志、固件升级、数据图表等。可以用来直接交付给智能设备的运营方使用。
量产中心	可单独分配激活码给管理者或产线人员使用，以免误操作修改其他产品信息	主要用于消耗激活码、生成激活码和下载激活码等。

## 操作步骤

1. 在项目主页面，单击项目设置。



2. 在项目设置 > 成员管理中，单击添加成员。

### 3. 配置授权的阿里云账号，勾选授权范围，并单击确定完成。

#### 添加成员 ×

项目成员账号：

[? 如何成为阿里云账号](#)

备注：

权限范围：

- 全选
- 运营中心（设备运维、用户运营）
- 开发中心（产品开发、APP开发、服务中心）
- 量产中心（量产管理）

#### 预期结果

使用被授权的账号登录平台，在授权项目中可以看到该授权项目和权限范围。

### 项目5

ID: a124m8TDuorHebm5 [复制ID](#)

设备产品	App	成员数
7	1	2

[开发中心](#) [运营中心](#)

## 4.3 重命名项目

修改项目名称。

### 背景信息

仅支持对自己创建的项目进行重命名操作，且新的项目名称不可以与已有项目名称重复。

### 操作步骤

1. 在生活物联网平台主页面上，单击需要重命名项目对应的 。

2. 输入新的项目名称。

#### 重命名

名称

 6/20

#### 数据节点

设备可在中国大陆任意地区使用，自动连接最近的节点：华东2

[确定](#) [取消](#)

3. 单击确定完成。

## 4.4 删除项目

当创建的项目不需要时，可进行删除操作。

### 背景信息

如果项目中包含产品、App等信息，删除项目的时候，相关数据会一并被删除。

删除项目不可逆，请谨慎操作。

### 操作步骤

1. 在生活物联网平台主页面上，单击待删除项目对应的



2. 在弹出的提示框中，单击确定。



### 3. 进行身份验证。

可通过手机获取短信验证码方式进行验证，或通过手机阿里云App扫描二维码进行人脸识别验证。



根据提示进行身份验证。验证通过后，即可删除该项目。

## 4.5 设置设备数据同步

当生活物联网平台与第三方企业服务器之间需要进行数据传输时（如身份认证、消息接收等），需要通过HTTP/2 SDK实现数据同步。

### 背景信息

数据同步通过HTTP/2推送方式，原理图如下。



#### 说明:

设备数据HTTP/2方式推送是尽量但不保证送达，不会进行重传，因此会出现一定概率的数据丢失。

### 操作步骤

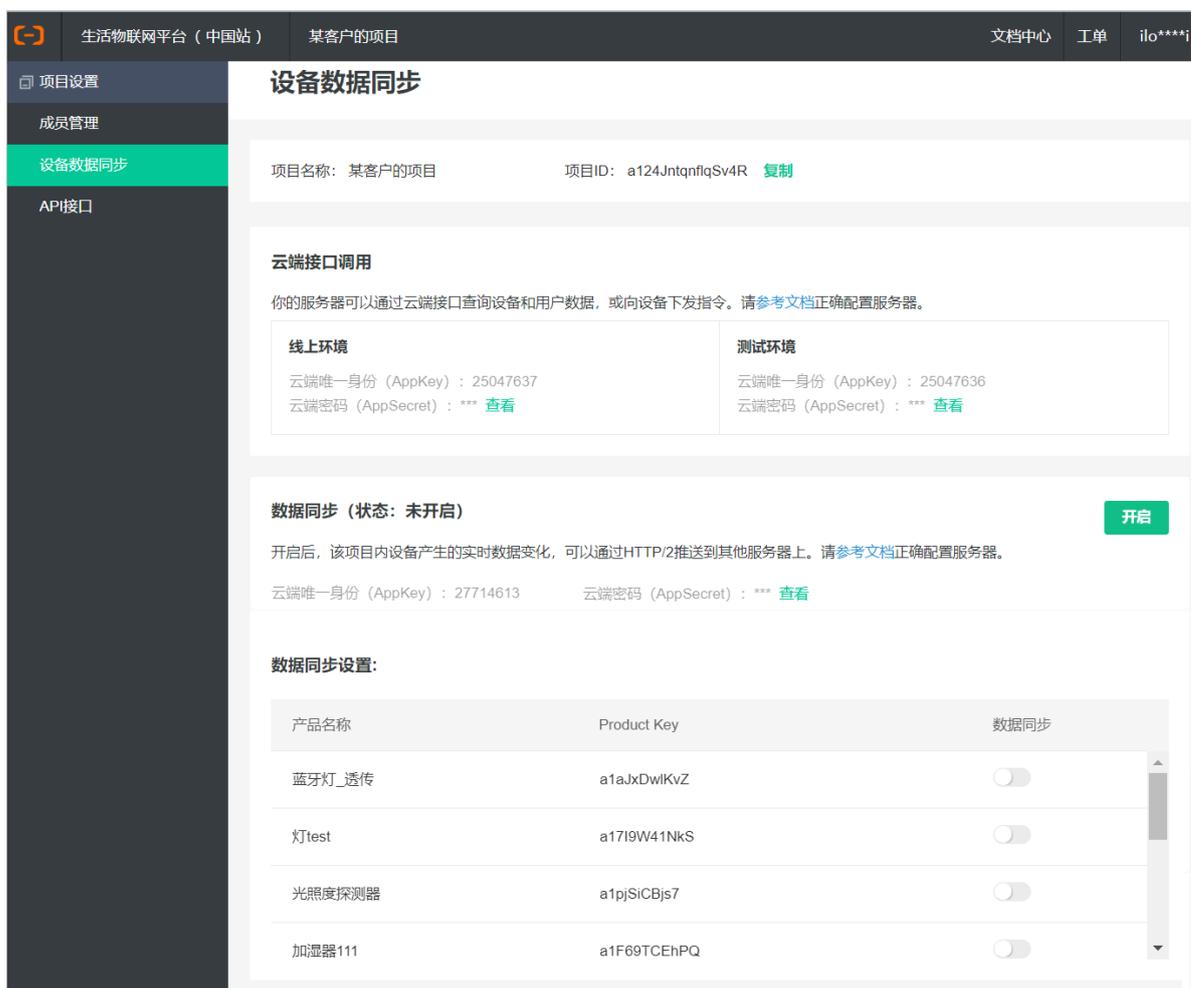
1. 登录生活物联网平台，单击项目对应的



或者单击项目名称，进入项目主页面，单击项目设置。



2. 选择设备数据同步。
3. 单击开启，打开数据同步的开关。



4. 选择数据同步的产品名称。

开启数据同步后，所有的产品默认都打开数据同步。用户可根据需要，关闭无需数据同步的产品。

#### 后续步骤

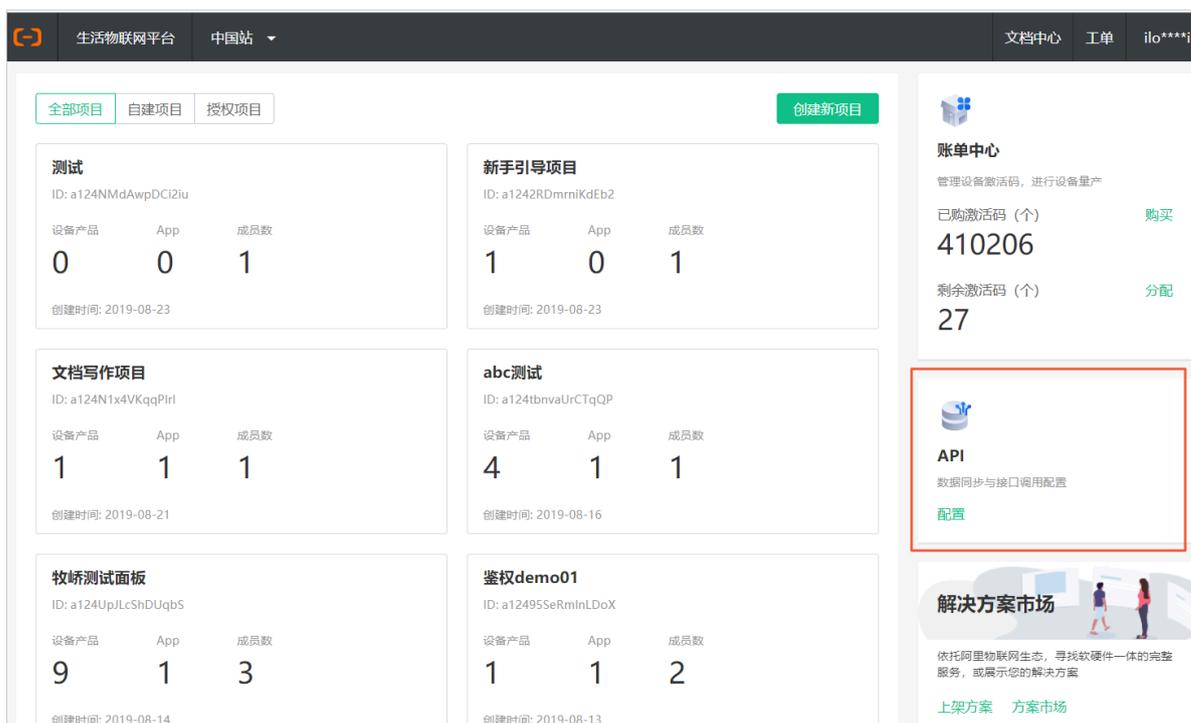
更多操作，请参见[#unique\\_14](#)。

## 4.6 设置用户数据同步

当生活物联网平台调用第三方云提供的URL接口，实现实时上报设备相关数据时（如用户绑定、解绑数据等），可以通过HTTP/HTTPS协议的POST方法实现用户数据同步。

### 操作步骤

1. 在生活物联网平台主页面，单击API对应的配置。



2. 选择用户数据。



3. 单击开启。

#### 4. 在用户数据中，单击新增地址。



#### 5. 配置云端URL地址。

### 后续步骤

更多操作，请参见[#unique\\_16](#)。

## 4.7 查看API接口

可以通过生活物联网平台查看API接口，以及各接口对应的功能和服务开通状态等。

### 背景信息

查看API接口的控制台入口：

- 项目设置
- API服务

通过项目设置页面进入

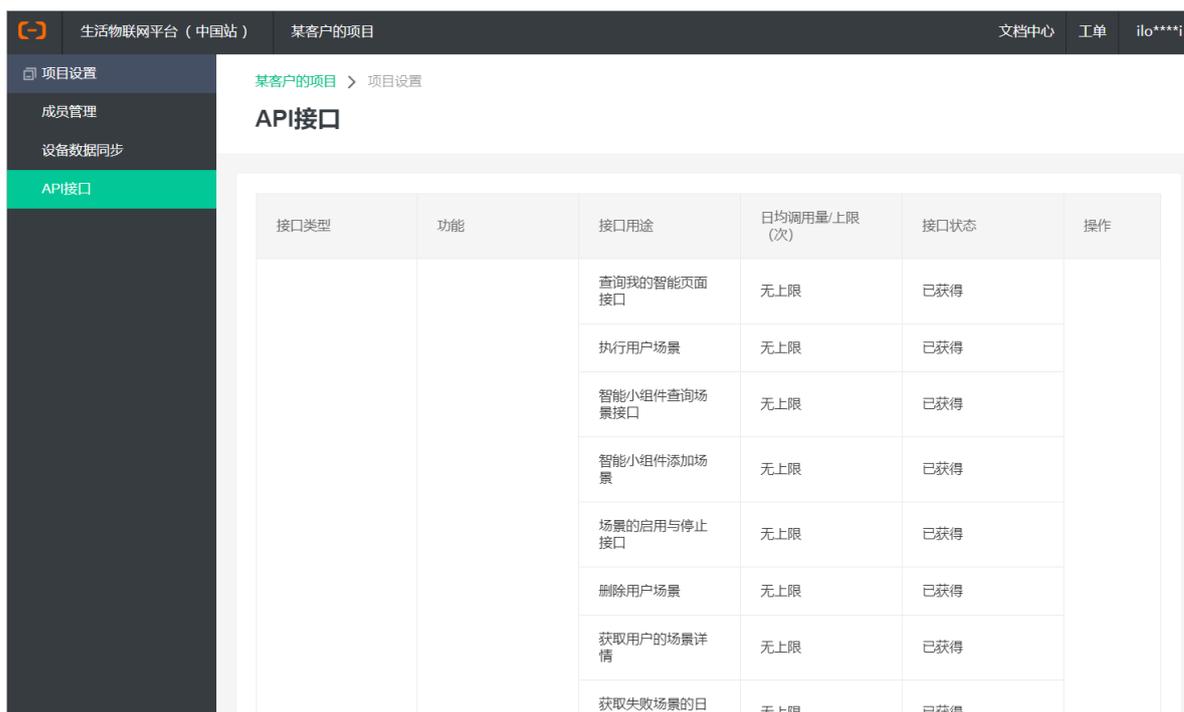
1. 登录生活物联网平台，单击项目对应的



或者单击项目名称，进入项目主页面，单击项目设置。

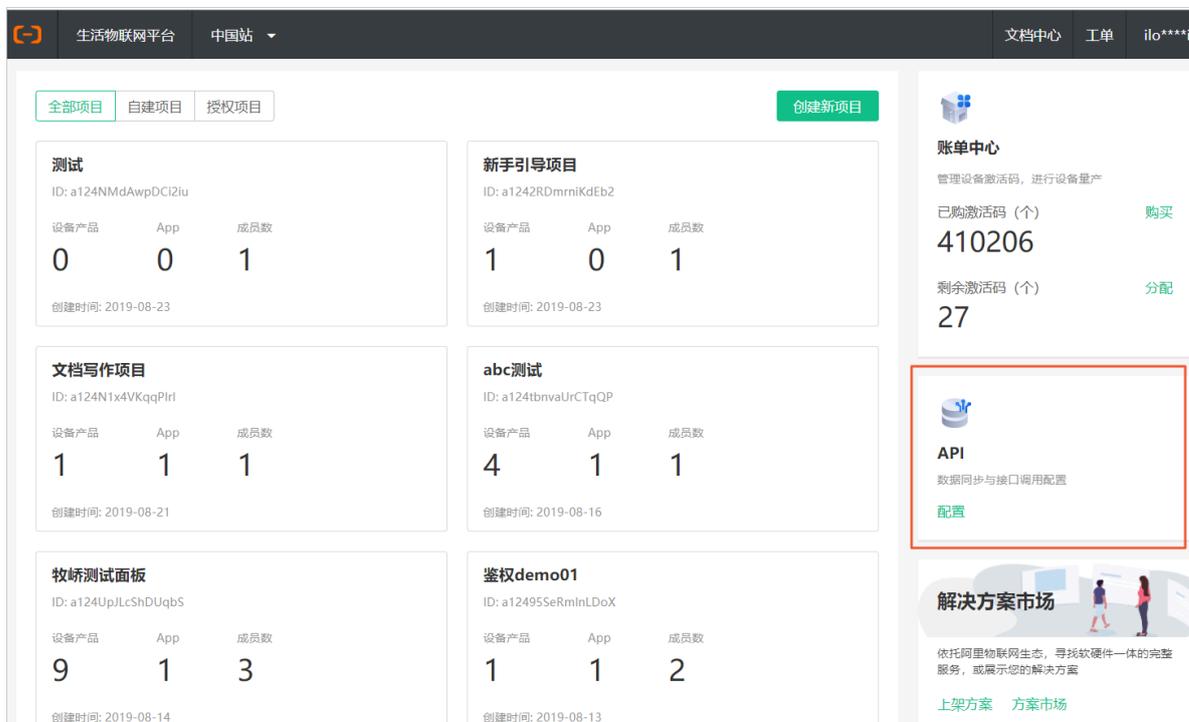


## 2. 选择API接口。



通过API服务页面进入

### 1. 在生活物联网平台主页面，单击API对应的配置。



## 2. 选择设备数据与API。

**设备数据与API**

平台不再提供通过HTTP的方式同步设备数据，请前往每个项目中，配置HTTP/2的方式同步您的设备数据。原先已使用HTTP的方式推送的客户不受影响，可继续使用。

**项目列表**

项目名称	产品数量	创建时间	操作
测试	0	2019-08-23 18:34:29	<a href="#">设备数据同步</a> <a href="#">API接口</a>
新手引导项目	1	2019-08-23 10:35:46	<a href="#">设备数据同步</a> <a href="#">API接口</a>
文档写作项目	1	2019-08-21 14:14:35	<a href="#">设备数据同步</a> <a href="#">API接口</a>
abc测试	4	2019-08-16 11:19:11	<a href="#">设备数据同步</a> <a href="#">API接口</a>
牧桥测试面板	9	2019-08-14 13:51:33	<a href="#">设备数据同步</a> <a href="#">API接口</a>
鉴权demo01	1	2019-08-13 16:32:52	<a href="#">设备数据同步</a> <a href="#">API接口</a>
鉴权demo	1	2019-08-13 16:29:18	<a href="#">设备数据同步</a> <a href="#">API接口</a>
魔智测试	1	2019-08-12 17:21:11	<a href="#">设备数据同步</a> <a href="#">API接口</a>

## 3. 单击项目名称对应的API接口。

**API接口**

接口类型	功能	接口用途	日均调用量/上限(次)	接口状态	操作
		查询我的智能页面接口	无上限	已获得	
		执行用户场景	无上限	已获得	
		智能小组件查询场景接口	无上限	已获得	
		智能小组件添加场景	无上限	已获得	
		场景的启用与停止接口	无上限	已获得	
		删除用户场景	无上限	已获得	
		获取用户的场景详情	无上限	已获得	
		获取失败场景的日	无上限	已获得	

## 5 产品管理

---

### 5.1 新建产品

每个项目中，可以新建多个产品，一个产品代表一种设备型号。

#### 操作步骤

1. 在项目主页面单击创建新产品。

## 2. 配置产品参数。

产品信息 -----

\* 产品名称

\* 所属分类 ?

功能定义

节点类型 -----

\* 节点类型

设备     网关 ?

\* 是否接入网关

是     否

连网与数据 -----

\* 连网方式

\* 数据格式

 ?

更多信息 ----- ∨

取消
完成

参数名称	参数解释
产品名称	填写产品名称，后期可修改。
所属分类	选择产品所属的行业和品类。可以单击功能定义，预览该品类定义的标准功能。

参数名称	参数解释
节点类型	<ul style="list-style-type: none"> <li>· 设备</li> <li>· 网关</li> </ul>
是否接入网关	是否需要通过网关入网。
连网方式	当是否接入网关配置为否时出现该参数，设备连网方式： <ul style="list-style-type: none"> <li>· WiFi</li> <li>· 蜂窝（2G, 3G, 4G）</li> <li>· 以太网</li> <li>· 其他</li> </ul>
接入网关协议	当是否接入网关配置为是时出现该参数，该产品下设备作为子设备与网关的通讯协议类型： <ul style="list-style-type: none"> <li>· 自定义</li> <li>· Modbus</li> <li>· OPC UA</li> <li>· ZigBee</li> <li>· BLE</li> </ul>
数据格式	设备上下行的数据格式，两种格式二选一，不能混合使用。 <ul style="list-style-type: none"> <li>· ICA标准数据格式（Alink JSON）：是阿里云IoT为开发者提供的设备与云端的标准数据交换协议，采用JSON格式。</li> <li>· 透传/自定义：使用自定义的串口数据格式，则需要编写自定义格式数据转换为Alink JSON的脚本（参见<a href="#">透传/编辑脚本</a>），并在云端配置数据解析脚本。</li> </ul>
产品描述	选填，可用于填写产品描述信息。

3. 单击完成。

## 5.2 关联产品

在生活物联网平台中，除了直接新建产品，还可以通过关联，将物联网平台的产品转换为生活物联网平台的产品。

### 前提条件

已经在物联网平台中[../SP\\_67/DNLKIT1868664/ZH-CN\\_TP\\_12827\\_V19.dita#task\\_lxd\\_pnl\\_vdb](#)，例如：产品123。

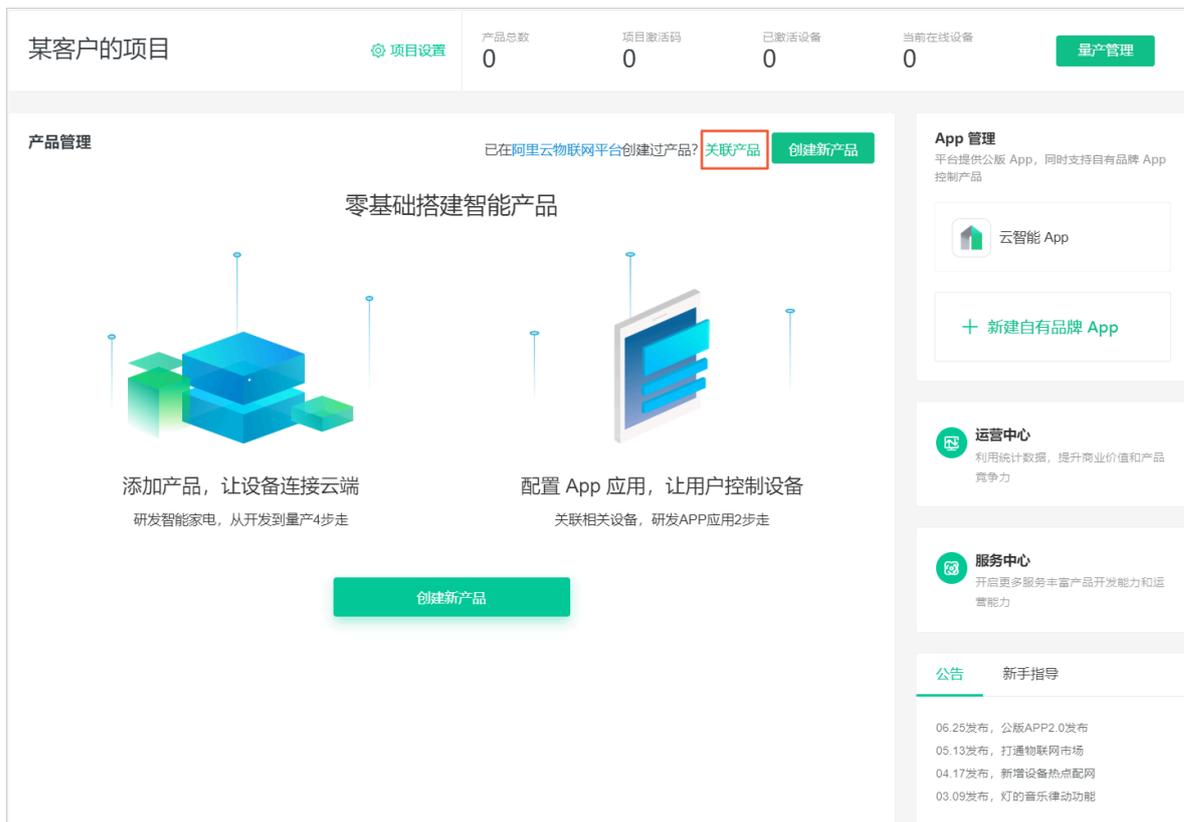
### 背景信息

关联产品将切换收费模式，添加成功后原物联网平台将停止计费。操作不可恢复，需谨慎。

- 生活物联网平台：按每台设备一个激活码收费 [查看详情](#)
- 物联网平台：按日活设备数量收费 [././SP\\_67/DNLKIT1863380/ZH-CN\\_TP\\_803573\\_V1.dita#concept\\_995208](https://sp.67/dnlkit1863380/zh-cn_tp_803573_v1.dita#concept_995208)

### 操作步骤

1. 在项目主页面，单击关联产品。



2. 选择待关联的产品，例如：产品123。

关联物联网平台产品 ×

选择关联产品：

产品A123 ▼

产品信息：

所属分类：灯  
节点类型：设备  
通讯方式：WIFI  
数据格式：Alink JSON  
Product Key：a1xlmXgDLcl  
创建时间：2019-04-23

关联产品将切换收费模式，本平台要求为每台设备购买激活码；  
添加成功后原物联网平台将停止计费。

产品已添加设备数：0  
项目可用激活码数：0

继续 取消



说明：

只有物联网平台中产品品类为“智能生活”的产品才会出现在关联产品列表里。

### 3. 单击继续，弹出待确认的对话框。



### 4. 单击确定完成。

## 5.3 复制产品

如果已有现成的产品，可以直接对其进行复制。

### 前提条件

自建项目中或被授权项目中已存在产品。例如：电工照明插座。

### 背景信息

相同站点下，可以对其他项目中的产品或被授权项目中的产品进行复制。

复制产品不可以跨站点进行操作。

### 操作步骤

1. 进入待复制产品所在的项目。
2. 选择待复制的产品，单击 。

### 3. 选择新产品所在的项目。

可选择现有项目或新建一个项目。

复制产品 ×

---

**将复制产品到**

现有项目     新项目

▼

复制产品到现有项目，需要拥有目标项目产品创建权限

复制产品 ×

---

**将复制产品到**

现有项目     新项目

0/20

创建一个新的项目并复制该产品

### 4. 单击确定完成。

#### 预期结果

产品复制成功后，会出现在产品管理页面中，且产品的名称包含“-复制产品”。

### 某客户的项目

项目设置

产品总数	项目激活码	已激活设备
3	0	0

---

#### 产品管理

已在阿里云物联网平台创建过产品? [关联产品](#) 创建新产品



#### 智能机器人

开发中 WiFi 扫地机器人

ProductKey: a1aZe1reREG

创建时间: 2019-07-23



#### 智能灯123

开发中 WiFi 灯

ProductKey: a1CPv6nDARk

创建时间: 2019-07-23



#### 电工照明插座-复制产品

开发中 WiFi 插座

ProductKey: a1AwdKEVzUs

创建时间: 2019-07-23

## 5.4 修改产品

可以对已有产品的相关信息进行修改，包括产品名称、节点类型、通讯方式和数据格式。

### 操作步骤

1. 在项目主页面，单击待修改的产品。

## 2. 单击产品信息中基本信息对应的编辑。

### 智能灯123

更新时间: 2019-07-23 13:41:04

**基本信息** [编辑](#)

所属分类: 灯  
节点类型: 设备  
通讯方式: WIFI  
数据格式: ICA标准数据格式 (推荐)  
Product Key: a1CPv6nDArk  
Product Secret: \*\*\*\* [显示](#)  
Product Id: 2265370

**模组** [重选](#)

品牌: 未认证  
型号: 未认证

创建时间: 2019-07-23

### 3. 修改产品信息。

产品参数查看[参数介绍](#)。

编辑智能灯123 ×

---

**产品名称**

**节点类型**

设备     网关 ?

**通讯方式**

WIFI     蜂窝(2/3/4G)     蓝牙BLE     ZigBee     以太网     其它 ?

**数据格式**

ICA标准数据格式 (推荐)     透传/自定义格式

---

### 4. 单击确定完成。

## 5.5 删除产品

当产品不再需要时，可进行删除操作。

### 背景信息

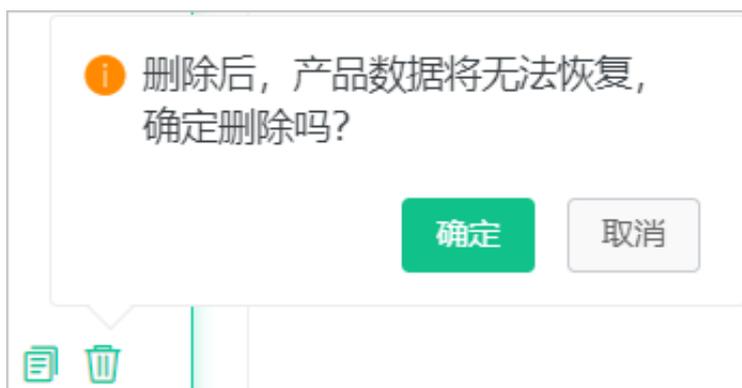
只可以对状态为开发中的产品进行删除。状态为已发布的产品不可删除。此时若要删除产品，需先将产品状态撤回到开发中，再进行删除。

删除后，产品数据无法恢复，请谨慎操作。

### 操作步骤

1. 选择待删除产品所在的项目。
2. 选择待删除的产品，单击 。

3. 在弹出的提示框中，单击确定。



4. 进行身份验证。

可选择通过手机获取短信验证码方式进行验证，或通过手机阿里云App扫描二维码进行人脸识别验证。



验证通过后，即可删除该产品。

## 6 开发中心

### 6.1 定义产品功能

#### 6.1.1 功能概述

为产品定义功能，即定义一套物的模型。

在每个产品中，都需要定义一套物的模型，具体查看 [物模型概述](#)。

每个产品有以下两类功能：

功能类型	描述
标准功能	根据ICA数据标准，平台为每个品类创建了默认的标准功能。
自定义功能	如果标准功能不符合您的设备特征，您也可以自定义功能。

功能的组成分为三要素：属性、事件、服务。

功能三要素	描述
属性	用于描述设备运行时的状态，支持状态读取和设置，如开关、温度、颜色等。
服务	用于实现更复杂的业务逻辑，可包含输入参数和输出参数，可用于让设备执行某项特定的任务。如烤箱的烘焙功能中，需要设定时间和温度等。
事件	用于描述设备运行时的事件，如瞬时性的通知信息，也可包含多个输出参数。如人体红外传感器识别到有人经过、空调上报了故障等。

#### 6.1.2 新增标准功能

当系统提供的默认标准功能没法满足使用时，可新增标准功能。新增后的标准功能为可选功能。

##### 操作步骤

1. 在项目主页面产品管理中，选择产品，进入产品-功能定义页面。

## 2. 单击标准功能对应的添加功能。

功能定义

标准功能 ? 导入物模型 查看物模型 添加功能

功能类型	功能名称	标识符	数据类型	数据定义	操作
属性	主灯开关 <span>必选</span>	LightSwitch	bool (布尔型)	布尔值: 关闭 - 0; 开启 - 1;	<span>编辑</span>
事件	故障上报 <span>必选</span>	Error	-	事件类型: 故障	<span>编辑</span>

自定义功能 ? 添加功能

功能类型	功能名称	标识符	数据类型	数据定义	操作
------	------	-----	------	------	----

  
无自定义功能

- 单击左侧选择功能区域中的功能类型。  
系统会自动添加到右侧已选功能区域中。



除了可以选择本类型的功能，还可以通过搜索添加跨品类的功能。



- 单击确定完成。

### 6.1.3 修改标准功能

当标准功能中的设置不符合预期，可以对功能的内容进行修改。

#### 背景信息

可选标准功能和必标准选功能都支持修改。

#### 操作步骤

1. 进入产品-功能定义页面。
2. 单击标准功能中对应数据的编辑。



3. 修改参数，并单击确定完成。

功能类型中部分参数不支持修改（灰色显示部分），根据具体的功能类型来修改。单击此处[查看参数介绍](#)。

### 6.1.4 删除标准功能

新增的标准功能不再需要的时候，可以删除标准功能。

#### 背景信息

只有可选的标准功能支持修改。

产品状态如果是已发布，所有的标准功能都不支持删除。

#### 操作步骤

1. 进入产品-功能定义页面。

## 2. 单击标准功能中对应数据的删除。

功能定义

标准功能 ● 导入物模型 查看物模型 添加功能

功能类型	功能名称	标识符	数据类型	数据定义	操作
属性	主灯开关 <span>必选</span>	LightSwitch	bool (布尔型)	布尔值: 关闭 - 0; 开启 - 1;	编辑
属性	环境湿度 <span>可选</span>	EnvironmentH...	float (单精度浮点型)	取值范围: 0 ~ 100	编辑 <span>删除</span>
事件	故障上报 <span>必选</span>	Error	-	事件类型: 故障	编辑

## 3. 在弹出的对话框中，单击确定完成。

### 6.1.5 新增自定义功能

当标准功能没法满足需要的时候，可以根据自身业务自定义功能。

#### 操作步骤

1. 进入产品-功能定义页面。
2. 单击自定义功能对应的添加功能。

自定义功能 ● 添加功能

功能类型	功能名称	标识符	数据类型	数据定义	操作
 无自定义功能					

### 3. 配置自定义功能参数。

您可为产品自定义属性、服务和事件。

- 自定义属性。在添加自定义功能页面，选择功能类型为属性。设置参数完成后，单击确认。

#### 添加自定义功能

\* 功能类型：  
 属性  服务  事件

\* 功能名称：  
工作电压

\* 标识符：  
LightVolt

\* 数据类型：  
float (单精度浮点型)

\* 取值范围：  
0 ~ 4

\* 步长：  
0.1

单位：  
伏特 / V

读写类型：  
 读写  只读

描述  
显示设备电压；电参数采用4个字节浮点型数据

21/100

属性参数设置如下：

参数	描述
功能名称	<p>属性的名称，如用电量。同一产品下功能名称不能重复。</p> <p>支持中文、大小写字母、数字、短划线和下划线，且必须以中文、英文或数字开头，不超过30个字符。</p> <p>如果您创建产品时选择了功能模板，输入功能名称时，将从标准功能库中筛选匹配的标准属性，供您选择。</p> <p> <b>说明:</b> 当接入网关协议为Modbus时，不支持标准属性，仅支持自定义属性。</p>
标识符	<p>属性唯一标识符，在产品中具有唯一性。即 Alink JSON 格式中的 identifier 的值，作为设备上报该属性数据的 Key，云端根据该标识符校验是否接收数据。可包含英文、数字、下划线，长度不超过50个字符，如 PowerConsumption。</p> <p> <b>说明:</b> 不能用以下系统保留参数作为标识符： set、get、post、time、value。</p>
数据类型	<ul style="list-style-type: none"> <li>- int32: 32位整型。需定义取值范围、步长和单位符号。</li> <li>- float: 单精度浮点型。需定义取值范围、步长和单位符号。</li> <li>- double: 双精度浮点型。需定义取值范围、步长和单位符号。</li> <li>- enum: 枚举型。定义枚举项的参数值和参数描述，如 1-加热模式、2-制冷模式。</li> <li>- bool: 布尔型。采用 0 或 1 来定义布尔值，如 0-关、1-开。</li> <li>- text: 字符串。需定义字符串的数据长度，最长支持 2048 字节。</li> <li>- date: 时间戳。格式为 string 类型的 UTC 时间戳，单位：毫秒。</li> <li>- struct: JSON对象。定义一个 JSON 结构体，新增 JSON 参数项，如定义灯的颜色是由 Red、Green、Blue 三个参数组成的结构体。不支持结构体嵌套。</li> <li>- array: 数组。需声明数组内元素的数据类型，可选择int32、float、double、text或struct。需确保同一个数组元素类型相同。数组内可包含1-128 个元素。</li> </ul> <p> <b>说明:</b> 当设备协议为Modbus时，无需设置该参数。</p>

参数	描述
步长	<p>属性值和事件以及服务中输入输出参数值变化的最小粒度。数据类型为int32、float、double时，需要根据您的业务需要设置步长。</p> <p>例如，为温度计产品定义温度属性时，将数据类型设置为int32，步长为2，单位为°C，取值范围0~100。即温度每变化两度，设备上报温度值，如0°C、2°C、4°C、6°C、8°C……。</p>
单位	单位可选择为无或根据实际情况选择。
读写类型	<ul style="list-style-type: none"> <li>- 读写：请求读写的方法支持 GET（获取）和 SET（设置）。</li> <li>- 只读：请求只读的方法仅支持 GET（获取）。</li> </ul> <div style="background-color: #f0f0f0; padding: 5px; margin-top: 10px;">  <b>说明：</b> 当接入网关协议为Modbus时，无需设置该参数。         </div>
描述	输入文字，对该功能进行说明或备注。长度限制为100字。

参数	描述
扩展描述	<p>设备通信协议到标准物模型的映射关系。</p> <p>产品接入网关协议为自定义、OPC UA或Modbus时，需填写该参数。</p> <ul style="list-style-type: none"> <li>- 接入网关协议为自定义时，填写JSON格式的自定义配置信息，长度限制为1024字符。</li> <li>- 接入网关协议为OPC UA时，设置节点名称。节点名称需保证属性维度下唯一。</li> <li>- 接入网关协议为Modbus时，需设置以下参数： <ul style="list-style-type: none"> <li>■ 操作类型： <ul style="list-style-type: none"> <li>■ 线圈状态（只读，01）</li> <li>■ 线圈状态（读写，读取使用01，写入使用05）</li> <li>■ 线圈状态（读写，读取使用01，写入使用0F）</li> <li>■ 离散量输入（只读，02）</li> <li>■ 保持寄存器（只读，03）</li> <li>■ 保持寄存器（读写，读取使用03，写入使用06）</li> <li>■ 保持寄存器（读写，读取使用03，写入使用10）</li> <li>■ 输入寄存器（只读，04）</li> </ul> </li> <li>■ 寄存器地址：十六进制，必须以0x开头，且限制范围是0x0~0xFFFF。例如，0xFE。</li> <li>■ 原始数据类型：支持int16、uint16、int32、uint32、int64、uint64、float、double、string、bool、自定义（原始数据）多种数据类型。</li> <li>■ 取值范围：这是原始数据经过缩放因子处理之后的取值范围。不在该取值范围内的数据会被丢弃。物联网平台已为各操作类型设置了默认取值范围： <ul style="list-style-type: none"> <li>■ 线圈状态类型：0~1</li> <li>■ 离散量输入类型：0~1</li> <li>■ 保持寄存器类型：-2147483648 ~ 2147483647</li> <li>■ 输入寄存器类型：-2147483648 ~ 2147483647</li> </ul> </li> <li>■ 交换寄存器内高低字节：是否把寄存器内16位数据的前后8个bits互换。 <ul style="list-style-type: none"> <li>■ true：互换。</li> <li>■ false：不互换。</li> </ul> </li> <li>■ 交换寄存器顺序：是否把原始数据32位数据的bits互换。 <ul style="list-style-type: none"> <li>■ true：互换。</li> <li>■ false：不互换。</li> </ul> </li> <li>■ 缩放因子：不能为0，默认为1，可以为负数。</li> <li>■ 采集间隔：数据采集间隔，单位ms，不能小于10。</li> <li>■ 数据上报方式：可选按时上报和变更上报。</li> </ul> </li> </ul>

- 自定义服务。在添加自定义功能页面，选择功能类型为服务。设置参数完成后，单击确认。



说明:

接入网关的协议选择为Modbus时，不支持定义服务。

### 添加自定义功能

\* 功能类型：  
 属性  服务  事件

\* 功能名称：  
自动喷灌

\* 标识符：  
SetTimerTask

\* 调用方式：  
 异步  同步

输入参数：

<input type="checkbox"/> 参数名称：喷灌时间	编辑 删除
<input type="checkbox"/> 参数名称：喷灌量	编辑 删除

+增加参数

输出参数：

<input type="checkbox"/> 参数名称：土壤湿度	编辑 删除
------------------------------------	-------

+增加参数

描述  
请输入描述  
0/100

服务参数设置如下：

参数	描述
功能名称	<p>服务名称。</p> <p>支持中文、大小写字母、数字、短划线和下划线，且必须以中文、英文或数字开头，不超过30个字符。</p> <p>如果您创建产品时选择了功能模板，输入功能名称时，将从标准功能库中筛选匹配的标准服务，供您选择。</p> <p> <b>说明:</b> 当接入网关协议为Modbus时，不支持自定义服务。</p>
标识符	<p>服务唯一标识符，在产品下具有唯一性。即 Alink JSON 格式中该服务的 identifier 的值。可包含英文、数字、和下划线，长度不超过30个字符。</p> <p> <b>说明:</b> 不能用以下系统保留参数作为标识符： set、get、post、time、value。</p>
调用方式	<ul style="list-style-type: none"> <li>- 异步：服务为异步调用时，云端执行调用后直接返回结果，不会等待设备的回复消息。</li> <li>- 同步：服务为同步调用时，云端会等待设备回复；若设备没有回复，则调用超时。</li> </ul>
输入参数	<p>设置该服务的入参，可选。</p> <p>单击新增参数，在弹窗对话框中添加服务入参。</p> <p>当接入网关协议为 OPC UA 时，需设置参数索引，用于标记参数的顺序。</p> <p> <b>说明:</b></p> <ul style="list-style-type: none"> <li>- 不能用以下系统保留参数作为输入参数的标识符：set、get、post、time、value。</li> <li>- 您可以使用某个属性作为入参，也可以自定义参数。如在定义自动喷灌服务功能时，将已定义的属性喷灌时间和喷灌量作为自动喷灌服务的入参，则调用该参数时传入这两个参数，喷灌设备将按照设定的喷灌时间和喷灌量自动进行精准灌溉。</li> <li>- 一个服务最多支持定义 20 个入参。</li> </ul>

参数	描述
输出参数	<p>设置该服务的出参，可选。</p> <p>单击新增参数，在弹窗对话框中添加服务出参。</p> <p>当接入网关协议为OPC UA时，需设置参数索引，用于标记参数的顺序。</p> <div style="background-color: #f0f0f0; padding: 5px;"> <p> 说明:</p> <ul style="list-style-type: none"> <li>- 不能用以下系统保留参数作为输出参数的标识符：set、get、post、time、value。</li> <li>- 您可以使用某个属性作为出参，也可以自定义参数，如将已定义的属性土壤湿度作为出参，则云端调用自动喷灌服务时，将返回当前土壤湿度的数据。</li> <li>- 一个服务最多支持定义20个出参。</li> </ul> </div>
扩展描述	当接入网关协议为OPC UA时，设置节点名称。节点名称需保证服务维度下唯一。
描述	输入文字，对该服务功能进行说明或备注。长度限制为100字。

- 自定义事件通知。在添加自定义功能页面，选择功能类型为事件。设置参数完成后，单击确认。



说明:

接入网关的协议选择为Modbus时，不支持定义事件。

添加自定义功能
✕

---

**\* 功能类型：**

属性
服务
事件
?

**\* 功能名称：**

故障上报
?

**\* 标识符：**

Error
?

**\* 事件类型：**

信息
  告警
  故障
 ?

输出参数：

参数名称：故障代码
 编辑 删除

参数名称：电压
 编辑 删除

+增加参数

描述

请输入描述

0/100

事件参数设置如下：

参数	描述
功能名称	<p>事件的名称。</p> <p>支持中文、大小写字母、数字、短划线和下划线，且必须以中文、英文或数字开头，不超过30个字符。</p> <div style="background-color: #f0f0f0; padding: 5px; margin-top: 10px;">  <b>说明：</b> 当接入网关协议为Modbus时，不支持自定义事件。         </div>

参数	描述
标识符	<p>事件唯一标识符，在产品下具有唯一性。即 Alink JSON 格式中该事件的 identifier 的值，作为设备上报该事件数据的 Key，如 ErrorCode。</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;"> <p> 说明： 不能用以下系统保留参数作为标识符： set、get、post、time、value。</p> </div>
事件类型	<ul style="list-style-type: none"> <li>- 信息：指设备上报的一般性通知，如完成某项任务等。</li> <li>- 告警：设备运行过程中主动上报的突发或异常情况，告警类信息，优先级高。您可以针对不同的事件类型进行业务逻辑处理和统计分析。</li> <li>- 故障：设备运行过程中主动上报的突发或异常情况，故障类信息，优先级高。您可以针对不同的事件类型进行业务逻辑处理和统计分析。</li> </ul>
输出参数	<p>该事件的出参。单击增加参数，在弹窗对话框中添加一个服务出参。您可以使用某个属性作为出参，也可以自定义参数。如，将已定义的属性电压作为出参，则设备上报该故障事件时，将携带当前设备的电压值，用于进一步判断故障原因。</p> <p>当接入网关协议为 OPC UA 时，需设置参数索引，用于标记参数的顺序。</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;"> <p> 说明：</p> <ul style="list-style-type: none"> <li>- 不能用以下系统保留参数作为输出参数的标识符：set、get、post、time、value。</li> <li>- 一个事件最多支持定义 50 个出参。</li> </ul> </div>
扩展描述	<p>当接入网关协议为 OPC UA 时，设置节点名称。节点名称需保证事件维度下唯一。</p>
描述	<p>输入文字，对该事件功能进行说明或备注。长度限制为 100 字。</p>

## 6.1.6 修改自定义功能

当自定义功能中的设置不符合预期，可以对功能的内容进行修改。

### 操作步骤

1. 进入产品-功能定义页面。
2. 单击自定义功能中对应数据的编辑。
3. 修改参数，并单击确定完成。

功能类型中部分参数不支持修改（灰色显示部分），根据具体的功能类型来修改。单击此处[查看参数介绍](#)。

## 6.1.7 删除自定义功能

新增的自定义功能不再需要时，可以删除该自定义功能。

### 背景信息

产品状态如果是已发布，所有的自定义功能都不支持删除。

### 操作步骤

1. 进入产品-功能定义页面。
2. 单击自定义功能中对应数据的删除。
3. 在弹出的对话框中，单击确定完成。

## 6.1.8 查看并导出物模型

功能定义完成后，平台根据“标准功能”和“自定义功能”，自动生成JSON格式的物模型。

### 前提条件

已完成标准功能和自定义功能的配置。

### 背景信息

物模型是对设备是什么，能做什么的描述，包括设备的属性（properties）、服务（services）、事件（events）等。阿里IoT通过定义一种物的描述语言来描述物模型，称之为 TSL（即 Thing Specification Language）。[更多物模型介绍](#)

### 操作步骤

1. 进入产品-功能定义页面。

## 2. 在产品功能定义中，单击查看物模型。

功能定义

标准功能 ●

导入物模型 查看物模型 添加功能

功能类型	功能名称	标识符	数据类型	数据定义	操作
属性	主灯开关 必选	LightSwitch	bool (布尔型)	布尔值：关闭 - 0；开启 - 1；	编辑
属性	环境湿度 可选	EnvironmentH...	float (单精度浮点型)	取值范围：0 ~ 100	编辑 删除
事件	故障上报 必选	Error	-	事件类型：故障	编辑

查看物模型
✕

物模型是对设备在云端的功能描述，包括设备的属性、服务和事件。物联网平台通过定义一种物的描述语言来描述物模型，称之为 TSL (即 Thing Specification Language)，采用 JSON 格式，您可以根据 TSL 组装上报设备的数据。您可以导出完整物模型，用于云端应用开发；您也可以只导出精简物模型，配合设备端 SDK 实现设备开发。

完整物模型

精简物模型

```

1 {
2   "schema": "https://iotx-tsl.oss-ap-southeast-1.aliyuncs.com/sche
3   "profile": {
4     "productKey": "a1I1sYe1Roi"
5   },
6   "services": [
7     {
8       "outputData": [],
9       "identifier": "set",
10      "inputData": [
11        {
12          "identifier": "LightSwitch",
13          "dataType": {
14            "specs": {
15              "0": "关闭",
16              "1": "开启"
17            },
18            "type": "bool"
19          }
20        }

```

导出模型文件

3. 单击导出模型文件，可以将JSON文件保存到本地。

### TSL字段描述说明

```

{
  "schema": "物的TSL描述schema",
  "link": "云端系统级uri,用来调用服务/订阅事件",
  "profile": {
    "productKey": "产品key",
    "deviceName": "设备名称"
  },
  "properties": [
    {
      "identifier": "属性唯一标识符(产品下唯一)",

```

```

        "name": "属性名称",
        "accessMode": "属性读写类型, 只读(r), 只写(w), 读写(rw)",
        "required": "是否是必选属性",
        "dataType": {
            "type": "属性类型: int(原生), float(原生), double(原生), text(原生), date(String类型UTC毫秒), bool(0或1的int类型), enum(int类型), struct(结构体类型, 可包含前面6种类型)",
            "specs": {
                "min": "属性最小值(int,float,double类型特有)",
                "max": "属性最大值(int,float,double类型特有)",
                "unit": "属性单位",
                "unitName": "单位的名称"
            }
        }
    ],
    "events": [
        {
            "name": "事件名称",
            "identifier": "事件唯一标识符",
            "desc": "事件描述",
            "type": "事件类型(info,alert,error)",
            "required": "是否是必选事件",
            "outputData": [
                {
                    "identifier": "参数唯一标识符",
                    "name": "参数名称",
                    "dataType": {
                        "type": "参数类型: int(原生), float(原生), double(原生), text(原生), date(String类型UTC毫秒), bool(0或1的int类型), enum(int类型), struct(结构体类型, 可包含前面6种类型)",
                        "specs": {
                            "min": "参数最小值(int,float,double类型特有)",
                            "max": "参数最大值(int,float,double类型特有)",
                            "unit": "参数单位",
                            "unitName": "单位的名称"
                        }
                    }
                }
            ]
        },
        {
            "method": "事件对应的方法名称(根据identifier生成)"
        }
    ],
    "services": [
        {
            "name": "服务名称",
            "identifier": "服务唯一标识符",
            "desc": "服务描述",
            "required": "是否是必选服务",
            "inputData": [
                {
                    "identifier": "入参唯一标识符",
                    "name": "入参名称",
                    "dataType": {
                        "type": "入参类型: int(原生), float(原生), double(原生), text(原生), date(String类型UTC毫秒), bool(0或1的int类型), enum(int类型), struct(结构体类型, 可包含前面6种类型)",
                        "specs": {

```

```

    "min": "入参最小值(int,float,double
类型特有)",
    "max": "入参最大值(int,float,double
类型特有)",
    "unit": "入参单位",
    "unitName": "单位的名称"
  }
}
],
"outputData": [
  {
    "identifier": "出参唯一标识符",
    "name": "出参名称",
    "dataType": {
      "type": "出参类型: int(原生), float(原
生), double(原生), text(原生), date(String类型UTC毫秒), bool(0或1的
int类型), enum(int类型), struct(结构体类型, 可包含前面6种类型)",
      "specs": {
        "min": "出参最小值(int,float,double
类型特有)",
        "max": "出参最大值(int,float,double
类型特有)",
        "unit": "出参单位",
        "unitName": "单位的名称"
      }
    }
  }
],
"method": "服务对应的方法名称(根据identifier生成)"
}
]
}

```

### TSL样例

```

{
  "schema": "http://aliyun/iot/thing/desc/schema",
  "link": "/sys/q408EXte2fy/airCondition/thing/",
  "profile": {
    "productKey": "q408EXte2fy",
    "deviceName": "airCondition"
  },
  "properties": [
    {
      "identifier": "fan_doule_property",
      "name": "风扇双精度型属性",
      "accessMode": "r",
      "required": true,
      "dataType": {
        "type": "double",
        "specs": {
          "min": "0.0",
          "max": "100.0",
          "unit": "g/ml",
          "unitName": "毫升"
        }
      }
    },
    {
      "identifier": "fan_int_property",
      "name": "风扇整数型属性",
      "accessMode": "rw",

```

```
        "required": true,
        "dataType": {
          "type": "int",
          "specs": {
            "min": "0",
            "max": "100",
            "unit": "g/ml",
            "unitName": "毫升"
          }
        }
      },
      {
        "identifier": "batch_enum_attr_id",
        "name": "风扇枚举型属性",
        "accessMode": "r",
        "required": true,
        "dataType": {
          "type": "enum",
          "specs": {
            "0": "one",
            "1": "two",
            "2": "three"
          }
        }
      },
      {
        "identifier": "fan_float_property",
        "name": "风扇浮点型测试",
        "accessMode": "r",
        "required": true,
        "dataType": {
          "type": "float",
          "specs": {
            "min": "0.0",
            "max": "100.0",
            "unit": "g/ml",
            "unitName": "毫升"
          }
        }
      },
      {
        "identifier": "fan_text_property",
        "name": "风扇字符型属性",
        "accessMode": "r",
        "required": true,
        "dataType": {
          "type": "text",
          "specs": {
            "length": "64",
            "unit": "g/ml",
            "unitName": "毫升"
          }
        }
      },
      {
        "identifier": "fan_date_property",
        "name": "风扇时间型属性",
        "accessMode": "r",
        "required": true,
        "dataType": {
          "type": "date",
          "specs": {}
        }
      }
    ],
  },
```

```

    {
      "identifier": "batch_boolean_attr_id",
      "name": "风扇布尔型属性",
      "accessMode": "r",
      "required": true,
      "dataType": {
        "type": "bool",
        "specs": {
          "0": "close",
          "1": "open"
        }
      }
    },
    {
      "identifier": "fan_struct_property",
      "name": "风扇结构型属性",
      "accessMode": "r",
      "required": true,
      "dataType": {
        "type": "struct",
        "specs": [
          {
            "identifier": "fan_struct_property_
float_child",
            "name": "风扇双精度型属性",
            "dataType": {
              "type": "double",
              "specs": {
                "min": "0.0",
                "max": "100.0",
                "unit": "g/ml",
                "unitName": "毫升"
              }
            }
          },
          {
            "identifier": "fan_struct_property_
float_child",
            "name": "风扇结构型属性浮点子属性",
            "dataType": {
              "type": "float",
              "specs": {
                "min": "0.0",
                "max": "255.0",
                "unit": "°",
                "unitName": "度"
              }
            }
          },
          {
            "identifier": "fan_struct_property_
int_child",
            "name": "风扇结构型属性整数子属性",
            "dataType": {
              "type": "int",
              "specs": {
                "min": "0",
                "max": "255",
                "unit": "°",
                "unitName": "度"
              }
            }
          }
        ]
      }
    },
  ],
}

```

```

        "identifier": "fan_struct_property_
text_child",
        "name": "风扇结构型属性字符子属性",
        "dataType": {
            "type": "text",
            "specs": {
                "length": "200",
                "unit": "unit",
                "unitName": "单位"
            }
        }
    },
    {
        "identifier": "fan_struct_property_
date_child",
        "name": "风扇结构型属性时间子属性",
        "dataType": {
            "type": "date",
            "specs": {}
        }
    },
    {
        "identifier": "fan_struct_property_
enum_child",
        "name": "风扇结构型属性枚举子属性",
        "dataType": {
            "type": "enum",
            "specs": {
                "0": "one",
                "1": "two",
                "2": "three"
            }
        }
    },
    {
        "identifier": "fan_struct_property_
boolean_child",
        "name": "风扇结构型属性布尔子属性",
        "dataType": {
            "type": "bool",
            "specs": {
                "0": "close",
                "1": "open"
            }
        }
    }
]
},
"events": [
    {
        "name": "alarm",
        "identifier": "alarm",
        "desc": "警报",
        "type": "info",
        "required": true,
        "outputData": [
            {
                "identifier": "errorCode",
                "name": "错误码",
                "dataType": {
                    "type": "text",
                    "specs": {

```

```

        "length": "255",
        "unit": ""
    }
}
],
"method": "thing.event.alarm.post"
},
{
    "identifier": "post",
    "name": "post",
    "type": "info",
    "required": true,
    "desc": "属性上报",
    "method": "thing.event.property.post",
    "outputData": [
        {
            "identifier": "fan_double_property",
            "name": "风扇双精度型属性",
            "dataType": {
                "type": "double",
                "specs": {
                    "min": "0.0",
                    "max": "100.0",
                    "unit": "g/ml",
                    "unitName": "毫升"
                }
            }
        },
        {
            "identifier": "fan_int_property",
            "name": "风扇整数型属性",
            "dataType": {
                "type": "int",
                "specs": {
                    "min": "0",
                    "max": "100",
                    "unit": "g/ml",
                    "unitName": "毫升"
                }
            }
        },
        {
            "identifier": "batch_enum_attr_id",
            "name": "风扇枚举型属性",
            "dataType": {
                "type": "enum",
                "specs": {
                    "0": "one",
                    "1": "two",
                    "2": "three"
                }
            }
        },
        {
            "identifier": "fan_float_property",
            "name": "风扇浮点型测试",
            "dataType": {
                "type": "float",
                "specs": {
                    "min": "0.0",
                    "max": "100.0",
                    "unit": "g/ml",
                    "unitName": "毫升"
                }
            }
        }
    ]
}

```

```

    }
  },
  {
    "identifier": "fan_text_property",
    "name": "风扇字符型属性",
    "dataType": {
      "type": "text",
      "specs": {
        "length": "64",
        "unit": "g/ml",
        "unitName": "毫升"
      }
    }
  },
  {
    "identifier": "fan_date_property",
    "name": "风扇时间型属性",
    "dataType": {
      "type": "date",
      "specs": {}
    }
  },
  {
    "identifier": "batch_boolean_attr_id",
    "name": "风扇布尔型属性",
    "dataType": {
      "type": "bool",
      "specs": {
        "0": "close",
        "1": "open"
      }
    }
  },
  {
    "identifier": "fan_struct_property",
    "name": "风扇结构型属性",
    "dataType": {
      "type": "struct",
      "specs": [
        {
          "identifier": "fan_struct
_property_double_child",
          "name": "风扇双精度型属性",
          "dataType": {
            "type": "double",
            "specs": {
              "min": "0.0",
              "max": "100.0",
              "unit": "g/ml",
              "unitName": "毫升"
            }
          }
        },
        {
          "identifier": "fan_struct
_property_float_child",
          "name": "风扇结构型属性浮点子属
性",
          "dataType": {
            "type": "float",
            "specs": {
              "min": "0.0",
              "max": "255.0",

```

```

        "unit": "°",
        "unitName": "度"
    }
}
},
{
    "identifier": "fan_struct
    "name": "风扇结构型属性整数子属
    "dataType": {
        "type": "int",
        "specs": {
            "min": "0",
            "max": "255",
            "unit": "°",
            "unitName": "度"
        }
    }
},
{
    "identifier": "fan_struct
    "name": "风扇结构型属性字符子属
    "dataType": {
        "type": "text",
        "specs": {
            "length": "200",
            "unit": "unit",
            "unitName": "单位"
        }
    }
},
{
    "identifier": "fan_struct
    "name": "风扇结构型属性时间子属
    "dataType": {
        "type": "date",
        "specs": {}
    }
},
{
    "identifier": "fan_struct
    "name": "风扇结构型属性枚举子属
    "dataType": {
        "type": "enum",
        "specs": {
            "0": "one",
            "1": "two",
            "2": "three"
        }
    }
},
{
    "identifier": "fan_struct
    "name": "风扇结构型属性布尔子属
    "dataType": {

```

```

        "type": "bool",
        "specs": {
            "0": "0",
            "1": "1"
        }
    }
}
],
"services": [
    {
        "name": "upgrade",
        "identifier": "upgrade",
        "desc": "升级服务",
        "inputData": [
            {
                "identifier": "model",
                "name": "模型",
                "dataType": {
                    "type": "int",
                    "specs": {
                        "min": "1",
                        "max": "100",
                        "unit": "cm"
                    }
                }
            },
            {
                "identifier": "mac",
                "name": "mac号",
                "dataType": {
                    "type": "text",
                    "specs": {
                        "length": "255"
                    }
                }
            }
        ],
        "outputData": [
            {
                "identifier": "firmVersion",
                "name": "固件版本号",
                "dataType": {
                    "type": "int",
                    "specs": {
                        "min": "1",
                        "max": "100",
                        "unit": ""
                    }
                }
            }
        ],
        "method": "thing.service.upgrade"
    },
    {
        "identifier": "set",
        "name": "set",
        "required": true,
        "desc": "属性设置",
        "method": "thing.service.property.set",
    }
]

```

```

    "inputData": [
      {
        "identifier": "fan_int_property",
        "name": "风扇整数型属性",
        "accessMode": "rw",
        "required": true,
        "dataType": {
          "type": "int",
          "specs": {
            "min": "0",
            "max": "100",
            "unit": "g/ml",
            "unitName": "毫升"
          }
        }
      }
    ],
    "outputData": []
  },
  {
    "identifier": "get",
    "name": "get",
    "required": true,
    "desc": "属性获取",
    "method": "thing.service.property.get",
    "inputData": [
      "fan_double_property",
      "fan_int_property",
      "batch_enum_attr_id",
      "fan_float_property",
      "fan_text_property",
      "fan_date_property",
      "batch_boolean_attr_id",
      "fan_struct_property"
    ],
    "outputData": [
      {
        "identifier": "fan_double_property",
        "name": "风扇双精度型属性",
        "dataType": {
          "type": "double",
          "specs": {
            "min": "0.0",
            "max": "100.0",
            "unit": "g/ml",
            "unitName": "毫升"
          }
        }
      }
    ],
    {
      "identifier": "fan_int_property",
      "name": "风扇整数型属性",
      "dataType": {
        "type": "int",
        "specs": {
          "min": "0",
          "max": "100",
          "unit": "g/ml",
          "unitName": "毫升"
        }
      }
    }
  },
  {
    "identifier": "batch_enum_attr_id",

```

```

        "name": "风扇枚举型属性",
        "dataType": {
            "type": "enum",
            "specs": {
                "0": "one",
                "1": "two",
                "2": "three"
            }
        }
    },
    {
        "identifier": "fan_float_property",
        "name": "风扇浮点型测试",
        "dataType": {
            "type": "float",
            "specs": {
                "min": "0.0",
                "max": "100.0",
                "unit": "g/ml",
                "unitName": "毫升"
            }
        }
    },
    {
        "identifier": "fan_text_property",
        "name": "风扇字符型属性",
        "dataType": {
            "type": "text",
            "specs": {
                "length": "64",
                "unit": "g/ml",
                "unitName": "毫升"
            }
        }
    },
    {
        "identifier": "fan_date_property",
        "name": "风扇时间型属性",
        "dataType": {
            "type": "date",
            "specs": {}
        }
    },
    {
        "identifier": "batch_boolean_attr_id",
        "name": "风扇布尔型属性",
        "dataType": {
            "type": "bool",
            "specs": {
                "0": "close",
                "1": "open"
            }
        }
    },
    {
        "identifier": "fan_struct_property",
        "name": "风扇结构型属性",
        "dataType": {
            "type": "struct",
            "specs": [
                {
                    "identifier": "fan_struct
_property_double_child",
                    "name": "风扇双精度型属性",

```

```

        "dataType": {
            "type": "double",
            "specs": {
                "min": "0.0",
                "max": "100.0",
                "unit": "g/ml",
                "unitName": "毫升"
            }
        }
    },
    {
        "identifier": "fan_struct
        "name": "风扇结构型属性浮点子属
        "dataType": {
            "type": "float",
            "specs": {
                "min": "0.0",
                "max": "255.0",
                "unit": "°",
                "unitName": "度"
            }
        }
    },
    {
        "identifier": "fan_struct
        "name": "风扇结构型属性整数子属
        "dataType": {
            "type": "int",
            "specs": {
                "min": "0",
                "max": "255",
                "unit": "°",
                "unitName": "度"
            }
        }
    },
    {
        "identifier": "fan_struct
        "name": "风扇结构型属性字符子属
        "dataType": {
            "type": "text",
            "specs": {
                "length": "200",
                "unit": "unit",
                "unitName": "单位"
            }
        }
    },
    {
        "identifier": "fan_struct
        "name": "风扇结构型属性时间子属
        "dataType": {
            "type": "date",
            "specs": {}
        }
    }
},

```

\_property\_float\_child",  
性",

\_property\_int\_child",  
性",

\_property\_text\_child",  
性",

\_property\_date\_child",  
性",

```
{
  "identifier": "fan_struct",
  "name": "风扇结构型属性枚举子属性",
  "dataType": {
    "type": "enum",
    "specs": {
      "0": "one",
      "1": "two",
      "2": "three"
    }
  }
},
{
  "identifier": "fan_struct",
  "name": "风扇结构型属性布尔子属性",
  "dataType": {
    "type": "bool",
    "specs": {
      "0": "close",
      "1": "open"
    }
  }
}
]
```

### 6.1.9 导入物模型

导入物模型可以从现有产品直接进行拷贝，也可以是上传本地JSON文件。

#### 前提条件

本项目中存在其他已定义功能的产品，或本地存在一份JSON文件。

#### 操作步骤

1. 进入产品-功能定义页面。

2. 在产品功能定义中，单击导入物模型。

功能定义

标准功能 ●

导入物模型 查看物模型 添加功能

功能类型	功能名称	标识符	数据类型	数据定义	操作
属性	主灯开关 必选	LightSwitch	bool (布尔型)	布尔值：关闭 - 0；开启 - 1；	编辑
属性	环境湿度 可选	EnvironmentH...	float (单精度浮点型)	取值范围：0 ~ 100	编辑 删除
事件	故障上报 必选	Error	-	事件类型：故障	编辑

### 3. 选择导入方式，并进行配置。

- 选择待复制物模型的产品名称，并单击确定完成。

导入物模型 ×

**i** 注：导入的物模型会覆盖原来的功能

**拷贝产品** 导入物模型

\* 选择产品:

请选择产品 ∨

**确定** 取消

- 单击上传文件，选择本地JSON文件，并单击确定完成。

导入物模型 ×

**i** 注：导入的物模型会覆盖原来的功能

拷贝产品 **导入物模型**

\* 上传物模型文件

上传文件

**确定** 取消

## 6.2 调试设备

### 6.2.1 选择认证模组/芯片

为保障客户的设备联网质量和性能，我们会对模组、芯片合作伙伴的品质进行认证，以确保最终设备厂家的设备在联网过程中的高稳定、高性能。

#### 背景信息

您的设备在接入平台前，我们推荐您采购阿里IoT认证通过的芯片或模组，请查看 [阿里IoT认证通过的芯片或模组列表](#)。

#### 操作步骤

1. 进入产品-设备调试页面。

某客户的项目 > 智能灯123

功能定义 2 设备调试 3 人机交互 4 批量投产

选择认证模组/芯片

模组 芯片 常用

品牌: 海凌科  
型号: HLK-M50 详情

品牌: 四川鉴联科...  
型号: WF-R71... 详情

品牌: 乐鑫/Espre...  
型号: ESP-WR... 详情

品牌: 乐鑫/Espre...  
型号: ESP32-... 详情

品牌: 利尔达  
型号: LSDGW... 详情

品牌: 上海汉枫电...  
型号: Efin-EW... 详情

品牌: 上海汉枫电...  
型号: HF-LPB... 详情

品牌: 双驰  
型号: IOT-WF... 详情

品牌: Belon Solutions

品牌: ICOM

智能灯123  
更新时间: 2019-07-25 17:39:40

基本信息 编辑  
所属分类: 灯  
节点类型: 设备  
通讯方式: WIFI  
数据格式: ICA标准数据格式 (推荐)  
Product Key: a10pJXitepE  
Product Secret: \*\*\*\* 显示  
Product Id: 2302549

功能定义 查看  
标准功能: 7  
自定义功能: 0

创建时间: 2019-07-25

上一步: 功能定义 下一步: 人机交互

## 2. 选择其中任一的模组/芯片。

### 芯片信息

[重新选择](#) [采购](#)



类型: 芯片 已认证  
品牌: BEKEN  
认证类型: 阿里云IoT技术认证  
[查看详情](#)

通讯类型: Wi-Fi  
型号: BK7231S  
检测项: 内核测试,通道测试,连接测试,FOTA测试

### 设备端开发



已选择认证芯片  
[下载SDK](#)

>



嵌入式开发  
[设备端开发指南](#)

### 测试设备

产品开发阶段允许添加最多50个测试设备，上线发布后将不再限制设备接入数。

已添加设备0/50 [在线调试](#) [新增测试设备](#)

DeviceName	状态	最后上线时间	操作
------------	----	--------	----

如果没有您适合的模组/芯片，或者您希望自行集成Link Kit SDK实现和云端的连接，可以选择最后的“未认证”。



3. (可选) 单击重新选择，可以对模组/芯片进行重新选择。
4. (可选) 单击采购，可直接购买该模组/芯片。

## 6.2.2 透传/编辑脚本

如果设备无法直接和云端传输JSON格式的数据，则需要设备通过二进制格式将数据透传到云端，由云端运行解析脚本将透传的数据转换成标准ICA格式的JSON数据。

### 前提条件

已创建产品，且产品的数据格式设置为透传/自定义。否则在产品-设备调试页面不显示该内容。

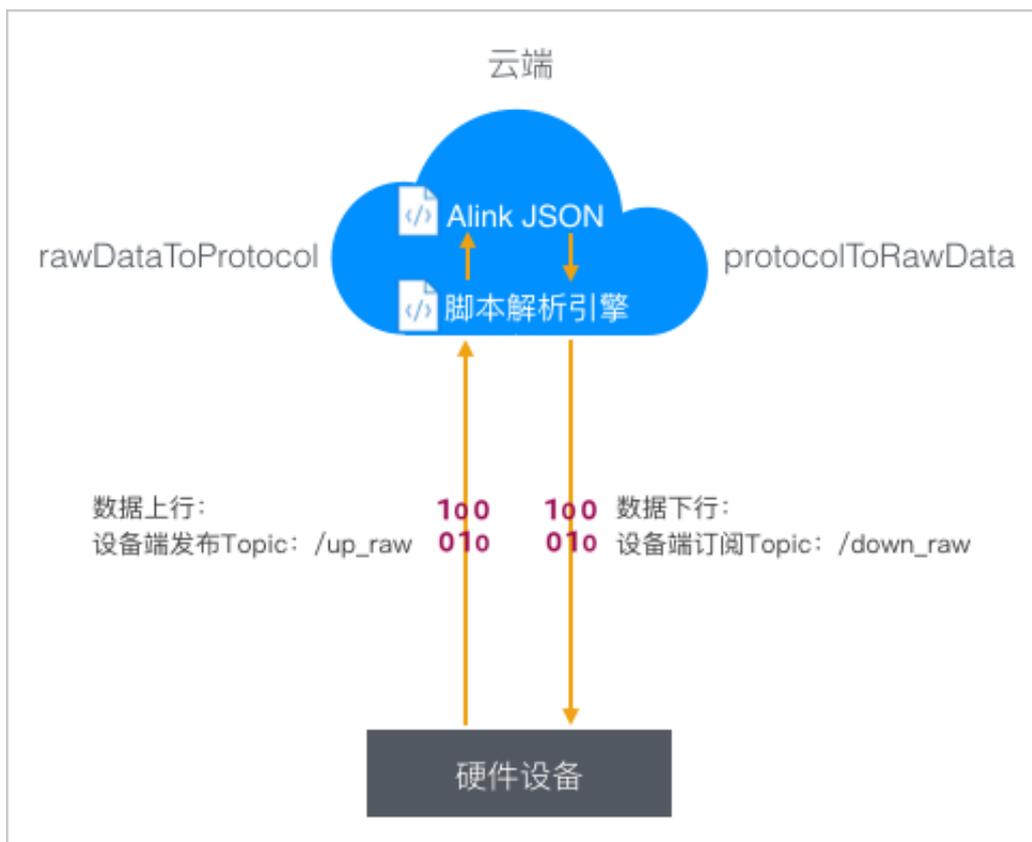
### 透传解析脚本功能介绍

目前解析脚本通过JavaScript开发。设备和脚本的数据协议格式支持标准和自定义两种方式。

- 使用标准协议开发的设备可以直接使用云端自动生成的脚本。
- 如果协议自定义则需要开发者自行开发JS脚本。

脚本需要支持以下两个方法即可和云端进行通信。

- ICA协议数据转二进制数据（protocolToRawData）
- 二进制数据转ICA协议数据（rawDataToProtocol）



### 操作步骤

1. 进入产品-设备调试页面。

## 2. 在数据解析中单击编辑脚本。

文档测试项目 > 透传产品\_灯

功能定义 2 设备调试 人机交互 4 批量投产

**模组信息**

类型: 模组  
品牌: 未认证  
认证类型: 无

通讯类型: 无  
型号: 未认证  
检测项: 无

重新选择 采购

**透传产品\_灯**  
更新时间: 2019-08-15 10:34:24

**基本信息** 编辑  
所属分类: 灯  
节点类型: 设备  
通讯方式: WIFI  
数据格式: 透传/自定义格式  
Product Key: a1dgFrUsm4W  
Product Secret: \*\*\*\* 显示  
Product Id: 2503534

**功能定义** 查看  
标准功能: 7  
自定义功能: 0

**模组** 重选  
品牌: 未认证  
型号: 未认证  
创建时间: 2019-08-15

**设备端开发**

已选择未认证模组 > 嵌入式开发 > 数据解析

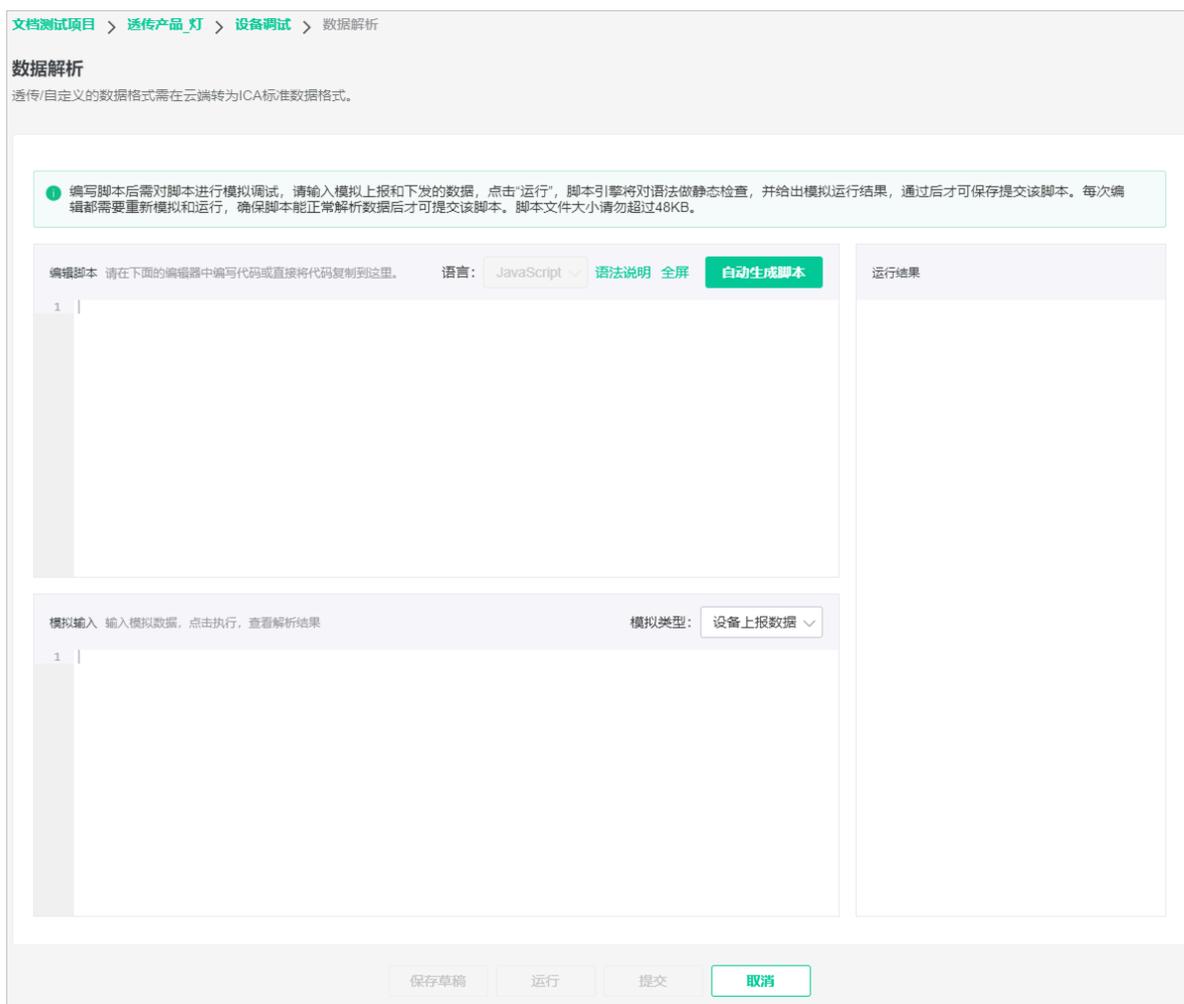
下载SDK 设备端开发指南 需在云端转为ICA标准数据格式 编辑脚本

**测试设备**

产品开发阶段允许添加最多50个测试设备, 上线发布后将不再限制设备接入数。 已添加设备0/50 在线调试 新增测试设备

DeviceName	状态	最后上线时间	操作
------------	----	--------	----

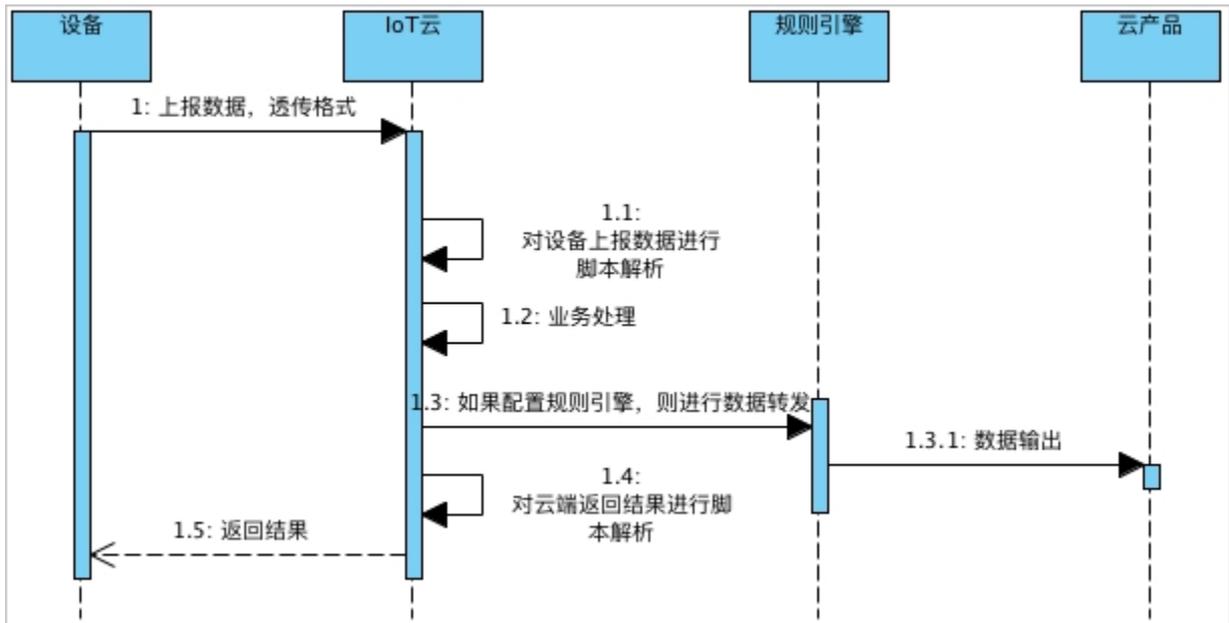
### 3. 编辑脚本，并可以模拟数据调试。



#### 脚本定义

目前脚本仅支持符合ECMAScript 5.1的JavaScript语法。

脚本上传数据的流程如下。



1. 设备上报透传数据。
2. 云端先对设备上报的数据，通过脚本进行解析转换为IoT平台标准数据格式。
3. 使用转换后的数据进行业务处理。
4. 对于云端返回的结果，通过脚本进行解析。
5. 推送转换后的返回结果给设备。

### 脚本编写

#### · 产品定义

以某一个测试产品为例，假设产品有三个属性prop\_float, prop\_int16, prop\_bool。

#### · 脚本示例

```

var COMMAND_REPORT = 0x00; //属性上报
var COMMAND_SET = 0x01; //属性设置
var COMMAND_REPORT_REPLY = 0x02; //上报数据返回结果
var COMMAND_SET_REPLY = 0x03; //属性设置设备返回结果
var COMMAND_UNKOWN = 0xff; //未知的命令
var ALINK_PROP_REPORT_METHOD = 'thing.event.property.post'; //标准
ALink JSON格式topic, 设备上传属性数据到云端
var ALINK_PROP_SET_METHOD = 'thing.service.property.set'; //标准ALink
JSON格式topic, 云端下发属性控制指令到设备端
var ALINK_PROP_SET_REPLY_METHOD = 'thing.service.property.set'; //标准
ALink JSON格式topic, 设备上报属性设置的结果到云端
/*
示例数据:
设备上报数据
传入参数 ->
    0x000000000100320100000000
输出结果 ->
    {"method":"thing.event.property.post","id":"1","params":{"
prop_float":0,"prop_int16":50,"prop_bool":1},"version":"1.0"}
属性设置的返回结果
  
```

```

传入参数 ->
    0x0300223344c8
输出结果 ->
    {"code":"200","data":{},"id":"2241348","version":"1.0"}
*/
function rawDataToProtocol(bytes) {
    var uint8Array = new Uint8Array(bytes.length);
    for (var i = 0; i < bytes.length; i++) {
        uint8Array[i] = bytes[i] & 0xff;
    }
    var dataView = new DataView(uint8Array.buffer, 0);
    var jsonMap = new Object();
    var fHead = uint8Array[0]; // command
    if (fHead == COMMAND_REPORT) {
        jsonMap['method'] = ALINK_PROP_REPORT_METHOD; //ALink JSON格式
        - 属性上报topic
        jsonMap['version'] = '1.0'; //ALink JSON格式 - 协议版本号固定字段
        jsonMap['id'] = '' + dataView.getInt32(1); //ALink JSON格式
        - 标示该次请求id值
        var params = {};
        params['prop_int16'] = dataView.getInt16(5); //对应产品属性中
prop_int16
        params['prop_bool'] = uint8Array[7]; //对应产品属性中 prop_bool
        params['prop_float'] = dataView.getFloat32(8); //对应产品属性中
prop_float
        jsonMap['params'] = params; //ALink JSON格式 - params标准字段
    } else if (fHead == COMMAND_SET_REPLY) {
        jsonMap['version'] = '1.0'; //ALink JSON格式 - 协议版本号固定字段
        jsonMap['id'] = '' + dataView.getInt32(1); //ALink JSON格式
        - 标示该次请求id值
        jsonMap['code'] = '' + dataView.getUint8(5);
        jsonMap['data'] = {};
    }

    return jsonMap;
}
/*
示例数据:
属性设置
传入参数 ->
    {"method":"thing.service.property.set","id":"12345","version":"1.0",
    "params":{"prop_float":123.452, "prop_int16":333, "prop_bool":1}}
输出结果 ->
    0x0100003039014d0142f6e76d

设备上报的返回结果
传入数据 ->
    {"method":"thing.event.property.post","id":"12345","version":"1.0",
    "code":200,"data":{}}
输出结果 ->
    0x0200003039c8
*/
function protocolToRawData(json) {
    var method = json['method'];
    var id = json['id'];
    var version = json['version'];
    var payloadArray = [];
    if (method == ALINK_PROP_SET_METHOD) // 属性设置
    {
        var params = json['params'];
        var prop_float = params['prop_float'];
        var prop_int16 = params['prop_int16'];
        var prop_bool = params['prop_bool'];
        //按照自定义协议格式拼接 rawData
    }
}

```

```

        payloadArray = payloadArray.concat(buffer_uint8(COMMAND_SET
    )); // command字段
        payloadArray = payloadArray.concat(buffer_int32(parseInt(id
    ))); // ALink JSON格式 'id'
        payloadArray = payloadArray.concat(buffer_int16(prop_int16
    )); // 属性'prop_int16'的值
        payloadArray = payloadArray.concat(buffer_uint8(prop_bool
    )); // 属性'prop_bool'的值
        payloadArray = payloadArray.concat(buffer_float32(prop_float
    )); // 属性'prop_float'的值
    } else if (method == ALINK_PROP_REPORT_METHOD) { //设备上报数据返回
结果
        var code = json['code'];
        payloadArray = payloadArray.concat(buffer_uint8(COMMAND_RE
PORT_REPLY)); //command字段
        payloadArray = payloadArray.concat(buffer_int32(parseInt(id
    ))); // ALink JSON格式 'id'
        payloadArray = payloadArray.concat(buffer_uint8(code));
    } else { //未知命令, 对于有些命令不做处理
        var code = json['code'];
        payloadArray = payloadArray.concat(buffer_uint8(COMMAD_UNKOWN
    )); //command字段
        payloadArray = payloadArray.concat(buffer_int32(parseInt(id
    ))); // ALink JSON格式 'id'
        payloadArray = payloadArray.concat(buffer_uint8(code));
    }
    }
    return payloadArray;
}
//以下是部分辅助函数
function buffer_uint8(value) {
    var uint8Array = new Uint8Array(1);
    var dv = new DataView(uint8Array.buffer, 0);
    dv.setUint8(0, value);
    return [].slice.call(uint8Array);
}
function buffer_int16(value) {
    var uint8Array = new Uint8Array(2);
    var dv = new DataView(uint8Array.buffer, 0);
    dv.setInt16(0, value);
    return [].slice.call(uint8Array);
}
function buffer_int32(value) {
    var uint8Array = new Uint8Array(4);
    var dv = new DataView(uint8Array.buffer, 0);
    dv.setInt32(0, value);
    return [].slice.call(uint8Array);
}
function buffer_float32(value) {
    var uint8Array = new Uint8Array(4);
    var dv = new DataView(uint8Array.buffer, 0);
    dv.setFloat32(0, value);
    return [].slice.call(uint8Array);
}
}

```

### 控制台模拟数据调试



#### 说明:

模拟时需要注意产品定义的属性的读写属性，否则将会产生错误。

- 模拟设备上报数据

模拟类型选择设备上报数据，填写测试数据。控制台中模拟输入的数据为设备上报数据的十六进制格式数据。

```
0x00002233441232013fa00000
```

单击运行，查看上报数据输出结果。

```
{
  "method": "thing.event.property.post",
  "id": "2241348",
  "params": {
    "prop_float": 1.25,
    "prop_int16": 4658,
    "prop_bool": 1
  },
  "version": "1.0"
}
```

- 设备上报数据返回结果

模拟类型选择设备接收数据，填写测试数据。

```
{
  "id": "12345",
  "version": "1.0",
  "code": 200,
  "method": "thing.event.property.post",
  "data": {}
}
```

单击运行，查看接收数据输出结果，输出结果为脚本转换结果的十六进制格式数据。

```
0x0100003039014d0142f6e76d
```

- 模拟属性设置设备返回结果

模拟属性设置设备返回属性设置结果，填写测试数据。

```
0x0300223344c8
```

单击运行，查看设备上报的数据。

```
{
  "code": "200",
  "data": {},
  "id": "2241348",
  "version": "1.0"
}
```

```
}

```

### 本地调试脚本

仅用于本地测试，控制台请使用[控制台模拟数据调试](#)，为了方便开发及调试脚本，可将脚本放在本地环境中进行调用，参考如下。

```
// rawDataToProtocol和protocolToRawData的实现放在这里
// Test Demo
function Test()
{
    //0x001232013fa00000
    var rawdata_report_prop = new Buffer([
        0x00, //固定command头, 0代表是上报属性
        0x00, 0x22, 0x33, 0x44, //对应id字段, 标记请求的序号
        0x12, 0x32, //两字节 int16, 对应属性 prop_int16
        0x01, //一字节 bool, 对应属性 prop_bool
        0x3f, 0xa0, 0x00, 0x00 //四字节 float, 对应属性 prop_float
    ]);

    rawDataToProtocol(rawdata_report_prop);

    var setString = new String('{"method":"thing.service.property.set","id":"12345","version":"1.0","params":{"prop_float":123.452, "prop_int16":333, "prop_bool":1}}');
    protocolToRawData(JSON.parse(setString));
}

Test();
```

### 简易数据透传协议

为了让开发者免去脚本的开发，以及考虑减轻MCU的运算，我们制定了一套简易的数据协议，核心数据传输采用TLV格式。

1Byte	2Byte	1Byte	1Byte	nByte	1Byte
head	length	ver	cmd	payload	chksum

组成部分	字段名	长度 (字节)	说明
帧头	head	1	固定位 0xAA
	length	2	表示 ver 到 chksum 的长度
	Ver	1	协议版本号
	cmd	1	帧类型 (数据 / 控制两大类)
主体	payload	n	有效数据
帧尾	chksum	1	前面所有字节的和校验

- 数据传输统一使用大端（即高字节在前，低字节在后）字节序
- 为保证传输可靠性，通信需要实现应答、超时及重传机制



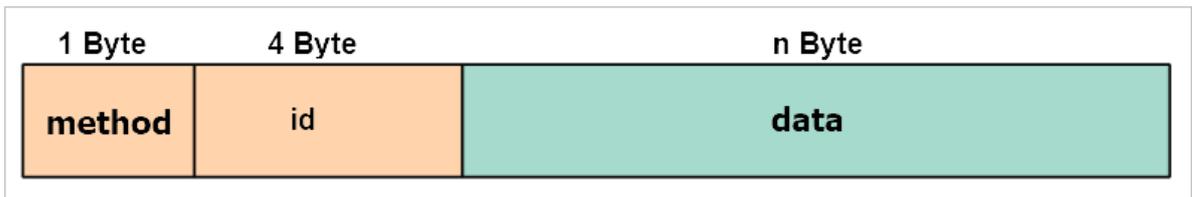
### 说明:

本文档对通信方式和物理参数不做要求，例如UART通信，需要指定如下参数：波特率：115200，数据位：8，奇偶校验：无，停止位：1。

### · 协议帧类型定义

指令		cmd	说明	
<b>MCU → 模组（上行）最高位置 0</b>				
数据帧	透传数据	发送	0x00	模组转发，无需解析
		回应	0x00	
控制帧	保留			
<b>MCU ← 模块（下行）最高位置 1</b>				
数据帧	透传数据	发送	0x80	模组转发，无需解析
		回应	0x80	
控制帧	保留			

· payload格式定义



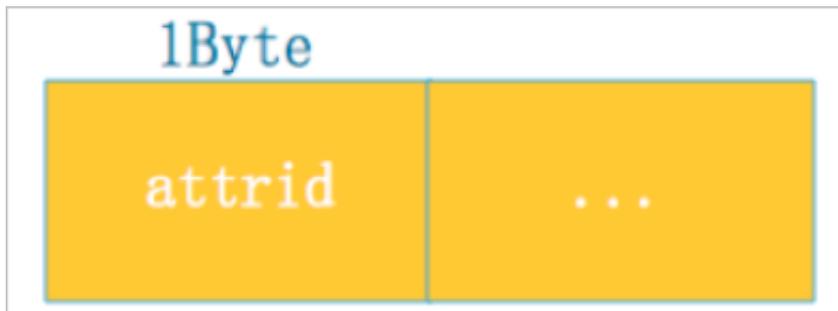
- method: 操作的方法, 定义如下。

method name		method	说明
下行	Get	0x00	服务端获取设备属性。成功则通过 Get 回复; 失败则通过 Ack 回复
	Set	0x01	服务端配置设备属性。通过 Ack 回复
	Service	0x02	服务端下发服务。通过 Service 回复
上行	Report	0x80	设备上报属性。通过 Ack 回复
	Event	0x81	设备上报事件。通过 Ack 回复
通用	Ack	0xFF	通用应答。带状态码 (error code) 可用于请求的出错回复

- id: 帧标识符, 用于区分不同的请求。回复帧与请求帧的id必须相同, 表示对该帧的回复。

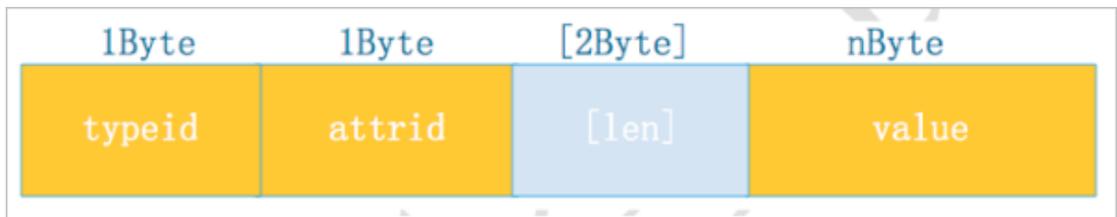
- data: 数据域, 具体格式根据method来确定。

■ Get方法的data域格式



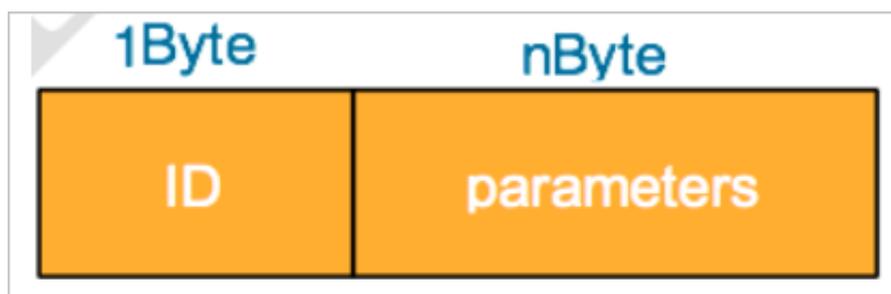
attrid即云端需要读取属性的ID, 通过编号表示。

■ Set/Report方法的data域格式



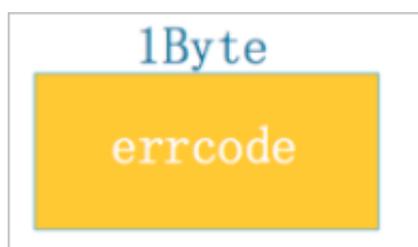
协议中将类型、属性进行编号表示。len非必须, 仅在类型为数组和文本 (text) 的情况下需要, 表示长度。

■ Service/Event方法的data域格式



对服务和事件进行编号传输，ID表示服务或者事件的编号，parameters表示服务或者事件携带的参数。

#### ■ Ack方法的data域格式



errcode表示错误码。

## MCU SDK

针对上述提到的二进制的标准协议，我们提供了MCU SDK实现了协议的封装。此外，我们还会根据开发者的产品功能定义在MCU SDK中生成与之对应的代码和上报、接收处理逻辑。开发者使用MCU SDK开发就不用实现通讯协议和产品功能的定义，直接按照提供的API接口调用以及添加自己的业务逻辑即可。

例如，从云端下发一个关灯（对应属性标识Switch）的请求，需要在开发者在特定的API内部实现Switch的处理，MCU SDK默认实现了对云端的回复。设备本地灯的开关状态变化，开发者的程序识别到后，调用修改Switch属性的API后，MCU SDK会将变化上报到云端。开发者就只需要关注设备业务功能的开发即可。

目前MCU SDK支持如下几种芯片型号生成对应开发工程，开发者可以直接基于此工程直接开发自己的应用。如果选择其他平台，我们会提供SDK和简单的示例demo，开发者可以在Linux进行编译运行。

- STM8S207

开发IDE使用IAR for STM8（EWSTM8）。

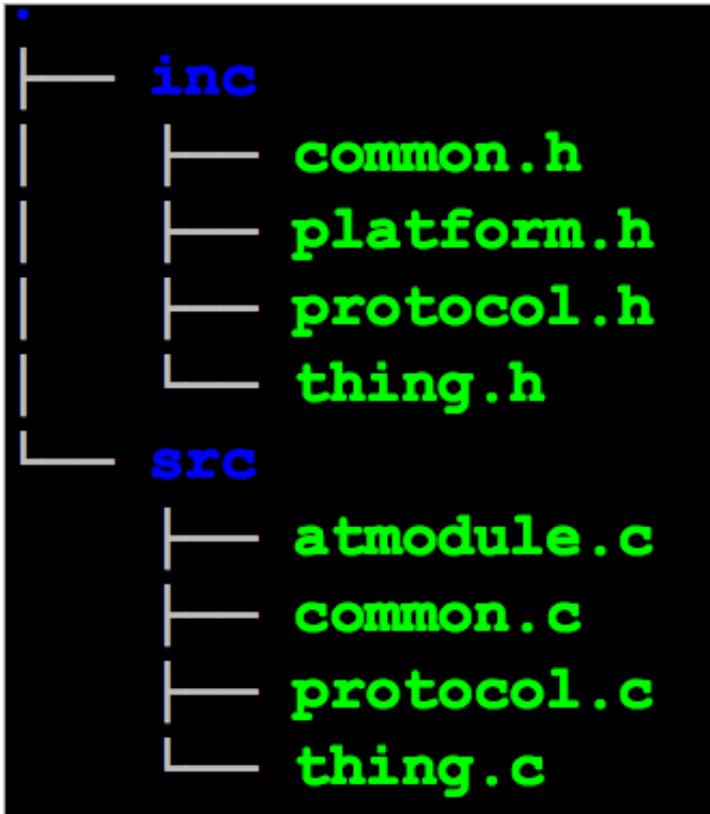
- STM32L053R8

开发IDE使用Keil MDK5。

- 其他平台

仅提供简单的示例。可以在Linux中编译运行。

MCU SDK的核心代码位于`sdk-core`目录，目录结构如下所示，包括了头文件目录`inc`和源代码目录`src`。



头文件说明如下。

- `common.h`

SDK共用的头文件，包括了类型定义、SDK全局对象定义以及公共的API。需要开发者关注。

- `platform.h`

定义了需要开发者实现或者处理的函数。需要开发者关注。

- `protocol.h`

定义了和云端通讯协议的使用的接口。开发者可以不用关注。

- `thing.h`

包含了产品功能（属性、服务、事件）相关定义。需要开发者关注。

源文件说明如下。

- **common.c**

SDK公共代码的实现。

- **protocol.c**

和云端通信协议的定义及接口实现。

- **thing.c**

产品功能相关的接口实现。

*thing.h*和*thing.c*部分代码会根据产品的TSL自动生成。

## API说明

需要用户调用的接口如下。

- 公共接口

- SDK初始化函数: `void boneSdkInit(void)`
- SDK运行函数, 在while中调用: `void boneSdkRun(void)`
- 接收串口的字节数据, 在串口中断服务程序中调用: `Int32_t boneRcvFromUart( Uint8_t *data, Uint16_t length);`
- 系统运行时间计算 (定时1ms调用, 暂时可以不用实现): `void boneSystimeInc(void );`

- 产品功能相关接口

```

/ 属性值范围定义 /
#define ATTR_XXX

/ 属性值的设置和获取, AttrName表示的是属性名称, 实际的接口名称和属性标识对应,
AttrVal表示的是属性的值, ValueType表示的是属性的类型 /
void boneSet_AttrName(AttrVal);
ValueType boneGet_AttrName(void);

/ 事件上报, EventName表示的是事件名称, 实际的接口名称和事件标识对应, Params表示
事件的输出参数 /
void boneEvtPost_EventName(Params);

```



### 说明:

以上接口及参数定义的只是一个模版, 具体的API要视自己定义的产品功能而定。以下是示例产品自动生成的API。

- 属性值范围定义

```
#define RANG_PROPINT8_R_MIN -100
```

```
#define RANG_PROPINT8_R_MAX 100
```

- 属性值的设置和获取

```
void boneSet_PropInt8_r(Int8_t data);
Int8_t boneGet_PropInt8_r(void);

void boneSet_PropUint8_r(Uint8_t data);
Uint8_t boneGet_PropUint8_r(void);
```

- 事件上报

```
void boneEvtPost_EventInfo(eo_EventInfo_t *arg);

void boneEvtPost_EventAlarm(void);
```

- 需要用户实现的接口

```
/ 串口发送协议数据 /
Int32_t boneUartSend(Uint8_t *buffer, Uint16_t length);

/* 锁相关接口, 无os可空实现 */
void *boneMutexCreate(void);
void boneMutexDestroy(void *mutex);
void boneMutexLock(void *mutex);
void boneMutexUnlock(void *mutex);
```

- 需要用户添加处理方法的接口

```
/* 属性变化处理函数 */
void bonePropChangeHandler(Int32_t index);

/* 服务处理函数, 如果没有服务可忽略。接口名称中的ServiceName表示的是服务名称, 实际的接口名称和服务标识对应 */
static Int32_t boneServCall_ServiceName(Uint32_t id, Uint8_t *data,
    Uint16_t length)
```

示例如下。

```
static Int32_t boneServCall_SrvAsync1(Uint32_t id, Uint8_t *data,
    Uint16_t length);

static Int32_t boneServCall_SrvAsync2(Uint32_t id)
```

## 6.2.3 新增测试设备

可以在平台上直接添加测试设备, 可以是实物设备, 也可以是虚拟设备。

### 操作步骤

1. 进入产品-设备调试页面。
2. 单击新增测试设备。

### 3. 配置参数。

DeviceName可配置自定义字符串、MAC地址、IMEI号或自定义SN等。如果为空，则由系统自动生成。

#### 新增测试设备

DeviceName可以是MAC地址、IMEI号或自定义SN等，须确保产品下唯一，为空将由系统自动颁发，您可以烧录到设备中，并上报到云端进行鉴权认证。

DeviceName ?

确定 取消

- 单击确定，页面提示“添加成功”，并显示该设备的激活凭证（ProductKey、DeviceName、DeviceSecret）。



说明：

针对每一款产品，平台提供50个免费的激活码（ProductKey，DeviceName和DeviceSecret），用于设备的开发调试。

- 单击确定，此时设备信息展示在测试设备的列表中。

#### 测试设备

产品开发阶段允许添加最多50个测试设备，上线发布后将不再限制设备接入数。 已添加设备 1/50 在线调试 新增测试设备

DeviceName	状态	最后上线时间	操作
11.11.11.11	● 未激活	-	<a href="#">查看</a> <a href="#">调试</a> <a href="#">激活凭证</a>

## 6.2.4 添加设备标签

可以为设备添加标签，如地理位置标签等，也可以自定义标签名称。

### 前提条件

已成功添加了某测试设备。

### 操作步骤

1. 进入产品-设备调试页面。
2. 单击设备对应的查看。

### 测试设备

产品开发阶段允许添加最多50个测试设备，上线发布后将不再限制设备接入数。 已添加设备1/50

在线调试
新增测试设备

DeviceName	状态	最后上线时间	操作
11.11.11.11	● 未激活	-	查看 调试 激活凭证

3. 在设备信息页签中，单击标签信息中的 立即添加。

11.11.11.11 未激活

产品: 智能灯123
ProductKey: a10pJXltepE 复制
DeviceSecret: \*\*\*\*\* 显示

设备信息
Topic列表
运行状态
事件管理
服务调用
设备影子
文件管理
日志服务
在线调试

#### 设备信息

产品名称	智能灯123	ProductKey	a10pJXltepE <span style="color: green;">复制</span>	区域	-
节点类型	设备	DeviceName	11.11.11.11 <span style="color: green;">复制</span>	DeviceSecret	***** <span style="color: green;">显示</span>
备注名称 <span style="color: gray;">●</span>	<span style="color: green;">编辑</span>	IP地址	-	固件版本	-
添加时间	2019/07/26 09:54:02	激活时间		最后上线时间	
当前状态	未激活	实时延迟 <span style="color: gray;">●</span>	<span style="color: green;">测试</span>		

#### 设备扩展信息

SDK 语言	-	版本号	-	模组商	-
模组信息	-				

#### 标签信息

设备标签: 无标签信息, 立即添加

此页面包含设备的详细信息，包括设备信息、Topic列表、运行状态、事件管理、服务调用、设备影子、文件管理、日志管理、在线调试等信息。

文档版本：20190911

77

#### 4. 配置地理位置标签，或单击新增标签进行自定义。



添加标签

地理位置标签:

coordinate: 无坐标信息 重置

设备标签:

+ 新增标签

确认 取消

#### 5. 单击确认完成。

添加成功后，标签信息处的立即添加就会变成编辑。可以单击编辑对标签进行修改和继续添加。

## 6.2.5 在线调试设备

在线调试用来调试设备端与云端的连接，实现在云端可以查看设备运行日志。平台支持连接线下真实设备进行调试和模拟真实设备调试。

### 前提条件

已成功添加了某测试设备，参见[新增测试设备](#)。

如果调试真实设备，还需该设备已配置与平台的连接，参见[#unique\\_44](#)。

### 背景信息

在线调试当前仅支持对单个功能进行调试，您可以在下方选择一个功能进行在线实时调试，支持分别对设备的属性和事件进行调试。

调试设备的信息推送方式有以下几种：

- 普通推送
- 策略推送
  - 定时推送
  - 连续推送

在指定的某个将来的时间点，进行自动推送。

在某个将来的时间范围内，按一定的时间间隔，连续推送。

设备上线后，默认实时刷新设备的上下行数据，包括设备上线、激活、数据上报、云端下发等日志。

### 操作步骤

1. 进入产品-设备调试页面。
2. 选择其中任一方式进入调试页面。

- 单击设备对应的调试。



- 单击在线调试，并下拉选择设备的DeviceName。



### 3. 调测真实设备。

**在线调试**

调试设备: 11.11.11.11

**调试设备**

调试真实设备 | 虚拟真实设备

功能: 请选择 方法: 请选择

1
---

发送指令 重置

**实时日志** • 检测到设备 在线 自动刷新  刷新 清屏

类型 / 时间	内容
 暂无数据	



说明:

调试真实设备需要先建立设备与平台的连接，且设备已完成配网等操作。

a) 在调试真实设备中，选择设备的功能和他方法。



b) 单击发送指令。

提示“指令发送成功”后，右侧会显示相应的实时日志。

### 在线调试

调试设备: 11.11.11.11

---

#### 调试设备

调试真实设备 | 虚拟真实设备

功能: 主灯开关(Light... | 方法: 设置

```
1 {
2   "LightSwitch": 0
3 }
```

[发送指令](#) [重置](#)

---

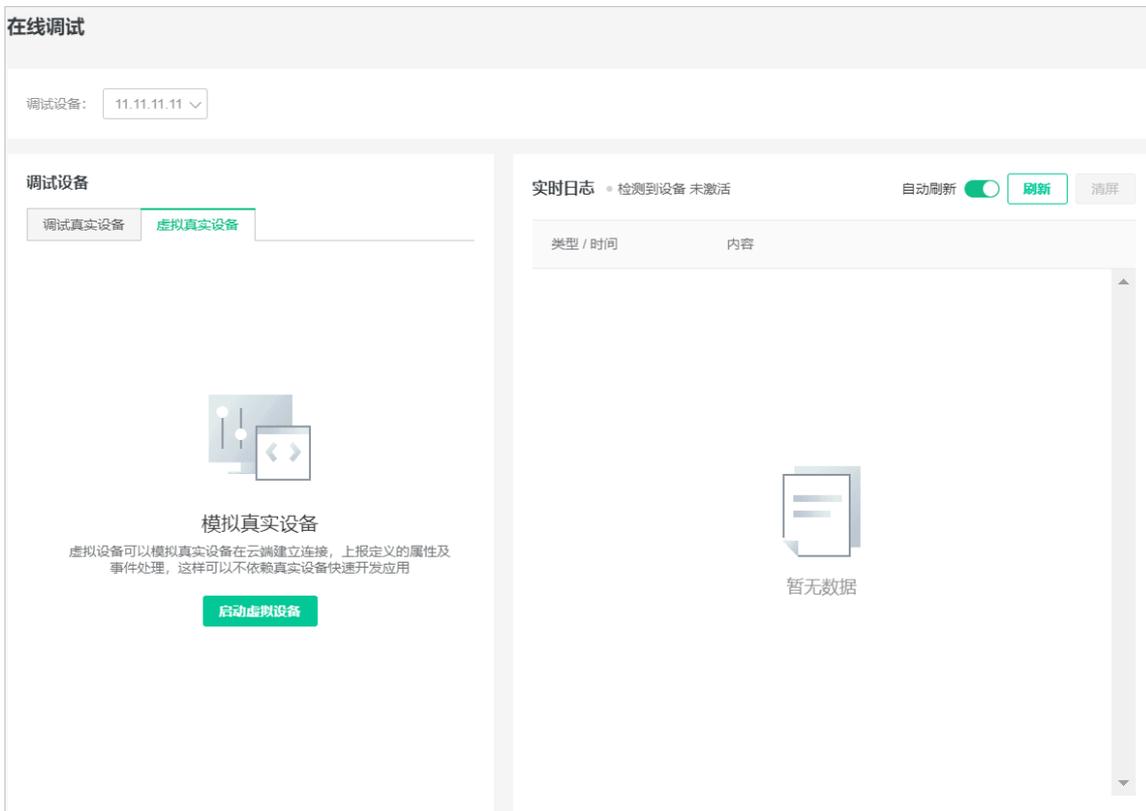
#### 实时日志

检测到设备 在线 自动刷新 [刷新](#) [清屏](#)

类型 / 时间	内容
设备上报数据 2019-09-04 16:00:07	2019-09-04 16:00:07.443, 17ADAA5F7B114677A22BBF4A21D1B266, upstream - bizType=PROPERTY_REPORT,params={\"aliyunPk\":\"1917535506509689\", \"id\":\"12345\", \"iotId\":\"C6SmbhwA9H6x1L6ftxa000100\", \"methodName\":\"thing event property.post\", \"params\":{\"LightSwitch\":0}, \"proxyName\":\"thing event property.post\", \"topic\":\"/sys/a1Q2kD11SmZ/online_light/thing/event/property/post\", \"uniMsgId\":\"1169158183122423298\", \"version\":\"1.0\"}, result=code:200,message:success,topic=/sys/a1Q2kD11SmZ/online_light/thing/event/property/post,response={\"code\":200,\"data\":{}, \"id\":\"12345\", \"message\":\"success\", \"methodName\":\"thing event property.post\", \"version\":\"1.0\"}, device={\"alinyu nCommodityCode\":\"iohub_senior\", \"deviceKey\":\"C6SmbhwA9H6x1L6ftxa\", \"deviceSecret\":\"****\", \"gmtCreate\":1567581371000, \"gmtModified\":1567581371000, \"id\":52304166, \"iotId\":\"C6SmbhwA9H6x1L6ftxa000100\", \"name\":\"online_light\", \"productKey\":\"a1Q2kD11SmZ\", \"rbacTenantId\":\"17ADAA5F7B114677A22BBF4A21D1B266\", \"region\":\"cn-shanghai\", \"status\":0, \"statusLast\":0, \"thingType\":\"DEVICE\", \"scriptData={}, useTime=2, tr acelId=0bc061d915675840074211642d30db
云端下发数据	2019-09-04 16:00:07.283, 17ADAA5F7B114677A22BBF4A2

#### 4. 调测虚拟设备。

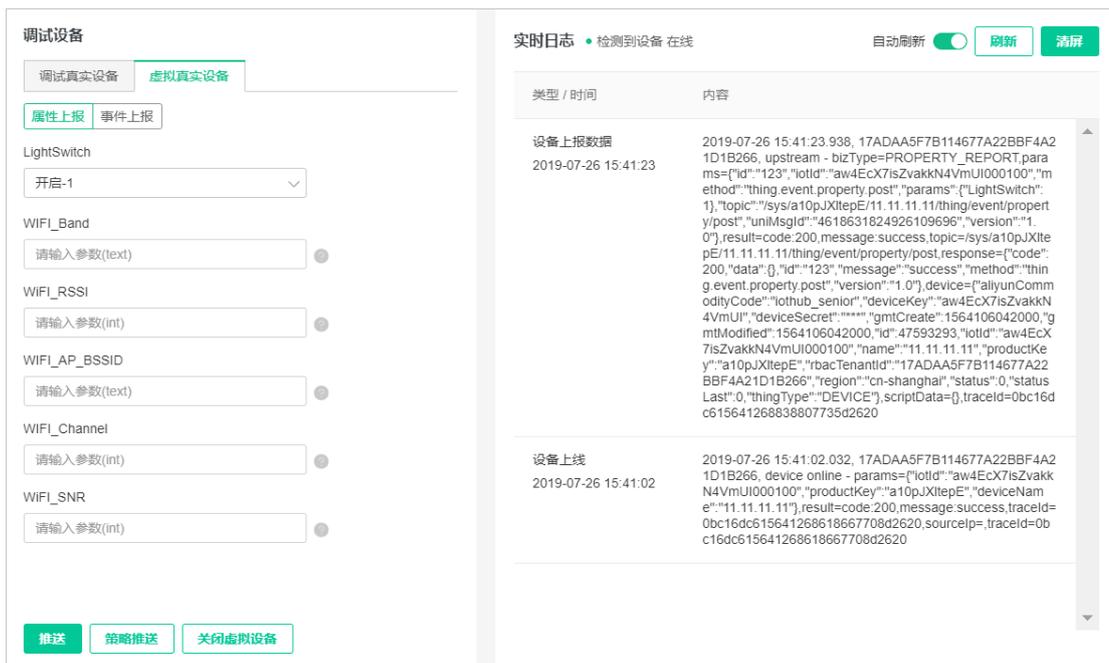
a) 选择虚拟真实设备，并单击启动虚拟设备。



设备上线后，右侧会显示相应的实时日志。

b) 调试设备的某个属性或事件。

- 配置相应的参数，单击推送。右侧会显示相应的实时日志。



- 配置相应的参数，单击策略推送，并配置详细的推送方式。

当到达设定的时间，会自动进行相应的推送。



策略推送

\* 策略推送

定时推送 ^

请选择日期

✓ 定时推送

连续推送

确认 取消

## 6.3 人机交互

### 6.3.1 功能概述

由于生活领域的智能设备在使用过程中，会和消费者产生人机互动，用户希望能够通过App控制、语音控制或设备互相联动，来全方位体验智能生活。为了让开发者可以更容易地实现，只需要进行人机交互端的设置，如App、语音中和产品相关的配置。

当前生活物联网领域，依赖用户使用App进行登录和绑定。为了满足不同客户的需求，平台当前提供两种App的交互终端，都可以让最终用户从应用市场下载使用。

- 自有App：开发者可以开发自己的品牌App，对该产品进行配网和控制。

适合希望定制自己的品牌App，自己去开发各种个性化功能和业务逻辑的客户。其中物联的部分，可以基于平台提供的SDK、API和插件等，具体开发参考开发指南中相关文档。

- 公版App（免开发）：平台提供一个公版App，无需额外开发，即可直接对设备进行配网和控制（目前公版App已支持WiFi、以太网、蜂窝网、蓝牙、zigbee设备）。

公版App的名称为云智能，不带特殊品牌，App的Logo如下。



### 配置规则

平台提供以下人机交互的配置项，针对不同站点，具体的配置项也会不同，请注意区分。

配置项	自有品牌App		公版App		产品发布后
	中国站	国际站	中国站	国际站	
多语言管理	支持	支持	支持	支持	不可修改
分享方式	支持	支持	默认“授权式”	默认“授权式”	可修改
设备告警	支持	支持	支持	支持	可修改
配网引导	支持	支持	支持	支持	可修改
APP界面	支持	支持	支持	支持	可修改
产品联动	支持	支持	支持	支持	不可修改
天猫精灵	支持	不适用	支持	不适用	可修改
Amazon Echo	不适用	支持	不适用	支持	可修改
Google Home	不适用	支持	不适用	支持	可修改
IFTTT	不适用	支持	不适用	支持	可修改



说明:

产品发布后，如需修改配置项，需将产品撤回发布后才能修改。撤回发布期间，会暂停量产生设备证书，已经量产的设备不影响正常激活和使用。

## 6.3.2 配置App交付终端

平台提供两种App交付终端，用户可以根据自身需求以及技术能力进行选择。

### 背景信息

平台提供以下两种App的交互终端。

- 自有App：开发者可以开发自己的品牌App，对该产品进行配网和控制。  
适合希望定制自己的品牌App，自己去开发各种个性化功能和业务逻辑的客户。其中物联的部分，可以基于平台提供的SDK、API和插件等，具体开发参考开发指南中相关文档。
- 公版App（免开发）：平台提供一个公版App，无需额外开发，即可直接对设备进行配网和控制（目前公版App已支持WiFi、以太网、蜂窝网、蓝牙、zigbee设备）。

公版App的名称为云智能，不带特殊品牌，App的Logo如下。



### 操作步骤

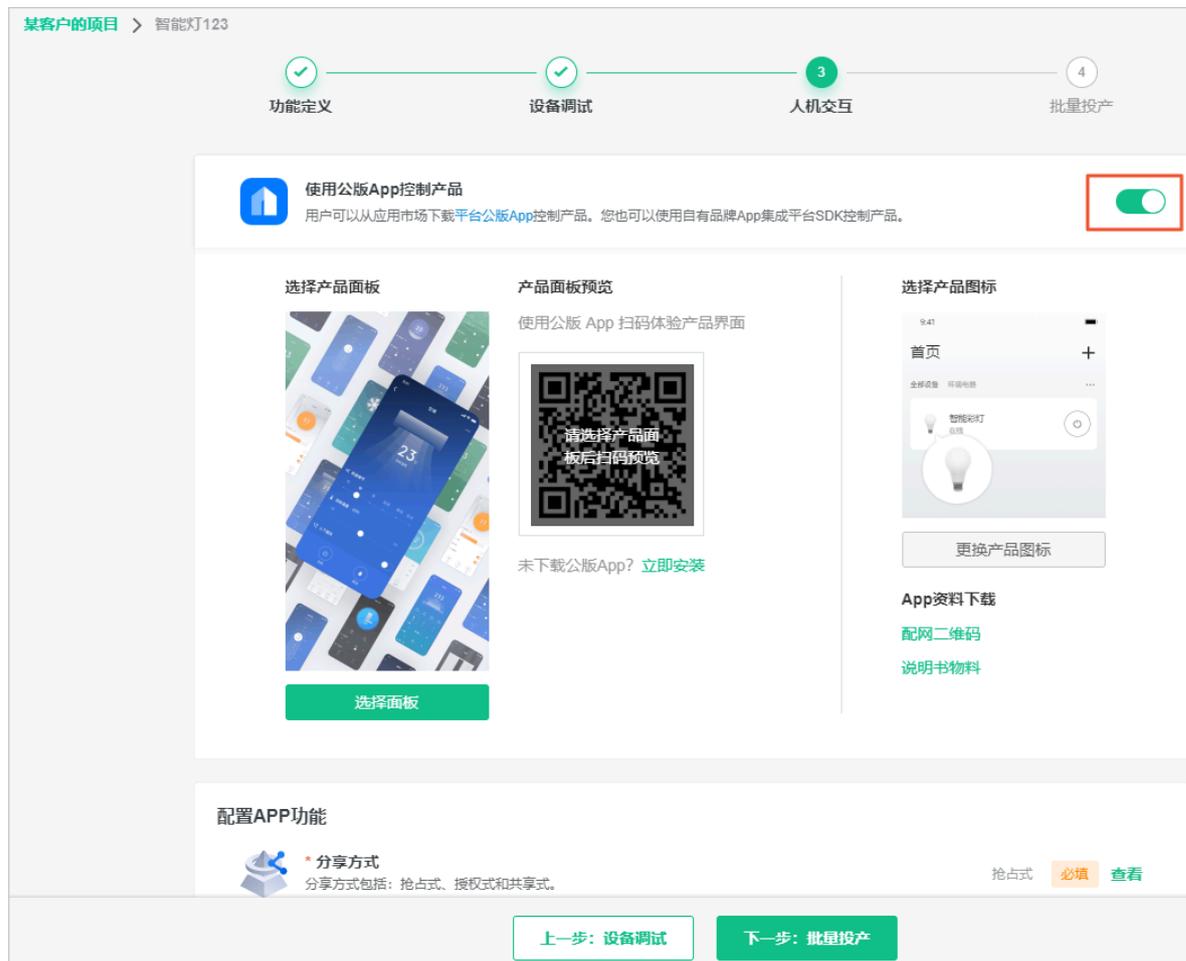
### 1. 进入人机交互页面。

页面默认选用自有App，即未开启使用公版App控制产品。



## 2. 打开使用公版App控制产品的开关。

打开开关后，页面如下。



### 6.3.3 配置App分享方式

为设备选择一种分享方式，用户使用App绑定设备时，设备的归属关系受此处的分享方式约束。

#### 背景信息

平台支持的分享方式有抢占式、授权式、共享式。

分享方式为必填项，默认值为授权式。产品发布后，分享方式支持修改。

公版App对设备分享安全的要求较高，分享方式仅支持“授权式”，不可更改。

#### 操作步骤

1. 进入人机交互页面。
2. 单击分享方式对应的修改。

修改分享方式，仅针对自有品牌App进行操作。

### 3. 选择新的分享方式，并单击返回完成。

分享方式	场景描述	适用范围
<input checked="" type="radio"/> 抢占式	用户A绑定设备后，A成为管理员； 用户B再来绑定时，绑定成功，并成为管理员。用户A自动和该设备解绑，无法继续使用。	适用于安全等级一般的设备，或放在公共场合的公用设备，比如共享设备
<input type="radio"/> 授权式	用户A绑定设备后，A成为管理员； 用户B再来绑定时，无法绑定，APP提示：“该设备已被用户“18411123442”绑定，如果您也想绑定该设备，需要他解除绑定，或将设备分享给您”。	适用于安全等级比较高的设备，或携带私人数据设备，比如手环
<input type="radio"/> 共享式	用户A绑定设备后，A成为管理员； 用户B再来绑定时，绑定成功，成为子成员，用户A还是管理员。子成员数量无上限。	适用于安全等级比较低的设备，或家庭共享的设备，比如体重秤

分享方式	场景描述	适用范围
抢占式	用户A绑定设备后，A成为管理员；用户B再来绑定时，绑定成功，并成为管理员。用户A自动和该设备解绑，无法继续使用	适用于安全等级一般的设备，或放在公共场合的公用设备，如共享设备
授权式	用户A绑定设备后，A成为管理员；用户B再来绑定时，无法绑定，APP提示：“该设备已被18411123442用户绑定，如果您也想绑定该设备，需要A解除绑定，或将设备分享给您”。	适用于安全等级比较高的设备，或携带私人数据设备，如手环
共享式	用户A绑定设备后，A成为管理员；用户B再来绑定时，绑定成功，成为子成员，用户A还是管理员。子成员数量无上限。	适用于安全等级比较低的设备，或家庭共享的设备，如体重秤

## 6.3.4 配置App多语言管理

此项用于配置展示的产品名称和功能名称，以及产品销往国外的当地语言。

### 背景信息

产品名称和功能名称，会影响到公版App和官方插件中的展示名称，请按照您自身产品本身进行修改中文和其他多语言的名称，影响页面包括但不限于以下所示。

类型	展示文字	展示位置
----	------	------

产品名称	品牌名称 + 产品名称 + 产品型号（字段拼接，尽量不要有重复），支持多语言	公版App/插件中所有展示产品名称的地方，如“待配网列表”、“首页”设备列表、“设备界面”顶部等。切换语言时，这些位置会展示相应的多语言名称。
功能名称	根据具体功能配置	公版App/插件中所有展示产品功能名称的地方，如“设备界面”中的功能名称、“我的智能”中选择设备功能、“云端定时”中选择设备功能等。切换语言时，这些位置会展示相应的多语言名称。



### 说明:

配置的产品名称需经过审核后才可以使用。如果审核不通过，需要重新修改后再提交审核，直至审核通过。

公版App中，多语言为必填参数，自有App中可以根据实际情况配置。

### 操作步骤

1. 进入人机交互页面。
2. 单击配置多语言对应的未设置。
3. 根据页面提示的配置规则，配置产品信息、产品简单功能、产品复杂功能。

#### 多语言管理

中文
英文
西班牙语
法语
俄语
德语
日语
韩语
印地语
意大利语

**产品信息** 用于APP界面和配网列表中显示的产品名称 如：大象电器 - 加湿器 - FY510

① 请确认所需语言的产品名称已填写完整，否则无法在对应语言的公版App产品待配网列表中露出。

\* 产品名称:  4/20  3/20  6/20

**产品简单功能** 用于APP界面和场景配置显示对应的功能名称。

序号	功能名称	属性值	中文
1	主灯开关	—	<input type="text" value="主灯开关"/>
		关闭	<input type="text" value="关闭"/>
		开启	<input type="text" value="开启"/>
2	频段	—	<input type="text" value="频段"/>
3	信号强度	—	<input type="text" value="信号强度"/>
4	热点BSSID	—	<input type="text" value="热点BSSID"/>
5	信道	—	<input type="text" value="信道"/>

#### 4. 切换其他语言页签进行配置，例如“英语”。

配置方法与配置中文语言相同。



说明：

如果您的设备将销往国外，请务必配置英文和其他多语言，否则用户切换App多语言时可能出现中文或不准确的英文。

#### 5. 单击保存完成。

### 6.3.5 配置App配网方案和引导文案

根据自身产品情况，配合阿里云IoT提供的WiFi配网方案，并配置对应的引导文案。设置默认配网方案及备选方案。

#### 背景信息

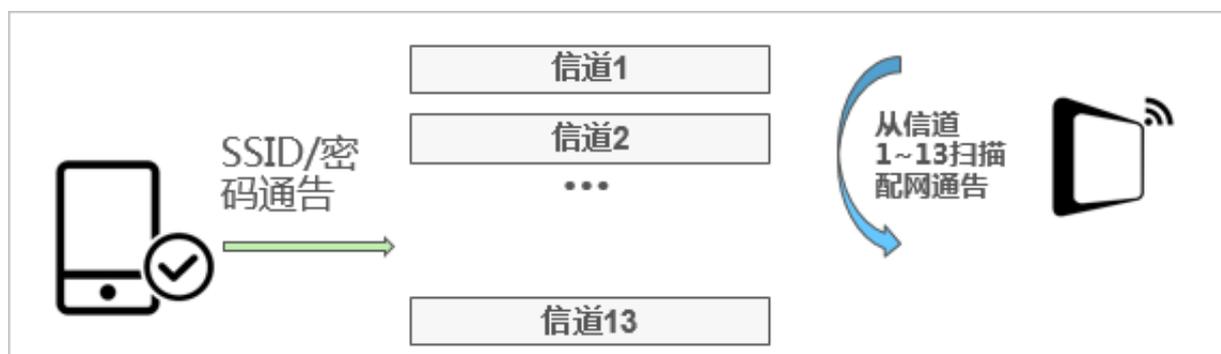
针对有屏可输入设备，如手机、平板等，用户可以直接通过界面输入SSID和PASSWORD让设备上WiFi网络。对于无屏设备，如插座、灯泡等，用户无法通过输入上网信息让设备接入网络，此时则需要对这些设备进行配网操作。

配网也就是将路由器的WiFi的SSID和PASSWORD通过某种方式传递到终端设备，让终端设备可以接入WiFi网络的过程。

阿里提供的WiFi配网方案有：一键配网（又称smartConfig或广播配网）、手机热点配网、设备热点配网、蓝牙辅助配网、零配（需设备支持）、摄像头扫码配网（需设备支持）。

#### 一键配网

一键配网是手机通过广播发送上网热点信息（路由器的SSID和PASSWORD），设备通过监听方式获取信息，并连接上网热点的方式。该方案只需要用户输入WiFi的密码，单击开始配网即可完成配网操作。



## 手机热点配网

手机启动一个预定义热点（例如：热点名为aha，密码为12345678），设备先连接到手机热点获取路由器上网热点信息，再连接上网热点的方式。该方案需要WiFi切换相关的权限，如果没有授予权限可能会导致配网失败，建议将该方案作为备选方案。



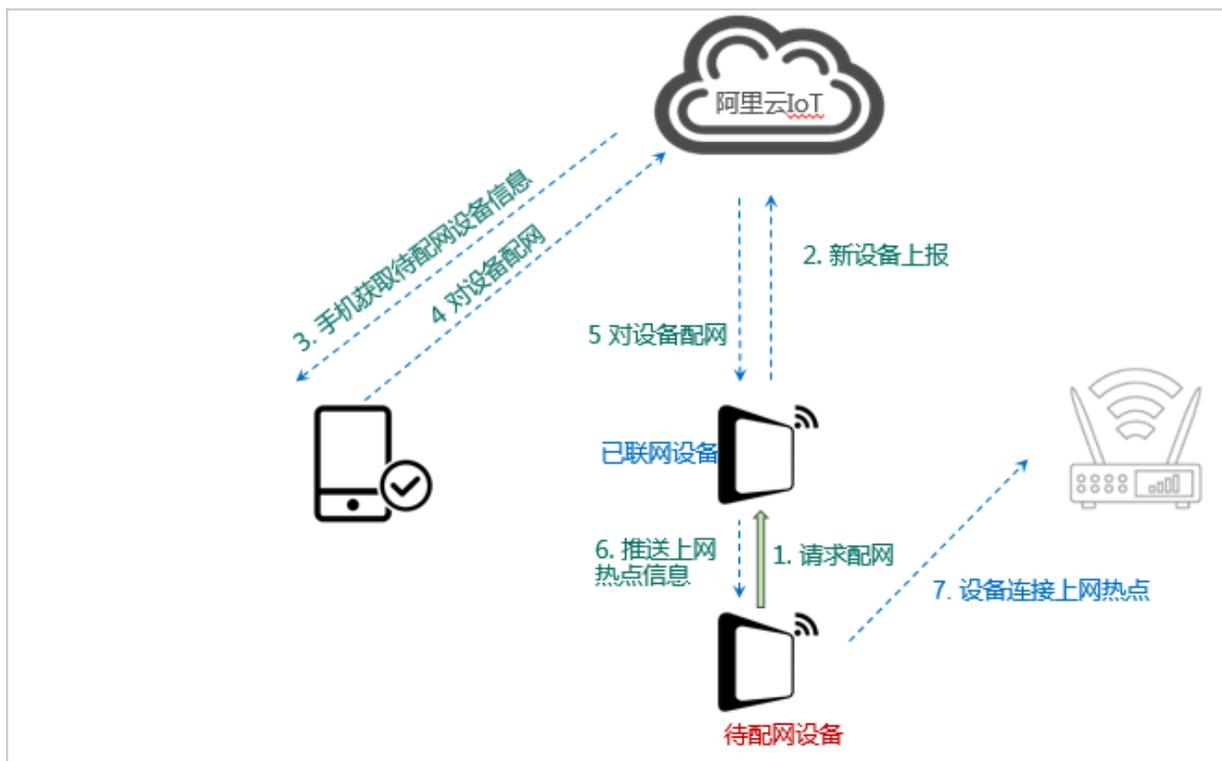
## 设备热点配网

方案简述：设备开启设备WiFi热点，手机搜索发现到热点后连接到该热点，手机将路由器上WiFi热点的SSID/密码发送给WiFi设备，然后WiFi设备使用该SSID/密码连接WiFi热点。



## 零配

通过已连接上网热点的WiFi设备，将热点信息发送给待配网设备的方式。该方案不需要用户在手机端输入WiFi热点的SSID/密码，因此称为零配。



### 摄像头扫码配网

针对拥有摄像头及二维码解码能力设备，可以通过App生成携带SSID、password等信息的二维码，摄像头根据App的二维码画面解码出信息，连接路由器，即完成配网绑定流程。开发者将SSID、password信息传入SDK后，SDK会将完整二维码信息回调并开启局域网监听，开发者需要展现二维码并监听配网结果。

二维码中携带的信息如下。

Key	描述	是否必选
v	version, 版本号, 区分为阿里的二维码规则, 当前为Ali_1。	必选
s	ssid, 家庭路由SSID	必选
p	password, 家庭路由密码	必选
b	BSSID, 家庭路由的BSSID信息, Hex String, 解决gbk编码等问题	必选
t	token, 绑定需要的Token信息, 3 Bytes random string	可选

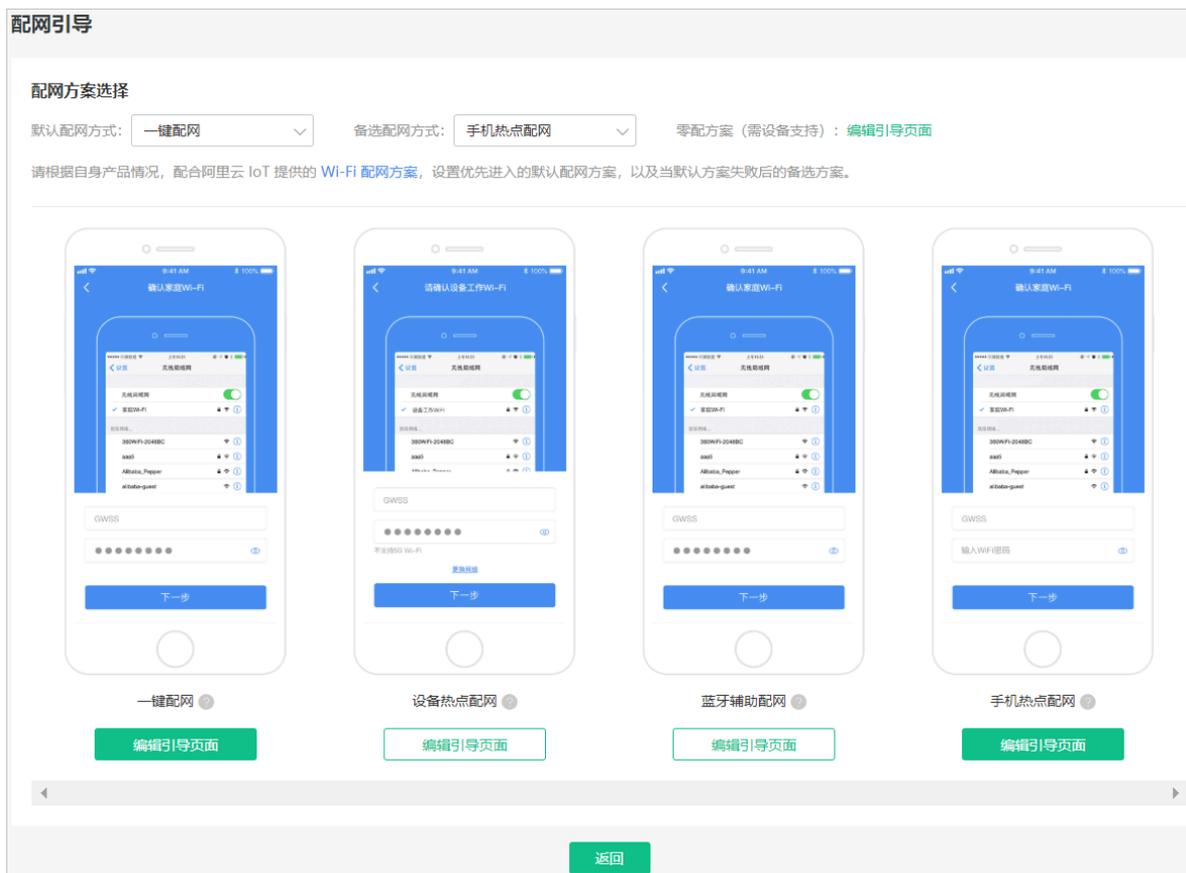
二维码按JSON规则拼接，示例如下。

```
{"v": "Ali_1", "s": "Router", "p": "12345678", "b": "EC26CAB68CC", "t": "93DAB7"}
```

### 配置配网方案和引导文案

公版App中，配网引导为必填参数，自有App中可以根据实际情况配置。

1. 进入人机交互页面。
2. 单击配网引导对应的未设置。
3. 选择默认配网方案和备选配网方案。



#### 4. 单击默认配网方案对应的编辑引导页面。

配置引导图和相关提示文案。

### 一键配网引导文案设置

手机热点配网与一键配网配置内容相同，建议使用相同引导文案

中文	英文	西班牙语	法语	俄语	德语	日语	韩语	印地语	意大利语
----	----	------	----	----	----	----	----	-----	------

**引导页**

**配网引导图**

建议尺寸：600px \* 600px,支持png、jpg、gif格式，大小不超过1M

**配网文案：**

**按钮文案：**

**帮助标题：**



#### 5. 单击备选配网方案对应的编辑引导页面。

按照相同方法，配置备选方案的引导图和相关提示文案。

### 6.3.6 配置App设备告警

控制台提供了一套设备告警的可视化功能，您可以填写告警的名称、告警内容、告警触发规则、告警等级和推送范围。当设备满足条件时，会触发该告警，并推送至App或管理后台。

#### 背景信息

设备告警为可选配置项，可根据情况决定是否配置。

#### 操作步骤

1. 进入产品-人机交互页面。
2. 单击配置告警对应的未设置。
3. 单击新建告警规则。

4. 配置详细的告警参数，并单击确定完成。

支持多种语言的告警消息配置，连接国内服务器的App默认推送中文，连接海外服务器的App默认推送英文。

新建告警
✕

\* 中文

英文

西班牙

法语

< >

**告警名称:**

请输入

**告警内容:**

告警内容支持文本和参数，如：“当前湿度已达\${TargetHumidity}，请注意除湿”。TargetHumidity为湿度值的功能标识符。

**告警规则:**

请选择 ▾

请选择 ▾

请选择 ▾

请输入

**告警等级:**

请选择 ▾

**权限范围:**

- 通知用户
  - 消息中心 (推送至消息中心, APP中可以通过查询收到)
  - 应用推送 (仅推送至手机通知栏, 无法通过APP查询到)
- 通知管理后台

确定

取消

参数名称	参数解释
告警名称	配置告警的名称。

参数名称	参数解释
告警内容	<p>可以通过文字+参数变量的方式，推送告警内容。参数变量支持<code>\${identifier}</code>的格式，其中<code>identifier</code>为设备TSL中定义的属性或事件标识符。</p> <p>示例：设备上报了数据<code>currentHumidity:50</code>，则<code>\${currentHumidity}</code>会输出50。</p> <p>告警还支持以下几种宏。</p> <ul style="list-style-type: none"> <li>· <code>#TSL_REPLACE(\${identifier})</code>：根据<code>identifier</code>配置的值，到设备对应的TSL中查找相关属性定义和事件定义。如果属性为ENUM或BOOL类型，则输出定义的内容。</li> </ul> <p>示例：一款灯的TSL中定义了名为<code>LightSwitch</code>的BOOL类型属性，其中取值0代表关闭，1代表打开。如果触发告警的事件中包含<code>LightSwitch:0</code>数据，则<code>TSL_REPLACE("LightSwitch")</code>输出的内容为关闭。</p> <ul style="list-style-type: none"> <li>· <code>#STRING_HEAD(\$DEVICE_NAME, n)</code>：输出DeviceName前n个字符。</li> <li>· <code>#STRING_TAIL(\$DEVICE_NAME, n)</code>：输出DeviceName后n个字符。</li> <li>· <code>#DEVICE_NICKNAME()</code>：使用设备的<code>nickName</code>来替换，如果<code>nickName</code>为空，则使用<code>ProductName</code>的值。</li> </ul>
告警规则	<p>支持属性、事件的简单规则，以及设备上下线的事件触发。</p> <ul style="list-style-type: none"> <li>· 事件</li> <li>· 属性</li> <li>· 设备状态</li> </ul>
告警等级	<ul style="list-style-type: none"> <li>· 提醒通知</li> <li>· 轻微问题</li> <li>· 严重告警</li> </ul>

参数名称	参数解释
权限范围	<ul style="list-style-type: none"> <li>· 通知用户： <ul style="list-style-type: none"> <li>- 消息中心（推送至消息中心，App中可以通过查询收到）</li> <li>- 应用推送（仅推送至手机通知栏，无法通过App查询到）</li> </ul> </li> <li>· 通知后台：厂家可以在运营中心 &gt; 设备运维 &gt; 告警中心中进行查看。</li> </ul>

### 6.3.7 配置App界面

消费级的IoT设备，特别是和生活息息相关的智能家电、智能家居设备，往往需要搭配一个手机App页面，对智能设备进行控制和查看。平台针对这类设备，提供了App界面。

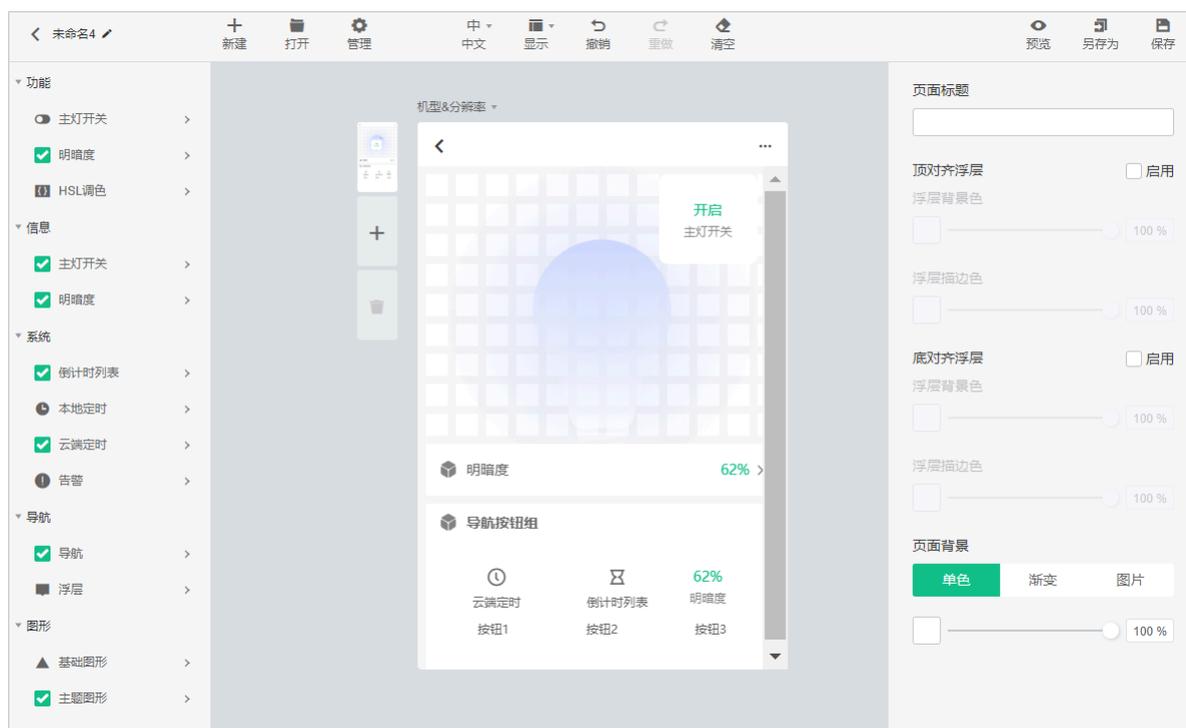
#### 背景信息

平台提供了两种App界面配置方式，其中，公版App只需设置面板主题即可。

- 标准界面：针对部分常见品类，我们提供了一套完整的界面，如灯、开关、插座，对功能定义有所限制。



- **配置化界面：**通过拖拽和配置化，即可搭建一套设备界面，灵活性高，不限制设备的功能定义。



### 6.3.8 配置App自动化和定时

选择智能场景、自动化、云端定时、本地定时和本地倒计时中可用的设备功能。

#### 背景信息

此处会列出该产品所有可以支持联动的功能，请勾选您希望在场景、定时中出现的触发条件和执行任务。勾选后，就可以在场景和定时中让终端用户去设置。可理解为：如果触发设备的某个条件，就执行某个动作。

平台还支持本地定时和本地倒计时功能。定时和倒计时功能，设备端需要配置开发才能使用，可参见[#unique\\_54](#)中的具体功能开发。

使用公版App，可通过扫码下载公版App体验产品界面。

#### 操作步骤

1. 进入产品-人机交互页面。
2. 单击配置自动化与定时对应的修改。

### 3. 配置联动场景。

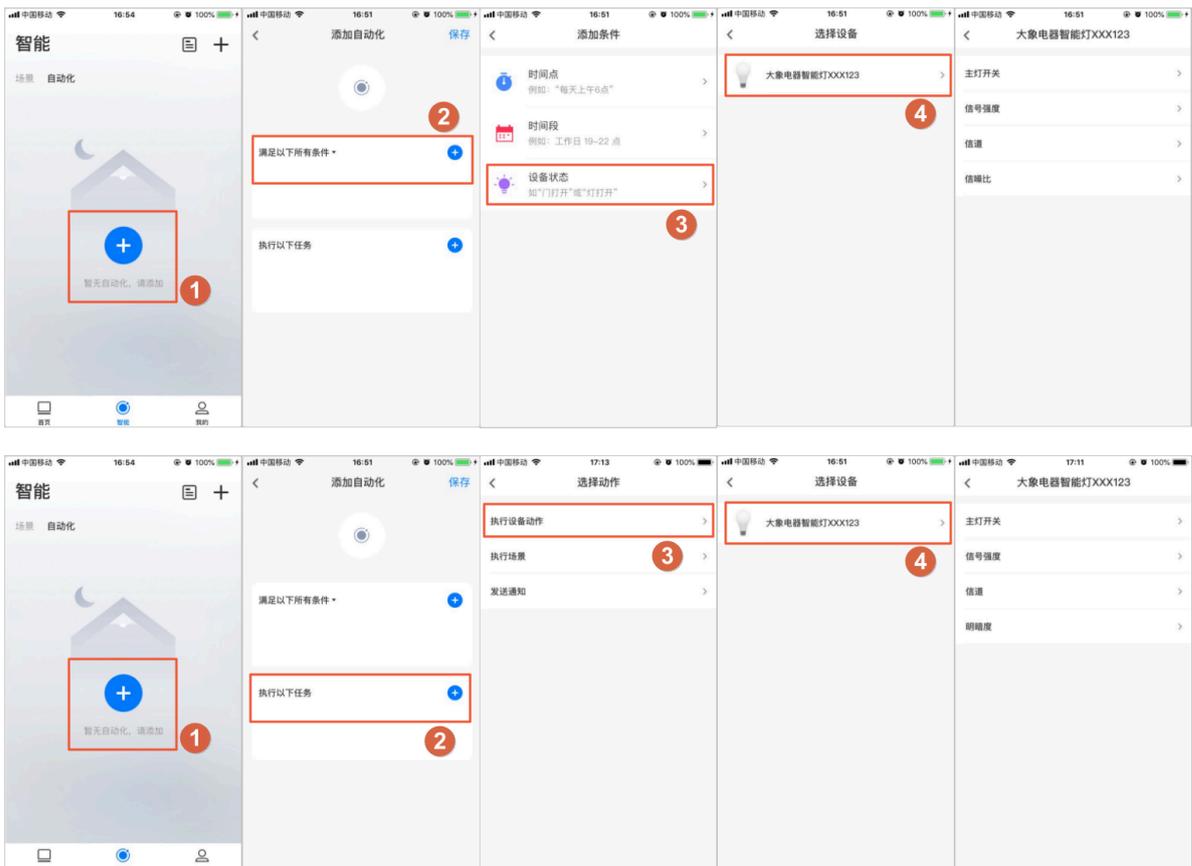
**自动化与定时**

产品联动功能设置 功能参数

功能名称	智能场景	本地定时	本地倒计时
主灯开关	<input checked="" type="checkbox"/> 作为条件 <input checked="" type="checkbox"/> 作为执行	<input checked="" type="checkbox"/> 开启	<input checked="" type="checkbox"/> 开启
信号强度	<input checked="" type="checkbox"/> 作为条件 <input checked="" type="checkbox"/> 作为执行	<input checked="" type="checkbox"/> 开启	
信道	<input checked="" type="checkbox"/> 作为条件 <input checked="" type="checkbox"/> 作为执行	<input type="checkbox"/> 开启	
信噪比	<input checked="" type="checkbox"/> 作为条件 <input type="checkbox"/> 作为执行	<input type="checkbox"/> 开启	
明暗度	<input type="checkbox"/> 作为条件 <input checked="" type="checkbox"/> 作为执行	<input type="checkbox"/> 开启	
故障上报	<input type="checkbox"/> 作为条件 <input type="checkbox"/> 作为执行		
设置主灯开关倒计时	<input type="checkbox"/> 作为条件 <input type="checkbox"/> 作为执行		

保存 返回

勾选的功能，会出现在App的场景配置中，供终端用户使用。



## 6.3.9 配置接入第三方平台

平台支持对接第三方设备，主要有天猫精灵、Amazon Alexa、Google Assistant、IFTTT等。前三种是语音设备，可通过语音直接对设备进行控制，IFTTT是第三方服务联动平台。

### 背景信息

中国站仅支持天猫精灵接入，国际站可支持Amazon Alexa、Google Assistant语音设备接入和IFTTT第三方平台接入。详细介绍和接入教程如下。

- 天猫精灵

[../DNICMS19102448/ZH-CN\\_TP\\_1436549\\_V1.dita#task\\_1813802](#)

[../DNICMS19102448/ZH-CN\\_TP\\_1437036\\_V1.dita#task\\_1814475](#)

- Amazon Alexa

[../DNICMS19102448/ZH-CN\\_TP\\_1436621\\_V1.dita#task\\_1813986](#)

[../DNICMS19102448/ZH-CN\\_TP\\_1437087\\_V1.dita#task\\_1814573](#)

- Google Assistant

[../DNICMS19102448/ZH-CN\\_TP\\_1436757\\_V1.dita#task\\_1814166](#)

[../DNICMS19102448/ZH-CN\\_TP\\_1448968\\_V1.dita#task\\_1830019](#)

- IFTTT

[../DNICMS19102448/ZH-CN\\_TP\\_1437024\\_V1.dita#task\\_1814419](#)

### 操作步骤

1. 进入产品-人机交互页面。
2. 单击对应的第三方设备名称对应的去配置。
3. 开启第三方设备接入开关。

选择支持品类，可查看设备的状态属性映射和设备控制指令。



说明：

Amazon Alexa、Google Assistant、IFTTT是第三方收费设备，需要购买相应的服务才可以接入使用。

## 6.4 批量投产

产品开发完成后，需要发布产品，购买激活码，才能进行量产。

### 发布产品

1. 进入产品-批量投产页面。
2. 确定产品信息。
3. 勾选“我已充分了解并同意《开发者须知》和《授权协议》”。
4. 单击开发完成。

发布产品完成后，产品的状态变为已发布。

状态为已发布的产品不可修改和删除。此时若要修改或删除产品，需先单击撤回发布返回至开发中状态，再进行修改或删除。



#### 说明:

如果使用公版App，在产品量产之后，需等待产品名称审核通过后产品才能上架。若产品名称审核不通过则会被驳回，并要求重新命名。查看[设置产品名称](#)。

### 购买激活码

在购买激活码环节可单击去购买，直接跳转至购买设备激活码页面。

更多进入购买激活码页面的方式，参见[购买激活码](#)。

## 6.5 创建自有App

每个项目中，允许创建一个自有品牌App。

### 背景信息

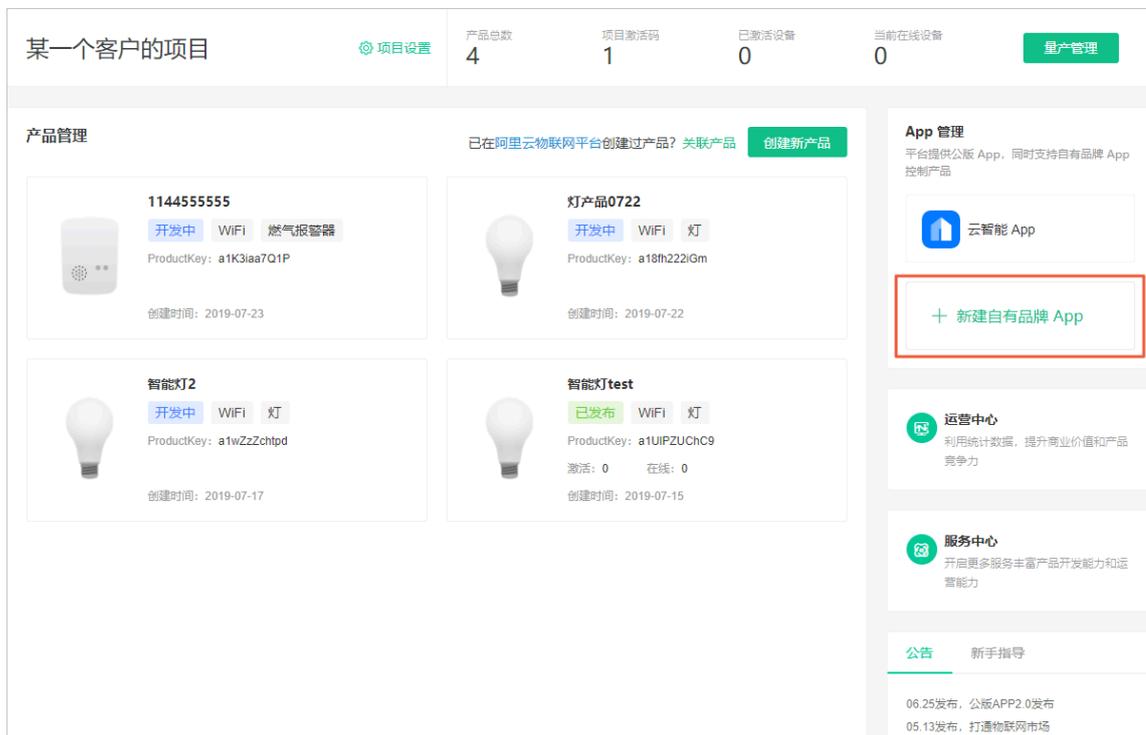
每个App应用包括4个实际的App Key，可分别用于您的App的测试开发和正式发布。

iOS	开发版（用于开发调试）
	用户版（用于正式对外发布）
Android	开发版（用于开发调试）
	用户版（用于正式对外发布）

### 操作步骤

## 1. 创建App。

a) 在项目主页面右侧，单击新建自有品牌App。



b) 配置应用名称和应用包名等信息，并单击确定完成。

配置的应用名称和应用包名等信息，后期可修改。

新建自有品牌 App ×

---

**应用名称**

请输入20字以内应用名称0/20

**应用包名 (正式版) ?**

请输入iOS应用唯一标识, 要求格式com.xxx.xxx

请输入Android应用唯一标识, 要求格式com.xxx.xxx

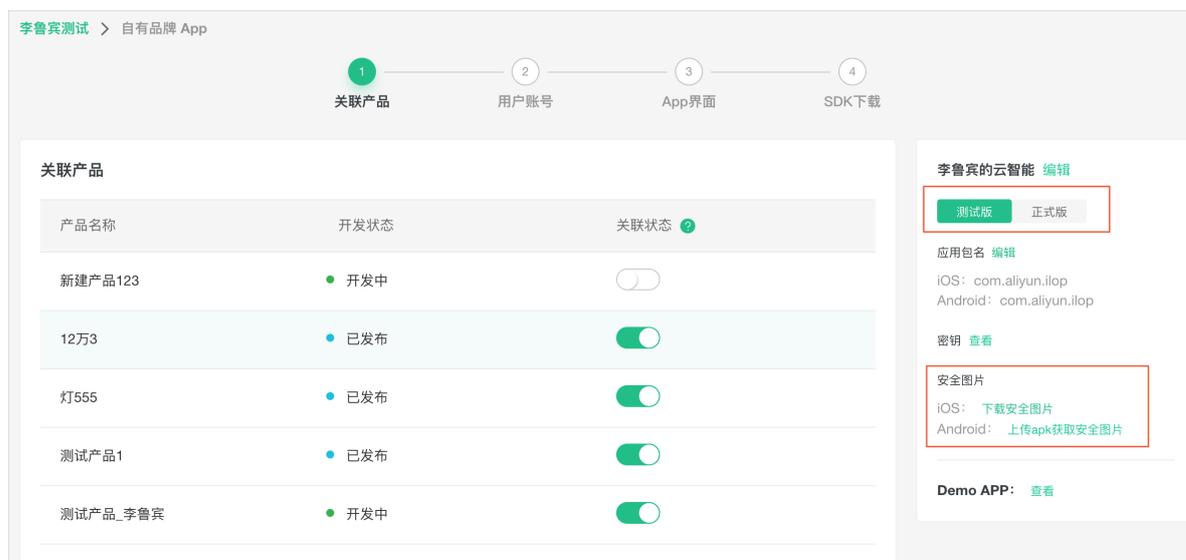
**应用包名 (测试版) ?** 参数与正式版保持一致或 [重新填写](#)

请输入iOS应用唯一标识, 要求格式com.xxx.xxx

请输入Android应用唯一标识, 要求格式com.xxx.xxx

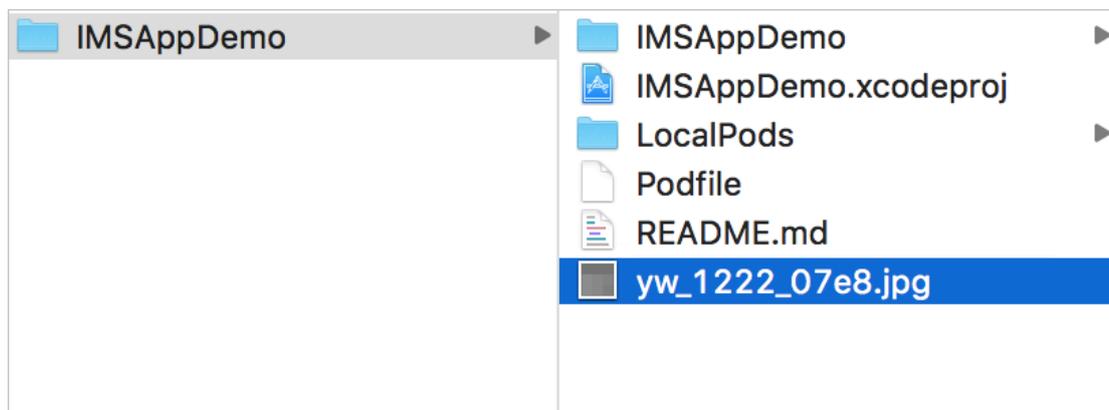
确定 取消

## 2. 获取安全图片。



### · 获取iOS安全图片

- 下载安全图片，文件名为`yw_1222_07e8.jpg`。
- 将安全图片放到工程根目录中。



- 打开xcode工具，将安全图片拖至`xcodeproj`根目录下。

- iOS端为了安全性，工程的Bundle Identifier与创建自有App时输入的bundleID（包名）要保持一致。

### · 获取Android安全图片

- 下载安全图片。

下载Android图片之前需要先上传APK包（即.apk格式的文件），可通过Android Studio工具生成。

下载后的安全图片的文件名为`yw_1222_xxxx.jpg`。放置在Android工程目录中的`src\oversea\res\drawable`下。

#### b. 配置签名。

Android 端上，安全图片是需要和签名搭配使用。

确认`build.gradle`文件中包含以下内容，并确保与生成APK包中配置的内容一致。

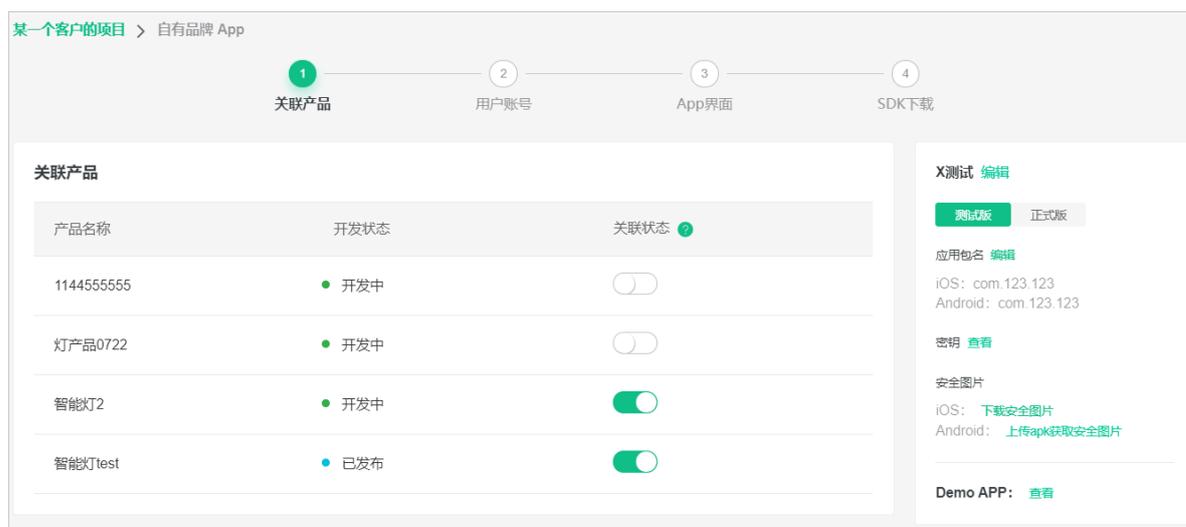
```
android{
    signingConfigs {
        debug {
            storeFile file("./debug.keystore")
            storePassword '123456'
            keyAlias 'android'
            keyPassword '123456'
        }
    }
    buildTypes{
        debug {
            signingConfig signingConfigs.debug
        }
    }
}
```

### 3. 打开产品的关联状态开关。

关联产品列表中的产品，为当前项目主页面中显示的产品。关联后，在App端调用待配网设备的API时，可以获取到该产品的信息。

产品的开发状态决定产品是否具备进入正式版App的能力。

- 若开关开启，且该产品开发中，则默认展示在测试版App的配网入口。
- 若开关开启，且该产品已发布，则默认展示在正式版App的配网入口。
- 若开关关闭，该产品状态无论是开发中还是已发布，测试版和正式版App配网入口都隐藏。



某一个客户的项目 > 自有品牌 App

1 关联产品    2 用户账号    3 App界面    4 SDK下载

产品名称	开发状态	关联状态
1144555555	● 开发中	<input type="checkbox"/>
灯产品0722	● 开发中	<input type="checkbox"/>
智能灯2	● 开发中	<input checked="" type="checkbox"/>
智能灯Test	● 已发布	<input checked="" type="checkbox"/>

X测试 [编辑](#)

测试版     正式版

应用包名 [编辑](#)

iOS: com.123.123  
Android: com.123.123

密钥 [查看](#)

安全图片

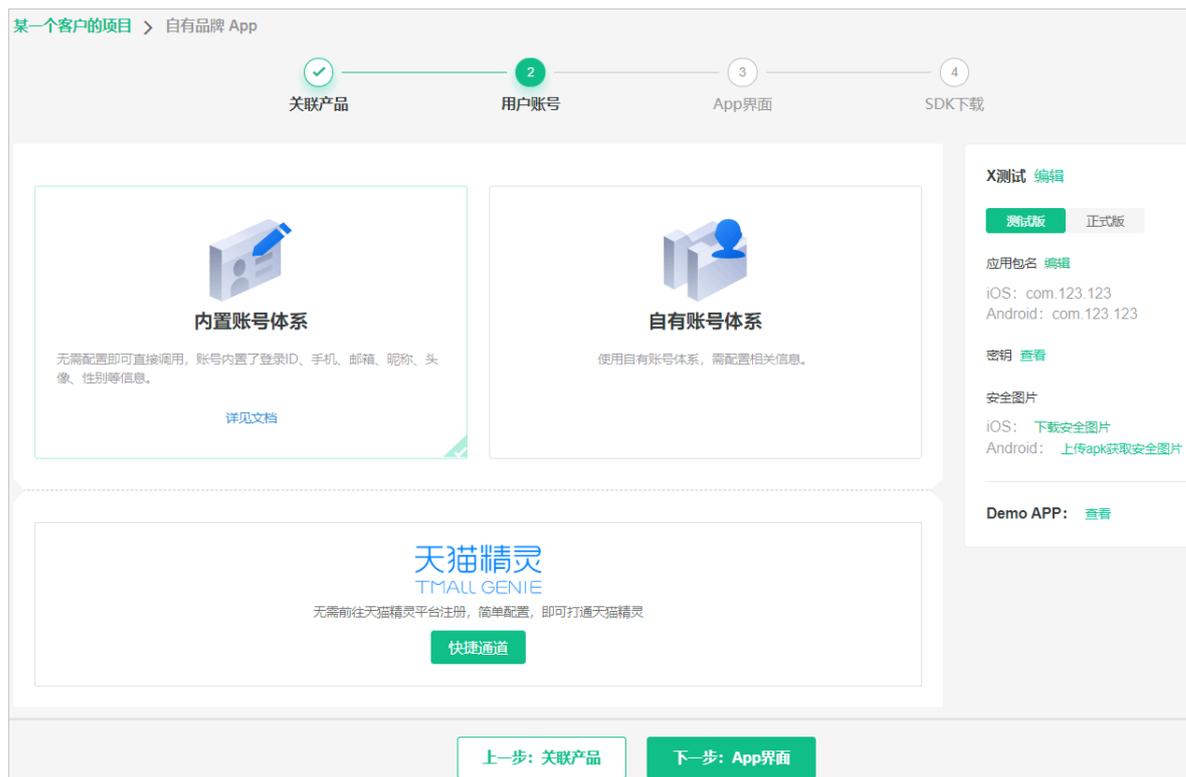
iOS: [下载安全图片](#)  
Android: [上传apk获取安全图片](#)

Demo APP: [查看](#)

#### 4. 配置用户账号。

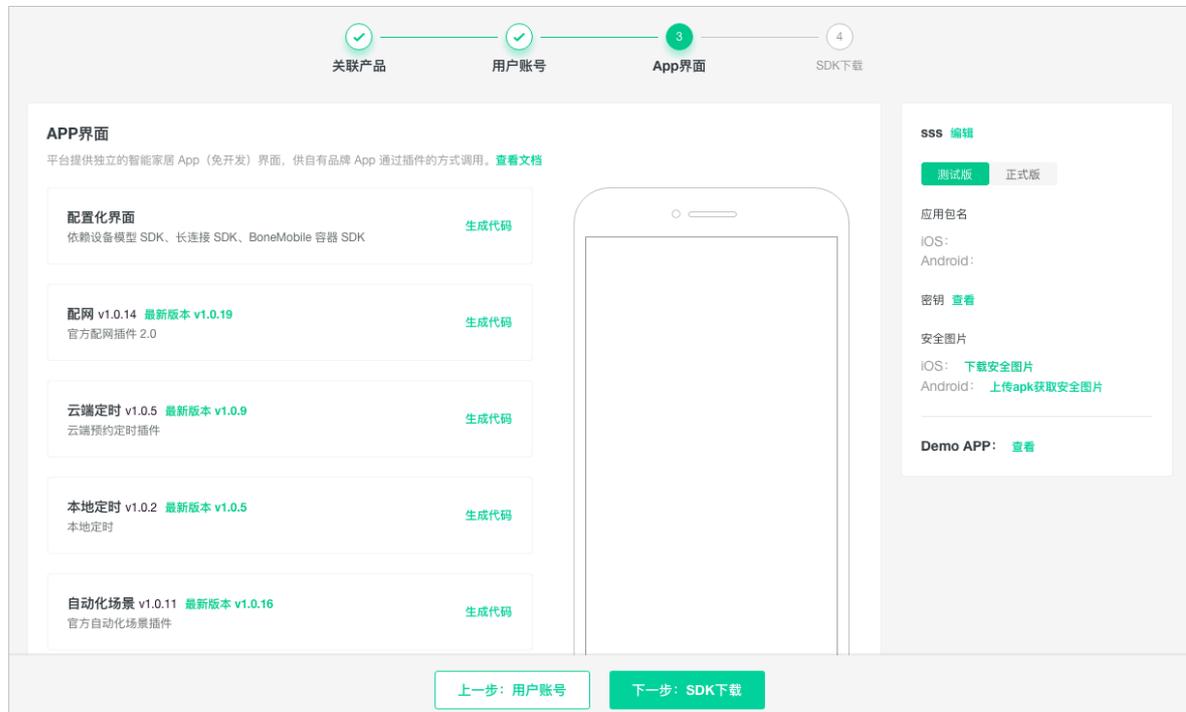
用户账号支持两种账号体系：**内置账号体系**和**自有账号体系**。

- 如果开发者没有自己的账号体系，可以采用**内置账号体系**，内置账号体系无需配置，已经内置在SDK套餐包中，提供文档调用方法。
- 如果开发者有自己的账号体系，可以采用**自有账号体系**，若勾选自有账号体系，则必须填写页面中的配置信息，否则自有账号将无法正常使用。



## 5. 配置App界面。

平台提供了一些常用的免开发界面，如配网、设备界面、定时、自动化场景、意见反馈等智能生活类App常用的页面模块。单击生成代码，将生成的代码嵌到自己的代码中，即可实现该功能模块的调用。



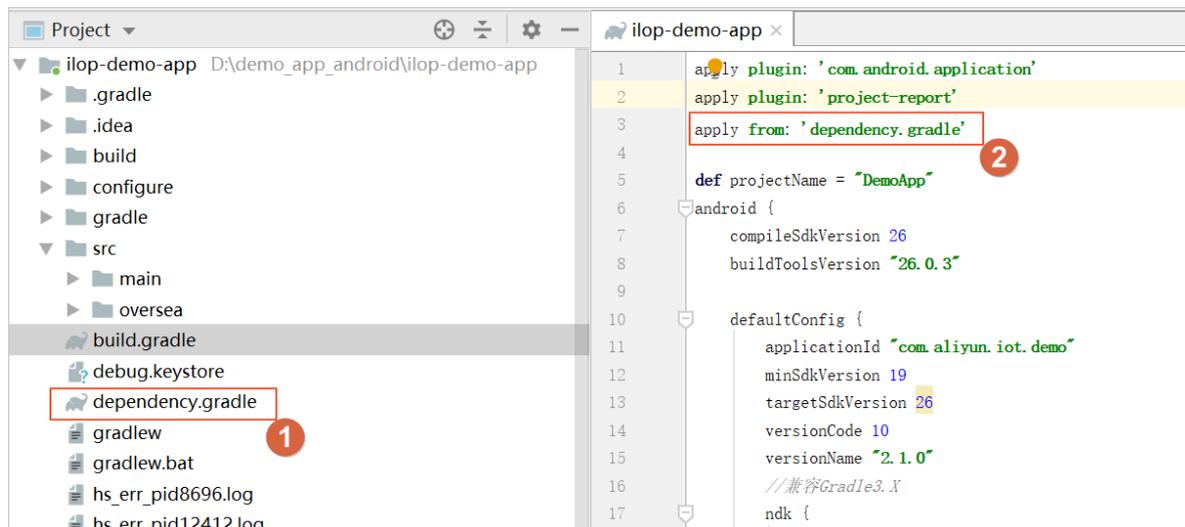
## 6. SDK下载。

根据关联的产品，选择合适的“API Level”（SDK套餐包版本），并勾选和配置需要的SDK套餐项。平台会自动生成的SDK配置包。您只需下载集成开发即可。

API Level是SDK套餐项的功能版本等级，用来解决新老SDK套餐项不兼容问题，每个API Level包含一组SDK套餐项，不同API Level的SDK套餐项功能不同，先选择适合的API Level，一般建议选择最新的API Level。

## 7. Android集成SDK。

推荐开发工具为Google官方的Android Studio（建议使用最新版本3.0.x）。



a) 将下载的Android SDK Pom *dependency.gradle*文件放到Module的根目录下。

b) 在模块根目录的*build.gradle*文件中，添加以下代码。

```
apply from: 'dependency.gradle'
```

## 8. iOS集成SDK。

本SDK采用cocoapods。

a) 将下载的IOS SDK *Podfile*文件放到工程文件夹根目录。

b) 修改文件中的target名称与xcode工程的设置一致。

c) 执行pod update。

### 后续步骤

更多操作，查看开发指南中App端开发。

# 7 账单中心

## 7.1 购买激活码

激活码是设备连接阿里云IoT的唯一凭证，请根据以下步骤提前购买设备激活码。每台设备需要购买一个激活码，不可重复使用。

### 背景信息

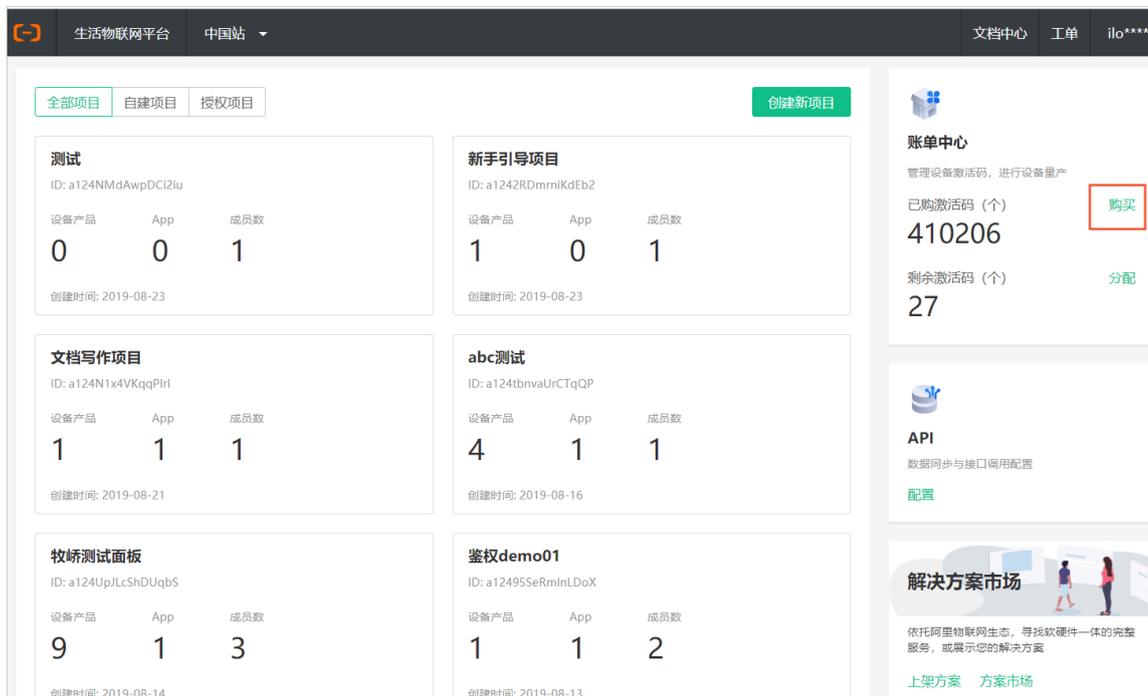
购买激活码的入口：

- 量产管理
- 账单中心

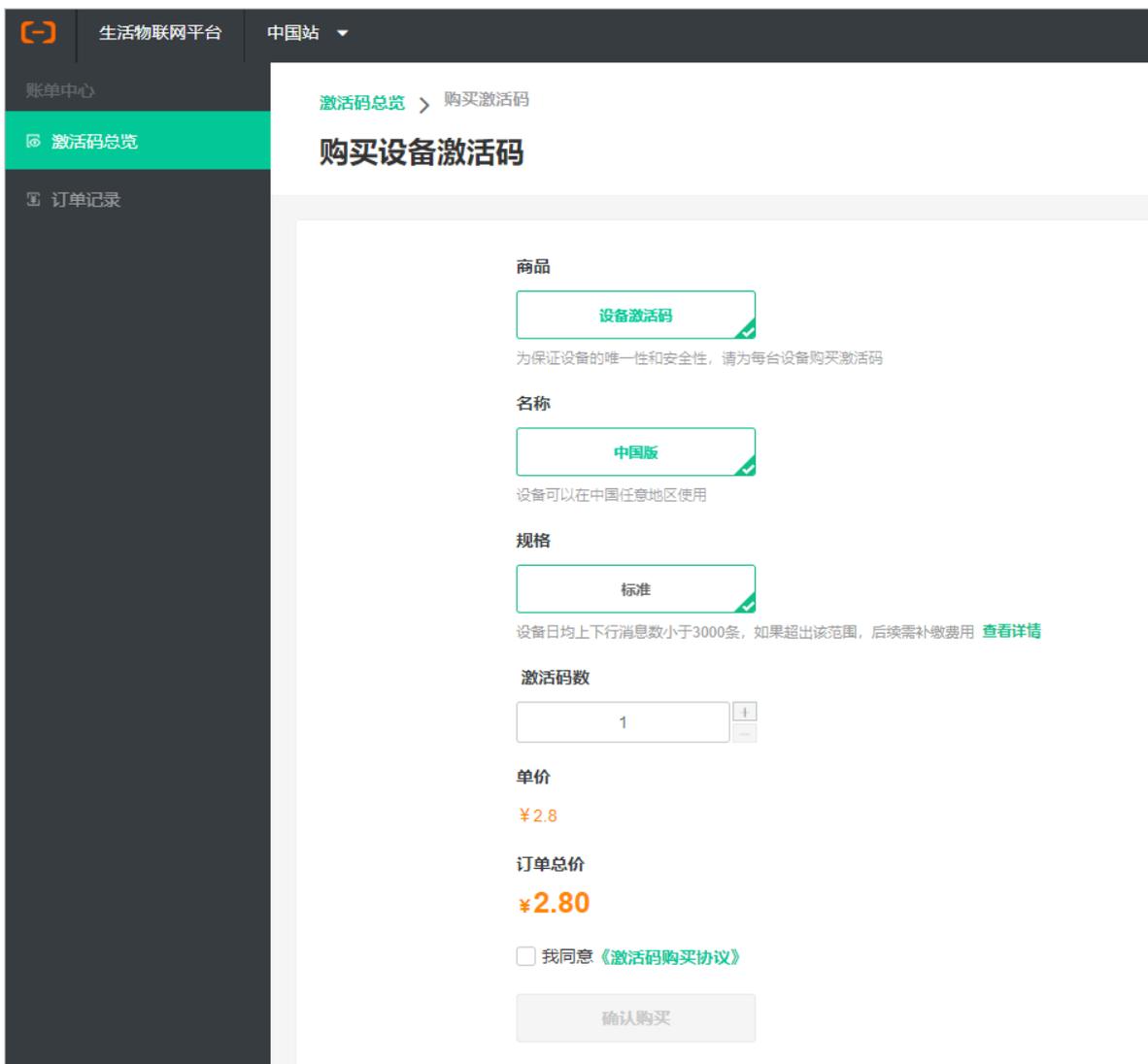
### 操作步骤

1. 选择以下任意方式进入激活码购买页面。

- 在量产管理中单击购买激活码，进入购买设备激活码页面。
- 在控制台主页面单击账单中心的购买中，进入购买设备激活码页面。



## 2. 配置购买激活码的数量，并根据提示进行支付。



中国站与国际站设备激活码价格不一致，需要分开购买。详细参见[#unique\\_22](#)。

## 7.2 分配激活码

为了数据安全，用户可以创建多个项目。并可以统一购买设备激活码，分配到不同的项目中。

### 操作步骤

1. 在生活物联网平台首页单击账单中心。
2. 在激活码总览中单击分配到项目。
3. 下拉选择项目名称，并分配激活码数量。

## 7.3 查看消费记录

生活物联网平台的所有消费明细，可通过账单中心进行查看。

### 操作步骤

1. 登录生活物联网平台的控制台。
2. 在主页面右上侧单击账单中心。
3. 单击订单记录。

该页面展示所有的消费明细。



商品类型	订单号	名称	规格	单价	数量(个)	原价	应付金额	操作
设备激活码	201907151625514738	中国版	标准	¥2.80	1	¥2.80	¥2.80	<a href="#">查看</a>
服务	201907022004203429	视频服务	-	¥0.99	1	¥0.99	¥0.99	<a href="#">查看</a> <a href="#">付款</a>
设备激活码	201907021948427612	中国版	标准	¥2.80	1	¥2.80	¥2.80	<a href="#">查看</a> <a href="#">付款</a>
设备激活码	201907021948067897	中国版	标准	¥2.80	1	¥2.80	¥2.80	<a href="#">查看</a> <a href="#">付款</a>
服务	201906271520509849	地理分布	-	¥3000.00	1	¥3,000	¥3,000	<a href="#">查看</a> <a href="#">付款</a>
服务	201906271456131536	视频服务	-	¥0.99	1	¥0.99	¥0.99	<a href="#">查看</a> <a href="#">付款</a>

## 8 量产中心

### 8.1 配置激活码烧录方式

平台要求每台设备需要烧录平台颁发的身份信息，才能实现设备注册上云。

#### 前提条件

- 已完成产品开发，并发布成功。
- 已购买一定数量的激活码。

#### 背景信息

激活码是设备连接阿里云IoT的唯一凭证，平台提供的烧录激活码方式有以下两种。

- 一机一密（推荐）：平台为每台设备颁发一个密钥（DeviceSecret），设备量产时，每台设备需要烧录唯一的密钥，安全性高，推荐使用。
- 一型一密：平台为每种产品型号颁发一个密钥（ProductSecret），设备量产时，同一个型号的设备，仅需烧录相同的密钥。

为了保障您的设备不会被黑客攻击和伪造，平台要求一型一密的设备必须预注册每台设备的DeviceName（如MAC地址、SN或IMEI号等）。当设备首次连云时，平台会通过该DeviceName进行身份核对。



#### 说明：

由于网关会代理Zigbee子设备的联网，平台针对Zigbee设备，推荐使用一型一密的烧录方式，即网关仅需代理子设备的ProductKey，即可帮助子设备连接云端。

两种烧录方式详细对比如下。

对比项	一机一密	一型一密
设备端烧录信息	设备证书，即： ProductKey（产品型号） DeviceName（设备名称） DeviceSecret（设备密码）	ProductKey（产品型号） ProductSecret（产品密码） DeviceName（设备名称，一般为设备本身的MAC地址、SN等）
安全性	较高	较低
是否有配额限制	有，单个产品50万上限	有，单个产品50万上限
控制台如何开启	默认开启	量产管理 > 烧录方式 > 开启一型一密

对比项	一机一密	一型一密
控制台如何关闭	默认一机一密，无法关闭	量产管理 > 烧录方式 > 取消一型一密   说明： 高风险，请谨慎操作。
DeviceName生成方式	自动生成/手动上传	手动上传

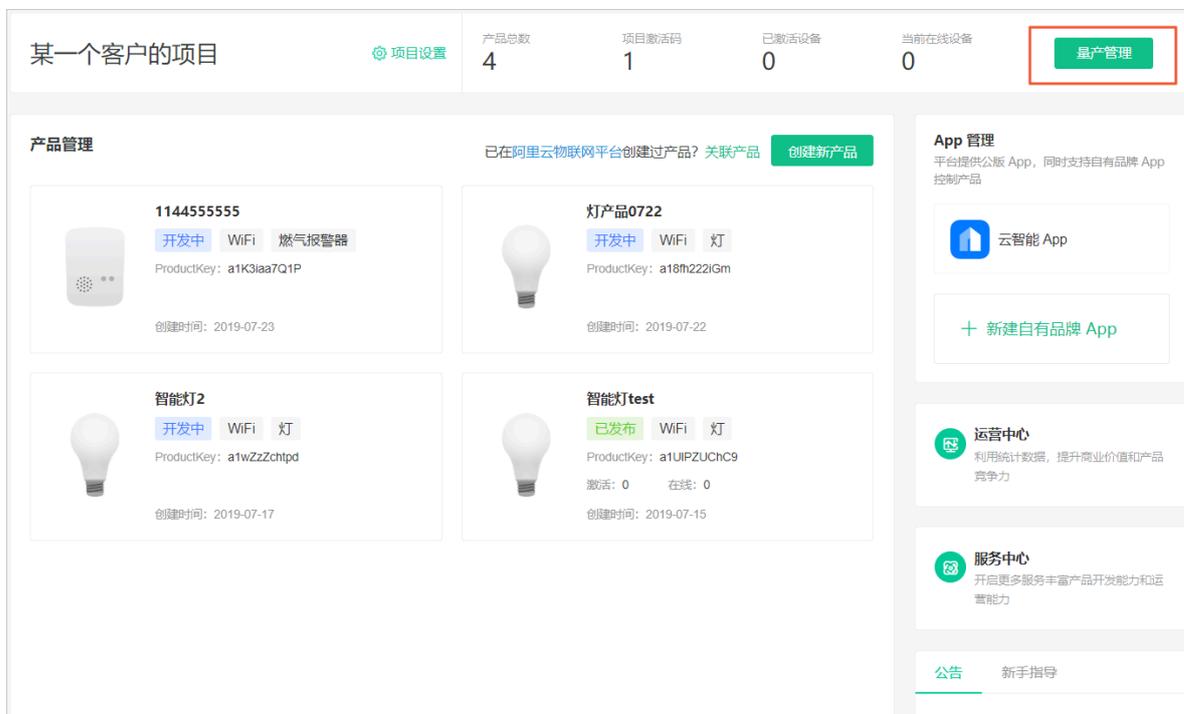


说明：

- 为了避免设备ProductKey和ProductSecret泄漏导致设备被仿冒和攻击，云端不允许已激活的设备重新获取设备证书，要求开发者/厂商对设备端的设备证书进行持续化存储和安全防护，恢复出厂设置时切勿把设备证书信息清除。
- 关闭一型一密后，会导致已经按照一型一密烧录的设备无法再激活联网（但不影响已联网设备）。如果在设备量产后关闭一型一密，可能导致消费者投诉和经济损失，请谨慎操作。

## 烧录一机一密

### 1. 在项目主页面单击量产管理。



某一个客户的项目

项目设置

产品总数 4    项目激活码 1    已激活设备 0    当前在线设备 0

**量产管理**

产品管理

已在阿里云物联网平台创建过产品？ [关联产品](#) [创建新产品](#)

1144555555  
开发中 WiFi 燃气报警器  
ProductKey: a1K3laa7Q1P  
创建时间: 2019-07-23

灯产品0722  
开发中 WiFi 灯  
ProductKey: a18f222iGm  
创建时间: 2019-07-22

智能灯2  
开发中 WiFi 灯  
ProductKey: a1wZzchtPd  
创建时间: 2019-07-17

智能灯test  
已发布 WiFi 灯  
ProductKey: a1UIPZUChC9  
激活: 0    在线: 0  
创建时间: 2019-07-15

App 管理  
平台提供公版 App，同时支持自有品牌 App 控制产品

云智能 App

+ 新建自有品牌 App

运营中心  
利用统计数据，提升商业价值和产品竞争力

服务中心  
开启更多服务丰富产品开发能力和运营能力

公告 新手指导

2. 选择状态为已发布的产品，单击操作列的批量投产。

烧录方式默认是一机一密，无需修改。



### 3. 配置激活码生成方式，并根据选择进行相应的操作。

一机一密的激活码生成方式，可选择自动生成，或批量上传。单产品可添加设备数量的上限为50万。

- 自动生成：平台会自动生成DeviceName和DeviceSecret，客户可以烧录到每台设备中。自动生成方式仅需配置烧录数量即可。

### 批量投产

量产设备

智能灯test

通讯方式: WiFi Product Key: a1UIPZUChC9

所用激活码类型

设备激活码

激活码规格

标准

日均消息量小于3000条

烧录方式

一机一密(推荐)

每台设备需要烧录唯一的激活码 (一组ProductKey、DeviceName和DeviceSecret) , 安全等级高

激活码生成方式

自动生成 批量上传

系统自动生成全局唯一的DeviceName和DeviceSecret

量产数量

1   预计需要时间2秒

最多量产10,000个

确定 取消

- 批量上传：预先将需要激活的设备标识（如MAC地址、SN或IMEI号等）作为DeviceName录入到excel中。将excel上传到平台中，平台会自动为每个DeviceName生成一个DeviceSecret。

### 批量投产

量产设备

智能灯test

通讯方式: WiFi Product Key: a1UIPZUChC9

所用激活码类型

设备激活码

激活码规格

标准

日均消息量小于3000条

烧录方式

一机一密(推荐)

每台设备需要烧录唯一的激活码（一组ProductKey、DeviceName和DeviceSecret），安全等级高

激活码生成方式

自动生成 批量上传

单个文件不超过2M，一次最多包含10,000条记录，[下载 csv模板](#)

上传文件

确定 取消

4. 单击确定完成。

#### 烧录一型一密

1. 在项目主页面单击量产管理。

## 2. 选择状态为已发布的产品，单击烧录方式列的一机一密。

项目激活码总数	剩余可用激活码	已量产激活码
1	1	0

产品名称	Product Key	通讯方式	状态	已量产(个)	烧录方式	操作
1144555555	a1K3iaa7Q1P	WiFi	● 开发中	0	一机一密	操作
灯产品0722	a18fh222iGm	WiFi	● 开发中	0	一机一密	操作
智能灯2	a1wZzZchtpd	WiFi	● 开发中	0	一机一密	操作
智能灯Test	a1UIPZUChC9	WiFi	● 已发布	0	一机一密	操作

## 3. 选择一型一密，并在弹出提示框中单击确定。

此时需要进行身份认证，根据提示操作，认证成功后，即可完成烧录方式的变更。

## 4. 单击操作列的批量投产。

## 5. 配置激活码生成方式。

一型一密的激活码生成方式只有批量上传。单产品可添加设备数量的上限为50万。

预先需要激活的设备标识（如MAC地址、SN或IMEI号等）作为DeviceName录入到excel中。将excel文件上传到平台中，平台会自动为每个DeviceName生成一个DeviceSecret。

### 批量投产

量产设备

智能灯test

通讯方式: WiFi Product Key: a1UIPZUChC9

所用激活码类型

设备激活码

激活码规格

标准

日均消息量小于3000条

烧录方式

一机一密(...) 一型一密 ?

同一批设备可以烧录相同的ProductKey和ProductSecret，但需要预先批量上传Device Name（如MAC地址、SN、IMEI等）

激活码生成方式

批量上传

单个文件不超过2M，一次最多包含10,000条记录，[下载 csv模板](#)

上传文件

确定 取消

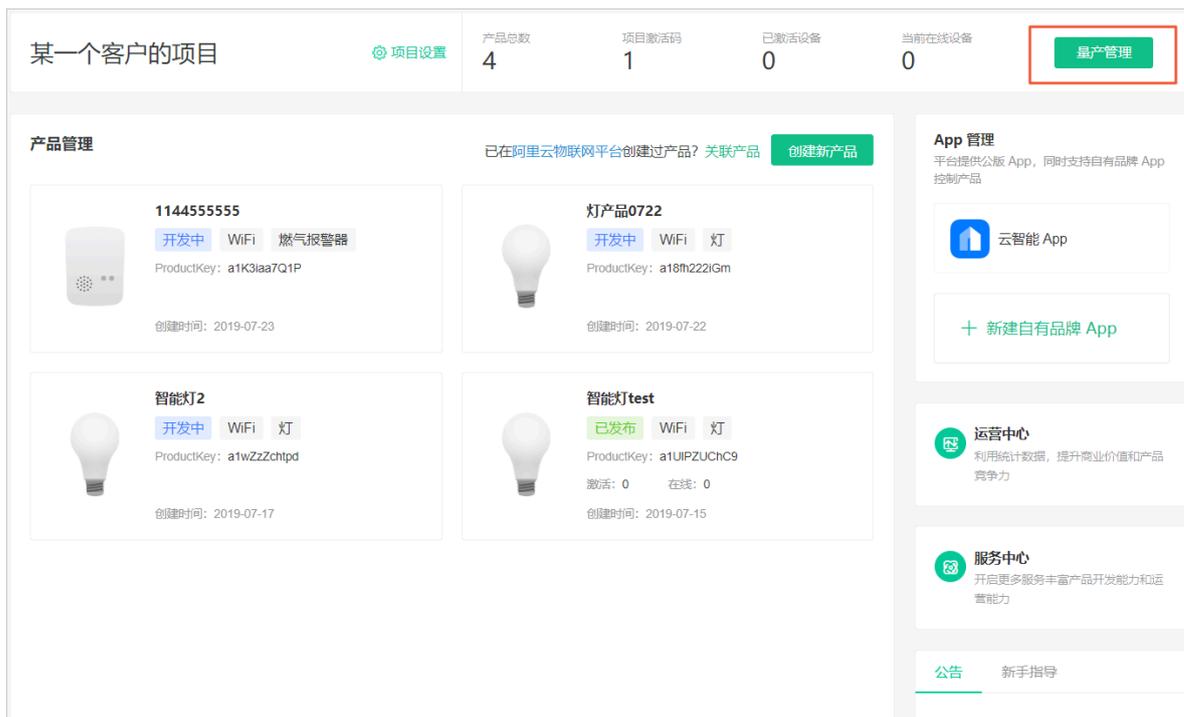
6. 单击确定完成。

## 8.2 查看量产记录

所有产品的量产记录，可在量产管理中查看。

### 操作步骤

1. 在项目主页面单击量产管理。



2. 选择量产管理 > 量产记录。

可查看详细的质量信息。

量产批次ID	产品名称	Product Key	通讯方式	消耗激活码类型	激活码规格	提交数量	完成数量	提交时间	操作
212930	水宝牌智能金灯	a2ULH4d...	WiFi	设备激活码	标准	1	1	2019-06-20 10:52:26	查看
212922	水宝牌智能金灯	a2ULH4d...	WiFi	设备激活码	标准	1	1	2019-06-17 17:32:29	查看
212844	语音控制窗帘	a2dMSCa...	WiFi	设备激活码	标准	1	1	2019-05-31 19:56:41	查看
212843	扫地机器人	a2JpynJ3...	WiFi	设备激活码	标准	1	1	2019-05-31 16:41:36	查看
212842	语音控制开关	a22bnWjr...	WiFi	设备激活码	标准	1	1	2019-05-31 15:34:21	查看
212841	语音控制香薰机	a2zSF7xc...	WiFi	设备激活码	标准	1	1	2019-05-31 15:34:05	查看
212840	水宝牌智能金灯	a2ULH4d...	WiFi	设备激活码	标准	1	1	2019-05-31 10:12:15	查看

## 9 服务中心

生活物联网平台还为用户提供了很多额外的服务，丰富了平台的功能。收费服务需开通后，相关功能在页面上才会显示。

### 功能入口

选择具体项目名称，在项目主页面右侧单击服务中心。

### 服务介绍

平台主要提供了以下服务。其中未特殊说明的服务，都属于免费免开通服务，用户可以直接使用。

- 数据统计
  - 数据大屏
  - 地理分布：需付费开通
  - 设备激活数据
  - 设备活跃数据
  - 用户新增数据
  - 用户活跃数据

- 通用服务
  - 告警中心
  - OTA升级
  - ID<sup>2</sup>安全服务：需提交申请开通
  - 用户反馈
- 精品应用服务
  - 智能门锁：需付费开通，参见[#unique\\_66](#)
  - Link Visual：限时免费开通，参见[#unique\\_67](#)
- 语音操控
  - 天猫精灵：仅中国站中显示
  - Amazon Alexa：仅国际站中显示，需付费开通
  - Google Home：仅国际站中显示，需付费开通
  - IFTTT：仅国际站中显示，需付费开通

## 10 运营中心

### 10.1 能力介绍

每个项目有一个独立的运营中心，提供了一个免开发的运营管理后台，包括设备的运维监管，和用户的运营统计，以及DataV大屏。目前支持中英文的切换。

#### 功能入口

选择具体项目名称，在项目主页面右侧单击服务中心。

The screenshot displays the '国内阿达项目勿删' (Domestic A-da Project Do Not Delete) dashboard. At the top, it shows project statistics: 6 total products, 10 activated projects, 0 activated devices, and 0 currently online devices. A '项目设置' (Project Settings) button is visible. Below this, the '产品管理' (Product Management) section lists six products with their status (e.g., '已发布' or '开发中'), features (e.g., WiFi, 插座), and creation dates. To the right, the 'App 管理' (App Management) section offers options for '云智能 App' and '自有品牌APP'. At the bottom right, the '运营中心' (Operation Center) and '服务中心' (Service Center) are highlighted with red boxes, indicating the entry point for the user's focus. A '公告' (Notice) section at the very bottom mentions a public APP 2.0 release on 06.25.

#### 涵盖内容

运营中心包含以下内容。

· 信息总览



## · 设备运维

设备管理

全部产品

概览

设备总数 4	激活设备 1	当前在线 0
--------	--------	--------

设备列表

请输入DeviceName 搜索

<input type="checkbox"/>	DeviceName	所属产品	节点类型	状态	最后上线时间	操作
<input type="checkbox"/>	9uWfIDo8fPvHvaYW7alz	蜂窝联网方式问题验证	设备	● 未激活	-	<a href="#">激活凭证</a> <a href="#">查看</a>
<input type="checkbox"/>	bovGzkJcHOhTv8kcS72f	预发鱼缸WiFi设备	设备	● 未激活	-	<a href="#">激活凭证</a> <a href="#">查看</a>
<input type="checkbox"/>	BRmIjCoavwwWFXV9lTv	预发插座WiFi设备	设备	● 未激活	-	<a href="#">激活凭证</a> <a href="#">查看</a>
<input type="checkbox"/>	China_pre	预发网关产品	网关	● 离线	2019-07-02 16:58:26	<a href="#">激活凭证</a> <a href="#">查看子设备(1)</a>

- 设备列表：可见所有已量产的设备以及其状态、日志、详情等。
- 固件升级：对某个产品进行固件升级。
- 告警中心：当设备触发在控制台-人机交互-设备告警中配置的规则时，会展示在告警中心，可用于收集设备异常。
- 激活数据：激活设备的数据统计，如7日、14日、30日的激活曲线等。
- 活跃数据：活跃设备的数据统计，如7日、14日、30日的活跃曲线等。
- 地理分布：在服务中心开通地理分布功能后，可以在此处看到设备在全国各省、各城市的分布情况和排名。

· 用户运营

The screenshot shows the 'User Operation' (用户运营) dashboard. On the left is a navigation menu with options like '运营中心', '设备运维', '用户运营', '用户列表', '活跃用户', '新增用户', '用户反馈', and '数据大屏'. The main content area is titled '用户运营' and includes a '用户概览' (User Overview) section with three key metrics: '注册用户总数' (Total Registered Users) at 3, '设备用户 比例66.67%' (Device Users, 66.67% ratio) at 2, and '月活用户 比例33.33%' (Monthly Active Users, 33.33% ratio) at 1. Below this is a '用户列表' (User List) section with search filters and a table of users.

注册用户总数	设备用户 比例66.67%	月活用户 比例33.33%
3	2	1

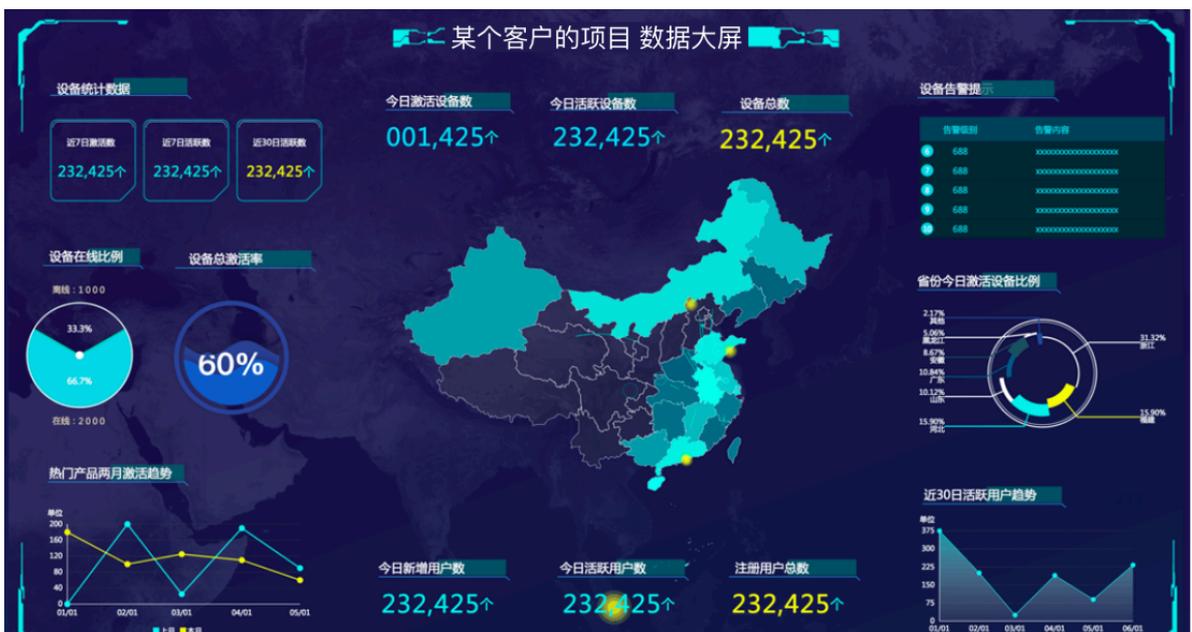
  

用户ID	用户名	手机号	创建时间	最后登录时间	操作
5074op73dfb68e320b808428d101d1feedba2061	13041000000	13041000000	2019-07-03 16:31:18	2019-07-03 16:31:19	<a href="#">查看详情</a>
508eop56cec27ef003ad4cdd0fe462d64e081965	18557000000	18557000000	2019-07-02 17:22:57	2019-07-02 17:22:57	<a href="#">查看详情</a>
509dop9c109439c051f957c77d47925c72f8018	18262000000	18262000000	2019-05-30 20:07:28	2019-07-03 16:29:01	<a href="#">查看详情</a>

- 用户列表：所有APP注册用户的列表，可以看到每个用户绑定的设备，也可以进行解绑操作。如果使用平台提供的内置账号体系，此处将展示所有用户信息；如果使用三方账号体系，此处展示的用户信息，依赖在账号对接时，三方云反馈的用户信息。
- 活跃用户：活跃用户的数据统计，如7日、14日、30日的激活曲线等。
- 新增用户：新增用户的数据统计，如7日、14日、30日的激活曲线等。
- 用户反馈：用户在公版APP中提交的意见反馈，会直接在此处呈现。如果品牌自有App使用了意见反馈插件，自有App中收集的意见反馈，也会在此处呈现。可以对反馈进行收藏和回复。

· 数据大屏

平台打通了阿里云的DataV功能，DataV是一款阿里云的可视化产品，是天猫双十一的实时数据大屏。此处呈现了所有的设备、用户、告警的综合数据，可用于客户演示Demo及数据监测。



## 可授权管理

运营中心可以单独授权给客户或运营者，无需用户开发即可交付一个完整的运营管理后台。运营中心和开发中心完全隔离，客户无法查看或修改开发中心的任何配置和价格，可以直接交付给客户。[如何授权项目？](#)

## 10.2 设置运营中心样式

平台支持可修改运营中心的标题、数据大屏的标题、企业图标，可将链接直接发给客户或运营者使用。

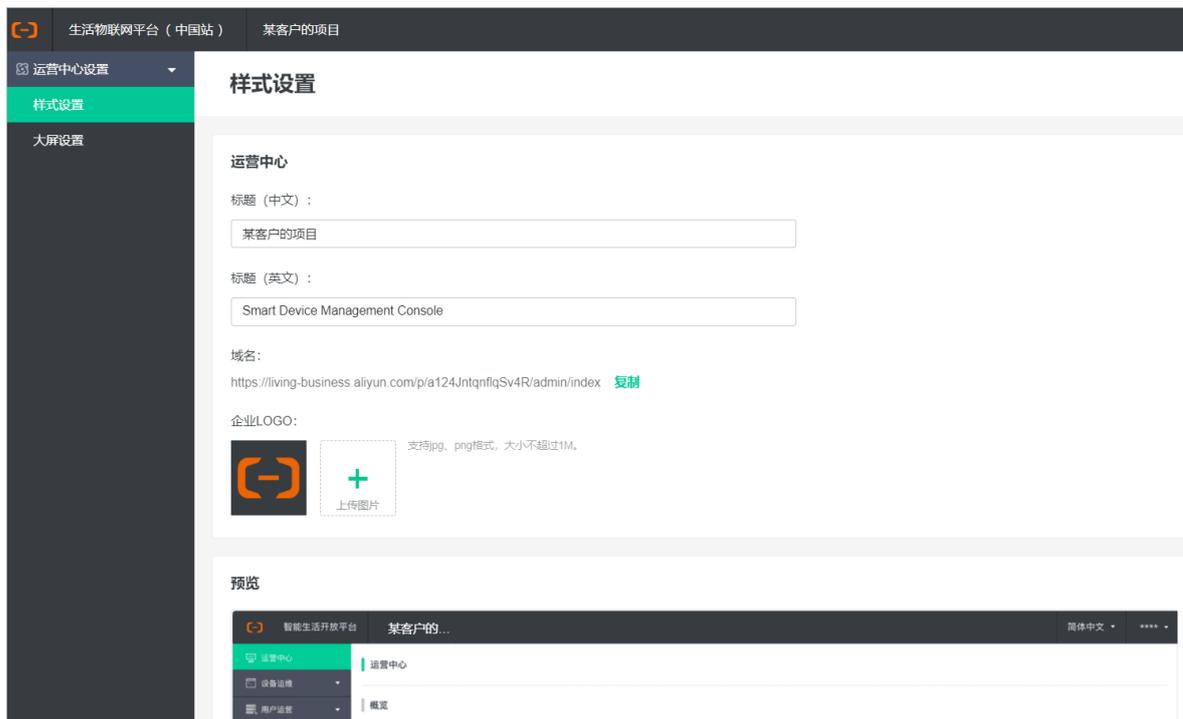
### 操作步骤

1. 进入项目主页面，将鼠标移至运营中心处，单击出现的设置。

The screenshot displays the '某客户的项目' (Project of a certain customer) management page. At the top, there are statistics: 产品总数 (6), 项目激活码 (0), 已激活设备 (0), and 当前在线设备 (0). A '项目设置' (Project Settings) icon is visible. Below the statistics is a '产品管理' (Product Management) section with a grid of product cards. Each card includes a product icon, name, status (e.g., '开发中'), tags (e.g., 'WiFi', '加湿器'), ProductKey, and creation time. On the right side, there is an 'App 管理' (App Management) section with '云智能 App' and '预发国内自有品牌APP'. Below that is the '运营中心' (Operational Center) section, which is highlighted with a red box around its '设置' (Settings) button. The '运营中心' description is '利用统计数据，提升商业价值和产品竞争力'. Below it is the '服务中心' (Service Center) section with the description '开启更多服务丰富产品开发能力和运营能力'. At the bottom, there is a '公告' (Announcement) section with '新手指导' (New User Guide) and a version update notice: '06.25发布，公测APP2.0发布'.

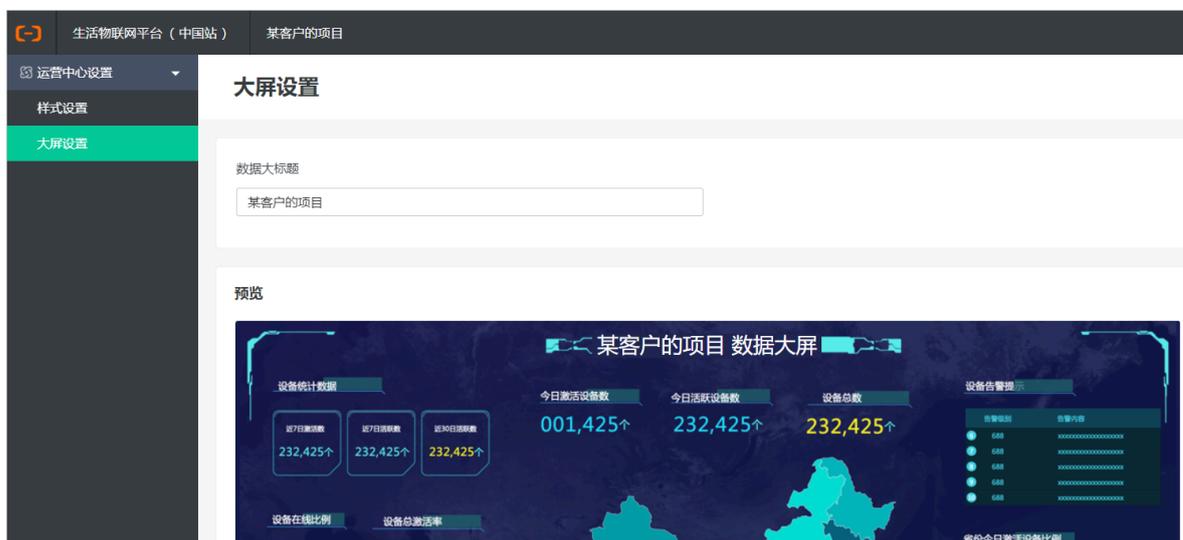
## 2. 选择样式设置，可以修改运营中心标题和公司图标。

修改完成后，在预览处可看到设置的效果。



## 3. 选择大屏设置，可以修改数据大屏的标题。

修改完成后，在预览处可看到设置的效果。



## 10.3 固件升级

固件升级支持对设备的固件进行远程空中升级（Over-The-Air），实现对设备的远程维护、功能升级、问题修复等场景的使用。您可以指定产品新增一个固件，对固件进行验证，验证通过后开始批量升级，并在固件详情中查看升级结果。

### 背景信息

固件在线升级功能是网络设备所必须支持的基础服务。在设备固件出现安全漏洞、软件BUG维护、自身功能优化、以及设备性能改良等需更新固件时，在线升级功能就尤为重要。

飞燕平台目前提供两种升级方式：

- 静默升级：设备连云后若发现有新的固件版本就自动进行升级，无需用户干预。
- App提醒升级：手机App会提示用户有新版本可用，并询问用户是否确认升级，只有在用户确认升级的情况下才会进行设备升级。

## 新增固件

1. 在项目主页面，单击运营中心。
2. 选择设备运维 > 固件升级

固件名称	固件版本号	所属产品	创建时间	状态	操作
test1	0.1.3.45	取暖器	2018-05-24 20:09:23	未验证	验证固件 批量升级 升级详情 删除
2	0.9.1	透传测试	2018-05-18 11:05:38	已验证	验证固件 批量升级 升级详情 删除
奥斯卡角度看	1.0.2.1	测试蓝牙	2018-05-10 20:19:21	未验证	验证固件 批量升级 升级详情 删除
1	0.9	测试2	2018-05-10 10:35:58	已验证	验证固件 批量升级 升级详情 删除
固件	1.0.0	测试2	2018-05-09 17:51:49	已验证	验证固件 批量升级 升级详情 删除

### 3. 单击新增固件。

您可以根据实际情况设置固件版本号，选择该固件的所属产品并上传固件文件（仅支持bin格式）。

#### 新增固件

\* 固件名称  ?

\* 固件版本号  ?

\* 所属产品  ▾

\* 选择固件

仅支持bin类型的文件，大小不超过64M

版本描述:  0/200

新增的固件显示在固件升级列表中，状态为未验证。

固件名称	固件版本号	所属产品	创建时间	状态	操作
test1	0.1.3.45	取暖器	2018-05-24 20:09:23	未验证	验证固件 批量升级 升级详情 删除
2	0.9.1	透传测试	2018-05-18 11:05:38	已验证	验证固件 批量升级 升级详情 删除
奥斯卡角度看	1.0.2.1	测试蓝牙	2018-05-10 20:19:21	未验证	验证固件 批量升级 升级详情 删除
1	0.9	测试2	2018-05-10 10:35:58	已验证	验证固件 批量升级 升级详情 删除
固件	1.0.0	测试2	2018-05-09 17:51:49	已验证	验证固件 批量升级 升级详情 删除

## 验证固件

在进行批量升级之前，您需要首先选取某个或者某些设备进行固件，确认固件的版本号、各项功能使用正常之后才可进行固件批量升级，避免因固件缺陷导致在线的多个设备升级后因故障而造成不能正常工作。

1. 选择要验证的固件，单击操作中的验证固件。

2. 配置验证固件的参数。

- 在待升级版本号中，选择被升级设备的版本号。

选择待升级版本号之前，需确保该产品下已经有设备上报了固件版本，否则将会选择不到任何版本号。升级的目标版本号支持多选。

- 在设备的deviceName中，指定需要被升级的设备名称。
- 在升级策略中，选择合适的升级策略。
- 单击添加按钮，将指定的设备添加进被升级的设备列表中。

验证固件 ✕

为了确保固件批量升级后设备能正常工作，请在批量升级前选择单台或者多台设备进行验证固件测试，防止将错误的固件升级到大量设备造成损失

\* 待升级版本号

\* 设备的deviceName:

\* 升级策略

3. 单击推送对设备进行升级。

- 进行验证固件操作后，固件的状态会从未验证变为已验证。该状态与设备的实际升级结果无关。
- 验证固件的操作可反复进行，目的是指定少量设备进行升级。

## 批量升级

固件验证后，请确认固件各项功能正常，验证通过即可开始进入固件批量升级流程，此固件可以在大批设备上投入使用。批量升级的本质也是对大批设备定向推送升级通知。

批量升级操作前的说明如下。

- 测试设备不能进行批量升级。
- 禁止使用未验证的固件进行批量升级操作。
- 设备从收到升级通知开始直至升级完成是一个渐进的过程，请耐心等待升级结果。
- 批量升级所覆盖的设备可能会因为设备上一次的升级动作没有结束（设备处于待升级或者升级中），而导致本次升级中该部分设备升级失败。
- 设备在实际升级过程中出现错误（如下载失败、校验失败、解压失败等），并且通知OTA系统后，系统会将本次升级动作置为完成（而实际升级操作结果为失败）。
- 可以在固件详情页，看到批量升级对应设备的升级情况，升级失败列表选项卡会显示简要的升级失败原因。



### 说明：

由于设备端存在的差异化，即使固件验证升级成功，也无法完全体现设备升级后各项功能是否正常可用，建议在设备固件升级后自行检查设备是否正常使用，避免批量升级后引起大批量的设备故障。

1. 选择要进行批量升级的固件，单击操作中的批量升级。
2. 配置批量升级的参数。

选择待升级的版本号，支持对多个版本进行选择。

批量升级 ×

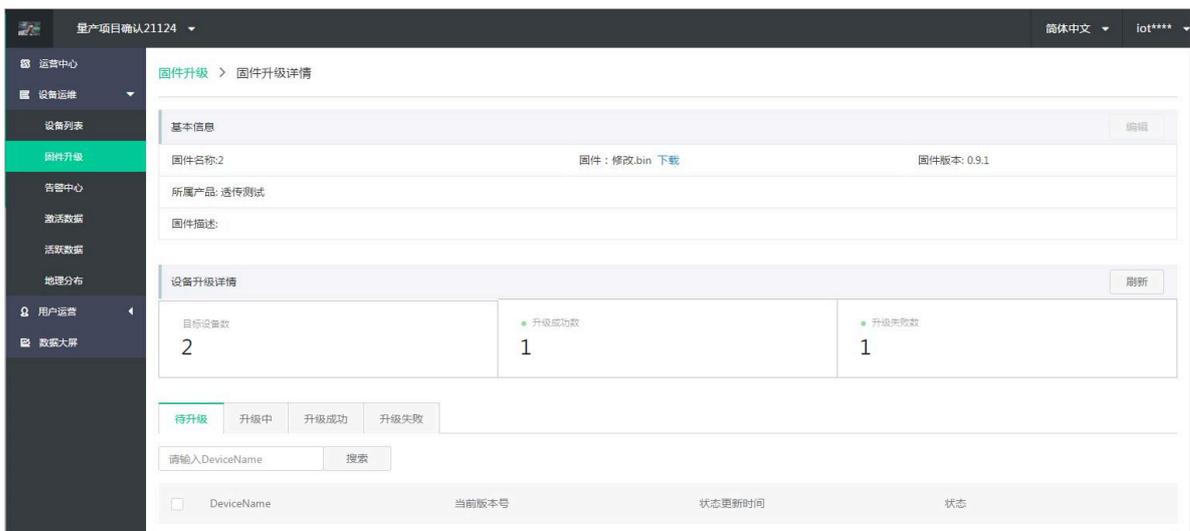
所属产品：透传测试(-)

\* 待升级版本号

\* 升级策略

3. 单击确定，启动批量升级操作。

4. 单击操作中的升级详情，可以查看升级的详细信息。



在该页面中可以查看升级各状态（待升级，升级中）的设备信息的设备数量，以及升级结果（升级成功，升级失败）的信息显示。