# 阿里云 物联网数据分析服务

最佳实践

文档版本: 20190828

为了无法计算的价值 | [-] 阿里云

# <u>法律声明</u>

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。

- 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。
- 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。	禁止: 重置操作将丢失用户配置数据。
A	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。	▲ 警告: 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。	道 说明: 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
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定。
courier 字体	命令。	执行 cd /d C:/windows 命令,进 入Windows系统文件夹。
##	表示参数、变量。	bae log listinstanceid Instance_ID
[]或者[a b ]	表示可选项,至多选择一个。	ipconfig [-all -t]
	表示必选项,至多选择一个。	<pre>swich {stand   slave}</pre>

# 目录

法律声明	I
通用约定	I
1 快速搭建室外停车场三维模型	1
2 快速接入设备位置到地图	7
3 在二维可视化页面嵌入第三方URL	
4 数据开发之分析决策直达设备	29

# 1 快速搭建室外停车场三维模型

本文描述如何使用物联网数据分析中的三维数据可视化功能,快速搭建室外停车场。

前提条件

在您的电脑设备上,下载并安装模型编辑器。

#### 操作步骤

1. 参考搭建工具手册,在模型编辑器中创建一个模型。

使用栅栏墙圈地,并找到交通模块中的马路、停车位,放置好合适的位置。



2. 为模型的停车位增加ObjectType=IOT的属性,方便被物联网平台识别为一个设备与地磁感应器关联。



#### 车位较多时,可以复制模型使用。

![](_page_5_Figure_5.jpeg)

 CampusBuilder(Momoda) v1.11.1608
 文件编辑选择 工具 视角 插件 帮助 新建 Ctrl+N ? i Q 打开 导入CAD Ctrl+O 6 2D Q 请输入搜索内容 1 **●** 生物 保存/另存为 Ctrl+S ۲ ₩ 交通设施(带电) 导出 • CampusBuilder场景包 Alt+F4 退出 い 一 生活 ThingJS场景包 GLTF全量场景包 uDCV完整场景文件 **♀** 交通 4 uDCV场景建筑文件 Ţ  $\bigcirc$ uDCV场景布局文件 **▲** 消彼 **#** max资源包 fbx资源包 3ds资源包 Î Î Ĩ 0° **□**. 机房 obj资源包 dae资源包 stl资源包 Þ . Ô+ ••• 其它 wrl资源包 0 ł 8 f 团撇 qITF资源包 ■ 设备 Ċ, **●** 个人 Ⅲ 交通设施(不带电) Ⅲ 交通工具 **杂** 功能 Ⅲ 禁令标志 警告标志

#### 3. 场景搭建完成之后保存并导出ThingJS场景包。

4. 参考创建产品,创建地磁感应器产品。

物联网平台	产品管理	新建产品 ×	
快速入门	我的产品(126)	产品值息	
设备管理 へ	产品列表	* 产品名称	1948F 04582/*54
/~~~ 设备	请输入产品名称查询 请选择产品标签	地磁感应離 ★所愿分类 ◎	
分组	产品存款 Pm	智能城市/公共服务/地磁检测器 > 功能定义	Ehmshi间 場例
規则引擎		百点迷惑	2019/04/04 14:17:32 查查 影除
边缘计算	-	* 节点逆型	2019/04/02 16:26:39
开发服务 シ		* 是否後入网关 ○ 是 ● 否	2019/04/02 16:09:12 直看 删除
◎ 祝頑服务 ◇	1000	○ - ○ - ○ - ○ - ○ - ○ - ○ - ○ - ○ - ○ -	2019/04/02 15:38:28 查看 翻除
<u>監控遺</u> 権 〜		* i 连网 / j っ c	2019/04/01 17:26:23 查看 最除
实例管理		WiFi ~ ~ · · · · · · · · · · · · · · · · ·	2019/04/01 14:29:52 查看 删除
7 10000		ICA 标准数据格式 (Alink JSON) V ●	2019/03/26 16:32:40 <b>25 8 B</b>
		* 他用 ID <sup>2</sup> 认证: ● ○ 是 ● ⑦ 否	2019/03/20 22:24:44 And Book 2019/03/20 21:04:35 Big Big
		更多信息	2019/03/20 10:15:20 ==== ====
		产品描述 请输入产品描述	3 4 13 下一页 > 1/13 到第 页 确定 每页显示: 10 >
		0100	
		使用文档 宠居 安居	

#### 5. 参考#unique\_5, 为刚刚创建的地磁感应器添加设备。

批量添加设备	×
* 产品: 地磁感应器(高级版) ~	
* 添加方式: 自动生成   批量上传	
* 设备数量:    10	
	确认取消

6. 参考#unique\_6, 创建一个分组, 并在将地磁感应器的所有设备添加到该分组中。

分组管理 > 分组详情				
停车场				
分组层级: <b>分组 / 停车场</b> 设备总数:10	分组ID:pvvklq3bTsmP2uG6WcyE010200 复制 激活设备:0		当前在线:0	
分组信息 设备列表 子分组列表				
设备列表				刷新 添加设备到分组
全部产品 ~ 请输入DeviceName查询	搜索			
DeviceName	设备所属产品	节点类型	状态/启用状态 • 最后上线时间	操作
unID9E1IMM8Kq0xWjd5r	地磁烈的企業	设备	• 未激活	查看 从分组中删除
swsu5B83ZX6RnyToIVdT	北达威风的江南	设备	● 未激活	查看 从分组中删除
jAY935NaQei7qKTOkDjY	北步 延迟 周晓 小江 周報	设备	• 未激活	查看 从分组中删除
RfemJzduSa3H06TznCox	地石灰虎彩山江音音	设备	• 未激活	查看 从分组中删除
SVLjaPtr4sBaiv5ZjiCt	地石砚间的江县晋	设备	• 未激活	查看 从分组中删除

7. 下载设备端SDK, 激活已创建的地磁感应器产品下所有设备。

### 8. 所有设备激活成功后,参考#unique\_7,创建三维数据可视化场景,关联步骤6中创建的设备 组,导入已制作完成的ThingJS场景包。

创建场景	$\times$
1 如何关联设备模型?点击查看文档	
* 场景名称:         停车场         * 目标产品或设备组:         广品       设备组         停车场         少日         * 授权形式:         • 只读取设备状态         • 副新频率:         15	
* 场景模型 停车场模型.tjs ×	

确定	取消

### 9. 在三维数据可视化场景中,将模型中的设备与已在物联网平台创建的设备进行关联。

空间数	据可视化 > 三维数据可	「视化			
停车	场				编辑 预览 分享
目标设计	备组: <b>停车场</b>		授权形式:只读取设备状态	刷新频率: 15 秒	设备总数:28
Ð	显 关联设备				⑦ 场景模型 🗋 事件告警 🔞 JS开发
	关联列表				
	∨ Campus				
	停车位	O	UJgfw2iz8M803Js >		
	停车位	Ø	rov2LPIpEk95pA0v >		
	停车位	Ø	选择关联设备 ^		
	停车位	O	> 站江		
	停车位	Ø	UJgfw2iz8M80:		
	停车位	Ø	<ul><li>7JsnsowgBi2m</li></ul>		<b>P</b> /
	停车位	Ø	选择关联设备 ~		
					7/

10.参考空间数据可视化分享场景,分享已配置好的三维数据可视化场景。通过分享的场景观察停车 场现状,智能化管理停车场业务。

# 2 快速接入设备位置到地图

本文描述如何使用物联网数据分析中的二维数据可视化功能,快速接入设备位置到地图。一般在空 气质量检测、基站分布、资产管理等场景,需要接入设备位置到地图上,实时展示设备的运行状 态,查看并管理设备。

步骤一、创建产品

- 1. 登录物联网平台控制台。
- 2. 参考#unique\_4, 创建带有位置属性的产品。

物联网平台	产品管理		新建产品	×			
快速入门	我的产品 (130)	_	产品信息				
设备管理  产品	产品列表		*产品名称 test_product				剧新创建产品
设备分组			* 所屬分类 ③ 自定义品类	~ 功能定义			
规则引擎	产品名称		サム光田		时间	操作	
数据分析	~		Fridee * 节点类型		3/04/19 10:17:33	查看 删除	
开发服务	~		<ul> <li>● 设备</li> <li>● 两关 ●</li> <li>* 是否接入网关</li> </ul>		9/04/10 15:40:31	查看 删除	
应用托管 视频服务	×	10 million (1990)	○ 是 ● 否 连网与数据		9/04/09 14:37:49	查看 删除	
监控运维	×	-	*连网方式		9/04/04 14:17:32	查看 删除	
产品文档			WFi * 款据格式	$\sim$	3/04/02 16:26:39	査者 删除 在を 删除	
	-		ICA 标准数据格式 (Alink JSON)	~ @	3/04/02 15:38:28	查看 删除	
	1000		· 1094 10 以起 · · · · · · · · · · · · · · · · · ·		3/04/01 17:26:23	查看 删除	
			更多信息		9/04/01 14:29:52	查看 删除	
			) 和別地社: 请输入产品描述		13 下一页 >	1/13 到第 页 确定	= 毎页显示: 10 ~

### 3. 参考#unique\_9, 添加设备。

添加设备 💿	×
i 特别说明:deviceName可以为空,当为空时,阿里云会颁放符作为deviceName。	发全局唯一标识
* 产品: test_product	
DeviceName :	
备注名称:	
请输入备注名称	Ø
	<b>确认</b> 取消

保存设备的设备证书(ProductKey、DeviceName、DeviceSecret),以备后续使用。

 参考二维数据可视化设备定位中的设备上报经纬度内容,为设备所属产品添加地理位置功能,并 上报设备位置到云端。

产品管理 → 产品详情 test_product ProductKey :	委加中运准功能			×	ET2	发布
产品信息 Topk 标准功能 ● 功能类型	逸择功能: 自定义品类 地理位置 地理位置 転用符:GeoLocation 近門映 転用符:GeoLocation 近門映 地理位置 属性 転用符:GeoLocation 近門映 地理位置 属性 転用符:GeoLocation 近門映	会務活加	已逃功能: 	①売請請除     ○     □     ○     □	9入物模型	操作
自定义功能 💿				确定取消		添加功能
功能类型 功能	能名称	标识符		数据类型	数据定义	操作

#### 步骤二、模拟设备

使用Link Kit SDK提供的Java SDK Demo模拟设备的GPS模块。

- 1. 下载Java SDK Demo并在本地解压。
- 2. 在device\_id.json文件中,填写已保存的设备证

书 (ProductKey、DeviceName、DeviceSecret) 信息。

Project 👻 😤 🛧 —	💿 MqttSample.java 🗴 💿 ThingSample.java 🛛 🚭 HelloWorld.java 🛛 💿 DeviceInfcData.java 🗶 💿 LabelSample.java 🗴 💼 🦓 device_id.json 🗡
▼ In JavaLinkKitDemo ~/Desktop/idea-workspace/pr ▶ In .idea ▼ In src ▼ Im main	1 f "productKey": w", deviceName": ", 4 productSecret": ", 5 "deviceSecret": ", 3",
<ul> <li>▼ isva</li> <li>▼ isva</li> <li>♥ aconaliyun.alink.devicesdk.demo</li> <li>♥ BaseSample</li> <li>♥ DeviceInfoData</li> <li>♥ DeviceInfoData</li> <li>♥ Fielutiis</li> <li>♥ GatewaySample</li> <li>♥ HeiloWorld</li> <li>■ LabelSample</li> <li>♥ Mutti Sample</li> <li>♥ SubThingSample</li> <li>♥ ThingData</li> <li>♥ ThingData</li> <li>♥ ThingData</li> </ul>	<pre>6</pre>
JavaLinkKitDemo.iml	

3. 在test\_case.json文件中,设置设备要上报的位置属性。

由于在物联网平台控制台,位置属性产品的产品详情 > 功能定义页面中设置的GeoLocation属 性定义是struct结构体,因此test\_case.json文件中的value参数需要填结构体子属性 的json串。可单击子属性名字后的编辑查看其标识符。

产品管理 > 产品详情	编辑标准功能	×			
test_product ProductKey: 复制 产品信息 Topic类列表 功能	<ul> <li>・功能送型: 属性</li> <li>・功能名称</li> <li>・功能名称</li> </ul>	] 0	设备数:0 前往	管理	发布
标准功能 💿	+ 标识符: GeoLocation + 教根送型:	-		导入物模型 查	看物模型 添加功能
功能类型 功能名称	struct (结构体)		牧据类型	数据定义	操作
属性 地理位置 司选	* JSON对象:	7	truct (结构体)		信号 删除
	参数名称:经度 编辑 删除		n det (spinglek)		3001444 A001804
	参数名称:纬度 编辑 删除				
白宁义功能 🔿	参数名称:海拔 编辑 删除				Stimtak
口足入功能	参数名称:坐标系统 编辑 副除				28CMU46
功能类型 功能名称	+ 新増参数		牧据类型	数据定义	操作
	读写类型: 只读 描述 ·				
	max 薄输入描述				
		0/100			

test\_case.json文件中,填好结构体子属性的json串后,文件内容如下所示:

ThingSample.java $ imes$	$\bigcirc$ HelloWorld.java $ imes$	🍖 device_id.json ×	🎒 test_case.json 🛛		
"identifi "value":"	er":"GeoLocation", {\"Longitude\":116.400	)372,\"Latitude\"::	39.919316,\"Altitude	<pre>%\":0,\"CoordinateSystem\":1}"</pre>	
▲ }					

### - 说明:

可使用高德开放平台经纬度拾取工具,设置设备经纬度(Longitude、Latitude)。

4. Demo程序的入口在HelloWorld. java文件中,本Demo中我们只需要设备上报位置的属

性,因此需简化HelloWorld.java代码。简化后的代码如下所示:

```
public class HelloWorld {
    private static final String TAG = "HelloWorld";
    private String pk, dn;
    private ThingSample thingTestManager = null;

    public static void main(String[] args) {
        ALog.d(TAG, "Hello world!");
        ALog.setLevel(ALog.LEVEL_DEBUG);
        HelloWorld manager = new HelloWorld();
        ALog.d(TAG, "args=" + Arrays.toString(args));
        System.out.println(System.getProperty("user.dir"));
    }
}
```

```
String diPath = System.getProperty("user.dir") + "/device_id
.json";
        String deviceInfo = FileUtils.readFile(diPath);
        if (deviceInfo == null) {
    ALog.e(TAG, "main - need device info path.");
           return;
        }
        Gson mGson = new Gson();
        DeviceInfoData deviceInfoData = mGson.fromJson(deviceInfo,
DeviceInfoData.class);
       if (deviceInfoData == null) {
           ALog.e(TAG, "main - deviceInfo format error.");
           return;
       ALog.d(TAG, "测试一机一密和物模型");
       manager.init(deviceInfoData);
    }
    public void init(final DeviceInfoData deviceInfoData) {
        this.pk = deviceInfoData.productKey;
        this.dn = deviceInfoData.deviceName;
        LinkKitInitParams params = new LinkKitInitParams();
        /**
        * 设置 Mqtt 初始化参数
        */
       IoTMqttClientConfig config = new IoTMqttClientConfig();
        config.productKey = deviceInfoData.productKey;
        config.deviceName = deviceInfoData.deviceName;
       config.deviceSecret = deviceInfoData.deviceSecret;
        config.channelHost = pk + ".iot-as-mqtt." + deviceInfoData.
region + ".aliyuncs.com:1883";
        /**
        * 是否接受离线消息
        * 对应 mqtt 的 cleanSession 字段
        */
        config.receiveOfflineMsg = false;
        params.mqttClientConfig = config;
        /**
        * 设置初始化三元组信息,用户传入
        */
        DeviceInfo deviceInfo = new DeviceInfo();
        deviceInfo.productKey = pk;
       deviceInfo.deviceName = dn;
        deviceInfo.deviceSecret = deviceInfoData.deviceSecret;
        params.deviceInfo = deviceInfo;
        /**
        * 设置设备当前的初始状态值,属性需要和云端创建的物模型属性一致
        * 如果这里什么属性都不填,物模型就没有当前设备相关属性的初始值。
         * 用户调用物模型上报接口之后,物模型会有相关数据缓存。
        */
        Map<String, ValueWrapper> propertyValues = new HashMap<
       ValueWrapper>();
String,
        // 示例
        // propertyValues.put("LightSwitch", new ValueWrapper.
BooleanValueWrapper(0));
        params.propertyValues = propertyValues;
        thingTestManager = new ThingSample(pk, dn);
        LinkKit.getInstance().init(params, new ILinkKitConnectListe
ner() {
```

5. 执行HelloWorld. java文件, 使设备的位置生效。

#### 步骤三、创建二维数据可视化场景

1. 参考#unique\_10, 创建一个场景。

物联网平台	空间数据可视化		
快速入门	二维数据可视化 三维数据可视化	创建场票	×
设备管理 🗸 🗸	免费公测 仅支持高级版产品使用此功能,建议		
规则引擎		如何配置设备位置信息?点击查看文档	
数据分析 へ			
数据管理		* 功豪凸标: test场震	
数据开发	金融运用	* 目标产品或设备组:	
流数据分析	CONFE AND DEL	产品 设备组	
空间数据可视化		test_product ~	
边缘计算 🛛 🗸 🗸		* 授权形式:	
开发服务		● 只读取设备状态 ○读取并修改设备状态	
应用托管 🗸 🗸		* 刷新频率:	
视频服务		15	
监控运维 🗸 🗸			
产品文档		确定目	取消

其中,目标产品或设备组选择产品,并从下拉框中选择已创建的位置属性产品。

![](_page_15_Figure_8.jpeg)

2. 进入二维可视化场景页面查看设备实时位置。

至此,您已经将您的物联网设备快速的接入到地图中了。

#### 下一步

可参考#unique\_11,将已创建二维数据可视化场景分享给其他人。

# 3 在二维可视化页面嵌入第三方URL

本文描述如何在物联网数据分析中的二维数据可视化场景中,嵌入第三方连接。例如,二维场 景中有监控摄像头等设备,且设备本身具有自定义气泡弹框需求等场景,可以在二维场景中嵌 入URL,在查看设备的运行状态、管理设备的同时查看监控画面。

实现此功能,需要二维数据可视化场景与IoT Studio中的Web可视化配合使用,详细步骤请见下 文。

#### 步骤一、创建产品并添加设备

1. 参考#unique\_13, 创建一个产品。本文以test\_product为例。

* 产品名称	
test_product	
* 所属分类 🕜	
自定义品类	→ 功能定义
市点类型	
*节点类型	
💿 设备 🛛 🔾 🖉	
* 是否接入网关	
* 是否接入网关 〇 是	
<ul> <li>* 是否接入网关</li> <li>○ 是</li> <li>● 否</li> <li>至</li> </ul>	
<ul> <li>* 是否接入网关</li> <li>一 是</li> <li>● 否</li> <li>年网与数据</li> <li>* 连网方式</li> </ul>	
* 是否接入网关 ② 是 ③ 否 至网与数据 * 连网方式 WiFi	~
* 是否接入网关 ② 是 ③ 否 至网与数据 * 连网方式 WiFi * 数据格式	
* 是否接入网关 ② 是 ③ 否 生网与数据 * 连网方式 WiFi * 数据格式 ICA 标准数据格式 (Alink JSON)	V
* 是否接入网关 ② 是 ③ 否 至网与数据 * 连网方式 WiFi * 数据格式 ICA 标准数据格式 (Alink JSON) * 使用 ID <sup>2</sup> 认证 ② ③ 是 ④ 否	V
* 是否接入网关 ② 是 ③ 否 至网与数据 * 连网方式 WiFi * 数据格式 ICA 标准数据格式 (Alink JSON) * 使用 ID <sup>2</sup> 认证 ③ ③ 是 ④ 否	
* 是否接入网关 ② 是 ③ 否 在 至 至 至 至 至 至 至 二 二 二 二 二 二 二 二 二 二 二 二 二	

- 产品管理 > 产品详情 添加自定义功能 test\_product ProductKey : 200 设备数:2 前往管理 \* 功能举型 产品信息 Topic类列表 功能定义 服务端记 原性服务 事件 💿 \* 功能名称: 标准功能 🌑 三方连接 0 导入物模型 查看物模型 \* 标识符: 功能类型 功能名称 数据定义 操作 WEB URL 0 \* 数据类型: text (字符串 \* 数据长度 字节 2048 读写类型: ● 读写 ○ 只读 描述: 自定义功能 💿 功能类型 功能名称 数据定义 操作 确认 取消
- 2. 参考#unique\_14, 在产品详情页面,为test\_product产品添加自定义功能。

3. 参考#unique\_15/unique\_15\_Connect\_42\_section\_u23\_ssb\_wdb,为test\_product产 品添加标签。

est_product						2
roductKey :	复制	ProductSecret:******* 显示		设备数:2 前往管理		
产品信息 Topi	c类列表 功能定义 服	务端订阅 日志服务 在线调试				
品信息						\$
产品名称	test_product	添加标签	×	创建时间	2019/04/25 19:41:38	
所属分类	自定义品类	产品标签:				
动态注册 🍥		studioType	删除			
状态	• 开发中	+新增标签		连网协议	WiFi	
产品描述			変換され、目交活動			

其中,标签key设置为studioType,标签value有如下两种设置方法,不同的value,会影响 第三方弹窗的弹出方式:

- ・标签value设置为url:单击设备气泡弹出气泡窗口,再通过气泡窗口中的URL链接弹出第 三方弹窗。
- ・标签value设置为pop:直接单击设备气泡弹出第三方窗口。

两种弹出方式的区别请见步骤三中的第8小步。

### 4. 参考#unique\_17,为产品添加设备。

添加设备 💿	×
特别说明: deviceName可以为空,当为空时,阿里云会颁发全局唯一标识符 作为deviceName。	
* 产品: test_product v DeviceName:	
test02	
备注名称:	
请输入备注名称 2	
确认 耶	消

#### 步骤二、创建二维数据可视化场景

1. 参考#unique\_18, 为设备设置地理位置。

设备管理 > 设备详情					
test02 在线					
产品:test_product 查看		ProductKey :	复制	DeviceSecret: ********* 显示	
设备信息 Topic列表	表 运行状态 事件管理 服务调用 设备	副子 文件管理 日	志服务		
设备信息					
产品名称	test_product	ProductKey	<b>王 - 王 - 王</b> - 夏制	区域	-
节点类型	设备	DeviceName	test02 复制	DeviceSecret	******** 显示
备注名称 🔘	编辑	IP地址	47.101.109.73	固件版本	-
添加时间	2019/04/29 15:26:46	激活时间	2019/05/08 19:35:21	最后上线时间	2019/05/08 19:35:21
当前状态	在线	实时延迟 🔘	测试		
设备扩展信息					
KHA KHO					
SDK 语言	-	版本号	-	模组商	-
模组信息	-				
标繁信白					APPetra
1013年1日404 设备标签: coordinate: 120.08	0104:30.132311				369121

2. 参考#unique\_19, 使用test\_product产品创建一个二维数据可视化场景。

创建场景	$\times$
1 如何配置设备位置信息?点击查看文档	
* 场景名称: LD测试 * 目标产品或设备组: <u>产品</u> 设备组 test_product * 授权形式: ④ 只读取设备状态 M新频率: 15 秒 v	
确定取	消

### 3. 设置设备的WEB\_URL属性的值为设备需要链接到的第三方页面地址。可以使用在线调试功

能,上报属性值。

物联网平台	在线调试
概览	请选择设备: test_product v test02 v
设备管理	
规则引擎	调试设备
数据分析	调试真实设备虚拟真实设备
边缘计算	属性上报事件上报
开发服务	WEB_URL
应用托管	https://j .com/embed/XNDE2NTI4
视频服务	
监控运维	
实时监控	
在线调试	
日志服务	
固件升级	
远程配置	
实例管理	
产品文档	
	推送 策略推送 关闭虚拟设备 查看数据

添加第三方链接后,单击二维数据场景中设备气泡后,如下所示:

空间数据可视化 > 二维数据可视化				
LD测试			批量导入位置 《编辑 预	览 <b>分享</b>
目标产品:test_product	授权形式:只读取设备状态	刷新频率:15秒	设备总数:2	
← 全部設备 ✓ ▼ 24/#84/5/第		の州市西湖 教育基金会 画 満江西湖 画 満等研究院	巴士落驾点	<ul> <li>热力图</li> </ul>
	四路	test02 (在线)	杭州中达电机有限公司	。 限公 。
	。 杭州崇胜机械 制造有限公司	居住 事件 三方连接 https:// com/e ○ 弊種配置 ● 医标面置		() () () () () () () () () () () () () (
⊖ 雜處漆	۹	阿里云 铁石区 Costa Coffee	基礎或估计》) 交換大 生物科技有限公司 网络生	会出租车 。 正 非落答点3 。 正
	石龙山村	0		

#### 步骤三、IoT Studio可视化大屏搭建

IoT Studio可视化相关详细内容,请参考Web可视化开发。

1. 拖动二维数据可视化场景(地图组件)到IoT Studio大屏中,占满整个屏幕。

页面 🛙 🗑	- 1	0 100	200	300 40	0 500	600	700	800	900	1000	1100	1200	1300	1400
圖 自定义新增页1	•													
	8													
				选择地图场景							×			
可视化组件														
基础 控制 图表 表单	20					and a	T BALER ACTER	MEANER BER	2.879	ates ales	BEA987 MARABOR			
基础					_		REALING STAR	A. STRAR ST STRAR		No an	NUMAR STATE			
丁 □ ■	300			<i>#</i> ""	"维数据可视化"中	10	10	nit in New .		- 7	ART REAL			
					创建场景	LD测试	t	· D	4		. 15			
椭圆形 三角形 星形	400		р			© 2019	9-05-08 16:16:37							
● ▼ ● 多边形 前形 指示灯				Lev V		5120 <sup>20</sup>								
E = 0	0			1										
卡片 表格 时钟	8				AVENUE DECEM	1								
时 地图				4+602	And the	6								
控制	600													
<b>CD</b> ( <b>0</b> )				-										
并天 渭宗 投钮	700			刷新列表						确定	取消			
图表														
· 비 · 배 · · · · · · · · · · · · · · · ·														
<b>€</b> <u>∽</u> (3)	80													
饼图 曲线图 仪表盘														
表单	8 视图	☑ 标尺 □ 网格								适合画布 🖸	缩放			☑ 自适应

2. 添加一个iframe组件到大屏中,用于在点击设备气泡时,弹出一个第三方窗口。

同时为了能够在不需要的时候关闭第三方窗口,需要添加一个关闭按钮到iframe的右上角。

![](_page_25_Figure_4.jpeg)

#### 3. 配置变量的联动关系。

为了点击设备气泡的时候,能够打开第三方地址,需要设置地图和iframe组件的交互关系,让 iframe组件中设置的地址能够从设备传入iframe。

a. 单击地图,新增一个交互,并设置动作为赋值给变量。

![](_page_26_Figure_5.jpeg)

b. 新增一个变量。

![](_page_26_Figure_7.jpeg)

#### 设置变量的名称,默认值设置为第三方URL。

新增变量		×
	* 变量名称:	
	第三方链接 📀	
	默认值:	
	https://com/embed/.	
	描述:	
	请输入变量说明,限100字	
	确认	取消

c. 配置变量。

赋值给变量			×	1	样式	交互	
				▼ 交互1			Ī
				事件		点击设备属性	
	请选择该组件的变量值			动作		赋值给变量	
	url(链接)	值给某个变量			型墨	答理亦是	
	自定义 (固定值)	記旧似文重			印度		
	+ 赋值	管理变量			<b>→</b> 辛	新博森石	
					тљу	加相义ユ	

赋值给变量			×	样式	交互	
				▼ 交互1		Ī
变量名	变量值	说明		事件	点击设备属性	
第三方链接 🔺	url	链接	Ŵ	动作	赋值给变量	,
◆第三方 ◆	管理变量	]		配置	管理变量	
				+	·新增交互	

4. 单击iframe组件, 配置关联链接。

_	0 100 200	300 400 500	600 700 800	900 1000	1100 1200	0 1300 1400		样式
0	□ <del>単</del> 全部段格 ~	凌蒙获			· ② 事件告警 》新留科的团	历史轨迹 ⑧ 热力图	25 X	150 Y 0 角度
0 100			大切 石皮山市 の新語	。 洲市西湖 南富企会 四北檔書点 四球 研究院		世首年纪24 按有别公司 以 政	组件名称	iframe-3pi5
30	区菜 1月21	配置链接			×	ULET) REC •	组件可见性	
300			] 目前只支持HTTPS协计 が	1果你捕写了HTTP地址		抗州通域急新医 疗科技有限公司	不透明度	100%
	请配置	"关 需要绑定自己的	1, 古前人名特尔 10 0.0 年 , 4	and and a second s	靠德威抗州生物和 有限公司(东北门)	料技 非正方聲电器有 課公司(問問门)	关联链接	配置
400				绑定变	a la	为的车落客点3 · I	✔ ▼ 边框	
20		-0					粗细	1
				确认 取	消	/ ;	颜色	#E8EBEEFF -
900			, e e e e e e e e e e e e e e e e e e e	。 吃大会酒		• 杭州威镇: 业有限公:	样式	· · · ·
200	P.	石龙村桥 😁		注接效点		INTE NOTAR		

在弹出框中单击绑定变量,选择已配置好的第三方链接变量。

ĵ	选择变量			×
	变量名	描述	默认值	操作
	第三方链接	-	https://	赋值来源 编辑
				确认取消

5. 因为只有点击设备气泡时,才显示第三方窗口,并且第三方窗口显示后,才可看到关闭按钮,因此需要默认隐藏组件。关闭iframe组件和按钮组件的可见性。

![](_page_29_Figure_3.jpeg)

6. 单击地图,配置隐藏组件与地图的交互关系,实现单击设备气泡时,弹出三方页面和关闭按钮的功能。

![](_page_29_Figure_5.jpeg)

7. 为按钮组件添加交互,实现单击关闭按钮后关闭iframe。

![](_page_29_Figure_7.jpeg)

8. 在界面右上方单击保存,然后单击预览,查看已配置完成的Web可视化大屏。

根据产品标签value的不同设置, 第三方窗口的弹出方式不同:

・标签value为url:单击设备气泡,在弹出窗口中单击第三方连接,则显示第三方弹窗,单 击关闭按钮,可关闭弹窗。

![](_page_30_Picture_5.jpeg)

·标签value为pop:单击设备气泡,可直接弹出第三方窗口。

![](_page_30_Figure_7.jpeg)

9. 切换到IoT Studio Web可视化开发编辑页面,在页面右上方单击发布,填写版本内容,单击确

![](_page_31_Figure_3.jpeg)

系统显示发布成功后,后续操作您可以参考Web可视化开发内容处理您的Web可视化应用。

# 4数据开发之分析决策直达设备

物联网数据分析服务提供的数据开发功能,不仅支持分析设备上报云端的数据,还可以通过SQL将 分析后的决策结果下发至设备进行远程控制。

前提条件

· 仅支持使用真实设备或使用SDK模拟设备实现下发决策至设备的功能。

· 下发数据的设备属性,必须是读写类型。

背景信息

本文以使用Java SDK模拟空调设备为例,利用物联网数据分析服务提供的数据开发功能编写SQL

,从某一个提供数据的产品获取设备数据,并将该数据下发到空调设备。

操作步骤

1. 登录物联网平台控制台。

#### 2. 创建一个空调设备。

a) 参考#unique\_21, 创建如下图所示的空调产品。

新建产品	×
产品信息	
* 产品名称	
空调	
* 所属分类 💿	
智能生活 / 大家电 / 空调机 🛛 🗸 🗸 功	能定义
节点类型	
* 节点类型	
● 设备 ○ 网关 @	
* 是否接入网关	
连网与数据	
* 连网方式	
WiFi ~	
* 数据格式	
ICA 标准数据格式 (Alink JSON) V @	)
* 使用 ID <sup>2</sup> 认证 ②	
○ 是 ● 否	
更多信息	~
<b>使田文</b> 档	当時に

产品创建成功后,到产品详情页面获取ProductSecret,以备后续使用。

b) 参考#unique\_22, 添加空调设备。

添加设备 💿	×
特别说明: deviceName可以为空,当为空时,阿里云会颁发 作为deviceName。	这全局唯一标识符
* 产品: 空调 ✓	
airconditioner_1 备注名称:	0
请输入备注名称	0
	<b>确认</b> 取消

设备创建成功后,保存设备证书(ProductKey、DeviceName、DeviceSecret),以备 后续使用。

#### 3. 使用Link Kit SDK提供的Java SDK Demo模拟空调设备,使其上线。

- a) 下载Java SDK Demo并解压到本地。
- b) 解压后在device\_id.json文件中,填写已保存的设备信
  - 息(ProductKey、DeviceName、DeviceSecret)和产品ProductSecret。

![](_page_36_Picture_6.jpeg)

#### c) 启动HelloWorld.java文件中的main方法。

	/ La devices	ak / 🔤 demo / 🈒 Helloworld /
Project ▼	C HelloWorld.java	C MqttSample.java ×  do device id.json ×
JavaLinkKitDemo D:\learn\JavaLinkKitDemo	1 pac	kage com alivun alink devicesdk demo:
> 🖿 .idea	2	
V src	2	
V main	51шр	ort
<ul> <li>Java</li> <li>Com align aligh devices dk demo</li> </ul>	24	
C BaseSample	25 🕨 pub	lic class HelloWorld {
COTASample	26	<pre>private static final String TAG = "HelloWorld";</pre>
C DeviceInfoData	27	
C DeviceShadowSample	28	private String pk, dn;
© FileUtils	29	private ThingSample thingTestManager = null:
C GatewaySample	30	private mingetapre eningresentandor marr,
C LabelSample	31	
C MattSample		
C SubThingSample	32	public static void main(String[] args) {
C ThingData		ALog. d(TAG, msg: "Hello world!");
C ThingSample	34	ALog. setLevel(ALog. LEVEL_DEBUG);
> target	35	HelloWorld manager = new HelloWorld();
in device_id.json	36	ALog. d(TAG, msg: "args=" + Arrays. toString(args));
m pom.xml	37	System. <i>out</i> . println(System. <i>getProperty</i> ("user. dir"));
test_case.json	38	String diPath = System getProperty("user dir") + "/device id ison"
test_sub_case.json	30	String deviceInfo = Fileltils readFile(diPath):
╡ 使用说明.txt	40	if (luniorInfo == == 11) {
III External Libraries	40	
Scratches and Consoles	41	ALog. e(TAG, msg: main - need device info path. );
	42	return;
	43	}
	44	Gson mGson = new Gson();
	45	DeviceInfoData deviceInfoData = mGson.fromJson(deviceInfo, DeviceInfoData.class);
	46	deviceInfoData.region = "cn-shanghai":
	47	if (deviceInfoDate == null) {
	40	$M_{\text{eff}} = \left( \frac{TAC}{T} \right) = \frac{1}{T} \left( \frac{TAC}{T} \right)$
	40	ALOg. e(IAG, Insg. main - devicennio format error. );
	49	return;
		}
	51	<pre>if (StringUtils. isEmptyString(deviceInfoData. deviceSecret)) {</pre>
	52	manager.deviceRegister(deviceInfoData);
	53	ALog. d(TAG, msg: "测试一型一密动态注册,只测试动态注册");
	54	return;
	55	3

返回物联网平台控制台,在设备管理页面选择空调产品后,可看到产品下的设备状态为在

线。

设备管理					
空调	设备总数 💿 1	• 激活设备 🔘 1	• 当前在线 💮 1		刷新
<b>设备列表</b> 批次管理					
设备列表					批量添加 添加设备
DeviceName ~ 请输入Dev	viceName 请选	≩设备标签 ∨	搜索		
DeviceName/备注名	称    设备所属产品	节点类型	状态/启用状态 🗸	最后上线时间	操作
airconditioner_1	空调	设备	• 在线 <b>(</b> )	2019/07/03 17:49:27	查看 删除

- 4. 创建商场温度传感器设备,并上报数据到云端。
  - a) 参考#unique\_21, 创建一个商场温度传感器产品。

新建产品			×
产品信息			
* 产品名	称		
商场温	度传感器		
* 征屋八	* •		
* 川周刀:			古台中で
自形四	区/至内画度後感情	~	MIRKEX
节点类型			
* 节点类	፹		
• 设备	🔾 网关 🕘		
* 是否接	入网关		
〇是	• 否		
连网与数据			
* 连网方:	<del>ت</del> ل		
WiFi		~	
* 数据格	<del>بار</del>		
ICA标	~ 准数据格式 (Alink JSON)	) ~	0
* 使用 IC	<sup>2</sup> 认证 <b>②</b>		
	• 4		
更多信息			~
使用文档			取消 完成
		文档版本:	20190828

- b) 参考#unique\_23, 批量添加5个商场温度传感器设备。
- c) 参考#unique\_24, 为商场温度传感器添加自定义功能。

![](_page_40_Picture_4.jpeg)

d) 在物联网平台,选择监控运维 > 在线调试,在在线调试页面,选择调试产品和设备后上报属 性数据。

物联网平台		在线调试		
概況		清选择设备:		
设备管理	^		1	
产品		调试设备	实时日志 • 检测到设备在	送 目动朝新 <b>一 </b> 刷新 <b>汤新</b>
设备		调试真实设备 虚拟真实设备		
分组			美型/时间	内容
规则引制		CurrentTemperature	设备上报数据	2019-07-21 17:24:34.514, 3ABD B4A15, upstream - bizType=PROPERTY_REPORT, params=["id"."123","fot
数据分析	^	28	2019-07-21 17:24:34	Idf":"WX gJ000100","method":"thing.event.property.post","params":["CurrentTemperature":28,"AreaIdf":"一提"),"topic:":/sy s/a1 wi/vX BgJ/thing/event/property/post","uniMsgIdf":"4611487928280940544","version":"1.0"),result=code:20
数据管理		CurrentHumidity		0,message:success,topic=/sys/a1 witvX BgJ/thing/event/property/post,response=("code":200,"data":{},"d":"12 3","message":"success","method":"thing.event.property.post","version":"1.0"),device=("aliyunCommodityCode":"tothub_senior","deviceKe
数据开发		请输入参数(double)		y"."\vX gJ","deviceSecret"."***","gmtCreate".1563285215000,"gmtModified".1563285215000,"id".45837239,"iotid"."\vXV gJ000100","name"."\vX gJ","productKey"."a1 w"."rbacTenantId"."3ABD
流数据分析		Areald		B4A15", "region" "Cn-shanghai", "status".0, "statusLast".0, "thingType" "DEVICE"), scriptData=(), traceId=0bc06 4 3187d574b
空间数据可视化		-łġ 💿		
边缘计算	$\sim$			
开发服务	~			
视频服务	~			
监控运维	^			
实时监控				
运维大屏				
在线调试				
日志服务				
固件升级				
远程配置		推送 策略推送 关闭虚拟设备 查看数据		

商场温度传感器产品下的5个设备,分别上报数据,可上报相同的属性值。

- 5. 在数据分析 > 数据开发页面,查看空调设备和商场温度传感器设备的数据表。
  - a) 在数据开发页面,选择表管理 > 平台设备数据表并刷新。

刷新后,可在设备数据表中看到新建的空调产品。

数据开发		
▶ 数据开发	+	*
▶ 运行历史	С	
▶ API服务	+	
▼ 表管理		
▶ 囲 平台系统表		
▼ 圖 平台设备数据表	С	
▼ 🛇 设备数据表		
10/17/11/2		
空调		
100000		

b) 找到商场温度传感器,保存数据概览页面中\$product\_key的值,并查看设备属性标识符,以备后续SQL语句中使用。

数据开发								
▼ 目 平台设备数据表 C  ▼ 응 设备数据表	表结构 数据概定 概定 新以查询最示母新的20生数组							i
호행	\$device_name	CurrentHumidity	Sevent_time	CurrentTemperature	Areald	Sproduct_key	\$iot_id	Stype
商场温度传感器	IvXVnjgH88rMZU8b5BgJ		1563714092629	28.6	一楼	2010/02/15	and the second second	DEVICE
100,000	I9EPkPoM60I6no8sUO3r		1563714087764	28.6	—楼			DEVICE
▶ ◎ 设备快照表	epA5U7bRlgdHlLjEEl4N		1563714083383	28.6		10000		DEVICE
▶ © 设备事件表 ▶ Q 用户数据	aqvGQxHWOJw2kaJFW 0fN		1563714079017	28.6	一楼	10.000		DEVICE
▼ 函数列表	aW3L05S8XtZeeLYOYP St		1563714076296	28.6	一接	10000	and the second s	DEVICE

- 6. 创建数据开发任务。
  - a) 参考#unique\_25, 创建一个下发数据至设备的任务。

数据开发					
▼ 数据开发 +	下发数据至设备				
▼ 団 开发任务	〇 保存 ④ 运行	× 取消 🛛 🔂 定时运行	园 语法校验	① 生成API	⑤最近保存时间: 2019-07-11 16:57:06
下发数据至设备 上 □ 模板区SOL文件来	1 SELECT NULL				
▶ 运行历史 C	运行日志 结	果			♀ 下载CSV
▶ API服务 +					

b) 在任务中编写如下SQL。

```
-- 以下示例中的函数均可在函数列表中找到
-- 查询商场温度传感器上报温度且上报时间在五分钟内,且高温数量传感器数量大于5
CREATE VIEW HIGH_TEMPERATURE AS
   SELECT count(CurrentTemperature) AS high_temp_num,
 AreaId ,AVG(CurrentTemperature) AS current_temper
   FROM ${pk.Your_ProductKey1}
   WHERE CurrentTemperature IS NOT NULL AND
 DATE_DIFF('MINUTE', FROM_UNIXTIME($event_time/1000), NOW()) < 5 AND
 CurrentTemperature > 26
       GROUP BY AreaId HAVING count(CurrentTemperature) > 5;
INSERT INTO ${pk.Your_ProductKey2.Your_DeviceName2}
SELECT
1 AS PowerSwitch,
-- 根据当前温度调整风速等级
CASE current_temper
   WHEN 35 THEN 5
        32 THEN 4
   WHEN
        30 THEN 3
   WHEN
   WHEN
        28 THEN 0
   ELSE 0 END AS WindSpeed,
  根据月份切换风力类型 自动 - 0; 制冷 - 1; 制热 - 2; 通风 - 3; 除湿 - 4;
CASE
```

```
WHEN MONTH(NOW()) > 6 AND MONTH(NOW()) < 11 THEN 1
WHEN MONTH(NOW()) > 11 AND MONTH(NOW()) < 4 THEN 2
ELSE 0 END AS WorkMode
FROM HIGH_TEMPERATURE WHERE AreaId='→楼';
```

其中:

- · FROM语句中, Your\_ProductKey1替换为商场温度传感器的ProductKey。
- INSERT INTO语句中, Your\_ProductKey2和Your\_DeviceName2替換为空调设备
   的ProductKey和DeviceName。

0

📃 说明:

语法解释如下:

```
INSERT INTO {目标表} SELECT {属性标识符} FROM {数据表/视图} WHERE {过 滤条件} [LIMIT (数据条数限制)]
```

编写完成后需进行语法校验。

c)保存并运行SQL任务。

(近行成功)								
下发数据至设备								
□ 保存 ③ 运行 × 取消 贷 定时运行 □ 语法校验 ① 生成API		③最近保存时间: 2019-07-21 21:43:07						
<pre>1 以下示例中的函数均可在函数列表中找到 2 空调商场温度将医器上把者度且上找时间在五分钟内,且高温数量传感器数量大于5 3 空调商场温度将医器上比相度 TePEArture AS 5 SELECT Count(CurrentTemperature) AS high_temp_num, Areaid_AVX(CurrentTemperature) AS current_temper 5 FROM 5 K. 7 Index CurrentTemperature IS NOT NULL AND DATE_DIFF('NIINTE',FROM_UNIXTIME(Sevent_time/1000),NOM()) &lt; 60 AND CurrentTemperature &gt; 26 6 0000 FOY Areaid (AVXINC count(CurrentTemperature) &gt; 5; 6 01 1 INSERT INTO 5 (pc )</pre>								
11 DISERT INTO \${ps								
运行日志 结果		♀ <b>下戦</b> CSV						
PowerSwitch	WindSpeed	WorkMode						
1	0	1						
总数1条								

### 7. 查看HelloWorld.java文件中设备接收到的下发属性。

Runc	🗟 HelloWork	4 ×	\$• ±.
đ.	2019-0	17-21 05:20:36.407 - null[MqttDefaulCallback.java] - messageArrived(74):MqttDefaulCallback:messageArrived, topic = [/sys/ ////////////////////////////////	g
•	🤹 . ser	vice.property.set", "id": "968974803", "params": ("PowerSwitch":1, "WindSpeed":0, "WorkMode":1), "version": "1.0.0"}],	
н -	1 2019-0	17-21 05:20:36.407 - null[PersistentEventDispatcher. java] - broadcastMessage(150):com. aliyun. alink. linksdk. channel. core. persistent. event. PersistentEventDispatcher:broadcastMessage(), what=3	
	2019-0	77-21 05:20:36.407 - 15 / ? D/PersistentConnect:shouldHandle(),topic =/sys/	
-11	2019-0	17-21 05:20:36.407 - null[PersistentEventDispatcher. java] - doNotify (206):com.aliyun.alink.linksdk.channel.core.persistent.event.PersistentEventDispatcher:enter doNotify	
8	2019-0	V7-21 05:20:36.407 - null[PersistentEventDispatcher.java] - doNotify(213):com.aliyun.alink.linksdk.channel.core.persistent.event.PersistentEventDispatcher:call onCommand	
2	2019-0	V7-21 05:20:36.407 - 15 / ? D/PersistentConnect:onCommand(),topic = /sys/ //thing/service/property/set	
×	2019-0	77-21 05:20:36.407 - 15 / ? D / PersistentConnect:onCommand(),data = [B@a200880	
	2019-0	V7-21 05:20:36.407 - 15 / ? D/ConnectManager:onNotify(), connectId = LINK_PERSISTENT , topic = /sys/	
	2019-0	77-21 05:20:36.408 - 15 / ? D / ConnectManager:onNotify(), listener = com. aliyun. alink. linksdk. channel. gateway. impl. GatewayChannel	
	2019-0	Y7-21 05:20:36.408 - 15 / ? D / ConnectManager:onNotify(),item should handle return false	
	2019-0	V7-21 05:20:36.408 - 15 / ? D / ConnectManager:onNotify(), listener = com.aliyun. alink. dm. deliver. LinkKitNotifyDispatcher	
	2019-0	V7-21 05:20:36.408 - 15 / ? D / ConnectManager:onNotify(), send notify	
	2019-0	V7-21 05:20:36.408 - 15 / ? D/LinkKitNotifyDispatcher:onNotify() called with: topic = [/sys/ LinkKitNotifyDispatcher:onNotify() called with: topic = [/sys/	
	. AMe	ssage@4fab48f7]	
	2019-0	V7-21 05:20:36.408 - 15 / ? D / ConnectManager:onNotify(), listener = com.aliyun.alink.linksdk.tmp.connect.entity.cmp.CmpNotifyManager@c689af6	
	2019-0	V7-21 05:20:36.408 - 15 / ? D / [Tmp]CmpNotifyManager:shouldHandle connectId:LINK_PERSISTENT topic:/sys/	
	2019-0	V7-21 05:20:36.408 - 15 / ? D / [Tmp]CmpNotifyManager:getRegistedTopic connectId:LINK_PERSISTENT topic:/sys/	
	real	Topic:/sys/ thing/service/property/set	
	2019-0	V7-21 05:20:36.408 - 15 / ? D/[Tmp]CmpNotifyManager:onNotify handler:null realTopic:/sys/ thing/service/property/set	
	2019-0	V7-21 05:20:36.408 - 15 / ? D/ConnectManager:onNotify(),item should handle return false	
	2019-0	V7-21 05:20:36.408 - null[PersistentEventDispatcher.java] - doNotify(206):com.aliyun.alink.linksdk.channel.core.persistent.event.PersistentEventDispatcher:enter doNotify	
	2019-0	V7-21 05:20:36.408 - null[PersistentEventDispatcher.java] - doNotify(213):com.aliyun.alink.linksdk.channel.core.persistent.event.PersistentEventDispatcher:call onCommand	
	2019-0	)7-21 05:20:36.408 - null[MqttDefaulCallback.java] - handleRrpcRequest(112):MqttDefaulCallback:handleRrpcRequest()	
	Il files are up-to	-date (43 minutes ago) 7,81 CRLF1 UTF-81	°a ⊕ ()

至此,您已完成利用数据开发SQL将数据下发至设备的操作。