

# 阿里云 物联网边缘计算

用户指南

文档版本：20181115

# 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 <b>Ctrl + A</b> 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。
斜体	表示参数、变量。	<code>bae log list --instanceid Instance_ID</code>
[ ]或者[a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ }或者{a b}	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
<b>1 配置边缘计算节点.....</b>	<b>1</b>
1.1 控制台创建网关.....	1
1.2 搭建边缘环境.....	2
<b>2 流数据分析.....</b>	<b>5</b>
2.1 边缘部分流数据分析.....	5
2.2 边缘部分流数据分析SQL.....	8
2.2.1 概述.....	9
2.2.2 创建源表.....	9
2.2.3 创建结果表.....	13
2.3 使用示例.....	15
<b>3 驱动管理.....</b>	<b>18</b>
3.1 驱动概述.....	18
3.2 官方驱动.....	19
3.3 自定义驱动.....	21
3.4 设备驱动应用示例.....	24
<b>4 边缘实例.....</b>	<b>28</b>

# 1 配置边缘计算节点

## 1.1 控制台创建网关

边缘计算节点，即承载计算能力的边缘网关。配置边缘计算节点包含控制台创建网关和搭建边缘环境两部分内容，本节介绍在控制台创建边缘网关与设备的步骤。

### 操作步骤

1. 登录物联网平台控制台。
2. 选择设备管理 > 产品、单击创建产品。在弹出页面，按如下参数设置说明设置参数后，单击确认，创建网关产品。

产品是一批具有相同功能的设备的集合，您可以创建产品，用于批量管理设备。比如，同一产品型号的设备，功能都相同。您可以为该型号创建产品，以此来批量管理设备。

创建产品

\* 版本选择：  
 基础版  高级版

产品名称：  
gateway\_linkedge

\* 节点类型：  
 设备  网关

设备类型：  
边缘网关

\* 数据格式：  
Alink JSON

\* 使用ID认证：  
 是  否

产品描述：  
请输入产品描述

确认 取消

### 参数设置说明：

- 版本选择：选择高级版。
- 产品名称：在此处为产品命名，产品名称需保持账号内唯一。
- 节点类型：在此处选择产品类型。此例中，选择网关。
- 设备类型：在此处选择边缘网关。

- 数据格式：在此处选择该产品的数据传输格式。边缘网关请选择**Alink JSON**。
  - 使用ID<sup>2</sup>认证：ID<sup>2</sup>是物联网平台提供了一种安全认证方式，边缘计算中并不涉及这种认证。此处，请选择否。
  - 产品描述：使用文字描述该产品，可为空。
3. 单击设备管理、添加设备。按如下参数设置说明设置参数后，单击确认，创建一个网关设备。

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：

DeviceName：

参数设置说明：

- 产品：选择上一步中创建的产品。
- **DeviceName**：为该网关设备命名。DeviceName需保持产品内唯一。如不填写，系统将自动生成。



说明：

DeviceName支持大写字母[A-Z]、小写字母[a-z]、数字[0-9]和下划线( \_ )。且不能以下划线开头或结尾。

4. 网关设备创建完成后，系统会弹出设备证书，即设备三元组（包括ProductKey、DeviceName和DeviceSecret）。您可以一键复制并保存这些信息，用于后续设备开发使用。

## 1.2 搭建边缘环境

本节介绍如何使用Docker镜像搭建边缘环境，配置边缘计算节点。

前提条件

目前支持在以下系统中搭建边缘环境：

- Windows x86\_64

- MAC x86\_64
- Ubuntu 18.04 x86\_64
- Ubuntu 16.04 x86\_64

## 操作步骤

### 1. 安装Docker环境。

- a) 下载Docker软件，具体请参考[Docker官网](#)。
- b) 安装Docker，具体安装方法请参考[Docker官方文档](#)。



#### 说明：

若您的系统为Linux版，请至Docker Store下载安装适配您Linux发行版的Docker客户端。

### 2. 下载边缘计算节点启动脚本。

- Windows或Mac版：`curl -O http://aliyun-iotedge.oss-cn-hangzhou.aliyuncs.com/link-iot-edge.sh`
- Linux版：`wget http://aliyun-iotedge.oss-cn-hangzhou.aliyuncs.com/link-iot-edge.sh`

### 3. 启动边缘计算节点，使设备上线。

```
./link-iot-edge.sh {version} {productkey} {devicename} {devicesecret}
```



#### 说明：

- 请将{**version**}替换为需要的[Docker镜像版本号](#)，建议您使用最新的版本号。
- 请将{**productkey**} {**devicename**} {**devicesecret**}替换为实际的网关设备三元组信息。
- 执行此命令，会删除已存在的Docker容器，创建新的Docker容器，并保存网关三元组信息。

### 4. 边缘计算节点成功启动后，您可以在设备管理界面，对应网关设备详情页上，看到设备状态变为在线。



5. (可选) 停止边缘计算节点。

```
./link-iot-edge.sh --stop
```



说明：

执行此命令会终止边缘计算节点的运行，但不会删除Docker容器。若您需要删除Docker容器可以使用docker rm命令。

6. (可选) 重启边缘计算节点。

```
./link-iot-edge.sh --restart {version}
```



说明：

- 请将{version}替换为您的Docker镜像版本号。
- 如果已执行3中的命令，启动过边缘计算节点，使用此命令可以实现快速重启，并且不再需要配置网关设备三元组信息。
- 如果边缘计算节点已被删除，使用此命令会重新拉取指定版本的Docker镜像，并启动新的Docker容器。

## 2 流数据分析

### 2.1 边缘部分流数据分析

**流数据分析**是一种用流的方法快速实时处理数据的计算方式，边缘计算中的流数据分析，继承于物联网平台数据分析产品的流数据分析的能力，可以无缝对接数据分析产品，在数据分析产品中创建流数据分析任务，并将该任务下发到边缘端，通过边缘设备实时运行。运行结果可以存储在边缘端也可以直接传输至云端。

边缘部分流数据分析主要特点如下：

- 运行在边缘端，不依赖网络，低时延。
- 对数据进行采集、清洗、加工、聚合之后再上云，大大减少数据传输成本。
- 支持对消息乱序的处理。

#### 操作步骤

1. 登录[物联网平台控制台](#)。
2. 参考[流数据分析](#)，创建、设置并发布数据分析任务。

其中，执行任务需选择边缘端。



说明：

图片以组件编排任务为例。

### 创建任务 ×

**\* 任务类型：**  
 组件编排  SQL

**\* 任务名称：**

**\* 任务描述：**

**\* 执行任务：**  
 云端  边缘端

3. 单击左侧导航栏中边缘计算 > 边缘实例。

找到要分配流数据分析任务的实例，单击对应操作栏中的查看。

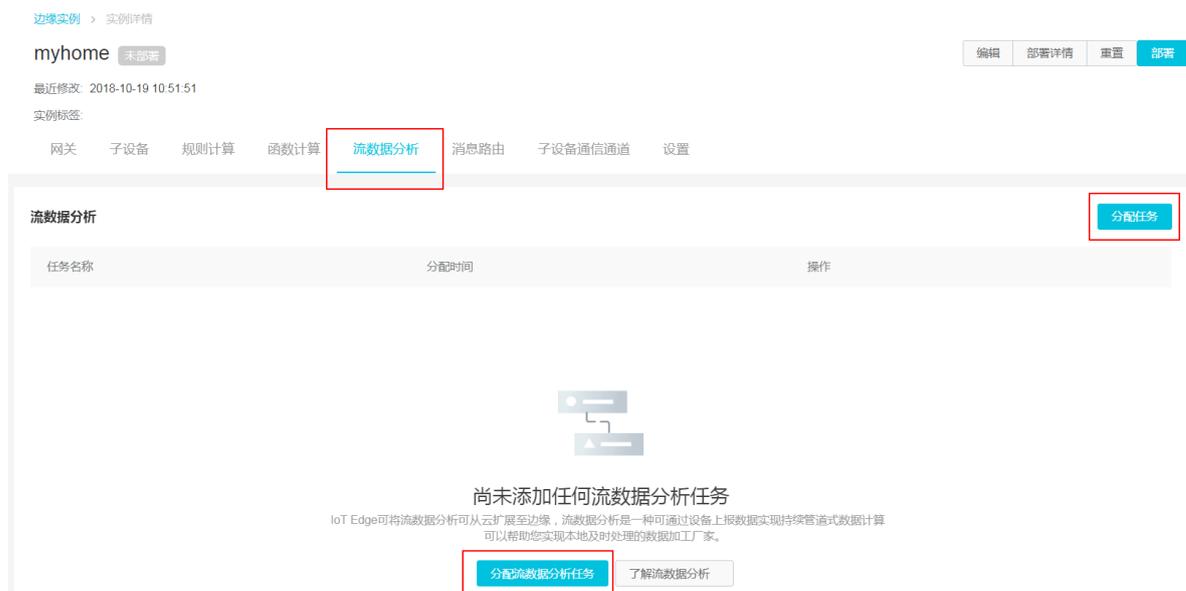
物联网平台 **边缘实例**

实例列表 刷新

实例

实例名称	网关名称	网关产品	实例状态	创建时间	修改时间	操作
myhome	linshi_gw	gateway_linkedge	未部署	2018-06-06 10:40:50	2018-10-19 10:51:51	<input checked="" type="button" value="查看"/> <input type="button" value="删除"/>

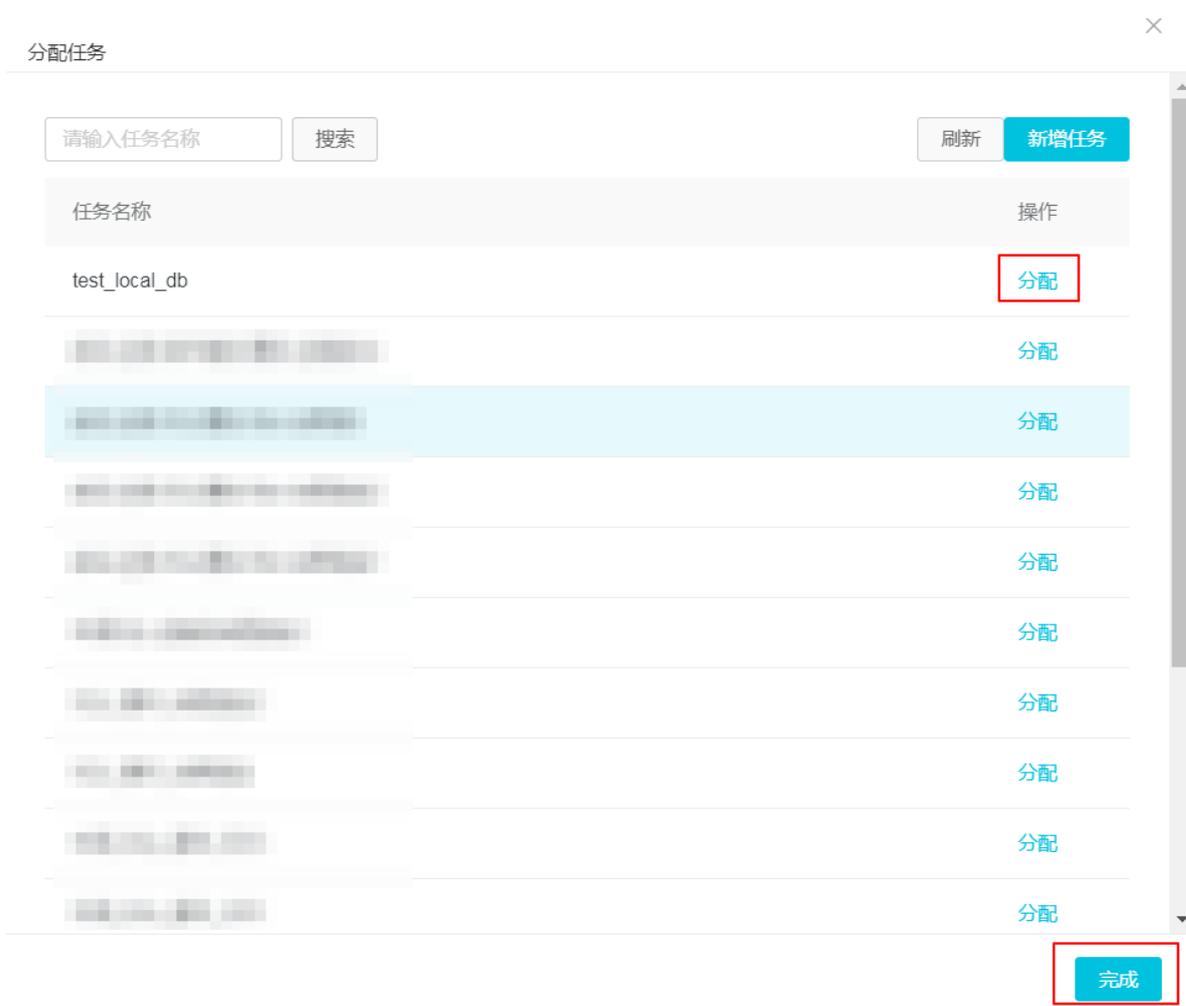
4. 在实例详情页面，选择流数据分析，然后单击分配任务或分配流数据分析任务。



5. 在弹出的分配任务页面中，选择要分配给实例的任务，单击分配。

本例中，流数据分析任务名称为test\_local\_db。

分配成功后，单击完成。



完成规则添加后，便可在该实例的实例详情页面上，查看该实例的流数据分析任务信息。

- 6. 部署实例。为实例分配任务后，单击页面右上角部署，并在弹出对话框中，单击确认。

您可以在本页面中查看部署状态和部署详情。



## 2.2 边缘部分流数据分析SQL

## 2.2.1 概述

您可以创建SQL类型的流数据分析任务，并在边缘端执行该任务。

边缘端SQL语法与云端Flink SQL语法完全一致，语法说明请参见[Flink SQL](#)。

但由于运行环境不一样，边缘端与云端支持的数据存储完全不一样。在流数据分析中，数据存储用源表、维表、结果表来表示。

类型	描述
源表	流式数据存储，驱动流数据分析的运行。每个流数据分析任务必须提供至少一个流式数据存储。 <ul style="list-style-type: none"> <li>消息路由：边缘消息路由，会将消息路由配置的数据导入到流数据分析任务中。</li> </ul>
维表	关系型数据存储，常用来做数据连接（Join）。边缘端暂不支持维表。
结果表	目的数据存储，用于存储流数据分析的计算结果。每个流数据分析任务必须提供至少一个目的数据存储。 <ul style="list-style-type: none"> <li>消息路由：边缘消息路由，会将流数据分析任务的处理结果输出到边缘消息路由中。</li> <li>RDS for MySQL：会将流数据分析任务的处理结果输出到阿里云数据库中。</li> <li>File：会将流数据分析任务的处理结果输出到边缘网关的文件中，主要用于边缘端调试。</li> </ul>

## 2.2.2 创建源表

本章介绍边缘端SQL创建源表的语法、示例及参数说明。

创建源表的语法如下：

```
CREATE TABLE tableName
  (columnName dataType [, columnName dataType]*)
  | PRIMARY KEY (key_part,...)
  [ WITH (propertyName=propertyValue [, propertyName=propertyValue]*) ];
```

### 消息路由

示例：

```
create table property (
  propertyName varchar,
  propertyValue varchar,
  productKey varchar,
```

```

deviceName varchar,
gmtCreate varchar,
ts varchar,
tstamp as to_timestamp (cast (ts as bigint)),
WATERMARK wk FOR tstamp as withOffset (tstamp, 2000)
) with (
    type = 'custom',
    tablefactoryclass = 'com.alibaba.blink.streaming.connector.edgehub.EdgeHubTableFactory',
    jsonParser = 'device_property'
);
    
```

WITH参数说明如下：

参数	描述
type	固定值custom，与tablefactoryclass共同使用，表示获取边缘端消息路由的消息。具体的消息流转方式，需要您在控制台配置消息路由规则。
tablefactoryclass	固定值com.alibaba.blink.streaming.connector.edgehub.EdgeHubTableFactory，与type共同使用，表示获取边缘端消息路由的消息。具体的消息流转方式，需要您在控制台配置消息路由规则。
jsonParser	定义消息解析器，取值如下： <ul style="list-style-type: none"> <li>• device_property：属性消息，不符合属性消息格式的消息将会被丢弃</li> <li>• device_event：事件消息，不符合事件消息格式的消息将会被丢弃</li> <li>• default：不使用解析器，直接使用json格式字符串，不丢弃任何消息</li> </ul> 不同的jsonParser对应不同的字段，详情请见下文字段说明内容。

字段说明：

- jsonParser = 'device\_property'：

示例：

```

create table property (
    propertyName varchar,
    propertyValue varchar,
    productKey varchar,
    deviceName varchar,
    gmtCreate varchar,
    ts varchar,
    tstamp as to_timestamp (cast (ts as bigint)),
    WATERMARK wk FOR tstamp as withOffset (tstamp, 2000)
) with (
    type = 'custom',
    tablefactoryclass = 'com.alibaba.blink.streaming.connector.edgehub.EdgeHubTableFactory',
    jsonParser = 'device_property'
);
    
```

```
);
```

字段名	类型	描述
productKey	String	产品的唯一标识ProductKey。
deviceName	String	设备名称，该产品下设备唯一标识。
propertyName	String	属性名。
propertyValue	String	属性值。
time	String	消息产生时间。
gmtCreate	String	流数据分析接收到消息的时间。

- jsonParser = 'device\_event' :

示例 :

```
create table event (
  eventCode varchar,
  params varchar,
  productKey varchar,
  deviceName varchar,
  gmtCreate varchar,
  ts varchar,
  tstamp as to_timestamp (cast (ts as bigint)),
  WATERMARK wk FOR tstamp as withOffset (tstamp, 2000)
) with (
  type = 'custom',
  tablefactoryclass = 'com.alibaba.blink.streaming.connector.
edgeworker.EdgeHubTableFactory',
  jsonParser = 'device_event'
);
```

字段名	类型	描述
productKey	String	产品的唯一标识ProductKey。
deviceName	String	设备名称，该产品下设备唯一标识。
eventCode	String	事件标识符。
params	String	事件参数。
time	String	事件产生时间。
gmtCreate	String	流数据分析收到事件消息的时间。

- jsonParser = 'default' :

示例 :

```
create table property (
  json varchar,
```

```

jsonType varchar,
gmtCreate as to_timestamp (
    cast (json_value (json, '$.gmtCreate') as bigint)
),
deviceName as json_value (json, '$.deviceName'),
productKey as json_value (json, '$.productKey'),
ts as to_timestamp (
    cast (
        json_value (json, '$.items.temperature.time') as bigint
    )
),
temperature as cast (
    json_value (json, '$.items.temperature.value') as int
)
) with (
    type = 'custom',
    tablefactoryclass = 'com.alibaba.blink.streaming.connector.
edgehub.EdgeHubTableFactory'
);
    
```

字段名	描述
json	从消息路由获取到的消息内容，该消息内容的格式为字符串型json格式。
jsonType	从消息路由获取到的消息类型。 <ul style="list-style-type: none"> <li>• device_property</li> <li>• device_event</li> </ul>

Json格式：

— jsonType = device\_property

```

{
  "gmtCreate": 1510292739881,
  "items": {
    "attribute_9": {
      "time": 1510292697471,
      "value": 560542025
    },
    "attribute_8": {
      "time": 1510292697470,
      "value": 715665571
    }
  },
  "productKey": "X5eCzh6fEH7",
  "deviceName": "xxxxxxxxxxxxxxxxxxxx"
}
    
```

— jsonType = device\_event

```

{
  "eventCode": "BrokenInfo",
  "value": {
    "Power": "on",
    "structParam": {
    
```

```

        "param1": "abc",
        "param2": 123
    },
    "time": 1510799670074,
    "productKey": "5RS5XTnNADg",
    "deviceName": "xxxxxxxxxxxxxxxxxxxx",
    "gmtCreate": 1510799670074
}

```

### 2.2.3 创建结果表

流数据分析使用CREATE TABLE作为输出结果数据的格式定义，同时定义数据如何输出到目的数据存储中。

创建结果表的语法如下：

```

CREATE TABLE tableName
    (columnName dataType [, columnName dataType]*)
    [ WITH (propertyName=propertyValue [, propertyName=propertyValue]*) ];

```

#### 消息路由

示例：

```

create table edgehub_output (
    productKey varchar,
    deviceName varchar,
    temperature int,
    productTime timestamp,
    inputTime timestamp,
    outputTime timestamp,
    delayTime bigint
) with (
    type = 'custom',
    class = 'com.alibaba.blink.streaming.connector.edgehub.sink.EdgeHubTableSink',
    topic = '/sys/streamcompute/high_temperature'
);

```

WITH参数说明如下：

参数	描述
type	固定值custom，与class共同使用，表示输出消息到边缘端。具体的消息流转方式，需要您在控制台配置消息路由规则。
class	固定值com.alibaba.blink.streaming.connector.edgehub.sink.EdgeHubTableSink，与type共同使用，表示输出消息到边缘端。具体的消息流转方式，需要您在控制台配置消息路由规则。

参数	描述
topic	消息流转到的目的topic，根据需求自定义参数值。

### RDS for MySQL

示例：

```
create table db (
  productKey varchar,
  deviceName varchar,
  ts timestamp,
  temperature int
) with (
  type = 'rds',
  url = 'jdbc:mysql://${host}:${port}/${dbName}',
  tableName = '${tableName}',
  userName = '${userName}',
  password = '${password}'
);
```

WITH参数说明如下：

参数	描述
type	固定值rds，表示结果表的类型为RDS。
url	RDS数据库地址，取值为jdbc:mysql://\${host}:\${port}/\${dbName}。其中： <ul style="list-style-type: none"> <li>• host：MySQL数据库host地址。</li> <li>• port：MySQL数据库端口号。</li> <li>• dbName：数据库名。</li> </ul> 示例: jdbc:mysql://192.168.1.5:3306/myDB
tableName	表名。
userName	数据库用户名。
password	数据库密码。

更多参数介绍，请参考[创建云数据库结果表](#)中RDS相关参数说明。

### File

示例：

```
create table print_sink (
  productKey varchar,
  deviceName varchar,
  eventCode varchar,
  ts timestamp,
```

```

    temperature int
  ) with (
    type = 'custom',
    class = 'com.alibaba.blink.connector.file.FileSink',
    tag = 'prop_filter_b',
    filePath = '/linkedge/run/debug/prop_filter_b.txt'
  );

```

with参数说明如下：

参数	描述
type	固定值custom，与class同时使用，表示输出消息到文件。
class	固定值com.alibaba.blink.connector.file.FileSink，与type同时使用，表示输出消息到文件。
tag	可选参数，表示为输出的消息设置标签。 <ul style="list-style-type: none"> <li>若设置该参数，表示每条输出的消息前都附带该参数的取值。</li> <li>若无需为输出消息设置标签，则删除参数。</li> </ul>
filePath	输出消息的目的文件路径。

示例表示，将消息输出到本地/linkedge/run/debug/prop\_filter\_b.txt文件中，并在每条输出消息前带prop\_filter\_b标签。

## 2.3 使用示例

设备数据可通过消息路由能力流转至流数据分析，您可以利用流数据分析方便地对设备数据进行实时计算分析并将结果输出到RDS或本地文件中。

本文以设置当CPU使用率大于10%时输出报警信息到为本地文件的任务为例，说明边缘部分流数据分析的使用方法。

### 操作步骤

1. 登录[物联网平台控制台](#)。
2. 参考[流数据分析](#)，创建、设置并发布数据分析任务。

其中，任务类型需选择**SQL**，执行任务需选择**边缘端**。

创建任务
✕

---

**\* 任务类型：**

组件编排  SQL

**\* 任务名称：**

**\* 任务描述：**

**\* 执行任务：**

云端  边缘端

3. 为创建的任务，编辑SQL。

```

-- CPU高负载报警
create table property (
  propertyName varchar,
  propertyValue varchar,
  productKey varchar,
  deviceName varchar,
  gmtCreate varchar,
  ts varchar
) with (
  type = 'custom',
  tablefactoryclass = 'com.alibaba.blink.streaming.connector.
edgehub.EdgeHubTableFactory',
  jsonParser = 'device_property'
);

create table print_file (
  productKey varchar,
  deviceName varchar,
  cpu double,
  ts timestamp,
  gmtCreate timestamp
) with (
  type = 'custom',
  class = 'com.alibaba.blink.connector.file.FileSink',
  filePath = '/linkedge/run/debug/high_cpu_alter.txt'
);
    
```

```
insert into
  print_file
select
  productKey,
  deviceName,
  cast (propertyValue as double),
  to_timestamp (cast (ts as bigint)),
  to_timestamp (cast (gmtCreate as bigint))
from
  property
where
  propertyName = 'cpu_usage'
  and cast (propertyValue as double) > 10;
```

4. 保存任务并发布。
5. 参考[边缘部分流数据分析](#)，将任务分配到已有的边缘实例中，并部署边缘实例。
6. 登录到边缘网关查看报警信息。

进入 `/linkedge/run/debug/high_cpu_alter.txt` 文件，查看输出结果。

```
2018-10-13 15:01:26.404 -> productKey, deviceName, cpu, ts, gmtCreat
2018-10-13 15:01:43.964 -> alhVtA6RAiU,gateway1010,21.243523,2018-10
-13 15:01:43.6,2018-10-13 15:01:43.618
2018-10-13 15:02:13.607 -> alhVtA6RAiU,gateway1010,20.971867,2018-10
-13 15:02:13.587,2018-10-13 15:02:13.597
```

## 3 驱动管理

### 3.1 驱动概述

驱动是用于做联网设备数据采集或者控制的通信协议程序。

#### Link IoT Edge驱动

Link IoT Edge的驱动分为阿里官方驱动（简称官方驱动）和用户自定义驱动（简称自定义驱动）。

- 官方驱动：默认包含在Link IoT Edge服务内，目前包括Modbus和OPC UA官方驱动，Light和LightSensor官方示例驱动四种。
- 自定义驱动：是由您自定义创建的驱动，可以通过驱动管理页面上传驱动文件。目前支持Node.js8和Python3版本的驱动，上传的驱动建议您按照[开发指南](#)中的方式开发，您需要确保该驱动的正确性。

使用驱动的常用流程是：

1. 创建驱动。
2. 将驱动分配到边缘实例中。
3. 分配设备到边缘实例，并关联驱动。
4. 为该实例中的网关设备添加驱动通道。
5. 部署边缘实例。

#### 驱动管理

驱动管理是指驱动的管理服务，Link IoT Edge驱动管理提供如下服务：

操作	描述
创建驱动	除了使用官方驱动以外，您可以自定义创建驱动。创建方法请见 <a href="#">自定义驱动</a> 。
编辑驱动	创建完驱动后，通过编辑驱动修改驱动名称、驱动描述或重新上传驱动文件。
删除驱动	<p>Link IoT Edge支持对已创建的自定义驱动进行删除。删除驱动会影响所有使用该驱动的设备，请谨慎操作。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <b>说明：</b> 若要删除的驱动已分配到部署成功的边缘实例中，那么删除该驱动后边缘实例仍然可以运行，若重新部署该边缘实例，将影响边缘实例的运行。</p> </div>

操作	描述
移除驱动	对于已分配到边缘实例中的驱动，Link IoT Edge支持移除驱动，移除驱动会影响该实例中分配的设备，请谨慎操作。

## 3.2 官方驱动

官方驱动是由阿里云提供的通信协议驱动，包括Modbus、OPC UA官方驱动和Light、LightSensor官方示例驱动。

### Modbus驱动介绍

Modbus是常用的应用层数据通信协议，阿里云官方Modbus驱动（以下简称Modbus驱动）支持Modbus-RTU和Modbus-TCP两种交互。

Modbus驱动支持的功能有读取输入状态和输入寄存器，读/写线圈状态和保持寄存器。

Modbus驱动支持从控制台下载，您可以对下载后的驱动代码进行更改，可作为您的自定义代码使用。

Modbus驱动使用步骤如下：

1. 创建高级版产品，并选择接入网关协议为Modbus，具体创建产品步骤请参见[创建产品\(高级版\)](#)。
2. 为产品添加设备，具体添加设备步骤请参见[单个创建设备](#)和[批量创建设备](#)。
3. 为产品定义物模型，具体定义方法请参见[新增物模型](#)。
4. 完成上述设备配置后，需要配置设备与网关的交互方式。
  - a. 创建子设备通道，具体方法请参见[子设备通道管理](#)。
  - b. 添加子设备，具体方法请参见[子设备管理](#)。
5. 到边缘实例中分配相关设备和网关，选择Modbus驱动进行部署。

### OPC UA驱动介绍

Link IoT Edge产品中提供用于接入OPC UA设备的驱动，下面简称OPC UA驱动。

OPC UA驱动支持从控制台下载，您可以对下载后的驱动代码进行更改，可作为您的自定义代码使用。

OPC UA驱动包含三部分功能：

- 负责接入OPC UA设备。

- 对接入设备数据进行转换，符合阿里云物模型规范。
- 设备数据上行和下行控制。

OPC UA驱动使用步骤如下：

1. 创建高级版产品，并选择设备协议为OPC UA，具体创建产品步骤请参见[创建产品\(高级版\)](#)。
2. 为产品添加设备，具体添加设备步骤请参见[单个创建设备](#)和[批量创建设备](#)。
3. 为产品定义物模型，具体定义方法请参见[新增物模型](#)。
4. 完成上述设备配置后，需要配置设备与网关的交互方式，目前仅支持OPC UA交互。
  - a. 创建子设备通道，具体方法请参见[子设备通道管理](#)。

其中，对参数进行如下设置：

参数	参数设置
通道名称	设置OPC UA Server通道名称
连接地址	OPC UA Server监听地址
用户名	登录OPC UA Server用户名
密码	登录OPC UA Server密码
安全模式	访问OPC UA Server安全数据交互加密策略
安全策略	访问OPC UA Server数据交互加密算法
方法调用超时	OPC UA驱动调用OPC UA Server方法调用超时时间
数字证书	访问OPC UA Server鉴权认证证书
私钥证书	访问OPC UA Server数据加密密钥

- b. 添加子设备，具体方法请参见[子设备管理](#)。

其中，对参数进行如下设置：

参数	参数设置
产品	使用OPC UA驱动的产品
设备名	使用OPC UA驱动的设备
关联通道	已创建的OPC UA Server通道
节点路径	设备在OPC UA服务器下的节点路径

5. 到边缘实例中分配相关设备和网关，选择OPC UA驱动进行部署。

## Light驱动介绍

Light驱动是Link IoT Edge提供的一款智能灯泡的模拟驱动，是用于说明驱动开发方式的示例驱动。

Light驱动支持智能灯泡的开关操作，可以用于模拟快速入门中的灯设备。

Light驱动的使用示例请参见[快速入门](#)。

## LightSensor驱动介绍

LightSensor驱动是Link IoT Edge提供的一款光照传感器的模拟驱动，是用于说明驱动开发方式的示例驱动，光照强度数据按照一定周期重复，可以用于模拟快速入门中的光照传感器设备。

LightSensor驱动的使用示例请参见[快速入门](#)。

## 3.3 自定义驱动

自定义驱动是您利用Link IoT Edge开发包开发并上传到驱动管理中的驱动程序，可以使用Link IoT Edge的边缘实例部署功能部署到边缘网关中，您需要确保该驱动程序可执行且功能符合您的要求。

### 操作步骤

1. 登录[物联网平台控制台](#)。
2. 在左侧导航栏中，单击边缘计算 > 驱动管理。
3. 在驱动管理页面，单击右侧创建驱动。

# 驱动管理

## 驱动管理列表

请输入驱动名称

驱动名称	驱动语言
Modbus <span>官方</span>	python
OPCUA <span>官方</span>	python

4. 设置驱动参数。

## 新建驱动

---

\* 语言类型：

nodejs8

\* 驱动名称：

test\_driver

驱动描述：

请输入驱动描述

\* 驱动文件：

上传文件



Light.zip上传成功

参数说明如下：

- 语言类型：驱动的语言类型，支持Node.js8和Python3类型。
- 驱动名称：为您的驱动设置名称，同一个账号内驱动名称必须为唯一。
- 驱动描述：描述您创建的驱动，可以为空。
- 驱动文件：单击上传文件，上传您的驱动文件。推荐您按照Link IoT Edge[开发指南](#)中的方式开发您的驱动文件。

5. 单击确认，完成自定义驱动的创建。

可在驱动管理列表中查看创建的驱动。

6. 自定义驱动创建完成后关联边缘网关与使用该驱动的设备。

a) 创建子设备通道，具体方法请参见[子设备通道管理](#)。

b) 添加子设备，具体方法请参见[子设备管理](#)。

7. 驱动创建完成之后，需要分配到边缘实例中，详细操作内容请见[边缘实例](#)中分配设备驱动的步骤。

8. （可选）单击驱动名称右侧的编辑，可以修改已创建的驱动名称、驱动描述或重新上传驱动文件。

9. （可选）单击驱动名称右侧的删除，可以删除已创建的自定义驱动。

## 3.4 设备驱动应用示例

本节为您介绍本地设备连接方法，通过设备驱动可实现设备连接网关。设备连接网关的API，请参见[设备接入开发](#)。

### 前提条件

您需要准备一个驱动文件。建议您按照[开发指南](#)中的方式开发驱动文件，并确保该驱动文件的正确性和可用性。

### 背景信息

以定义光照传感器数据发送至云端为例。

### 操作步骤

1. 在物联网平台控制台，选择边缘计算 > 驱动管理
2. 在驱动管理页面，单击新建驱动，创建驱动。

新增驱动
✕

---

**\* 语言类型：**

nodejs8
▼

**\* 驱动名称：**

Light\_Sensor
?

**驱动描述：**

请输入驱动描述
0/100

**\* 驱动文件：**

重新上传
?

Light (1).zip上传成功
✕

确定

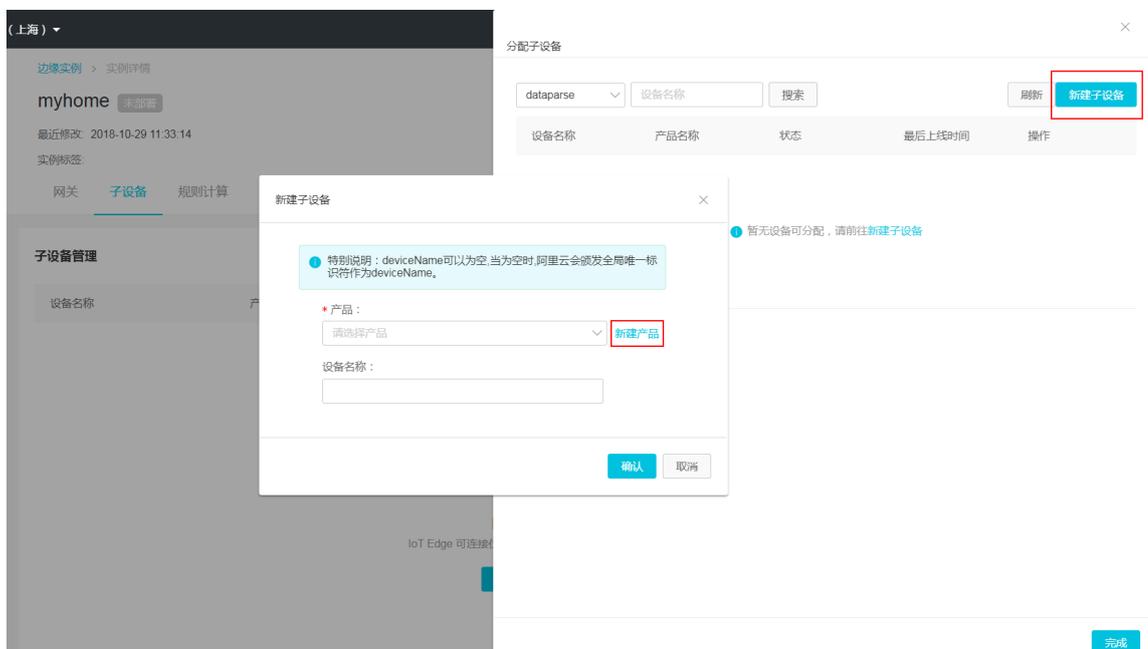
取消

参数	描述
语言类型	选择驱动的语言类型，可以选择 <b>nodejs8</b> 。
驱动名称	可填写为 <b>Light_Sensor</b> 。
驱动描述	描述将要创建的驱动，可以为空。
驱动文件	上传提前准备的驱动文件。

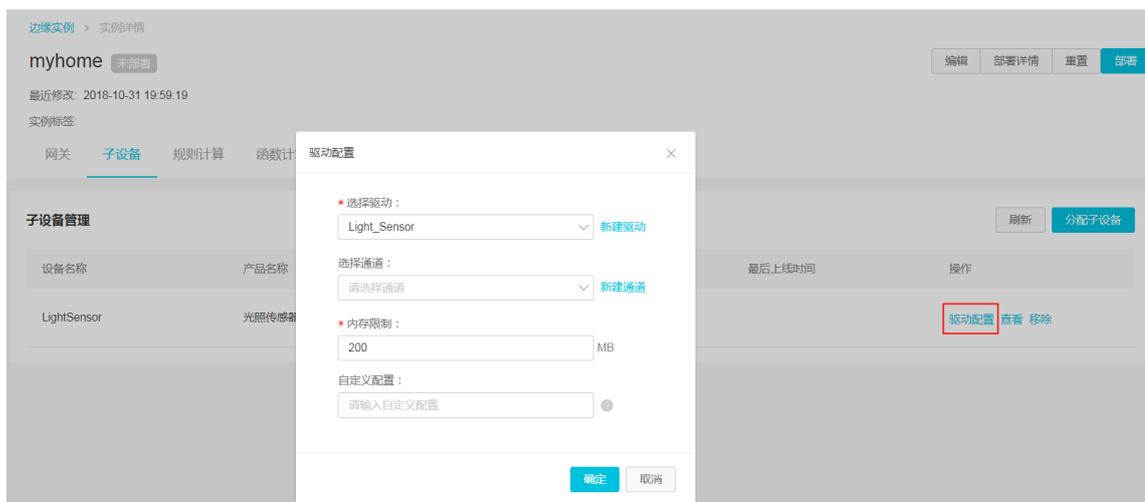
- 在边缘实例页面创建一个边缘实例，关联[控制台创建网关](#)中创建的网关，并配置实例内容。



- a) 在实例详情 > 子设备页面，单击分配子设备。
- b) 在分配子设备页面，单击新建子设备。



- c) 单击新建产品，创建名称为光照传感器的高级版产品。
- d) 设置产品参数后单击确认。
- e) 在新建子设备页面，设置设备名称，单击确认完成子设备的创建。
- f) 分配上一步创建的子设备到边缘实例中。
- g) 为子设备配置步骤2中创建的Light\_Sensor驱动。



4. 单击实例详情页面右上角部署，将已分配到实例的资源部署至网关中。

### 预期结果

部署成功后，设备上报的属性和事件将会每隔2秒被同步到云端，您可以在IoT控制台设备运行状态页面查看具体信息。

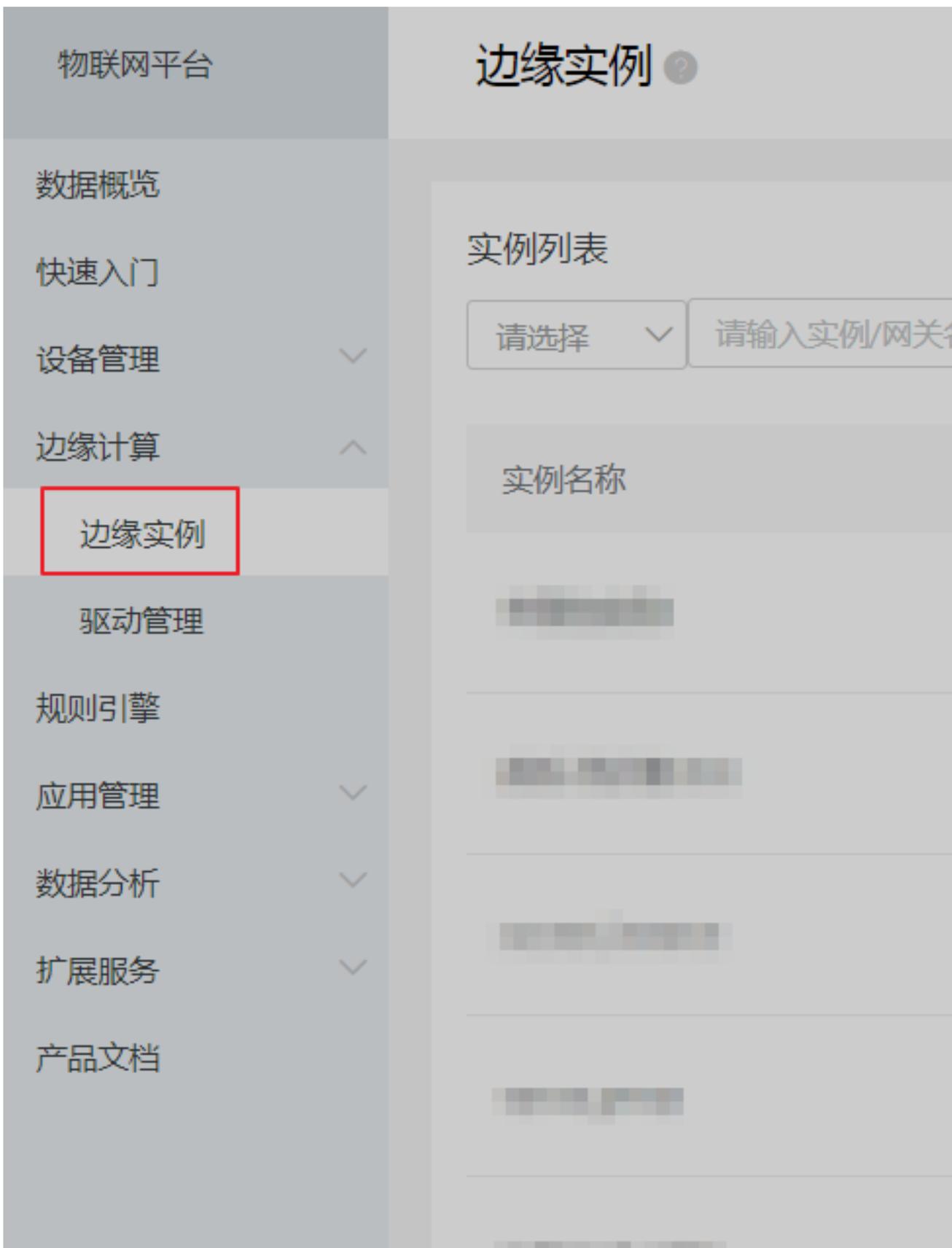
## 4 边缘实例

---

边缘实例是一种类似文件夹的管理功能，您可以通过实例的方式管理边缘相关的网关、设备，同时也可以管理规则计算、函数计算和消息路由内容。通过部署实例，将添加在实例中的资源部署至网关中。

### 操作步骤

1. 创建边缘实例。
  - a) 在 [IoT控制台](#)，选择边缘计算 > 边缘实例。
  - b) 单击新增实例。



- c) 设置实例名称，用于后续的查询及识别实例。我们还为您提供了标签能力，通过标签您可以更加有效的去归类及识别实例。

- d) 关联网关产品和网关设备，选择已创建的网关产品和网关设备。
- e) 单击确认。
- f) 在边缘实例页面，选择已创建的实例，单击右侧的查看，进入实例详情页面。

## 2. 分配网关。

网关页签下，显示已分配的网关产品和网关设备。



说明：

若需要更换网关，则先移除已分配的网关，再根据如下步骤重新分配网关到实例中。

- a) 实例详情页面，选择网关，单击分配网关。
- b) 在分配网关页面中，选择需要关联的网关，单击网关右侧的分配。  
后续部署及重置的指令都将发送至该网关，目前一个实例有且仅有一个网关。
- c) 单击完成。

## 3. 分配子设备。

- a) 在实例详情页面，选择子设备，单击分配子设备。
- b) 在分配子设备页面中，将之前创建的设备（只支持高级版产品创建的设备）分配至实例中，单击设备右侧的分配。  
若无可分配子设备，单击新建子设备创建。

## 分配子设备

我的光照传感器 ▾

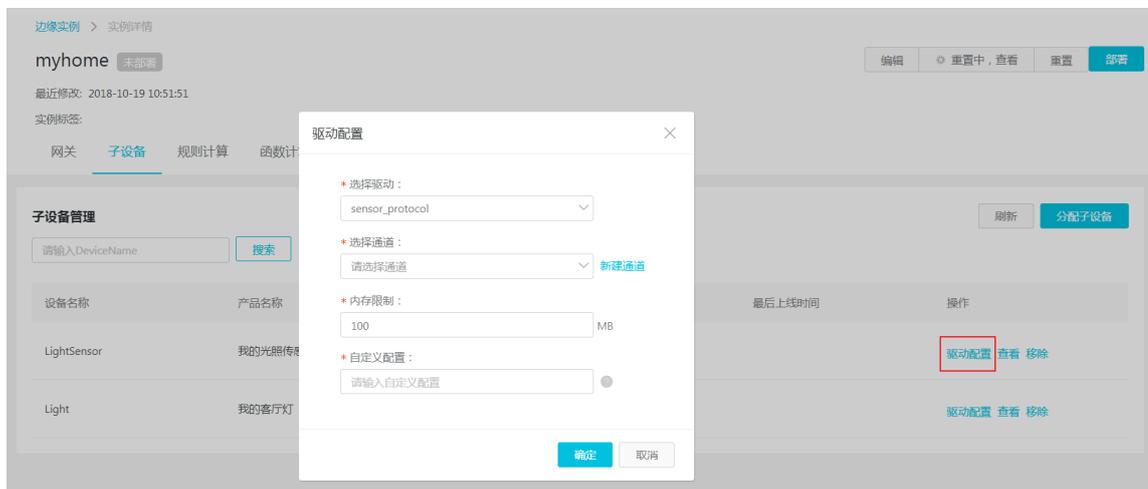
DeviceName

搜索

设备名称	产品名称	状态
LightSensor	我的光照传感器	● 未激活

所有被分配到实例中的设备列表才是最终网关被允许访问的范围。

- c) 单击完成，完成设备的分配。
- d) 单击已分配的子设备右侧的驱动配置，为设备分配官方驱动或自定义驱动。



 **说明：**

- 若使用自定义驱动，需要根据[自定义驱动](#)创建驱动，并分配到边缘分组中。
- 若使用官方驱动，需要根据[官方驱动](#)中的驱动使用步骤，创建符合要求的产品和设备。

e) 单击完成，您已成功为设备配置了驱动。

**4. (可选) 分配规则。**

- a) 在实例详情页面，选择规则计算，单击分配规则。
- b) 在分配规则页面中，将之前创建的[规则计算](#)中的规则分配至实例中，单击设备右侧的分配。  
这些规则将在部署时发送至网关上，并在网关上执行规则定义的逻辑。
- c) 单击完成，完成分配规则计算到实例中。

**5. (可选) 分配函数。**

- a) 在实例详情页面，选择函数计算，单击分配函数。
- b) 在分配函数页面中，将之前创建的[函数计算](#)中的函数分配至实例中，单击分配。  
这些规则将在部署时发送至网关上，并在网关本地运行。

**6. (可选) 分配任务。**

- a) 在实例详情页面，选择流数据分析，单击分配任务。
- b) 在分配任务页面中，选择已经存在的任务单击分配，使用流数据分析功能对您的数据进行分析处理。

若无可选任务，请参考[任务管理](#)创建任务。

c) 单击完成，您已成功为实例添加了任务。

#### 7. (可选) 添加角色。



说明：

实例中的函数在边缘网关运行时，若要调用阿里云其他云资源（例如调用访问OSS的API），则需要为实例添加角色。

a) 在实例详情页面，选择设置，单击添加角色。

b) 在添加角色页面中，选择已经存在的角色，授予实例操作对应云资源的能力。

边缘分组 > 分组详情

myhome 未部署

最近修改: 2018-09-27 11:04:51

分组标签:

网关      设备      规则计

### 分组角色管理

角色名称

### 添加角色

 选择已经存在的角

\* 角色:

请选择

\* 系统模

权限将根

- 角色：选择已经存在的服务角色。该服务角色必须已经选择IoT物联网为授信服务。若无可选服务角色，请参考[角色](#)内容，创建所需的服务角色。
  - 系统模板授权：权限根据已选的角色，自动选取。若需要编辑权限，请参考[授权](#)内容，编辑角色的授权权限。
- c) 单击确定，您已成功为实例添加了角色。
8. (可选) 添加消息路由。
- a) 在实例详情页面，选择消息路由，单击添加路由。
- b) 在添加消息路由页面中，配置参数，您可以通过[消息路由](#)中查看具体配置方式。
- c) 单击确定，您已成功为实例添加了消息路由。
9. 在实例详情页面，单击子设备通信通道，为实例中的网关和设备配置交互方式。  
详情请参考[子设备通道管理](#)。
10. 部署实例。
- a) 在实例详情页面，单击部署实例，将上述实例中添加的资源部署至网关中。



说明：

当您实例中的网关处于离线状态时，您的部署请求将被托管，直到网关上线后平台将主动通知网关进行资源部署操作。

边缘分组 > 分组详情

myhome **部署成功**

最近修改: 2018-08-31 17:45:47

分组标签:

网关 设备 规则计

### 分组网关

设备名称

mygw

### 部署详情

动作内容

DeviceScript.Ligh

DeviceScript.Ligh

DeviceScript.Ligh

message routing

b) 您可以通过单击部署详情，查看部署过程。

如果您对当前实例内的资源做了修改，可以单击实例部署再次进行部署。

#### 11. ( 可选 ) 重置实例。



说明：

- 由于该操作会将网关从工作状态切换为初始状态所以请谨慎操作。
- 当您实例中的网关处于离线状态时，您的部署请求将被托管，直到网关上线后平台将主动通知网关进行资源重置操作。

a) 在实例详情页面，单击重置，重置已部署至网关中的资源（例如，将下发的规则计算、函数计算、消息路由等资源做清理）。

边缘分组 > 分组详情

myhome 重置中

最近修改: 2018-08-31 17:45:47

分组标签:

网关 设备 规则计

### 分组网关

设备名称

mygw

### 部署详情

动作内容

DeviceScript.Ligh

DeviceScript.Ligh

DeviceScript.Ligh

message routing

b) 您可以单击部署详情，查看重置进展。