



物联网边缘计算 用户指南

文档版本: 20220429



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。

- 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。
- 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
⚠ 危险	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。	♪ 危险 重置操作将丢失用户配置数据。
▲ 警告	该类警示信息可能会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 时间约十分钟。
〔〕) 注意	用于警示信息、补充说明等,是用户必须 了解的内容。	大意 权重设置为0,该服务器不会再接受新 请求。
? 说明	用于补充说明、最佳实践、窍门等 <i>,</i> 不是 用户必须了解的内容。	⑦ 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在 结果确认 页面,单击 确定 。
Courier字体	命令或代码。	执行 cd /d C:/window 命令,进入 Windows系统文件夹。
斜体	表示参数、变量。	bae log listinstanceid
[] 或者 [alb]	表示可选项,至多选择一个。	ipconfig [-all -t]
{} 或者 {a b}	表示必选项,至多选择一个。	switch {act ive st and}

目录

1.环境搭建	80
1.1. 专业版环境搭建	08
1.1.1. 基于Ubuntu 16.04搭建环境	08
1.1.2. 基于Windows搭建环境	16
1.2. 标准版环境搭建	26
1.2.1. 基于Ubuntu 16.04搭建环境	26
1.2.2. 基于树莓派搭建环境	37
1.2.3. 基于阿里云Cloud Shell快速体验Link IoT Edge	48
1.3. 轻量版环境搭建	56
1.3.1. 基于Ubuntu 16.04搭建环境	56
2.设备接入	65
2.1. 设备接入简介	65
2.2. 示例驱动	67
2.3. 官方驱动	75
2.3.1. Modbus驱动	75
2.3.2. Modbus调试工具	86
2.3.3. OPC UA驱动	91
2.3.4. WebSocket驱动	98
2.3.5. LinkDriver驱动	103
2.4. 驱动开发	107
2.4.1. 概览	107
2.4.2. 驱动编码	108
2.4.3. 驱动调试	113
2.4.4. 常见问题	129
2.5. 驱动发布	129
2.5.1. 云端发布	129

2.5.2. 内置发布	134
2.6. 已购驱动	143
3.场景联动	147
3.1. 什么是场景联动	147
3.2. 云端场景联动	148
3.3. 边缘场景联动	152
3.4. 告警中心	156
4.边缘应用	159
4.1. 什么是边缘应用	159
4.2. 新增自研应用	161
4.2.1. 函数计算应用	161
4.2.2. 容器镜像应用	165
4.2.3. 本地应用	168
4.3. 分配边缘应用到边缘实例	171
4.4. 应用场景	172
4.4.1. 设备上云数据筛选	172
4.4.2. 阿里云OSS服务访问	177
4.4.3. 本地过程控制	182
4.4.4. 对接InfluxDB	185
4.4.5. 设备数据存入本地MySQL数据库	192
4.4.6. 机器学习边缘推理	197
4.5. 小程序	207
4.5.1. 网络时间同步助手	207
4.5.2. 本地数据库存储助手	210
5.断网续传	214
6.双机热备	219
7.流数据分析	222
7.1. 什么是边缘流数据分析	222

7.2. 分配流数据分析到边缘实例	222
7.3. 边缘端流数据分析SQL	225
7.3.1. 概述	225
7.3.2. 创建源表	226
7.3.3. 创建结果表	229
7.3.4. 创建维表	232
7.4. 使用示例	233
7.4.1. 概述	233
7.4.2. 高温报警	233
7.4.3. 温度平均值计算	238
7.4.4. 连续高温报警	248
7.4.5. 订单结算	252
7.4.6. 人流量展示	260
8.消息路由	268
8.1. 什么是消息路由	268
8.2. 设置消息路由	269
8.3. 消息路由设置示例	271
9.云资源访问	280
10.监控信息	282
11.远程运维管理	288
11.1. 远程服务访问	288
11.2. 远程访问助手	295
11.2.1. 远程访问助手概览	295
11.2.2. 软件安装	295
11.2.3. 使用方法	302
12.运维工具	307
12.1. 设备模拟器	307
12.2. 设备批量处理工具	319

12.3. 管理运维工具lectl	320
13.边缘网关控制台	336
13.1. 概述	336
13.2. 登录边缘网关控制台	336
13.3. 网关管理	338
13.4. 驱动管理	339
13.5. 设备管理	340
13.6. TSL管理	341
13.7. OTA管理	342
13.8. 场景联动	343
13.9. 日志管理	344
13.10. 账户管理	345
14.边缘安全服务	347
15.日志服务	350
16.OTA升级	352
17.OTA升级SDK	353
18.产线烧录	354
18.1. 产线烧录概览	354
18.2. 一机一密产线烧录	354
18.3. 一型一密产线烧录	355

1.环境搭建

1.1. 专业版环境搭建

1.1.1. 基于Ubuntu 16.04搭建环境

本文介绍如何在Ubuntu 16.04的系统中搭建Link IoT Edge专业版(LE Pro)的Docker运行环境,实现网关与 云端连接的步骤。

专业版(LE Pro)规格的详细说明请参见产品规格。

准备工作

LE Pro版需要您提前安装好Docker环境,请参见Docker官方文档安装使用您Ubuntu 16.04系统的Docker客户端。要求Docker版本大于V17.03。

创建边缘实例和网关

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击边缘实例。
- 3. 创建一个边缘实例。
 - i. 单击新增实例,在弹出对话框中设置实例名称。

⑦ 说明 实例名称支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长度不 超过20个字符,1个汉字算2个字符。

ii. 根据所搭建的环境,选择对应的Link loT Edge产品规格。详细介绍,请参见产品规格。

物联网边缘计算支持自动分配网关到边缘实例,您可以无需设置其它新增实例相关参数,直接跳转 到步骤4。物联网边缘计算自动为您的边缘实例创建名为LEGatewayAuto的网关产品,并在该产品 下添加一个随机命名的网关设备。

? 说明

- 物联网边缘计算自动为您创建LEGatewayAuto产品时,若您已有该名称的产品,则会 在该名称后添加随机后缀,作为自动创建的网关产品。
- 物联网边缘计算已为您自动创建过一次LEGatewayAuto产品后,在您后续新建边缘实例并自动分配网关时,会在LEGatewayAuto产品下新增网关设备,分配到您新的边缘实例中。

iii. (可选)单击标签信息下的新增标签,可以设置实例标签。通过标签您可以更加有效地归类及识别实例。您也可以不设置标签。

若设置实例标签,请填写标签key和标签value。参数说明

参数	描述
标签key	不可为空,仅支持英文字母大小写,长度不超过20个字符,同一个边缘实例不可 以重复定义标签key。
标签value	不可为空,支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长 度不超过20个字符,1个中文汉字算2个字符。

- 4. (可选)若您想要手动创建网关产品和设备并分配到边缘实例,请按如下步骤操作。
 - i. 在新增实例对话框, 单击高级选项。
 - ii. 在**网关产品**下单击新建网关产品,为实例创建网关。

物联网边缘计算中的网关承载边缘计算能力,每个实例必须分配一个网关设备,并且该网关设备同一时间只能被分配到一个边缘实例。

新增实例		×
* 实例名称		
LinkloTEdge_Node		
产品规格 🕜		
100		\sim
标签信息		
+新增标签		
高级选项		
* 网关产品		
请选择网关产品		\sim
新建网关产品		
	确定	取消

iii. 在**创建产品**对话框中,设置网关产品参数,然后单击完成。

物联网边缘计算中的**新建网关产品**,继承了物联网平台**设备管理 > 产品**中的产品功能,此处已自动为您简化了创建产品的步骤,以便您更快速地创建适合物联网边缘计算中使用的网关产品。

创建产品	×
产品信息	
* 产品名称	
LinkloTEdge_Gateway	
* 所属品类 💿	
○ 标准品类 ◎ 自定义品类	
更多信息	
∨产品描述	
使用文档	完成取消

参数说明

参数	说明
产品名称	为网关产品设置名称,用于后续的查询及识别网关产品。支持中文、英文字母大 小写、数字和下划线(_),长度限制4~30个字符,一个中文汉字算2位。
所属品类	选择品类,为该产品定义物模型。 可选择的值为: •标准品类:选择任一物联网平台预定义的品类,快速完成产品的功能定义。 选择产品模板后,您可以在该模板基础上,编辑、修改、新增功能。 •自定义品类:需根据实际需要,定义产品功能。 若您需要的网关没有特殊功能定义,建议您选择自定义品类。
产品描述	可输入文字,用来描述产品信息。字数限制为100个字符。可以为空。

产品创建成功后,自动跳转回**新增实例**对话框,并且在**网关产品**参数下自动分配刚刚创建的网关 产品。 iv. 在新增实例对话框,单击网关设备下的新建网关设备,为网关产品添加设备。 物联网边缘计算中的新建网关设备功能,继承物联网平台设备管理 > 设备的功能。

新增实例	×
* 实例名称	
LinkloTEdge_Node	
产品规格 🕜	
10.0	\sim
标签信息 +新增标签	
高级选项 へ	
* 网关产品	
LinkloTEdge_Gateway	\sim
新建网关产品	
* 网关设备	
请选择网关设备	\sim
新建网关设备	
	确定取消

v. 根据界面提示设置参数后, 单击确认。

参数说明

参数	描述
产品	系统已自动关联上一步创建的网关产品。
	为该网关设备命名。设备名称需保持产品内唯一。如不填写,系统将自动生成。
设备名称	 ⑦ 说明 设备名称长度为4~32个字符,可包含英文字母、数字和特殊字符,包括短划线(-)、下划线(_)、at(@)、英文句号(.)和英文冒号(:)。

5. 实例参数设置完成后,单击确定,至此您已创建边缘实例和网关。

安装并启动Link IoT Edge

根据本文上方**创建边缘实例和网关**内容,创建完成边缘实例并分配网关后,您需要在网关上安装并启动Link loT Edge。

1. 在边缘实例页面,单击实例名称右侧的软件安装。

边缘	边缘实例										
新增到	印刷	授权	请输入实例名称	Q							C
实例名	品称				网关状态		实例状态	创建时间	操作		
Linklo	TEdge_Nod	le			 在线 		● 未部署	2019-12-18 23:10:43	査看	删除软件安装	

2. 根据环境设置软件安装相关参数,然后单击生成安装命令。

软件安装		×
* 边缘网关 CPU 架构 👔		
x86-64		\sim
产品规悟 👔		
专业版		
* 边缘版本 🕜		
2.4.0		\sim
*操作系统		
Linux		\sim
了解软件安装?	生成安装命令	取消

参数说明

参数	描述
边缘网关CPU架构	您的设备系统对应的CPU架构。此处选择x86-64。
产品规格	在创建边缘实例时,已选择实例中使用的Link IoT Edge版本。此处不可操作。
边缘版本	选择Link IoT Edge的 <mark>发布版本</mark> 。
操作系统	选择您的设备对应的操作系统。此处选择Linux。

3. 复制软件包下载命令备用。

软件安装 ×				
✔ 请确保边缘设备上已经安装Docker,之后请复制命令并在边缘设备控制台上 以root权限执行命令。				
边缘实例 CPU 架构	x86-64			
产品规格	专业版			
边缘版本	2.4.0			
安装网关	gateway_01			
操作系统	Linux的Docker			
运行所需 RAM	≥ 2 GB			
安装存储空间需求	≥ 2 GB			
软件包下载命令	复制命令			
没有合适的硬件载体?	完成重新生成			

- 4. 登录您的Ubuntu系统机器。
- 5. 任意目录下执行步骤3中已复制的命令。

该命令实现一键下载、配置并启动Link IoT Edge。命令执行完成后,会在当前目录中下载*link-iot-edge. sh*脚本。

- 如果第一次启动Link loT Edge,则需要完成如下交互式配置,您可以直接按Enter键使用默认配置。
 - 确认启动版本。
 - 确认函数计算的runtime, 函数默认运行在独立的进程中。
 - 确认是否启动流式计算,默认开启流式计算。
 - 确认是否卸载之前已安装的版本,默认卸载。

拉取Docker镜像完成并启动可能需要等待5~10分钟,启动完成后通过**docker ps**命令查看相关 Docker容器是否已启动,若系统显示如下图所示信息,表示启动成功。



○ 如果不是第一次安装启动Link IoT Edge, 可使用已下载的*link-iot-edge.sh*脚本, 对Link IoT Edge进行 重启、停止、获取状态、修改配置参数等操作, 命令详情请见下图。

6. 返回边缘计算控制台,在左侧导航栏选择边缘实例,找到已创建好的边缘实例,单击右侧操作栏中的查看,进入实例详情页面,选择网关查看网关状态。

打开远程访问按钮,可对网关进行远程管理,详细操作步骤,请参见远程服务访问。

网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置		
苔换网 关	软件安装	Ē								
网关名称		产品名	称			远程访问		状态	最后上线时间	操作
gateway	.01	Linklo	FEdge_Gateway					● 在线	2020-01-17 16:28:00	查看 分享远程SSH终端 远程SSH终端 更多远程服务

- 7. (可选)在**实例详情**页面,查看CPU使用率、内存使用率、存储使用率以及实例进程需要授权访问阿里云云监控(CloudMonitor)服务。
 - i. 添加角色或分配已有的角色,并确保该角色拥有**管理云监控(CloudMonitor)的权限**。详细操作,请参见云资源访问。

ii. 选择**监控信息**页签,打开页面右侧的**云监控状态**开关,如下图所示。



云监控状态打开后,可在**监控信息**页签,查看网关的各类监控信息。

8. (可选) Link IoT Edge支持将边缘实例授权给其他阿里云账号操作。

在边缘实例页面,单击左上角授权,在弹出对话框中单击新增授权,根据界面提示设置参数。

边缘实例					
新增实例 授权 请输入实例名称	Q				C
实例名称	网关状态	定场代本	创建时间	操作	
performent.	新増授权	×	2019-12-18 23:10:43	查看 删除 软件安装	
and approximation of the second se	* 授权用户 UID 请输入授权用户 UID	Ţţ	2019-12-13 10:47:13	查看 删除 软件安装	
and the local of	* 授权实例	功	2019-12-13 10:37:35	查看 删除 软件安装	
participation (请选择授权实例	~ 功	2019-12-11 17:29:03	查看 删除 软件安装	
1010.01		确认 取消	2019-12-04 15:02:01	查看 删除 软件安装	
Taxan .	 	● 部署成功	2019-11-13 15:38:53	<u> </u> 五君	

参数说明

参数	描述
授权用户UID	获取被授权用户的阿里云账号ID,填入此处。
授权实例	选择允许该阿里云账号ID的用户进行远程运维操作的边缘实例。

Link IoT Edge的其它操作

• 重新配置Link IoT Edge。

使用如下命令,对当前已安装的Link IoT Edge版本配置进行修改。

./link-iot-edge.sh --reconfig {Version}

其中, {Version}替换为目标版本号,例如目标版本号为V1.8.2,则实际命令为 ./link-iot-edge.sh --re config v1.8.2 。

• 停止Link IoT Edge。

使用如下命令,可以停止所有Link IoT Edge运行的容器,但是不会删除。

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

● 重新启动Link IoT Edge。

在容器已存在且没有运行的状态下,执行如下命令可重新启动Link loT Edge。

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

其中, {Version}替换为目标版本号,例如目标版本号为V1.8.2,则实际命令为 ./link-iot-edge.sh --re start v1.8.2 。

• 清理Link IoT Edge。

执行如下命令,可停止当前运行的Link loT Edge相关容器,并会删除所有已安装的相关镜像,删除相关数据卷以及启动配置文件。

./link-iot-edge.sh --clean

• 提取Link IoT Edge的日志。

执行如下命令,可打包Link IoT Edge的所有日志,并拷贝到当前目录。

./link-iot-edge.sh --packagelog

下一步

环境搭建完成后,您可以根据设备接入内容,把您的设备接入到物联网边缘计算。同时也可以为边缘实例分 配其他资源(如函数计算、消息路由等)管理您的设备。

接入设备或分配其他资源到边缘实例后,需要根据如下步骤部署边缘实例。

- 1. 在实例详情页面,单击右上角部署后,在弹出对话框中单击确定,部署边缘实例。
- 当部署状态显示为部署成功,表示部署实例完成。您可以单击查看日志,查看部署详情。您也可以通 过单击**实例详情**页面右上角的查看部署日志,查看历史部署情况。

边缘计算 / 边缘实例 / 实例详情				
← LinkloTEdge_Nod	e 部署成功			查看部署日志重置部署日志
产品現格 专业版 CPU使用率 2.58 % 查看 网关 监控信息 设备与驱动	部署进程 ♥ gateway_01 网关部署完成	×		
昔庚阿关 软件安装 阿关名称 产品名	部署进程 • 部署开始	2020-01-17 16:38:30	最后上线时间	操作
gateway_01 LinkloTi	- m-a zeau - 部署打包 - 部署分发	 ✓ 元成 ◇ 完成 ◇ 完成 	2020-01-17 16:28:00	查看 分享远程SSH终端 远程SSH终端 更多远程服务
	· 節審結束	2020-01-17 16:38:31 查看日志 关闭		

1.1.2. 基于Windows搭建环境

本文介绍如何在Windows7和Windows10的系统中搭建Link IoT Edge专业版(LE Pro)的Docker运行环境, 实现网关与云端连接的步骤。

专业版(LE Pro)规格的详细说明请参见产品规格。

准备工作

- LE Pro版需要您提前安装好Docker环境, Windows7系统请参见Inst all Docker Toolbox on Windows, Windows10系统请参见Inst all Docker for Windows。要求Docker版本大于v17.03。
- 在您的Windows 7或Windows 10系统机器上安装最新版本的Git bash。安装包以及安装操作请参见官方文档。

约束条件

- 目前仅支持在Windows 7和Windows 10系统中搭建专业版运行环境。
- LE Pro版需要bash的运行环境,用于运行Link IoT Edge的脚本工具,请务必安装好Git bash。

创建边缘实例和网关

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击边缘实例。
- 3. 创建一个边缘实例。
 - i. 单击新增实例,在弹出对话框中设置实例名称。

⑦ 说明 实例名称支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长度不 超过20个字符,1个汉字算2个字符。

ii. 根据所搭建的环境,选择对应的Link loT Edge产品规格。详细介绍,请参见产品规格。

物联网边缘计算支持自动分配网关到边缘实例,您可以无需设置其它新增实例相关参数,直接跳转 到步骤4。物联网边缘计算自动为您的边缘实例创建名为LEGatewayAuto的网关产品,并在该产品 下添加一个随机命名的网关设备。

? 说明

- 物联网边缘计算自动为您创建LEGatewayAuto产品时,若您已有该名称的产品,则会 在该名称后添加随机后缀,作为自动创建的网关产品。
- 物联网边缘计算已为您自动创建过一次LEGatewayAuto产品后,在您后续新建边缘实例并自动分配网关时,会在LEGatewayAuto产品下新增网关设备,分配到您新的边缘实例中。
- iii. (可选)单击标签信息下的新增标签,可以设置实例标签。通过标签您可以更加有效地归类及识 别实例。您也可以不设置标签。

若设置实例标签,请填写标签key和标签value。参数说明

参数	描述
标签key	不可为空,仅支持英文字母大小写,长度不超过20个字符,同一个边缘实例不可 以重复定义标签key。
标签value	不可为空,支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长 度不超过20个字符,1个中文汉字算2个字符。

4. (可选)若您想要手动创建网关产品和设备并分配到边缘实例,请按如下步骤操作。

i. 在新增实例对话框, 单击高级选项。

ii. 在**网关产品**下单击新建网关产品,为实例创建网关。

物联网边缘计算中的网关承载边缘计算能力,每个实例必须分配一个网关设备,并且该网关设备同一时间只能被分配到一个边缘实例。

新增实例		\times
* 实例名称		
LinkloTEdge_Node		
产品规格 🕜		
1.01		\sim
标签信息 +新增标签		
高级选项へ		
* 网关产品		
请选择网关产品		\sim
新建网关产品		
	确定	取消

iii. 在**创建产品**对话框中,设置网关产品参数,然后单击完成。

物联网边缘计算中的**新建网关产品**,继承了物联网平台**设备管理 > 产品**中的产品功能,此处已自动为您简化了创建产品的步骤,以便您更快速地创建适合物联网边缘计算中使用的网关产品。

创建产品	\times
产品信息	
* 产品名称	
LinkloTEdge_Gateway	
* 所属品类 💿	
更多信息	
∨产品描述	
使用文档 完成 取消	á

参数说明

参数	说明
产品名称	为网关产品设置名称,用于后续的查询及识别网关产品。支持中文、英文字母大 小写、数字和下划线(_),长度限制4~30个字符,一个中文汉字算2位。
所属品类	选择品类,为该产品定义物模型。 可选择的值为: • 标准品类:选择任一物联网平台预定义的品类,快速完成产品的功能定义。 选择产品模板后,您可以在该模板基础上,编辑、修改、新增功能。 • 自定义品类:需根据实际需要,定义产品功能。 若您需要的网关没有特殊功能定义,建议您选择自定义品类。
产品描述	可输入文字,用来描述产品信息。字数限制为100个字符。可以为空。

产品创建成功后,自动跳转回新增实例对话框,并且在网关产品参数下自动分配刚刚创建的网关 产品。 iv. 在新增实例对话框,单击网关设备下的新建网关设备,为网关产品添加设备。 物联网边缘计算中的新建网关设备功能,继承物联网平台设备管理 > 设备的功能。

新增实例	×
* 实例名称	
LinkloTEdge_Node	
产品规格 ②	
10.0	\sim
标签信息 +新增标签	
高级选项 へ	
* 网关产品	
LinkloTEdge_Gateway	\sim
新建网关产品	
* 网关设备	
请选择网关设备	\sim
新建网关设备	
	确定 取消

v. 根据界面提示设置参数后, 单击确认。

参数说明

参数	描述
产品	系统已自动关联上一步创建的网关产品。
设备名称	为该网关设备命名。设备名称需保持产品内唯一。如不填写,系统将自动生成。
	 ⑦ 说明 设备名称长度为4~32个字符,可包含英文字母、数字和特殊字符,包括短划线(-)、下划线(_)、at(@)、英文句号(.)和英文冒号(:)。

5. 实例参数设置完成后,单击确定,至此您已创建边缘实例和网关。

安装并启动Link IoT Edge

根据本文上方**创建边缘实例和网关**内容,创建完成边缘实例并分配网关后,您需要在网关上安装并启动Link loT Edge。

1. 在边缘实例页面,单击实例名称右侧的软件安装。

边缘实例				
新增实例 授权 请输入实例名称	Q			C
实例名称	网关状态	实例状态	创建时间	操作
LinkloTEdge_Node	 ● 在线 	● 未部署	2019-12-18 23:10:43	查看 删除 软件安装

2. 根据环境设置软件安装相关参数,然后单击生成安装命令。

软件安装		×
* 边缘网关 CPU 架构 💿		
x86-64		\sim
产品规格 💿 专业版		
* 边缘版本 💿		
2.4.0		\sim
* 操作系统		
Windows		~
了解软件安装?	生成安装命令	取消

参数说明

参数	描述
边缘网关CPU架构	您的设备系统对应的CPU架构。此处选择x86-64。
产品规格	在创建边缘实例时,已选择实例中使用的Link loT Edge版本。此处不可操作。
边缘版本	选择Link IoT Edge的发布版本。
操作系统	选择您的设备对应的操作系统。此处选择Windows。

3. 复制软件包下载命令备用。

软件安装	软件安装 ×					
✔ 请确保边缘设备上已经安装Docker,之后请复制命令并在边缘设备控制台上 执行命令。						
边缘实例 CPU 架构	x86-64					
产品规格	专业版					
边缘版本	2.4.0					
安装网关	gateway_01					
操作系统	Windows的Docker					
运行所需 RAM	≥ 2 GB					
安装存储空间需求	≥ 2 GB					
软件包下载命令	复制命令					
没有合适的硬件载体?	完成重新生成					

- 4. 登录您的Windows7或Windows10系统的机器。
- 5. 在git bash运行环境任意目录下执行步骤3中已复制的命令。

HINGKOT
\$ curl -0 http://link-iot-edge-packet.oss-cn-shanghai.aliyuncs.com/config/link-i
pt-edge.sh && chmod +x link-iot-edge.sh && ./link-iot-edge.sh v2.4.1
docker windows
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 22397 100 22397 0 0 107k 0::: 107k
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2158 100 2158 0 0 19798 0
Please confirm the version of Linkedge $[v2.4.1]$ you want to start $(y/n, default:$
y): y
Supported Function Compute runtimes are:
1 Docker-Runtime-All (Install all the supported runtimes below)
2 Docker-Runtime-Python3
3 Docker-Runtime-Nodejs8
4 Docker-Runtime-C
Each runtime image costs about 200MB disk space.
Which runtime do you want to install? (Please input the number and use ' ' to sp
lit, default: 1):
Do you want to install Stream Compute (need 1GB Disk Space) (y/n, default: y): n
Do you want to uinstall previously installed Linkedge (y/n, default: y): y
Removing linkedge
Removing stream compute edge
Removing fcedge daemon
Removing FCedge runtimes
Installing linkedge
Installing registry.cn-hangzhou.aliyuncs.com/iotedge/edge x86 alpine:v2.4.1
image already exists
Installing registry.cn-hangzhou.aliyuncs.com/iotedge/runtime-python3:v1.0.0
v1.0.0: Pulling from iotedge/runtime-python3
4fe2ade4980c: Already exists
139114e5b0b6: Already exists
67d08cc0d008: Already exists

该命令实现一键下载、配置并启动Link IoT Edge。命令执行完成后,会在当前目录中下载*link-iot-edge.sh*脚本。

- 如果您第一次启动网关,则需要完成如下交互式配置,您可以直接按Enter键使用默认配置。
 - 确认启动版本。
 - 确认函数计算的runtime, 函数默认运行在独立的进程中。
 - 确认是否启动流式计算,默认开启流式计算。
 - 确认是否卸载之前已安装的版本,默认卸载。

拉取Docker镜像完成并启动可能需要等待5~10分钟,启动完成后通过**docker ps**命令查看相关 Docker容器是否已启动,若系统显示如下图所示信息,表示启动成功。

Status: Downloaded Starting linkedg Initializing lin d30	newer ima e kedge	uge for regis	try.cn-hangzhou.aliyun	cs.com/iotedge/edge_x86_ ab	_sc:v1.8.3	
Start linkedge s	uccess !	MTNGW64 ~				
\$ docker ps						
CONTAINER ID	IMAGE				COMMAND	CREATED
STATUS		PORTS	NAMES			
30de3eb67f13	registry	.cn-hangzhou	.aliyuncs.com/iotedge/	edge_x86_sc:v1.8.3	"/linkedge/stream-co…"	About a minu
te ago Up About a	minute		edge-stream-c	ompute		
d309cba1525c	registry	.cn-hangzhoι	i.aliyuncs.com/iotedge/	edge_x86_alpine:v1.8.3	"/linkedge/gateway/b"	About a minu
te ago Up About a	minute		linkedge			

• 如果不是第一次安装启动Link IoT Edge,可使用已下载的*link-iot-edge.sh*脚本,对Link IoT Edge进行重启、停止、获取状态、修改配置参数等操作,命令详情请见下图。

Start the linkedge container with startup parameters USAGE1: ./link-iot-edge.sh Version ProductKey DeviceName DeviceSecret
Restart the linkedge container USAGE2: ./link-iot-edge.shrestart Version
Stop the linkedge USAGE3: ./link-iot-edge.shstop
Get the linkedge log USAGE4: ./link-iot-edge.shpackagelog
Clean all the linkedge configuration USAGE5: ./link-iot-edge.shclean
Clean configuration and then configure linkedge again USAGE5: ./link-iot-edge.shreconfig Version

6. 返回边缘计算控制台,在左侧导航栏选择边缘实例,找到已创建好的边缘实例,单击右侧操作栏中的查看,进入实例详情页面,选择网关查看网关状态。

打开远程访问按钮,可对网关进行远程管理,详细操作步骤,请参见远程服务访问。

网关 监控信息	设备与驱动 场景联动	边缘应用	流数据分析	消息路由	日志服务	设置		
替换网 关 软件安装								
网关名称	产品名称			远程访问		状态	最后上线时间	操作
gateway_01	LinkloTEdge_Gateway					● 在线	2020-01-17 16:28:00	查看 分享远程SSH终端 远程SSH终端 更多远程服务

- 7. (可选)在**实例详情**页面,查看CPU使用率、内存使用率、存储使用率以及实例进程需要授权访问阿里 云云监控(CloudMonitor)服务。
 - i. 添加角色或分配已有的角色,并确保该角色拥有**管理云监控(CloudMonitor)的权限**。详细操作,请参见云资源访问。
 - ii. 选择监控信息页签,打开页面右侧的云监控状态开关,如下图所示。

网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置	
() 注意	: 新増资源监控I	功能 (场景联动、副	酸计算、驱动、涉	充数据分析等) 霊	要依赖v2.1.0及以	上版本,否则无法	5准确反映运行状态	5, 请确保;	升级到最新版本。
监控信息									◎ 云监控状态 ○
主机监	空 场景联初	边缘应用	驱动监控	流数据监	空				
报警配置	最近1小	49 V							G

云监控状态打开后,可在**监控信息**页签,查看网关的各类监控信息。

- (可选)在实例详情页面单击网关页签,打开远程访问开关后,可对网关进行远程管理。详细操作步骤,请参见远程服务访问。
- (可选) Link IoT Edge支持将边缘实例授权给其他阿里云账号操作。
 在边缘实例页面,单击左上角授权,在弹出对话框中单击新增授权,根据界面提示设置参数。

边缘实例					
新增实例 授权 请输入实例名称	Q				G
实例名称	网关状态	定肠状态	创建时间	操作	
	新增授权	×	2019-12-18 23:10:43	查看 删除 软件安装	
and top (and	* 授权用户 UID 请输入授权用户 UID	7 7 3	2019-12-13 10:47:13	查看 删除 软件安装	
conclusion of the second	* 授权实例	功	2019-12-13 10:37:35	查看 删除 软件安装	
conciling team	请选择授权实例	~ 功	2019-12-11 17:29:03	查看 删除 软件安装	
10		确认 取消	2019-12-04 15:02:01	查看 删除 软件安装	
Constant (● 高线	● 部署成功	2019-11-13 15:38:53	查看 删除 软件安装	

参数说明

参数	描述
授权用户UID	获取被授权用户的阿里云账号ID,填入此处。
授权实例	选择允许该阿里云账号ID的用户进行远程运维操作的边缘实例。

Link IoT Edge的其它操作

• 重新配置Link IoT Edge。

使用如下命令,对当前已安装的Link loT Edge版本配置进行修改。

./link-iot-edge.sh --reconfig {Version}

其中, {Version}替换为目标版本号,例如目标版本号为V1.8.2,则实际命令为 ./link-iot-edge.sh --re config v1.8.2 。

• 停止Link IoT Edge。

使用如下命令,可以停止所有Link IoT Edge运行的容器,但是不会删除。

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

• 重新启动Link IoT Edge。

在容器已存在且没有运行的状态下,执行如下命令可重新启动Link loT Edge。

./link-iot-edge.sh --restart {Version}

其中, {Version}替换为目标版本号,例如目标版本号为V1.8.2,则实际命令为 ./link-iot-edge.sh --re start v1.8.2 。

• 清理Link IoT Edge。

执行如下命令,可停止当前运行的Link loT Edge相关容器 ,并会删除所有已安装的相关镜像 , 删除相关数 据卷以及启动配置文件。

./link-iot-edge.sh --clean

● 提取Link IoT Edge的日志。

执行如下命令,可打包Link IoT Edge的所有日志,并拷贝到当前目录。

./link-iot-edge.sh --packagelog

下一步

环境搭建完成后,您可以根据设备接入内容,把您的设备接入到物联网边缘计算。同时也可以为边缘实例分 配其他资源(如函数计算、消息路由等)管理您的设备。

接入设备或分配其他资源到边缘实例后,需要根据如下步骤部署边缘实例。

- 1. 在实例详情页面,单击右上角部署后,在弹出对话框中单击确定,部署边缘实例。
- 当部署状态显示为部署成功,表示部署实例完成。您可以单击查看日志,查看部署详情。您也可以通 过单击实例详情页面右上角的查看部署日志,查看历史部署情况。

边缘计算 / 边缘实例 / 实例详情				
← LinkloTEdge_Nod	e ^{部署成功}			查看部署日志 重置 部署
产品源格 专业版 CPU 使用率 2.58 % 畫着 网关 监控信息 设备与驱动	部署进程 ✔ gateway_01 网关部署完成	×	5	
苔换网关 软件安装 网关名称 产品名	部署进程 - 部署开始	2020-01-17 16:38:30	最后上线时间	操作
gateway_01 LinkloTi	· 即著次的	● 元加	2020-01-17 16:28:00	查看 分享远程SSH终端 远程SSH终端 更多远程服务
	 ・部署分发 ・部署结束 	交 完成		
		査石日志 关闭		

1.2. 标准版环境搭建

1.2.1. 基于Ubuntu 16.04搭建环境

本文介绍基于Ubuntu 16.04系统,搭建Link IoT Edge标准版(LE Standard)运行环境的方法。

Link IoT Edge标准版软件包支持在Ubuntu 16.04 ~ Ubuntu 18.04系统上运行,并在下列平台上进行测试和验证。

架构	操作系统
x86_64	Ubuntu 16.04 ~ Ubuntu 18.04
ARMv7	Ubuntu 16.04 ~ Ubuntu 18.04
ARMv8 (AArch64)	Ubuntu 16.04 ~ Ubuntu 18.04

尽管Link IoT Edge可以在其它版本的Ubuntu操作系统上运行,但为了最佳的稳定性和安全性,建议您选择在 官方支持的系统版本上运行。接下来,我们将基于x86_64 Ubuntu 16.04的平台上为您介绍Link IoT Edge标 准版安装部署的方法。Ubuntu16.04~18.04系统在其它架构上的标准版软件包您可以在下载地址中的"标准 版"内容下获取。

准备工作

准备Ubuntu 16.04系统的PC或者虚拟机,并且该PC或者虚拟机能够正常连接网络。如果您不熟悉Ubuntu 16.04的安装,可参考Ubuntu 16.04安装指南。

环境设置

在x86_64 Ubuntu 16.04机器上需要设置Link IoT Edge运行所依赖的环境。

1. 在x86_64 Ubunt u 16.04机器的本地终端窗口或者SSH终端窗口执行以下命令,下载环境检查工具并运行。

```
wget http://iotedge-web.oss-cn-shanghai.aliyuncs.com/public/testingTool/link-iot-edge_e
nv-check.sh
sudo chmod +x ./link-iot-edge_env-check.sh
sudo ./link-iot-edge_env-check.sh
```

② 说明 *link-iot-edge_env-check.sh*脚本在Link IoT Edge支持的平台需要以root权限运行,同时 Link IoT Edge支持的平台需要支持以下Linux系统命令:

printf、echo、cat、ls、expr、grep、test、uname、set、head、sort、cut、uniq、xargs、if config

2. 按照运行环境检查工具的提示在您的机器上安装所有必需的依赖项,当检查工具成功运行完成后,返回如下图信息,表示Link IoT Edge能够在您的机器上成功运行。

```
Linux Kernel Version:
 for x86_{64}: >= 2.6.32
 for ARMv7: >= 3.2.0
for ARMv8_64: >= 3.7.0
 for ARMv7:
 RAM:
                             >= 128MB
                             >= 128MB
 FLASH:
  CPU Frequency: >= 1GHZ
  Link IoT Edge v1.8 Run Environment Check
Check Depended Commands:
                   Checking for command wget
                ] Checking for command realpath
] Checking for command tar
      yes ] Checking for command unzip
yes ] Checking for command readlink
yes ] Checking for command basename
      yes ] Checking for command basename
yes ] Checking for command dirname
yes ] Checking for command pidof
yes ] Checking for command df
yes ] Checking for command grep
yes ] Checking for command ps
yes ] Checking for command kill
yes ] Checking for command dirname
yes ] Checking for command xargs
      yes ] Checking for command unshare
yes ] Checking for command unshare
yes ] Checking for command awk
               ] Checking for command mkdir
Kernel architecture: x86_64
Kernel version: 4.15
Total memory: 7.7g
Available flash space: 17g
 C library version: 2.27
C library: Ubuntu GLIBC 2.27-3ubuntu1
 Init process: /lib/systemd/systemd
Loopback is: up
 Suggestion:
  l. It looks like the kernel uses 'systemd' as the init process.
  Please using systemd to manage Link IoT Edge service.
```

创建边缘实例和网关

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击边缘实例。
- 3. 创建一个边缘实例。
 - i. 单击新增实例,在弹出对话框中设置实例名称。

⑦ 说明 实例名称支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长度不超过20个字符,1个汉字算2个字符。

ii. 根据所搭建的环境,选择对应的Link loT Edge产品规格。详细介绍,请参见产品规格。

物联网边缘计算支持自动分配网关到边缘实例,您可以无需设置其它新增实例相关参数,直接跳转 到步骤4。物联网边缘计算自动为您的边缘实例创建名为LEGatewayAuto的网关产品,并在该产品 下添加一个随机命名的网关设备。

? 说明

- 物联网边缘计算自动为您创建LEGatewayAuto产品时,若您已有该名称的产品,则会 在该名称后添加随机后缀,作为自动创建的网关产品。
- 物联网边缘计算已为您自动创建过一次LEGatewayAuto产品后,在您后续新建边缘实例并自动分配网关时,会在LEGatewayAuto产品下新增网关设备,分配到您新的边缘实例中。
- iii. (可选)单击标签信息下的新增标签,可以设置实例标签。通过标签您可以更加有效地归类及识 别实例。您也可以不设置标签。

若设置实例标签,请填写标签key和标签value。参数说明

参数	描述
标签key	不可为空,仅支持英文字母大小写,长度不超过20个字符,同一个边缘实例不可 以重复定义标签key。
标签value	不可为空,支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长 度不超过20个字符,1个中文汉字算2个字符。

4. (可选) 若您想要手动创建网关产品和设备并分配到边缘实例,请按如下步骤操作。

i. 在新增实例对话框,单击高级选项。

ii. 在**网关产品**下单击新建网关产品,为实例创建网关。

物联网边缘计算中的网关承载边缘计算能力,每个实例必须分配一个网关设备,并且该网关设备同一时间只能被分配到一个边缘实例。

新增实例		×
* 实例名称		
LinkloTEdge_Node		
产品规格 💿		
100		\sim
标签信息		
+新增标签		
高级选项へ		
* 网关产品		
请选择网关产品		\sim
新建网关产品		
	确定	取消

iii. 在**创建产品**对话框中,设置网关产品参数,然后单击完成。

物联网边缘计算中的**新建网关产品**,继承了物联网平台**设备管理 > 产品**中的产品功能,此处已自动为您简化了创建产品的步骤,以便您更快速地创建适合物联网边缘计算中使用的网关产品。

创建产品	\times
产品信息	
* 产品名称	
LinkloTEdge_Gateway	
* 所属品类 💿	
更多信息	
∨产品描述	
使用文档 完成 取消	á

参数说明

参数	说明
产品名称	为网关产品设置名称,用于后续的查询及识别网关产品。支持中文、英文字母大 小写、数字和下划线(_),长度限制4~30个字符,一个中文汉字算2位。
所属品类	选择品类,为该产品定义物模型。 可选择的值为: • 标准品类:选择任一物联网平台预定义的品类,快速完成产品的功能定义。 选择产品模板后,您可以在该模板基础上,编辑、修改、新增功能。 • 自定义品类:需根据实际需要,定义产品功能。 若您需要的网关没有特殊功能定义,建议您选择自定义品类。
产品描述	可输入文字,用来描述产品信息。字数限制为100个字符。可以为空。

产品创建成功后,自动跳转回**新增实例**对话框,并且在**网关产品**参数下自动分配刚刚创建的网关 产品。 iv. 在新增实例对话框,单击网关设备下的新建网关设备,为网关产品添加设备。 物联网边缘计算中的新建网关设备功能,继承物联网平台设备管理 > 设备的功能。

新增实例	×
* 实例名称	
LinkloTEdge_Node	
产品规格 ②	
10.0	\sim
标签信息 +新增标签	
高级选项 へ	
* 网关产品	
LinkloTEdge_Gateway	\sim
新建网关产品	
* 网关设备	
请选择网关设备	\sim
新建网关设备	
	确定 取消
	确定 取消

v. 根据界面提示设置参数后, 单击确认。

参数说明

参数	描述
产品	系统已自动关联上一步创建的网关产品。
	为该网关设备命名。设备名称需保持产品内唯一。如不填写,系统将自动生成。
设备名称	 ⑦ 说明 设备名称长度为4~32个字符,可包含英文字母、数字和特殊字符,包括短划线(-)、下划线(_)、at(@)、英文句号(.)和英文冒号(:)。

5. 实例参数设置完成后,单击确定,至此您已创建边缘实例和网关。

安装并启动Link IoT Edge

根据本文上方**创建边缘实例和网关**内容,创建完成边缘实例并分配网关后,您需要在网关上安装并启动Link loT Edge。

1. 在边缘实例页面,单击实例名称右侧的软件安装。

边缘实例					
新增实例 授权 请输入实例名称	Q				G
实例名称	网关状态	实例状态	创建时间	操作	
LinkloTEdge_Node	●在线	 未部署 	2019-12-18 23:10:43	查看 删除 软件安装	

2. 根据环境设置下载参数,然后单击生成安装命令。

软件安装			×
	还没有网关?通过免费虚拟 您可免费使用 CloudShell 虚拟 的产品功能。 立即体验	以机快速体验! _{几,快速体验物联网边}	边缘计算
* 边缘网关 CPU	架构 💿		
x86-64			\sim
产品规格 🌚 标准版			
*边缘版本 🕜			
2.4.0			\sim
* 操作系统			
Linux			\sim
了解软件安装?		生成安装命令	取消

参数说明

参数	描述
边缘网关CPU架构	您的设备系统对应的CPU架构。此处选择x86-64。
产品规格	在创建边缘实例时,已选择实例中使用的Link loT Edge版本。此处不可操作。
边缘版本	选择Link loT Edge的发布版本。
操作系统	选择您的设备对应的操作系统。此处选择Linux。

如果您暂时没有合适的硬件载体来运行Link IoT Edge,您可以单击对话框上方的**立即体验**,使用Cloud Shell虚拟机快速体验标准版Link IoT Edge的产品功能。使用**立即体验**功能有诸多注意事项,详情请参见基于阿里云Cloud Shell快速体验Link IoT Edge。

○ 注意 同一个边缘实例只能启动一个Link IoT Edge,若在已经商用的边缘实例(即边缘实例已 在硬件网关上启动了Link IoT Edge)使用**立即体验**功能时,会导致硬件网关上启动的Link IoT Edge 和虚拟机上启动的Link IoT Edge不断互踢,影响您的正常业务。强烈建议您不要在已启动Link IoT Edge的边缘实例上使用**立即体验**功能。

3. 复制操作系统命令备用。

软件安装		×
✔ 请复制命令,并在目标设	备的控制台上以root权限执行命令。	
边缘实例 CPU 架构	x86-64	
产品规格	标准版	
边缘版本	2.4.0	
安装网关	gateway_01	
操作系统	Linux	
运行所需 RAM	≥ 128 MB	
安装存储空间需求	≥ 256 MB	
软件包下载命令	复制命令	
没有合适的硬件载体?	完成重新生命	¢

- 4. 登录您的x86_64 Ubunt u 16.04机器。
- 5. 任意目录下执行步骤3中已复制的命令。

该命令实现一键下载、配置并启动Link IoT Edge。命令执行完成后,会在当前目录中下载*link-iot-edge-standard.sh*脚本。

⑦ 说明 如果不是第一次安装启动Link IoT Edge,可使用已下载的link-iot-edge-standard.sh脚本,对Link IoT Edge进行重启、停止、获取状态、修改配置参数等操作,命令详情请见下图。
 # Intsall the linkedge with parameters
USAGE1: ./link-iot-edge-standard.sh --install Arch Version
 Arch : x86-64, armv7, armv7-hf, aarch64
 Version: v1.8.1,v1.8.2

 # Config the linkedge with startup parameters
USAGE2: ./link-iot-edge-standard.sh --config ProductKey DeviceName DeviceSecret

 # Start the linkedge
 USAGE4: ./link-iot-edge-standard.sh --start

 # Get the linkedge status
 USAGE4: ./link-iot-edge-standard.sh --status

6. 执行如下命令查看Link IoT Edge核心服务的运行状态。

sudo ./link-iot-edge-standard.sh --status

若系统显示如下信息,表示Link loT Edge核心服务启动成功。

				:/\$./link-iot-edge-standard.shstatus
				Link IoT Edge Service Status
Г	+	٦	Service	mbusd conf is active pid is 10066
Ē	+	í	Service	config-manager is active pid is 10093
Ē	+	ĺ	Service	data-manager is active, pid is 10112
Ī	+	j	Service	watch-dog is active, pid is 10145
Ē	+	j	Service	cloud-proxy is active, pid is 10102
[+]	Service	dimu is active, pid is 10179
[+]	Service	message-router is active, pid is 10174
[+]	Service	fota is active, pid is 10195
[+]	Service	gateway-monitor is active, pid is 10213
[+]	Service	task-dispatcher is active, pid is 10208
[+]	Service	credential is active, pid is 10108
	+]	Service	fc-base is active, pid is 10359
	+]	Service	ifttt is active, pid is 10230
	+		Service	redis-server is active, pid is 10061
	+		Service	logger is active, pid is 10069
L	+]	Service	service-monitor is active, pid is 10371
L	+	ļ	Service	RemoteTerminalDaemon is active, pid is 13593
	+		Service	nginx is active, pid is 10043

您也可以在<mark>边缘计算控制台</mark>,左侧导航栏选择**边缘实例**,在已创建好的边缘实例右侧单击查看进入实例详情页面,选择网关,查看网关状态。

网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置		
替换网关	软件安装	E								
网关名称		产品名	称			远程访问		状态	最后上线时间	操作
gateway_01		Linklo	LinkloTEdge_Gateway			D		● 在线	2020-01-17 16:28:00	查看 分享远程SSH终端 远程SSH终端 更多远程服务

打开远程访问按钮,可对网关进行远程管理,详细操作步骤请参见远程服务访问。

- 7. (可选)在**实例详情**页面,查看CPU使用率、内存使用率、存储使用率以及实例进程需要授权访问阿里云云监控(CloudMonitor)服务。
 - i. 添加角色或分配已有的角色,并确保该角色拥有管理云监控(CloudMonitor)的权限。详细操作,请参见云资源访问。
 - ii. 选择监控信息页签, 打开页面右侧的云监控状态开关, 如下图所示。

网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置	
1 注意	: 新増资源监控功	能(场景联动、函		充数据分析等) 雷	需要依赖v2.1.0及以	上版本,否则无法	准确反映运行状态	5, 请确保升	H级到最新版本。
监控信息	<u>监控信息</u> ③ 云监控状态								
主机监持	空 场景联动) 边缘应用	驱动监控	流数据监	控				
报警配置	最近1小	н ~							C

云监控状态打开后,可在**监控信息**页签,查看网关的各类监控信息。

8. (可选) Link IoT Edge支持将边缘实例授权给其他阿里云账号操作。

在边缘实例页面,单击左上角授权,在弹出对话框中单击新增授权,根据界面提示设置参数。

边缘实例					
新增实例 授权 请输入实例名称	Q				G
实例名称	网关状态	定例状态	创建时间	操作	
particular in the second	新增授权	×	2019-12-18 23:10:43	查看 删除 软件安装	
and top (seal	* 授权用户 UID 请输入授权用户 UID	193	2019-12-13 10:47:13	查看 删除 软件安装	
and the local data of the loca	* 授权实例	功	2019-12-13 10:37:35	查看 删除 软件安装	
performer.	请选择授权实例	~ Ђ	2019-12-11 17:29:03	查看 删除 软件安装	
58-1845		确认取消	2019-12-04 15:02:01	查看 删除 软件安装	
Taxabas	• 高线	● 部署成功	2019-11-13 15:38:53	查看 删除 软件安装	

参数说明

参数	描述					
授权用户UID	获取被授权用户的阿里云账号ID,填入此处。					
授权实例	选择允许该阿里云账号ID的用户进行远程运维操作的边缘实例。					

使用systemd管理Link IoT Edge

您可以使用systemd来管理Link loT Edge服务的启动(start)、停止(stop)和查看状态(status)。 Link loT Edge的systemd service如下所示。
[Unit]
Description=Link IoT Edge
[Service]
Type=forking
Restart=on-failure
<pre>ExecStart=/linkedge/gateway/build/script/iot_gateway_start.sh</pre>
<pre>ExecReload=/linkedge/gateway/build/script/iot_gateway_start.sh</pre>
ExecStop=/linkedge/gateway/build/script/iot_gateway_stop.sh
[Install]
WantedBy=multi-user.target

您可以运行如下命令,配置开机自启动功能。

wget http://iotedge-web.oss-cn-shanghai.aliyuncs.com/public/testingTool/LinkIoTEdge.service sudo cp LinkIoTEdge.service /etc/systemd/system/LinkIoTEdge.service sudo systemctl enable LinkIoTEdge.service

其它常用命令如下所示。

- 启动命令: sudo systemctl start LinkIoTEdge.service
- 重启命令: sudo systemctl restart LinkIoTEdge.service
- 停止命令: sudo systemctl stop LinkIoTEdge.service
- 禁用开机自启动: sudo systemctl disable LinkIoTEdge.service

下一步

环境搭建完成后,您可以根据设备接入章节内容,把您的设备接入到物联网边缘计算。同时也可以为边缘实 例分配其他资源(如函数计算、消息路由等)管理您的设备。

接入设备或分配其他资源到边缘实例后,需要根据如下步骤部署边缘实例。

- 1. 在实例详情页面,单击右上角部署后在弹出框中单击确定,部署边缘实例。
- 2. 当部署状态显示为**部署成功**,表示部署实例完成。您可以单击**查看日志**,查看部署详情。您也可以通 过单击**实例详情**页面右上角的**查看部署日志**,查看历史部署情况。

边缘计算 / 边缘实例 / 实例	详情				
← LinkloTEdg	je_Node	部署成功			查看部署日志 重置 部署
 产品规格 标准版 CPU使用率 2.58% 查報 网关 监控信息 返 	备与驱动	部署进程 ♥ gateway_01 网关部署完成		×	
苔換网关 软件安装 网关名称	产品名利	部署进程 • 部署开始	2020-01-17 16:38:30	最后上线时间	操作
gateway_01	LinkloTl	 部審技配 部署打包 - 部署分分 	✓ 元成 ◇ 元成	2020-01-17 16:28:00	查看 分享远程SSH终端 远程SSH终端 更多远程服务
		- 部署结束	2020-01-17 16:38:31		
			查看日志 关闭		

1.2.2. 基于树莓派搭建环境

本文介绍基于树莓派(Raspberry Pi),搭建Link IoT Edge标准版(LE Standard)运行环境的方法。

Link IoT Edge标准版软件包支持在树莓派上运行,并在下面平台上进行了测试和验证。

硬件	操作系统
树莓派3B	Raspbian

理论上Link IoT Edge也可以在其它版本的树莓派(2B、3B+、4B版本的树莓派)上运行,但为了最佳的稳定 性和安全性,建议您选择在官方支持的系统版本上运行。

准备工作

准备符合如下表格中要求的树莓派设备。

项目	规格
树莓派	树莓派3B
操作系统	Raspbian Stretch with desktop
内核	≥4.13
Micro SD卡	≥8GB
外设	HDMI线、键盘、鼠标、MicroUSB线

环境设置

在树莓派设备上需要设置Link IoT Edge运行所依赖的环境,如果您的树莓派设备上已经安装好Raspbian,请 跳过步骤1至步骤3。

- 1. 下载Raspbian镜像和SD镜像烧写工具。
 - 操作系统镜像: Raspbian Stretch with desktop
 - SD卡镜像烧写工具: Et cher

⑦ 说明 下载适合您计算机操作系统的SD卡镜像烧写工具。

2. 使用Etcher将Raspbian镜像烧写到SD卡。

将Micro SD卡插入读卡器,并接入到您的计算机。请参考树莓派操作系统镜像安装指南将Raspbian镜像 烧写到Micro SD卡。

3. 启动并配置Raspbian操作系统。

将Micro SD卡插入您的树莓派设备上,连接好鼠标和键盘,通过HDMI线连接到监视器上,使用MicroUSB 线进行供电,启动您的树莓派设备。

- i. 通过WiFi网络或者以太网将树莓派设备连接到Internet, 配置方法请参考树莓派配置WiFi无线网络。
- ii. 配置SSH服务。
 - 执行 sudo systemctl start ssh 命令开启SSH服务。
 - 执行 sudo systemctl status ssh 命令查看SSH服务。
 - 执行 sudo systemctl enable ssh 设置开机自动启动SSH服务。

在启动SSH服务之后,执行 hostname -I 命令获取树莓派的IP地址,通过网络SSH远程登录的方式登录树莓派设备。

- 4. 检查Link IoT Edge运行环境依赖。
 - i. 在树莓派设备的本地终端窗口或者SSH终端窗口执行以下命令, 下载环境检查工具并运行。

wget http://iotedge-web.oss-cn-shanghai.aliyuncs.com/public/testingTool/link-iot-ed ge_env-check.sh sudo chmod +x ./link-iot-edge_env-check.sh sudo ./link-iot-edge env-check.sh

⑦ 说明 *link-iot-edge_env-check.sh*脚本在Link IoT Edge支持的平台需要以root权限运行并 需要以下Linux系统命令: printf、echo、cat、ls、expr、grep、test、uname、set、head、 sort、cut、uniq、xargs、if config。

ii. 按照运行环境检查工具的提示在您的机器上安装所有必需的依赖项,当检查工具成功运行完成后, 返回如下图信息,表示Link IoT Edge能够在您的机器上成功运行。

System configuration:			
Kernel architecture: x86 64			
Kernel version: 4.15			
C library version: 2.23			
C library: Ubuntu GLIBC 2.23-Oubuntul0			
Init process: /lib/systemd/systemd			
Check Depended Software Version:			
OpenSSL version: 1.0.1			
Python version: 3.5.2			
Check Depended Commands:			
[yes] Checking for command wget			
[yes] Checking for command realpath			
[yes] Checking for command tar			
[yes] Checking for command unzip			
[yes] Checking for command readlink			
[yes] Checking for command basename			
[yes] Checking for command dirname			
[yes] Checking for command pidof			
[yes] Checking for command df			
[yes] Checking for command grep			
[yes] Checking for command ps			
[yes] Checking for command kill			
[yes] Checking for command dirname			
[yes] Checking for command xargs			
[yes] Checking for command umount			
[yes] Checking for command unshare			
[yes] Checking for command awk			
[yes] Checking for command mkdir			
*******************Link IoT Edge v1.8 Run Environment Check Result:************************************			
Suggestion:			
The recognized to represent the representation of the thirt process.			
Flease using systema to manage Link for Eage service.			
You are install the link to Dida of a setting of the device			
Tou can install the Link for Edge VI.8 software on the device.			

创建边缘实例和网关

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击边缘实例。

3. 创建一个边缘实例。

i. 单击新增实例,在弹出对话框中设置实例名称。

⑦ 说明 实例名称支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长度不超过20个字符,1个汉字算2个字符。

ii. 根据所搭建的环境,选择对应的Link loT Edge产品规格。详细介绍,请参见产品规格。

物联网边缘计算支持自动分配网关到边缘实例,您可以无需设置其它新增实例相关参数,直接跳转 到步骤4。物联网边缘计算自动为您的边缘实例创建名为LEGatewayAuto的网关产品,并在该产品 下添加一个随机命名的网关设备。

? 说明

- 物联网边缘计算自动为您创建LEGatewayAuto产品时,若您已有该名称的产品,则会 在该名称后添加随机后缀,作为自动创建的网关产品。
- 物联网边缘计算已为您自动创建过一次LEGatewayAuto产品后,在您后续新建边缘实例并自动分配网关时,会在LEGatewayAuto产品下新增网关设备,分配到您新的边缘实例中。
- iii. (可选)单击标签信息下的新增标签,可以设置实例标签。通过标签您可以更加有效地归类及识 别实例。您也可以不设置标签。

参数	描述
标签key	不可为空,仅支持英文字母大小写,长度不超过20个字符,同一个边缘实例不可 以重复定义标签key。
标签value	不可为空,支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长 度不超过20个字符,1个中文汉字算2个字符。

若设置实例标签,请填写标签key和标签value。参数说明

4. (可选)若您想要手动创建网关产品和设备并分配到边缘实例,请按如下步骤操作。

i. 在新增实例对话框,单击高级选项。

ii. 在**网关产品**下单击新建网关产品,为实例创建网关。

物联网边缘计算中的网关承载边缘计算能力,每个实例必须分配一个网关设备,并且该网关设备同一时间只能被分配到一个边缘实例。

新增实例		\times
* 实例名称		
LinkloTEdge_Node		
产品规格 ②		
100		\sim
标签信息		
+新增标签		
高级选项		
* 网关产品		
请选择网关产品		\sim
新建网关产品		
	确定	取消

iii. 在**创建产品**对话框中,设置网关产品参数,然后单击完成。

物联网边缘计算中的**新建网关产品**,继承了物联网平台**设备管理 > 产品**中的产品功能,此处已自动为您简化了创建产品的步骤,以便您更快速地创建适合物联网边缘计算中使用的网关产品。

创建产品	×
产品信息	
* 产品名称	
LinkloTEdge_Gateway	
* 所属品类 👔 🔿 标准品类 💿 自定义品类	
更多信息	
∨产品描述	
使用文档	完成取消

参数说明

参数	说明
产品名称	为网关产品设置名称,用于后续的查询及识别网关产品。支持中文、英文字母大 小写、数字和下划线(_),长度限制4~30个字符,一个中文汉字算2位。
所属品类	选择品类,为该产品定义物模型。 可选择的值为: • 标准品类:选择任一物联网平台预定义的品类,快速完成产品的功能定义。 选择产品模板后,您可以在该模板基础上,编辑、修改、新增功能。 • 自定义品类:需根据实际需要,定义产品功能。 若您需要的网关没有特殊功能定义,建议您选择自定义品类。
产品描述	可输入文字,用来描述产品信息。字数限制为100个字符。可以为空。

产品创建成功后,自动跳转回新增实例对话框,并且在网关产品参数下自动分配刚刚创建的网关 产品。 iv. 在新增实例对话框,单击网关设备下的新建网关设备,为网关产品添加设备。 物联网边缘计算中的新建网关设备功能,继承物联网平台设备管理 > 设备的功能。

新增实例	×
* 实例名称	
LinkloTEdge_Node	
产品规格 ②	
10.0	\sim
标签信息 +新增标签	
高级选项 へ	
* 网关产品	
LinkloTEdge_Gateway	\sim
新建网关产品	
* 网关设备	
请选择网关设备	\sim
新建网关设备	
	确定 取消
	确定 取消

v. 根据界面提示设置参数后,单击确认。

参数说明

参数	描述
产品	系统已自动关联上一步创建的网关产品。
	为该网关设备命名。设备名称需保持产品内唯一。如不填写,系统将自动生成。
设备名称	⑦ 说明 设备名称长度为4~32个字符,可包含英文字母、数字和特殊字符,包括短划线(-)、下划线(_)、at(@)、英文句号(.)和英文冒号(:)。

5. 实例参数设置完成后,单击确定,至此您已创建边缘实例和网关。

安装并启动Link IoT Edge

根据本文上方**创建边缘实例和网关**内容,创建完成边缘实例并分配网关后,您需要在网关上安装并启动Link loT Edge。

1. 在边缘实例页面,单击实例名称右侧的软件安装。

边缘实例				
新增实例 授权 清输入实例名称	Q			C
实例名称	网关状态	实例状态	创建时间	操作
LinkloTEdge_Node	● 在线	● 未部署	2019-12-18 23:10:43	查看 删除 软件安装

2. 根据环境设置软件安装相关参数,然后单击生成安装命令。

软件安装			×
	还没有网关?通过免费虚执 您可免费使用 CloudShell 虚拟和 的产品功能。 立即体验	以机快速体验! 肌,快速体验物联网边	2家计算
* 边缘网关 CPU	架构 💿		
ARMv7 VFPv	3		\sim
产品规格 💿 标准版			
*边缘版本 🕜			
2.4.0			\sim
* 操作系统			
Linux			\sim
了解软件安装?		生成安装命令	取消

参数说明

参数	描述
边缘网关CPU架构	您的设备系统对应的CPU架构。此处选择ARMv7 VFPv3。
产品规格	在创建边缘实例时,已选择实例中使用的Link loT Edge版本。此处不可操作。
边缘版本	选择Link IoT Edge的 <mark>发布版本</mark> 。
操作系统	选择您的设备对应的操作系统。此处选择Linux。

如果您暂时没有合适的硬件载体来运行Link IoT Edge,您可以单击对话框上方的**立即体验**,使用Cloud Shell虚拟机快速体验标准版Link IoT Edge的产品功能。使用**立即体验**功能有诸多注意事项,详情请参见基于阿里云Cloud Shell快速体验Link IoT Edge。

○ 注意 同一个边缘实例只能启动一个Link IoT Edge,若在已经商用的边缘实例(即边缘实例已 在硬件网关上启动了Link IoT Edge)使用**立即体验**功能时,会导致硬件网关上启动的Link IoT Edge 和虚拟机上启动的Link IoT Edge不断互踢,影响您的正常业务。强烈建议您不要在已启动Link IoT Edge的边缘实例上使用**立即体验**功能。

3. 复制软件包下载命令备用。

软件安装							
✔ 请复制命令,并在目标设	✔ 请复制命令,并在目标设备的控制台上以root权限执行命令。						
边缘实例 CPU 架构	ARMv7 VFPv3						
产品规格	标准版						
边缘版本	2.4.0						
安装网关	gateway_01						
操作系统	Linux						
运行所需 RAM	≥ 128 MB						
安装存储空间需求	≥ 256 MB						
软件包下载命令	复制命令						
没有合适的硬件载体?	完成重新生成						

- 4. 登录您的树莓派系统机器后台。
- 5. 任意目录下执行步骤3中已复制的命令。

该命令实现一键下载、配置并启动Link IoT Edge。命令执行完成后,会在当前目录中下载*link-iot-edge-standard.sh*脚本。

⑦ 说明 如果不是第一次安装启动Link IoT Edge,可使用已下载的Link-iot-edge-standard.sh脚本,对Link IoT Edge进行重启、停止、获取状态、修改配置参数等操作,命令详情请见下图。
 # Intsall the linkedge with parameters
USAGE1: ./link-iot-edge-standard.sh --install Arch Version
 Arch : x86-64, armv7, armv7-hf, aarch64
 Version: vl. 8. 1, vl. 8. 2
 # Config the linkedge with startup parameters
USAGE2: ./link-iot-edge-standard.sh --config ProductKey DeviceName DeviceSecret
 # Start the linkedge
USAGE3: ./link-iot-edge-standard.sh --start
 # Get the linkedge status
USAGE4: ./link-iot-edge-standard.sh --status

6. 执行如下命令查看Link IoT Edge核心服务的运行状态。

sudo ./link-iot-edge-standard.sh --status

若系统显示如下信息,表示Link loT Edge核心服务启动成功。

				:/\$./link-iot-edge-standard.shstatus
				Link IoT Edge Service Status
[+]	Service	mbusd.conf is active, pid is 10066
[+]	Service	config-manager is active, pid is 10093
Ē	+]	Service	data-manager is active, pid is 10112
Ē	+]	Service	watch-dog is active, pid is 10145
[+]	Service	cloud-proxy is active, pid is 10102
[+]	Service	dimu is active, pid is 10179
[+]	Service	message-router is active, pid is 10174
	+]	Service	fota is active, pid is 10195
[+]	Service	gateway-monitor is active, pid is 10213
	+]	Service	task-dispatcher is active, pid is 10208
	+]	Service	credential is active, pid is 10108
	+]	Service	fc-base is active, pid is 10359
L	+		Service	ifttt is active, pid is 10230
ļ	+		Service	redis-server is active, pid is 10061
L	+		Service	logger is active, pid is 10069
	+		Service	service-monitor is active, pid is 10371
L	+		Service	RemoteTerminalDaemon is active, pid is 13593
	+		Service	nginx is active, pid is 10043

您也可以在<mark>边缘计算控制台</mark>,左侧导航栏选择**边缘实例**,在已创建好的边缘实例右侧单击查看进入实 例详情页面,选择网关,查看网关状态。

网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置		
替换网关	软件安装	Ē								
网关名称		产品名	称			远程访问		状态	最后上线时间	操作
gateway	01	Linklo	TEdge_Gateway					● 在线	2020-01-17 16:28:00	查看 分享远程SSH终端 远程SSH终端 更多远程服务

打开远程访问按钮,可对网关进行远程管理,详细操作步骤请参见远程服务访问。

- 7. (可选)在**实例详情**页面,查看CPU使用率、内存使用率、存储使用率以及实例进程需要授权访问阿里 云云监控(CloudMonitor)服务。
 - i. 添加角色或分配已有的角色,并确保该角色拥有管理云监控(CloudMonitor)的权限。详细操作,请参见云资源访问。
 - ii. 选择监控信息页签,打开页面右侧的云监控状态开关,如下图所示。

网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置	
1 注意	: 新增资源监控功	能(场景联动、函	数计算、驱动、 沪	〔数据分析等) 霊	需要依赖v2.1.0及以_	上版本,否则无法	准确反映运行状态	5. 请确保升	升级到最新版本。
监控信息									
主机监护	空 场景联动	边缘应用	驱动监控	流数据监	空				
报警配置	最近1小	4 ~							G

云监控状态打开后,可在**监控信息**页签,查看网关的各类监控信息。

8. (可选) Link IoT Edge支持将边缘实例授权给其他阿里云账号操作。

在边缘实例页面,单击左上角授权,在弹出对话框中单击新增授权,根据界面提示设置参数。

边缘实例					
新增实例 授权 请输入实例名称	Q				C
实例名称	网关托本	定确状态	创建时间	操作	
particular in the second	新增授权	×	2019-12-18 23:10:43	查看 删除 软件安装	
and top (seal	 授权用户 UID 请输入授权用户 UID 	功	2019-12-13 10:47:13	查看 删除 软件安装	
and the local of	* 授权实例	功	2019-12-13 10:37:35	查看 删除 软件安装	
performent.	请选择授权实例		2019-12-11 17:29:03	查看 删除 软件安装	
101603		确认取消	2019-12-04 15:02:01	查看 删除 软件安装	
Transform	● 高线	● 部署成功	2019-11-13 15:38:53	查看 删除 软件安装	

参数说明

参数	描述
授权用户UID	获取被授权用户的阿里云账号ID,填入此处。
授权实例	选择允许该阿里云账号ID的用户进行远程运维操作的边缘实例。

使用systemd管理Link IoT Edge

您可以使用systemd来管理Link loT Edge服务的启动(start)、停止(stop)和查看状态(status)。 Link loT Edge的systemd service如下所示。

[Unit]
Description=Link IoT Edge
[Service]
Type=forking
Restart=on-failure
<pre>ExecStart=/linkedge/gateway/build/script/iot_gateway_start.sh</pre>
<pre>ExecReload=/linkedge/gateway/build/script/iot_gateway_start.sh</pre>
ExecStop=/linkedge/gateway/build/script/iot_gateway_stop.sh
[Install]
WantedBy=multi-user.target

您可以运行如下命令,配置开机自启动功能。

wget http://iotedge-web.oss-cn-shanghai.aliyuncs.com/public/testingTool/LinkIoTEdge.service sudo cp LinkIoTEdge.service /etc/systemd/system/LinkIoTEdge.service sudo systemctl enable LinkIoTEdge.service

其它常用命令如下所示。

- 启动命令: sudo systemctl start LinkIoTEdge.service
- 重启命令: sudo systemctl restart LinkIoTEdge.service
- 停止命令: sudo systemctl stop LinkIoTEdge.service
- 禁用开机自启动: sudo systemctl disable LinkIoTEdge.service

下一步

环境搭建完成后,您可以根据设备接入章节内容,把您的设备接入到物联网边缘计算。同时也可以为边缘实 例分配其他资源(如函数计算、消息路由等)管理您的设备。

接入设备或分配其他资源到边缘实例后,需要根据如下步骤部署边缘实例。

- 1. 在实例详情页面,单击右上角部署后在弹出框中单击确定,部署边缘实例。
- 2. 当部署状态显示为**部署成功**,表示部署实例完成。您可以单击**查看日志**,查看部署详情。您也可以通 过单击**实例详情**页面右上角的**查看部署日志**,查看历史部署情况。

产品规格 标准版 CPU使用率 2.58% 3	臣者	部署进程	×	Ē	
网关 监控信息	设备与驱动	✔ gateway_01 网关部署完成			
苔换网关 软件安装		部署进程			
网关名称	产品名利	 ・部署开始 	2020-01-17 16:38:30	最后上线时间	操作
		· 部署装配	完成		春看 公言沅程SSH终端
gateway_01	LinkloTi	· 部署打包		2020-01-17 16:28:00	远程SSH终端 更多远程服务
		 部署分发 	元成		
		 部署结束 	2020-01-17 16:38:31		

1.2.3. 基于阿里云Cloud Shell快速体验Link IoT Edge

如果您暂时没有合适的硬件载体来运行Link IoT Edge,您可以使用阿里云云命令行(Cloud Shell)虚拟机快速体验标准版Link IoT Edge的产品功能。

Cloud Shell是阿里云提供的网页版命令行工具。在开始操作本章内容前,请您仔细阅读Cloud Shell介绍及使用 限制。

Cloud Shell会启动一台虚拟机,并在该虚拟机上为边缘实例搭建最新版本的Link IoT Edge标准版(LE Standard)。

创建边缘实例和网关

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击边缘实例。
- 3. 创建一个边缘实例。
 - i. 单击新增实例,在弹出对话框中设置实例名称。

⑦ 说明 实例名称支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长度不超过20个字符,1个汉字算2个字符。

ii. 根据所搭建的环境,选择对应的Link loT Edge产品规格。详细介绍,请参见产品规格。

物联网边缘计算支持自动分配网关到边缘实例,您可以无需设置其它新增实例相关参数,直接跳转 到步骤4。物联网边缘计算自动为您的边缘实例创建名为LEGatewayAuto的网关产品,并在该产品 下添加一个随机命名的网关设备。

? 说明

- 物联网边缘计算自动为您创建LEGatewayAuto产品时,若您已有该名称的产品,则会 在该名称后添加随机后缀,作为自动创建的网关产品。
- 物联网边缘计算已为您自动创建过一次LEGatewayAuto产品后,在您后续新建边缘实例并自动分配网关时,会在LEGatewayAuto产品下新增网关设备,分配到您新的边缘实例中。
- iii. (可选)单击标签信息下的新增标签,可以设置实例标签。通过标签您可以更加有效地归类及识 别实例。您也可以不设置标签。

若设置实例标签,请填写标签key和标签value。参数说明

参数	描述
标签key	不可为空,仅支持英文字母大小写,长度不超过20个字符,同一个边缘实例不可 以重复定义标签key。
标签value	不可为空,支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长 度不超过20个字符,1个中文汉字算2个字符。

4. (可选)若您想要手动创建网关产品和设备并分配到边缘实例,请按如下步骤操作。

i. 在新增实例对话框,单击高级选项。

ii. 在网关产品下单击新建网关产品,为实例创建网关。

物联网边缘计算中的网关承载边缘计算能力,每个实例必须分配一个网关设备,并且该网关设备同一时间只能被分配到一个边缘实例。

新增实例		\times
* 实例名称		
LinkloTEdge_Node		
产品规格 🕜		
1.01		\sim
标签信息 +新增标签		
高级选项へ		
* 网关产品		
请选择网关产品		\sim
新建网关产品		
	确定	取消

iii. 在**创建产品**对话框中,设置网关产品参数,然后单击完成。

物联网边缘计算中的**新建网关产品**,继承了物联网平台**设备管理 > 产品**中的产品功能,此处已自动为您简化了创建产品的步骤,以便您更快速地创建适合物联网边缘计算中使用的网关产品。

创建产品	\times
产品信息	
* 产品名称	
LinkloTEdge_Gateway	
* 所属品类 💿 「标准品类 ● 自定义品类	
更多信息	
∨产品描述	
使用文档 完成 取消	á

参数说明

参数	说明
产品名称	为网关产品设置名称,用于后续的查询及识别网关产品。支持中文、英文字母大 小写、数字和下划线(_),长度限制4~30个字符,一个中文汉字算2位。
所属品类	选择品类,为该产品定义物模型。 可选择的值为: • 标准品类:选择任一物联网平台预定义的品类,快速完成产品的功能定义。 选择产品模板后,您可以在该模板基础上,编辑、修改、新增功能。 • 自定义品类:需根据实际需要,定义产品功能。 若您需要的网关没有特殊功能定义,建议您选择自定义品类。
产品描述	可输入文字,用来描述产品信息。字数限制为100个字符。可以为空。

产品创建成功后,自动跳转回新增实例对话框,并且在网关产品参数下自动分配刚刚创建的网关 产品。 iv. 在新增实例对话框,单击网关设备下的新建网关设备,为网关产品添加设备。 物联网边缘计算中的新建网关设备功能,继承物联网平台设备管理 > 设备的功能。

新增实例	×
* 实例名称	
LinkloTEdge_Node	
产品规格 ②	
10.0	\sim
标签信息 +新增标签	
高级选项 へ	
* 网关产品	
LinkloTEdge_Gateway	\sim
新建网关产品	
* 网关设备	
请选择网关设备	\sim
新建网关设备	
	确定 取消
	确定 取消

v. 根据界面提示设置参数后, 单击确认。

参数说明

参数	描述
产品	系统已自动关联上一步创建的网关产品。
	为该网关设备命名。设备名称需保持产品内唯一。如不填写,系统将自动生成。
设备名称	 ⑦ 说明 设备名称长度为4~32个字符,可包含英文字母、数字和特殊字符,包括短划线(-)、下划线(_)、at(@)、英文句号(.)和英文冒号(:)。

5. 实例参数设置完成后,单击确定,至此您已创建边缘实例和网关。

安装并启动Link IoT Edge

根据本文上方**创建边缘实例和网关**内容,创建完成边缘实例并分配网关后,您需要在网关上安装并启动Link loT Edge。

1. 在边缘实例页面,单击实例名称右侧的软件安装。

边缘实例					
新增实例 授权 请输入实例名称	Q			C	
实例名称	网关状态	实例状态	创建时间	操作	
LinkloTEdge_Node	 ● 在线 	 ● 未部署 	2019-12-18 23:10:43	查看 删除 软件安装	

2. 在**软件安装**对话框中,单击**立即体验**,使用Cloud Shell虚拟机快速体验标准版Link IoT Edge产品功能, Cloud Shell详细操作请参见云命令行。

软件安装	×
还没有网关?通过免费虚拟机快速体验! 您可免费使用 CloudShell 虚拟机,快速体验物联网边缘计算的产品功能。 立即体验	
* 边缘网关 CPU 架构 🕜	
请选择CPU 架构	\sim
产品规格 @ 标准版 * 边缘版本 @	~
* 操作系统	
请选择操作系统	\sim
了解软件安装? 生成安装命令 取消	Ξ

3. 启动Cloud Shell后, 会在虚拟机上为当前边缘实例安装最新的标准版Link IoT Edge。安装完成后Cloud Shell中会出现 Finish install and run LinkIoTEdge 的提示,实例中的网关会变成在线状态。

产品规格	标曲	Ŕ						服务版本号	v2.4.0			
CPU 使用率	- 査者	1						内存使用率	- 查看			
	1次+办/合白	20.4z 는 317.5h		14 AB ch m	巡自攻由	口士吧友						
MX	페지막소	反留 与 兆 4 月	功轰跃纠	纪纳亚用	间尽哈田	口心服劳	议皇					
XIXMX	软件安装	Ē										
网关名称			产品名称				ì	元程访问	状态	最后上线时间	操作	
gateway_	01		LinkloTEdge_G	ateway1					● 在线	2020-01-20 17:22:32	查看 分享远程SSH终端 远程SSH终端 更多远期	5]]服务
Clor	ud Shell	×	+								-	
$\leftarrow \rightarrow$	C 🔒 🤋	shell.aliyun.co	m/	ar birda	-		1000	September 1	ter and the second second	Renard at 2008p	\$	Θ:
▶ Clo	oud Shell											2 :
U 🕸	📑 < 您的	文件会随实例释放	女而销毁, 挂载	存储空间以持久	化存储 🗙							
linkedg linkedg linkedg linkedg *******	linkedged: running nginx service linkedged: running lada-mgtt service linkedged: running high-availability service linkedged: start up completed Finish install and run LinkloTEdge											
您可以说 https:/	方问阿里云物 /iot.conso	Ŋ联网边缘计 le.aliyun.c	算平台,进 om/le/insta	行更多功能的 ance/detail	的体验: l?id=h2uk1	Q431yT8JqT	R500C					
shell@A	licloud:~\$											

使用**立即体验**功能时,需要注意如下事项:

- 单击**立即体验**,系统会为您启动Cloud Shell并自动创建一台Linux虚拟机供您使用;重启Cloud Shell
 时,会将已创建的Linux虚拟机销毁并创建一台新的Linux虚拟机。
- 关闭Cloud Shell会话窗口并不能立即停止运行Link IoT Edge,在Cloud Shell会话窗口无操作45分钟 后,或者关闭所有Cloud Shell会话窗口15分钟后,虚拟机会被销毁,此时Link IoT Edge才会停止运 行。
- 如果您需要同时在多个窗口中访问虚拟机,单击控制台右上角的 ∑ 图标,启动Cloud Shell会话窗

口,最多同时可打开5个会话窗口。Cloud Shell在同一时刻只会创建一台虚拟机,因此打开的多个会话窗口都会自动连接到此虚拟机上。

费用	工单	备案	企业	支持与服务	D	Ċ.	Ä	?	ŝ	简体中文	@

- 单击**立即体验**启动Cloud Shell后,若发现该虚拟机已经安装过Link IoT Edge(例如,同一个边缘实例 多次单击**立即体验**时),会先卸载之前的Link IoT Edge,并在虚拟机上为当前的边缘实例安装新的标 准版Link IoT Edge。
- 已通过**立即体验**功能在虚拟机上启动Link IoT Edge后,如果您需要在自己的硬件中安装Link IoT Edge,请先将运行在虚拟机上的Link IoT Edge停止,否则可能会导致部分功能异常。可以通过如下两种方式停止Link IoT Edge:
 - 在Cloud Shell会话窗口左上角单击 🕛 图标重启Cloud Shell。
 - 在Cloud Shell中执行/linkedge/gateway/build/script/iot_gateway_stop.sh命令。
- (可选)在实例详情页面单击网关页签,打开远程访问开关后,可对网关进行远程管理。详细操作步骤,请参见远程服务访问。
- 5. (可选)在**实例详情**页面,查看CPU使用率、内存使用率、存储使用率以及实例进程需要授权访问阿里 云云监控(CloudMonitor)服务。
 - i. 添加角色或分配已有的角色,并确保该角色拥有管理云监控(CloudMonitor)的权限。详细操作,请参见云资源访问。

ii. 选择**监控信息**页签,打开页面右侧的**云监控状态**开关,如下图所示。



云监控状态打开后,可在**监控信息**页签,查看网关的各类监控信息。

6. (可选) Link IoT Edge支持将边缘实例授权给其他阿里云账号操作。

在边缘实例页面,单击左上角授权,在弹出对话框中单击新增授权,根据界面提示设置参数。

边缘实例					
新增实例 授权 请输入实例名称	Q				G
实例名称	颐羊状态	实际状态	创建时间	操作	
performent.	新增授权	×	2019-12-18 23:10:43	查看 删除 软件安装	
intelling (seal	* 授权用户 UID 请输入授权用户 UID	τh	2019-12-13 10:47:13	查看 删除 软件安装	
parametering a financial	* 授权实例	功	2019-12-13 10:37:35	查看 删除 软件安装	
per la terre	请选择授权实例	~ 功	2019-12-11 17:29:03	查看 删除 软件安装	
58-1503		确认 取消	2019-12-04 15:02:01	查看 删除 软件安装	
Taulos I	● 离 线	● 部署成功	2019-11-13 15:38:53	查看 删除 软件安装	

参数说明

参数	描述
授权用户UID	获取被授权用户的阿里云账号ID,填入此处。
授权实例	选择允许该阿里云账号ID的用户进行远程运维操作的边缘实例。

下一步

环境搭建完成后,您可以根据设备接入章节内容,把您的设备接入到物联网边缘计算。同时也可以为边缘实 例分配其他资源(如函数计算、消息路由等)管理您的设备。

接入设备或分配其他资源到边缘实例后,需要根据如下步骤部署边缘实例。

- 1. 在实例详情页面,单击右上角部署后在弹出框中单击确定,部署边缘实例。
- 2. 当部署状态显示为**部署成功**,表示部署实例完成。您可以单击**查看日志**,查看部署详情。您也可以通 过单击**实例详情**页面右上角的**查看部署日志**,查看历史部署情况。

边缘计算 / 边缘实例 / 实例详情				
← LinkloTEdge_Noc	de ^{msaub}			查看部署日志 重置 部署
产品规格 标准版 CPU 使用率 2.58 % 查看	部署进程	×		
网关 监控信息 设备与驱动	✓ gateway_01 网关部署完成			
苗扱网关 软件安装 网关名称 产品名	部署进程 • 部署开始 ————————————————————————————————————	2020-01-17 16:38:30	最后上线时间	操作
gateway_01 LinkioTi	 部署装配 部署打包 	売成 売成	2020-01-17 16:28:00	查看 分享远程SSH终端 远程SSH终端 更多远程服务
	 ・部署分发 ・部署结束 	交 完成		
		查看日志关闭		

1.3. 轻量版环境搭建

1.3.1. 基于Ubuntu 16.04搭建环境

本文介绍基于Ubuntu 16.04系统搭建Link IoT Edge轻量版本(LE Lite)运行环境的方法。

前提条件

- Link IoT Edge的远程登录功能依赖设备的SSH服务,请确保设备上已经开启了SSH服务。SSH的详细信息可参考OpenSSH的使用。
- 请确保设备的本地环回端口已启用,即在设备上执行ping 127.0.0.1命令时,返回结果正常。

```
root@ :~# ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.021 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.040 ms
64 bytes from 127.0.0.1: icmp seq=3 ttl=64 time=0.044 ms
```

创建边缘实例和网关

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击边缘实例。
- 3. 创建一个边缘实例。
 - i. 单击新增实例,在弹出对话框中设置实例名称。

⑦ 说明 实例名称支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长度不超过20个字符,1个汉字算2个字符。

ii. 根据所搭建的环境,选择对应的Link loT Edge产品规格。详细介绍,请参见产品规格。

物联网边缘计算支持自动分配网关到边缘实例,您可以无需设置其它新增实例相关参数,直接跳转 到<mark>步骤4</mark>。物联网边缘计算自动为您的边缘实例创建名为LEGatewayAuto的网关产品,并在该产品 下添加一个随机命名的网关设备。

? 说明

- 物联网边缘计算自动为您创建LEGatewayAuto产品时,若您已有该名称的产品,则会 在该名称后添加随机后缀,作为自动创建的网关产品。
- 物联网边缘计算已为您自动创建过一次LEGatewayAuto产品后,在您后续新建边缘实例并自动分配网关时,会在LEGatewayAuto产品下新增网关设备,分配到您新的边缘实例中。
- iii. (可选)单击标签信息下的新增标签,可以设置实例标签。通过标签您可以更加有效地归类及识 别实例。您也可以不设置标签。

若设置实例标签,请填写标签key和标签value。参数说明

参数	描述
标签key	不可为空,仅支持英文字母大小写,长度不超过20个字符,同一个边缘实例不可 以重复定义标签key。
标签value	不可为空,支持中文、英文字母大小写、数字、下划线(_)和短划线(-),长 度不超过20个字符,1个中文汉字算2个字符。

4. (可选) 若您想要手动创建网关产品和设备并分配到边缘实例,请按如下步骤操作。

i. 在新增实例对话框,单击高级选项。

ii. 在**网关产品**下单击新建网关产品,为实例创建网关。

物联网边缘计算中的网关承载边缘计算能力,每个实例必须分配一个网关设备,并且该网关设备同一时间只能被分配到一个边缘实例。

新增实例		\times
* 实例名称		
LinkloTEdge_Node		
产品规格 🕜		
1.01		\sim
标签信息 +新增标签		
高级选项へ		
* 网关产品		
请选择网关产品		\sim
新建网关产品		
	确定	取消

iii. 在**创建产品**对话框中,设置网关产品参数,然后单击完成。

物联网边缘计算中的**新建网关产品**,继承了物联网平台**设备管理 > 产品**中的产品功能,此处已自动为您简化了创建产品的步骤,以便您更快速地创建适合物联网边缘计算中使用的网关产品。

创建产品	\times
产品信息	
* 产品名称	
LinkloTEdge_Gateway	
* 所属品类 💿 「标准品类 ● 自定义品类	
更多信息	
∨产品描述	
使用文档 完成 取消	á

参数说明

参数	说明
产品名称	为网关产品设置名称,用于后续的查询及识别网关产品。支持中文、英文字母大 小写、数字和下划线(_),长度限制4~30个字符,一个中文汉字算2位。
所属品类	选择品类,为该产品定义物模型。 可选择的值为: • 标准品类:选择任一物联网平台预定义的品类,快速完成产品的功能定义。 选择产品模板后,您可以在该模板基础上,编辑、修改、新增功能。 • 自定义品类:需根据实际需要,定义产品功能。 若您需要的网关没有特殊功能定义,建议您选择自定义品类。
产品描述	可输入文字,用来描述产品信息。字数限制为100个字符。可以为空。

产品创建成功后,自动跳转回**新增实例**对话框,并且在**网关产品**参数下自动分配刚刚创建的网关 产品。 iv. 在新增实例对话框,单击网关设备下的新建网关设备,为网关产品添加设备。 物联网边缘计算中的新建网关设备功能,继承物联网平台设备管理 > 设备的功能。

新增实例	×
* 实例名称	
LinkloTEdge_Node	
产品规格 ②	
10.0	\sim
标签信息 +新增标签	
高级选项 へ	
* 网关产品	
LinkloTEdge_Gateway	\sim
新建网关产品	
* 网关设备	
请选择网关设备	\sim
新建网关设备	
	确定 取消
	确定 取消

v. 根据界面提示设置参数后, 单击确认。

参数说明

参数	描述
产品	系统已自动关联上一步创建的网关产品。
	为该网关设备命名。设备名称需保持产品内唯一。如不填写,系统将自动生成。
设备名称	 ⑦ 说明 设备名称长度为4~32个字符,可包含英文字母、数字和特殊字符,包括短划线(-)、下划线(_)、at(@)、英文句号(.)和英文冒号(:)。

5. 实例参数设置完成后,单击确定,至此您已创建边缘实例和网关。

安装并启动Link IoT Edge

根据本文上方**创建边缘实例和网关**内容,创建完成边缘实例并分配网关后,您需要在网关上安装并启动Link loT Edge。

1. 在边缘实例页面,单击实例名称右侧的软件安装。

边缘实例				
新增实例 授权 调输入实例名称	Q			C
实例名称	网关状态	实例状态	创建时间	操作
LinkloTEdge_Node	● 在线	 未部署 	2019-12-18 23:10:43	查看 删除 软件安装

2. 根据环境设置下载参数,然后单击生成安装命令。

・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	软件安装	×
 边缘网关 CPU 架构 ② x86-64 ✓ 全品规格 ③ 全量版 边缘版本 ② 2.4.0 ✓ 操作系统 Linux ✓ 	还没有网关?通过免费虚拟机快速体验! 您可免费使用 CloudShell 虚拟机,快速体验物联网边缘计算的产品功能。 立即体验	
x86-64 ✓ 辛品规格 ② 2里版 24.0 24.0 ✓ 操作系统 ✓	边缘网关 CPU 架构 🕜	
 辛品规格 ② 全量版 辺缘版本 ③ 2.4.0 ✓ 操作系统 Linux ✓ 	x86-64	\sim
辺缘版本	←品规格 ② 径量版	
2.4.0 × 操作系统 Linux ×	・ 边缘版本 🕜	
操作系统 Linux	2.4.0	\sim
Linux	* 操作系统	
	Linux	\sim
了解软件安装?	了解软件安装? 里成安装命令 取消	Í

参数说明

参数	描述
边缘网关CPU架构	您的设备系统对应的CPU架构。此处选择x86-64。
产品规格	在创建边缘实例时,已选择实例中使用的Link loT Edge版本。此处不可操作。
边缘版本	选择Link loT Edge的 <mark>发布版本</mark> 。
操作系统	选择您的设备对应的操作系统。此处选择Linux

3. 复制操作系统命令备用。

软件安装	×
✔ 请复制命令,并在目标设	备的控制台上以root权限执行命令。
边缘实例 CPU 架构	x86-64
产品规格	轻量版
边缘版本	2.4.0
安装网关	gateway_04
操作系统	Linux
运行所需 RAM	≥ 32 MB
安装存储空间需求	≥ 256 MB
软件包下载命令	复制命令
没有合适的硬件载体?	完成重新生成

- 4. 登录您Ubuntu 16.04系统的机器。
- 5. 任意目录下执行步骤3中已复制的命令。

该命令实现一键下载、配置并启动Link IoT Edge。命令执行完成后,会在当前目录中下载*link-iot-edge-lite.sh*脚本。

② 说明 如果不是第一次安装启动Link IoT Edge,可使用已下载的*link-iot-edge-lite.sh*脚本,对 Link IoT Edge进行重启、停止、获取状态、修改配置参数等操作,命令详情请见下图:

```
# Intsall the linkedge with parameters
USAGE1: ./link-iot-edge-lite.sh --install Arch Version
Arch : x86-64, armv7, armv7-hf, aarch64, win64, win32, Mac
Version: v1. 8. 1, v1. 8. 2
# Start the linkedge
USAGE3: ./link-iot-edge-lite.sh --start ProductKey DeviceName DeviceSecret
# Get the linkedge status
USAGE4: ./link-iot-edge-lite.sh --status
# Stop the linkedge
USAGE4: ./link-iot-edge-lite.sh --status
# Stop the linkedge
USAGE4: ./link-iot-edge-lite.sh --status
```

若系统显示如下信息,表示Link loT Edge核心服务启动成功。



您也可以在<mark>边缘计算控制台</mark>,左侧导航栏选择**边缘实例**,在已创建好的边缘实例右侧单击查看进入**实** 例详情页面,选择网关,查看网关状态。

← Li	nkloTE	dge_	Node				
产品规格	轻量	板		服务版本号	v2.4.0		
CPU 使用率	- 查	3		内存使用率	- 查看		
网关	监控信息	设置					
替接网)	关 软件安装	ę					
网关名称	8		产品名称	远程访问	状态	最后上线时间	操作
gateway	_01		LinkloTEdge_Gateway		 在线 	2020-01-20 17:41:04	查看 分享远程SSH终端 远程SSH终端 更多远程服务

6. (可选)Link loT Edge支持将边缘实例授权给其他阿里云账号操作。

在边缘实例页面,单击左上角授权,在弹出对话框中单击新增授权,根据界面提示设置参数。

边缘实例					
新增实例 授权 请输入实例名称	Q				C
实例名称	网关状态	实际状态	创建时间	操作	
perfigures i	新增授权	×	2019-12-18 23:10:43	查看 删除 软件安装	
and so had	* 授权用户 UID	Th	2010-12-13 10:47:13	杏 芳 曽陸 約件穴準	
	请输入投伙用户 UID		2010 12 10 10.47.10		
and the second second	* 授权实例	功	2019-12-13 10:37:35	查看 删除 软件安装	
and the local	请选择授权实例	لم لي	2019-12-11 17:29:03	查看 删除 软件安装	
10		确认 取消	2019-12-04 15:02:01	查看 删除 软件安装	
Transfer II.	● 高 线	● 部署成功	2019-11-13 15:38:53	查看 删除 软件安装	

参数说明

参数	描述
授权用户UID	获取被授权用户的阿里云账号ID,填入此处。
授权实例	选择允许该阿里云账号ID的用户进行远程运维操作的边缘实例。

使用systemd管理Link IoT Edge

• 配置开机自启动功能。

```
wget http://remote-access-oxs.oss-cn-shanghai.aliyuncs.com/%E8%84%9A%E6%9C%AC/LinkIoTEdge
Lite.service
sudo cp LinkIoTEdgeLite.service /etc/systemd/system/LinkIoTEdgeLite.service
sudo systemctl enable LinkIoTEdgeLite.service
```

• 启动Link IoT Edge。

sudo systemctl start LinkIoTEdgeLite.service

• 重启Link IoT Edge。

sudo systemctl restart LinkIoTEdgeLite.service

• 停止Link IoT Edge。

sudo systemctl stop LinkIoTEdgeLite.service

• 禁用开机自启动。

sudo systemctl disable LinkIoTEdgeLite.service

下一步

- 1. 实例详情页面选择设置页签,并在实例信息下,打开远程访问后的按钮。
- 2. 实例详情页面选择网关页签,在网关名称右侧的操作栏中单击远程SSH终端、更多远程服务或者分享 远程SSH终端,远程控制网关设备、分享远程控制权限或对网关设备上的文件进行管理。详细说明请参见远程服务访问。

2.设备接入 2.1. 设备接入简介

设备接入是Link IoT Edge提供的基础能力,设备接入模块在Link IoT Edge中称为驱动(driver)或设备接入驱动。所有连接到Link IoT Edge的设备都需要通过驱动实现接入。

设备接入驱动在Link IoT Edge框架位置如图所示。



设备接入驱动

一个完整的驱动(设备接入模块)由设备的连接管理、设备的数据(协议)转换和设备的数据与命令处理三 个模块组成。

● 连接管理

指设备与网关建立通信连接。Link loT Edge不限制建立通信连接的协议,您可根据业务需求灵活选择。

● 数据转换

指设备接入驱动将获取到的终端设备数据转换为符合阿里云IoT物模型规范的数据格式,并上报到阿里云 IoT Cloud。阿里云物联网平台物模型规范请参考物模型。

• 数据与命令处理

指驱动可以处理云端对于设备的操作请求,并完成对设备的服务调用和处理调用结果,最终将结果返回到 阿里云物联网平台。 设备接入驱动是Link IoT Edge中独立的服务模块,您可以根据业务协议需求开发自定义设备接入驱动。下图 展示了自定义驱动的功能和数据流向,并指出了开发一个自定义驱动需要做的开发工作。



驱动使用介绍

Link IoT Edge针对不同用户需求提供了两种形态的驱动。

- 一是由Link IoT Edge提供常用协议驱动(又称官方驱动),如Modbus、OPC UA驱动。
- 二是提供设备接入驱动开发SDK,设备厂商或开发者使用驱动开发SDK开发私有协议驱动(又称自定义驱动)。



您可通过示例驱动章节提供的两个官方示例驱动快速熟悉驱动使用流程。

Link IoT Edge提供官方驱动,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。 您可以参照<mark>官方驱动</mark>章节内容,使用官网驱动做设备接入。 Link IoT Edge也提供设备接入SDK,您可以使用设备接入SDK,完成自定义协议驱动的开发。关于自定义驱动开发的详细介绍,请参考驱动开发。

2.2. 示例驱动

Link loT Edge提供Light和LightSensor两个示例驱动,通过这两个示例驱动,可帮助您快速熟悉驱动的使用流程。

Light驱动是Link IoT Edge提供的一款智能灯的模拟驱动。Light驱动支持智能灯的开关操作,可以用于模拟灯设备。

Light Sensor驱动是Link IoT Edge提供的一款光照度传感器的模拟驱动,光照强度数据按照一定周期重复,可以用于模拟光照度传感器设备。

前提条件

已创建边缘实例并上线网关。具体操作,请参见环境搭建。

示例驱动使用方法

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击边缘实例,在已创建的边缘实例对应操作列单击查看。
- 3. 分配驱动和子设备到边缘实例。
 - i. 在**实例详情**页面**设备与驱动**页签,单击**全部驱动**右侧的 + 图标,选择**官方驱动**并分配Light驱 动到边缘实例。

边缘计算 / 边缘实例 / 实例详情		分配驱动					×
		官方驱动 💛 Light		Q 😣			C
CPU使用率 1.6% 査督		驱动名称	驱动语言	通信协议类型	CPU架构	操作	
网关 监控信息 设备与驱动 场景联动 边缘应用 流数据分析 消息路由	日志朋	LightSensor 官方示例	Node.js 8	自定义	-	分配	
✓ ±2000 +		Light 官方示例	Node.js 8	自定义	-	分配	
暂无驱动分配							
		关闭					**

 ii. (可选)在Light驱动页面,单击设备列表右侧的驱动配置,可对当前驱动进行配置。配置完成后 单击确定。

本示例可以不进行驱动配置。



边缘计算 / 边缘实例 / 实例消情	驱动配置	×
← LinkloTEdge_Node 部第総約 P回時期 至3歳 CPU 使用単 2.02 % 重要 网关 监控信息 投稿与解約 指数解約 边缘应用 消数据分析 陽電路由 日本服务	 記述何式 記述何式 違述り記述 投資 提高り完選末 ● 投資 日本のため 工 	~
◆ 全部取功 Light + 23名列4 <u>認知定</u> 詳細定用 236577.05 7942年後入7 全部24日 ● 32日4日第 - 32日4日4日 - 32日4日第 - 32日4日1日 - 32日41日1日 - 32日4日1日 - 32日41日1日 - 32日411日1日 - 32日411日1日 - 32日411日 - 32111日 - 32111日 - 32111日 - 32111日 - 321111日 - 32111日 - 321111日 - 321111日 - 321111日 - 321111日 - 3211111 - 32111111 - 32111111 - 32111111 - 32111111 - 32111111 - 321111111 - 321111111 - 32111111111 - 32111111111111111111111111111111111111	 ※高校会 ※高校会 ※高大協会 ※高大協会 ※高大協会 ※高大協会 ※高大協会 ※高大協会 	
	確定 取消	**

驱动配置参数说明

参数	描述
配置格式	配置格式有如下三种: 键值对配置 JSON格式 配置文件
键值对配置表	配置格式为键值对配置时显示的参数。 单击 新增配置 ,通过设置配置名称、值、注释来配置驱动。 最多添加100个键值对。
配置内容	配置格式为JSON格式时显示的参数。 编写JSON格式配置内容,内容不超过1 KB。编写内容时,系统自动校验格式。 若格式错误,可根据系统提示编写正确格式的内容。
上传配置文件	配置格式为配置文件时显示的参数。 需要在本地编写配置文件并上传。配置文件的编写,请参见 <mark>驱动编码</mark> 。

iii. (可选)在**设备列**表右侧单击容器配置,对当前驱动进行容器配置。配置完成后单击保存。

⑦ 说明 仅在产品规格为专业版的边缘实例中,允许设置容器配置。

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 是:表示不隔离容器的网络,直接使用宿主机网络环境。 否:表示隔离容器的网络,需要设置网络端口映射。
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境 是完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某 一个端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多 支持映射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提 供服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问 到 fc-http-server 函数。因此需要将 fc-http-server 函数所在的容 器端口映射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机 8080端口),允许其它主机上的客户端程序通过访问 宿主机的IP地址: 8080 ,访问到容器内部的 fc-http-server 函数。
是否启动特权模式	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。 ⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔 离的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运 行函数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时, 需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 当容器内存使用超过限制时,容器会被重启。 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

iv. 单击**分配子设备**,在Light驱动下为边缘实例分配子设备。

✓ 全部驱动 · Light	+	设备列表驱动配	置 容器配置					
ugn		分配子设备	了解设备接入?	全部产品	➤ 设备名:	<i>х</i>	Q	
		设备名称		所属产品	格式配置	设备状态		操作
					Ğ	尚未添加任何子	设备	
						loT Edge 可连接位于i	边缘的设备, 了解设备接	并且提供了完整边缘设备接入工具。 多入



分配子设备			×
· · · · ·	设备名称	Q	全部 未分配
设备名称	所属产品	状态	操作
No. Information	10000	- 195	-
分配关闭		有	新设备? 添加子设备

v. 在添加设备对话框中, 单击新建产品, 创建客厅灯产品。

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
~
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置 DeviceName
请输入DeviceName
确定取消

在创建产品对话框中设置参数后,单击完成。

创建产品	×
产品信息	
* 产品名称	
客厅灯	
* 所属品类 💿	
● 标准品类 ○ 自定义品类	
智能生活 / 电工照明 / 灯 くうちょう くうちょう くうちょう くうちょう くうちょう しょうしょう くうちょう しょうしょう しょう	查看功能
连网与数据	
* 接入网关协议	
自定义 🗸	
✔认证方式	
更多信息	
✔产品描述	
使用文档	完成取消

参数说明

参数	描述
产品名称	此处设置为客厅灯。
所属品类	选择品类,为该产品定义物模型。此处选择 智能生活 > 电工照明 > 灯 。
接入网关协议	此处选择自定义。
认证方式	选择适合您设备的认证方式。详细信息,请参见 <mark>设备安全认证</mark> 。
vi. 在**添加设备**对话框中,产品自动分配已创建的客厅灯产品,DeviceName输入Light后单击**确定**,添加一个客厅灯设备。

添加i	设备	×
0	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		
客厅	ī火J	\checkmark
产品中 务操作 Device	可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者 。如需配置此产品功能清: <mark>前往配置</mark> Name	馜
Ligł	ht	
	确定取	消

vii. 在分配子设备面板,分配客厅灯产品下的Light设备到边缘实例中。

分配子设备			×
客厅灯 🗸	设备名称	Q	全部 未分配
设备名称	所属产品	状态	操作
Light	客厅灯	● 未激活	分配
分配 关闭		有新	设备? 添加子设备

viii. 参考分配Light驱动和客厅灯产品及其设备的步骤,分配LightSensor官方示例驱动到边缘实例,并 在该驱动下创建光照度传感器产品和名称为LightSensor的设备。

其中,创建光照度传感器产品的参数设置如下所示。

参数	描述
产品名称	设置为光照度传感器。
所属品类	此处选择智 能生活 > 家居安防 > 光照度传感器 。

至此您已成功分配LightSensor设备和Light设备到边缘实例。

ix. (可选)分配设备到边缘实例后,单击设备名称对应操作列中的设备配置,可对当前设备进行配置,例如配置设备序列号,方便驱动处理指定序列号设备的相关信息。

本示例可以不进行设备配置。

▶ 全部驱动	+	设备列表 驱动配置 容器配置				
Light		分配子设备 了解设备接入?	全部产品	∨ 设备名称		Q
		设备名称	所属产品	格式配置	设备状态	操作
		Light	客厅灯	未配置	● 未激活	设备配置 查看 移除

编写JSON格式配置内容,内容大小不超过1 KB。单击校验格式,通过校验后单击确定完成配置。

4. 部署边缘实例。

在实例详情页面,单击右上角的部署后在弹出对话框中单击确定,部署边缘实例。

您可以通过部署进程来查看部署进度及结果。

5. 查看设备上线状态。

部署实例后,在**设备与驱动**页签单击**全部驱动**,您可以看到**设备列表**下的Light设备和LightSensor设 备状态已显示在线。

✓ 全部驱动 +	设备列表				
Light	最大分配数 1000 当前分配数 2	● 未激活 0	● 已禁用 0	● 在线 2 🔰 🛛 ● 离线	ê 0
LightSensor	全部产品 🗸 设备名称		Q		
	设备名称	所属产品	驱动名称	设备状态 🔽	操作
	LightSensor	光照度传感器	LightSensor	● 在线	设备配置 查看 移除
	Light	客厅灯	Light	● 在线	设备配置 查看 移除

6. 查看设备运行状态。

分别单击LightSensor和Light设备右侧操作列中的查看,系统跳转到物联网平台控制台的**设备详情**页面,选择**物模型数据 > 运行状态**,看到设备数据已经发送至云端。

LightSensor设备运行状态如下图所示。

	← Ligh	tSenso	在线							
	产品 ProductKey	光照度传感	器 查看 复制						DeviceSecret	***** 查看
	设备信息	Topic 列表	物模型数据	设备影子	文件管理	日志服务	在线调试	分组		
	运行状态	事件管理	服务调用							
	请输入模块名	称 Q	请输入属性名	称或标识符		Q				
<	默认模块		光照度检测 100 Lux 2020/06/12	值 x ① 18:26:09.296		Ē	昏数据			

Light设备运行状态如下图所示。

	← Ligh	t 在线								
	产品 ProductKey	客厅灯 查祥	雪复制						DeviceSecret	******* 查看
	设备信息	Topic 列表	物模型数据	设备影子	文件管理	日志服务	在线调试	分组		
	运行状态	事件管理	服务调用							
	请输入模块名	称 Q	请输入属性名	称或标识符	(Q				
~	默认模块		主灯开关 1 (开 月 2020/06/12	1 7:07:33.664		査	看数据			

至此,您已经完成通过示例驱动接入设备的流程。如果您想进一步熟悉官方驱动和驱动开发,请参见官方驱动和驱动开发章节。如果您想了解更多业务逻辑的开发和运行框架,请参见边缘应用、场景联动和流数据分析。

2.3. 官方驱动

2.3.1. Modbus驱动

Link IoT Edge提供Modbus官方驱动,用于支持工业领域广泛应用的Modbus通信协议设备。本文主要介绍 Modbus驱动及其用法。

概览

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驱动可以直接连接Modbus从设备,示意图如下所示。



Modbus驱动也可以通过Modbus网关连接Modbus从设备,示意图如下所示。



Modbus驱动支持的功能有读取输入状态和输入寄存器、读取和写入线圈状态、保持寄存器。

Link IoT Edge提供C和Python语言Modbus驱动,同时根据CPU架构的不同,有多种C语言Modbus驱动。您可以直接从控制台部署Modbus驱动到网关,也可以从控制台下载Modbus驱动代码进行修改,作为您的自定义驱动使用。

本文以示例方式介绍Modbus官方驱动的使用步骤,具体操作如下文所示。

前提条件

请确保已创建边缘实例并上线网关,具体操作请参见<mark>环境搭建</mark>。

一、分配驱动

- 1. 登录边缘计算控制台。
- 2. 左侧导航栏单击边缘实例,在已创建的边缘实例右侧单击查看。
- 3. 在实例详情页面设备与驱动页签下,单击全部驱动右侧的 + 图标。
- 4. 在**分配驱动**面板中,选择**官方驱动**,根据网关CPU架构选择需要使用的Modbus驱动,单击对应操作栏中的**分配**。然后单击**关闭**。

? 说明

- 。 C语言Modbus驱动, 需在v1.8.4及以上版本的Link IoT Edge中使用。
- Python语言Modbus驱动, 仅支持在Link IoT Edge专业版中使用。

· 这次像计算 / 这是你来例 / 实例详细	分配驱动				×
	官方驱动 ~	Modbus	Q 🔕		C
CPU使用率 2.83% 查看	驱动名称	驱动语言	通信协议类型	CPU架构	操作
网关 监控信息 设备与驱动 场景联动 边缘应用 流数据分析 满意路由 日志制	Modbus 官方	с	modbus	x86-64	分配
★ 全部取消 +	Modbus 官方	с	modbus	aarch64	不可分配
前无意动分配	Modbus 官方	с	modbus	armv7-hf	不可分配
	Modbus 首方	с	modbus	armv7	不可分配
	Modbus 官方	Python 3.5	modbus		分配
	关闭				

二、配置驱动

- 1. 单击已分配的Modbus驱动,在**设备列表**右侧单击**驱动配置**。
- 2. 在弹出面板中单击添加通道。

通道是网关与具体物理设备之间的连接介质。

边缘计算 / 边缘实例 / 实例详情	3	枢动配置		×
← LinkloTEdge_Node ®™®®™ → States = 54.46	服务版本带	添加通道 通道名称	Q	C
CPU 使用率 2.83% 查查	内存使用率	通道名称 传输核	団 操作	
网关 监控信息 设备与驱动 场震联动 边缘应用 流数据分析 消息路由 日志服务	设置			
・	 2回 私 2次加任何子设备 dge 可法論位子辺論的 7所 	关 词	智无通道 文/即移动	B

3. 根据界面提示设置参数,然后单击确定。

参数	描述
通道名称	设置通道名称,需在网关维度具有唯一性。支持中文汉字、英文字母、数字和下划 线(_),长度不超过1~30个字符,1个中文汉字算2个字符。
传输模式	支持RTU、TCP和LoRa LAN三种。
当传输模式为RTU时,需设	置以下参数:
串口	设置串口,如 <i>/dev/ttyUSB0、/dev/ttyUSB1</i> 。支持英文字母、数字、正斜杠(/) 和下划线(_),长度限制1~64字符。
波特率	表示每秒传送的符号个数,从下拉列表中选择。

参数	描述		
数据位	表示一组数据实际包含的数据位数,从下拉列表中选择。		
校验位	从下拉列表中选择奇偶校验、或者无校验。		
停止位	用于表示单个包的最后一位,从下拉列表中选择。		
当传输模式为TCP时,需设	置以下参数:		
IP地址	Modbus设备IP地址,输入点分十进制格式的地址。		
端口号	Modbus设备端口号,输入1~65535范围的整数。		
当传输模式为LoRa LAN时,需设置以下参数:			
DevAddr	设备地址。请填写8位十六进制数值,例如 66be**** 。		
АррЅКеу	应用会话密钥。请填写32位十六进制数值,例如 623bd505f042090b5af660954 509**** 。		
NwkSKey	网络会话密钥。请填写32位十六进制数值,例如 e1336a94a03aa3beae55b737a cda**** 。		
Class	通信节点的特定类。当前仅支持C特定类,即随时接收设备上报的数据。		
上行 FPort	上行应用端口,取值范围为1~223。		
下行 FPort	下行应用端口,取值范围为1~223。		

(可选)在设备列表右侧,单击容器配置,根据如下参数说明,对当前驱动进行容器配置。配置完成后 单击保存。

? 说明 仅在产品规格为专业版的边缘实例中,允许设置容器配置。

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 • 是:表示不隔离容器的网络,直接使用宿主机网络环境。 • 否:表示隔离容器的网络,需要设置网络端口映射。

参数	描述
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境是 完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某一个 端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多支持映 射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提供 服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问到 f c-http-server 函数。因此需要将 fc-http-server 函数所在的容器端口映 射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机8080端口),允
	许其它主机上的客户端程序通过访问 <mark>宿主机的IP地址:</mark> 8080 ,访问到容器内部 的fc-http-server函数。
是否启动特权模式	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。
	⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所有 设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔离 的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运行函 数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时,需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 • 当容器内存使用超过限制时,容器会被重启。 • 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

三、驱动关联子设备

1. 单击**设备列表**区域框下的**分配子设备**,在Modbus驱动下为边缘实例分配设备。 您可以分配已有的Modbus设备,也可以根据下面的步骤,新建Modbus设备。

⑦ 说明 分配已有的Modbus设备时,该设备所属产品必须接入网关,且接入网关协议为 Modbus。详细说明请参见创建产品。

2. 在右侧弹出的分配子设备面板中,单击添加子设备。

分配子设备			×
· · · ·	设备名称	Q	全部 未分配
设备名称	所属产品	状态	操作
The state of the	10101080	- 195	-
分配 关闭		有新	设备? 添加子设备

3. 在**添加设备**对话框,单击**新建产品**,创建Modbus设备所属产品。

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
· · · · · · · · · · · · · · · · · · ·
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服务操作。如需配置此产品功能请:前往配置 DeviceName
请输入DeviceName
确定取消

4. 在创建产品对话框设置参数后,单击完成。

创建产品	×
产品信息	
* 产品名称	
Modbus产品	
连网与数据	
* 接入网关协议	
Modbus	
✔认证方式	
更多信息	
✔产品描述	
使用文档	完成取消

参数说明

参数	描述
产品名称	设置产品名称,产品名称在账号内具有唯一性。支持中文、英文字母、数字、下划 线(_)、短划线(-)、at符号(@)和英文圆括号,长度限制4~30个字符,一个 中文汉字算2个字符。
所属品类	选择品类,为该产品定义物模型。此处选择自定义品类。
接入网关协议	此处必须选择Modbus。
认证方式	选择适合您设备的认证方式。详细信息,请参见 <mark>设备安全认证</mark> 。
产品描述	添加对该产品的描述。可以为空。

5. 在添加设备对话框,产品自动分配已创建的产品,单击产品下的前往配置,为产品添加自定义功能。

② 说明 您也可以使用Modbus调试工具配置Modbus产品,但需要先完成添加设备并分配到边缘 实例的操作,再使用调试工具。

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
×
新建产品
产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服务操作。如需配置此产品功能请 <mark>前往配置</mark>
DeviceName
请输入DeviceName
确定取消

系统跳转到物联网平台控制台的产品详情页面功能定义页签。单击编辑草稿,在弹出页面中单击添加 自定义功能。

物联网平台 /	设备管理 / 产品	/ 产品详情						
← Moo	lbus产品							
ProductKey		复制					ProductSecret	******** 查看
设备数	1 前往管理							
产品信息	Topic类列表	功能定义	数据解析	服务端订阅	设备开发			
1 当前展示	的是已发布到线上的	功能定义,如需(^像 改,请点击 <mark>编</mark>	員草稿				
物模型 TSL	生成设备端代码	9						

6. 在添加自定义功能对话框,根据实际情况设置属性参数。具体操作请参见单个添加物模型。

在配置物模型属性的过程中,单击**新增扩展描述**设置如下**扩展描述**参数,将属性映射到寄存器中。官 方Modbus驱动会将所有的属性聚合为Modbus数据请求,驱动收到Modbus数据之后再转换为物模型数据。

操作类型			
保持寄存器 (读写 , 读取使用0	x03 , 写入	(使用0x06)	\sim
寄存器地址 ②			
请输入奇存器地址			
原始数据类型			
int16			\sim
取值范围 🔿			
-2147483648	~	2147483647	
交换寄存器内高低字节 💿			
false			\sim
交换寄存器顺序 💿			
false			\sim
缩放因子 💿			
1			
数据上报方式 📀			
按时上报			\sim

名称

描述

名称	描述
	指操作Modbus的功能码。
操作类型	 ⑦ 说明 其中: ● 线圈状态:读取时统一使用0x01功能码。写入时支持使用单个写入功能码0x05,或多个写入功能码0x0F。 ● 保持寄存器:读取时统一使用0x03功能码,写入时支持使用单个写入功能码0x06,或多个写入功能码0x10。 ● 写入时选择0x06:原始数据类型参数下只可选择int16或者uint16。选择其它数据类型时,上报属性数据会报错。 ● 写入时选择0x10:原始数据类型参数下所有数据类型初可选。 ● 离散输入:读取时统一使用0x02功能码,该类型不支持写入。 ● 输入寄存器:读取时统一使用0x04功能码,该类型不支持写入。 Modbus协议相关规范,请参见GB/T 19582.1-2008、GB/T 19582.2-2008和GB/T 19582.3-2008、
寄存器地址	填写十六进制,以 0x 开头。 根据您自己设备的属性地址设置寄存器地址。例如,要调试温度属性,您 设备的温度属性地址为1,则寄存器地址可设置为0x1。
原始数据类型	如采集的温度值的数据类型为浮点型。
取值范围	取值范围指的是原始数据经过缩放因子处理之后的取值范围,超出取值范 围的数据会被丢弃。
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。此处设置为互换true。
交换寄存器顺序	是否把原始数据32位数据的bits互换。此处设置为不互换false。
缩放因子	指缩放系数,如采集的值为100, 但真实的值为10,因此需要缩放10倍, 故缩放因子填写0.1即可。如放大10倍(即真实的值为1000),则放大因 子为10。

名称	描述
数据上报方式	 有两种数据上报方式。 按时上报:选择按时上报后,根据步骤9中为子设备设置的数据采集间隔,采集数据并上报。 变更上报:采集后的值发生变化后才会上报。

7. 返回**实例详情**页面,添加Modbus设备。

添加设备
1 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
Modbus产品 V
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置
DeviceName
device1
确定 取消

- 8. 将新建的Modbus设备分配到边缘实例。
- 分配设备到边缘实例后,单击设备名称对应操作栏中的设备配置,通过关联通道,关联设备与Modbus 驱动。

产品规格 专业税 CPU 使用率 2.76 9	版 [%] 查看		设备配置	X	⁻ 0 2 % <u>查</u> 看	
网关 监控信息 * 全調WG动	设备与驱动 + 谈	场景联动 边线	DeviceName: device1 所廣产品: Modbus产品			
Modbus		分配子设备 了解	* 关联通道]	
		设备名称	Channel 新建编辑	\checkmark	态	操作
		device1	* 从站号 💿		歉活	设备配置 调试 查看 移除
			2			
			数据采集间隔(ms) 📀			
			1000			
			確定	取消		

参数说明

参数	描述		
关联通道	选择 <mark>配置驱动</mark> 步骤2中已创建的通道。		
从站号	从站号是Modbus设备标识信息,在同一个通道中是唯一的。		
	Modbus协议是半双工协议,由网关主动请求数据,因此需要指定对数据点的采集间 隔时间。单位为毫秒。		
	⑦ 说明 单个属性点的采集耗时时间大概为60毫秒(ms),则采集间隔的 计算方式为:		
	采集耗时时间(60 ms) * 该通道的所有属性点个数		
数据采集间隔	例如,当前Modbus总线通道上有10个设备,且每个设备有10个属性点,即采 集间隔时间应大于等于6000 ms(60 ms * 10 * 10 = 6000 ms),这样才能保 证每个属性点能正常上报。		

四、部署边缘实例

- 1. (可选)在部署实例前,可以使用Modbus调试工具,测试网关能否连接该Modbus设备,同时也可以测 试Modbus设备所属产品的物模型是否配置正确。
- 2. 在实例详情页面,单击右上角部署,部署边缘实例。

常见问题

Modbus官方驱动上报消息到云端时,是按照设备维度上报,还是属性维度上报?

答: Modbus官方驱动在每个采集周期内,按照设备维度进行采集和上报数据。例如,1个Modbus设备有 100个属性,数据采集间隔为5秒,那么Modbus驱动每5秒上报云端1条消息(因为是1个设备)。

2.3.2. Modbus调试工具

Modbus调试工具用于测试网关能否连接指定的Modbus设备,同时也可以测试该Modbus设备所属产品的物 模型是否配置正确。

前提条件

- 请您确保已创建完成边缘实例,操作详情请参见环境搭建。
- 请您确保已为边缘实例分配Modbus驱动和子设备,操作详情请参见Modbus驱动。

操作步骤

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏中单击边缘实例。

找到已分配Modbus设备的实例,单击对应操作栏中的查看。

3. 在实例详情页面设备与驱动页签,单击全部驱动下的Modbus驱动,再单击设备名称右侧的调试。



⑦ 说明 若调试功能为灰色不可用时,请在实例详情页面右上角单击重置,重置边缘实例。

◇ 全部驱动	+	设备列表 驱动配置				
		分配子设备 了解设备接入?	全部产品 🗸	设备名称	Q	
Modbus		设备名称	所属产品	关联通道	设备状态	操作
		device1	Modbus产品	channel	● 未激活	设备配置 调试 查看 移除

4. 在通信通道调试页面,单击添加属性点,配置调试。

。 线圈配置参数说明如下所示。

← 通信通道调试				
通道: channel V 子设备: device1 V				
配置调试 调试日志				
绘图 寄存描				通道设置
·地址 💿 ·馬性名称	读取功能码	写入功能码	当前值	操作
	0x01 ~	无 ~	0	读取 写入 删除
+ 添加關性液				
经分配置 里亚				

线圈参数说明

参数	描述		
地址	线圈的地址。十六进制,以 0x 开头。 根据您自己设备的属性地址设置线圈地址。例如,要调试温度属性,您设备的温 度属性地址为1,则线圈地址可设置为 0x1 。		
属性名称	方便阅读的功能名称。 设置属性名称,并保存配置后,在该调试设备所属产品的 产品详情 页面 功能定 义页签自定义功能区域框中,功能名称一列中可看到已设置的属性名称。		
读取功能码	用于读取或输入该线圈状态的功能码。		
写入功能码	用于写入该线圈的功能码。 ⑦ 说明 读取功能码设置为 0x02 (输入状态)的线圈不支持此功能。		
当前值	显示被调试的Modbus设备所属产品物模型中设置的测量值,例如光照度、温度、 湿度等的当前值。		

参数	描述
	 读取:向网关发送读取测量值的指令。网关会尝试向该设备发送读取指令并将结果返回到当前值一栏。 写入:向网关发送写入测量值的指令。网关会尝试向该设备发送写入指令,并根据结果更新当前值一栏。
操作	 ⑦ 说明 读取功能码设置为 0×02 (输入状态)时不支持写入指 令。
	■ 删除:从设备所属产品的物模型中删除此线圈。
	注意 执行删除操作,并保存配置后,该操作会影响此产品下的所有 设备。

。 寄存器配置参数说明如下所示。

通道: channel 配置调试 调	∨ 子	设备: device1	~				
线圈 寄存器							通道设
*地址 🎯	* 居性名称	读取功能码	写入功能码	* 数据类型	单位	当前值	操作
		0x03 ~	无 ~	int16 V	无 ~	0	读取 写入 高级 删除
+ 添加属性点							
保存配置重	<u>27</u>						

寄存器参数说明

参数	描述	
地址	属性的起始地址。十六进制,以 0x 开头。	
	注意 添加多个配置项时寄存器地址不可重叠,若产生重叠,在保存配置时会提示配置异常。	
	根据您自己设备的属性地址设置寄存器地址。例如,要调试温度属性,您设备的 温度属性地址为1,则寄存器地址可设置为 0x1 。	
属性名称	方便阅读的功能名称。	
读取功能码	用于读取该寄存器的功能码。	

参数	描述			
写入功能码	用于写入该寄存器的功能码。			
	 ⑦ 说明 ● 读取功能码设置为 0x04 (输入寄存器)的寄存器不支持此功能。 ● 对于除int16和uint16之外的数据类型,如果想要写入保持寄存器,请选择 0x10 功能码。 			
数据类型	寄存器的数据类型,可选数据类型有int16、uint16、int32、uint32、int64、 uint64、float、double。不同数据类型使用的寄存器个数不同,如本文下方"各 数据类型寄存器个数说明"表格所示。			
单位	选择被调试Modbus设备的测量值单位。			
当前值	显示被调试Modbus设备的所属产品物模型中设置的测量值,例如光照度、温度、 湿度等的当前值。			
	 读取:向网关发送读取测量值的指令。网关会尝试向该设备发送读取指令并将结果返回到当前值一栏。 写入:向网关发送写入指令,输入十进制数字。网关会尝试向该设备发送写入指令,并根据结果更新当前值一栏。 			
操作	 ⑦ 说明 读取功能码设置为 0x04 (输入寄存器)时不支持写入指 令。 			
	 高级:为寄存器配置更多配置项,参数说明请参考本文下方"高级参数配置说明"表格。 删除:从设备所属产品的物模型中删除此寄存器。 			
	注意 执行删除操作,并保存配置后,该操作会影响此产品下的所有 设备。			

高级参数配置说明

参数	描述
----	----

参数	描述
	此开关控制解析寄存器内部的字节顺序。系统默认使用大端序(big-endian)的 解析方式。
	■ true: 使用小端序(little-endian)的解析方式。
	■ false:使用大端序的解析方式。
交换字节	示例1,当写入12345时,对应的十六进制值为0x3039,当该功能设置为false 时,发送/接收顺序为30 39;设置为true时,发送/接收顺序为39 30。
	示例2,若使用0x10功能码向slaveID为1的从站发送写入保持寄存器的RTU报文, 当该功能设置为false时,报文完整内容是:01 10 02 00 01 30 39 F7 E1;设置为 true时,报文完整内容是:01 10 02 00 01 39 30 33 14。
	此开关控制解析寄存器的顺序(仅在除int16、uint16的数据类型中有效)。系统 默认认为较高位的数值在较低地址的寄存器中,因此先解析较低地址的寄存器, 即先发送较高位的数据。
交换寄存器	例如,发送int64数据类型的数据:0x1234567890ABCDEF到设备端,会拆分成4 个寄存器发送(假设上文中交换字节设置为false),当该功能设置为false时,发 送顺序为12 34 56 78 90 AB CD EF;设置为true时,发送顺序为CD EF 90 AB 56 78 12 34。
	设置如何缩放设备的原始数据。通常情况下,浮点运算性能较差时会以整数型表 示浮点型,如果设备使用此方案,则可以使用缩放因子缩放设备的原始数据。
缩放因子	例如设备使用1234表示12.34时,缩放因子参数应当设置为0.01。此时所有向该设 备设置的数值,会被自动乘以100并发送到设备上,当云端或控制台设置12.34数 值时,驱动会自动将12.34转换为1234发送到设备上。反过来,设备发送的1234 也会被驱动转换为12.34发送到云端或其他设备上。

各数据类型寄存器个数说明

数据类型	暂用寄存器个数	描述
int 16	1	16位有符号整型
uint 16	1	16位无符号整型
int 32	2	32位有符号整型
uint 32	2	32位无符号整型
int 64	4	64位有符号整型
uint 64	4	64位无符号整型
float	2	单精度浮点型(32位)
double	4	双精度浮点型(64位)

5. 配置调试完成后,单击保存配置,将产品的功能定义更新到服务器中。

系统弹出保存成功的提示,表示数据已经保存到服务器中。

⑦ 说明 保存配置后,配置项所设置的内容会在该调试设备所属产品的产品详情页面功能定
 义页签的自定义功能区域框中,生成一条自定义功能,该条自定义功能中的标识符,是在保存配置后系统自动生成的标识符。生成原理如下所示。

标识符是物模型中ldentifier参数的值,ldentifier参数是由<mark>寄存器类型_4位十六进制地址</mark>该规范生成。例如,设置配置项中的读取功能码为01时,寄存器类型为coilStatus;设置地址为0x00,则4位十六进制地址是0000,因此ldentifier的值(即标识符)为coilStatus_0000。

其他操作

- 重置: 在配置调试页面左下方单击重置, 可重置当前已配置的内容
- 通道设置:在配置调试页面右上方单击通道设置,可修改该Modbus通信通道的参数,设置完成后需单 击确定保存修改。
- 调试日志:在通信通道调试页面单击调试日志,可查看调试日志信息;单击清屏可清除该页面日志信息。

功能码与寄存器/线圈类型对应表

类型	功能码	对应操作类型(operateType)
线圈(线圈状态)	0x01	coilState
输入状态(输入线圈)	0x02	inputState
保持寄存器	0x03	holdingRegister
输入寄存器	0x04	inputRegister

2.3.3. OPC UA驱动

Link IoT Edge产品提供用于接入OPC UA设备的驱动(简称OPC UA驱动)。您可以直接从控制台部署OPC UA 驱动到网关,也可以从控制台下载OPC UA驱动代码进行修改,作为您的自定义驱动使用。

OPC UA驱动目前仅支持在Link IoT Edge专业版(LE Pro)上运行。

概述

OPC UA驱动和OPC UA设备的连接是通过OPC UA服务器关联的,OPC UA驱动通过操作OPC UA服务器对外暴露的协议接口操作OPC UA设备,示意图如下所示。



本文以示例方式介绍OPC UA官方驱动的使用步骤,具体操作如下文所示。

前提条件

使用OPC UA官方驱动之前请确保已创建边缘实例并上线网关,操作详情请参见环境搭建。

一、分配驱动

- 1. 左侧导航栏单击边缘实例,在已创建的边缘实例右侧单击查看。
- 2. 在实例详情页面设备与驱动页签下,单击全部驱动右侧的 + 图标。
- 3. 在**分配驱动**对话框中,选择**官方驱动**,找到OPC UA驱动,单击对应操作栏中的分配。然后单击完成。

边缘计算 / 边缘实例 / 实例详情	分配驱动 ×
← LinkloTEdge_Node ®≣###	首方驱动 ∨ OPCUA Q ②
CPU使用率 2.56% 查看	驱动名称 驱动语言 通信协议类型 CPU架构 操作
网关 监控信息 设备与驱动 场景联动 边缘应用 流数据分	OPCUA 官方 Python 3.5 opc-ua - 分配
 ✓ 全部構成 ¥ ¥	
	žя

二、配置驱动

- 1. 单击已分配的OPC UA驱动,然后在设备列表右侧单击驱动配置。
- 2. 在弹出页面中单击添加通道。

边缘计算 / 边缘实例 / 实例详情				驱动配置				×
	de ^{部署成功}		Res	添加通道	通道名称		Q	C
CPU使用率 2.58% 查看			内存但	通道名称		通道地址	操作	Έ
网关 监控信息 设备与驱动	场景联动 边缘应用	流数据分析 消息路由	日志					
◇ 全部驱动 + 设	备列表 驱动配置 容器配置				1	11 11 11 11 11 11 11 11 11 11 11 11 11	首	
OPCUA	分配子设备 了解设备接入:	全部产品	设备名		1	立即添	加	
	设备名称	所属产品						
			前未添加 T Edge व 分配子i					
				关闭				

3. 根据界面提示设置参数,然后单击确定。

参数	描述
通道名称	需在网关维度具有唯一性。支持中文、英文字母、数字和下划线,长度限制1~30, 中文算2位。
通道地址	如 opc.tcp://localhost:4840 。
安全策略	指加密算法策略。有无、Basic128Rsa15、Basic256三种。
安全模式	指签名类型。有无、Sign、SignAndEncrypt三种。
用户名	非必填。
密码	非必填。
方法调用超时时间	单位为秒。

4. (可选)在设备列表右侧,单击**容器配置**,根据如下参数说明,对当前驱动进行容器配置。配置完成后 单击**保存**。

⑦ 说明 仅在产品规格为专业版的边缘实例中,允许设置容器配置。

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 • 是:表示不隔离容器的网络,直接使用宿主机网络环境。 • 否:表示隔离容器的网络,需要设置网络端口映射。

参数	描述
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境是 完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某一个 端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多支持映 射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提供 服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问到 f c-http-server 函数。因此需要将 fc-http-server 函数所在的容器端口映 射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机8080端口),允 许其它主机上的客户端程序通过访问 宿主机的IP地址:8080 ,访问到容器内部 的 fc-http-server 函数。
是否启动特权模式	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。 ⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔离 的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运行函 数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时,需 要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 • 当容器内存使用超过限制时,容器会被重启。 • 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

三、驱动关联子设备

1. 单击设备列表区域框下的分配子设备,在OPC UA驱动下为边缘实例分配设备。 您可以分配已有的OPC UA设备,也可以根据下面的步骤,新建OPC UA设备。

② 说明 分配已有的OPC UA设备时,该设备所属产品必须接入网关,且接入网关协议为OPC UA。详细说明请参见创建产品。

2. 在右侧弹出的分配子设备面板中,单击添加子设备。

分配子设备			×
· · · ·	设备名称	Q	全部 未分配
设备名称	所属产品	状态	操作
-	1010100	- 181	
分配 关闭		有新述	设备? 添加子设备

3. 在**添加设备**对话框,单击**新建产品**,创建OPC UA设备所属产品。

添加设备
1 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
••••••
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置 DeviceName
请输入DeviceName
确定 取消

4. 在创建产品对话框设置参数后,单击完成。

创建产品	×
产品信息	
* 产品名称	
* 所属品类 🔞	
○ 标准品类 ● 自定义品类	
连网与数据	
* 接入网关协议	
OPC UA 🗸	
◇认证方式	
更多信息	
▼产品描述	
	-
使用文档	完成取消

参数说明

参数	描述
产品名称	设置产品名称,产品名称在账号内具有唯一性。支持中文、英文字母、数字、下划 线(_)、短划线(-)、@符号和英文圆括号,长度限制4~30,一个中文汉字算2 位。
所属分类	选择品类,为该产品定义物模型。此处选择自定义品类。
接入网关协议	此处必须选择OPC UA。
认证方式	选择适合您设备的认证方式。详情请参见 <mark>设备安全认证</mark> 。

5. 返回添加设备对话框,添加OPC UA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
OPCUA产品 ~
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置 DeviceName
device1
确定取消

- 6. 在**分配子设备**页面,将新建的OPC UA设备分配到边缘实例。
- 7. 分配设备到边缘实例后,单击设备名称对应操作栏中的**设备配置**,通过关联通道,关联设备与OPC UA 驱动。

CPU 使用率 1.56 9	《 查看		设备配置	× ⁶¹	% 查看	
网关 监控信息 全部驱动	设备与驱动 +	动 场景 设备列表	DeviceName: device1 所屬产品: OPCUA产品			
OPCUA		分配子	* 关联通道		Q	
		— 送	opcuaserver	~	设备状态	操作
		de de	新建 编辑		● 未激活	设备配置 查看 移除
			例如: Objects/Device1			
				确定取消		

参数说明

参数	描述		
关联通道	选择二、配置驱动的步骤3中添加好的通道。		
节点路径	设备在OPC UA Server中,从Objects开始到设备节点的绝对路径。		
	⑦ 说明 OPC UA Server相关说明请参见OPC UA设备接入实践。		

四、部署边缘实例

在实例详情页面,单击右上角部署,重新部署边缘实例。

2.3.4. WebSocket驱动

Link IoT Edge提供用于接入WebSocket通信协议设备的驱动。根据网关设备使用架构的不同, Link IoT Edge 提供多种WebSocket驱动。您可以直接从控制台部署WebSocket驱动到网关。

使用WebSocket驱动接入设备时需要遵循Link IoT Egde的WebSocket接入协议。详情请参见物联网边缘计算 设备接入协议(WebSocket版)。

WebSocket 驱动可以直接接受设备的接入,示意图如下所示。



WebSocket驱动也可以通过网关接受设备的接入,示意图如下所示。



本文以示例方式介绍WebSocket官方驱动的使用步骤,具体操作如下文所示。

前提条件

请确保已创建边缘实例并上线网关,具体操作请参见环境搭建。

一、分配驱动

- 1. 登录边缘计算控制台。
- 2. 左侧导航栏单击边缘实例,在已创建的边缘实例右侧单击查看。
- 3. 在实例详情页面,选择设备与驱动右侧的 + 图标。
- 4. 在**分配驱动**面板,选择**官方驱动**并根据网关CPU架构选择需要使用的WebSocket驱动,单击对应操作 栏中的**分配**。然后单击关闭。

? 说明 WebSocket 驱动, 仅支持在v2.0.0及以上版本的Link loT Edge中使用。

边缘计算 / 边缘实例 / 实例评情	分配驱动				×
← LinkloTEdge_Node 認識的	官方驱动	WebSocket	Q 🕲		C
CPU 使用率 1.95% 査者	驱动名称	驱动语言	通信协议类型	CPU架构	操作
网关 监控信息 设备与驱动 场景联动 边缘应用 流数据分	WebSocket 首方	С	websocket	x86-64	分配
✓ 全部販助 +	WebSocket 首方	С	websocket	aarch64	不可分配
電元駆动分配	WebSocket 首方	С	websocket	armv7-hf	不可分配
	WebSocket 官方	С	websocket	armv7	不可分配
	关闭				

二、配置驱动

1. 单击已分配的WebSocket驱动,在设备列表右侧单击驱动配置。

根据界面提示设置参数,然后单击**确定**。

产品规格 专业版 CPU 使用率 1.9 % 查看		驱动配置	× 0 2% _{查看}
网关 监控信息 设备与驱动	场景联动 边线	* IP地址	-
✓ 全部販売 + 前 WebSocket	设备列表 驱动配置 容器 分配子设备 了算	* 端口号 17682	
	2 设备名称	* 是否加密 🔘 ● 是 〇 否	操作
		* 公钥文件 选择文件	,并且提供了完整边缘设备接入工具。
		* 私钥文件 选择文件	<u>意</u> 入
-		確定 取消	ğ

参数说明

参数	描述
IP地址	WebSocket驱动监听地址,默认为驱动所在机器的IP地址。
端口号	WebSocket驱动监听端口,范围为1~65535,默认端口为17682。
是否加密	 是否使用TLS加密,默认不加密。若选择是,需要上传公钥和私钥文件。 公钥文件:上传公钥证书文件,证书文件格式必须为.cer,名称不支持中文字符。 私钥文件:上传私钥文件,文件格式必须为.pvk,名称不支持中文字符。

2. (可选)在设备列表右侧,单击容器配置,根据如下参数说明,对当前驱动进行容器配置。配置完成后 单击**保存**。 ⑦ 说明 仅在产品规格为专业版的边缘实例中,允许设置容器配置。

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 • 是:表示不隔离容器的网络,直接使用宿主机网络环境。 • 否:表示隔离容器的网络,需要设置网络端口映射。
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境是 完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某一个 端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多支持映 射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提供 服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问到 f c-http-server 函数。因此需要将 fc-http-server 函数所在的容器端口映 射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机8080端口),允 许其它主机上的客户端程序通过访问 宿主机的IP地址: 8080 ,访问到容器内部 的 fc-http-server 函数。
是否启动特权模式	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。 ⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运行函数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时,需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 • 当容器内存使用超过限制时,容器会被重启。 • 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

三、驱动关联子设备

- 1. 单击**设备列表**区域框下的**分配子设备**,在WebSocket驱动下为边缘实例分配设备。 您可以分配已有的WebSocket设备,也可以根据下面的步骤,新建WebSocket设备。
- 2. 在右侧弹出的分配子设备面板中,单击添加子设备。

分配子设备			×
· · · ·	设备名称	Q	全部 未分配
设备名称	所属产品	状态	操作
- March 1997	1010100	- 185	-
分配关闭		有新	设备? 添加子设备

3. 在**添加设备**对话框,单击**新建产品**,创建WebSocket设备所属产品。

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
~
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置 DeviceName
请输入DeviceName
确定取消

4. 在创建产品对话框设置参数后,单击完成。

创建产品	×
产品信息	
* 产品名称	
请输入您的产品名称	
* 所属品类 💿	
● 标准品类 ○ 自定义品类	
请选择标准品类	查看功能
连网与数据	
* 接入网关协议	
自定义	
◇认证方式	
更多信息	
✓产品描述	
使用文档	完成取消

参数说明

参数	描述
产品名称	设置产品名称,产品名称在账号内具有唯一性。支持中文、英文字母、数字、下划 线(_)、连接号(-)、@符号和英文圆括号,长度限制4~30,一个中文汉字算2 位。
所属分类	选择品类,为该产品定义物模型。此处选择自定义品类。
接入网关协议	此处必须选择自定义。
认证方式	选择适合您设备的认证方式。详情请参见 <mark>设备安全认证</mark> 。
产品描述	添加对该产品的描述,可以为空。

- 5. 在**添加设备**对话框,输入DeviceName,添加WebSocket设备。
- 6. 将新建的WebSocket设备分配到边缘实例。
- 7. 分配设备到边缘实例后,单击设备名称对应操作栏中的设备配置,关联设备与WebSocket驱动。

◇ 全部驱动	+	设备列表 驱动配置 容器	设备配置	\times	
WebSocket		分配子设备 了解	DeviceName: device1 所属产品: WebSocket产品]
		设备名称			操作
		device1	设备 IP		设备配置 查看 移除
			请输入设备 IP		
			確定即	衍肖	

设备IP: 接入设备或设备所在网关的IP。

⑦ 说明 如果配置了该参数,则Link IoT Edge会严格校验IP。仅允许来自该IP地址的指定设备接入。

四、部署边缘实例

在**实例详情**页面,单击右上角**部署**,重新部署边缘实例。

2.3.5. LinkDriver驱动

对于集成了阿里云设备接入Link SDK的设备,物联网边缘计算提供LinkDriver驱动,将此类设备连接到Link IoT Edge实现与云端通信。

LinkDriver驱动示意图如下所示。



前提条件

- 已下载并集成Link SDK到您的设备。具体操作,请参见设备接入Link SDK文档。
- 已创建边缘实例并上线网关。具体操作,请参见环境搭建。

一、修改Link SDK

使用LinkDriver驱动之前,需要修改设备上的Link SDK内容,使设备能够连接到Link IoT Edge所在的网关。 本文以最新版C语言Link SDK 4.x版本为例,说明修改方法。其它语言Link SDK,请参考C语言SDK说明修改。

- 1. 打开C语言Link SDK的*demos/mqtt_basic_demo.c*文件。
- 2. 按照如下图红框所示的说明,修改接入MQTT的参数值。

↓ 注意 其中,将ⅠP地址 192.168.56.102 替换为您自己的网关ⅠP地址,其余参数设置与图片内容保持一致。

174	/* 创建1个MQTT客户端实例并内部初始化默认参数 */
	<pre>mqtt_handle = aiot_mqtt_init();</pre>
	if (mqtt_handle == NULL) {
177	<pre>printf("aiot_mqtt_init failed\n");</pre>
178	return -1;
179	
180	
181	_/* TODO: 如果以下代码不被注释, 则例程会用TCP而不是TLS连接云平台 */
182	// LinkDriver: 连接到Link IoT Edge网关修改项目:
183	// 1. 打开下面代码注释
184	
185	<pre>memset(&cred, 0, sizeof(aiot_sysdep_network_cred_t));</pre>
186	<pre>cred.option = AIOT_SYSDEP_NETWORK_CRED_NONE;</pre>
187	
188	
189	if (1 == public_instance) {
190	// LinkDriver: 连接到Link IoT Edge网关修改项目:
191	// 1. 注释下方 snprintf 一行
192	<pre>// snprintf(host, 100, "%s.%s", product_key, url);</pre>
193	// 2. 新增snprintf, 修改host为安装 Link IoT Edge 网关的IP地址, 如: 192.168.56.102
194	snprintf(host, 100, "%s", "192.168.56.102");
195	port = 1883;
196	} else 🛛
197	<pre>snprintf(host, 100, "%s", url);</pre>
198	

3. 保存修改的内容,然后重新编译Link SDK并更新到设备上。详细操作,请参见Link SDK文档。

二、分配驱动

- 1. 登录边缘计算控制台。
- 2. 左侧导航栏单击边缘实例,在已创建的边缘实例右侧单击查看。
- 3. 在实例详情页面设备与驱动页签下,单击全部驱动右侧的 + 图标。
- 4. 在**分配驱动**面板中,选择**官方驱动**,根据网关CPU架构选择需要使用的LinkDriver驱动,单击右侧操作 栏中的**分配**。然后单击确定。

边缘计算 / 边缘实例 / 实例详情	分配驱动				×
← LinkloTEdge_Node 部語版功	官方驱动 🗸	LinkDriver	Q 8		C
CPU 使用率 - 查看	驱动名称	驱动语言	通信协议类型	CPU架构	操作
网关 监控信息 设备与驱动 场景联动 边缘应用 消息路由	LinkDriver 官方	С	自定义	x86-64	分配
✓ 全部驱動 +	LinkDriver 盲方	C	自定义	ARMv8_64	不可分配
管无驱动分裂	LinkDriver 官方	C	自定义	ARMv7 VFPv3	不可分配
<.	LinkDriver 官方	C	自定义	ARMv7	不可分配
	确定取消				

三、驱动关联子设备

- 1. 单击已分配的LinkDriver驱动,在设备列表区域中单击分配子设备。
- 2. 在右侧弹出的分配子设备面板中,单击添加子设备。

分配子设备			×
· · · ·	设备名称	Q	全部 未分配
设备名称	所属产品	状态	操作
1000000	10001000		-
分配 关闭		有新	设备? 添加子设备

3. 在**添加设备**对话框,单击**新建产品**,创建Link SDK设备所属产品。

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					
· · · · · · · · · · · · · · · · · · ·					
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置					
DeviceName					
開報U/(Deviceivame) 确定 取消					

4. 在创建产品对话框设置参数后,单击完成。

参数说明

参数	描述
产品名称	设置产品名称,产品名称在账号内具有唯一性。支持中文、英文字母、数字、下划 线(_)、短划线(-)、at(@)和英文圆括号,长度限制4~30个字符,一个汉字 算2个字符。

参数	描述
所属品类	选择品类,为该产品定义物模型。此处选择自定义品类。
接入网关协议	此处选择自定义。
认证方式	选择适合您设备的认证方式。详细信息,请参见 <mark>设备安全认证</mark> 。
产品描述	添加对该产品的描述。可以为空。

5. 在添加设备对话框,产品自动分配已创建的产品,单击产品下的前往配置,为产品添加自定义功能。

添加设备 X
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
· · · · ·
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请 <mark>前往配置</mark> DeviceName
请输入DeviceName
确定取消

系统跳转到物联网平台控制台的产品详情页面功能定义页签。单击编辑草稿,在弹出页面的默认模 块中单击添加自定义功能。

- 6. 在**添加自定义功能**对话框,根据实际情况设置属性参数。具体操作,请参见单个添加物模型。
- 7. 返回**实例详情**页面,添加Link SDK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						
LinkSDK产品 V						
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置						
DeviceName						
device1						
确定 取消						

8. 将新建的Link SDK设备分配到边缘实例。

四、部署边缘实例

- 1. 在实例详情页面,右上角单击部署,部署边缘实例。部署成功后边缘实例名称后显示部署成功。
- 2. 查看设备运行状态。

实例部署成功后,可以查看设备连接状态和运行状态。这些信息可以在阿里云物联网平台(云端)查看,也可以在边缘网关运行环境中的驱动运行日志中查看。

i. 在**实例详情**页面设备与驱动页签,单击LinkDriver驱动,查看设备连接状态。

✓ 全部驱动 + LinkDriver	设备列表 驱动配置 容器配置					
		分配子设备 了解设备接入?	全部产品 >	设备名称		Q
		设备名称	所属产品	格式配置	设备状态	操作
		device1	LinkSDK产品	未配置	● 在线	设备配置 查看 移除

ii. 单击设备名称右侧操作栏中的查看, 跳转到物联网平台**设备详情**页面。

iii. 单击物模型数据页签下的运行状态,查看设备上报的数据。

- 3. 查看驱动运行数据和日志。
 - i. 登录Link IoT Edge所在网关。
 - ii. 进入LinkDriver驱动文件所在目录,查看驱动日志。

cd /linkedge/run/logger/fc-base/LinkDriver

2.4. 驱动开发

2.4.1. 概览

Link IoT Edge提供设备接入SDK(Link Edge Device Access,简称LEDA)方便您开发自己的驱动,SDK目前 支持C、Node.js、Java和Python四种版本的语言。本章节通过实现一个LED灯设备接入Link IoT Edge,介绍 自定义驱动开发流程。

设备接入SDK支持的四种开发语言详细内容,请参见C、Node.js、Java和Python。

背景信息

本驱动示例实现如下场景:通过自定义驱动,实现一个LED灯设备接入Link IoT Edge,您可以通过物联网平台控制台,控制LED灯设备的开关操作,也可以主动查询灯设备的当前温度。



在Link IoT Edge官方驱动中并没有可以完成该需求的驱动,因此需要在Link IoT Edge上开发LED设备驱动。 下面章节将基于C语言SDK,详细介绍LED设备驱动开发过程。

2.4.2. 驱动编码

在驱动开发过程中进行驱动编码时,需遵循物联网边缘计算的驱动编码规范和步骤。

驱动与设备信息配置

进行驱动编码前,需要了解Link IoT Edge的设备信息配置和驱动信息配置相关内容。

驱动信息配置

驱动信息配置在阿里云物联网平台进行配置。部署边缘实例时,驱动配置信息会被部署到边缘网关,其内容 以JSON格式存储在Link IoT Edge配置中心,可以通过leda_get_driver_info接口获取。

驱动信息配置有如下三种配置格式:

• 键值对配置

```
{
    "kv":[
        {
            "key":"ip",
            "value":"127.0.0.1",
            "note":"ip地址"
        },
        {
            "key":"port",
            "value":"54321",
            "note":"port端口"
        }
      ]
}
```
格式参数说明如下:

参数名称	说明
kv	驱动配置的格式为键值对配置。
key	键名称。
value	值内容。
note	键值注释。

● JSON格式

```
{
    "json":{
        "ip":"127.0.0.1",
        "port":54321
    }
}
```

格式参数说明如下:

参数名称	说明
json	驱动配置的格式为JSON格式配置。配置内容为自定义内容。

• 配置文件

```
{
    "fileList":[
        {
            "path":"device_config.json"
        }
    ]
}
```

格式参数说明如下:

参数名称	说明
fileList	驱动配置格式为配置文件列表。
path	配置文件路径。配置文件在驱动当前目录中。

设备信息配置

设备信息配置在阿里云物联网平台控制台配置。部署边缘实例时,设备信息配置会被部署到边缘网关,其内 容以JSON格式存储,可以通过leda_get_device_info接口获取。

设备信息配置格式定义:

{						
	"dev	iceList": [{				
		"custom": {				
		"ip":"12	7.0.0.1",			
		"port":22	2322			
		}, // 设备自定)	义配置			
		"productKey":	"xxxxxxxxx",	//	产品ProductKey,	在创建产品时生成
		"deviceName":	"demo_led",	//	设备DeviceName,	在创建设备时设置
	}]					
١						

设备信息配置参数说明:

配置名称	配置解释
deviceList	当前驱动下所有已进行设备配置的设备列表。
custom	设备自定义配置。
productKey	设备所在产品唯一标识符。
deviceName	设备名称。

操作步骤

1. 初始化驱动资源。调用leda_init接口完成资源初始化。

```
int main(int argc, char** argv)
{
    ...
    /* init driver */
    if (LE_SUCCESS != (ret = leda_init(WORKER_THREAD_NUMS)))
    {
        log_e(TAG_NAME_LED_DRIVER, "leda_init failed\n");
        return ret;
    }
    ...
    return LE_SUCCESS;
}
```

2. 解析驱动配置,完成设备上线。通过调用leda_get_driver_info接口获取驱动配置,解析设备的连接信息,并根据解析结果连接设备。设备连接成功后,调用leda_get_device_info接口获取设备配置并解析,根据解析结果验证设备功能。功能验证通过后,调

用leda_register_and_online_by_device_name接口完成设备注册并上线到阿里云物联网平台。

驱动信息配置格式定义请参见本文上方驱动与设备信息配置内容。

```
static int online_devices()
{
    ...
    /* 获取驱动和设备配置 */
    size = leda_get_device_info_size();
    if (size >0)
    {
```

```
device config = (char*)malloc(size);
       if (NULL == device config)
        {
           log e(TAG DEMO LED, "allocate memory failed\n");
           return LE ERROR INVAILD PARAM;
        }
        if (LE SUCCESS != (ret = leda get device info(device config, size)))
        {
           log e(TAG DEMO LED, "get device config failed\n");
           return ret;
        }
    }
    /* 解析驱动和设备配置 */
    devices = cJSON_Parse(device_config);
    if (NULL == devices)
    {
       log e(TAG DEMO LED, "device config parser failed\n");
       return LE ERROR INVAILD PARAM;
    }
    cJSON ArrayForEach(item, devices)
    {
       if (cJSON Object == item->type)
        {
           /* 解析配置内容 */
           result = cJSON GetObjectItem(item, "productKey");
           productKey = result->valuestring;
           result = cJSON_GetObjectItem(item, "deviceName");
           deviceName = result->valuestring;
           result = cJSON GetObjectItem(item, "custom");
           if (NULL != result)
            {
               log i(TAG DEMO LED, "custom content: %s\n", cJSON Print(result));
            }
           /* 注册并上线设备 */
                                            = get_properties_callback_cb;
= set_properties_callback_cb;
           device cb.get properties cb
           device_cb.set_properties_cb
                                                 = call service callback cb;
           device cb.call service cb
           device cb.service output max count
                                                 = 5;
           dev_handle = leda_register_and_online_by_device_name(productKey, deviceName
, &device cb, NULL);
           if (dev_handle < 0)
            {
               log e(TAG DEMO LED, "product:%s device:%s register failed\n", productKe
y, deviceName);
               continue;
            }
           g dev handle = dev handle;
           log i(TAG DEMO LED, "product:%s device:%s register success\n", productKey,
deviceName);
      }
   }
  . . .
  return LE SUCCESS;
}
```

3. 将收到的设备数据转换为阿里云IoT物模型格式并上报到物联网平台,调用leda_report_properties接口 上报设备属性数据,调用leda_report_event接口上报设备事件。

⑦ 说明 此处使用虚拟设备,直接构造物模型格式的数据进行上报。

```
/* 上报数据 */
while (1)
{
   /* 上报属性 */
   leda_device_data_t dev_proper_data[1] =
    {
        {
           .type = LEDA_TYPE_INT,
           .key = {"temperature"},
            .value = {0}
        }
    };
    sprintf(dev proper data[0].value, "%d", g dev temperature);
    leda_report_properties(g_dev_handle, dev_proper_data, 1);
    /* 上报事件 */
    if (g dev temperature > 50)
    {
       leda device data t dev event data[1] =
       {
            {
               .type = LEDA TYPE INT,
               .key = {"temperature"},
                .value = \{0\}
            }
       };
        sprintf(dev_event_data[0].value, "%d", g_dev_temperature);
       leda report event(g dev handle, "high temperature", dev event data, 1);
    }
    sleep(5);
}
```

4. 处理云端服务请求。实现服务请求的三个回调函数如下所示。

- get接口: 处理获取设备属性的请求。
- o set接口:处理设置设备属性的请求。
- 。 service接口:处理调用设备自定义方法的请求。

```
    ⑦ 说明 本示例使
    用get_properties_callback_cb、set_properties_callback_cb和call_service_callback_cb三个回调
    函数实现服务请求。
```

```
static int get properties callback cb(device handle t device handle,
                              leda device data t properties[],
                               int properties count,
                               void *usr data)
{
    . . .
  return ret;
}
static int set properties callback cb(device handle t device handle,
                              const leda device data t properties[],
                              int properties count,
                               void *usr data)
{
    . . .
   return ret;
}
static int call_service_callback cb(device handle t device handle,
                              const char *service name,
                              const leda device data t data[],
                               int data count,
                               leda device data t output data[],
                               void *usr_data)
{
    . . .
   return ret;
}
```

⑦ 说明 本示例的完整工程源码请参见Github源码库LED设备驱动。

至此, 您已完成驱动编码过程。

2.4.3. 驱动调试

设备接入驱动编码完成后,需进行驱动的调试。调试阶段包括:编译打包、上传驱动、搭建环境、创建物模型、创建设备、部署驱动、查看调试信息、调试数据交互和本地替换更新驱动。

前提条件

在开始操作本章内容前,请您确保已根据基于Ubuntu 16.04搭建标准版环境内容完成边缘实例的创建,为C语言 SDK开发的设备驱动准备运行环境。

一、编译打包

驱动打包相关要求,请参见本文下方驱动打包注意事项。

1. 下载示例代码。

git clone https://github.com/aliyun/linkedge-thing-access-sdk-c.git

2. 进入*linkedge-thing-access-sdk-c*目录,完成示例编译。

```
cd linkedge-thing-access-sdk-c
make prepare && make && make install
```

3. 查看示例编译产物。

cd build/bin/demo/demo_led/ && ls -l

系统显示类似如下图表示已完成编译并打包。

[root@	demo	_led]#ls -l				
total 1836						
-rw-rr	1 root	root	1876898	May 31	11:49	<pre>led_driver.zip</pre>
[root@	demo.	_led]#				

二、上传并发布驱动

更多操作详情及参数说明,请参见驱动发布。

- 1. 在边缘计算控制台左侧导航栏单击驱动管理,然后单击自研驱动页签。
- 在自研驱动页签下,单击新建驱动。
 参考云端发布内容,设置驱动参数。

驱动信息	
* 驱动名称 ②	
请输入驱动名称	
* 通信协议类型 📀	
请选择通信协议类型	þ
* 语言类型:	
请选择语言类型	,
 驱动是否内置 ② ● 是 ● 否 * 驱动文件 ② 上传文件 * 驱动版本 	
请输入驱动版本	
* 驱动适配的边缘版本 📀	
2.5.0版本及以上	*
驱动版本描述:	
请输入驱动版本描述	

本示例中,驱动是否内置需要选择否,然后在驱动文件下,上传已完成编译并打包的*led_driver.zip*文件。

3. 完成参数的设置并上传成功驱动文件后,单击确认。您可以在自研驱动页签中看到刚刚创建的驱动。

三、分配驱动和设备到边缘实例

部署驱动依赖边缘实例,即您在前提条件中完成的环境搭建操作。

- 1. 左侧导航栏单击边缘实例,在已创建的边缘实例右侧单击查看。
- 2. 在实例详情页面设备与驱动页签下,单击全部驱动右侧的 + 图标。
- 3. 在分配驱动面板中,选择自研驱动并在已创建的自定义驱动对应操作栏中单击分配。然后单击完成。

边缘计算 / 边缘实例 / 实例详情	分配驱动	×
← LinkloTEdge_Node ☆	田研記的 > 満能入脳的名称 Q	C
CPU 使用率 1.88 % 豐富 1.88 % 富 1.88 % п 1.88 % ша 1.88 % п 1.88 % ша 1.88 % п 1.88 % ша 1.88 %		分配
✓ 全部総約 +	demo_driver 自定义 x86-64 1.0.0	✓ 分配
	关闭	

- 4. 选择已分配的自定义驱动demo_driver,单击**分配子设备**,在demo_driver驱动下为边缘实例分配设备。
 - i. 在右侧弹出的**分配子设备**面板中,单击**添加子设备**。

分配子设备				×
· · · · ·	设备名称		Q	全部 未分配
设备名称	所属产品	:	状态	操作
-	10101080			
分配关闭			有新读	备? 添加子设备

ii. 在**添加设备**对话框, 单击**新建产品**, 创建一个产品。

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	
~	
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置	
DeviceName	
请输入DeviceName	
确定取消	

iii. 在**创建产品**对话框设置参数后,单击完成。

参数说明

参数	描述
产品名称	设置产品名称,产品名称在账号内具有唯一性。支持中文、英文字母、数字、下 划线(_)、短划线(-)、at符号(@)和英文圆括号,长度限制4~30个字符, 一个中文汉字算2个字符。
所属品类	选择品类,为该产品定义物模型。此处选择自定义品类。
接入网关协议	此处必须选择自定义。
认证方式	选择适合您设备的认证方式。详细信息,请参见 <mark>设备安全认证</mark> 。
产品描述	添加对该产品的描述。可以为空。

iv. 返回添加设备对话框,单击前往配置,为产品添加自定义功能。

添加设备	×
1 特别说明: DeviceName可以为空,当为空时,阿里云会颁发全局唯- 标识符作为DeviceName。	_
* 产品	
led_demo	\sim
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或 务操作。如需配置此产品功能请:前往配置 DeviceName	诸服
请输入DeviceName	
确定	取消

系统跳转到物联网平台控制台的**产品详情**页面,在**功能定义**页签下,参考单个添加物模型内容,为 产品添加属性和事件。 ■ 添加属性

添加自定义功能	×
* 功能类型 ② 属性 服务 事件	
* 功能名称 @	
temperature	
*标识符 📀	
temperature	
* 数据类型	
int32 (整数型)	\sim
* 取值范围	
-20 ~ 80	
* 步长	
1	
单位	
摄氏度 / ℃	\sim
* 读写类型	
描述	
请输入描述	
	0/100
确认	取消

■ 添加事件

添加自定义功能	×
* 功能类型 ② 属性 服务 事件	
* 功能名称 💿	
high_temperature	
* 标识符 💿	
high_temperature	
* 事件类型 💿 ● 信息 〇 告警 〇 故障	
输出参数 +增加参数	
	0/100
+新埠/ 展田还	
确认	取消

输出参数设置如下图所示。

新增参数		×
* 参数名称 💿		
temperature		
* 标识符 ②		
temperature		
* 数据类型		
int32 (整数型)		\sim
* 取值范围		
-20	~ 80	
* 步长		
1		
单位		
摄氏度 / ℃		\sim

v. 返回添加设备对话框,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
led_demo 🗸
产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服务操作。如需配置此产品功能请:前往配置 DeviceName
led_device
确定取消

vi. 在**分配子设备**对话框中,单击刚创建的设备右侧的**分配**,将设备分配到边缘实例。 5. 分配设备到边缘实例后,单击设备名称对应操作栏中的**设备配置**,配置设备信息。

设备配直		,
DeviceName: led_c	levice	
所属产品: led_demo		
配置内容 (JSON格式):		
1 • { 2 "ip":"12 3 "port":2	7.0.0.1", 2322	
4 }		
校验格式(当前内容	汏小: 0.04KB)	
确定 取消		

- 6. (可选)通信协议类型为LoRa WAN的驱动,分配子设备后需要进行ABP 入网配置。
 - i. 在关联LoRa WAN协议驱动的子设备对应操作栏中单击ABP 入网配置。

网关 监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置				
✔ 全部驱动	+ i	日本 11	置 容器配置								
test_driver		分配子设备	了解设备接入?	全部产品	~	设备名称		Q			
		设备名称		所屋产品		格式配置	设备	伏态	操作		
				测试设备		未配置	0 7	に激活	设备配置	ABP 入网配置	查看 移除

ii. 设置ABP入网配置参数。

② 说明 ABP (Activation By Personalization) 是LoRa WAN协议设备的一种入网认证方式。

参数说明

参数	描述
DevEUI	设备扩展的唯一标识符。请填写16位十六进制数值,例如 d896e0000900*** * 。
DevAddr	设备地址。请填写8位十六进制数值,例如 66be**** 。
АррЅКеу	应用会话密钥。请填写32位十六进制数值,例如 623bd505f042090b5af660 954509**** 。
NwkSKey	网络会话密钥。请填写32位十六进制数值,例如 e1336a94a03aa3beae55b7 37acda**** 。
Class	通信节点的特定类。有A、B和C三种特定类。
上行 FPort	上行应用端口,取值范围为1~223。
下行 FPort	下行应用端口,取值范围为1~223。

iii. 单击确定,完成配置。

四、部署边缘实例

- 1. 在**实例详情**页面,右上角单击部署,部署边缘实例。部署成功后边缘实例名称后显示部署成功。
- 2. 查看调试信息。

实例部署成功后,可以查看设备连接状态和运行状态。这些信息可以在阿里云物联网平台(云端)查看,也可以在边缘网关运行环境中的驱动运行日志中查看。

- 。 云端查看:
 - a. 在实例详情页面设备与驱动页签,选择自定义驱动demo_driver,查看设备连接状态。

✓ 全部図3カ +	设备列表 驱动配置 容器配置				
demo_driver	分配子设备 了解设备接入?	全部产品 >	设备名称		Q
	设备名称	所属产品	格式配置	设备状态	操作
	led_device	led_demo	JSON 格式	● 在线	设备配置 查看 移除

b. 单击设备名称右侧操作栏中的查看, 跳转到物联网平台设备详情页面。

c. 选择物模型数据 > 运行状态页签, 查看设备上报的数据。

物联网平台 / 设	遙管理 / 设备	备 / 设备详情						
← led_c	device	在线						
产品	led_demo	查看				De	viceSecret	******* 查看
ProductKey	1000	复制						
设备信息	Topic列表	物模型数据	设备影子	文件管理	日志服务	在线调试		
运行状态	事件管理	服务调用						
实时刷新 🔵								
temperature		查看数据						
30 ℃								
2019/12/20 7	下午2:09:40							

- 本地查看:
 - a. 通过 ./fctl show 命令,可以查看到部署驱动的具体位置。

cd /linkedge/gateway/build/bin/
./fctl show

系统显示类似如下图所示。

[root@bc bin]#./fctl show
FC run in [Process] Mode.
Show Function [1]
Configuration information:
Type: Driver
Executor: c
RunMode: LongLived
ServiceName: demo_driver
FunctionName: demo_driver
JobID: 297900
FunctionID: e8
Environments: {"FC_DRIVER_CONFIG":"{\"driverConfig\":\"\",\"serverList\":[{\"customJson\":\"{}\",\"serverId\":\"AC B29D\"}],\"deviceList\":[{
\"custom\":\"{}\",\"productKey\":\"\$``\",\"deviceName\":\"led_device\",\"serverId\":\"AC B29D\"}],\"driverType\":1}"}
DriverName: demo_driver
Deploy: Success
Process: Running
Process PID: 23159
Process MemoryUsage: 18548 KB
ProcessStartTime: 2018-12-28 15:15:41.683994
ProcessLastInvokeTime: 1970-01-01 08:00:00.000000
GroupID: 14
MemoryLimitSize: 1073741824 Bytes
TimeOut: 60
Category: 0
Version: 1545979754905
PackageType: zip
CodePath: /linkedge/run/downloaded-job/nodeengine/297900
ConfigFile: /linkedge/run/downloaded-job/nodeengine/297900/config.json
CheckSum: codeChecksumDefault
AccountID: accountId
RegionID: regionIdDefault
ForeignKey: driver#142091#e8: c93e
TriggerType:
Cron:
Handler: main handler

参数说明

字段名称	字段解释
DriverName	驱动名称,该名称为在二、上传并发布驱动步骤中设置的驱动名称。
CodePath	驱动部署到边缘网关中的位置路径。
Process PID	驱动启动后的进程ID。

b. 驱动在运行过程中会产生运行日志,通过查看运行日志可以了解驱动运行状态,每个自定义驱动 生成的日志文件统一放在/linkedge/run/logger/fc-base路径下。每个驱动日志文件路径为/link edge/run/logger/fc-base/xxxx/log_xxxx.txt。

⑦ 说明 驱动日志文件路径中的xxxx为在上传驱动时填写的名称。

cd /linkedge/run/logger/fc-base/demo driver && ls -l

详情如下图所示。

[root@bc	demo	_driver]#	owd			
/linkedge/ru	un/logger/fo	c-base/dem	o_driver			
[root@bc	demo	_driver]#				
[root@bc	demo	_driver]#	ls -l			
total 76						
lrwxrwxrwx	1 root	root	70	Dec 28	15:15	<pre>log.INFO -> /linkedge/run/logger/fc-base/demo_driver/log_2018-12-28-15-15-41-1.txt</pre>
-rw-rr	1 root	root	4134	Dec 28	15:09	log_2018-12-28-14-57-08-1.txt
-rw-rr	1 root	root	3571	Dec 28	15:13	log_2018-12-28-15-10-34-1.txt
-rw-rr	1 root	root	15283	Dec 28	15:15	log_2018-12-28-15-14-36-1.txt
-rw-rr	1 root	root	37830	Dec 28	15:20	log_2018-12-28-15-15-41-1.txt

驱动运行日志可以查看大部分驱动的运行状态,但是有时还需要查看其它日志配合了解当前驱动的运行情况,例如设备上线异常时可以查看 dimu 日志,日志文件路径为/linkedge/run/logge r/dimu/log_xxxxx.txt。

```
cd /linkedge/run/logger/dimu
ls -l
```

详情如下图所示。

[root@bc	dim	_] #pwd	
/ Linkeuge/ ru	un/ Logger / u		
Lroot@bc	dim	ג#	
[root@bc	dim.	⊿]#ls -l	
total 32			
lrwxrwxrwx	1 root	root	55 Dec 28 15:15 log.INFO -> /linkedge/run/logger/dimu/log_2018-12-28-15-15-40-1.txt
-rw-rr	1 root	root	1459 Dec 28 11:11 log_2018-12-28-11-10-26-1.txt
-rw-rr	1 root	root	3793 Dec 28 14:47 log_2018-12-28-14-47-04-1.txt
-rw-rr	1 root	root	3638 Dec 28 14:55 log_2018-12-28-14-47-57-1.txt
-rw-rr	1 root	root	3899 Dec 28 15:10 log_2018-12-28-14-57-07-1.txt
-rw-rr	1 root	root	3807 Dec 28 15:13 log_2018-12-28-15-10-33-1.txt
-rw-rr	1 root	root	5475 Dec 28 15:15 log_2018-12-28-15-14-35-1.txt
-rw-rr	1 root	root	3204 Dec 28 15:15 log 2018-12-28-15-15-40-1 +y+

设备上线成功后,如果设备数据上报有异常,则可以查看 cloud_proxy 日志。cloud_proxy 生的运行日志文件路径格式为/*linkedge/run/logger/cloud-proxy/log_xxxxx.txt*。

```
cd /linkedge/run/logger/cloud-proxy
ls -l
```

详情如下图所示。

[root@bc	clo	ud-proxy]#p	owd		
/linkedge/ru	un/logger/c	loud-proxy			
[root@bc	clo	ud-proxy]#			
[root@bc 📥	clo	ud-proxy]#]	.s -l		
total 1128					
lrwxrwxrwx	1 root	root	62 Dec 2	8 15:15	5 log.INFO -> /linkedge/run/logger/cloud-proxy/log_2018-12-28-15-15-40-1.txt
-rw-rr	1 root	root	7299 Dec 2	8 11:11	1 log_2018-12-28-11-10-25-1.txt
-rw-rr	1 root	root	30451 Dec 2	8 14:47	7 log_2018-12-28-14-47-04-1.txt
-rw-rr	1 root	root	165951 Dec 2	8 14:55	5 log_2018-12-28-14-47-57-1.txt
-rw-rr	1 root	root	472630 Dec 2	8 15:10	0 log_2018-12-28-14-57-07-1.txt
-rw-rr	1 root	root	136951 Dec 2	8 15:13	3 log_2018-12-28-15-10-33-1.txt
-rw-rr	1 root	root	76663 Dec 2	8 15:15	5 log_2018-12-28-15-14-35-1.txt
-rw-rr	1 root	root	247473 Dec 2	8 15:21	1 log_2018-12-28-15-15-40-1.txt

五、调试数据交互

如果设备上线成功,则可以使用阿里云物联网平台**在线调试**功能调试驱动和设备,该功能页面还可以实时查 看设备上报的数据信息,也可以触发对设备服务的调用请求。

- 1. 在物联网平台控制台,选择监控运维 > 在线调试,在在线调试页面,选择调试产品和调试设备。
- 2. 选择调试设备待调功能和服务,进行调试,并查看设备实时运行日志。

在线调试				
请选择设备: led_demo				
① 在线调试只支持调试真实设备,虚拟设备调试清使用设备模拟器 ×	实时日志 ● 在线	自动刷新 🌑 🕑 📥		
居性调试 服务调用 远程登录	时间	内容		
模块: 默认线块 >>	物模型服务调用 2021/09/26 18:22:20.594	("Status":"true","Instanceld":"iot-public","Params":"","Time":"2021-09-2618:22:20.594","O peration":"thing.service.property.set","Code":"200","Reason":","","UtcTime":"2021-09-2618: 22:20.594+0800","totid":",",",",",",",",",",",",",",",",",","		
temperature(temperature)		","ProductKey":"""""BizCode":"ThingServic e","DeviceName":"led_device","Messageld":""}		
30 调试 🗸				
获取 设置期望值 重置				

六、(可选)本地替换驱动

如果在驱动调试过程中发现问题,需要修改驱动代码重新生成新的驱动进行调试,这时只需要在本地替换修改编译后的驱动即可。

1. 找到驱动的位置。可以通过fctl命令进行查找,具体字段解释查看调试信息。

cd /lin ./fctl	<pre>ikedge/gateway/build/bin/ show</pre>
[root@bc FC run in [Proces:	<pre>bin]#./fctl show s] Mode.</pre>
Con Typ Exe Run Sem Fun Dei Dei Dei Dei Pro	<pre>show Function [1] ***********************************</pre>

- 2. 根据CodePath找到驱动所在位置路径,完成替换。
- 3. 根据Process PID得到驱动进程ID,使用kill -9 Pid命令重新启动驱动。

kill -9 Pid #Pid**为驱动进程**ID, 即通过fctl命令查看得到的Process PID

 替换驱动并完成驱动调试后,可参见本文上方二、上传并发布驱动步骤,将驱动发布到物联网平台控制 台。

至此,您已完成驱动的调试与发布过程。

驱动依赖注意事项

设备接入驱动根据协议和业务场景的不同,可能会涉及第三方库依赖。Link loT Edge针对开发设备接入驱动 所用不同开发语言,分别制定了第三方库依赖规则。

• C版本SDK:

C语言属于编译型语言,如果编译目标环境和运行环境不一致,则很可能导致不可运行。所以对于使用设备接入C版本SDK开发驱动,需要保证开发编译目标环境和运行环境相同。

驱动包中包含驱动程序和依赖动态库。如果该驱动依赖于第三方库,则需要将动态库和驱动程序一起打包 生成最终驱动程序包。

• Node.js版本SDK:

使用设备接入SDK Node.js版本开发驱动时,若依赖第三方库,需要到Link loT Edge运行环境上开发驱动, 并在驱动目录中使用如下命令安装依赖。

npm install **第三方库名**

• Python版本SDK:

使用设备接入SDK Python版本开发驱动时,若依赖第三方库,需要到Link IoT Edge运行环境上开发驱动, 并在驱动目录中使用如下命令安装依赖。

pip3 install -t . 第三方库名

驱动打包注意事项

基于Link IoT Edge提供的SDK开发驱动并完成调试后,需将产物打包为.*zip*包,并确保驱动Binary或index源 文件在.*zip*包第一级目录。

每个版本SDK开发的驱动在打包时,有不同的打包规则。

● 基于C SDK开发的驱动

对于C语言开发的驱动,驱动包中包含驱动程序和驱动依赖的动态库。如果驱动程序包含依赖库,则需要 将依赖库放置指定的位置,即在驱动程序当前路径下的lib文件夹下。具体操作步骤如下:

- i. 规定驱动程序需命名为main。
- ii. 在main当前路径下创建lib文件夹。
- iii. 将main依赖的动态库全部拷贝到lib文件夹下。
- iv. 使用zip命令对当前路径下的main和lib进行压缩处理生成zip包。

zip -r your driver name.zip main lib

● 基于Python SDK开发的驱动

驱动包文件中须包含*index.py*,并且在该文件中定义handler函数。驱动是一个在函数计算应用引擎中持续运行的函数,所以在驱动包中须包含*index.py*文件,并且在该文件中定义handler函数。

驱动运行时,会加载*index.py*文件。而该文件中,函数计算应用定义的handler函数是不会被调用,因此驱动代码须放在handler函数外,保证加载*index.py*文件时能直接执行。详情请参考Python版本SDK。

● 基于Node.js SDK开发的驱动

驱动包文件中须包含index.js,并且在该文件中定义handler函数。

驱动运行时,会加载*index.js*文件。而该文件中,函数计算应用定义的handler函数是不会被调用,因此驱 动代码须放在handler函数外,保证加载*index.js*文件时能直接执行。详情请参考Nodejs版本SDK。

2.4.4. 常见问题

本章主要描述,在驱动开发过程中遇到的问题及其解决方法。

• leda_register_and_online_by_device_name(const char *product_key, const char *device_name, leda device callback t *device cb, void *usr data) 接口的 device cb 作用是什么?

答:

用于接收来自云端或其他边缘应用对设备的服务请求调用。分别是请求获取设备属性、请求设置设备属性,请求调用设备服务。

```
typedef struct leda_device_callback
{
    ...
    int service_output_max_count; /* 设备服务回调结果数组最大长度 */
} leda device callback t;
```

字段 service output max count 表示什么意思?

答:

用于设置 call service cb 回调接口 output data 数组的最大长度。

 leda_init(const char *module_name, int worker_thread_nums) 接口的 worker_thread_nums 是用 来做什么的?

答:

用于初始化驱动的线程池。初始的线程池处理来自云端或边缘应用对设备的请求调用,具体大小可根据接 入设备量进行合理设置。

leda_register_and_online_by_device_name 和 leda_register_and_online_by_local_name 有什么 区别?

答:

- by_device_name 表示用此接口实现设备注册的设备名称device_name,必须与在阿里云物联网平台 创建设备时的名称一样。
- o by_local_name 表示用此接口实现设备注册的设备名称device_name,可使用自定义名称。

⑦ 说明 针对同一ProductKey下的设备注册,必须保证只能使用同一接口。即 by_device_name 和 by_local_name 接口不允许在同一ProductKey下同时使用。

 设备正常上线且从设备所在驱动的日志中看到设备数据正常上报,但在物联网平台控制台,设备数据上报 内容没有被更改?

答:

确认是否在物联网平台控制台配置了消息路由,详细操作请参见设置消息路由内容。

2.5. 驱动发布

2.5.1. 云端发布

驱动支持云端发布和内置发布两种形式。云端发布驱动,是将驱动文件托管到控制台中**驱动管理**页面;内置 发布驱动,是将驱动内置到网关。本文主要描述通过云端发布驱动的步骤。

前提条件

- 您完成了驱动的开发,具体操作步骤,请参见驱动编码。
- 您已打包了已开发的驱动。详细信息,请参见驱动打包注意事项。

操作步骤

- 1. 登录边缘计算控制台,在左侧导航栏单击驱动管理。
- 2. 在驱动管理页面自研驱动页签下,单击的新建驱动。
- 3. 在新建驱动页面,根据提示设置驱动参数。
 - 驱动信息

驱动信息	
* 驱动名称 <a>2 请输入驱动名称	
* 通信协议类型 💿	
请选择通信协议类型	\sim
* 语言类型:	
请选择语言类型	\sim
驱动是否内置②②是②③否	
* 驱动文件 ② 上传文件	
* 驱动版本	
请输入驱动版本	
* 驱动适配的边缘版本 💿	
2.5.0 版本及以上	\sim
驱动版本描述:	
请输入驱动版本描述	

驱动信息参数说明

参数	描述
驱动名称	自定义驱动的名称。支持大小写英文字母、数字和英文下划线(_),必须以英文 字母开头,不超过20个字符。
通信协议类型	驱动开发所依赖的行业标准通信协议。当前支持的协议类型 有Modbus、OPCUA、LoRa WAN和自定义。
语言类型	驱动的语言类型,支持Node.js 8、Python 3.5、C和Java 8类型。 其中,若选择C类型,则还需要选择驱动的CPU架构。
驱动是否内置	选择是否内置驱动文件。 是:无需上传驱动文件,使用本地内置驱动。详情请见内置发布。 否:需要上传驱动文件,通过云端部署通道直接部署驱动。 云端发布必须选择否。
驱动文件	驱动是否内置选择否后,单击 上传文件 ,上传您的驱动文件。
驱动版本	设置驱动的版本,必须是该驱动唯一的版本号,即一个驱动不可以设置两个相同 的版本号。
驱动适配的边缘版本	选择驱动适配的边缘版本,即该驱动只能在该边缘版本及以上版本的网关中运 行。
驱动版本描述	描述您创建的驱动,可以为空。

○ 驱动配置

驱动配置	
配置格式	
键值对配置	\sim
键值对配置表 💿	

驱动配置参数说明

参数 描述 描述

参数	描述
配置格式	配置格式有如下三种: 键值对配置 JSON格式 配置文件
键值对配置表	配置格式为键值对配置时显示的参数。 单击 新增配置 ,通过设置配置名称、值、注释来配置驱动。 最多添加100个键值对。
配置内容	配置格式为JSON格式时显示的参数。 编写JSON格式配置内容,内容不超过1 KB。编写内容时,系统自动校验格式。若 格式错误,可根据系统提示编写正确格式的内容。
上传配置文件	配置格式为配置文件时显示的参数。 需要在本地编写配置文件并上传。配置文件的编写 <i>,</i> 请参见 <mark>驱动编码</mark> 。

○ 容器配置

容器配置	
是否使用宿主机host模式	
○ 定 ● 百 ○ 定 ● 百 网络端口映射 ②	
+ 新增端口映射	
是否启动特权模式 💿 🔿 是 💿 否	
设备映射 📀	
+ 新增设备映射	
+ 新增卷映射	
参数 拍	

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 是:表示不隔离容器的网络,直接使用宿主机网络环境。 否:表示隔离容器的网络,需要设置网络端口映射。
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境 是完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某 一个端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多 支持映射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提 供服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问 到 fc-http-server 函数。因此需要将 fc-http-server 函数所在的容器 端口映射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机8080端 口),允许其它主机上的客户端程序通过访问 宿主机的IP地址:8080 ,访问 到容器内部的 fc-http-server 函数。
是否启动特权模式	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。 ⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔 离的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运 行函数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时, 需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 当容器内存使用超过限制时,容器会被重启。 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

。 配置校验



配置校验参数说明

参数	描述
驱动配置	勾选表示将该驱动配置到边缘实例并关联子设备后,必须设置 驱动配置 ,才可以 将驱动部署到边缘实例。
设备配置	勾选表示将该驱动配置到边缘实例并关联子设备后,必须设置 设备配置 ,才可以 将驱动部署到边缘实例。

4. 完成参数的设置并上传成功驱动文件后,单击确认。您可以在自研驱动页签中看到刚刚创建的驱动。

相关操作

1. 在驱动管理页面自研驱动页签,单击驱动操作栏中的版本管理。

驱动管理								
官方驱动	自研驱动	已购驱动						
新建築の対 清給入販売合称 Q						C		
驱动名称		通信协议类型	CPU架构	状态	最新版本	创建时间	操作	
-		自定义	-	• 已创建	2.0.0	2020-01-09 16:13:58	版本管理上架	

2. 在版本管理面板,单击相关操作按钮,管理该驱动的版本信息。

操作	描述
新增版本	单击 新增版本 ,可为该驱动增加新版本,详细操作及参数说明,请参见本 文上方 <mark>设置驱动参数步骤</mark> 。
	单击 发布 ,在弹出的 确认发布驱动版本 对话框,确认发布驱动信息后, 单击 发布 。
发布	⑦ 说明 驱动发布成功后,不可以删除该版本驱动,只能查 看或下载该版本驱动。
编辑	驱动版本的状态为未发布时才有的操作按钮。单击 编辑 , 可修改该版本驱 动的信息,参数说明,请参见本文上方 <mark>设置驱动参数步骤</mark> 。
下载	单击 下载 ,可下载该版本驱动文件。
删除	驱动版本的状态为未发布时才有的操作按钮。单击 删除 ,可删除该版本驱 动。
查看	驱动版本的状态为已发布时才有的操作按钮。单击查看,可查看该版本驱 动信息,也可以修改该版本驱动信息,参数说明,请参见本文上方 <mark>设置驱</mark> 动参数步骤。

- (可选)单击驱动操作栏中的上架,可将自研驱动作为商品,上架到物联网市场。上架驱动操作,请参见驱动市场。
- 4. 在云端发布驱动后,即可将驱动分配到边缘实例,关联子设备,通过边缘实例的部署,将驱动和子设备 部署到边缘端。详细信息,请参见驱动调试。

2.5.2. 内置发布

物联网边缘计算提供驱动内置能力,支持将驱动内置到网关。驱动内置功能,支持您在本地管理驱动文件, 再结合云端配置驱动和驱动信息,将驱动部署到边缘端使用。

前提条件

内置的驱动需要符合Link loT Edge的驱动规范,因此请确保已根据驱动规范,开发完成了您的驱动程序。驱动开发详情请参见<mark>驱动开发</mark>。

背景信息

物联网边缘计算提供驱动内置能力,支持将驱动程序内置到网关,然后在云端配置部署驱动描述信息(驱动 名称、驱动编码语言、驱动运行环境、硬件架构等)和驱动通道配置、设备配置等。



驱动内置发布流程图如下所示。



一、云端操作

- 1. 登录边缘计算控制台,在左侧导航栏单击驱动管理。
- 2. 在驱动管理页面自研驱动页签下,单击的新建驱动。
- 3. 在新建驱动页面,根据提示设置驱动参数。

• 驱动信息

驱动信息	
* 驱动名称 💿	
请输入驱动名称	
* 通信协议类型 💿	
请选择通信协议类型	\sim
* 语言类型:	
请选择语言类型	\sim
驱动是否内置 ②● 是 ○ 否	
* 驱动版本	
请输入驱动版本	
* 驱动适配的边缘版本 💿	
2.4.1 版本及以上	\sim
驱动版本描述:	
请输入驱动版本描述	

驱动信息参数说明

参数	描述
驱动名称	自定义驱动的名称。支持大小写英文字母、数字和英文下划线(_),必须以英文 字母开头,不超过20个字符。
通信协议类型	驱动开发所依赖的行业标准通信协议。当前支持的协议类型 有Modbus、OPCUA、LoRa WAN和自定义。
语言类型	驱动的语言类型,支持Node.js 8、Python 3.5、C和Java 8类型。 其中,若选择C类型,则还需要选择驱动的CPU架构。
驱动是否内置	选择是否内置驱动文件。 是:无需上传驱动文件,使用本地内置驱动。 否:需要上传驱动文件,通过云端部署通道直接部署驱动。详情请见云端发 布。 内置发布必须选择是。
驱动版本	设置驱动的版本,必须是该驱动唯一的版本号,即一个驱动不可以设置两个相同 的版本号。

参数	描述
驱动适配的边缘版本	选择驱动适配的边缘版本,即该驱动只能在该边缘版本及以上版本的网关中运行。
驱动版本描述	描述您创建的驱动,可以为空。

。 驱动配置

驱动配置	
配置格式	
键值对配置	\sim
键值对配置表 ② + 新增配置	

驱动配置参数说明

参数	描述
配置格式	配置格式有如下三种: 键值对配置 JSON格式 配置文件
键值对配置表	配置格式为键值对配置时显示的参数。 单击 新增配置 ,通过设置配置名称、值、注释来配置驱动。 最多添加100个键值对。
配置内容	配置格式为JSON格式时显示的参数。 编写JSON格式配置内容,内容不超过1 KB。编写内容时,系统自动校验格式。若 格式错误,可根据系统提示编写正确格式的内容。
上传配置文件	配置格式为配置文件时显示的参数。 需要在本地编写配置文件并上传。配置文件的编写,请参见 <mark>驱动编码</mark> 。

○ 容器配置

容器配置	
是否使用宿主机host模式 〇 是	
网络端口映射 📀	
+新増端口映射 是否启动特权模式 ②	
○ 是 ● 否	
设备映射 💿 + 新增设备映射	
卷映射 🕝 + 新增卷映射	

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 是:表示不隔离容器的网络,直接使用宿主机网络环境。 否:表示隔离容器的网络,需要设置网络端口映射。
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境 是完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某 一个端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多 支持映射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提 供服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问 到 fc-http-server 函数。因此需要将 fc-http-server 函数所在的容器 端口映射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机8080端 口),允许其它主机上的客户端程序通过访问 宿主机的IP地址: 8080 ,访问 到容器内部的 fc-http-server 函数。

参数	描述
	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统 时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。
是否启动特权模式	⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所 有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔 离的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运 行函数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时, 需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 当容器内存使用超过限制时,容器会被重启。 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

配置校验

配置校验	
驱动配置 💿	
し 设备配置 💿	

配置校验参数说明

参数	描述
驱动配置	勾选表示将该驱动配置到边缘实例并关联子设备后,必须设置 驱动配置 ,才可以 将驱动部署到边缘实例。
设备配置	勾选表示将该驱动配置到边缘实例并关联子设备后,必须设置 设备配置 ,才可以 将驱动部署到边缘实例。

4. 设置参数完成后,单击确认。您可以在自研驱动列表中看到刚刚创建的驱动。

~	官方驱动	自研驱动	已购驱动							
~	新花曲医动	请输入驱动名	称	Q						C
驱动ID:24562		294df87复制	通信协议类型		CPU願构	状态	最新版本	仓级建时间	攝作	
^	led_driver		自定义		x86-64		v1.0.0	2020/03/10 17:59:31	版本管理 上架	

鼠标悬浮到驱动名称上,可以查看该驱动的ID,请单击复制并保存驱动ID的文本内容,后续步骤中将使用该驱动ID。

二、边缘端操作

Link IoT Edge提供led_driver示例驱动程序,详情请参见led_driver源码。您可以使用示例驱动程序,体验驱动内置功能。

- 1. 创建边缘实例并上线网关,具体操作请参见环境搭建。
- 2. 登录网关执行如下命令, 创建内置驱动总目录。

sudo -E mkdir -p /linkedge/pre-installed/

3. 执行如下命令, 创建内置驱动文件夹, 并将"前提条件"中准备的驱动程序及依赖库放置到该目录下。

sudo -E mkdir -p /linkedge/pre-installed/{your_driver_name}/

其中, *{your_driver_name}*为您驱动程序的名称。例如驱动程序名称为led_driver,则实际执行的命令如下所示。

sudo -E mkdir -p /linkedge/pre-installed/led_driver/

4. 检查网关是否已安装如下内置驱动的依赖工具。

工具名称	用途
sed	查找和替换文本文件内容的工具。
jq	JSON文件解析工具。
base64	Base64加密工具。

- 5. (可选)若网关上没有sed、jq、base64工具,您需要执行如下步骤,确保内置驱动Shell脚本工具正常运行。
 - i. 进入Link IoT Edge默认配置目录。

cd /linkedge/gateway/build/config/config-manager

ii. 打开config_default.json文件,找到config条目。

iii. 在 config条目下,新增如下 key: value 键值对。

```
{
    "config": {
        "gw_drivercode_{your_driver_id}": "{your_driver_path}",
    }
}
```

其中, *{your_driver_id*}为云端操作中保存的驱动ID; *{your_driver_path*}为Base64加密后的内置驱动 文件夹路径。

例如,驱动ID为*f16f13322********3959cf3*,内置驱动文件夹路径/*linkedge/pre-installed/led_driver/*,该路径Base64加密后为*L2xpbmtlZGdlL3ByZS1pbnN0YWxsZWQvbGVkX2RyaXZlcg==*,则实际增加的 key: value 键值对如下。

```
{
    "config": {
        "gw_drivercode_f16f13322*******3959cf3": "L2xpbmt1ZGdlL3ByZS1pbnN0YWxsZW
    QvbGVkX2RyaXZlcg==",
     }
}
```

6. 在内置驱动文件夹下,下载内置驱动Shell脚本工具并解压。

脚本工具解压后的目录结构如下所示。

```
 led_driver
    |--- pre-installed.sh
    |--- lib
    ____ main
```

7. 进入内置驱动所在目录,执行内置驱动Shell脚本工具。

```
cd /linkedge/pre-installed/{your_driver_name}/
sudo -E ./pre-installed.sh {your driver id} --default
```

其中, {your_driver_name}为驱动程序名称, {your_driver_id}为云端操作中保存的驱动ID。

例如,驱动程序名称为led_driver,驱动ID为f16f13322*******3959cf3,则实际执行的命令如下。

cd /linkedge/pre-installed/led_driver/ sudo -E ./pre-installed.sh f16f13322*******3959cf3

8. 参考驱动调试内容,将内置驱动分配到边缘实例,并部署边缘实例,完成边缘端驱动内置流程。

常见问题

- 问:内置驱动Shell脚本执行完成后,没有看到设备上线?
- 答:必须在云端完成驱动的配置,并部署到网关后,才能上线与之关联的子设备。

相关操作

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏中,单击驱动管理。
- 3. 在驱动管理页面自研驱动页签,单击驱动操作栏中的版本管理。

驱动管3	理														
官方驱动	自研驱动	已购驱动													
新建驱动	请输入驱动名	称		Q											G
驱动名称		通信协	的议类型		CPU架构	状态	5	最新	航版本	êli	建时间		操作		
-		自定义	L.		-	•	已创建	2.0.	0	20.	20-01-09 16:13:5	58	版本管理	上架	

4. 在版本管理对话框,单击相关操作按钮,管理该驱动的版本信息。

操作	描述
新增版本	单击 新增版本 ,可为该驱动增加新版本,详细操作及参数说明请参见本文 上方 <mark>云端操作</mark> 。
	单击 发布 ,在弹出的 确认发布驱动版本 对话框,确认发布驱动信息后, 单击 发布 。
发布	⑦ 说明 驱动发布成功后,不可以删除该版本驱动,只能查 看或下载该版本驱动。
编辑	驱动版本的状态为未发布时才有的操作按钮。单击 编辑 ,可修改该版本驱 动的信息,参数说明请参见本文上方 <mark>云端操作</mark> 。
下载	单击 下载 ,可下载该版本驱动文件。
删除	驱动版本的状态为未发布时才有的操作按钮。单击 删除 ,可删除该版本驱 动。
查看	驱动版本的状态为已发布时才有的操作按钮。单击查看,可查看该版本驱 动信息,也可以修改该版本驱动信息,参数说明请参见本文上方 <mark>云端操</mark> <mark>作</mark> 。

- 5. (可选)单击驱动操作栏中的**上架**,可将自研驱动作为商品,上架到物联网市场。上架驱动操作请参见驱动市场。
- 在云端发布驱动后,即可将驱动分配到边缘实例,关联子设备,通过边缘实例的部署,将驱动和子设备 部署到边缘端。详情请参见驱动调试。

2.6. 已购驱动

本文介绍已购驱动的查看和使用方法 。

前提条件

已创建边缘实例并上线网关,详细操作,请参见环境搭建。

限制说明

当前Link loT Edge不支持新购买驱动,或再次购买已购驱动,本文仅适用于拥有已购驱动的用户。

操作步骤

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击驱动管理,然后单击已购驱动页签,查看您的已购驱动。

驱动管理	Ŧ					
官方驱动	自研驱动	已购驱动				
请输入驱动名称	R	Q				C
驱动名称		驱动语言	通信协议类型	CPU架构	可用数量/总数量	有效时间
		с	自定义	x86-64	98/100	2021/11/17 20:56:51

- 3. 将已购驱动分配到边缘实例中。
 - i. 在左侧导航栏单击边缘实例,找到已创建的边缘实例,单击对应操作列的查看。
 - ii. 在**实例详情**页面,单击**设备与驱动**页签,然后单击**全部驱动**右侧的 + 图标。
 - iii. 在**分配驱动**面板中选择已购驱动,单击对应驱动名称后的分配。

已购服动 > 调始入服	动名称	Q			
	已時期版物 >> 請給入販売台称 Q				
驱动名称	CPU卵构	有效时间	版本	操作	
1000.0100.000	x86-64	2024-12-20 13	v0.1 V	分配	
		2019-09-20 20	1.1.0-r 🗡	分配	
1710756840500000488	x86-64	2019-09-20 20		不可分配	
	x86-64	2019-09-20 17		不可分配	
10000000	x86-64	2024-09-13 00	1.0.0 🗸	分配	
		x85-64 - x85-64 x85-64 x85-64 x85-64	x86-64 2004-12-20 13 - 2019-09-20 20 1 x86-64 2019-09-20 20	x86-64 2024-12-20 13 v0.1 - 2019-09-20 20 1.1.0-r x86-64 2019-09-20 20 - x86-64 2019-09-20 17 - x86-64 2019-09-20 17 - x86-64 2019-09-20 17 -	

iv. (可选)在**设备与驱动**页签,在已分配驱动的**设备列表**区域,单击**驱动配置**,可对当前驱动进行 配置。配置完成后单击**确定**。

驱动配置参数说明

参数	描述
配置格式	配置格式有如下三种:
键值对配置表	配置格式为键值对配置时显示的参数。 单击 新增配置 ,通过设置配置名称、值、注释来配置驱动。 最多添加100个键值对。
配置内容	配置格式为JSON格式时显示的参数。 编写JSON格式配置内容,内容不超过1 KB。编写内容时,系统自动校验格式。 若格式错误,可根据系统提示编写正确格式的内容。
上传配置文件	配置格式为配置文件时显示的参数。 需要在本地编写配置文件并上传。配置文件的编写 <i>,</i> 请参见 <mark>驱动编码</mark> 。
v. (可选)在**设备列**表右侧单击容器配置,对当前驱动进行容器配置。配置完成后单击保存。

⑦ 说明 仅在产品规格为专业版的边缘实例中,允许设置容器配置。

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 是:表示不隔离容器的网络,直接使用宿主机网络环境。 香:表示隔离容器的网络,需要设置网络端口映射。
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境 是完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某 一个端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多 支持映射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提 供服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问 到 fc-http-server 函数。因此需要将 fc-http-server 函数所在的容 器端口映射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机 8080端口),允许其它主机上的客户端程序通过访问 宿主机的IP地址:8080 ,访问到容器内部的 fc-http-server 函数。
ᆸᇫᇦᆠᆄᄳᄲᆃ	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统 时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特 权。
定台后动特仪侯式	⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的 所有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔 离的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运 行函数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时, 需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 当容器内存使用超过限制时,容器会被重启。 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

vi. 单击**分配子设备**,在已购驱动下,为边缘实例分配子设备。

✓ 全部驱动 已购驱动	+	设备列表 驱动配 分配子设备	置 容器配置 了解设备接入?	全部产品	✔ 设备名称	C	2
		设备名称		所属产品	格式配置	设备状态	操作
					Ø	尚未添加任何子设备 IoT Edge 可连接位于边缘的设备 分配子设备 了解设备	番,并且提供了完整边缘设备接入工具。 接入

vii. (可选)分配设备到边缘实例后,单击设备名称对应操作列中的设备配置,可对当前设备进行配置,例如配置设备序列号,方便驱动处理指定序列号设备的相关信息。

编写JSON格式配置内容,内容大小不超过1 KB。单击校验格式,通过校验后单击确定完成配置。 4. 部署边缘实例。

i. 在**实例详情**页面, 单击右上角的部署后在弹出对话框中单击确定, 部署边缘实例。

ii. 当部署状态显示为**部署成功**,表示部署实例完成。您可以单击查看日志,查看部署详情。

3.场景联动 3.1. 什么是场景联动

场景联动是规则引擎中,一种开发自动化业务逻辑的可视化编程方式,您可以通过可视化的方式定义设备之间联动规则,并将规则部署至云端或者边缘端。

您需在物联网平台控制台,规则引擎 > 场景联动页面中创建场景联动规则。每个场景联动规则由触发器 (Trigger)、执行条件(Condition)、执行动作(Action)三个部分组成。这种规则模型称为TCA模型。

⑦ 说明 当前仅华东2(上海)地域下,企业版实例和新版公共实例,支持场景联动功能。
 实例的详细说明,请参见查看实例终端节点。

当触发器指定的事件或属性(不包含date、enum、struct和array类型属性)变化事件发生时,系统通过判断执行条件是否已满足,来决定是否执行规则中定义的执行动作。如果满足执行条件,则直接执行定义的执行动作;反之则不执行。

例如,您每天18:00下班回家。在炎热的夏天,您希望您到家后,家里的温度是凉爽、舒适的。您可以创建 一条规则,使空调设备自动化,实现这个需求。

参数设置如下图。

场景联动规则								
触发器 (Trigger) 🕜								
触发器1								
定时触发	~	0 18 * * *		0				
+ 新噴服及醬								
执行条件 (Condition) 💿								
执行条件1								
设备状态	~	温度传感器	\sim	temperatureSensor01	\sim			
温度	~	>	~	26				
Anna Conce								
+ 新增执行条件								
+ 新增执行条件 * 执行动作 (Action)								
+ 新增执行条件 * 执行动作 (Action) 执行动作1								
+ 新增执行条件 * 执行动作 (Action) 执行动作1 设备输出	~	空调	~	airconditioner_1	~			
+ 新增执行条件 * 执行动作 (Action) 执行动作1 设备输出 由调开关	~	空调 王中 1	~	airconditioner_1	~			
+ 新增执行条件 * 执行动作 (Action) 执行动作1 设备输出 电源开关	~	空调 开启-1	~	airconditioner_1	~			
 + 新增执行条件 * 执行动作 (Action) 执行动作1 设备输出 电源开关 展开高级选项 ∨ 	~	空调 开启-1	~	airconditioner_1	~			
 + 新增执行条件 * 执行动作 (Action) 执行动作 1 设备输出 电原开关 展开高级选项 ✓ 	~	空调 开启-1	~	airconditioner_1	~			
+ 新増执行条件 + 抗行动作 (Action) 抗行动作 1 ② 备輸出 电源开关 展开高级选项 ~ 执行动作2	~	空调 开启-1	~	airconditioner_1	~			
 新增执行条件 执行动作 (Action) 执行动作1 设备输出 电原开关 展开高级选项 > 执行动作2 设备输出 	~	空调 开启-1 空调	~	airconditioner_1	~			

参数说明如下:

参数	描述	并列关系
触发器	定时为每天18:00触发该规则。时间的cron表达式写作方法,请参 见CRONTAB网页。	或()

参数	描述	并列关系
执行条件	获取温度传感器上报的数据,若室内温度高于26摄氏度,则执行动 作。	与(&&)
执行动作	空调开关设置为打开;空调目标温度设置为26摄氏度。	与(&&)

创建场景联动规则的更多设置说明,请参见云端场景联动。

3.2. 云端场景联动

场景联动类型的规则是一种开发自动化业务逻辑的可视化编程方式,可以通过设备或时间维度的条件触发, 经过执行条件的过滤,执行预定的业务逻辑,输出数据到设备或者其他规则,实现海量设备的场景联动。

前提条件

已完成边缘实例的创建。具体操作步骤请参见环境搭建。

创建场景联动

- 1. 登录物联网平台控制台,选择华东2(上海)地域,单击公共实例。
- 2. 在左侧导航栏选择规则引擎 > 场景联动。
- 3. 单击创建规则。

物联网平台 / 规则引擎 / 场景明	联动				
场景联动					
创建规则 请输入规则名称		٩			
规则名称	规则描述		创建时间	云端运行状态	云端操作
$(a,b,c,b,b) \in [0,1] \times [0,1]$			10101-0111-010		查看 启动 日志 删除
10.000			10.0.0		查看 启动 日志 删除

4. 设置参数,然后单击确认。

参数	描述
规则名称	设置具体规则的名称。支持中文、英文字母、数字、下划线(_)和短划线(-), 长度限制为1~30个字符,中文字算两位字符。
规则描述	为规则添加描述,可以为空。

5. 完成场景联动的创建后, 在弹出的对话框中单击前往编辑, 管理配置场景联动。

您也可以在场景联动名称右侧单击查看,管理配置场景联动。

以空调设备自动化为例:在12:00至23:59之间,当温度传感器上报的室内温度低于16摄氏度时,空调 设备开始工作,且设置温度为26摄氏度。

具体参数设置,请见下图。

场	景联动规则						
触る	號器 (Trigger) 🕜						
ţ	独发器1						删除
	设备触发	\sim	温度传感器	-	temperatureSensor01 V		
	属性触发	~	温度	-	<	16	
+ 亲	^指 體設发器						
执行	ī祭件 (Condition) 🕖						
1	丸行祭件1						删除
	时间范围	~	****-**-** 12:00:00		****-**-** 23:59:00	0	
<							
+ 3	增执行条件						
* 扶	行动作 (Action)						
(+200					
	有不做成且力11月口音和例如1180次4	2009FK ; 3					
1	丸行动作1						删除
	设备输出	~	空调	-	airconditioner_1 V		
	目标温度	~	26	6			
J	展开高级选项 ~						
+ 亲	「増执行动作						
	保存 取消						

单击页面右上角编辑,可更改场景联动规则名称,其余参数说明请见下表。

参数	描述
	即规则入口。可设置为 设备触发 或定时触发。当设备上报的数据或当前时间满足设定的触发器时,触发执行条件判断。可以为一个规则创建多个触发器,触发器之间 是或(or)关系。
	 设置为设备触发,则需选择已创建的产品、设备(一个或全部设备)、和设备属性(一个或全部属性)或事件(一个或全部事件)。
触发器	 设置为定时触发,则需填写时间点。时间点格式为cron表达式。cron表达式的构成:分、小时、日、月、一周内的某天(0或7表示周日,1~6分别表示周一至周六),每项之间用空格隔开。如,每天18点整的cron表达式为:0 18 * * * (其中星号(*)是通配符);每周五18点整的表达式为:0 18 * * 5。cron表达式具体写作方法,请参见CRONT AB 网页。
	上图示例中,设置为 设备触发 :以温度传感器上报的室内温度低于16摄氏度作为 触发器。
	执行条件集。只有满足执行条件的数据,才能触发执行动作。可设置为 设备状 态或时间范围。可以为一个规则创建多个执行条件,执行条件之间是和(and)关 系。
执行条件	 设置为设备状态,则需选择已创建的产品、该产品下的某个设备、和设备功能中的某个属性或事件。
	• 设置为时间范围,则需设置起始时间和结束时间,格式为 yyyy-mm-dd hh24: mi:ss 。
	上图示例中,设置为 时间范围 :时间在12:00至23:59之间,则触发执行动作。

参数	描述
执行动作	 需执行的动作。您可以设置多个动作。某一动作执行失败时,不影响其他动作。 设置为设备输出,则需选择已创建的产品、该产品下的某个设备、和设备功能中的某个属性或服务(只有可写的属性或服务才能被设为执行动作)。当触发器和执行条件均被满足时,执行已定义的设备属性或服务的相关动作。 设置为规则输出,则需嵌套另外一个规则,即调用其他规则。被调用规则中的触发器将被跳过,直接进行执行条件检查。若执行条件满足,则执行该规则中定义的执行动作。 例如,被调用规则为A,那么规则A中的触发器被跳过,直接检查其执行条件内容,若满足规则A的全部执行条件(多个执行条件之间为&&关系)内容,则会执行规则A中的执行动作。 设置为函数输出,则需要选择一个已创建的函数。当触发器和执行条件均被满足时,运行已选定的函数。创建函数相关内容,请参见使用控制台创建函数。 设置为告警输出,则需要将该场景联动规则关联到告警中心。当触发器和执行条件均被满足时,触发告警。单击告警中心,前往告警中心设置告警规则。详细操作,请参见告警中心。 上图示例中,设置为设备输出:指定的空调设备,执行设置温度为26摄氏度的动作。
延时执行	展开高级选项后的参数。设置延时时间后,执行动作将延迟执行。取值范围为 0~86400秒。

运行场景联动

场景联动创建成功后,您可在**场景联动**页面中,启动此场景联动。

启动场景联动操作:

- 1. 在物联网平台控制台公共实例页面, 左侧导航栏选择规则引擎 > 场景联动。
- 2. 找到要启动的场景联动,单击右侧操作栏中的启动,使规则状态为运行中。

场景联动				
创建规则 请输入规则名称	Q			
规则名称	规则描述	创建时间	云端运行状态	云端操作
all the re-		30.13754	1 755	查看 启动 日志 删除
10.01010.00				查看 启动 日志 删除
我的家自动化场景		2019-11-11 16:47:56	● 运行中	查看 停止 触发 日志 删除

启动场景联动后:

- 若场景联动在云端运行,则需要为场景联动中的设备配置消息路由,使得设备的属性和事件能够发送到 loT Hub(云端)。消息路由的配置,请参见设置消息路由。
- 若场景联动在边缘端运行,则需要先停止其在云端的运行,再将场景联动分配到边缘实例中,分配方法, 请参见本文下方场景联动其他操作。

查看日志

您可以查看该场景联动的日志,并且可在详情中查看运行结果。

⑦ 说明 若某条场景联动既在云端运行又在边缘端运行,那么在物联网平台控制台规则引擎 > 场景 联动中,查看到的日志为云端运行日志和边缘端运行日志。

- 1. 在物联网平台控制台公共实例页面, 左侧导航栏选择规则引擎 > 场景联动。
- 2. 找到要查看日志的场景联动,单击右侧操作栏中的日志。
- 3. 单击详情,查看该条日志的详情信息。

物联网平台 / 规则引擎 / 场景联动 / 运行日志		
← 我的家自动化场景		
全部状态 > 自定义 >	· 2019-12-23 13:16:50 - 2019-12-23 13:31:50	C
执行时间	执行状态	操作
2019-12-23 13:31:48	成功	译语
2019-12-23 13:30:37	成功	详情

⑦ 说明 若有执行状态为失败的日志,可单击对应操作栏中的详情,查看场景联动执行失败的详细信息。

场景联动其他操作

- 删除场景联动:
 - i. 在场景联动页签中, 找到需要删除的场景联动规则名称。
 - ii. 单击规则名称右侧的删除,在弹出的对话框中单击确认,删除该条场景联动规则。
- 触发场景规则:

在启动场景联动规则后,方可显示触发操作按钮。

- i. 在场景联动页签中, 找到已启动的需要触发的场景联动规则名称。
- ii. 单击规则名称右侧的**触发**,表示手动触发规则一次,即忽略已管理配置的触发器,直接执行所有执行 条件和执行动作。
- 在边缘实例中运行场景联动:

您需要根据如下步骤,将场景联动部署到边缘实例中。

↓ 注意 请确保已停止场景联动在云端的运行。

- i. 登录边缘计算控制台,在左侧导航栏选择边缘实例,单击"前提条件"中已创建的边缘实例右侧的查 看。
- ii. 在**实例详情**页面,选择场景联动,单击分配场景。
- iii. 在**分配场景**对话框,单击待分配场景联动规则名称后的分配,然后单击关闭。

边缘计算 / 边缘实例 / 实例评情		分配场景	×
	服然斯太亮	汤加汤层IK动 请输入规则名称 Q	C
 CPU 使用本 - 畫音 	内存使用率	规则名称	操作
网关 监控信息 设备与驱动 场景联动 边缘应用 流数据分析 滿	總路由 日志服务 设置	ac.	分配
Эћ/6 8		Record Control of Cont	分配
规则名称 分配时间		National States	分配
			分配
	尚未分配任何场景联动	10(10),11	分配
	制造运行的应用程序,可以帮助您实现本地及时处理	我的家自动化场景	分配
	分配场景了解场景联动	人 上一页 1/1 下一页 > 共有6条每页显示	: 15 🗸
		关闭	

iv. 分配场景联动后, 重新部署边缘实例。

3.3. 边缘场景联动

部署至边缘实例的场景联动能够快速响应本地的消息,并且可在断网环境下正常运行。本文介绍利用场景联动规则,实现当光照度传感器(Light Sensor设备)检测到光照度大于等于500流明的时候,控制灯(Light设备)关闭,当光照度小于等于200流明的时候,控制灯打开。

流程如下图所示。



前提条件

- 1. 请您确保已根据环境搭建内容完成边缘实例的创建。
- 2. 请根据示例驱动内容,使用官方示例驱动完成LightSensor和Light设备的创建,并将设备分配到边缘实例中。

操作步骤

- 1. 创建场景联动规则。
 - i. 在边缘实例页面,选择已在前提条件中创建完成的边缘实例,单击右侧的查看。

ii. 在**实例详情**页面,选择场景联动页签,单击分配场景。

您可以为实例分配已创建的场景联动规则,若没有可分配的规则,请继续参考下方的步骤创建新的场景联动规则。

iii. 在分配场景对话框中,单击添加场景联动。系统自动跳转到物联网平台控制台公共实例 > 规则引
 擎 > 场景联动页面中的创建场景联动规则对话框。

边缘计算 / 边缘实例 / 实例详情	分配场景 X
← LinkloTEdge_Node	<u> 浅山の15年間(45)</u> 清晴会入り回り15日: Q C
☆島間路 专业版 脱劣版本号 CPU 使用率 - ■書 内容使用率	规则告称 操作
网关 监控信息 设备与驱动 场景联动 边缘应用 流数据分析 满意路由 日志服务 设置	分配
分離が長度	2010 - 20
规则名称 分散时间	分配
	9 8
尚未分配任何场景联动 IoT Edge可将简单规则从云扩展至边虑,简单规	98
	122 現的家自动化场景 分配
分散场展 了解话题联动	く 上一页 1/1 下一页 > 共有6条每页显示: 15 💙
	· · · · · · · · · · · · · · · · · · ·

- iv. 参考云端场景联动,创建名为Rule_Light_Open的场景联动规则。
- v. 创建完成后,管理配置场景联动规则,具体参数设置如下图所示。

← Rule_Ligh	t_Oper	ı			
描述					
场景联动规则					
触发器 (Trigger) 🛞					
触发器1					60 0F
设备触发	~	光照度传感器	LightSensor	<i>y</i>	
光照度检测值	~	<= ~	200	0	
+新増触发器					
执行条件 (Condition) 💿					
执行条件1					Ð
请选择执行条件	~				
+ 新增执行条件					
* 执行动作 (Action)					
执行动作1					
设备输出	~	客厅灯	Light	×	
主灯开关	~	开启-1 ~	·		
展开高级选项 ~		L			
+ 新增执行动作					

配置场景联动后,单击下一步跳转回边缘计算控制台**分配场景**对话框。

⑦ 说明 从实例详情页面场景联动页签的分配场景对话框跳转到规则引擎 > 场景联动的 情况下,原保存按钮变为下一步,方便您创建完成场景联动后直接跳转回分配场景对话框, 分配新建的场景联动规则到边缘实例。

- vi. 在新建的场景联动规则名称后,单击**分配**,将场景联动分配到边缘实例。
- vii. 在**分配场景**对话框中,单击**添加场景联动**,参考创建Rule_Light_Open场景联动的步骤,创建名为Rule_Light_Close的场景联动。

viii. 管理配置Rule_Light_Close场景联动,具体参数如下图所示。

-	Rule_Light_C	Close	;			
描述						
场景	银动规则					
触发	器 (Trigger) 💿					
AS	戦闘1					
	设备触发	~	光照度传感器	~	LightSensor	~
	光照度检测值	~	>=	~	500	
+ 新	増触发器					
执行	条件 (Condition) 🛞					
执	1行条件1					
	请选择执行条件	~				
+ 99	博地行各性					
* 执行	庁司が乍 (Action)					
执	4行动作1					
	设备输出	~	客厅灯	~	Light	~
	主灯开关	~	关闭-0	~		
展	研高级选项 🗸					

2. 部署边缘实例。

i. 在**实例详情**页面,单击右上角**部署**后在弹出对话框中单击**确定**,部署边缘实例。

ii. 当部署状态显示为**部署成功**,表示部署实例完成。您可以单击**查看日志**,查看部署详情。您也可 以在实例详情页面,右上角单击**查看部署日志**,查看历史部署情况。

如果部署失败,在**部署详情**页面,可以单击部署失败的资源名称前的 + 号图标,进行查看具体 失败原因。

部署详情					\times
<mark>网关1</mark> 网关:ga					
	资源名称	操作	状态	详情	
+	device_config 子设备配置	部署	● 成功		
+	tsl_config 物模型配置	部署	● 成功		
+	message_route_config 消息路由配置	部署	● 成功		
+	instance_config 实例配置	部署	● 成功		
	master_backup_config	部署	● 失败	11003	
2019 2019 2019 2019 2019 2019	-12-18 16:22:31 [INFO] [master_backup_config] [0] init succes -12-18 16:22:31 [INFO] [master_backup_config] [0] assembly -12-18 16:22:31 [INFO] [master_backup_config] [0] package s -12-18 16:22:31 [INFO] [master_backup_config] [0] dispatchir -12-18 16:22:32 [ERROR] [master_backup_config] [11003] No	ss success success ng o such netwo	rk interface		
4					►.

部署实例成功后,已分配到该实例的场景联动规则将会在边缘端开始运行、监听及触发执行。

3. 查看边缘实例场景联动日志。

⑦ 说明 若某条场景联动既在云端运行又在边缘端运行,那么在物联网平台控制台规则引擎 > 场景联动中,查看到的日志为云端运行日志和边缘端运行日志。

i. 在实例详情页面场景联动页签,单击规则名称右侧操作栏中的日志。

ii. 在规则日志页面,单击详情查看日志详情。

全部状态 ン 自定义	2019-12-24 09:49:52	- 2019-12-24 10:04:52			
有行時间		执行状态			操作
019-12-24 10:04:43		成功			详情
019-12-24 10:04:41	日志详情			×	详情
	执行设备	执行状态	失败原因		
	/Light	成功	success		

(可选)在**实例详情**页面监控信息页签,选择场景联动,查看已分配到边缘实例的场景联动监控信息。

单击场景名称对应操作栏中的查看,查看详细的监控信息。

边缘实例 > 实份详情 > 场景职动	
Rule_light_Close 通行中	
场景跃动信息 最近 1 天 🛛 🗸	RUBY
	异葉次数 ● ・ 异型の数(A)
1 0 1450 1650 1830 2050 2250 0050 0250 0450 0650 0850 1050 1250	0 1430 1650 1850 2050 2250 0050 0250 0450 0650 0850 1050 1230

3.4. 告警中心

物联网平台针对华东2(上海)地域的公共实例提供监控告警功能,处理规则引擎中场景联动规则触发的告 警信息。

前提条件

已创建场景联动规则,并应用到云端。具体操作,请参见云端场景联动。

操作步骤

- 1. 登录物联网平台控制台。
- 2. 在实例概览页,找到并单击公共实例。
- 3. 在左侧导航栏,选择监控运维>告警中心。
- 4. 单击告警配置页签, 单击创建告警规则。
- 5. 设置告警规则参数。

参数说明

参数	描述
告警规则名称	设置告警规则名称。支持中文、英文大小写字母、数字、下划线(_)和短划线 (-),长度限制4~32个字符。

参数	描述
告警规则描述	描述您的告警规则。可以为空。
告警级别	选择告警规则的告警级别。当前支持五个告警级别,告警严重程度关系为 1>2>3>4>5。
通知方式	 选择告警规则被触发后,通知告警处理人员的方式。 告警中心:告警规则被触发后,在告警列表页签下显示告警信息。默认全部告警信息都在告警列表页签下显示。 钉钉机器人:告警规则被触发后,在告警列表页签下显示告警信息的同时,以钉钉机器人的方式,发送消息到钉钉群。选择钉钉机器人后,需要填写钉钉机器人Webhook,具体操作,请参见钉钉官方文档。 说明 使用钉钉机器人通知方式时,请考虑钉钉机器人的如下限制情况: 每个钉钉机器人每分钟最多发送20条钉钉消息。如果超过20条,会限流10分钟。
关联触发场景	选择需要关联的场景联动规则。当所关联的场景被触发时 <i>,</i> 将会触发此告警。您也 可以单击 新增规则 ,新增一个场景联动规则,具体操作,请参见 <mark>云端场景联动</mark> 。

6. 单击确认,完成告警规则的创建。当告警规则的云端运行状态显示运行中,表示创建成功。

告警中心						
告警列表 告警配置						
创建告警规则						C
告譬规则名称	级别	告警规则描述	关联场景规则数量	云端运行状态	操作	
溫度异常告警	5		1	● 运行中	编辑 停止 删除	

可通过告警规则对应操作栏中的按钮,管理告警规则。

操作	描述
编辑	修改告警规则描述、通知方式或替换关联的场景联动规则。
停止或启动	停止或启动告警规则。
删除	删除该告警。

7. 在**告警列表**页签, 可查看当前的待处理告警或全部告警。

可通过告警对应操作栏中按钮,处理告警。

操作	描述
查看	查看该告警的详细信息。
忽略	忽略该告警。

操作	描述
处理	输入告警处理意见后单击确定。

⑦ 说明 已处理或已忽略的告警, 仅支持查看。

4.边缘应用

4.1. 什么是边缘应用

应用管理是Link IoT Edge提供的边缘应用管理能力,可以帮助您标准化的管理边缘端应用的版本、配置等。 当前,Link IoT Edge支持函数计算和容器镜像两种类型的边缘应用。

函数计算应用

函数计算应用是一种依托于阿里云函数计算服务的边缘应用类型,可以在云端完成代码开发,到边缘端执行 代码。函数计算应用继承了阿里云函数计算事件驱动的编程模型,同时作为Serverless计算框架,让您专注 于业务逻辑开发,无需为程序启动、消息流转、日志查询、进程保活等基础工作耗费精力。

函数计算应用的代码来源有两种:使用阿里云函数计算服务开发的函数和本地开发的函数,其中本地开发的函数必须遵循函数开发指南中的要求。

目前,标准版Link IoT Edge (LE Standard)不支持使用Python版本函数计算。

函数计算应用的示意图如下所示。



函数计算有众多特点和优势,可帮助您更好地管理边缘设备。

- 函数计算应用的优势:
 - 。 灵活的编程模型:边缘函数计算继承阿里云函数计算服务提供的按需运行模式功能,同时也提供持续 运行模式,将代码部署后选择持续运行,可以有效地支撑设备驱动和传统的服务端常驻程序的开发。
 - 高实时性:函数运行在边缘端网关,可以更靠近信息源进行数据的收集和分析,提升响应速度。
 - 。 离线运行:设备数据可传输到本地网络处理,不必连接到云端,从而在断网情况下设备依然可以正常工作。
- 函数计算应用的部分组成:
 - o 云端代码托管: 您可随时在云端对代码进行增、删、改、查,并热部署到网关并运行。
 - > 边缘端程序管理:负责管理部署到网关的函数,提供程序运行、服务保活、日志查询、性能监控、资源 监控和定时触发等。
 - 开发API: 边缘函数计算提供了设备驱动开发和应用开发的API, 降低程序开发成本。
- 触发函数计算应用的事件源:

在触发事件时, Link IoT Edge会调用函数代码中的handler函数进行处理,而按需运行模式的函数,只有当首次事件触发时才会被加载运行。能够触发函数计算的事件源如下:

- 。 定时触发:在物联网平台控制台将函数设置为定时运行,通过配置函数被触发的时间点(例如每分钟触发一次)触发函数计算。
- 消息触发:通过消息路由流转到函数计算的多种来源(例如设备、函数计算、流数据分析等消息源)的 消息,触发函数计算。
- 函数调用者:边缘网关提供调用本地函数计算的API,从外部程序中直接调用本地函数计算或在函数计算代码中调用其它的函数时,触发函数计算。

容器镜像应用

容器镜像应用是一种基于容器技术的边缘应用,可以直接从镜像仓库中拉取镜像作为边缘应用。

容器镜像应用的来源也有两种:阿里云容器镜像服务提供的镜像仓库和公共镜像仓库。

容器镜像应用的生命周期由Link IoT Edge托管,可通过边缘端OpenAPI访问Link IoT Edge的服务。OpenAPI 详情请参见<mark>边缘端HTTP API</mark>。



更多容器镜像应用相关介绍,请参见Docker官方文档。

边缘应用使用步骤

- 1. 创建边缘应用。
- 2. 分配边缘应用到边缘实例。

4.2. 新增自研应用

4.2.1. 函数计算应用

物联网边缘计算支持基于阿里云函数计算,创建函数计算类型的边缘应用。

前提条件

已创建了函数,具体操作,请参见阿里云函数计算。

操作步骤

1. 登录边缘计算控制台。

- 2. 左侧导航栏单击**应用管理**。
- 3. 在应用管理页面自研应用页签下,单击新增应用。
- 4. 设置应用参数,然后单击确认。
 - 应用信息

参数	描述
应用名称	设置您应用的名称,允许英文字母、数字和下划线(_),最多不超过128个字 符。
应用类型	有如下三种类型。
地域	选择您的函数计算所在的地域。
服务	选择您函数所在的服务。
函数	选择您已创建的函数。 〇 注意 使用Python语言开发的函数,仅支持在Link IoT Edge专业版实 例中使用。
授权	选择授权阿里云函数计算的角色。角色相关说明,请参见云资源访问。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同 的版本号。
版本描述	描述您的应用版本,例如版本功能等。可以为空。

○ 函数配置

参数	描述
启用默认配置	 是:使用系统默认的配置,无需设置其他函数配置参数和容器配置参数。 否:自定义函数配置,需要设置其他参数。

参数	描述
运行模式	 分为按需运行和持续运行两种模式。 按需运行:是函数计算原生的运行模式,函数部署到边缘端后不会立即运行,只有当事件触发时函数才会被加载运行,处于运行状态的函数一段时间内没有新的事件触发会停止运行。当您的程序是基于事件触发运行,且事件触发频率低于20分钟一次,建议选择按需运行,可以有效减少对边缘网关的CPU、内存等资源的压力。如果事件触发频率较高,按需运行的函数也会保持运行,以保证事件的响应速度。 持续运行:是边缘端特有的运行模式,函数部署到边缘端后会立即加载运行,用来满足传统的服务端程序常驻运行的需要。当您的程序是服务端程序,需要监听端口独立响应请求(如HTTP Server),请选择持续运行。Link IoT Edge 会负责函数的保活,当函数异常退出时,函数将会被重新运行。
超时限制	当事件触发时,会调用代码中handler接口处理事件。正常情况下事件处理完成会 发送应答消息返回处理结果,同时表明事件处理结束。如果函数超过此处的设定 时间仍未返回结果,则认为函数运行异常,此时函数会被重启。
定时运行	配置函数定时触发的时间点和频率,格式为Cron表达式。例如, * * * * * 表示每分钟触发一次函数, 20 * * * * 表示每小时的第20分钟触发一次 函数。详细信息,请参见CRONT AB 网页。
环境变量	自定义配置的环境变量,在函数代码运行时可以直接读取到。单击 新增环境变 量,通过设置变量名称、变量值增加环境变量。最多可配置10条环境变量。 该功能是为了做到代码和配置的分离,增强代码的可移植能力。例如,函数代码 需要请求网络上其它主机的服务,在不同环境下服务器的IP地址和端口通常是不同 的。通过环境变量将IP和端口信息作为配置参数传入,不同环境下的网关就可以共 用同一份函数代码。

。 容器配置

⑦ 说明 若自定义配置容器配置,需将函数配置区域下的启用默认配置选择为否。

参数	描述
是否使用宿主机host模	选择是否隔离容器的网络。直接使用宿主机网络环境。
式	 是:表示不隔离容器的网络,直接使用宿主机网络环境。 否:表示隔离容器的网络,需要设置网络端口映射。

参数	描述
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境 是完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某 一个端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多 支持映射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提 供服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问 到 fc-http-server 函数。因此需要将 fc-http-server 函数所在的容器 端口映射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机8080端 口),允许其它主机上的客户端程序通过访问 宿主机的IP地址: 8080 ,访问 到容器内部的 fc-http-server 函数。
是否启动特权模式	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。 ⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所 有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔 离的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运 行函数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时, 需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 当容器内存使用超过限制时,容器会被重启。 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

5. 创建完成应用后,可在**应用管理**页面**自研应用**页签中查看。您也可以通过**操作**栏按钮,进行更多操 作。

应用管理							
官方应用	自研应用						
新増应用	请输入应用名称 Q						C
应用名称		来源	开发语言	最新版本	版本描述	操作	
10.000		100-0	-	1.0.0	-	版本管理 删除	

○ 版本管理

单击**版本管理**,可对该应用的版本进行新增、修改和删除操作。

版本管理				×
应用名和 _{来源}	称:	onida R	最新版本 1.0.0	
请输入版本	本号	Q		新增版本
ID	版本号	版本描述	创建时间	操作
1	1.0.0	-	2019-11-11 19:04:46	编辑 删除
关闭				

○ 删除

单击**删除**,可删除该应用。

⑦ 说明 删除应用前,需要先删除其下所有版本的应用,否则应用不可删除。

后续步骤

将已创建好的边缘应用分配到边缘实例,详细操作,请参见分配边缘应用到边缘实例。

4.2.2. 容器镜像应用

本文主要介绍如何创建容器镜像类型的边缘应用。

前提条件

- 已开通容器镜像服务ACR。
- 若使用阿里云私有仓库中的镜像创建应用,则需要根据容器镜像文档构建镜像。

操作步骤

- 1. 登录边缘计算控制台。
- 2. 在应用管理页面自研应用页签下,单击新增应用。
- 3. 设置应用参数,然后单击确认。
 - 应用信息

参数	描述
应用名称	设置您应用的名称,允许英文字母、数字和下划线(_),最多不超过128个字 符。

参数	描述
应用类型	有如下三种类型。 函数计算:使用阿里云函数计算创建应用。 容器镜像:使用容器镜像服务ACR创建应用。 本地上传:使用本地开发的函数。 此处选择容器镜像。
仓库类型	选择镜像仓库的类型。有如下两种选项: 公共仓库:使用公共仓库中的镜像,创建应用。 阿里云私有镜像:使用阿里云容器镜像服务中的镜像,创建应用。
镜像地址	当仓库类型为公共仓库时出现的参数。填写带有镜像标签(tag)的公共镜像地址。填写格式示例: mysql:latest 。
选择地域	当仓库类型为阿里云私有仓库时出现的参数。选择您私有仓库所在的地域。
命名空间	当仓库类型为阿里云私有仓库时出现的参数。选择您私有仓库所在的命名空间。
镜像仓库	当仓库类型为阿里云私有仓库时出现的参数。选择您的私有仓库。
镜像版本	当仓库类型为阿里云私有仓库时出现的参数。选择您私有仓库下的镜像版本。
授权	选择授权阿里云容器镜像服务的角色。角色相关说明,请参见 <mark>云资源访问</mark> 。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同 的版本号。
版本描述	描述您的应用版本,例如版本功能、用途等。可以为空。
环境变量	自定义配置的环境变量,在容器应用代码运行时可以直接读取到。单击 新增环境 变量,通过设置变量名称、变量值增加环境变量。最多可配置10条环境变量。

○ 容器配置

参数	描述				
Entrypoint	可选参数。如果指定该参数,将作为容器启动的第一个命令,字符串长度≤128字 节。				
命令行参数(Cmd)	可选参数。为Entrypoint提供参数,最多可配置10条命令行参数,每条字符串长 度≤128字节。				
	例如,让容器 ping 通 <i>aliyun.com</i> 一千次,那么命令为 ping aliyun.com -c 1 000 。此时:				
	Entrypoint=ping				
	■ 第一个命令行参数=aliyun.com				
	■ 第二个命令行参数=-c				
	■ 第三个命令行参数=1000				

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 是:表示不隔离容器的网络,直接使用宿主机网络环境。 香:表示隔离容器的网络,需要设置网络端口映射。
网络端口映射	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境 是完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某 一个端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多 支持映射10条网络端口。 例如,运行在宿主机容器内的 fc-http-server 函数,通过监听Socket的80 端口对外界提供服务。此时,其它主机上的客户端程序访问宿主机的80端口,是 无法直接访问 fc-http-server 函数。因此需要将 fc-http-server 函数 所在的容器端口映射到宿主机的某一个端口上(例如将容器内的80端口映射到宿 主机8080端口),允许其它主机上的客户端程序通过访问 宿主机的IP地址:80 80 ,访问到容器内部的 fc-http-server 函数。
是否启动特权模式	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。 ⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔 离的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运 行函数的容器内部。 最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时, 需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 当容器内存使用超过限制时,容器会被重启。 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

4. 创建完成应用后,可在**应用管理**页面**自研应用**页签中查看。您也可以通过**操作**栏按钮,进行更多操 作。

应用管理					
官 方应用 自研 应用					
新编应用 请输入应用名称 Q					C
应用名称	来源	开发语言	最新版本	版本描述	操作
August America	00010		1.0.0		版本管理 删除

○ 版本管理

单击**版本管理**,可对该应用的版本进行新增、修改和删除操作。

版本管理					×
应用名称: _{来源}	iption de Maria	ditta .	最新版本	1.0.0	
请输入版本专	3	Q			新增版本
ID	版本号	版本描述	创建时间		操作
1	1.0.0	-	2019-11-11 19:	04:46	编辑 删除
关闭					

○删除

单击删除, 可删除该应用。

⑦ 说明 删除应用前,需要先删除其下所有版本的应用,否则应用不可删除。

后续步骤

将已创建好的边缘应用分配到边缘实例,详情请参见分配边缘应用到边缘实例。

4.2.3. 本地应用

物联网边缘计算支持在本地开发函数后,可通过本地函数创建函数计算类型的边缘应用。

前提条件

已在您的本地开发完成函数,具体开发操作请参见函数开发指南。

操作步骤

- 1. 登录边缘计算控制台。
- 2. 在应用管理页面自研应用页签下,单击新增应用。

3. 设置应用参数,然后单击确认。

应用信息

参数	描述
应用名称	设置您应用的名称,允许英文字母、数字和下划线(_),最多不超过128个字 符。
应用类型	有如下三种类型。 ■ 函数计算:使用阿里云函数计算创建应用。 ■ 容器镜像:使用容器镜像服务ACR创建应用。 ■ 本地上传:使用本地开发的函数。 此处选择本地上传。
开发语言	选择您的本地函数的开发语言。目前支持Node.js、Python、C三种开发语言。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同 的版本号。
代码包	单击上传文件,上传您在本地已开发完成的代码包。
版本描述	描述您的应用版本,例如版本功能等。可以为空。

容器配置

参数	描述
是否使用宿主机host模 式	选择是否隔离容器的网络。直接使用宿主机网络环境。 是:表示不隔离容器的网络,直接使用宿主机网络环境。 否:表示隔离容器的网络,需要设置网络端口映射。
	当是否使用宿主机host模式为否时出现的参数。函数的网络环境和宿主机的环境 是完全隔离的。通过网络端口映射,将容器内函数的监听端口映射到宿主机的某 一个端口上,实现不同主机上的客户端程序,能够访问该函数提供的服务。最多 支持映射10条网络端口。
网络端口映射	例如,运行在宿主机容器内的 fc-http-server 函数,通过80端口对外界提供服务。此时,其它主机上的客户端程序,无法通过访问宿主机的80端口,访问 到 fc-http-server 函数。因此需要将 fc-http-server 函数所在的容器 端口映射到宿主机的某一个端口上(例如将容器内的80端口映射到宿主机8080端 口),允许其它主机上的客户端程序通过访问 宿主机的IP地址: 8080,访问 到容器内部的 fc-http-server 函数。

参数	描述
是否启动特权模式	容器内的root用户实际上只是宿主机的一个普通用户。若在容器内部做修改系统 时间、使用mount命令等需要root权限的操作,则需要赋予容器privileged特权。
	⑦ 说明 特权模式下,容器内部拥有宿主机的root权限,而且宿主机的所 有设备会默认映射到容器内部,即无需配置设备映射。
设备映射	当是否启动特权模式为否时出现的参数。设备管理系统和宿主机的环境是完全隔 离的。当一个函数需要访问宿主机的设备(例如串口)时,需要将设备映射到运 行函数的容器内部。最多可添加10个设备映射。
卷映射	文件系统和宿主机的环境是完全隔离的。当一个函数需要访问宿主机的文件时, 需要将文件映射到运行函数的容器内部。最多可添加10个卷映射。
内存限制	设置容器的内存上限。 当容器内存使用超过限制时,容器会被重启。 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

创建完成应用后,可在应用管理页面自研应用页签中查看。您也可以通过操作栏按钮,进行更多操作。

1		在在 T 田
M	ΗH	EI#

应用自连						
官方应用 <u>自研应用</u>						
新境边用 请输入应用名称 Q						C
应用名称	来源	开发语言	最新版本	版本描述	操作	
Automatic America		-	1.0.0	-	版本管理 删除	

○ 版本管理

单击**版本管理**,可对该应用的版本进行新增、修改和删除操作。

版本管理	1			×
应用名和 _{来源}	称:	Ondellar B	最新版本 1.0.0	
请输入版本	本号	Q		新增版本
ID	版本号	版本描述	创建时间	操作
1	1.0.0	-	2019-11-11 19:04:46	编辑 删除
关闭				=

○ 删除

单击**删除**,可删除该应用。

⑦ 说明 删除应用前,需要先删除其下所有版本的应用,否则应用不可删除。

后续步骤

将已创建好的边缘应用分配到边缘实例,具体操作请参见分配边缘应用到边缘实例。

4.3. 分配边缘应用到边缘实例

本文介绍将已创建的边缘应用分配到边缘实例的步骤。通过分配应用到边缘实例,您可以更好的管理边缘端业务。

前提条件

已根据新增边缘应用内容,创建完成边缘应用。

操作步骤

- 1. 登录边缘计算控制台。
- 2. 单击左侧导航栏中边缘实例。
 找到要分配边缘应用的边缘实例,单击对应操作栏中的查看。
- 3. 在实例详情页面,选择边缘应用,然后单击分配应用。
- 在弹出的分配应用页面中,选择要分配给边缘实例的应用,单击右侧的分配。
 分配成功后,单击关闭。

边缘计算 / 边缘实例 / 实例详情			分配应用			×
← LinkloTEdge_Node			请输入应用名称	Q		
		政分版平写	应用名称	应用来源	版本	操作
网关 监控信息 设备与驱动 场票联动 边缘应用 流数据分析	消息路由 日志服务	设置	1000	容器镜像	1.0.0 🗸	分配
⑥ 函数计算功能全新升级,现支持函数计算、本地上传、容器器像三种类型应用了解更多				函数计算	1.0.0 🗸	分配
分配应用						
应用名称 应用来源 启用版本		运行模式				
	尚未分配任何应用					
	Link loT Edge 可以将计算从 行,断网情况下仍然可以保证	、云鳞扩展到边缘,在边 还业务应用连续运行。应				
	支持函数计算、本地上传和语	容器镜像三种类型。				
	分配应用 了解边线	家应用				
			关闭			

完成应用分配后,便可在该边缘实例的实例详情页面的边缘应用页签下,查看边缘应用信息。

- 5. 部署边缘实例。单击实例详情页面右上角的部署,并在弹出对话框中,单击确认。
- (可选)在**实例详情**页面监控信息页签,选择边缘应用,查看已分配到边缘实例的边缘应用监控信息。

单击应用名称对应操作栏中的查看,查看详细的监控信息。

边碌计算 / 边缘实例 / 实例評情 / 监控评情 ← lightSFilter 运行中		
最近 1 小时 🗸		C
调用次数 💿	执行时间 💿	
	1 -	
0 19:10 19:20 19:30 19:40 19:50 20:00 ● 调用次数(次)	0 19:10 19:20 19:30 19:40 19:50 20:00 ● 平均执行时间(骤秒)	_
退出次数 ④		
0		
 退出次数(次) 		#

4.4. 应用场景

4.4.1. 设备上云数据筛选

本文以使用官方示例驱动的LightSensor设备为例,讲述如何筛选设备上报到云端的数据。

前提条件

- 请您确保已完成边缘实例的创建。具体操作步骤,请参见专业版环境搭建或标准版环境搭建。
- 创建光照度传感器产品以及该产品下的Light Sensor设备,并将Light Sensor设备分配至边缘实例。具体操作步骤,请参见示例驱动。

背景信息

通常情况传感器等设备会持续不断的上报采集到的数据,如温度计上报温度值、光照度传感器上报光照度等,这些数据往往差异不大,我们通常只关注超出阈值的数据。物联网边缘计算提供边缘函数计算,进行设备上云数据筛选,可以减少设备数据上云的数量,从而减少云端的成本。

本文中以LightSensor设备为例,因此在使用函数计算处理数据前,您可以在物联网平台控制台左侧导航栏选 择设备管理 > 设备,找到LightSensor设备,单击设备名称后的查看,进入设备详情页面的物模型数据 > 运行状态页签,查看设备状态。

物联网平台 / 设备管理 /	设备 / 设备详情		
← LightSe	查看数据		×
ProductKey a1	1 小时 🗸		图表 表格
设备信息 Topie	时间	原始值	
运行状态事件	2020/05/18 16:00:39.185	400	Â
实时刷新 🕕	2020/05/18 16:00:37.183	300	- 11
光照度检测值	2020/05/18 16:00:35.180	200	- 11
200 Lux	2020/05/18 16:00:33.178	100	- 11
2020/05/18 16:01:1	2020/05/18 16:00:31.163	200	- 11
	2020/05/10 16:00:20 160	200	-
			关闭

上图中,单击查看数据,可以看到数值在100~600之间,以100倍数的规律不断变化,且以每隔2秒的高频 率上报一次数据,日积月累会产生一些不必要的费用。因此可以使用函数计算来处理设备上报云端的数据, 即对数据进行筛选。

步骤一: 创建数据筛选函数

- 1. 下载设备数据筛选函数deviceDataFilter.zip。该函数用于筛选LightSensor设备上报的光照度数据,仅当 光照度值大于500或小于200时,才将数据上报给云端。
- 2. 登录函数计算控制台。

如尚未开通该服务,请阅读并选中我已阅读并同意,单击立即开通,开通服务。

 (可选)在左侧导航栏单击服务及函数,在服务及函数页面服务列表区域,单击新增服务,创建一个 服务。

其中, 服务名称必须填写, 此处设置为EdgeFC, 其余参数可根据您的需求设置, 也可以不设置。

⑦ 说明

- 若您首次在函数计算中创建服务,请根据配置向导配置参数。
- 若已操作过其他应用场景示例或小程序示例,即已创建EdgeFC服务,则无需重复创建。

4. 创建服务成功后,在服务及函数页面的EdgeFC区域,单击新增函数。

5. 在新建函数页面,单击事件函数区域中的配置部署。

6. 设置数据筛选函数的基础管理配置参数。

参数	描述
函数类型	保持默认选项。
所在服务	选择已创建的EdgeFC服务。
函数名称	设置为lightSensorDataFilter。
运行环境	设置函数的运行环境并选择上传代码的方式。此示例中选择Node.JS 8.x。 在 上传代码 右侧选择 上传代码包 ,单击 上传代码 ,上传步骤1中下载 的deviceDataFilter.zip代码包。
函数入口	使用默认值index.handler。

其余参数的值请根据您的实际需求设置,也可以不设置。更多信息,请参见函数计算。

确认函数信息后,单击**新建**完成操作。

7. 创建函数完成后,系统自动跳转到函数详情页面。您可在**代码执行**页签的**代码执行管理**区域下,选 中**在线编辑**单选框,查看源码。

← lightSensorData	Filter	服务版本 > LATEST > 図 日志	辿 监控 □ ARN
概览 代码执行 触发器	日志查询 函数指标 链路追踪 异步配置		
代码执行管理			
执行 触发事件 ?			
 在线编辑 OSS上传 (代码结 	ュ上传 ○ 文件夹上传		使用旧版编辑器
🆘 File Edit Selection Vi	iew Go Help	► Invoke	∰ Event 🕂 🖉
EXPLORER .	index.js x		
< C index js	<pre>11 * distributed under the License is distributed on an "AS IS 12 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either expr 13 * See the License for the specific language governing permi 14 * limitations under the License. 15 */ 16 17 /* 18 * Function description: 19 * This function is used to filter the data reported by the 20 * Report the device data to cloud only when illuminance val 21 */ 23 'use strict'; 24 25 const lesdk = require('linkedge-core-sdk'); 26 const lesdk = new lesdk.IoTData(); 27 var function_invoke_times_g = 0; 28 29 module.exports.handler = function (event, context, callback); 30 function_invoke_times_g+; 31 console.log(' Receive event: \${event.toString()}'); 31</pre>	<pre>S" BASIS, ress or implied. issions and LightSensor device. lue is greater than 500 or less than 200.) { g}}`);</pre>	<pre>Microsoftware International Systems International Systems Int</pre>

⑦ 说明 light SensorDat aFilter样例代码分为如下三步。

i. 从收到的设备上报数据(event参数)中解析出光照度的数值。

```
var illuminance = iotData.getThingPropertyByEvent(event, "MeasuredIlluminance")
;
```

ii. 判断数据是否满足数据上云的筛选条件。

if (illuminance > 500 || illuminance < 200)

iii. 上报满足条件的数据。

```
iotData.publish(message, (err, data) => {callback(err);});
```

8. (可选)调整数据过滤参数。

创建函数完成后,可以在线编辑代码,更改数据过滤条件。

样例代码默认的数据过滤条件如下,表示将函数分配到边缘实例并部署实例后,函数只过滤设备收集到的光照度大于500或者小于200的数据,显示在设备的**设备详情**页面**物模型数据 > 运行状态**页签中。

if (illuminance > 500 || illuminance < 200)

您可以更改数据过滤条件,例如函数只过滤设备收集到的光照度大于450的数据,则将数据过滤参数改 为如下内容:

if (illuminance > 450)

步骤二:分配函数到边缘实例

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击应用管理。
- 使用步骤一中已创建的函数,创建函数计算类型的边缘应用。具体操作,请参见函数计算应用。
 应用信息参数说明如下:

参数	描述
应用名称	设置您应用的名称,例如lightSensorDataFilter。
应用类型	选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择EdgeFC服务。
函数	选择lightSensorDataFilter函数。
授权	选择AliyunIOTAccessingFCRole。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。

函数配置区域框中, 启用默认配置选择是, 则其余参数无需设置。

- 4. 在左侧导航栏单击边缘实例。
- 5. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 6. 在**实例详情**页面的边缘应用页签,单击分配应用。
- 7. 在**分配应用**面板,找到已创建的数据筛选函数lightSensorDataFilter,单击对应操作栏中的分配,然 后单击关闭。

步骤三:配置消息路由

添加消息的详细步骤及各个参数的解释。详细信息,请参见设置消息路由。

1. 在**实例详情**页面,单击消息路由页签,移除该实例中设备路由到IoTHub的消息路由。

⑦ 说明 在操作"前提条件"中的示例驱动内容,创建LightSensor设备并分配到边缘实例时,为 边缘实例配置了消息路由,因此需要在此步骤移除该消息路由。

网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置		
分配路由	了解消息	息路由?								
路由名称		消息来源				消息主题过滤		消息目标 分配时间 操作		
(默认路由	日)	设备(全部)				全部		loT Hub	编辑 移除	

2. 单击**分配路由**,添加LightSensor设备到函数计算的消息路由。

按照界面提示,设置如下参数,参数设置完成后,单击确定完成第一个路由的添加。

参数	描述
路由名称	设置一个消息路由名称。
消息来源	此处选择 设备 ,选择 光照度传感器 > LightSensor 。
消息主题过滤	此处选择全部。
消息目标	此处选择 边缘应用 和EdgeFC/lightSensorDataFilter函数。

- 3. 在实例详情页面,单击设备与驱动页签,单击Light Sensor设备后的查看,界面跳转到设备详情页面。
- 4. 选择Topic列表 > 物模型通信 Topic 页签, 复制列表中第一条设备属性上报的Topic。

•	← LightSensor									
1	*品 ^分	化照度传感器 查看					DeviceSec	ret ******** 🛓	ē音	
1	ProductKey	复制								
	设备信息 Top	pic列表 物模型数据 设备影子 文件管理 日志服务 在线调试								
	基础通信 Topic 物模型通信 Topic 自定义 Topic									
-	物根型通信 Topic 列表									
	功能	Торіс美						操作权限	描述	
	Rid. L47	/sys/ /Ligh	htSensor/thing/eve	ent/property/post	1			发布	设备属性上报	
	刷社上版	/sys///LightSensor/thing/event/property/post_reply						订阅	云端响应腐性上报	

返回到**实例详情**页面,单击**分配路由**,添加函数计算到IoT Hub(云端)的消息路由。
 按照界面提示,设置如下参数,参数设置完成后,单击确定完成第二个路由的添加。

参数	描述
路由名称	设置一个消息路由名称。
消息来源	此处选择函数计算和EdgeFC/lightSensorDataFilter函数。
消息主题过滤	此处填写上一步复制的设备属性上报的Topic。
消息目标	此处选择IoT Hub。
服务级别	此处选择0。

步骤四: 部署边缘实例

用户指南·边缘应用

- 1. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。
- 2. 在**实例详情**页面,单击**设备与驱动**页签,然后单击LightSensor设备后的查看,系统会跳转到物联网平 合控制台设备详情页面。

在设备详情页面,选择物模型数据 > 运行状态页签,查看LightSensor设备的运行状态以及数据。

产品	光照度传感	器 查看	查看数据		×
ProductKey		复制	1 小时 🗸		图表表格
设备信息	Topic列表	物模型	时间	原始值	
运行状态	事件管理	服务调用	2020/04/26 下午8:31:14	500	
实时刷新 🕕					
光照度检测	值				
500 Lui	x				
2020/04/26	下午8:31:14				
					关闭

至此您已经完整地体验了使用边缘函数计算实现设备上云数据筛选功能。

4.4.2. 阿里云OSS服务访问

本文以常用的访问阿里云对象存储(OSS)服务为例,详细讲解如何使用边缘函数计算访问阿里云服务,完成本地文件上传和云端文件下载能力。

前提条件

- 已开通OSS服务,若未开通,请参见<mark>开通OSS服务</mark>内容开通。
- 请您确保已完成边缘实例的创建,具体操作,请参见专业版环境搭建或标准版环境搭建。

背景信息

物联网边缘计算提供边缘函数计算能力,更方便地访问阿里云各种云服务,支持将设备在边缘端运行时生成的各类文件(统计报表、日志、图片、视频等)传输到云端,或从云端下载配置文件作为程序的输入来影响边缘端程序的行为。通过边缘函数计算访问OSS服务,可以选择永久存储设备数据。

使用边缘函数计算访问阿里云各种云服务的操作流程主要分为三步:

- 1. 访问阿里云官网,开通对应的云服务,下载云服务SDK,将SDK和函数计算代码一起打包成.zip文件。
- 2. 访问阿里云函数计算控制台,创建函数,上传.zip文件。
- 3. 访问物联网平台控制台, 分配函数到边缘实例, 部署函数到边缘网关运行。

步骤一: 创建OSS Bucket

1. 创建存储空间,具体操作,请参见创建存储空间。

本示例中部分参数按如下表设置,其余参数可以使用默认值。

参数	描述						
Bucket 名称	设置为le-fc-bucket。						
	从下拉框中选择该存储空间的数据中心。 可以选择离您近的数据中心,访问速度更有保障。						
地域	注意 选择地域后,该参数下方会显示您已选地域对应的Endpoint值, 请复制该值到本地文件中,用于后续编辑函数代码。						

- 2. 在本地创建一个名为*ossCloudFile.txt*的TXT格式的文件,文件内容可随意设置。 该文件为后续通过函数下载OSS文件,提供文件源。
- 3. 将ossCloudFile.txt文件上传到已创建的le-fc-bucket存储空间中,具体操作,请参见上传文件。

步骤二: 创建访问OSS的函数

- 1. 下载访问OSS函数accessAliOSS-code.zip。该函数用于访问OSS,完成本地文件上传OSS和云端文件下 载到本地的操作。
- 2. 登录函数计算控制台。

如尚未开通该服务,请阅读并选中我已阅读并同意,单击立即开通,开通服务。

 (可选)在左侧导航栏单击服务及函数,在服务及函数页面服务列表区域,单击新增服务,创建一个 服务。

其中, 服务名称必须填写, 此处设置为EdgeFC, 其余参数可根据您的需求设置, 也可以不设置。

? 说明

- 若您首次在函数计算中创建服务,请根据配置向导配置参数。
- 若已操作过其他应用场景示例或小程序示例,即已创建EdgeFC服务,则无需重复创建。
- 创建服务成功后,在服务与函数页面的EdgeFC区域,单击新增函数,然后单击事件函数区域中的配置 部署。
- 5. 设置访问OSS函数的基础管理配置参数。

参数	描述
函数类型	保持默认选项。
所在服务	选择已创建的EdgeFC服务。
函数名称	设置为accessAliOSS。
运行环境	设置函数的运行环境,此示例中选择Node.JS 8.x。 在 上传代码 右侧选择 上传代码包 ,单击上传代码,上传步骤1中下载 的 <mark>accessAliOSS-code.zip</mark> 代码包。
函数入口	使用默认值index.handler。

其余参数的值请根据您的实际需求设置,也可以不设置。更多信息,请参见函数计算。

确认函数信息后,单击**新建**完成操作。

6. 创建函数完成后,系统自动跳转到函数详情页面。您可在**代码执行**页签的**代码执行管理**区域下,选 中**在线编辑**单选框,查看源码。

← accessAliOSS								服	务版本 🗸 🗸	LATEST	~	☑ 日志	匝 监控	(ARN
概览	代码执行	触发器	日志査询	函数指标	链路追踪	异步配置									
代码	好管理														
执行	执行 触发事件 ?														
 在线 	鳥編 ○ OSS上信	(代码包	上传 🔾 文件	夹上传										使用	日版编辑器
¢>	File Edit Se	election Vie	w Go Help)								► Invoke	🕷 Event	+	2
ß	<pre>EXPLORER</pre>					<pre>clal.</pre>	%s, Secur urityToker service. un/localFi	<pre>ity Token); le.txt"); romCloud.</pre>	: %s', txt");				The second		

其中:

- 将<Your OSS Region>替换为创建OSS Bucket中,创建存储空间时保存到本地的Endpoint值。
- 将<Your OSS Bucket>替换为创建OSS Bucket中创建存储空间时设置的Bucket名称,即le-fcbucket。
- 7. 登录前提条件中已完成的网关,执行如下命令生成本地测试文件,为样例代码中的本地上传文件功能, 准备一个名为*localFile.txt*的本地文件。

sudo echo "Hi, this is the file from edge GW." > /linkedge/run/localFile.txt

⑦ 说明 在函数计算代码中访问网关宿主机的文件时,对文件所在的路径有约束。在Link IoT Edge标准版(LE Standard)中,函数计算的代码是运行在与宿主机隔离的文件系统环境中,只有/ *linkedge/run*目录是函数计算和宿主机环境共享并且拥有可读写权限。所以在测试时,请把测试文件的路径指到/*linkedge/run*目录下。

8. 返回函数计算控制台,在accessAliOSS函数的代码执行页签代码执行管理区域中,单击保存。

9. (可选)如需体验下载云端OSS文件到本地功能,可以在accessAliOSS函数的**代码执行**页签在线编辑框中,注释掉第45行的 client.put 代码,去掉第48行开头的注释符 "//",调用OSS SDK的get方法 (client.get('ossCloudFile.txt', "/linkedge/run/fileFromCloud.txt");)。

其中*ossCloudFile.txt*是创建OSS Bucket中上传到le-fc-bucket存储空间的文件名,/linkedge/run/file FromCloud.txt是文件下载后保存在边缘端的文件名。

- ⑦ 说明 样例代码解读如下,其逻辑分为三步:
 - 1. 获取Credential, 作为访问云服务的临时Token。

credChain.resolvePromise()

2. 填写您的OSS Region信息和OSS Bucket信息。

region: '<Your OSS Region>', bucket: '<Your OSS Bucket>',

3. 调用OSS SDK的put方法,将本地文件上传到OSS。

client.put('fileFromEdge.txt', "/linkedge/run/localFile.txt");

其中*fileFromEdge.txt*是文件上传后在云端保存的文件名, /*linkedge/run/localFile.txt*是本地需要上传的文件, 两个文件名都可根据需求做修改, 前提是本地文件必须存在。

步骤三:分配函数到边缘实例

首先,需要为边缘实例分配包含管理对象存储服务(OSS)权限权限的角色,或者在已分配到边缘实例的 角色中添加管理对象存储服务(OSS)权限权限。具体操作,请参见云资源访问。

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击应用管理。
- 使用本文上方步骤二中已创建的函数,创建函数计算类型的边缘应用。具体操作,请参见函数计算应用。
 应用信息参数说明如下:

参数	描述
应用名称	设置您应用的名称,例如accessAliOSS。
应用类型	选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择EdgeFC服务。
函数	选择accessAliOSS函数。
参数	描述
------	--
授权	选择AliyunIOTAccessingFCRole。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。

函数配置说明如下:

参数	描述
启用默认配置	选择否。
运行模式	运行模式有两种。此处选择 持续运行 模式。程序部署后会立即执行。
超时限制(秒)	函数收到事件后的最长处理时间,此处使用默认值5秒。如超过该时间函数仍未返回结果,该函数计算程序将会被强制重启。
定时运行	打开定时运行开关,文本框中填入 * * * * * 。表示该函数会被定时触发运 行,每分钟执行一次。Cron表达式详细信息请参见CRONT AB 网页内容。

其余参数无需配置。

- 4. 在左侧导航栏单击边缘实例。
- 5. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 在实例详情页面,选择设置,单击添加角色与权限,为边缘实例分配AliyunIOTAccessingFCRole角色。操作详情,请参见云资源访问。
- 7. 在实例详情页面的边缘应用页签,单击分配应用。
- 8. 在**分配应用**面板,找到已创建的访问OSS的函数accessAliOSS,单击对应操作栏中的**分配**,然后单击**关** 闭。

步骤四: 部署边缘实例

- 1. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。
- 实例部署成功约一分钟后,可以在对象存储控制台,le-fc-bucket > 文件管理页面,可以看到*fileFro mEdge.txt*文件已经成功上传。

可以选择文件右侧的更多 > 下载, 将文件下载到PC上查看文件内容。

对象存储 / le-fc-buck	et / 文件管	理										(⑦ 通过 SC	XK 管理文件
le-fc-buc	ket							读写权限 私有	类型	低频访问 (本地冗余)	区域 华东2 (上海)	创建时间 2	2020年4月	27日 17:39
概览		上传文件	新建目录	碎片管理	授权	批量操作 ∨	刷新					请输入文件符	名前缀匹配	Q
文件管理	>		文件名					文件大小	N	存储类型	更新时间			操作
权限管理	>	T	fileFromEdge	.txt				0.016KI	3	低频访问	2020年4月27日 17:53	1	详情	<u>更多</u> へ
基础设置	>												标签	
冗余与容错	>												设置 HT	TP 头
传输管理	>												し て 戦	11XPR
日志管理	>												选取内容	;
数据处理	>												删除	
数据统计	>												修改存储	类型

至此您已经完整地体验了使用边缘函数计算实现阿里云OSS服务的访问功能。

常见问题

如果在OSS文件管理页没有看到fileFromEdge.txt文件,可能有如下几种原因:

- 函数触发时间还没到:函数定时每分钟运行一次,如果刚部署完成,函数有可能还没有运行。请等待2分钟刷新页面查看。
- 网关本地系统时间不正确:本地时间可通过 date 命令查看,如果时间和网络时间差距过大,OSS会拒 绝请求。
- 本地待上传文件不存在:如果在网关上执行 ls /linkedge/run/localFile.txt 命令,显示文件不存 在,请参考本文上方生成本地测试文件内容创建。
- OSS配置信息填写有误:请参考本文上方在线编辑代码内容填写正确的OSS配置信息。
- 如果上述配置都检查无误,可通过 cat /linkedge/run/logger/fc-base/accessAliOSS/log.INFO 命令
 查看函数运行日志,定位问题。

4.4.3. 本地过程控制

本文以一个典型的过程控制场景为例,描述如何使用边缘函数计算实现较复杂的本地过程控制。

前提条件

- 本示例仅适用于Link IoT Edge专业版,请您确保已完成边缘实例的创建。详细操作,请参见专业版环境搭建。
- 已创建光照度传感器、客厅灯产品以及该两个产品下的Light Sensor、Light设备,并将设备分配至边缘实例。详细操作,请参见示例驱动。
- 此场景需要本地数据库存储助手,将传感器上报的数据存储到网关的本地数据库中,因此请提前完成本地 数据库存储助手内容。

背景信息

在家居、石油、电力等传统物联网场景中,现场的各类传感器负责实时采集数据,通过3G、4G、有线宽带 等数据链路,将数据传输至远端数据中心服务器。远端数据中心服务器进行数据格式转换、处理、分析和运 算。处理的结果及生成的控制指令再通过公网传输至现场仪表层,实现过程控制。在生产中也会需要将传感 器的历史数据结合行业内部的算法模型(如线性回归、Isolation Forest等),分析和判断现场状态,进而控 制其他设备的行为。针对此类场景,您可以使用物联网边缘计算提供的边缘函数计算,实现本地过程控制。

物联网边缘计算,致力于构建近端边缘层与云端计算层协同处理的新模式。对于自动化领域来说,近端边缘 层可以对传感器数据进行处理或纠正,对传感器的故障进行预判断、预处理,完成对现场的控制和故障诊 断,同时将经过修正的数据实时上传到云端进行计算,完成一个立体分层式的数据处理过程。

本文中以LightSensor传感器和Light设备为例,获取LightSensor传感器在5分钟内上报的所有数据(历史数据),当其平均值超过某阈值时,关闭Light设备。

步骤一: 创建过程控制函数

- 1. 下载本地过程控制函数querySqliteDB-code.zip。
- 2. 登录函数计算控制台。

如尚未开通该服务,请阅读并选中我已阅读并同意,单击立即开通,开通服务。

 (可选)在左侧导航栏单击服务及函数,在服务及函数页面服务列表区域,单击新增服务,创建一个 服务。

其中, 服务名称必须填写, 此处设置为EdgeFC, 其余参数可根据您的需求设置, 也可以不设置。

⑦ 说明

- 若您首次在函数计算中创建服务,请根据配置向导配置参数。
- 若已操作过其他应用场景示例或小程序示例,即已创建EdgeFC服务,则无需重复创建。

4. 创建服务成功后,在服务及函数页面的EdgeFC区域,单击新增函数。

5. 在新建函数页面,单击事件函数区域中的配置部署。

6. 设置过程控制函数的基础管理配置参数。

参数	描述
函数类型	保持默认选项。
所在服务	选择已创建的EdgeFC服务。
函数名称	设置为querySqliteDB。
运行环境	设置函数的运行环境并选择上传代码的方式。此示例中选择Python 3。 在 上传代码 右侧选择 上传代码包 ,单击上传代码,上传步骤1中下载 的querySqliteDB-code.zip代码包。
函数入口	使用默认值index.handler。

其余参数的值请根据您的实际需求设置,也可以不设置。更多信息,请参见函数计算。

确认函数信息后,单击**新建**完成操作。

7. 创建函数完成后,系统自动跳转到函数详情页面。您可在**代码执行**页签的**代码执行管理**区域下,选 中**在线编辑**单选框,查看源码。

函数计算	函数计算	隼 / 服务·函数 /	EdgeFC ∽ / querySqliteD8 ∽		产品动态	帮助文档	
相近	←q	uerySqli	teDB	服务版本:	ATEST 🗸	I	
服务-函数		2 1					
白金以城名	概览	6 代码执行	触发器 日志查问 函数指标 AF	RN - acs:fc: :services/EdgeFC/functi	ons/queryS	qliteDB 🚺	
mail.com	任职物名	行管理					
资源中心	1.04-33/01	13 1944					
最近访问函数 ^	执行	脫发事件	0				
EdgeFC/querySqliteDB	 在 		SS上传 C 代码包上传 C 文件乘上传		使用		
EdgeFC/accessAliOSS	450	File Edit S	Selection View Go Help	► Invoke 🙊 Ev	ent 🕂	2	
EdgeEC/lightCopyorDataEilter	B	index.py ×					
Eager cynginoer oor oadar mer	L.	14 ci	onnect = sqlite3.connect(db_file_path+db_file_name)				
		15 ci	ursor = connect.cursor()				
		16					
		1/ d	et handler(event, parameters):				
		18	print(" Receive new event: ", event)				
		19	try:				
		20	earce = json.loads(event)				
		21	topic = edict[topic]				
		22	<pre>if thist[] = 'suc'.</pre>				
		2.5	noductive = thist[]				
		25	doviceName = thirt[3]				
		26	tablalame = nodurtKev+' '+devicalame				
		27	print(" Table Name : "+tableName)				
		28	else:				
		29	<pre>print(" [Warn] Event topic should start with /sys/. Ignore!")</pre>				
		30	return 'OK'				
		31					
		32	payload = json.loads(edict['payload'])			(BE	Ε.
		33	for key,value in payload.items():				2
		34	<pre>print(" Get average value of key:"+key+" in 5 minutes:")</pre>				
		35	<pre>curson.execute("SELECT avg(VALUE) FROM {} WHERE KEY = ? AND TIMESTAMP >= datetime('now','-5 minutes','</pre>	<pre>'localtime')".format(tableName), [key])</pre>			
		36	avg_value = cursor.fetchone()[0]				

```
    ⑦ 说明 querySqliteDB样例代码逻辑分为三步。
    i. 从收到的设备上报数据 (event参数) 中解析出productKey和deviceName, 组成数据库的表名。
    edict = json.loads(event)
    topic = edict['topic']
    tlist = topic.split('/')
    if tlist[1] == 'sys':
        productKey = tlist[2]
        deviceName = tlist[3]
        tableName = productKey+'_'+deviceName
        print("-- Table Name : "+tableName)
    i. 获取LightSensor传感器在5分钟内上报的所有数据,并求其平均值。
    cursor.execute("SELECT avg(VALUE) FROM {} WHERE KEY = ? AND TIMESTAMP >= dateti
        me('now', '-5 minutes', 'localtime')".format(tableName), [key])
        avg value = cursor.fetchone()[0]
```

iii. 当Light Sensor传感器上报的数据平均值大于某阈值(此处为240光照度)时,关闭Light设备。

```
if avg_value > 240:
    print("-- Turn off the Light.")
    set_params = {"productKey": "alZJTVs****","deviceName": "LightDev","payload":
    {"LightSwitch":0}}
    res = lesdk.setThingProperties(set params)
```

步骤二:分配函数到边缘实例

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击应用管理。
- 使用本文上方步骤一中已创建的函数,创建函数计算类型的边缘应用。具体操作,请参见函数计算应用。
 应用信息参数说明如下:

参数	描述
应用名称	设置您应用的名称,例如querySqliteDB。
应用类型	选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择EdgeFC服务。
函数	选择querySqliteDB函数。
授权	选择AliyunIOTAccessingFCRole。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。

函数配置区域框中, 启用默认配置选择是, 则其余参数无需设置。

- 4. 在左侧导航栏单击边缘实例。
- 5. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 6. 在实例详情页面的边缘应用页签, 单击分配应用。
- 7. 在**分配应用**面板,找到已创建的过程控制函数querySqliteDB,单击对应操作栏中的**分配**,然后单击**关** 闭。

步骤三:配置消息路由

添加消息的详细步骤及各个参数的解释。详细信息,请参见设置消息路由。

- 1. 在实例详情页面,单击消息路由页签。
- 2. 单击**分配路由**,添加LightSensor设备到函数计算的消息路由。
 按照界面提示,设置如下参数,参数设置完成后,单击确定。

参数	描述
路由名称	设置一个消息路由名称。
消息来源	此处选择设备,选择 光照度传感器 > LightSensor 。
消息主题过滤	此处选择全部。
消息目标	此处选择边缘应用和EdgeFC/querySqliteDB函数。

步骤四: 部署边缘实例

- 1. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。
- 2. 通过SSH登录您的网关,执行 tail -f /linkedge/run/logger/fc-base/querySqliteDB/log.INFO 命

边缘函数计算本地过程控制函数做为一个后台服务存在,可以查看该函数的日志,来观察其实际运行情况。

[root@d0 /]#tail -f /linkedge/run/logger/fc-base/querySqliteDB/log.INF0
(1942, 'MeasuredIlluminance', 3000.0, '2019-04-13 21:56:52')
2019-04-13 21:58:22.293420 [INFO] [querySqliteDB(pid=7972) (read_pipe_msg@fc_main.c:2547)]: Receive new event: {"type":"device","payload":"{\"MeasuredIllumina
nce\":{\"value\":3150,\"time\":1555163902281}}","topic":"/sys/alarJcX0Koy/Sensor_device1/thing/event/property/post","provider":{"groupId":0}}
Table Name : alarJcX0Koy Sensor device1
Get average value of key:MeasuredIlluminance in 5 minutes:
2019-04-13 2Ĭ:58:22.299562 [INFO] [querySqliteDB(pid=7972) (read_pipe_msg@fc_main.c:2547)]: Average value of MeasuredIlluminance=2900.0
Turn off the Light.
2019-04-13 21:58:22.312594 [INFO] [querySqliteDB(pid=7972) (read pipe msg@fc main.c:2547)]: Show table alarJcX0Koy Sensor device1 last 3 entries:
(1945, 'MeasuredIlluminance', 3150.0, '2019-04-13 21:58:22')
(1944, 'MeasuredIlluminance', 3100.0, '2019-04-13 21:57:52')
(1943, 'MeasuredIlluminance', 3050.0, '2019-04-13 21:57:22')
2019-04-13 21:58:52.297132 [INFO] [querySqliteDB(pid=7972) (read_pipe_msg@fc_main.c:2547)]: Receive new event: {"type":"device","payload":"{\"MeasuredIllumina
nce\":{\"value\":3200,\"time\":1555163932286}}","topic":"/sys/alarJcX0Koy/Sensor_device1/thing/event/property/post","provider":{"groupId":0}}
Table Name : alarJcX0Koy_Sensor_device1
Get average value of key:MeasuredIlluminance in 5 minutes:
2019-04-13 21:58:52.302681 [INFO] [querySqliteDB(pid=7972) (read pipe msq@fc main.c:2547)]: Average value of MeasuredIlluminance=2950.0
Turn off the Light.
2019-04-13 21:58:52.311461 [INFO] [querySqliteDB(pid=7972) (read_pipe_msg@fc_main.c:2547)]: Show table alarJcX0Koy_Sensor_device1 last 3 entries:
(1946, 'MeasuredIlluminance', 3200.0, '2019-04-13 21:58:52')
(1945, 'MeasuredIlluminance', 3150.0, '2019-04-13 21:58:22')
(1944, 'MeasuredIlluminance', 3100.0, '2019-04-13 21:57:52')

至此您已经完整地体验了使用边缘函数计算实现本地过程控制功能。

4.4.4. 对接InfluxDB

本文以一个典型的开源时序型数据库(InfluxDB)为例,描述如何使用边缘函数计算应用,实现时序存储设备数据。

前提条件

- 请您确保已完成边缘实例的创建,具体操作,请参见专业版环境搭建。
- 您已创建光照度传感器产品以及该产品下的LightSensor设备,并将设备分配至边缘实例,具体操作,请参见示例驱动。
- 您已创建容器镜像类型的边缘应用,具体操作,请参见容器镜像应用。

其中,部分参数设置按如下要求配置。

应用信息参数说明

参数	说明
应用类型	此处选择容器镜像。
仓库类型	此处选择公共仓库。
镜像地址	设置为 influxdb:latest 。
应用版本	设置为当前使用的数据库版本号,例如1.7.9版本。

容器配置参数说明

参数	描述
是否使用宿主机host模式	此处选择否。
网络端口映射	此处按如下说明,设置两条网络端口映射: • 第一条网络端口映射: = 宿主机端口: 8083 = 容器内端口: 8083 = 类型: TCP • 第二条网络端口映射: = 宿主机端口: 8086 = 容器内端口: 8086 = 类型: TCP
是否启动特权模式	此处选择是。
卷映射	此处按如下说明,设置卷映射: 源路径: /tmp/influxdb 目的路径: /var/lib/influxdb 读写权限:读写

背景信息

在楼宇、工业、农业等物联网场景中,现场的各类传感器负责实时采集数据(如温、湿度、空气质量等数据)。这些数据会随着时间的变化不断更新,因此也称之为时序数据。若需要对此类数据进行大数据分析和 看板展示数据等操作,您首先需要将其存储到时序数据库中。 物联网边缘计算提供的函数计算应用能力,可帮助您快速地将模型化的设备数据存储到时序数据库中。在此 过程中,只需配置该函数计算应用的消息路由并部署应用,无需二次开发。

本文中以Light Sensor传感器为例,将获取到的Light Sensor传感器数据,按照时间顺序存储到InfluxDB中。

步骤一: 创建应用函数

- 1. 下载应用函数代码包fcApp_influxdb_v1.0.zip。
- 2. 登录函数计算控制台。

如尚未开通该服务,请阅读并选中我已阅读并同意,单击立即开通,开通服务。

 (可选)在左侧导航栏单击服务及函数,在服务及函数页面服务列表区域,单击新增服务,创建一个 服务。

其中, 服务名称必须填写, 此处设置为EdgeFC, 其余参数可根据您的需求设置, 也可以不设置。

? 说明

- 若您首次在函数计算中创建服务,请根据配置向导配置参数。
- 若已操作过其他应用场景示例或小程序示例,即已创建EdgeFC服务,则无需重复创建。
- 4. 创建服务成功后,在服务及函数页面的EdgeFC区域,单击新增函数。
- 5. 在新建函数页面, 单击事件函数区域中的配置部署。
- 6. 设置应用函数的基础管理配置参数。

参数	描述
函数类型	保持默认选项。
所在服务	选择已创建的EdgeFC服务。
函数名称	设置为influxdbStore。
运行环境	设置函数的运行环境并选择上传代码的方式。此示例中选择Python 3。 在 上传代码 右侧选择 上传代码包 ,单击 上传代码 ,上传步骤1中下载 的fcApp_influxdb_v1.0.zip代码包。
函数入口	使用默认值index.handler。

其余参数的值请根据您的实际需求设置,也可以不设置。更多信息,请参见函数计算。

确认函数信息后,单击新建完成操作。

7. 创建函数完成后,系统自动跳转到函数详情页面。您可在**代码执行**页签的**代码执行管理**区域下,选 中**在线编辑**单选框,查看源码。

函数计算	開始計算 / 服务 造版 / EdgeFC Y / infludb3tore Y 学品語の 報知	助文档
概范	← influxdbStore BB%版本: LATET V I	ılı
服务-函数		_
自定义域名	能況 代始現行 脱发器 日志告約 始致消化 ARN-acsfc service/EdgefC/functions/influedSto	ie 🚺
资源中心	代码执行管理	
截近访问函数 ^	時行 乾炭開始年 ②	
EdgeFC/influxdbStore	制止年料文 (利止時限) 利止時限) (利止時限) (利止時限) (利止時限) () () () () () () () () ()	R1825
EdgeFC/saveSqliteDB	¢ń File Edit Selection View Go Help ▶ Invoke ∦ Event 🕂 🖌	•
EdgeFC/sendEmail EdgeFC/syncNetworkTime EdgeFC/querySqlteDB EdgeFC/accessAiOSS EdgeFC/lightSensorDataFilter	 ● Indexpy x # TOPIC 为一个字符律典型。首字母必须是?/*. 俗式, # Trigger 为设备事借性情。 "/syx/\${PRODUCT}/\$(DEVICE_NAME}/thing/event/propert/post" # Trigger 为设备事借性情。 "/syx/\${PRODUCT}/\$(DEVICE_NAME)/thing/event/\$(VUIT)/post" # Trigger 为设备事件目標。 "/syx/\${PRODUCT}/\$(DEVICE_NAME)/thing/event/\$(VUIT)/post" # Trigger 为设备事件且指psile(signet, syd音操性且指psile(sage 指定的topic, 临式由用户自定义。 #DAVLOUD 其类型是Geject, syd音操性且指psile(sage 指定的topic, 临式由用户自定义), #UGB事件上指psile(signet, syd音操性上指psile(sade), "MF":("value":(), "time":1524443722000)) #UGB事件上指psile(sadefat, "("pauem":("on", "MF":"2"), "time":1524443722000)) 31 	
	<pre>54 Ger Mandaftevent; context;; 55logger.info(vent) 56 event_json = json.loads(vent) 57 if(vent_json['topic'].tartswith('/sys')); 58 pk, dn, attribute = paraeTopic(vent_json['topic']) 59 psyload_json = json.loads(vent_json['topic']) 60 procPayload(pk, dn, attribute, payload_json) 63 returm 'hello world'</pre>	88

- ⑦ 说明 influxdbStore样例代码逻辑分为三步。
 - i. 从收到的设备上报数据(event参数)中解析出productKey、deviceName和attribute。

```
event_json = json.loads(event)
if(event_json["topic"].startswith('/sys'))
pk, dn, attribute = parseTopic(event_json['topic'])
```

ii. 解析payload(设备的属性或者事件)。

```
payload_json = json.loads(event_json["payload"])
```

iii. 将属性或者事件数据封装成InfluxDB存储的格式,数据格式定义请参见本文下方附录:数据格式定义。此处依赖了Python封装的InfluxDB客户端接口。

步骤二:分配函数到边缘实例

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击应用管理。
- 使用本文上方步骤一中已创建的函数,创建函数计算类型的边缘应用。具体操作,请参见函数计算应用。
 应用信息参数说明如下:

参数	描述
应用名称	设置您应用的名称,例如influxdbStore。
应用类型	选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择EdgeFC服务。
函数	选择influxdbStore函数。
授权	选择AliyunIOTAccessingFCRole。

参数	描述
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。

函数配置说明如下:

参数	描述
启用默认配置	选择否。
运行模式	运行模式有两种。此处选择 持续运行 模式。程序部署后会立即执行。
超时限制(秒)	函数收到事件后的最长处理时间,此处使用默认值5秒。如超过该时间函数仍未返回结果,该函数计算程序将会被强制重启。
定时运行	使用默认配置:关闭。
环境变量	单击 新增环境变量 ,添加如下方"环境变量配置"表格所示的环境变量。

环境变量配置

名称	值
host	InfluxDB的访问地址。
port	InfluxDB的访问端口,默认值为8086。
username	InfluxDB的用户名,默认值为root。
password	InfluxDB的密码,默认值为root。
database	InfluxDB的数据库名称,默认值为example。

其余参数无需配置。

- 4. 在左侧导航栏单击边缘实例。
- 5. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 6. 在**实例详情**页面的边缘应用页签,单击分配应用。
- 7. 在分配应用面板,找到已创建的应用函数influxdbStore,单击对应操作栏中的分配,然后单击关闭。

步骤三: 配置消息路由

添加消息的详细步骤及各个参数的解释。详细信息,请参见设置消息路由。

- 1. 在**实例详情**页面,选择消息路由,单击添加路由,添加Light Sensor设备到函数计算的消息路由。
- 2. 按照界面提示,设置如下参数,参数设置完成后,单击确认。

参数	描述
路由名称	设置一个消息路由名称。

参数	描述
消息来源	此处选择 设备 ,选择 光照度传感器 > LightSensor 。
消息主题过滤	此处选择 全部 。
消息目标	此处选择 边缘应用 和EdgeFC/influxdbStore函数。

步骤四: 部署边缘实例

- 1. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。
- 通过SSH登录您的网关,执行tail-f /linkedge/run/logger/fc-base/influxdbStore/log.INFO命 令,可以查看该函数的日志,来观察其实际运行情况。

2013-10-15	T1:02:00.0T0002	[INFO]	LintluxabStore(pia=944)	U.			erc_main.c:2906J]: <runtime.py:107> DEBUG - result:nello worla</runtime.py:107>
2019-10-12	17:05:06.618898	[INF0]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ws_client.py:84> DEBUG - request:{'tag': '72c656a1-07bc-4371-990d- ', 'opti</ws_client.py:84></pre>
2019-10-12	17:05:06.622979	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ws_client.py:38> DEBUG - receive_msg:{'value': {'statusCode': 200, 'headers': {}, 'bod</ws_client.py:38></pre>
2019-10-12	17:05:06.623026	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ipc_wrapper.py:79> DEBUG - staus:success</ipc_wrapper.py:79></pre>
2019-10-12	17:05:06.623051	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ipc_wrapper.py:109> DEBUG - postTaskResult:{'state': True, 'msg': {'statusCode': 200,</ipc_wrapper.py:109></pre>
2019-10-12	17:05:06.623074	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <runtime.py:112> DEBUG - run_custom_func:succ! result:hello world</runtime.py:112></pre>
2019-10-12	17:05:06.623097	[INFO]	[influxdbStore(pid=944)		1.00		<pre>@fc_main.c:2906)]: <ws_client.py:84> DEBUG - request:{'tag': 'df0c4cef-e73f-4f26-8cf6- ', 'opti</ws_client.py:84></pre>
2019-10-12	17:05:06.629111	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ws_client.py:38> DEBUG - receive_msg:{'value': {'statusCode': 200, 'headers': {'X-Fc-I</ws_client.py:38></pre>
2019-10-12	17:05:06.629167	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ipc_wrapper.py:41> DEBUG - staus:success</ipc_wrapper.py:41></pre>
2019-10-12	17:05:06.629205	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ipc_wrapper.py:69> DEBUG - get param:{'state': True, 'msg': {'statusCode': 200, 'heade</ipc_wrapper.py:69></pre>
2019-10-12	17:05:06.629235	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <index.py:56> INF0 - {"type":"device","payload":"{\"params\": {\"time\": 1570871106602,</index.py:56></pre>
2019-10-12	17:05:06.629455	[INFO]	[influxdbStore(pid=944)	(100	-	<pre>@fc_main.c:2906)]: <runtime.py:107> DEBUG - result:hello world</runtime.py:107></pre>
2019-10-12	17:05:06.629480	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ws_client.py:84> DEBUG - request:{'tag': '14a537ef-1fe2-410b-acce- ', 'opti</ws_client.py:84></pre>
2019-10-12	17:05:06.629503	[INFO]	[influxdbStore(pid=944)		Const.	-	<pre>@fc_main.c:2906)]: <ws_client.py:38> DEBUG - receive_msg:{'value': {'statusCode': 200, 'headers': {}, 'bod</ws_client.py:38></pre>
2019-10-12	17:05:06.629525	[INFO]	[influxdbStore(pid=944)				<pre>@fc_main.c:2906)]: <ipc_wrapper.py:79> DEBUG - staus:success</ipc_wrapper.py:79></pre>
2010 10 12	17.05.00 000540	ETHEOT	Fight Chandle Change (and d. O.(1))				AC and a 200003, day was an 400 DEDUC and Table 1. Flat table 1. Flat table 200

附录:数据查询

当数据存储到InfluxDB后,可通过InfluxDB API查询已存储的数据。部分数据查询示例如下所示,更多详情, 请参见InfluxDB API文档。

• 以产品为维度,查询某个属性所有时间点的值,返回结果会以时间顺序列出所有已存储的属性值。

```
curl -G 'http://localhost:8086/query?db=example&u=root&p=root' --data-urlencode "epoch=ms
" --data-urlencode 'q=SELECT * FROM "属性唯一标识符 (产品下唯一) " WHERE " productKey" = $pk
and " dataType" = $type' --data-urlencode 'params={"pk":"产品ProductKey", "type": "propert
y"}'
//响应结果
{
    "results": [{
       "statement id": 0,
       "series": [{
           "name": "temp",
           "columns": ["time", " dataType", " deviceName", " productKey", "value"],
            "values": [
               [1570782663561, "property", "0987654321", "产品ProductKey", 100.1],
               [1570782688527, "property", "0987654321", "产品ProductKey", 100.2],
                [1570782851708, "property", "0987654321", "产品ProductKey", 104.4]
           1
       }]
   }]
}
```

• 以设备为维度,查询某个属性所有时间点的值,返回结果会以时间顺序列出所有已存储的属性值。

```
curl -G 'http://localhost:8086/query?db=example&u=root&p=root' --data-urlencode "epoch=ms
" --data-urlencode 'q=SELECT * FROM "属性唯一标识符 (产品下唯一) " WHERE "_deviceName" = $dn
and "_dataType" = $type' --data-urlencode 'params={"dn":"设备名称", "type": "property"}'
```

以设备为维度,查询某个属性在一定时间范围内的值,返回结果会以时间顺序列出该时间范围内所有已存储的属性值。

```
curl -G 'http://localhost:8086/query?db=example&u=root&p=root' --data-urlencode "epoch=ms " --data-urlencode 'q=SELECT * FROM "属性唯一标识符(产品下唯一)" WHERE "_productKey" = $pk and "_dataType" = $type and "time" > $timestamp' --data-urlencode 'params={"pk":"产品Produ ctKey", "type": "property", "timestamp": "2019-10-11T09:03:15.611Z"}'
```

附录:数据格式定义

设备的属性或事件类型数据在InfluxDB的存储格式如下所示。

• 单值式属性

```
[
  {
                      // 属性名
   "measurement": "temp",
   "tags": {
                          // 属性的标签
    "_deviceName": "0987654321", // 设备证书信息deviceName
     " dataType" : "property"
                           // 类型: property表示属性, event表示事件
    },
    "time": 1346846400000,
                          // 时间戳,单位: ms
    "fields": {
                          // 属性的值
    "value": 123
   }
  }
  . . .
]
```

• 结构体式属性

```
[
   {
    "measurement": "temp", // 属性名
"tags": { // 属性的
                           // 属性的标签
     "_deviceName": "0987654321", // 设备证书信息deviceName
                             // 类型: property表示属性, event表示事件
      " dataType" : "property"
    },
    "time": 1346846400000,
                             // 时间戳,单位: ms
    "fields": {
                             // 属性的值
     "aa": 12,
                             // 属性的值
     "bb": 12,
                             // 属性的值
     "cc": 12,
      "dd": 12,
                             // 属性的值
    }
   }
   . . .
]
```

事件

Γ { // 事件名 "measurement": "event123", "tags": { // 事件的标签 " deviceName": "0987654321", // 设备证书信息deviceName " dataType" : "event" // 类型: property表示属性, event表示事件 }, "time": 1346846400000, // 时间戳,单位: ms "fields": { "speed" : 20.8, "level" : 4,

"direction" : "East",

4.4.5. 设备数据存入本地MySQL数据库

本文以使用官方示例驱动的Light Sensor设备为例,描述将设备的数据存储到本地MySQL数据库的方法。

前提条件

]

}

- 本示例仅适用于Link IoT Edge专业版,请您确保已完成边缘实例的创建。具体操作,请参见专业版环境搭建。
- 创建光照度传感器产品以及该产品下的LightSensor设备,并将设备分配至边缘实例。具体操作,请参见示例驱动。

背景信息

在生产中常需要结合传感器的历史数据和行业的算法模型,在边缘端进行现场状态的分析和判断。实现此操作,首先需要把设备上报的数据存储到本地数据库。

边缘函数计算提供本地数据库存储助手,根据 ProductKey_DeviceName 格式,将设备上报数据分为不同的 表,并存储到本地MySQL数据库,供函数计算中的其他函数(算法逻辑)查询。同时,因为边缘设备的存储 空间受限,本地数据库还提供数据库单表存储上限配置(默认为1万条记录)和数据写满回滚机制。

步骤一: 创建MySQL容器镜像应用

MySQL容器镜像应用,将作为MySQL数据库服务(server),供函数(client)访问并在其下创建MySQL数据库。

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击应用管理。
- 3. 创建容器镜像类型的边缘应用,具体操作,请参见容器镜像应用。

部分参数说明如下所示,其余参数保持默认值或无需配置。

应用信息参数说明

参数	描述
应用名称	设置您应用的名称,例如publicMySql。
应用类型	选择容器镜像。
仓库类型	选择公共仓库。
镜像地址	设置镜像地址为mysql:latest。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。
环境变量	单击 新增环境变量 ,添加如下方"环境变量配置"表格所示的一条环境变量。

环境变量配置

名称	值
MYSQL_ROOT_PASSWO RD	Mysql数据库的root账户密码,例如设置为123abc。

容器配置参数说明

参数	描述
是否使用宿主机host模 式	选择否。
网络端口映射	设置您应用的网络端口映射。 • 宿主机端口:设置为3306。 • 容器内端口:设置为3306。 • 类型:选择TCP。
是否启动特权模式	选择是。
卷映射	设置您的卷映射。 • 源路径: ■ Linux系统下,设置为/var/mysql/data。 ■ Windows系统下,设置为\ <i>您存放MySQL文件的磁盘:\mysql\data</i> ,例如\D: <i>\mysql\data</i> 。 • 目的路径:设置为/var/lib/mysql。 • 读写权限:选择读写。

4. 单击**确认**,完成MySQL容器镜像应用的创建。

步骤二:分配MySQL容器镜像应用到边缘实例

- 1. 在左侧导航栏单击边缘实例。
- 2. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 3. 在实例详情页面的边缘应用页签,单击分配应用。
- 4. 在**分配应用**面板,找到已创建的MySQL容器镜像应用publicMySql,单击对应操作栏中的**分配**,然后单击关闭。
- 5. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。

步骤三: 创建本地数据库存储函数

- 1. 下载本地数据库存储函数saveMysqlDB-code.zip。
- 2. 登录函数计算控制台。

如尚未开通该服务,请阅读并选中我已阅读并同意,单击立即开通,开通服务。

 (可选)在左侧导航栏单击服务及函数,在服务及函数页面服务列表区域,单击新增服务,创建一个 服务。

其中, 服务名称必须填写, 此处设置为EdgeFC, 其余参数可根据您的需求设置, 也可以不设置。

- ? 说明
 - 若您首次在函数计算中创建服务,请根据配置向导配置参数。
 - 若已操作过其他应用场景示例或小程序示例,即已创建EdgeFC服务,则无需重复创建。
- 4. 创建服务成功后,在服务及函数页面的EdgeFC区域,单击新增函数。
- 5. 在新建函数页面,单击事件函数区域中的配置部署。
- 6. 设置本地数据库存储函数的基础管理配置参数。

参数	描述
所在服务	选择已创建的EdgeFC服务。
函数名称	设置为saveMysqlDB。
运行环境	设置函数的运行环境,此示例中选择Python 3。 在 上传代码 右侧选择 上传代码包 ,单击 上传代码 ,上传步骤1中下载 的saveMysqlDB-code.zip代码包。
函数入口	使用默认值index.handler。

其余参数的值请根据您的实际需求设置,也可以不设置。更多信息,请参见函数计算。

确认函数信息后,单击**新建**完成操作。

7. 创建函数完成后,系统自动跳转到函数详情页面。您可在**代码执行**页签的**代码执行管理**区域下,选 中**在线编辑**单选框,查看源码。

← sa	veMysq	IDB						服务版本	✓ LATES	T V	日志	匝 监控	۵	ARN
概览	代码执行	触发器	日志查询	函数指标	链路追踪	异步配置								
代码。	收行管理													
执行	触发事件	?												
 在线 	編 OSS上作	(代码包	上传 🔾 文件	夹上传									使用旧	版编辑器
\$	File Edit Se	election Vie	w Go Help								► Invoke	∰ Event	÷	$e^{\theta_{i+1}}$
D	EXPLORER EXPLORER Magnetic provides the provided statement of the pr	₫ …	<pre>index.py 1 2 3 4 5 6 7 8 9 10 11 12</pre>	<pre>x # -*- codi import pym import time # User-def; db_name = if db_name db_name mysql_ip = if mysql_i mysql_i</pre>	ng: utf-8 -* ysql e,os,json ined paramet os.getenv("C is None: e = "mysql_c os.getenv(" p is None: ip = ":	ers B_NAME") b" MYSQL_IP") # Use	e docker0 IP as default							Marina Marina Marina Marina Marina Marina Marina Marina Marina Marina

8. (可选)配置数据库参数。

saveMysqlDB是一个设备数据存储到本地MySQL数据库的参考示例,您可以按需调整和更改示例。目前 预留的可调整参数有:

- table_max_entries:数据库单表最大记录条数,默认为1万条。
- table_clean_count:数据库记录到达上限,设置从时间最久远的数据开始删除的条数,默认为4000 条。

步骤四:分配本地数据库存储函数到边缘实例

1. 使用已创建的saveMysqlDB函数,创建函数计算类型的边缘应用。具体操作,请参见函数计算应用。 应用信息参数说明如下:

参数	描述	
应用名称	设置您应用的名称,例如appsaveMysqlDB。	
应用类型	选择函数计算。	
地域	选择您创建的服务所在的地域。	
服务	选择EdgeFC服务。	
函数	选择saveMysqlDB函数。	
授权	选择AliyunIOTAccessingFCRole。	
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。	

函数配置说明如下:

参数	描述
启用默认配置	选择否。
运行模式	运行模式有两种。此处选择 持续运行 模式。程序部署后会立即执行。

参数	描述
超时限制 (秒)	函数收到事件后的最长处理时间,此处使用默认值5秒。如超过该时间函数仍未返回结果,该函数计算程序将会被强制重启。
定时运行	使用默认配置:关闭。
环境变量	单击 新增环境变量 ,添加如下方"环境变量配置"表格所示的环境变量。

环境变量配置

名称	值
DB_NAME	MySQL的数据库名称,请根据MySQL数据库命名规则,自定义设置,例如设置 为mysql_db。
MYSQL_IP	MySQL服务地址,需通过MySQL容器镜像应用名称来访问。本示例中MySQL容器镜 像应用名称为publicMySql。

其余参数无需配置。

- 2. 在左侧导航栏单击边缘实例。
- 3. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 4. 在实例详情页面的边缘应用页签,单击分配应用。
- 5. 在**分配应用**面板,找到已创建的本地数据库存储函数appsaveMysqlDB,单击对应操作栏中的**分配**,然 后单击**关闭**。

步骤五: 配置消息路由

添加消息的详细步骤及各个参数的解释。详细信息,请参见设置消息路由。

- 1. 在实例详情页面,单击消息路由页签。
- 2. 单击**分配路由**,添加LightSensor设备到函数计算的消息路由。

按照界面提示,设置如下参数,参数设置完成后单击**确定**。

参数	描述
路由名称	设置一个消息路由名称。
消息来源	此处选择 设备 ,选择 光照度传感器 > LightSensor 。
消息主题过滤	此处选择 全部 。
消息目标	此处选择 边缘应用 和appsaveMysqlDB函数。

步骤六:重新部署边缘实例

- 1. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。
- 2. 登录您的网关,执行 tail -f /linkedge/run/logger/fc-base/appsaveMysqlDB/log.INFO 命令,可 以查看该应用的日志,来观察其实际运行情况。

在本地MySQL数据库(本示例中名为mysql_db的数据库)中执行 select * from YourProductKey_You rDeviceName; 命令,可以查询到通过appsaveMysqlDB应用存储到MySQL数据库的LightSensor设备数据。

⑦ 说明 将命令中的YourProductKey_YourDeviceName替换为您实际设备的设备证书信息,例如本示例中LightSensor设备的ProductKey为a1****xPAf, DeviceName为LightSensor,那么实际执行的命令为 select * from a1*****xPAf_LightSensor 。

	MeasuredIlluminance	300	2019-11-22 05:38:11
Ì	MeasuredIlluminance	200	2019-11-22 05:38:13
Ĩ	MeasuredIlluminance	100	2019-11-22 05:38:15
ĺ	MeasuredIlluminance	200	2019-11-22 05:38:17
Î	MeasuredIlluminance	300	2019-11-22 05:38:19
ĺ	MeasuredIlluminance	400	2019-11-22 05:38:21
Î	MeasuredIlluminance	500	2019-11-22 05:38:23
ĺ	MeasuredIlluminance	600	2019-11-22 05:38:25
Ĩ	MeasuredIlluminance	500	2019-11-22 05:38:27
Ì	MeasuredIlluminance	400	2019-11-22 05:38:29
1	MeasuredIlluminance	300	2019-11-22 05:38:31
	MeasuredIlluminance	200	2019-11-22 05:38:33
	MeasuredIlluminance	100	2019-11-22 05:38:35
	MeasuredIlluminance	200	2019-11-22 05:38:37
	MeasuredIlluminance	300	2019-11-22 05:38:39
	MeasuredIlluminance	400	2019-11-22 05:38:41
Ì	MeasuredIlluminance	500	2019-11-22 05:38:43
	MeasuredIlluminance	600	2019-11-22 05:38:45
	MeasuredIlluminance	500	2019-11-22 05:38:47
	MeasuredIlluminance	400	2019-11-22 05:38:49
	MeasuredIlluminance	300	2019-11-22 05:38:51
	MeasuredIlluminance	200	2019-11-22 05:38:53
	MeasuredIlluminance	100	2019-11-22 05:38:55
	MeasuredIlluminance	200	2019-11-22 05:38:57
	MeasuredIlluminance	300	2019-11-22 05:38:59
	MeasuredIlluminance	400	2019-11-22 05:39:01
	MeasuredIlluminance	500	2019-11-22 05:39:03
	MeasuredIlluminance	600	2019-11-22 05:39:05
+·		+	-++
6	887 rows in set (0 01	sec)	
0	567 IOWS IN Sec (0.01	3607	
Ш.	ysq1>		

4.4.6. 机器学习边缘推理

Link IoT Edge支持以边缘应用的方式,将云端的机器学习模型部署到边缘端,并在边缘端执行机器学习推理 (ML Inference)。此功能非常适用于在边缘端处理实时性强、数据量大的服务(例如计算机视觉识别)。

前提条件

- 在树莓派4B(Raspberry Pi 4B)或树莓派3B/B+(Raspberry Pi 3B/B+)系统上安装并启动Link IoT Edge软件包,具体操作,请参见基于树莓派搭建环境。
- 在您的树莓派上连接好摄像头,具体操作,请参见树莓派摄像头安装指南。

背景信息

您可以在阿里云机器学习平台PAI或者任何其它地方训练您的推理模型,把训练好的模型和相关代码托管在 阿里云的函数计算、对象存储OSS或者容器镜像等服务中,然后在Link IoT Edge边缘实例中以边缘应用的方 式,将模型部署到网关。在网关上使用本地模型执行推理后把结果上传到阿里云物联网平台。



本文通过在树莓派4B上部署基于Tensorflow Lite的深度学习Object Detection(目标检测)模型,讲述如何在Link IoT Edge上使用机器学习推理。

步骤一: 配置树莓派并安装边缘推理运行时

使用SSH工具连接到树莓派终端,执行如下步骤中的命令。

- 1. 打开树莓派配置工具,并使能摄像头。
 - i. 执行如下命令, 打开树莓派配置工具。

sudo raspi-config

ii. 根据界面提示,选择Interfacing Options,单击Select。

	Basherry	y Pi Software Configuration Tool (raspi-config)
	Ruspoerr.	ii boltware configuration foot (faspi config)
	Change User Password	Change password for the current user
	Network Options	Configure network settings
	Boot Options	Configure options for start-up
	Localisation Options	Set up language and regional settings to match your location
5	Interfacing Options	Configure connections to peripherals
	Overclock	Configure overclocking for your Pi
	Advanced Options	Configure advanced settings
	Update	Update this tool to the latest version
	About raspi-config	Information about this configuration tool
	<select:< th=""><th>> <finish></finish></th></select:<>	> <finish></finish>

iii. 选择Camera, 单击Select, 使能摄像头。

Raspherry Pi Soft	ware Configuration Tool (raspi-config)
	are configuration foor (fulpi config)
P1 Camera Enable/Disable	connection to the Raspberry Pi Camera
P2 SSH Enable/Disable	remote command line access to your Pi using SSH
P3 VNC Enable/Disable	graphical remote access to your Pi using RealVNC
P4 SPI Enable/Disable	automatic loading of SPI kernel module
P5 I2C Enable/Disable	automatic loading of I2C kernel module
P6 Serial Enable/Disable	shell and kernel messages on the serial connection
P7 1-Wire Enable/Disable	one-wire interface
P8 Remote GPIO Enable/Disable	remote access to GPIO pins
<select></select>	<back></back>

- iv. 根据界面提示,单击Finish后,重启树莓派。
- v. 在树莓派Shell终端执行下面的命令,使用树莓派自带raspistill工具测试摄像头是否正常工作。 该命令将在树莓派终端输出摄像头信息,并在当前目录保存名为*cam.jpg*的图片文件。

raspistill -v -o cam.jpg

2. 安装Tensorflow Lite Interpreter。

i. 在树莓派Shell终端执行如下命令,下载Tensorflow Lite Interpreter安装包。

curl -O https://iotedge-web.oss-cn-shanghai.aliyuncs.com/public/LeMLInterpreter/ARM
v7hf/linkedge ml tflite raspi4 cp3x armv7hf installer.tar.gz

ii. 解压并安装软件包。

```
tar xzvf linkedge_ml_tflite_raspi4_cp37_armv7hf_installer.tar.gz
cd linkedge_ml_tflite_raspi4_cp37_armv7hf_installer/
./le_ml_installer.sh
```

系统显示如下图内容,表示安装Tensorflow Lite Interpreter成功。

Looking in indexes:	https://pypi.org/simple, https://www.piwheels.org/simple
	satisfied: pip in /usr/local/lib/python3.7/dist-packages (20.0.2)
Looking in indexes:	https://pypi.org/simple, https://www.piwheels.org/simple
Requirement already	satisfied: picamera in /usr/lib/python3/dist-packages (1.13)
	https://pypi.org/simple, https://www.piwheels.org/simple
	satisfied: numpy in /usr/local/lib/python3.7/dist-packages (1.18.1)
Looking in indexes:	https://pypi.org/simple, https://www.piwheels.org/simple
Requirement already	satisfied: Pillow in /usr/lib/python3/dist-packages (5.4.1)
	https://pypi.org/simple, https://www.piwheels.org/simple
Looking in indexes:	https://pypi.org/simple, https://www.piwheels.org/simple
Looking in indexes:	
Looking in indexes:	https://pypi.org/simple, https://www.piwheels.org/simple
tflite_runtime-2.1.	0.post1-cp37-cp37m-linux_armv71.whl in /usr/local/lib/python3.7/dist-packages (2.1.0.post1)
Requirement already	
)	
Link Tom Edge MI In:	tempeter Fusi rement (notal) Successfully
Edge ML III	corpecter invitoiment inclair successfully.

步骤二:云端发布检测器设备驱动

- 1. 下载检测器设备驱动代码包object_detector_driver.zip。
- 2. 登录边缘计算控制台。
- 3. 在左侧导航栏,单击驱动管理。
- 4. 添加自研驱动,具体操作,请参见云端发布。

其中,部分参数设置如下所示。驱动信息参数说明

参数	描述
驱动名称	自定义驱动的名称,例如obj_detector_driver。
通信协议类型	选择自定义。
语言类型	选择Python 3.5。
驱动是否内置	选择否。
驱动文件	单击 上传文件 ,上传已下载到本地的object_detector_driver.zip驱动文件。
驱动版本	设置为v1.0.0。
驱动适配的边缘版本	选择2.4.0版本及以上。
驱动版本描述	描述您创建的驱动和版本信息,可以为空。

其余参数无需配置。

步骤三:分配检测器设备驱动到边缘实例

1. 左侧导航栏单击边缘实例,在本文"前提条件"中已创建的边缘实例右侧,单击查看。

- 2. 在实例详情页面设备与驱动页签,单击全部驱动右侧的 + 图标。
- 3. 在分配驱动面板中,选择自研驱动,单击obj_detector_driver驱动对应操作栏中的分配,然后单击确

定。 边缘计算 / 边缘实例 / 实例详情 分配驱动 × ← LE_ML_Interpreter 部署成功 ➤ 请输入驱动名称 自研驱动 Q G 产品规格 标准版 CPU 使用率 - 查看 驱动名称 通信协议类型 CPU架构 版本 操作 - 査看 obj_detector_driver 自定义 v1.0.0 🗸 分配 网关 监控信息 设备与驱动 场景联动 边缘应用 消息路 + ✔ 全部驱动 关闭

4. 单击**分配子设备**,在右侧弹出的**分配子设备**面板中,单击**添加子设备**,在obj_detector_driver驱动下,为边缘实例添加子设备。

边缘计算 / 边缘实例 / 实例详情	分配子设备	×
← LE_ML_Interpreter ^{部署成功} 产品原格 标准版		未分配
CPU 使用率 · 查看	内存使用: 设备名称 所属产品 状态 操作	
网关监控信息 设备与驱动 场景联动 边缘应用 消息路由 日志服务	设置	
◆ 全部認助 本 はj_detector_driver 分配子以合 ご協会称 所屋产品 で は は は の の で に し し の の で に し の の で に し の の の の の の の の の の の の の	设备名称 物式配置 均未添加1日 07 Edge 司旗: 分配于设备	
	分配 关闭 有新设备7 灏	加子设备

5. 在添加设备对话框中,单击新建产品,创建检测器产品。

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
×
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置 DeviceName
请输入DeviceName
确定 取消

在创建产品对话框中设置参数后,单击完成。参数说明

参数	描述
产品名称	此处设置为detector。
所属品类	选择品类,为该产品定义 <mark>物模型</mark> 。此处选择 自定义品类 。
接入网关协议	此处选择自定义。

在添加设备对话框,产品自动分配已创建的产品,单击产品下的前往配置,为产品添加自定义功能。具体操作和参数说明,请参见单个添加物模型。

设置如下图两种属性。

○ 物体类别属性

添加自定义功能	×
* 功能类型 ②	
属性 服务 事件	
* 功能名称 💿	
目标类别	
* 标识符 🕐	
object	
* 数据类型	
text (字符串)	~
* 数据长度:	
128	字节
* 读写类型	
描述	
请输入描述	
	0/100
	0,.00
л к===∞ · · · · · · · · · · · · · · · · · ·	
确认	取消

检测得分属性

添加自定义功能	×
* 功能 类型 💿 属性 服务 事件	
* 功能名称 ②	
检测得分	
* 标识符 ⑦ score	
* 数据类型	
float (单精度浮点型)	~
* 取值范围	
0.001	
单位	
请选择单位	~
* 读写类型 读写 页 读写 页 决读 	
描述	
请输入描述	
	0/100
扩展描述 🕐	
+新增扩展描述	
确认	取消

7. 返回边缘计算控制台**实例详情**页面添加设备对话框,为detector产品添加设备。

添加	设备		×
0)特别说明: DeviceName可以为空,当为空时,阿 标识符作为DeviceName。	里云会颁发全局	唯—
* 产品	a.		
det	etector		\sim
产品中 务操作 Device	中可以增加自定义的物模型配置,以便实现业务逻辑 作。如需配置此产品功能请: <mark>前往配置</mark> ceName	钟的属性、事 (牛或者服
tflit	lite_detector		
		确定	取消

8. 将新建的tflite_detector设备分配到边缘实例。

步骤四: 创建边缘推理函数

- 1. 下载边缘推理函数代码包object_detector_app.zip。
- 2. 登录函数计算控制台。

如尚未开通该服务,请阅读并选中我已阅读并同意,单击立即开通,开通服务。

 (可选)在左侧导航栏单击服务及函数,在服务及函数页面服务列表区域,单击新增服务,创建一个 服务。

其中, 服务名称必须填写, 此处设置为EdgeFC, 其余参数可根据您的需求设置, 也可以不设置。

? 说明

- 若您首次在函数计算中创建服务,请根据配置向导配置参数。
- 若已操作过其他应用场景示例或小程序示例,即已创建EdgeFC服务,则无需重复创建。
- 4. 创建服务成功后,在服务及函数页面的EdgeFC区域,单击新增函数。
- 5. 在新建函数页面,单击事件函数区域中的配置部署。
- 6. 设置边缘推理函数的基础管理配置参数。

参数	描述
函数类型	保持默认选项。
所在服务	选择已创建的EdgeFC服务。
函数名称	设置为object_detector_app。

参数	描述
运行环境	设置函数的运行环境并选择上传代码的方式。此示例中选择Python 3。 在 上传代码 右侧选择 上传代码包 ,单击 上传代码 ,上传步骤1中下载 的object_detector_app.zip代码包。
函数入口	使用默认值index.handler。

其余参数的值请根据您的实际需求设置,也可以不设置。更多信息,请参见函数计算。

确认函数信息后,单击**新建**完成操作。

7. 创建函数完成后,系统自动跳转到函数详情页面。您可在**代码执行**页签的**代码执行管理**区域下,选 中**在线编辑**单选框,查看源码。

步骤五:分配函数到边缘实例

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击应用管理。
- 使用本文上方步骤四中已创建的函数,创建函数计算类型的边缘应用。具体操作,请参见函数计算应用。
 应用信息参数说明如下:

参数	描述
应用名称	设置您应用的名称,例如le_object_detector。
应用类型	选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择EdgeFC服务。
函数	选择object_detector_app函数。
授权	选择AliyunIOTAccessingFCRole。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。

函数配置区域框中, 启用默认配置选择是, 则其余参数无需设置。

- 4. 在左侧导航栏单击边缘实例。
- 5. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 6. 在**实例详情**页面的边缘应用页签,单击分配应用。
- 7. 将已创建的边缘推理函数计算应用le_object_detector分配到边缘实例中,单击关闭。

步骤六: 部署边缘实例

- 1. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。
- 2. 部署完成后,在设备与驱动页签,可查到tflite_detector设备状态变为在线。

- 3. 单击tflite_detector设备右侧的查看,系统跳转到设备详情页面。
- 4. 在设备详情页面物模型数据 > 运行状态页签, 查看边缘推理结果。

放置常见的物体到树莓派摄像头前或人站在摄像头前,函数计算应用le_object_detector,会识别物体或人脸并将识别到的结果上报到物联网平台。

物联网平台 / 设备管理 / 设备 / 设备详情								
← tflite	← tflite_detector 在线							
产品	detector 査	看				Dev	viceSecret	
ProductKey	100000	复制						
设备信息	Topic列表	物模型数据	设备影子	文件管理	日志服务	在线调试		
运行状态	事件管理	服务调用						
实时刷新 🔵								
object 直看数据 person 重看数据 2020/05/20 下午3:34:12.212 2020/05/20 下午3:33:56.499								

至此您已经完整地体验了以边缘应用的方式,将机器学习模型部署到边缘端,并在边缘端执行推理的操 作。

4.5. 小程序

4.5.1. 网络时间同步助手

本文介绍如何通过函数计算部署网络时间同步助手,实现设备本地时间与网络时间同步。如果您的设备上已 经安装了NTP服务,有和云端服务器周期性同步系统时间的能力,则无需操作本文内容。

前提条件

本示例仅适用于Link IoT Edge专业版,在开始操作本文内容前,请您确保已根据专业版环境搭建内容完成边缘实例的创建。

背景信息

连接到边缘端的设备,很多Linux系统默认没有安装NTP服务,系统启动后时间常常是从1970年开始计时。 或者一些Linux虚拟机只在系统刚启动时做一次时间同步,后续随着系统休眠等情况,虚拟机计时会暂停。随 着时间的推移,系统时间与网络时间的差距会越来越大。

当设备本地时间和网络时间不同步时,可能会导致访问云服务失败,也可能导致无法在线安装某些软件。正确的时间戳,对于日志分析也能提供更有价值的参考信息。

您可以通过物联网边缘计算提供的边缘函数计算功能,很好的解决设备本地时间与网络时间不同步的问题。

一、创建时间同步函数

- 1. 下载时间同步函数syncNetworkTime-code.zip。该函数会自动和NTP服务器同步网络时间。
- 2. 登录函数计算控制台。

如尚未开通该服务,请阅读并选中我已阅读并同意,单击立即开通,开通服务。

 (可选)在左侧导航栏单击服务及函数,在服务及函数页面服务列表区域,单击新增服务,创建一个 服务。

其中, 服务名称必须填写, 此处设置为EdgeFC, 其余参数可根据您的需求设置, 也可以不设置。

? 说明

- 若您首次在函数计算中创建服务,请根据配置向导配置参数。
- 若已操作过其他应用场景示例或小程序示例,即已创建EdgeFC服务,则无需重复创建。
- 4. 创建服务成功后,在服务及函数页面的EdgeFC区域,单击新增函数。

5. 在新建函数页面, 单击事件函数区域中的配置部署。

6. 设置时间同步函数的基础管理配置参数。

参数	描述
所在服务	选择已创建的EdgeFC服务。
函数名称	设置为syncNetworkTime。
运行环境	设置函数的运行环境,此示例中选择Node.JS 8.x。 在上 传代码 右侧选择 上传代码包 ,单击上 传代码 ,上传步骤1中下载 的syncNetworkTime-code.zip代码包。
函数入口	使用默认值index.handler。

其余参数的值请根据您的实际需求设置,也可以不设置。更多信息,请参见函数计算。

确认函数信息后,单击**新建**完成操作。

7. 创建函数完成后,系统自动跳转到函数详情页面。您可在**代码执行**页签的**代码执行管理**区域下,选 中**在线编辑**单选框,查看源码。

← sy	ncNetwo	orkTim	ne						服务版本	~	LATEST	~	☑ 日志	匝 监控	(ARN
概览	代码执行	触发器	日志查询	函数指标	链路追踪	异步配置										
代码拼	好管理															
执行	触发事件	?														
 在线線 	離 OSS上传	🛛 〇 代码包	上传 🔾 文件	夹上传											使用	日版编辑器
\$	File Edit Se	election Vie	w Go Help)									Invoke	🕸 Event	+	2
B	EXPLORER	đ	undex.js	×												
	node_modu	les	21	'use strict	et :										NAME OF A DESCRIPTION	nen. MMS-
	undex.js		22		,										WEDTERS	
	😑 package-loo	:k.json	23	var exec =	require('c	nild_process	').exec;								TRY SELAR	
			24	var ntpClie	ent = requir	re('./node_r	nodules/ntp-	client');							Mars.	
			25	const ntpServerCn = "cn.pool.ntp.org";					300.00 000							
			26	const ntpSe	erverAllyun	= "ntpl.al:	Lyun.com";									
			27	const htps	suvengrobar	= poor.nct	org;									
			29	function s	/ncNetworkTi	ime(callback	() {									
			30	ntpClien	.getNetwork	Time(ntpSer	verAliyun,	123, function(err, date) {							
			31	if(err) {											
			32	COL	nsole.error((err);										
			33	ret	turn;											
			34	}												
			35	var cu	<pre>^Date = new</pre>	Date(date);	;									
			36	concol	log("NTD	otunn i "	cupData)									
			37	var cui	Month = nar	seInt(curDa	r curuale); ate.getMonth	())+1:								
			50	van cu	nonen – pai	serie(carba	accage chollen	(//'-)								

- 二、分配函数到边缘实例
 - 1. 登录边缘计算控制台。
 - 2. 在左侧导航栏单击应用管理。
 - 3. 使用已创建的syncNetworkTime函数,创建函数计算类型的边缘应用。具体操作,请参见函数计算应用。 应用信息参数说明如下:

参数	描述
应用名称	设置您应用的名称,例如syncNetworkTime。
应用类型	选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择EdgeFC服务。
函数	选择syncNetworkTime函数。
授权	选择AliyunIOTAccessingFCRole。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。

函数配置说明如下:

参数	描述
启用默认配置	选择否。
运行模式	运行模式有两种。此处选择 持续运行 模式。程序部署后会立即执行。
超时限制(秒)	函数收到事件后的最长处理时间,此处使用默认值5秒。如超过该时间函数仍未返回结果,该函数计算程序将会被强制重启。

参数	描述
定时运行	打开定时运行开关,文本框中填入 * * * * * 。表示该函数会被定时触发运行,每分钟执行一次。Cron表达式详细信息请参见CRONT AB 网页内容。

↓ 注意 若您搭建的环境是专业版(LE Pro)环境,则网络时间同步助手需要修改宿主机的系统时间,因此在容器配置中,是否启动特权模式选择是。详细的参数说明请参见"函数计算应用"章节中的容器配置。

其余参数无需配置。

- 4. 在左侧导航栏单击边缘实例。
- 5. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 6. 在实例详情页面的边缘应用页签,单击分配应用。
- 7. 在**分配应用**面板,找到已创建的时间同步函数syncNetworkTime,单击对应操作栏中的**分配**,然后单击关闭。
- 三、部署边缘实例
 - 1. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。
 - 2. 实例部署成功后,在网关上执行 date 命令查看系统的本地时间。此时网关的系统时间应该和网络时间一致。

至此您已经完整地体验了使用边缘函数计算实现网络时间同步的功能。

4.5.2. 本地数据库存储助手

本文以使用官方示例驱动的Light Sensor设备为例,描述将设备的数据存储到本地数据库的方法。

前提条件

- 本示例仅适用于Link IoT Edge专业版,请您确保已根据专业版环境搭建内容完成边缘实例的创建。
- 根据示例驱动内容创建光照度传感器产品以及该产品下的Light Sensor设备,并将设备分配至边缘实例。

背景信息

在生产中常需要结合传感器的历史数据和行业的算法模型,在边缘端进行分析和判断现场状态。实现此操作 首先需要把设备上报的数据存储到本地数据库。

边缘函数计算提供本地数据库存储助手,根据 ProductKey_DeviceName 格式,将设备上报数据分为不同的 表,并存储到本地SQLite数据库,供函数计算中的其他函数(算法逻辑)查询。同时,因为边缘设备的存储 空间受限,本地数据库存储助手还提供数据库单表存储上限配置(默认为1万条记录)和数据写满回滚机 制。

一、创建本地数据库存储函数

- 1. 下载本地数据库存储函数saveSqliteDB-code.zip。
- 2. 登录函数计算控制台。

如尚未开通该服务,请阅读并选中我已阅读并同意,单击立即开通,开通服务。

3. (可选)在左侧导航栏单击服务及函数,在服务及函数页面服务列表区域,单击新增服务,创建一个

服务。

其中, 服务名称必须填写, 此处设置为EdgeFC, 其余参数可根据您的需求设置, 也可以不设置。

? 说明

- 若您首次在函数计算中创建服务,请根据配置向导配置参数。
- 若已操作过其他应用场景示例或小程序示例,即已创建EdgeFC服务,则无需重复创建。

4. 创建服务成功后,在服务及函数页面的EdgeFC区域,单击新增函数。

5. 设置本地数据库存储函数的基础管理配置参数。

参数	描述
所在服务	选择已创建的EdgeFC服务。
函数名称	设置为saveSqliteDB。
运行环境	设置函数的运行环境,此示例中选择python3。
函数入口	使用默认值index.handler。

其余参数的值请根据您的实际需求设置,也可以不设置。更多信息,请参见函数计算。

确认函数信息后,单击完成。

6. 创建函数完成后,系统自动跳转到函数详情页面。请在代码执行页签下选择代码包上传,单击选择文件,上传步骤1中下载的saveSqliteDB-code.zip代码包,然后单击保存。

代码包上传成功后,您可在在线编辑框中查看源码。

函数计算	Ø	酸计算 /	服务·函数 / E	EdgeFC 🏏 🕧	saveSqliteDB 🗡																																																																									产品	动态	4	野助	文相	ă.
概范	•	← sav	eSqlite	eDB																																																																							服	另版	本 : U	TEST	~	5	¥.	h	L
服务-函数		概览	代码执行	触发器	日志查询	函数指标																																																											AF	RN - :	acs:fc							ervice	s/Edg	jeFC,	/functi	ons/s	aveS	qlite	eDB	a	1
自定义城名	e	行动物学会	1944																																																																																
资源中心		CARD AND E		•																																																																															
最近访问函数 へ	4	执行	党党党相利年	9																																																																															
EdgeFC/saveSqliteDB	٢	在线编辑	I 00	OSS上传	○ 代码包上传	〇 文件	件夹上	モ上	夹上	H	E	Ŀ	F	ŀ	1	F	Ŀ	Ŀ	E	f	代	ŧ	ŧ	ŧ	ŧ	ŧ	ŧ	ŧ	ŧ	LIN C	ŧ																																																使用	IBM	Eine	H 23	•
EdgeFC/sendEmail		¢∳ F	ile Edit S	Selection Vi	ew Go Help																																																																					▶ Inv	oke	3	i€ Eve	nt	÷		2		
		n *	index.py ×																																																																																
EdgeFC/syncNetworkTime		_	1 #	-*- coding	g: utf-8 -*-																																																																														
			2 10	mport sqlit	te3																																																																														
Edge+C/querySqliteDB	<		4	mport time,	os,json																																																																														
EdgeEC/accessAliOSS			5 #	User-defin	ned parameter	s																																																																													
coger exocess moss			6 db	b_file_path	/linkedg	e/run/fc-ru	runti	nti	ntir	ti	i	i	i	ti	ti	i	i	i	i	i.m	m	m	ne	ne	ne	n	n	n	ne	ie	1e	2.	-	-	- (d	ata	/"																																													
EdgeFC/lightSensorDataFilter			7 db	b_file_name	e = "device-d	ata-sqlite	e3.db	. db	.db'	зь	Ъ	Ь	łЬ	łĿ	db	łЬ	Ь	Ь	ь'	•"	•	•	•	•	•	•	•	•																																																							
5 . 5			8 ta	able_max_er	ntries = 1000	0 # Maximur	um en	en	ent	ent	ent	m	en	er	er	en	m	n	nt	١t	t	tı	tr	tr	tr	tı	tı	tı	tr	r	r	ni	ŝ	i	i	.e	s p	er	ta	ab]	e																																										
			9 ta	able_retair	n_count = 600	0 # Entries	es re	re	ret	ret	et	e	e	re	re	e	e	e	et	et	t	t	ta	ta	ta	t	ti	t	ta	a	a	a i	ì	i	i	.ne	ed	in	cl	lea	nu	D N	he	en	d	at	а	re	act	hes	t	:he	e 1	tab	ble	e_1	ma	×_	ent	tri	es																						
			10				-		- 1																																																																										
			12 17	os makadir	exists(db_tile_ns	e_patn) == +h)	= Fai	- 81	Falls			-	11				-	1			2	51	50	50	50		1		50	c	e	Î		1	1																																																
			13	02111010021	5(db_111c_pc	cit)																																																																													
			14 cc	onnect = sc	lite3.connec	t(db_file_p	_path	ath	ath-	th-	:h-	h	:h	٤ŀ	t٢	:h	h	h	h-	1+	+	+1	+0	нç	нç	н	+;	н	+d	d	-d	łł	łĿ	ь	Ь.		fil	e_r	nar	ne)																																											
			15 cu	ursor = cor	nnect.cursor()																																																																													
			16																																																																																
			17 de	ef db_init((table_name):																																																																														
			18	# Delete t	table when yo	u want to m	recr	ecn	ecri	- n	n	r	r	ŗ	cr	r	r	r	r i	e	e	e	ea	26	26	2	e	2	ea	9	e9	at	at	t	t	e	it																																														_
			19	#connect.e	execute('drop	table if e	exis	as	×15	151	LS	s	LS	1 5	19	15	s	s	s1	st.	1	τ:	11	5	5				ts 	s	s	5				Ĵ	+ t	ab.	1e	na	me	, 	~-																						 																		
			20	Sq1 = "CRE	CATE TABLE IF	NUL EXISTS	15 -4	*	+1	+1	+1	*	*	1	1	1	*	Ť	*1	rt.	e	τi	τā	C 8	C 8	5	Ci	5	сa	a	a	31	12	D	υ.	τı.	e_n	affe	e+.	0	EY.	16	~ 1		101	C I	nu	111	9.)	VAL	.UE		n.EJ	aL.	no	υC	. n	u1.	τ,	11	mE	514	artir	· 10	NULI	LUE	: - AU	LT.	(dati	etim	e(n	ж',	100	arcim	=))	1)"							-
			22	cu sor.exe	ruce(sq1)																																																																														
			23	# Delete t	trigger when	you want to	to up	up	up	Jpe	ipi	p	q	Jp.	up	ıp	p	p	pe	bd	d	d	da	12	12	4	da	4	da	a	la	a l	nt	t	te	e	it																																														
	_		24	#cursor.ex	kecute('DROP	TRIGGER if	f exi	exi	exi	d	d	i	d	d	xi	d	i	i	i	i s	s	s	st	51	51	51	s	51	st	t	t	ts		s	s		db	cle	ear		+t.	əbl	e_	na	me	e)																																					

7. (可选)配置数据库参数。

saveSqliteDB是一个设备数据存储到本地SQLite数据库的参考示例,您可以按需调整和更改示例。目前预留的可调整参数有:

• db file path: SOLite文件路径, 默认为/linkedge/run/fc-runtime-data/。

○ 注意 若重新部署边缘实例,该目录下的数据库文件*device-data-sqlite3.db*会被删除,您 需要重新配置数据库参数。

- db_file_name: SQLite文件名径,默认为device-data-sqlite3.db。
- table_max_entries:数据库单表最大记录条数,默认为1万条。
- table_retain_count:数据库记录到达上限,清理数据库时需要保留的最近记录数,默认为6000条。

二、分配函数到边缘实例

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏单击应用管理。
- 参考函数计算应用内容,使用本文上方步骤一中已创建的函数,创建函数计算类型的边缘应用。
 应用信息参数说明如下:

参数	描述
应用名称	设置您应用的名称,例如saveSqliteDB。
应用类型	选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择EdgeFC服务。
函数	选择saveSqliteDB函数。
授权	选择AliyunIOTAccessingFCRole。
应用版本	设置应用的版本,必须是该应用唯一的版本号,即一个应用不可以设置两个相同的 版本号。

函数配置说明如下:

容器配置参数说明如下:

参数	描述
卷映射	单击 新增卷映射 ,将文件系统和宿主机的环境是完全隔离的。当一个函数需要访问 宿主机的文件时,需要将文件映射到运行函数的容器内部。最多可添加10个卷映 射。

参数	描述
内存限制	设置容器的内存上限。 • 当容器内存使用超过限制时,容器会被重启。 • 如果内存限制值过小,可能会导致容器内的应用运行失败。 默认内存为1024 MB,请根据容器内应用大小,上调内存限制。

其余参数无需配置。

- 4. 在左侧导航栏单击边缘实例。
- 5. 在本文"前提条件"中创建的边缘实例右侧,单击查看。
- 6. 在**实例详情**页面的边缘应用页签,单击分配应用。
- 7. 将已创建的本地数据库存储函数saveSqliteDB分配到边缘实例中,单击关闭。

三、配置消息路由

将设备上报的数据发送给saveSqliteDB函数处理,通过saveSqliteDB函数将设备数据存储到本地数据库。添加消息的详细步骤及各个参数的解释,请参考<mark>设置消息路由</mark>。

- 1. 在实例详情页面,选择消息路由。
- 2. 单击添加路由,添加LightSensor设备到函数计算的消息路由。

按照界面提示,设置如下参数,参数设置完成后单击**确定**。

参数	描述
路由名称	设置一个消息路由名称。
消息来源	此处选择 设备 ,选择 光照度传感器 > LightSensor 。
消息主题过滤	此处选择 全部 。
消息目标	此处选择函数计算和EdgeFC/saveSqliteDB函数。

四、部署边缘实例

- 1. 在**实例详情**页面,单击右上角**部署**后,在弹出对话框中单击**确定**,将子设备、函数计算等资源下发到 边缘端。
- 2. 实例部署成功后,本地数据库存储函数做为一个后台服务存在,将LightSensor设备上报的数据存储到 SQLite数据库中。

可以在您的网关设备上查看/linkedge/run/fc-runtime-data/device-data-sqlite3.db文件内容。也可以 登录您的网关,执行 tail -f /linkedge/run/logger/fc-base/saveSqliteDB/log.INFO 命令,查看 该函数的日志来观察实际函数运行情况。



5.断网续传

物联网边缘计算提供断网续传服务。当本地断网或数据上传云端后超过15秒未接收到回复,上传云端的数据 将在本地进行持久化。网络恢复后,历史数据将重新上传。使用断网续传功能,需要在物联网平台上配置消 息路由,同时您也可以自定义边缘端持久化设置。

物联网平台配置消息路由

边缘端的消息可以有多个来源,例如设备、函数计算、流数据分析等。物联网边缘计算支持为每个消息来源 配置独立的消息路由。若希望某个来源的消息被本地持久化、支持断网续传,则您需要为该消息配置消息路 由,且必须将**消息目标**设置为*IoT Hub*,**服务级别**设置为*1*(即QoS = 1)。

本文以操作示例驱动文档内容为例,说明如何配置断网续传。

- 1. 操作示例驱动内容。
- 2. 为上一步中的边缘实例分配消息路由,具体操作请参见设置消息路由。

⑦ 说明 设置分配消息路由的参数时,必须将消息目标设置为 IoT Hub,将服务级别设置为 7。

分配消息路由				\times
*路由名称				
断网续传				
* 消息来源				
设备	\sim	全部产品		\sim
* 消息主题过滤				
全部				\sim
* 消息目标				
loT Hub				\sim
* 服务级别 ②				
1				~
			确认	取消

3. 在**实例详情**页面,右上角单击**部署**,重新部署边缘实例。

边缘端持久化配置项

边缘端持久化涉及文件存储目录、节点个数和节点存储空间三个配置项。

⑦ 说明 整个持久化文件最大存储空间 = 持久化节点个数 x 持久化节点大小。
 当整个持久化文件占用存储空间 > 阈值,最久的数据持久化节点将会被删除。

默认配置:

配置项	默认配置
文件存储目录 gw_history_record_storage_path	/linkedge/run/history_record
节点最大个数 gw_history_record_node_num	10个
节点最大存储空间 gw_history_record_node_size	104857600 (100 MB)

自定义配置:

配置项	操作
获取文件存储目录	 命令: /linkedge/gateway/build/bin/lectl config get gw_history_record_storage_path 返回值: Set config success. /linkedge/storage 表示获取配置项成功,存储路径为/linkedge/storage Get config fail, code:100100 表示持久化存储路径未配置,将使用默认路径/linkedge/run/history_record
设置文件存储目录	 命令: /linkedge/gateway/build/bin/lectl config set gw_history_record_storage_path \$存储路径 返回值: Set config success 表示设置成功。 Get config fail, code:100100 表示设置失败,请检查Link loT Edge服务状态。 边缘计算服务未对实际的存储路径进行限制,如果该路径下当前用户无读写权限,会造成 message-router 服务异常。
获取节点个数	 命令: /linkedge/gateway/build/bin/lectl config get gw_history_record_node_num 返回值: Set config success. 20 表示获取配置项成功,节点个数20。 Get config fail, code:100100 表示节点个数未配置,将使用默认值10。

配置项	操作
设置节点个数	 命令: /linkedge/gateway/build/bin/lectl config set gw_history_record_node_num \$节点个数 返回值: Set config success 表示设置成功。 Get config fail, code:100100 表示设置失败,请检查Link loT Edge服务状态。 节点个数可配范围[1-INT_MAX],如果配置越界,日志将会打印 the node num must be between 1 and INT32_MAX 错误。
获取节点最大存储空间	 命令: /linkedge/gateway/build/bin/lectl config get gw_history_record_node_size 返回值: Set config success. 10485760 表示获取配置项成功,节点最大存储空间 104857600 (100 MB)。 Get config fail, code:100100 表示节点最大存储空间未配置,将使用默认值104857600 (100 MB)。
设置节点最大存储空间	 命令: /linkedge/gateway/build/bin/lectl config set gw_history_record_node_size \$节点最大存储空间 返回值: Set config success 表示设置成功。 Get config fail, code:100100 表示设置失败,请检查Link loT Edge服务状态。 存储空间可配范围[1-LONGLONG_MAX],如果配置越界,日志将会打印 the node size must be between 1 and INT64_MAX 错误。

功能验证

以下将使用示例驱动中的光照传感器为例,验证断网续传功能。

该实例的消息路由目标已设置为 IoT Hub, 服务级别已设置为 1; 边缘端持久化配置项使用默认配置。

1. 查看历史数据空间占用。

查看当前持久化文件占用空间,数据结构会占用一部分磁盘空间。

⑦ 说明 持久化文件目录空间 !=0 并不意味着有历史数据存在。

```
#du /linkedge/run/history_record/ -d 0 -h
28.0K /linkedge/run/history_record/
```

2. 仿真网络异常状态。

拔掉网线,间隔几秒时间执行命令,可以看到持久化文件目录空间占用持续递增。
```
#du /linkedge/run/history_record/ -d 0 -h
28.0K /linkedge/run/history_record/
#du /linkedge/run/history_record/ -d 0 -h
56.0K /linkedge/run/history_record/
#du /linkedge/run/history_record/ -d 0 -h
92.0K /linkedge/run/history record/
```

3. 恢复网络。

重新插上网线,网络恢复后,登录控制台查看光照传感器运行状态页面,检查断网期间是否有数据丢 失。

i. 登录边缘计算控制台。

- ii. 在左侧导航栏中,单击边缘实例。
- iii. 在之前创建的实例右侧单击查看。本示例中,查看LinkloTEdge_Node实例。
- iv. 选择**设备与驱动**页面,单击目标驱动,查看该驱动下之前创建的子设备。本示例中,查看 Light Sensor设备。

◇ 全部驱动 +	设备列表 驱动配置 容器配置				
LightSensor	分配子设备 了解设备接入?	全部产品 >	设备名称		Q
	设备名称	所属产品	格式配置	设备状态	操作
	LightSensor	光照度传感器	未配置	● 在线	设备配置 查看 移除

v. 在设备详情页,选择物模型数据 > 运行状态,打开实时刷新,查看设备上报的最新属性数据。

物联网平台 / 设备管理 / 设备 / 设备详情						
← LightSensor						
光照度传感	器 查看					
	复制					
Topic列表	物模型数据	设备影子	文件管理	日志服务	在线调试	
事件管理	服务调用					
直		查看数据				
ĸ						
2020/04/28 下午7:37:24						
	设备管理 / 设	设备管理 / 设备 / 设备详情 正式Sensor 在线 光照度传感器 查看 复制 Topic列表 物模型数据 事件管理 服务调用 直 下午7:37:24	设备管理) 设备 / 设备详情 ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	设备管理 / 设备 / 设备详情 无照度传感器 查看 夏制 下中行で理 正 下午7:37:24 化 化 化 化 化	设备管理 / 设备 / 设备详情 K照度传感器 查看 复制 复制 てpic列表 物模型数据 设备影子 文件管理 服务调用 「 」 「 」 「 」 「 「 」 」 」 」 」	

vi. 单击查看数据,可以看到该属性的历史数据。您可以根据时间戳判断断网期间是否有数据。 本示例中,断网期间的数据,也上传到了云端。



常见问题

• 断网续传会对磁盘IO有影响吗?

答:有影响。断网期间Link loT Edge会对上云数据进行本地持久化操作,对磁盘IO的具体影响,视消息量 大小而定。

- 网络恢复后,在控制台查看发现断网期间有数据未上报?
 答:目前对历史数据上报限流为10条数据/秒。如果边缘端有大量的历史数据等待上报,完成这一动作将 会花费较长的时间。
- 如何自定义持久化文件存储路径及最大存储空间?

答: 可参考本文上方边缘端持久化配置项内容。

- 断网期间,数据持久化文件占用空间达到设定的阈值,新的数据会保存吗?
 - 答:会的。当历史数据持久化文件占用空间达到阈值,最久的历史数据将会被删除。
- 如何查看断网续传模块日志?

答:执行如下命令,查看断网续传模块日志。

- cd /linkedge/run/logger/message-router/
- tail -f log.INFO

6.双机热备

Link IoT Edge提供双机热备方案,支持一个边缘实例关联主备两个网关。

前提条件

请确保已创建边缘实例并上线网关。具体操作,请参见标准版环境搭建。

○ 注意

- 仅v2.4.0及以上版本的标准版Link IoT Edge支持主备网关功能。
- 两个硬件网关上的网卡,需要在同一个网络内。

操作步骤

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏中,单击边缘实例。
- 3. 在"前提条件"中已创建的边缘实例右侧,单击查看。
- 4. 在实例详情页面网关页签下,单击关联网关。

边缘计算 / 边缘实例 / 实例详情		关联网关					×
	82412×13	LinkloTEdge 🗸	设备名称		Q		C
CPU使用率 · 查看	内存使用率	设备名称	产品名称	状态	最后上线时间	操作	
网关 监控信息 设备与驱动 场景联动 边缘应用 满意路由 日志服务 设置		gateway_04	LinkloTEdge_Gate	● 未激活	2020-01-20 17:41:04	关联	
 软件嵌接		gateway_03	LinkloTEdge_Gate	● 未激活	2020-01-05 12:44:55	关联	
网关名称 产品名称 远距	到近问	gateway_02	LinkloTEdge_Gate	● 未激活	2020-01-05 12:44:49	关联	
gateway_01 LinkloTEdge_Gateway							
		关闭				有新网关?	溕加网关

5. 在关联网关页面,选择需要关联到该边缘实例的第二个网关。

注意 待关联的网关必须与已分配到该边缘实例的网关同属一个网关产品下,若无可关联的网关设备,请先为网关产品添加设备。具体操作,请参见单个创建设备。

6. 在弹出的**主备配置**对话框中,根据界面提示设置参数,然后单击确定。

主备配置	×
* 主备策略	
gateway_02 优先	\sim
* 网卡名称(网关: gateway_02)	
请输入网卡名称	
* 网卡名称(网关: gateway_01)	
请输入网卡名称	
浮动虛拟 IP 💿	
请输入浮动虚拟 IP	
确定 耳	以肖

参数说明

参数	描述
主备策略	选择两个网关的优先级。 • 选择某一个网关:表示使用该网关作为主网关。在主网关不可用时,使 用备网关,一旦主网关可用,则切换回主网关。 • 自主协商:始终使用可用的网关作为主网关,若两个网关都为可用状 态,那么由后台确定一个主网关。当主网关不可用时,原来的备网关会 转换为主网关使用。
	两个硬件网关上的网卡名称。
网卡名称	注意 硬件网关上的网卡,需要在同一个网络内。
浮动虚拟IP	网关服务IP地址,会随着主备状态的切换而迁移(始终浮动到主网关 上),确保访问该IP时可以访问到主网关。

7. 上线已关联的第二个网关。

i. 在**网关**页签下,单击软件安装。

ii. 在弹出的对话框中设置软件安装参数, 然后单击**生成安装命令**。



参数说明

参数	描述
边缘网关CPU架构	您的设备系统对应的CPU架构。
产品规格	在创建边缘实例时,已选择实例中使用的Link IoT Edge版本。此处不可操作。
边缘版本	选择Link IoT Edge已发布的版本,详情请参见发布版本。
操作系统	选择您的网关设备对应的操作系统。
选择网关	选择您需要上线的网关。

至此您已完成主备网关的配置。

- (可选)在网关页签下,单击主备配置,可修改主备网关配置。参数说明,请参见本文上方步骤6的参数说明。
- 9. (可选)在网关页签下,单击关联网关,可在弹出的对话框中选择取消关联网关。

7.流数据分析

7.1. 什么是边缘流数据分析

流数据分析是一种使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,继承了物联网平 台的流数据分析能力。

在数据分析控制台创建<mark>流数据分析任务</mark>,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。

⑦ 说明 仅专业版Link IoT Edge支持流数据分析功能。

边缘端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。
- 提供和阿里云流式计算完全相同的SQL语法。您开发一次SQL,既可以在云上执行、也可以在边缘执行。
- 提供了内建的字符串处理和时间、统计等各类计算函数。
- 支持对消息乱序的处理。
- 提供了流数据分析开发的图形化控制台。



7.2. 分配流数据分析到边缘实例

本文介绍创建流数据分析任务、发布任务和分配流数据分析任务到边缘实例的整个步骤。

前提条件

请您确保已创建完成边缘实例,具体操作请参见专业版环境搭建。

操作步骤

- 1. 登录数据分析控制台。
- 2. 创建、设置并发布流数据任务。具体操作,请参见流数据任务开发。
 - 其中,运行环境需选择边缘。

\bigcirc	说明	图片以节点编程任务为例。
------------	----	--------------

新建流数据任务	×
* 流数据任务名称	
test_local_db	
任务描述	
边缘端执行任务	
7/1	00
* 开发类型 🕜	
 ● 节点编程 ○ SQL 	
* 运行环境 🕜	
○ 云端	
确认取	消

- 3. 登录边缘计算控制台。
- 4. 在左侧导航栏单击边缘实例。
 找到要分配流数据分析任务的边缘实例,单击对应操作栏中的查看。
- 5. 在实例详情页面流数据分析页签下,单击分配任务或分配流数据分析任务。
- 在弹出的分配任务页面中,选择要分配给边缘实例的任务,然后单击分配。
 本例中,流数据分析任务名称为test_local_db。
 分配成功后,单击关闭。

分配任务				×
新增任务	请输入任务名称	Q		G
任务名称			操作	
test_local_db)		分配	
关闭				

分配任务完成后,便可在该边缘实例的**实例详情**页面**流数据分析**页签下,查看流数据分析任务信息。 7. 部署边缘实例。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**。

⑦ 说明 当任务在数据分析控制台撤回时,会通知到边缘计算,并在边缘计算中实例详情页 面流数据分析页签下,该任务的状态会显示为已撤回。

- 如果不重新部署边缘实例,将不会影响流数据任务继续处理设备数据。
- 如果重新部署了边缘实例,那么流数据任务将无法处理设备数据,需要重新分配已发布的流数据任务到边缘实例,并部署边缘实例。
- (可选)在**实例详情**页面监控信息页签,选择流数据监控,查看已分配到边缘实例的流数据任务监控 信息。

单击任务名称对应操作栏中的查看,查看详细的监控信息。

边缘计算 / 边缘实例	」 / 实例详情 /	/ 监控详情				
运行中						
流数据监控信息						
最近1小时	~					
触发次数 💿						
			● 触发次数(次)			
10						
6 -						
4 -						
2 -						
16:23	16:33	16:43	16:53	17:03	17:13	

7.3. 边缘端流数据分析SQL

7.3.1. 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,语法说明请参见概述。

在流数据分析中,数据存储用源表、维表、结果表来表示。但由于运行环境不一样,边缘端与云端支持的数 据存储大部分是不相同的。

类型	描述
	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。
源表	边缘端支持的类型如下所示。
	EdgeBus:边缘端消息总线(边缘端跨进程通信机制),EdgeBus的数据作为输入源, 会根据消息路由(管理边缘端消息流向)的配置将数据输入到流作业中,目前支持设备 数据流向流作业。
	······································

类型	描述
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。 边缘端支持的类型如下所示。 • EdgeBusSink:边缘端消息总线,会把流作业的处理结果输出到EdgeBus,可通过消 息路由的配置决定流作业处理结果的具体流向,目前支持流向函数计算、IoTHub。 • RDS:关系型数据库(MySQL),会将流数据分析任务的处理结果输出到阿里云关系 型数据库中。 • 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,主要用于本地调 试。
维表	关系型数据存储,必须显式定义数据变化周期,常用来做数据连接(Join)。 边缘端支持的类型如下所示。 RDS:会把关系型数据库(MySQL)的数据当成维表输入。

7.3.2. 创建源表

流数据分析的源表是指流式数据存储,流式数据存储驱动流数据分析的运行。因此,每个流数据分析任务必须提供至少一个流式数据存储。

创建源表的语法如下所示。

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

边缘端消息总线 (EdgeBus)

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 = 'edgebus',
    jsonParser = 'device_property'
);
```

WITH 参数说明如下:

参数	是否必选	描述
type	是	定义源表类型为edgebus。
jsonParser	否	定义消息解析器,默认为default,取值如下: device_property:属性消息,不符合属性消息格式的消息将会被丢弃 device_event:事件消息,不符合事件消息格式的消息将会被丢弃 default:不使用解析器,直接使用JSON格式字符串,不丢弃任何消息 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 = 'edgebus',
    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。
ts	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 = 'edgebus',
    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。
ts	String	事件产生的时间。
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 = 'edgebus'
);
```

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,详细示例请 参见本文下方JSON格式内容。
jsonType	从消息路由获取到的消息类型。 o device_property o device_event

JSON格式:

```
o jsonType = device_property
```

o jsonType = device_event

```
{
   "value": {
       "BrokenInfo": {
           "Power": "on",
           "structParam": {
               "param1": "abc",
                "param2": 123
           }
       },
       "eventCode2": {
       }
   },
   "time": 1510799670074,
   "productKey": "5RS5XTnNADg",
   "deviceName": "xxxxxxxxxxxxxxx,
   "gmtCreate": 1510799670074
}
```

7.3.3. 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 ]*) ];
```

边缘消息总线 (EdgeBusSink)

示例:

```
create table edgehub_output (
    productKey varchar,
    deviceName varchar,
    temperature int,
    productTime timestamp,
    inputTime timestamp,
    outputTime timestamp,
    delayTime bigint
) with (
    type = 'edgebus_sink',
    topic = '/sys/streamCompute/highTemperature'
);
```

WITH 参数说明如下:

参数	是否必选	描述				
type	是	定义结果表类型为edgebus_sink。				
	是	定义消息主题,必须以斜线"/"开头,支持英文大小写、数字、斜 线"/"、下划线"_"和中划线"-",不超过256字符。				
topic		↓ 注意 若topic需要流转到loTHub(云端),则需要在网 关的产品里定义topic,且该topic必须与SQL中的topic保持一 致。				

关系型数据库 (RDS)

目前仅支持MySQL数据存储类型,数据库中必须要有真实的表存在。 示例: create table db (
productKey varchar,
deviceName varchar,
ts timestamp,
temperature int
) with (
type = 'rds',
url='jdbc:mysql://your_db_host:your_db_port/your_db_name',
tableName='your_table_name',
userName='config://local_db_username', -- local_db_username为在配置中存储的密码的key
);

↓ 注意 在实际使用过程中需要把所有 your_xxxxx参数值, 替换为实际的数据库相关参数值, 并根据如下步骤, 在配置中心存入数据库的用户名密码。

- 1. 在物联网平台控制台左侧导航栏中,单击边缘计算 > 边缘实例。
- 2. 在相应的边缘实例右侧单击查看,进入**实例详情**页面,在网关页签下的网关右侧打开远程访问按钮,登录远程控制台,执行如下命令:

cd /linkedge/gateway/build/bin
./tool_config -s local_db_username root
./tool config -s local db password xxxxxxx

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。
host	MySQL数据库(若为localhost,则为本地数据库)的host地址。
port	MySQL数据库端口号。
dbName	数据库名。
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RDS MySQL版结果表中RDS相关参数说明。

边缘文件系统 (File)

示例:

```
create table print_sink (
    productKey varchar,
    deviceName varchar,
    eventCode varchar,
    ts timestamp,
    temperature int
) with (
    type = 'file'
    tag = 'prop_filter_b',
    filePath = '/linkedge/run/debug/prop_filter_b.txt'
);
```

with 参数说明如下:

参数	是否必选	描述
type	是	定义结果表类型为file。
tag	否	为输出的消息设置标签。 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。 若无需为输出消息设置标签,则删除参数。
filePath	是	输出消息的目的文件路径,不同的流数据分析任务不可共用一个文件 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。

7.3.4. 创建维表

流数据分析没有专门为维表设计的DDL语法,只在标准的CREATE TABLE语法基础上增加一行PERIOD FOR SYSTEM_TIME的声明。该声明定义了维表的变化周期,即表明维表是一张会变化的表。

关系型数据库 (RDS)

目前仅支持MySQL数据存储类型,数据库中必须要有真实的表存在。

示例:

```
CREATE TABLE rds_dim_table(
    id int,
    len int,
    content VARCHAR,
    PRIMARY KEY (id), -- 用作维表时,必须有声明的主键。
    PERIOD FOR SYSTEM_TIME--定义维表的变化周期,表明该表是一张会变化的表。
) with (
    type='rds',
    url='jdbc:mysql://your_db_host:your_db_port/your_db_name',
    tableName='your_table_name',
    userName='config://local_db_username', -- local_db_username为在配置中存储的用户名的key
    password='config://local_db_password' -- local_db_password为在配置中存储的密码的key
);
```

? 说明

- 声明一个维表时,必须指明主键。在进行维表JOIN的时候,JOIN ON的条件必须包含所有主键的等 值条件。RDS的主键可以定义为表的主键或唯一索引列。
- 在实际使用过程中需要把所有 your_xxxxx参数值, 替换为实际的数据库相关参数值, 并根据如下步骤, 在配置中心存入数据库的用户名密码。
 - i. 在物联网平台控制台左侧导航栏中,单击边缘计算 > 边缘实例。
 - ii. 在相应的边缘实例右侧单击查看,进入**实例详情**页面,在网关页签下的网关右侧打开远程 访问按钮,登录远程控制台,执行如下命令:

```
cd /linkedge/gateway/build/bin
./tool_config -s local_db_username root
./tool_config -s local_db_password xxxxxxxx
```

WITH参数

与Flink SQL创建RDS维表的WITH参数一致,详情请参见创建云数据库RDS MySQL版维表。

7.4. 使用示例

7.4.1. 概述

流数据分析提供了连接边缘端消息总线的能力,您可以通过配置消息路由将设备数据传入到流数据分析中,利用流数据分析方便地对设备数据进行实时计算,并将结果输出到RDS、本地文件、函数计算或IoTHub中。

本使用示例将会从多个使用场景来讲述流数据分析的能力及使用方法。阅读本使用示例,需要您了解SQL的 基本知识。下表展示了每个示例的输入、输出,以及使用到的流数据分析算子,您可以根据需求选择性阅 读。

示例名称	输入	流数据分析算子	输出
高温报警	设备属性	过滤	边缘端文件
温度平均值计算	设备属性	过滤、聚合	边缘端文件、MySQL数据 库
连续高温报警	设备属性	复杂事件处理	边缘端文件
订单结算	设备事件、MySQL数据库	聚合、数据连接(Join)	函数计算
人流量展示	设备事件	聚合	函数计算

7.4.2. 高温报警

本文以高温报警(传感器温度>80度时,输出报警信息到本地文件中)的场景为例,详细说明如何使用SQL 进行流数据分析作业的编写以及流数据分析作业如何运作。

流程图如下所示。



前提条件

请您确保已创建边缘实例,具体操作请参见专业版环境搭建。

一、添加设备

参考<mark>设备模拟器</mark>章节中**使用设备模拟器驱动**部分的内容,创建一个使用设备模拟器驱动的设备,并将设备分 配到边缘实例中。

⑦ 说明 将设备分配到边缘实例后,请先不要部署边缘实例,待完成其他操作后统一部署边缘实例。

二、添加流数据分析任务

1. 创建、设置并发布流数据任务,具体操作请参见流数据任务开发。
 其中,开发类型选择SQL,运行环境选择边缘。

新建流数据任务		×
* 流数据任务名称		
高温报警		
任务描述		
当传感器温度大于80°时,输出报警信息到本地文件中		
		25/100
* 开发类型 ?		
○ 节点编程 ● SQL		
* 运行环境 ?		
○ 云端 ● 边缘		
	确认	取消

系统自动进入流数据任务开发SQL工作台。
 复制如下SQL内容到编辑框中。

```
-- 温度>80时,输出报警信息到本地文件中
-- 定义数据源表,源表会驱动流数据分析作业的运行。
-- 每个流数据分析作业必须提供至少一个流式数据存储。
create table property (
   -- 定义源表输入字段
  propertyName varchar, -- 属性名
  propertyValue varchar, -- 属性值
   productKey varchar,
   deviceName varchar,
   ts varchar, -- 数据产生的时间,由设备产生,即设备属性变化发生的时间
   gmtCreate varchar -- 流引擎接收到数据的时间,大于事件时间
) with (
   type = 'edgebus', -- 定义了源表类型, edgebus表示从边缘消息总线获取数据(json格式的字符串)
   jsonParser = 'device property' -- 定义了json字符串解析器, device property表示使用设备属
性json解析器解析,不符合设备属性json格式的数据会被丢弃
);
-- 定义数据结果表,用于存储流数据分析的计算结果。
-- 每个流数据分析作业必须提供至少一个结果表。
create table file (
   -- 定义结果表输出字段
   productKey varchar,
  deviceName varchar,
  temperature double, -- 温度值
  ts timestamp, -- 数据产生的时间
   gmtCreate timestamp -- 流引擎接收到数据的时间
) with (
  type = 'file', -- 定义了结果表类型, edgebus表示将结果输出到文件中
   filePath = '/linkedge/run/debug/case01_high_temperature_alarm.txt' -- 定义了输出的文
件路径
);
-- insert语句定义具体的执行逻辑,源表收到数据后会触发这段逻辑的执行
insert into
  file -- 将结果输出到结果表(文件)
select
  productKey,
  deviceName,
   cast (propertyValue as double), -- propertyValue默认类型是字符,温度值为浮点类型,需要做
数据转化
   to timestamp (cast (ts as bigint)), -- 将数据产生的utc时间转为timestamp
  to timestamp (cast (gmtCreate as bigint)) -- 将流引擎接收到数据的时间转为timestamp
from
  property -- 将源表接收到的设备属性数据拿出来处理
where
   propertyName = 'temperature' -- 通过属性名过滤出包含温度值的数据
   and cast (propertyValue as double) > 80; -- 塞选出温度值>80的数据
```

3. 保存任务并发布。

4. 将该SQL任务分配到边缘实例中,具体操作请参见分配流数据分析到边缘实例。

三、添加消息路由

在边缘实例的**实例详情**页面,选择**消息路由**,将温度传感器的属性变化数据路由到高温报警的流数据分析 作业中,具体操作请参见设置消息路由。

分配消息路由		\times
*路由名称		
高温报警		
* 消息来源		
设备		\sim
* 产品		
温度传感器 X		\sim
设备		
temperatureSensor01 X		\sim
* 消息主题过滤		
属性		\sim
* 消息目标		
流数据分析 🗸 🗸 🗸	高温报警	\sim
	确	认 取消

四、部署边缘实例

- 1. 在边缘实例的实例详情页面,单击部署,将子设备、流数据分析作业及消息路由下发到边缘端。
- 2. 在**实例详情**页面**网关**页签,单击网关名称右侧的远程SSH终端,打开两个远程控制台,例如远程控制 台1和远程控制台2。远程控制台1用于改变温度传感器温度值,远程控制台2用于查看计算结果。

⑦ 说明 需要您先打开网关名称右侧的远程访问按钮,远程SSH终端才可以使用。

网关	监控信息	设备与驱动	场景联动	边缘应用	消息路由	日志服务	设置			
关联网关	软件安装	ŧ								
网关名称		产品名	称			远程访问		状态	最后上线时间	操作
gateway_	_01	Linklo	Edge_Gateway					● 在线	2020-02-19 15:47:04	查看 分享远程SSH终端 远程SSH终端 更多远程服务

3. 在远程控制台1, 进入/linkedge/gateway/build/bin目录, 执行如下命令:

./ds_ctrl property alWuxHr**** temperatureSensor01 '{"temperature":70}'

4. 在远程控制台2, 执行 tail -f /linkedge/run/debug/case01_high_temperature_alarm.txt 命令, 查看输出结果。

此时因为设备温度只达到了70,因此不会输出任何信息。

5. 在远程控制台1, 执行如下命令。

./ds ctrl property alWuxHr**** temperatureSensor01 '{"temperature":90}'

此时温度已经超过了80,可在远程控制台2可看到如下信息。

2019-01-24 18:09:48.490 -> alWuxHr****,temperatureSensor01,90.0,2019-01-24 18:09:48.456 ,2019-01-24 18:09:48.464

? 说明

- 流数据分析作业部署到边缘端后,会先解析 create table 语句,即源表和结果表的定义。在 本例中,源表定义解析完成之后会开始监听边缘消息总线,结果表定义解析后会尝试打开文件。
- 流数据分析作业会一直处于运行的状态,只要接收到边缘消息路由的数据(在本例中为温度传感器的属性变化数据)就会立即执行具体的逻辑,即执行 insert 语句。

7.4.3. 温度平均值计算

本文以计算五秒温度平均值为例,详细说明如何使用流数据分析预置的算子(统计、窗口等)实现五秒温度平均值的计算并存入数据库中。

本文操作中,步骤一至步骤四主要说明如何计算平均温度,步骤五主要说明如何将计算结果保存到数据库中,步骤六主要说明如何根据需求修改分组维度。步骤五至步骤六为可选步骤,可根据自身需求选择操作。



前提条件

请您确保已创建边缘实例,具体操作请参见专业版环境搭建。

背景信息

物联网数据有频率高、数据量大、数据变化小、数据价值较低等特点,将数据直接存储或全部上云的性价比 非常低。为减少数据存储及传输成本,需要先将数据进行过滤、聚合之后再存储或上云。

例如,在工厂中有一台机器上安装了多个温度传感器,用于检测机器各部位的温度变化,现在需要把这些传 感器的状态变化记录下来,以便进行数据分析及问题追溯。由于传感器每次产生的变化都非常小,为节约成 本,需要每五秒(真实场景中可能为一分钟甚至会更长,为了让读者更快看到计算结果,本例采用5秒)求 一次温度的平均值再存储到数据库中。

一、添加设备

参考设备模拟器章节中**使用设备模拟器驱动**部分的内容,创建一个使用设备模拟器驱动的设备,并将设备分 配到边缘实例中。

⑦ 说明

- 设备分配到边缘实例后,请先不要部署边缘实例,待完成其他操作后统一部署。
- 若已操作过高温报警示例内容,并创建了温度传感器产品和设备,则无需重复创建。

二、添加流数据分析任务

1. 参考流数据任务开发, 创建、设置并发布流数据任务。

其中,开发类型选择SQL,运行环境选择边缘。

新建流数据任务		×
* 流数据任务名称		
平均温度		
任务描述		
计算每5秒的温度平均值		
		11/100
* 开发类型 👔		
○ 节点编程 ● SQL		
* 运行环境 👔		
○ 云端 ● 边缘		
	确认	取消

2. 系统自动进入流数据任务开发SQL工作台。

复制如下SQL内容到编辑框中。

计算5秒平均温度,并将结果输出到本地文件中	
定义数据源表	
create table property (
propertyName varchar,	
propertyValue varchar,	
productKey varchar,	
deviceName varchar,	
,一、同类之子。	

```
ts varchar, -- 髙住文化的时间
   tstamp as to timestamp (cast (ts as bigint)), -- ts为long型字符串,单位为毫秒,需要转化
为时间戳格式
   WATERMARK wk FOR tstamp as withOffset (tstamp, 2000) -- 使用时间窗口必须要在源表定义时
声明Watermark。Watermark是插入到数据流中的一种特殊的数据,Watermark上带了一个时间戳,其含义是:
在这个之后不会收到小于或等于该时间戳的数据。假设数据的乱序程度是1分钟,也就是说等1分钟之后,绝大部
分迟到的数据都到了,那么我们就可以定义watermark计算方式为偏移1分钟。2000表示偏移为2秒
) with (
   type = 'edgebus',
   jsonParser = 'device property'
);
-- 定义数据结果表
create table output (
  productKey varchar,
   deviceName varchar,
   avg temperature double,
   t start timestamp,
   t end timestamp
) with (
   type = 'file', -- 定义了结果表类型, file表示将结果输出到文件中
   filePath = '/linkedge/run/debug/case02_avg_temperature.txt' -- 定义了输出的文件路径
);
-- 计算平均温度并写入结果表中
insert into
  output
select
   productKey,
   deviceName.
   avg (temperature) as avg temperature, -- 计算平均温度
   tumble start (tstamp, interval '5' second), -- 时间窗口开始时间(时间窗口长度为5秒)
   tumble end (tstamp, interval '5' second) -- 时间窗口结束时间 (时间窗口长度为5秒)
from (
      select
         productKey,
          deviceName,
          cast (propertyValue as int) as temperature,
          tstamp
      from
          property
      where
         propertyName = 'temperature' -- 筛选出温度属性
   )
where temperature >= 0 and temperature <=100 -- 数据过滤,只计算合法数据
group by
   tumble (tstamp, interval '5' second), -- 按时间窗口维度分组计算(时间窗口长度为5秒)
   productKey, -- 按productKey维度分组计算
   deviceName; -- 按deviceName维度分组计算
```

- 3. 保存任务并发布。
- 4. 将该SQL任务分配到边缘实例中,具体操作请参见分配流数据分析到边缘实例。

三、添加消息路由

在边缘实例的**实例详情**页面,选择**消息路由**,将温度传感器的属性变化数据路由到平均温度的流数据分析 任务中。

分配消息路由			\times
*路由名称			
温度平均值计算			
* 消息来源			
设备			\sim
* 产品			
温度传感器 ×			\sim
设备			
temperatureSensor01 X			\sim
* 消息主题过滤			
属性			\sim
* 消息目标			
流数据分析 く	平均温度		\sim
		确认	取消

↓ 注意 若边缘实例中有关于将温度传感器到IoTHub的消息路由,请将其删除。

四、部署边缘实例

- 1. 在边缘实例的实例详情页面,单击部署,将子设备、流数据分析作业及消息路由下发到边缘端。
- 2. 在**实例详情**页面**网关**页签,单击网关名称右侧的远程SSH终端,打开两个远程控制台,例如远程控制 台1和远程控制台2。远程控制台1用于改变温度传感器温度值,远程控制台2用于查看计算结果。

⑦ 说明 需要您先打开网关名称右侧的远程访问按钮,远程SSH终端才可以使用。

				1				1		
网关	监控信息	设备与驱动	场景联动	边缘应用	消息路由	日志服务	设置			
关联网络	关 软件安装	ŧ								
网关名称	ŝ	产品名	称			远程访问		状态	最后上线时间	操作
gateway	/_01	Linklo	Edge_Gateway					● 在线	2020-02-19 15:47:04	查看 分享远程SSH终端 远程SSH终端 更多远程服务

3. 在远程控制台2, 执行 tail -f /linkedge/run/debug/case01_high_temperature_alarm.txt 命令, 查看输出结果。

在有数据产生的情况下,计算结果每五秒产生一次。

 4. 在远程控制台1,进入/linkedge/gateway/build/bin目录,多次执行./ds_ctrl property a1WuxHr**** temperatureSensor01 '{"temperature":30}'命令,改变温度传感器状态,温度值 可以在30、31两个数值间不断变化,5秒后将会在远程控制台2可看到如下信息。

2019-01-30 15:46:43.045 -> alWuxHr****,temperatureSensor01,30.5,2019-01-30 15:46:35.0,2 019-01-30 15:46:40.0

执行25秒后可看到如下信息。

```
2019-01-30 15:46:43.045 -> alWuxHr****,temperatureSensor01,30.5,2019-01-30 15:46:35.0,2
019-01-30 15:46:40.0
2019-01-30 15:46:49.783 -> alWuxHr****,temperatureSensor01,30.5,2019-01-30 15:46:40.0,2
019-01-30 15:46:53.096 -> alWuxHr****,temperatureSensor01,30.6666666666666666668,2019-01-3
0 15:46:45.0,2019-01-30 15:46:50.0
2019-01-30 15:46:58.119 -> alWuxHr****,temperatureSensor01,30.333333333333332,2019-01-3
0 15:46:50.0,2019-01-30 15:46:55.0
2019-01-30 15:47:02.710 -> alWuxHr****,temperatureSensor01,30.8333333333333332,2019-01-3
0 15:46:55.0,2019-01-30 15:47:00.0
```

五、(可选)保存温度平均值计算结果到数据库

1. 在数据库中创建表。

i. (可选)新建一个RDS实例,或者在搭建边缘环境的机器上安装一个MySQL数据库(也可以跨机器 安装,但必须保证两台机器网络互通)。安装命令如下所示。

⑦ 说明 若已有MySQL数据库(本地数据库或RDS),可直接跳过本步骤。

sudo docker run --name mysql -p your_db_port:3306 -e MYSQL_ROOT_PASSWORD=your_db_pa
ssword -d mysql:5.6.35

其中, *your_db_port*为在宿主机(即启动mysql docker的机器)访问数据库的端口号, *your_db_p assword*为数据库root账户的密码,请根据实际情况更改。例如,端口号为1234,密码为 xxxxxxx,则实例执行的命令为。

```
sudo docker run --name mysql -p 1234:3306 -e MYSQL_ROOT_PASSWORD=xxxxxxxx -d mysql:
5.6.35
```

- ii. 为了保证数据库用户名密码的安全性,您需要把数据库用户名密码保存在配置中心(会加密后再存储)。按如下方式设置。
 - a. 在物联网平台控制台左侧导航栏中,单击边缘计算 > 边缘实例。
 - b. 在相应的边缘实例右侧单击查看,进入**实例详情**页面,在网关页签下单击远程SSH终端,登 录远程控制台,执行如下命令。

```
cd /linkedge/gateway/build/bin
./tool_config -s local_db_username root
./tool_config -s local_db_password xxxxxxx
```

iii. 查看数据库host,即安装MySQL的宿主机IP。在宿主机上执行 ifconfig 命令可看到如下结果。

```
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
  options=1203<RXCSUM, TXCSUM, TXSTATUS, SW TIMESTAMP>
  inet 127.0.0.1 netmask 0xff000000
 inet6 ::1 prefixlen 128
 inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
 nd6 options=201<PERFORMNUD, DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
XHCO: flags=0<> mtu 0
XHC20: flags=0<> mtu 0
en0: flags=8863<UP, BROADCAST, SMART, RUNNING, SIMPLEX, MULTICAST> mtu 1500
  ether f0:18:98:37:81:b4
 inet6 fe80::1c6e:7a87:1c27:724d%en0 prefixlen 64 secured scopeid 0x6
 inet 30.43.83.169 netmask 0xfffff000 broadcast 30.43.95.255
  nd6 options=201<PERFORMNUD, DAD>
 media: autoselect
  status: active
```

其中, *inet*参数后的 30.43.83.169 为宿主机对外IP, 即数据库host (用于流数据分析任务中替 换 your db host)。

iv. 下载mysql-client命令行工具或者mysql-client图形化工具(如Navicat、Sequel Pro),进行 MySQL数据库的访问,并创建 iot_data 数据库,用于在流数据分析任务中替

换 your_db_name 参数。

```
CREATE DATABASE iot_data;
USE iot data;
```

v. 创建名为 table case 02 的表,用于在流数据分析任务会中替换 your table name 参数。

```
CREATE TABLE table_case_02 (

id INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,

productKey varchar(255),

deviceName varchar(255),

avg_temperature double NOT NULL,

t_start timestamp NOT NULL,

t_end timestamp NOT NULL
);
```

2. 更新流数据分析任务,并将结果存入数据库。

在**实例详情**页面**流数据分析**页签,单击流数据任务后的查看,进入**流数据任务开发SQL工作台**页面撤 回任务,并将任务内容改为如下SQL后保存并发布。

```
--- 计算5秒平均温度,并将结果输出到数据库中

-- 定义数据源表

create table property (

propertyName varchar,

propertyValue varchar,

productKey varchar,

deviceName varchar,

ts varchar, -- 属性变化的时间

tstamp as to_timestamp (cast (ts as bigint)), -- ts为long型字符串,单位为毫秒,需要转化

为时间戳格式
```

```
WATERMARK wk FOR tstamp as withOffset (tstamp, 2000) -- 使用时间窗口必须要在源表定义时
声明Watermark。Watermark是插入到数据流中的一种特殊的数据,Watermark上带了一个时间戳,其含义是:
在这个之后不会收到小于或等于该时间戳的数据。假设数据的乱序程度是1分钟,也就是说等1分钟之后,绝大部
分迟到的数据都到了,那么我们就可以定义watermark计算方式为偏移1分钟。2000表示偏移为2秒
) with (
   type = 'edgebus',
   jsonParser = 'device property'
);
-- 定义数据结果表 [此处为变化部分,由之前的文件改成了数据库]
create table output (
  productKey varchar,
   deviceName varchar,
   avg temperature double,
   t start timestamp,
   t end timestamp
) with (
   type = 'rds',
   url='jdbc:mysql://30.43.83.169:1234/iot data', -- 格式: 'jdbc:mysql://your db host:y
our db port/your db name'
   tableName = 'table case 02', -- your table name
   userName = 'config://local db username', -- local db username为在配置中存储的用户名的k
ev
   password = 'config://local db password' -- local db password为在配置中存储的密码的key
);
-- 计算平均温度并写入结果表中
insert into
  output
select
   productKey,
   deviceName,
   avg (temperature) as avg_temperature, -- 计算平均温度
   tumble start (tstamp, interval '5' second), -- 时间窗口开始时间(时间窗口长度为5秒)
   tumble end (tstamp, interval '5' second) -- 时间窗口结束时间 (时间窗口长度为5秒)
from (
      select
         productKey,
          deviceName,
          cast (propertyValue as int) as temperature,
          tstamp
       from
          property
       where
         propertyName = 'temperature' -- 筛选出温度属性
   )
where temperature \geq 0 and temperature \leq 100
group by
   tumble (tstamp, interval '5' second), -- 按时间窗口维度分组计算 (时间窗口长度为5秒)
   productKey, -- 按productKey维度分组计算
   deviceName; -- 按deviceName维度分组计算
```

3. 部署边缘实例并查看设备运行结果。

i. 在边缘实例的**实例详情**页面,单击**部署**,将子设备、流数据分析任务及消息路由下发到边缘端。

ii. 在实例详情页面网关页签下,单击网关右侧的远程SSH终端进入远程控制台。

iii. 在远程控制台中,进入/linkedge/gateway/build/bin目录,多次执行如下命令,每次执行时改变temperature参数后的值。

./ds ctrl property alWuxHr**** temperatureSensor01 '{"temperature":30}'

5秒后可在数据库中看到类似如下数据。

```
mysql> select * from table_case_02;
+----+
 -----+
| id | productKey | deviceName | avg_temperature | t_start
                                            1
t end
          +----+-----+--
  ----+
| 1 | alWuxHr**** | temperatureSensor01 |
                                31 | 2019-01-30 16:05:30 |
2019-01-30 16:05:35
| 2 | alWuxHr**** | temperatureSensor01 |
                                30 | 2019-01-30 16:05:35 |
2019-01-30 16:05:40
| 3 | alWuxHr**** | temperatureSensor01 | 31.6 | 2019-01-30 16:05:40 |
2019-01-30 16:05:45 |
-----+
3 rows in set (0.00 sec)
```

六、(可选)增加数据计算维度

- 1. 添加一个温度传感器。
 - i. 参考一、添加设备内容, 再添加一个温度传感器设备temperatureSensor02。
 - ii. 将温度传感器temperatureSensor02的属性变化数据,路由到平均温度的流数据分析任务中。
 - iii. 参考四、部署边缘实例内容, 部署边缘实例并查看设备运行状态。

需多次改变温度传感器temperatureSensor01和temperatureSensor02状态,每5秒可在数据库中 看到两条结果(每个传感器产生一条结果)。

```
mysql> select * from table case 02 order by t start desc limit 10;
-----+
| id | productKey | deviceName | avg_temperature | t_start
t end
      1
31 | 2019-01-30 16:31:30 |
| 24 | alWuxHr**** | temperatureSensor02 |
2019-01-30 16:31:35 |
| 25 | alWuxHr**** | temperatureSensor01 |
                                        31 | 2019-01-30 16:31:30 |
2019-01-30 16:31:35
| 23 | alWuxHr**** | temperatureSensor01 | 30 | 2019-01-30 16:31:25 |
2019-01-30 16:31:30 |
| 22 | a1WuxHr**** | temperatureSensor02 | 30 | 2019-01-30 16:31:25 |
2019-01-30 16:31:30 |
                                       30.5 | 2019-01-30 16:31:20 |
| 21 | alWuxHr**** | temperatureSensor02 |
2019-01-30 16:31:25
| 20 | alWuxHr**** | temperatureSensor01 |
                                      30.5 | 2019-01-30 16:31:20 |
2019-01-30 16:31:25 |
| 19 | alWuxHr**** | temperatureSensor02 | 31 | 2019-01-30 16:31:15 |
2019-01-30 16:31:20 |
| 18 | alWuxHr**** | temperatureSensor01 | 31 | 2019-01-30 16:31:15 |
2019-01-30 16:31:20 |
| 17 | alWuxHr**** | temperatureSensor01 |
                                        31 | 2019-01-30 16:31:10 |
2019-01-30 16:31:15 |
| 16 | a1WuxHr**** | temperatureSensor02 |
                                        31 | 2019-01-30 16:31:10 |
2019-01-30 16:31:15
_____+
10 rows in set (0.00 sec)
```

2. 求所有传感器的平均温度。

将本文中的SQL按时间窗口、productKey、deviceName三个维度进行分组计算,可以计算出每个温度 传感器5秒的平均温度。若在某些不需要关注每个温度的平均值,只需要知道机器温度的平均值(即所 有温度传感器的平均值)的场景中可以将流数据分析任务的SQL改为如下内容(从分组维度中去掉 productKey和deviceName)。

```
-- 计算5秒平均温度,并将结果输出到数据库中
-- 定义数据源表
create table property (
  propertyName varchar,
  propertyValue varchar,
  productKey varchar,
  deviceName varchar,
  ts varchar, -- 属性变化的时间
   tstamp as to_timestamp (cast (ts as bigint)), -- ts为long型字符串,单位为毫秒,需要转化
为时间戳格式
  WATERMARK wk FOR tstamp as withOffset (tstamp, 2000) -- 使用时间窗口必须要在源表定义时
声明Watermark。Watermark是插入到数据流中的一种特殊的数据,Watermark上带了一个时间戳,其含义是:
在这个之后不会收到小于或等于该时间戳的数据。假设数据的乱序程度是1分钟,也就是说等1分钟之后,绝大部
分迟到的数据都到了,那么我们就可以定义watermark计算方式为偏移1分钟。2000表示偏移为2秒
) with (
   type = 'edgebus',
```

```
Joomearser - device_broberch
);
-- 定义数据结果表 [此处为变化部分,由之前的文件改成了数据库]
create table output (
   productKey varchar,
   deviceName varchar,
   avg temperature double,
   t start timestamp,
   t_end timestamp
) with (
   type = 'rds',
   url='jdbc:mysql://30.43.83.169:1234/iot data', -- 格式: 'jdbc:mysql://your db host:y
our db port/your db name'
   tableName = 'table_case_02', -- your_table_name
   userName = 'config://local db username', -- local db username为在配置中存储的用户名的k
ey
   password = 'config://local db password' -- local db password为在配置中存储的密码的key
);
-- 计算平均温度并写入结果表中
insert into
   output
select
   ·'',
   ۰۰,
   avg (temperature) as avg temperature, -- 计算平均温度
   tumble start (tstamp, interval '5' second), -- 时间窗口开始时间(时间窗口长度为5秒)
   tumble end (tstamp, interval '5' second) -- 时间窗口结束时间(时间窗口长度为5秒)
from (
       select
          cast (propertyValue as int) as temperature,
          tstamp
       from
          property
       where
          propertyName = 'temperature' -- 筛选出温度属性
   )
where temperature \geq 0 and temperature \leq 100
group by
   tumble (tstamp, interval '5' second); -- 按时间窗口维度分组计算(时间窗口长度为5秒)
```

```
运行结果如下所示。
```

<pre>mysql> select * from table_case_02 order by t_start desc limit 5;</pre>									
id	+ productKey	deviceName	avg_temperature	t_start		t_end			
	+			+					
35	I		30.5	2019-01-30	16:40:55	2019-01-30 16:			
41:00	L								
34	I		31	2019-01-30	16:40:50	2019-01-30 16:			
40 : 55									
33	I		31	2019-01-30	16:40:40	2019-01-30 16:			
40 : 45	I								
32	I		30.5	2019-01-30	16:40:35	2019-01-30 16:			
40:40	I								
31	I		30	2019-01-30	16:40:05	2019-01-30 16:			
40:10	I								
++	+	++	+	+		+			
	+								
5 rows	in set (0.00) sec)							

7.4.4. 连续高温报警

现实生活中,温度单次超出阈值并不会对设备造成影响,但连续多次超出阈值可能会引发机器故障等风险。 本文以连续高温报警(连续3次超过80度)为例,详细说明如何使用流数据分析复杂事件处理的功能实现连 续高温事件的判断,并将结果输出到文件中。

示意图如下所示。



前提条件

请您确保已创建边缘实例,具体操作请参见专业版环境搭建。

一、添加设备

参考设备模拟器章节中**使用设备模拟器驱动**部分的内容,创建一个使用设备模拟器驱动的设备,并将设备分 配到边缘实例中。 ? 说明

- 设备分配到边缘实例后,请先不要部署边缘实例,待完成其他操作后统一部署。
- 若已操作过高温报警示例内容,并创建了温度传感器产品和设备,则无需重复创建。

二、添加流数据分析任务

1. 参考流数据任务开发内容, 创建、设置并发布连续高温报警的流数据分析任务。

其中,开发类型选择SQL,运行环境选择边缘。

新建流数据任务	×
* 流数据任务名称	
连续高温报警	
任务描述	
连续高温报警	
	6/100
* 开发类型 👔	
○ 节点编程 ● SQL	
* 运行环境 ?	
○ 云端 ● 边缘	
	确认 取消

2. 系统自动进入流数据任务开发SQL工作台。

复制如下SQL内容到编辑框中。若想了解更多复杂事件处理(CEP)的语法,请参见复杂事件处理(CEP) 语句。

```
-- 连续3次温度>80时,输出报警信息到本地文件中
create table property (
   propertyName varchar,
   propertyValue varchar,
   productKey varchar,
   deviceName varchar,
   ts varchar,
   tstamp as to timestamp (cast (ts as bigint)),
   WATERMARK wk FOR tstamp as withOffset (tstamp, 0)
) with (
   type = 'edgebus',
   jsonParser = 'device property'
);
create table file (
  productKey varchar,
   deviceName varchar,
   temperatureList varchar,
   startTime timestamp,
   endTime timestamp
) with (
   type = 'file',
   filePath = '/linkedge/run/debug/case05_continue_high_temperature_alarm.txt'
);
insert into
file
select
  productKey,
   deviceName,
   temperatureList, -- 每次的温度示数
   startTime, -- 开始时间
   endTime -- 结束时间
from
   property
   MATCH RECOGNIZE (
      PARTITION BY productKey, deviceName
       ORDER BY tstamp -- 在窗口内,对事件时间进行排序
       MEASURES -- 定义如何根据匹配成功的输入事件构造输出事件
           CONCAT(LAST(e.propertyValue, 2), '|', LAST(e.propertyValue, 1), '|', e.prop
ertyValue) as temperatureList,
          LAST(e.tstamp, 2) as startTime,
           e.tstamp as endTime
       ONE ROW PER MATCH -- 匹配成功输出一条
       AFTER MATCH SKIP TO NEXT ROW -- 匹配跳转到下一行后继续处理
       PATTERN (e{3}) -- e连续发生3次则输出一个事件
       DEFINE -- 定义在PATTERN中出现的patternVariable的具体含义
           e as e.propertyName = 'temperature' and cast (e.propertyValue as double) >
80
   );
```

3. 保存任务并发布。

4. 将该SQL任务分配到边缘实例中,具体操作请参见分配流数据分析到边缘实例。

三、添加消息路由

分配消息路由	×
* 路由名称 连续高温报警	
 * 消息来源: 设备 込备 温度传感器 temperatur 	
* 消息主题过滤: 属性 ~	
 * 消息目标: 流数据分析 ∨ 连续高温报警 ∨ 	
	确定取消

在边缘实例的**实例详情**页面,选择**消息路由**,将温度传感器的属性变化数据路由到连续高温报警的流数据 分析作业中。

四、部署边缘实例

- 1. 在边缘实例的实例详情页面,单击部署,将子设备、流数据分析作业及消息路由下发到边缘端。
- 2. 在**实例详情**页面**网关**页签,单击网关名称右侧的远程SSH终端,打开两个远程控制台,例如远程控制 台1和远程控制台2。远程控制台1用于改变温度传感器温度值,远程控制台2用于查看计算结果。

⑦ 说明 需要您先打开网关名称右侧的远程访问按钮,远程SSH终端才可以使用。

网关	监控信息	设备与驱动	场景联动	边缘应用	消息路由	日志服务	设置			
关联网关	€ 软件安装	ŧ								
网关名称	ĸ	产品名	称			远程访问		状态	最后上线时间	操作
gateway	_01	LinkloT	FEdge_Gateway					● 在线	2020-02-19 15:47:04	查看 分享远程SSH终端 远程SSH终端 更多远程服务

3. 在远程控制台1,进入/linkedge/gateway/build/bin目录,执行3次如下命令,每次更改 temperature 参数的值,分别改为80、81、82。

./ds_ctrl property alWuxHr**** temperatureSensor01 '{"temperature":80}'

4. 在远程控制台2,执行 tail -f /linkedge/run/debug/case05_continue_high_temperature_alarm.tx t 命令,查看输出结果。

此时因为设备温度只是连续两次超过80度,因此不会输出任何信息。

5. 在远程控制台1,多次执行如下命令,每次更改 temperature 参数的值,按83~89、78~83,每次增加

一度的规律执行命令。

./ds_ctrl property alWuxHr**** temperatureSensor01 '{"temperature":90}'

命令执行完成后,可在远程控制台2可看到如下信息。

```
2019-02-01 14:51:23.890 -> alWuxHr****,temperatureSensor01,81|82|83,2019-02-01 14:51:20
.1,2019-02-01 14:51:23.723
2019-02-01 14:51:30.472 -> alWuxHr****,temperatureSensor01,82|83|84,2019-02-01 14:51:22
.053,2019-02-01 14:51:30.401
2019-02-01 14:51:32.582 -> alWuxHr****,temperatureSensor01,83|84|85,2019-02-01 14:51:23
.723,2019-02-01 14:51:32.509
2019-02-01 14:51:34.494 -> alWuxHr****,temperatureSensor01,84|85|86,2019-02-01 14:51:30
.401,2019-02-01 14:51:34.398
2019-02-01 14:51:36.602 -> alWuxHr****,temperatureSensor01,85|86|87,2019-02-01 14:51:32
.509,2019-02-01 14:51:36.559
2019-02-01 14:51:38.817 -> alWuxHr****, temperatureSensor01,86|87|88,2019-02-01 14:51:34
.398,2019-02-01 14:51:38.802
2019-02-01 14:51:41.031 -> alWuxHr****,temperatureSensor01,87|88|89,2019-02-01 14:51:36
.559,2019-02-01 14:51:40.967
2019-02-01 14:51:59.790 -> alWuxHr***,temperatureSensor01,81|82|83,2019-02-01 14:51:53
.35,2019-02-01 14:51:59.68
```

7.4.5. 订单结算

本文以订单结算为例,详细说明如何通过流数据分析的维表Join功能,将物联网设备产生的数据(识别器识别的商品及订单编号)和数据库中的数据(商品价格表)进行连接。

示意图如下所示。



前提条件
请您确保已创建边缘实例,具体操作请参见专业版环境搭建。

背景信息

物联网设备的产生的数据大多都是一些动态的属性信息,很多时候我们需要将设备的动态属性和更多维度的 数据(如设备产地、型号,商品的颜色、价格等)进行关联,才能计算出我们想要的结果。

例如,某无人小店有一台商品自助结算机,包含商品识别器及价格显示屏。识别器每识别到一个商品会将商品及订单编号通过事件的方式上报,显示屏实时显示当前累积总价,而当前累积总价需要将识别器上报的数据和数据库里的商品价格表关联后计算得出。

一、添加设备

参考设备模拟器章节中使用设备模拟器驱动部分的内容,添加如下两个设备。

- 添加商品识别器设备:
 - i. 新建商品识别器产品,添加如下表格自定义功能。

参数	描述			
功能类型	选择事件。			
标识符	设置为 recognizedGoods 。			
事件类型	选择信息。			
输出参数	增加如下两个数据类型为 <i>int32(整数型)</i> 的输出参数。 商品编号 goodsId 。 订单编号 orderId 。			

ii. 为商品识别器产品添加设备,并分配进边缘实例中。

iii. 将商品识别器设备的驱动设为设备模拟器驱动。

- 添加显示器设备:
 - i. 新建显示器产品,添加如下表格自定义功能。

参数	描述
功能类型	选择属性。
标识符	设置为 content 。
数据类型	选择text(字符串)。

ii. 为显示器产品添加设备,并分配进边缘实例中。

iii. 将显示器设备的驱动设为设备模拟器驱动。

⑦ 说明 将设备分配到边缘实例后,请先不要部署边缘实例,待完成其他操作后统一部署边缘实例。

二、添加流数据分析任务

1. 创建、设置并发布流数据分析任务,具体操作请参见流数据任务开发。

其中,开发类型选择SQL,运行环境选择边缘。

新建流数据任务		×
* 流数据任务名称		
订单结算		
任务描述		
订单结算任务		
		6/100
* 开发类型 🕐		
○ 节点编程 ● SQL		
* 运行环境 🕐		
○ 云端 ● 边缘		
	确认	取消

2. 系统自动进入流数据任务开发SQL工作台。

复制如下SQL内容到编辑框中。

```
create table event (
  eventCode varchar,
  params varchar,
   ts varchar,
   tstamp as to timestamp (cast (ts as bigint)),
   WATERMARK wk FOR tstamp as withOffset (tstamp, 2000)
) with (
   type = 'edgebus',
   jsonParser = 'device_event'
);
CREATE TABLE goods info(
  id bigint,
   name varchar,
  price double,
   PRIMARY KEY(id),
   PERIOD FOR SYSTEM TIME)
WITH(
   type= 'rds',
```

```
url='jdbc:mysql://your_db_host:your_db_port/your_db_name', -- 请将your_db_host、your
_db_port、your_db_name替换成真实值
   userName='config://local db username', -- 通过./tool config -s local db username xxx
设置
   password='config://local db password', -- 通过./tool config -s local db password xxx
设置
   tableName='table_case_06_goods_info',
   cache='LRU',
   cacheSize='1000',
   cacheTTLMs='600000'
);
CREATE TABLE edgebus sink (
   orderId bigint,
   orderPrice double
) WITH (
   type='edgebus sink',
   topic = '/sys/streamCompute/orderPrice'
);
create view scan goods as
select
   cast (json value(params, '$.goodsId') as bigint) as goodsId,
   cast (json value(params, '$.orderId') as bigint) as orderId
from
   event
where
   eventCode = 'recognizedGoods';
insert into edgebus sink
select
   orderId,
   sum(goods_info.price)
from
   scan_goods
join
   LATERAL goods_info FOR SYSTEM_TIME AS OF PROCTIME()
ON
   goods_info.id = scan_goods.goodsId
group by
   orderId;
```

3. 保存任务并发布。

4. 将该SQL任务分配到边缘实例中,具体操作请参见分配流数据分析到边缘实例。

三、添加函数

- 1. 创建服务,并在该服务下创建名为DisplayPrice的使用Nodejs8运行环境的函数,具体操作请参见使用控制合创建函数。
- 2. 创建函数完成后,在线编辑代码,添加如下内容。

⑦ 说明 请将 your_displaydevice_product Key 以及 your_displaydevice_deviceName 换成实际显示器设备的Product Key和DeviceName。

```
const leSdk = require('linkedge-core-sdk');
const iotData = new leSdk.IoTData();
module.exports.handler = function(event, context, callback) {
   const payload = JSON.parse(JSON.parse(event.toString()).payload);
   const time = payload.time;
   const params = payload.params;
   iotData.setThingProperties({
        productKey: 'your_displaydevice_productKey',
       deviceName: 'your displaydevice deviceName',
       payload: {
           content: `orderId: ${params.orderId}, orderPrice: ${params.orderPrice}, tim
e: ${time}`
       }
    }, (err) => {
       if (err) {
           console.log(`display price err: ${err}`);
        }
   });
};
```

3. 创建边缘应用,具体操作请参见函数计算应用。

应用信息部分参数说明如下:

参数	描述
应用类型	此处选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择DisplayPrice函数所在的服务名称。
函数	选择DisplayPrice函数。
授权	选择AliyunIOTAccessingFCRole。

函数配置部分参数说明如下:

参数	说明
启用默认配置	选择否。
运行模式	选择按需运行。
超时限制 (秒)	设置为5秒。
定时运行	使用默认配置:关闭。

其余参数无需配置。

4. 将应用分配到边缘实例,具体操作请参见分配边缘应用到边缘实例。

四、添加消息路由

1. 在边缘实例的实例详情页面,选择消息路由,将商品识别器的数据路由到订单结算的流数据分析任务

中,具体操作请参见设置消息路由。

 * 路由名称 「订单结算1 * 消息来源 设备 V 商品识别器 V displayDevi V * 消息主题过滤 事件 V * 消息目标 	分配消息路由					×
订单结算1 * 消息来源 设备 商品识别器 displayDevi * 消息主题过滤 事件 * 消息目标	* 路由名称					
 * 消息来源 设备 商品识别器	订单结算1					
设备 商品识别器 displayDevi * 消息主题过滤 事件 ✓ * 消息目标 ✓	* 消息来源					
 * 消息主题过滤 事件 * 消息目标 (近前结管) 	设备	\sim	商品识别智	롬 V	displayDevi	\sim
 事件 ✓ * 消息目标 (近) 注意 (近) 注意 (近) 注意 (近) 注意 (近) 注意 (近) 注意 	* 消息主题过滤					
	事件					\sim
	* 消息目标					
	流数据分析		\sim	订单结算		\sim
					确认	取消

2. 将订单结算的流数据分析任务路由到DisplayPrice函数计算中。消息主题过滤设置为 /sys/streamCompu te/orderPrice 。

分配消息路由			\times
* 路由名称			
订单结算2			
* 消息来源			
流数据分析	\sim	订单结算	\sim
* 消息主题过滤			
/sys/streamCompute/orderPrice	е		
* 消息目标			
边缘应用	\sim	DisplayPrice	\sim
		确认	取消

3. 将显示器的属性变换数据路由到IoTHub中,方便在云端查看显示器设备的状态。

分配消息路由					\times
* 路由名称					
订单结算3					
* 消息来源					
设备	\sim	显示器	\sim	displayDevi	\sim
* 消息主题过滤					
属性					\sim
* 消息目标					
loT Hub					\sim
*服务级别 🕜					
1					\sim
				确认	取消

五、创建商品价格表

若您已经操作过温度平均值计算示例并完成了该章节中五、(可选)保存温度平均值计算结果到数据库这一步内容,则在本文步骤中可以直接跳到子步骤5.创建商品价格表操作。

(可选)新建一个RDS实例,或者在搭建边缘环境的设备上安装一个MySQL数据库(也可以跨设备安装,但必须保证两台设备网络互通)。安装命令如下所示。

⑦ 说明 若已有MySQL数据库(本地数据库或RDS),可直接跳过本步骤。

sudo docker run --name mysql -p your_db_port:3306 -e MYSQL_ROOT_PASSWORD=your_db_passwo
rd -d mysql:5.6.35

其中, your_db_port为在宿主机(即启动mysql docker的机器)访问数据库的端口号, your_db_passw ord为数据库root账户的密码,请根据实际情况更改。例如,端口号为1234,密码为xxxxxxx,则实例执行的命令如下。

sudo docker run --name mysql -p 1234:3306 -e MYSQL_ROOT_PASSWORD=xxxxxxxx -d mysql:5.6. 35

为了保证数据库用户名密码的安全性,您需要把数据库用户名密码保存在配置中心(会加密后再存储)。按如下方式设置。

i. 登录物联网平台控制台, 在左侧导航栏中选择边缘计算 > 边缘实例。

ii. 在相应的边缘实例右侧单击查看,进入实例详情页面,选择网关页签,单击网关右侧的远程SSH 终端,登录远程控制台,执行如下命令。

⑦ 说明 需要您先在网关页签下打开远程访问按钮,远程SSH终端才可以使用。

cd /linkedge/gateway/build/bin ./tool_config -s local_db_username root ./tool config -s local db password xxxxxxx

3. 查看数据库host,即安装MySQL的宿主机IP。在宿主机上执行 ifconfig 命令可看到类似如下结果。

```
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
 options=1203<RXCSUM, TXCSUM, TXSTATUS, SW TIMESTAMP>
 inet 127.0.0.1 netmask 0xff000000
 inet6 ::1 prefixlen 128
 inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
 nd6 options=201<PERFORMNUD, DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
XHCO: flags=0<> mtu 0
XHC20: flags=0<> mtu 0
en0: flags=8863<UP, BROADCAST, SMART, RUNNING, SIMPLEX, MULTICAST> mtu 1500
 ether f0:18:98:37:81:b4
 inet6 fe80::1c6e:7a87:1c27:724d%en0 prefixlen 64 secured scopeid 0x6
 inet 192.168.83.169 netmask 0xfffff000 broadcast 192.168.95.255
 nd6 options=201<PERFORMNUD, DAD>
 media: autoselect
 status: active
```

其中, *inet*参数后的 192.168.83.169 为宿主机对外IP, 即数据库host(用于流数据分析任务中替换 your db host)。

4. 下载mysql-client命令行工具或者mysql-client图形化工具(如Navicat、Sequel Pro),进行MySQL数 据库的访问,并创建 iot_data 数据库,用于在流数据分析任务中替换 your_db_name 参数。

```
CREATE DATABASE iot_data;
USE iot data;
```

5. 创建商品价格表。

```
CREATE TABLE table_case_06_goods_info(
    id INT UNSIGNED NOT NULL PRIMARY KEY,
    name varchar(128),
    price double
);
INSERT INTO table_case_06_goods_info
(id, name, price)
VALUES
(1, 'Coke', 3.00),
(2, 'Sprite', 4.00),
(3, 'Juice', 5.00),
(4, 'Coffee', 6.00),
(5, 'Tea', 7.00);
```

六、部署边缘实例

- 1. 在边缘实例的实例详情页面,单击部署,将子设备、流数据分析作业及消息路由下发到边缘端。
- 2. 在**实例详情**页面,选择网关页签,单击远程SSH终端,登录远程控制台。

⑦ 说明 需要您先打开网关名称右侧的远程访问按钮,远程SSH终端才可以使用。

3. 进入/linkedge/gateway/build/bin目录,执行如下命令上报识别到的商品识别器事件。

```
./ds_ctrl event alivc15**** goodsRecognizer01 recognizedGoods '{"orderId": 12, "goodsId
": 1}'
```

返回物联网平台控制台,左侧导航栏选择设备管理>设备,在相应设备名称右侧单击查看。
 在设备详情页面,选择物模型数据>运行状态,查看显示器设备的运行状态以及数据。

7.4.6. 人流量展示

在物联网场景中,经常需要根据设备的状态分析结果及时作出响应。例如某会场需要实时统计最近十秒人流 量并立即更新显示屏上的数据。

本文以人流量实时显示为例,详细说明如何使用流数据分析进行人流量的统计,并将计算结果输出到函数计 算中,在函数计算中更新显示屏内容。示意图如下所示。



前提条件

请您确保已创建边缘实例,具体操作请参见专业版环境搭建。

一、添加设备

参考设备模拟器章节中使用设备模拟器驱动部分的内容,添加如下两个设备。

- 添加闸口产品和设备:
 - i. 新建闸口产品,添加如下表格自定义功能。

添加自定义功能	\times
* 功能类型 ② 属性 服务 事件	
* 功能名称 💿	
检测到人通过	
* 标识符 🕐	
peoplePassed	
* 事件类型 ② ● 信息 () 告警 () 故障	
输出参数	
+ 参数名称:通过人数 编辑 删除	È
+増加参数	
描述 	
0/1	00
确认取消	É

参数	描述
功能类型	选择事件。
标识符	设置为 peoplePassed 。
事件类型	选择信息。
输出参数	增加一个数据类型为 <i>int 32(整数型)</i> 、标识符为 <i>count</i> 的关于通过人数统计的输 出参数。

ii. 为闸口产品添加gate01设备,并分配到边缘实例中。

iii. 将闸口产品gate01设备的驱动设为设备模拟器驱动。

- 添加显示器设备:
 - i. 新建显示器产品,添加如下表格自定义功能。

参数	描述
功能类型	选择属性。
标识符	设置为 content 。
数据类型	选择text(字符串)。

ii. 为显示器产品添加设备,并分配到边缘实例中。

iii. 将显示器设备的驱动设为设备模拟器驱动。

⑦ 说明 将设备分配到边缘实例后,请先不要部署边缘实例,待完成其他操作后统一部署。

二、添加流数据分析任务

1. 创建、设置并发布流数据分析任务,具体操作请参见流数据任务开发。

其中,开发类型选择SQL,运行环境选择边缘。

新建流数据任务		×
* 流数据任务名称		
计算人流量		
任务描述		
计算人流量		
		5/100
* 开发类型 📀		
○ 节点编程 ● SQL		
* 运行环境 👔		
○ 云端 ● 边缘		
	确认	取消

2. 系统自动进入流数据任务开发SQL工作台。

复制如下SQL内容到编辑框中。

```
-- 计算每个闸口每十秒的人流量并显示到屏幕上
-- 每个闸口检测到人会上报peoplePassed事件,并附带经过的人数count
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 = 'edgebus',
   jsonParser = 'device event'
);
create table edgebus_sink (
  productKey varchar,
   deviceName varchar,
   flowrate int,
   t_start timestamp,
   t end timestamp
) with (
   type = 'edgebus sink',
   topic = '/sys/streamCompute/peopleFlowrate'
);
insert into
   edgebus sink
select
  productKey,
   deviceName,
   sum (cast (json value (params, '$.count') as int)),
   tumble start (tstamp, interval '10' second),
   tumble end (tstamp, interval '10' second)
from
   event
where
   eventCode = 'peoplePassed'
group by
   tumble (tstamp, interval '10' second),
   productKey,
   deviceName;
```

3. 保存任务并发布。

4. 将该SQL任务分配到边缘实例中,具体操作请参见分配流数据分析到边缘实例。

三、添加函数

- 1. 登录阿里云函数计算控制台。
- 2. 创建服务、并在该服务下创建名为**DisplayFlowrate**的使用Nodejs8运行环境的函数,具体操作请参见使用控制台创建函数。
- 3. 创建函数完成后,在线编辑代码,添加如下内容。

⑦ 说明 请将*your_displaydevice_product Key*以及*your_displaydevice_deviceName*换成实际网 关设备的Product Key和DeviceName。

```
const leSdk = require('linkedge-core-sdk');
const iotData = new leSdk.IoTData();
module.exports.handler = function(event, context, callback) {
   const payload = JSON.parse(JSON.parse(event.toString()).payload);
   const params = payload.params;
   iotData.setThingProperties({
       productKey: 'your display device productKey',
       deviceName: 'your display device deviceName',
       payload: {
           content: `flowrate: ${params.flowrate}`
        }
    }, (err) => {
       if (err) {
           console.log(`display flowrate err: ${err}`);
        }
    });
};
```

4. 登录物联网平台控制台,创建边缘应用,具体操作请参见函数计算应用。

应用信息部分参数说明如下:

参数	描述
应用类型	此处选择函数计算。
地域	选择您创建的服务所在的地域。
服务	选择DisplayFlowrate函数所在的服务名称。
函数	选择DisplayFlowrate函数。
授权	选择AliyunIOTAccessingFCRole。

函数配置部分参数说明如下:

参数	说明
启用默认设置	选择否。
运行模式	选择按需运行。
超时限制 (秒)	设置为5秒。
定时运行	选择关闭。

其余参数无需配置。

5. 将应用分配到边缘实例,具体操作请参见分配边缘应用到边缘实例。

四、添加消息路由

在边缘实例的**实例详情**页面,选择**消息路由**,将闸口设备的数据路由到人流量计算的流数据分析任务中。

分配消息路由					\times
* 路由名称					
人流量展示1					
* 消息来源					
设备	\sim	闸口	\sim	displayDevi	\sim
* 消息主题过滤					
事件					\sim
* 消息目标					
流数据分析		\sim	计算人流	₽ ₩	\sim
				确认	取消

2. 将人流量计算的流数据分析任务路由到DisplayFlowrate函数计算中。消息主题过滤设置为 /sys/str eamCompute/peopleFlowrate 。

分配消息路由			×
* 路由名称			
人流量展示2			
* 消息来源			
流数据分析	\sim	计算人流量	\sim
* 消息主题过滤			
/sys/streamCompute/peopleFl	owrate		
* 消息目标			
边缘应用	\sim	DisplayFlowrate	\sim
		确认	取消

3. 将显示器的属性变换数据路由到IoTHub中,方便在云端查看显示器设备的状态。

分配消息路由					\times
* 路由名称					
人流量展示3					
* 消息来源					
设备	\sim	显示器	\sim	displayDevi	\sim
* 消息主题过滤					
属性					\sim
* 消息目标					
loT Hub					\sim
*服务级别 ②					
1					\sim
				确认	取消

五、部署边缘实例

- 1. 在边缘实例的实例详情页面,单击部署,将子设备、流数据分析作业及消息路由下发到边缘端。
- 2. 在**实例详情**页面,选择网关页签,单击远程SSH终端,登录远程控制台。

⑦ 说明 需要您先打开网关名称右侧的远程访问按钮,远程SSH终端才可以使用。

3. 进入/linkedge/gateway/build/bin目录,执行如下命令上报有人员通过的事件。

./ds_ctrl event alWn0t8**** gate01 peoplePassed '{"count": 1}'

返回物联网平台控制台,左侧导航栏选择设备管理>设备,在相应设备名称右侧单击查看。
 在设备详情页面,选择物模型数据>运行状态,查看显示器设备的运行状态以及数据。

8.消息路由 8.1. 什么是消息路由

消息路由即通过路由规则动态规划消息的传输路径,使消息按照过滤条件,从消息源路由到目标节点。通过 消息路由,可实现对数据路由的灵活控制和提高数据安全性。

IoT边缘计算基于消息路由,将数据在设备、函数计算应用、IoT Hub之间进行流转。支持同时存在多条路由路径。

目前支持以下几种消息路由路径:

- 设备至函数计算应用。
- 设备至流数据分析。
- 设备至IoT Hub。
- 函数计算应用至函数计算应用。
- 函数计算应用至 loT Hub。
- 流数据分析至IoT Hub。
- 流数据分析到函数计算应用
- IoT Hub至函数计算应用。

关于消息路由QoS(消息投递质量),目前Link loT Edge八种消息路由路径全部支持QoS=0;目的端为loT Hub的消息路由路径还支持QoS=1。

⑦ 说明 QoS=0代表最多一次, QoS=1代表最少一次, QoS=2代表有且仅有一次。后续所有路由路径 将完全支持其他两种QoS标准。



8.2. 设置消息路由

消息路由即通过路由规则动态规划消息的传输路径,使消息按照过滤条件,从消息源路由到目标节点。

前提条件

消息路由需在边缘实例中设置。您需在设置消息路由之前,先创建边缘实例,并为该实例分配网关、驱动、 设备和边缘应用。

操作步骤

- 1. 登录边缘计算控制台。
- 2. 在左侧导航栏中,单击边缘实例。
- 3. 找到您要设置消息路由的实例,单击操作栏中的查看。
- 4. 在**实例详情**页面,选择消息路由页签,单击分配路由。

分配消息路由	>	<
*路由名称		
请输入路由名称		
* 消息来源		
设备	~	
* 产品		
请选择产品名称	~	
* 消息主题过滤		
全部	~	
* 消息目标		
边缘应用 ソ	请选择应用 ・ ・ 、	
	确认取消	

5. 在弹出的分配消息路由对话框中,设置分配消息路由的配置项。配置完成后,单击确定。

参数名称	参数说明
路由名称	设置一个消息路由名称。长度限制为4~32个字符,汉字和中文符号算2个字符。

参数名称	参数说明
	选择消息的发送者。 • 设备:表示已分配至该实例的设备发送消息。 此时需要配置消息来源的具体产品或设备,请参见本表格中的 产品和设备 参数说明。 • 边缘应用:表示消息由边缘应用发出,需选择已分配给该边缘实例的应用名称。
	⑦ 说明 仅函数计算类型的边缘应用,支持使用消息路由。
消息来源	 IoT Hub:表示消息由云端发出。 此时,依赖网关产品下的如下自定义Topic类,请确认您的网关产品下是否已存在该Topic类。若没有,请添加该Topic类(操作方法请见自定义Topic)。 Topic类: /YourProductKey/\${deviceName}/user/# ,其中YourProductKey; \$/deviceName/通配DeviceName。 设备操作权限:订阅。 流数据分析:表示消息由流数据分析发出,需选择已分配给该实例的流数据分析任务名称。
产品	 当消息来源为设备时,需要选择消息来源的产品。 全部产品:该边缘实例下的全部设备都是消息来源。 选择多个产品:这些产品下已分配到边缘实例的所有设备都是消息来源。 ①说明 一个消息路由,最多可同时选择100个产品。 选择某一个产品:使用该产品下已分配到边缘实例的设备作为消息来源。此时您需要选择具体的设备,请参见本表格中的设备参数说明。
设备	 当消息来源为设备,且产品参数仅选择了一个产品时,需要选择该产品下具体的设备,作为消息来源。 全部设备:该产品下分配至边缘实例的全部设备都是消息来源。 选择多个设备:表示这些设备将作为消息来源。 ①说明 一个消息路由,最多可同时选择100个设备。 选择某一个设备:表示指定该设备作为消息来源。
消息主题过滤	设置消息过滤条件。只有满足此条件的消息才能到达消息目标。设置方法如下: 当消息来源为设备时:可选择为设备的属性、事件或全部。 当消息来源为边缘应用或流数据分析时:需自定义(定义方法请参见什么是Topic)。可使用通配符 + 和 # 。 当消息来源为IoT Hub时:从下拉框中选择需要过滤的Topic类。

参数名称	参数说明
消息目标	 消息的接收者。 IoT Hub:表示消息需要上传到云端。则需要设置服务级别,请参见本表格中的服务级别参数说明。 边缘应用:需要指定已分配到边缘实例的应用名称。 ⑦ 说明 仅函数计算类型的边缘应用,支持使用消息路由。 流数据分析:仅消息来源为设备时可选,需要指定流数据分析的任务名称。
服务级别	 当消息目标为loT Hub时,需要设置服务级别来确定您的设备数据是否支持断网续传。 0:表示需要上传到云端的消息在网络异常状态下会丢失。 1:表示在网络异常时,需要上传到云端的消息将被保存在本地存储区,在网络恢复后Link loT Edge将存储区中的历史数据续传到云端。 当消息路由路径是边缘应用至loT Hub,并选择服务级别为1时,需在网关产品下,自定义如下Topic类(操作方法请见自定义Topic),并重启网关。 Topic类: /YourProductKey/YourDeviceName/user/restore,其中,将YourProductKey和YourDeviceName/user/restore,其中,将YourProductKey和YourDeviceName替换为网关的设备证书信息。 设备操作权限:订阅。 断网续传的历史数据会通过此Topic上传至云端。您可以通过云端规则引擎,分类处理网络恢复后续传的历史数据。历史数据格式如下: ("topic":"yyyyy", "payload": #string型或object型 } 其中 topic 为原上云数据的Topic, payload 为原上云的数据内容。 ② 说明 服务级别选择1时,在网络异常状态下CPU和磁盘lo占用量会增加,当数据量很大时可能会耗尽系统磁盘和CPU资源。不同的系统性能其承受的负载上限有所不同,请您根据自身需求选择是否支持断网续传。

? 说明

- 有相同消息来源和消息目标、不同消息主题过滤的路由,其消息最多转发一次。即路由执行后,如果有一条路由匹配成功,则剩余的路由不再匹配执行。
- 有不同**消息来源**或消息目标的所有路由, 会依次匹配执行。

此时可能会有逻辑上的重复执行转发消息的情况。例如,一条路由**消息来源**是全部设备,**消息目标**是loT Hub,另一条路由**消息来源**是设备A,**消息目标**也是loT Hub;当设备A 发送消息时,上述两条路由都会匹配执行,即设备A的数据向loT Hub发送两次。

8.3. 消息路由设置示例

一个边缘实例可以设置一个或多个消息路由路径。本节将以示例来展示Link IoT Edge支持的八种消息路由路径的设置方法和消息查看方法。

设置消息路由的操作请参见设置消息路由。

消息路由路径:从设备路由到IoT Hub

新建边缘实例后,会自动分配一个从设备路由到loT Hub的默认路由,将设备的全部消息(属性和事件)上 报至云端(loT Hub)。如果边缘实例中的默认路由被移除,或者您需要将单个设备的属性、事件或全部消 息上报到云端,您可以手动添加从设备路由到loT Hub的消息路由。

下图示例表示,Light和Light02设备的属性及事件信息会上报到云端,并且在断网后设备数据无法上报到云端时,Link IoT Edge会将这些数据存储到本地,在恢复网络后把本地数据重新传到云端。

分配消息路由	×
* 路由名称	
设备到IoTHub	
* 消息来源	
设备	\sim
*产品	
客厅灯 X	\sim
设备	
Light02 X Light X	\sim
* 消息主题过滤	
全部	\sim
* 消息目标	
IoT Hub	\sim
* 服务级别 ②	
1	\sim
	确认 取消

查看消息:通过消息主题过滤后的设备数据发送到云端后,您便可以在物联网平台控制台,该子设备所属产品的**日志服务**页面查看设备上报的数据。

消息路由路径:从设备路由到边缘应用

将设备的属性或全部消息(属性和事件)发送至边缘应用中。

↓ 注意 仅函数计算类型的边缘应用,支持使用消息路由。

下图中的示例表示,将Light设备的属性信息发送到指定的lightSensorDataFilter边缘应用中。

分配消息路由		×
* 路由名称		
设备到边缘应用		
* 消息来源		
设备		\sim
* 产品		
客厅灯 🗙		\sim
设备		
Light X		\sim
* 消息主题过滤		
全部		\sim
* 消息目标		
边缘应用 🗸 🗸	lightSensorDataFilter	\sim
	确认	取消

查看消息:目前,由于路由到边缘应用的消息未上报云端,所以只能在网关本地日志中查看。用tail命令查 看网关本地日志。命令格式: tail -f /linkedge/run/logger/fc-base/{module name}/log.INFO。其 中,变量*{module name}*需填写您实际的模块名称。示例: tail -f /linkedge/run/logger/fcbase/function-compute/log.INFO, function-compute 为边缘应用的模块名。

消息路由路径:从设备路由到流数据分析

将设备的属性或全部消息(属性和事件)上报至流数据分析。

下图示例表示,将全部设备的属性及事件信息发送到名为高温报警的流数据分析任务中。

分配消息路由		×
* 路由名称		
设备到流数据分析		
* 消息来源		
设备		\sim
* 产品		
温度传感器 X		\sim
设备		
temperatureSensor01 X		\sim
* 消息主题过滤		
全部		\sim
* 消息目标		
流数据分析 く	高温报警	\sim
	确	以 取消

查看消息:路由到流数据分析的数据只能在本地日志中查看。用tail命令查看本地日志。命令格式: tail -f /linkdge/run/log_slf4j/blink/flink--localsession-*{max_index}-{container_id}*.log 。其中,变 量*{max_index}*为当前最大日志序列号,可在*/linkdge/run/log_slf4j/blink*目录下通过ls命令查看,变 量*{container_id*}为当前容器ID。

消息路由路径:从边缘应用路由到IoT Hub

将边缘应用中的数据发送到云端。

↓ 注意 仅函数计算类型的边缘应用,支持使用消息路由。

下图中的示例表示,lightSensorDataFilter边缘应用过滤出符合 /productkey/devicename/user/# 的消息 主题,并将该消息主题中的消息发送给云端,当网络故障后,Link IoT Edge将lightSensorDataFilter边缘应 用无法上报的消息存储到本地,并在网络恢复后将本地数据重新上报到云端。

⑦ 说明 /productkey/devicename/user/# 是您自定义的一个发送消息到云端(IoT Hub)的消息 主题。具体设置方法请参见自定义Topic。

分配消息路由	×
* 路由名称	
边缘应用到IoTHub	
* 消息来源	
边缘应用 く	lightSensorDataFilter 🗸 🗸
* 消息主题过滤	
/productkey/devicename/user/#	
* 消息目标	
loT Hub	\sim
* 服务级别 🔞	
1	\sim
	确认取消

查看消息:边缘应用的数据上报到云端后,您可以在物联网平台控制台中,实例网关设备所属产品的日志服 务页面查看上报的数据。

消息路由路径:从边缘应用A路由到边缘应用B

将边缘应用A的数据发送到边缘应用B中。

↓ 注意 仅函数计算类型的边缘应用,支持使用消息路由。

下图中的示例表示,边缘应用A过滤出符合 /xxxx 的消息主题,并将该消息主题中的消息发送给边缘应用 B。

分配消息路由			\times
* 路由名称			
边缘应用A到边缘应用B			
* 消息来源			
边缘应用	\sim	lightSensorDataFilter	\sim
* 消息主题过滤			
/xxxx			
* 消息目标			
边缘应用	\sim	le object detector	\sim

查看消息:路由到边缘应用的数据只能在网关设备本地日志中查看。具体方法,请参见本文档消息路由路径: 从设备路由到边缘应用内容,查看消息日志的方法。

消息路由路径:从IoT Hub路由到边缘应用

云端发送消息至边缘应用中。

↓ 注意 仅函数计算类型的边缘应用,支持使用消息路由。

下图中的示例表示, IoT Hub滤条出符合 /a1******/gateway_01/user/get 的消息主题,并将此消息主题 中的消息发送给边缘应用。

⑦ 说明 云端使用以 /\$Your_ProductKey/\$Your_DeviceName/user 为前缀的消息主题,发送消息给边缘端。

分配消息路由		\times
*路由名称		
loTHub到边缘应用		
* 消息来源		
loT Hub		\sim
* 消息主题过滤		
/a1 /gateway_01/user/get		\sim
Topic 管理		
* 消息目标		
边缘应用 イン	lightSensorDataFilter	\sim
	确认	取消

查看消息:路由到边缘应用的数据只能在网关设备本地日志中查看。具体方法,请参见本文档<u>消息路由路径:</u> 从设备路由到边缘应用内容,查看消息日志的方法。

消息路由路径:从流数据分析路由到边缘应用

将流数据分析的数据发送到边缘应用中。

↓ 注意 仅函数计算类型的边缘应用,支持使用消息路由。

下图中的示例表示,高温报警流数据分析任务过滤出符合 /xxxx 的消息主题,并将该消息主题中的消息 发送给light SensorDataFilter边缘应用。

② 说明 /xxxx 是您自定义的一个发送消息到云端(IoT Hub)的消息主题。具体设置方法请参 见自定义Topic。

分配消息路由			\times
*路由名称			
流数据分析到边缘应用			
* 消息来源			
流数据分析	\sim	高温报警	\sim
* 消息主题过滤			
/xxxx			
* 消息目标			
边缘应用	\sim	lightSensorDataFilter	\sim
		确认	取消

查看消息:路由到边缘应用的数据只能在网关设备本地日志中查看。具体方法,请参见本文档<u>消息路由路径:</u> 从设备路由到边缘应用内容,查看消息日志的方法。

消息路由路径:从流数据分析路由到IoT Hub

将流数据分析的数据发送到云端。

下图中的示例表示,高温报警流数据分析任务过滤出符合 /xxxx 的消息主题,并将该消息主题中的消息 发送给云端。当网络故障后,Link IoT Edge将高温报警流数据分析任务无法上报的消息存储到本地,并在网 络恢复后将本地数据重新上报到云端。

② 说明 /xxxx 是您自定义的一个发送消息到云端(IoT Hub)的消息主题。具体设置方法请参 见自定义Topic。

分配消息路由		×
* 路由名称		
流数据分析到IoT Hub		
* 消息来源		
流数据分析	/ 高温报警	\sim
* 消息主题过滤		
/xxxxx		
* 消息目标		
loT Hub		\sim
* 服务级别 🕜		
1		\sim
		确认取消

查看消息:流数据分析的数据上报到云端后,您可以在物联网平台控制台中,实例网关设备所属产品的日志 服务页面查看上报的数据。

9.云资源访问

边缘实例中的函数计算、云监控等服务在网关上运行时,若需要调用阿里云其他云资源(例如调用OSS的 API),边缘实例使用RAM角色来实现对云资源权限的访问控制。

您可以参考RAM角色管理和权限策略管理内容,了解RAM角色和创建RAM角色并为角色授权的操作步骤。

前提条件

在开始操作本章内容前,请您确保已根据环境搭建内容创建完成边缘实例。

操作步骤

- 1. 登录边缘计算控制台。
- 2. 左侧导航栏选择边缘实例,找到需要分配角色的边缘实例,并单击对应操作栏中的查看。
- 3. 在实例详情页面,选择设置,为实例分配角色。
 - 分配已有角色
 - a. 单击**分配已有角色**,选择一个已创建的RAM角色。选择角色后,该角色拥有的权限会被使用到 边缘实例中。

若修改角色权限,可以单击**立即前往**跳转至RAM控制台,参考权限策略管理内容操作。

网关监控信	息 设备与驱动 场景	分配已有角色	×	
实例信息 ∠ 实例名称	编辑 LinkloTEdge_Node	① 选择已经存在的角色,可以授予边缘实例操作对应云资源的能力。	へ 边線 CPU 架株	© x86-64
创建时间	2019-11-04 10:25:02	* 角色 LinkloTEdgeRole	~	
Open API		織定取消	•	
实例标签				
实例角色管理				分配已有角色 添加角色与权限

b. 单击确定,完成边缘实例角色的添加。

○ 新建角色与权限

1 <u> </u>				 			
网关监控信	息 设备与驱动 场景	添加角色与权限	×				
	400.458	() 为边缘实例新建角色,同时也可以修改该角色的权限。					
实例信息 🖉	(0) 324	*角色名称					
实例名称	LinkloTEdge_Node	请输入角色名称		边缘 CPU 架构	x86-64		
创建时间	2019-11-04 10:25:02	*角色授权 🔞					
Open API 🕜		请选择角色权限	~				
实例标签		職定取消					
实例角色管理						分配已有角色	添加角色与权限

a. 单击**添加角色与权限**,设置角色与权限参数,为边缘实例新建角色。

参数说明

参数	描述		
角色名称	设置角色名称,支持英文大小写字母、数字和中划线(-),长度限制为1~64 个字符。		
角色权限	 为角色选择访问或调用阿里云其他资源时所需的权限,可以选择多个权限。 说明 鼠标可以移到选择框右侧的 ? 图标,可以快速地选择物联网 边缘计算常用的几个权限。 单击角色权限后的 x 图标,可以删除该权限。 		

- b. 单击确定,系统跳转到RAM控制台界面。
- c. 在RAM控制台界面再次确认参数后单击确定,完成边缘实例角色的添加。
- d. 返回边缘计算控制台,在实例详情页面设置页签中,可以看到已分配的角色以及该角色拥有的 权限。
- 4. (可选)后续如果在使用物联网边缘计算过程中,需要访问更多的云资源,则可以继续为已分配的角色 添加相应服务的权限。
 - i. 单击已分配角色右侧的编辑,更改角色或角色权限。

参数说明

参数	描述
角色	可以替换为其他已创建的角色。
角色权限	增加或删除已选角色的权限。 单击立即前往跳转至RAM控制台,参考权限策略管理内容,修改角色权限。 单击角色权限后的 × 图标,删除该角色权限。

ii. 在编辑角色与权限对话框,单击确定,保存修改的内容。

5. 添加角色完成后单击页面右上角**部署**,并在弹出对话框中,单击**确定**,部署边缘实例。

10.监控信息

物联网边缘计算提供对网关的主机监控和对驱动、函数计算、场景联动等资源的监控服务。

前提条件

- 1. 已根据环境搭建内容, 创建了边缘实例并上线网关。
- 2. 为边缘实例分配了驱动、函数计算、场景联动等资源,并成功部署实例。
- 3. 已根据云资源访问内容,为边缘实例添加了管理云监控(CloudMonitor)的权限。
- 4. 在实例详情页面监控信息页签下,已开启云监控状态按钮。

主机监控

创建边缘实例并上线网关后,可在**实例详情**页面**监控信息**页签选择**主机监控**,查看主机监控信息。 **主机监控**为您提供网关的CPU、内存、磁盘、网络等多种监控项,满足网关的基本监控运维需求。

用户指南·监控信息

物联网边缘计算



主机监控页签左上角选择待显示信息的采样周期后,可查看该采样周期内网关的如下表格监控项。

监控项说明

监控项	参数名称	说明
CPU使用率	cpu_total	当前已消耗CPU占总CPU的百分比。
内存使用率	memoryusage_utilizatio n	当前已使用内存占总内存的百分比。
	load_1m	过去1分钟的网关平均负载。
	load_5m	过去5分钟的网关平均负载。
糸统半均负载	load_15m	过去15分钟的网关平均负载。

监控项	参数名称	说明
磁盘使用率	diskusage_utilization	当前已使用磁盘占总磁盘的百分比。
网络流入流出速率	networkin_rate	网卡每秒接收的比特数,即网卡的上行带宽。
(bps)	networkout_rate	网卡每秒发送的比特数,即网卡的下行带宽。
	networkin_errorpackage s	网卡每秒接收的错误包数量。
网络流入流出数据包	networkin_packages	网卡每秒接收的数据包数量。
(pps)	networkout_errorpacka ges	网卡每秒发送的错误包数量。
	networkout_packages	网卡每秒发送的数据包数量。

您可以为主机监控设置报警配置,请按如下步骤操作。

1. 在监控信息页签下单击报警配置。

2. 在您边缘实例对应的云监控控制台报警规则页签中,新建报警规则。

其中,新建报警规则时关联资源无需更改,其余参数说明及详细操作请参见使用报警服务。

在实例进程区域框中可查看实例各个进程的详细信息和CPU使用率、内存使用率等情况。

场景联动

⑦ 说明 仅专业版和标准版的边缘实例有场景联动监控。

分配场景联动到边缘实例并部署实例成功后,在**监控信息**区域框的**场景联动**页签下,单击场景名称右侧的查看,查看场景联动的监控信息。

在场景联动监控详情页面左上角选择待显示信息的采样周期后,可查看该采样周期内场景联动的触发次 数和异常次数。

边缘计算 / 边缘实例 / 实例详情 / 监控详情			
← Rule_Light_Open 运行中 最近 1 小时 ∨	C		
触发次数 💿	异常次数 💿		
10 9.8 9.6 9.4 9.2 9	暂无数据		
19:08 19:18 19:28 19:38 19:48 19:58 ● 触送之次政(次)			

边缘应用

⑦ 说明 仅专业版和标准版的边缘实例有边缘应用监控。

分配边缘应用到边缘实例并部署实例成功后,在**监控信息**区域框的**边缘应用**页签下,单击应用名称右侧的查看,查看边缘应用的监控信息。

在边缘应用监控详情页面左上角选择待显示信息的采样周期后,	可查看该采样周期内边缘应用的如下表格监
控项。	



监控项说明

监控项	说明
调用次数	采样周期内函数被调用的次数。
执行时间	每次调用函数时,平均处理请求的时间,单位为毫秒。
退出次数	采样周期内函数所属进程或者容器退出的次数。

驱动监控

⑦ 说明 仅专业版和标准版的边缘实例有驱动监控。

分配驱动和子设备到边缘实例并部署实例成功后,在**监控信息**区域框的**驱动监控**页签下,单击驱动名称右侧的**查看**,查看驱动的监控信息。

在驱动监控详情页面左上角选择待显示信息的采样周期后,可查看该采样周期内驱动所属进程或者容器的退 出次数。

 LightSensor 运行中 最近1小时 ✓ 退出次数 ● 19:14 19:24 19:34 19:44 19:54 20:04 ● 退出次数(次) 	象计算 / 边缘实例	列 / 实例详	青 / 监控详情				
最近 1 小时 マ	- LightS	ensor	运行中				
退出次数 ② 19:14 19:24 19:34 19:44 19:54 20:04 ● 退出次数(次)	最近1小时	\sim					
0 19:14 19:24 19:34 19:44 19:54 20:04 ● 退出次数(次)	退出次数 💿						
0 19:14 19:24 19:34 19:44 19:54 20:04 ● 退出次数(次)							
0 19:14 19:24 19:34 19:44 19:54 20:04 ● 退出次数(次)							
0 19:14 19:24 19:34 19:44 19:54 20:04 ● 退出次数(次)							
19:14 19:24 19:34 19:44 19:54 20:04 退出次数(次) 	0	-					_
 退出次数(次) 	19:14	19:24	19:34	19:44	19:54	20:04	
			•	退出次数(次)			

流数据监控

⑦ 说明 仅专业版的边缘实例有流数据监控。

分配流数据任务到边缘实例并部署实例成功后,在**监控信息**区域框的**流数据监控**页签下,单击任务名称右侧的**查看**,查看流数据任务的监控信息。

在流数据监控详情页面左上角选择待显示信息的采样周期后,可查看该采样周期内流数据任务的被触发次 数。

边缘计算 / 边缘实例 / 实例	洋情 / 监控详情			
平均温度运行中				
流数据监控信息				
最近1小时 🗸				
触发次数 💿				
	•	触发次数(次)		
10				
8 -				
6 -				
4 -				
2 -				
0			,	
16:23 16:33	16:43	16:53	17:03	17:13

11.远程运维管理

11.1. 远程服务访问

远程服务访问包含了远程连接(SSH协议)、远程文件管理(SFTP协议)和其他基于TCP协议的网络服务远 程访问。

Link IoT Edge的v1.8及以上版本默认具有远程服务访问功能,同时也允许您单独集成该功能。集成方法,请参见git hub开源库。

远程连接和远程文件管理基于SSH技术实现,方便您远程登录网关设备进行设备控制和文件管理,其使用的 用户名和密码是Docker镜像或网关上SSH服务的用户名与密码。

其他网络服务的远程访问,在通过控制台操作后,由系统拉起远程访问助手来实现。

准备工作

- 1. 登录边缘计算控制台。
- 2. 创建边缘实例并上线网关。具体操作步骤,请参见环境搭建。
- 3. 边缘实例创建完成后,在该边缘实例的实例详情页面选择网关页签,开启远程访问功能。



4. 登录网关设备,开启SSH服务,且端口默认为22。开启SSH服务方法请参见OpenSSH Server。

⑦ 说明 如果端口有变更,需要局域网登录网关,并手动修改/linkedge/gateway/build/config/remote_terminal.json中的port字段为新的端口,同时网关必须支持 127.0.0.1 本地回环端口的访问。

远程SSH终端

- 1. 在边缘计算控制台左侧导航栏选择边缘实例。
- 2. 选择已创建的实例,单击右侧的查看。
- 3. 在**实例详情**页面,选择网关页签,在网关名称右侧操作栏中单击远程SSH终端。

网关	监控信息	设备与驱动	场景联动	边缘应用	消息路由	日志服务	设置			
关联网关	软件安装	Ē								
网关名称	t	产品名	称			远程访问		状态	最后上线时间	操作
gateway_	_01	LinkloT	FEdge_Gateway					● 在线	2020-02-19 15:47:04	查看 分享远程SSH终端 远程SSH终端 更多远程服务

4. 在弹出对话框中,输入远程登录网关的用户名和密码,单击确定。

连接成功后,您将看到如下远程控制台页面。

注意 远程登录网关,具有高级权限,可进行更新、删除、改动、查看各类网关设备的信息和 文件。请谨慎使用。
〔-〕 返回物联网平台		lge_Gateway1 PoductKe	
gateway_01 恨目录 / root Q			<pre>Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-170-generic x86_64) * Documentation: https://help.ubuntu.com * Management: https://landscape.canonical.com</pre>
上传文件 新增目录	文件十八	C 海炸	<pre>* Support: https://ubuntu.com/advantage * Multipass 1.0 is out! Get Ubuntu VMs on demand on your Linux, Windows or Mac. Supports cloud-init for fast, local, cloud devops simulation.</pre>
▲ 回上级	XIIXA	4 1 7 0	https://multipass.run/ New release '18.04.4 LTS' available. Run 'do-release-upgrade' to upgrade to it.
.bash_history	790 B	下载 删除	Welcome to Alibaba Cloud Elastic Compute Service !
.bashrc	3.03 KB	下载 删除	Last login: Wed Feb 19 15:58:30 2020 from 112.10.64.162
Cache/	4.00 KB	查看 删除	
D.pip/	4.00 KB	查看 删除	
.profile	148 B	下载 删除	
.pydistutils.cfg	205 B	下载 删除	
.ssh/	4.00 KB	查看 删除	

5. (可选)在远程控制台,可以对远程文件进行管理。

目前远程文件管理提供目录浏览与创建,文件上传与下载等功能。各功能说明如下所示。

操作	说明		
刷新	刷新远程文件目录。		
	单击 新增目录 ,在弹出的对话框中输入目录名称并单击 确定 ,可增加一个 新的远程文件目录。		
新增目录	⑦ 说明 目录名称支持中文、英文字母和数字,长度不超过64个 字节。		
上传文件	单击 上传文件 ,在弹出的对话框中单击上传按钮上传文件,上传成功后单 击 确定 ,可上传本地文件到远程文件目录。		
下载	单击 下载 ,可下载远程文件到本地。		
	单击 删除 ,可删除远程文件。		
删除	注意 删除远程文件后不可恢复,请谨慎操作。		

分享远程服务

1. 在**实例详情**页面,选择**网关**,在网关名称右侧操作栏中单击**分享远程SSH终端**。

2. 在系统弹出的对话框中打开远程分享按钮。

文変時安楽 分享远程服务 × 図 大名称 产品名称 ① 注意: 关闭分享功能,将禁止新的连接产生,但并不会主动断开处于连接状态的控制台,如需关闭所有连接,请直接关闭远程调试功能。 最后上编时间 操作 gateway_01 LinkloTEdge,G 正程分享 最后上编时间 操作 远程分享 三米印 ① 支援分享 三米印 ① 支援分享 三米印 ① 支援分 三米印 ①	网关监控信息	设备与驱动 场景型	法 计保持用 迷白败山 日士服务 计	<u></u>		
网关名称 gateway_01 LinkloTEdge_G gateway_01 LinkloTEdge_G gateway_01 gateway_01 LinkloTEdge_G gateway_01 LinkloTEdge_G gateway_01 LinkloTEdge_G gateway_01 LinkloTEdge_G GR25 gateway_01	关联网关 软件安装		分享远程服务	×		
gateway_01 LinkloTEdge_Gi 近程分享 上規制	网关名称	产品名称	1 注意:关闭分享功能,将禁止新的连接产生, 动断开处于连接状态的控制台,如需关闭所有	,但并不会主 有连接, 请直	最后上线时间	操作
	gateway_01	LinkloTEdge_Ga	接关闭远程调试功能。		2020-02-19 15:47:04	查看 分享远程SSH终端 远程SSH终端 更多远程服务
				关闭		

- 在分享远程SSH终端对话框中显示远程分享的链接和密码,单击复制链接和密码将远程服务分享给其 他人。
 - ? 说明
 - 开启远程分享24小时后,将自动关闭分享,届时被分享远程服务的用户不可再次登录远程服务,但已登录的用户仍然可以继续使用。若要禁止已登录的用户使用远程服务,请在网关页签下关闭远程访问开关。
 - 单击重置密码,系统会重新分配密码。重置密码后,其他人无法通过重置前的密码再次登录远程服务,但已登录的用户仍然可以继续使用,退出登录后则不可使用。
- 4. 获取到分享远程服务链接的用户,按如下步骤登录设备,使用远程服务。
 - i. 在浏览器中打开分享链接, 输入分享密码。
 - ii. 输入由分享者提供的设备用户名和密码。
 - iii. 成功登录设备,使用远程服务。

更多远程服务

更多远程服务有如下两种连接情况:

• 远程访问网关的SSH和SFTP服务时,可通过网关的Linux系统用户名和密码,登录远程服务。

⑦ 说明 在边缘网关Linux系统命令行中执行whoami,即可获取用户名;若Link IoT Edge在边缘网关的Docker环境中运行,则默认用户名为root,密码为iiot。

在**实例详情**页面,选择**网关**页签,在网关名称右侧操作栏中单击**更多远程服务**,并在弹出窗口中选择需 要访问的SSH或SFTP服务。

← LinkloTEdge_Node-	更多远程服务	×		查看部署日志 重重 部署
 产品規格 5% CPU使用率 - 查看 网关< 监控信息 设备与驱动 场景 	● 特别说明:如需访问非SSH、SFTP、HTTP、HTTPS协议的远程服务,您需要先下载远程服务助手。立即下载什么是远程助手?		1	
关联网关 软件安装	* 远程服务			
	HTTPS openapi_linkedge	^		
网关名称 产品名称	✓ HTTPS openapi_linkedge	Â	最后上线时间	操作
gateway_01 LinkloTEdge_Ga	HTTP http_localhost		2020-02-19 15:47:04	查看 分享远程SSH终端 远程SSH终端 更多远程服务
	SSH ssh_hostmachine			
	SFTP sftp_hostmachine			
	SSH ssh_localhost			
	SFTP sftp_localhost	-		
	请输入资源路径			
	使用文档	消		

② 说明 访问宿主机的远程服务SFTP sftp_hostmachine和SSH ssh_hostmachine, 仅支持在使用 Link IoT Edge专业版, 且版本号为v2.0.0及以上版本的网关中进行远程连接。

- 若使用非SSH、SFTP服务时,需要您在电脑设备上下载并安装远程访问助手概览。并按如下步骤操作:
 - i. 在实例详情页面,选择网关页签,在网关名称右侧操作栏中单击更多远程服务。
 - ii. 在弹出对话框的远程服务下拉框中选择HTTP openapi_linkedge后,单击确定。

更多远程服务	×
针别说明:如需访问非SSH、SFTP、HTTP、HTTPS协议的远程服务,您需要先下载远程服务助手。立即下载什么是远程助手?	
* 沅程服务	
HTTPS openapi_linkedge	\sim
新增	
服务详情	
IP 地址: 127.0.0.1 端口号: 9999	
资源路径 https://127.0.0.1:9999/	
请输入资源路径	
使用文档 取消	í

iii. 在系统弹出的对话框中,选中始终在关联的应用中打开这些类型的链接复选框,单击打开Remote Access Assistant。

要打开 Remote Access Assistant 吗?				
✓ 始终在关联的应用中打开这些类型的链接				
	打开 Remote Access Assistant	取消		

系统跳转到边缘网关控制台登录页面,输入初始账号和密码后,您可以登录边缘网关控制台,进行远 程访问操作。



 ✓ 注意 边缘网关控制台初始账号为 admin , 初始密码为 admin1234 , 登录边缘网关控制 台后,请您及时通过账户管理修改初始账号和密码。

自定义远程服务

在更多远程服务中,允许添加自定义远程服务。有如下两种添加方式:

⑦ 说明 最多可新增20个自定义远程服务。

- 动态添加自定义远程服务
 - i. 在实例详情页面,选择网关页签,在网关名称右侧操作栏中单击更多远程服务。
 - ii. 在远程服务下拉框下方单击新增。

更多远程服务	×
● 特别说明:如需访问非SSH、SFTP、HTTP、HTTPS协议的远程服务,您需要先下载远程服务助手。立即下载什么是远程助手?	
* 远程服务	
HTTPS openapi_linkedge	\sim
新增	
服务详情	
IP 地址: 127.0.0.1 端口号: 9999	
资源路径 https://127.0.0.1:9999/	
请输入资源路径	
使田文档	щ
	-

iii. 在弹出的对话框中设置参数,然后单击确定。

参数	描述
	根据边缘端的远程服务类型,从下拉框中选择服务类别。
服务类别	⑦ 说明 若服务类型不是SSH、SFTP、HTTP、HTTPS类型,则 选择自定义。
服务名称	设置远程服务的名称,支持英文字母、数字、中划线(-)、下划线 (_)、at符号(@)、英文句号(.)、英文冒号(:)。
IP地址	远程服务的IP地址。
端口号	远程服务的端口号。
用户名	输入远程登录网关的用户名。
密码	输入远程登录网关的用户名和密码。

iv. (可选)您可以删除已添加的自定义远程服务。

在更多远程服务对话框的远程服务参数下拉框中找到需要删除的远程服务,单击远程服务名称后的 前 图标,在弹出对话框中单击确定删除远程服务。

更多远程服务	×
● 特别说明:如需访问非SSH、SFTP、HTTP、HTTPS协议的远程服务,您需要先下载远程服务助手。立即下载什么是远程助手?	
* 远程服务	
SSH SSH_test1111	^
HTTP http_localhost	•
SSH ssh_hostmachine	
SFTP sftp_hostmachine	
SSH ssh_localhost	
SFTP sftp_localhost	
✓ SSH SSH_test1111]

- 静态添加自定义远程服务
 - i. 登录您的网关设备。
 - ii. 进入/*linkedge/gateway/build/config/remote_terminal.json*文件,根据文件中的模板,添加自定义 远程服务。

 ↓ 注意 通过边缘网关静态添加的远程服务,可显示在控制台实例详情页面网关页签下的更 多远程服务列表中,但无法从该列表中删除。

文件中的模板示例如下所示。

1		
2	{	
3		"type": "SSH",
4		"name": "sftp_localhost",
5		"ip": "127.0.0.1",
6		"port": 22
7	},	
8	{	
9		"type": "VNC",
10		"name": "VNC_localhost",
11		"ip" : "30.43.83.246",
12		"port": 5900
13	},	
14	{	
15		"type": "HTTPS",
16		"name": "HTTPS_eweb",
17		"ip" : "127.0.0.1",
18		"port": 9999
19	}	
20		
21		

11.2. 远程访问助手

11.2.1. 远程访问助手概览

远程访问助手是阿里云IoT提供的,针对使用Link IoT Edge轻量版本(LE Lite)连接到物联网边缘计算的设备,进行远程访问的跨平台桌面版客户端软件。

远程访问助手可以跨公网访问远程设备的内置网络服务(SSH/TELNET/HTTP/RDP/ADB等),也可以配合阿 里云物联网平台上远程连接与远程文件管理功能访问HTTP服务。

主要功能如下:

- 集成阿里云账号体系(支持密钥自动保存)。
- 一键获取阿里云物联网边缘计算设备列表(支持翻页/跳转/检索等辅助功能)。
- 一键开启对指定设备的远程访问。
- 实时查看操作日志。

软件下载

操作系统	下载链接
Windows	Windows7及以上版本,32/64位操作系统: <mark>链接</mark>
macOS	OSX 10.10及以上版本操作系统: 链接
Ubuntu	Ubuntu12.04及以上版本, 32位操作系统: <mark>链接</mark>
	Ubuntu12.04及以上版本,64位操作系统: <mark>链接</mark>

11.2.2. 软件安装

本章描述在Windows、Ubuntu、macOS系统中安装远程访问助手的详细步骤。

前提条件

已下载适配您设备系统的远程访问助手软件包,软件包下载请参见软件下载。

Windows系统上的软件安装

1. 双击已下载的Windows版本远程访问助手软件,在系统弹出框中单击是。

	 用户帐户控制
E	 您想允许来自未知发布者的以下程序对此计算机进行更改吗?
Alibaba¢lo IoT Remot	程序名称: AlibabaCloud IoT Remote Access Assistant-v2.4.exe 发布者: 未知 文件源: 此计算机上的硬盘驱动器 程序位置: "C:\Users\tiger\Desktop\AlibabaCloud IoT Remote Access Assistant-v2.4.exe" /SPAWNWND=\$C0476 /NOTIFYWND=\$70340
	▲ 隐藏洋细信息(D) ■ これには、一番(N) ■ これには、「あれた」、「あれた」は、「あれた」は、「あれた」は、「あれた」は、「あれた」は、「あれた」は、「あれた」は、「あれた」は、「あれた」は、「あれた」は、「あれた」」、「あれた」」、「あれた」」、「あれた」」、「あれた」、「あれた」、「あれた」」、「あれた」」、「あれた」、「ままた」、「あれた」、「あれた」、「ままた」、「あれた」、「る」、「る」、「る」、「る」、「る」、「る」、「る」、「る」、「る」、「る

2. 在License Agreement 对话框,阅读协议内容后,接受用户协议并单击Next。

🖸 Setup - AlibabaCloud IoT Remote Access Assistant version 2.4 🛛 📼 💌
License Agreement Please read the following important information before continuing.
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.
本服务协议是阿里云计算有限公司(以下简称"阿里云")与您就阿里 🔺 云所提供产品及/或服务的相关事项所订立的有效协议。《物联网平 🗐 台的服务条款》以下或简称 "本协议"。
阿里云的产品、服务或产品及/或服务,以下或统称为"服务"。在 接受本协议之前,请您仔细阅读全部内容;如对相应条款有疑问,请 ↓ I accept the agreement ○ I do not accept the agreement
Next > Cancel

- 3. 在Information对话框,单击Next。
- 4. 在Select Additional Tasks对话框,选择创建桌面快捷方式,并单击Next。

CO Setup - AlibabaCloud IoT Remote Access Assistant version 2.4 📃 📼 💌
Select Additional Tasks Which additional tasks should be performed?
Select the additional tasks you would like Setup to perform while installing AlibabaCloud IoT Remote Access Assistant, then click Next.
Additional shortcuts:
Create a desktop shortcut
< Back Next > Cancel

5. 等待软件安装完成后,在系统弹出窗口中单击Finish。

6. 打开已安装好的软件,可以看到如下远程访问助手界面。

1922-00								
rcessKeyID		Access	sKeySecret				📝 自动保	存密钥信!
备列表								
导输入DeivceName	请输入标签名	3. 请辅	俞入标签值	搜索			刷新	动远程访
ProductKey	DeviceName	产品名称	设备标签	数据通道状态	运维通道状态	最后上线时间	运维配置信息	
					上一页 当前!	页一下一页	到第 1 🚽 页 🗌	确定

Ubuntu系统上的软件安装

1. 双击已下载的Ubuntu版本远程访问助手软件,然后单击Install。

😕 🖨 🗊 Ubun	tu Software Center	
< >	All Software	
	remote.access.assistant 阿里云IoT远程访问助手,帮助用户便捷访问远程设备。	
Only install	this file if you trust the origin.	Install
		No screenshot available
Version	remote.access.assistant 2.4.0	
Total size	89.6 MB when installed	
License	Unknown	
Updates	Unknown	

2. 在系统弹出窗口中单击Ignore and install,并等待安装完成。

- < >	🗳 🗸	Installed	•	History				
	「EMOLE 阿里云IoT远程访问		SS.a 用户便捷i	ssistant ^{疠向远程设备。}				
Only install	this file if you t	rust the o	rigin.					Install
		♥ Details Lintian chee E: remote.a	The pac The insta This could or organi :k results cccess.ass	ckage is of bad quality llation of a package which violates t d cause serious problems on your co sation who provided this package fi for /home/tiger/Desktop/RemoteA istant: control-file-has-bad-permiss	he quality standards isn't allo mputer. Please contact the p le and include the details bene ccessAssistant.deb: ions postinst 0555 I= 0755	wed. erson eath.	No screenshot available	
Version Total size	remote.access.as 89.6 MB when inst					- 1		
License	Unknown					- 81		
Updates	Unknown				OK Ignore and i	install	'RemoteAccessAssistant.deb''	selected

3. 打开已安装好的软件,可以看到如下远程访问助手界面。

8	■ 回 阿里云IoT边缘计算远程访问助手
访	可密钥
	AccessKeyID AccessKeySecret 自动保存密钥信息
iQ i	备列表
	请输入DeivceName 请输入标签名 请输入标签值 搜索 刷新 启动远程访问
	ProductKey DeviceName 产品名称 设备标签 数据通道状态 运维通道状态 最后上线时间 运维配置信息
	上一页 当前页 下一页 到第 1 🖕 页 确定
调试信息输出窗口	x迎使用阿里云IoT Link IoT Edge边缘计算远程访问服务。 版本号: 2.4.0 构建时间: 2019-03-12 16:08:59 。 请到阿里云官网申请 AccessKeyID/AccessSecretKey: https://ram.console.aliyun.com/users/new(注意,访问方式需勾选"编程访问",同时为该用户分配 IoT的读写权限"AliyunIOTFullAccess") 如有问题,请到 https://github.com/alibaba/iot_remote_access/issues 反馈或钉钉搜索 23115744 入群咨询。

macOS系统上的软件安装

1. 在 系统偏好设置 > 安全性与隐私 > 通用里,确保允许安装第三方应用。

	安全性与隐私	Q 搜索
	通用 FileVault 防火墙 隐私	
已经给此用户设定登录器 ✓ 进入睡眠或开始 ● 在屏幕锁定时显 ✓ 停用自动登录	8码 更改密码… 屏幕保护程序 5分钟后 ◆ 要求输入 示信息 设定锁定信息…	密码
允许从以下位置下载的应 App Store ● App Store 和被i	7用: 人可的开发者	
点按锁按钮以进行更改。		高级 ?

2. 双击已下载的macOS版本远程访问助手软件,然后单击继续。

	欢迎使用"RemoteAccessAssistant"安装器
0 介绍	您即将安装阿里云IoT远程访问助手以跨公网访问边缘侧设备网络服务。
许可	在使用本助手时,请您务必注意以下事项:
目的宗卷	1. 请务必妥善保管您录入本软件的AccessKeyID和AccessKeySecret(以下简称阿里云身份 信息), 谨防泄漏给他人。
安装类型	 为给您提供更好的使用体验,本软件在您录入'阿里云身份信息'并勾选'自动保 方' E 会教这信息保存在你的申請上 读关注田本人申除遗生局教的信息 逆震风险
安装	 法有法言思味好让思问电脑上,相关在这一人电脑是关于我的言思很确认应。 请合法合理使用本软件。
	如遇软件安装问题,请在阿里云官网通过工单联系物联网平台技术支持或关注 <u>https://github.com/alibaba/iot_remote_access</u> 以获得帮助。
	返回 继续

- 3. 在软件许可协议对话框,阅读协议内容后单击继续。
- 4. 在系统弹出的确认许可协议对话框中单击同意,继续安装软件。
- 5. 执行软件安装。

	标准安装将执行于"Macintosh HD"上
 介绍 许可 目的宗卷 安装类型 安装 摘要 	这将占用您的电脑上的 56.8 MB 空间。 点按"安装"以在磁盘"Macintosh HD"上执行此软件的标准安 装。
	自定返回安装

6. 输入macOS系统设备用户名与密码。

介绍 许可	这将占用您的电脑上的 56.8 MB 空间。
目的宗卷 安装类型 安装 摘要	"安装器"正在尝试安装新软件。 输入密码以允许此次操作。 用户名: 密码:
	取消 安装软件

7. 等待安装完成后,可以选择保留安装包。

 介绍 许可 目的宗卷 	☞ 安装 "RemoteAccessAssistant" 您要将 "RemoteAccessAssistant" 安装器移到 废纸篓吗? 若要將此软件包保留在其当前位置,请点按"保留"。 保留 移到废纸篓
 安装类型 安装 摘要 	安装成功。 ^{软件已安装。}
	返回 关闭

可在应用程序中查看已安装完成的远程访问助手。

8. 打开已安装好的软件,可以看到如下远程访问助手界面。

the location and					[] []	可里云 loT 边缘计算	远程访问助手			
访问密钥										
AccessKeyID				AccessKey	Secret					自动保存密钥信息
设备列表										
请输入DeivceN	ame	请输入标签名		请输入核	示签值	援	索			刷新 启动远程访问
ProductKey	DeviceName	产品名称	设备标签	数据通道状态	运维通道状态	最后上线时间	运维配置信息			
									上一页当前	瓦下一页 到第 1 0 页 确定
	로loT Link loT Edge	边缘计算远程访问	可服务。							
欢迎使用阿里 版本号: 2.4.0	云loT Link loT Edge 构建时间: 2019-03	。边缘计算远程访问 3-13 18:15:24 。	可服务。				雨力注:"伦理计问: 反	四十天,汝田 白八和1,十小天来	〒40 55 IIIA (:) (ATT-: 川	
x迎使用阿里版本号:2.4.0 请到阿里云宫如有问题,请	云IoT Link IoT Edge 构建时间: 2019-03 砌申请 AccessKeyl 刮 https://github.c	边缘计算远程访问 3-13 18:15:24 。 D/AccessSecret com/alibaba/iot_	可服务。 Key: https://rar remote_access,	n.console.aliyun. /issues 反馈或钉{	.com/users/new 钉搜索 23115744	 (注意,访问方式 入群咨询。	需勾选"编程访问",同	时为该用户分配IoT的读	写权限"AliyunlOTFull/	ccess")
欢迎使用阿里 版本号: 2.4.0 请到阿里云宫 如有问题,请	云loT Link loT Edge 构建时间: 2019-03 砌申请 AccessKeyl 別 https://github.c	o边缘计算远程访问 3-13 18:15:24。 D/AccessSecret com/alibaba/iot_	可服务。 Rey: https://rar remote_access,	n.console.aliyun. /issues 反馈或钉	.com/users/new 钉搜索 23115744	 (注意, 访问方式 入群咨询。 	需勾选"编程访问",同	时为该用户分配loT的读	写权限"AliyunlOTFull <i>l</i>	uccess")

11.2.3. 使用方法

软件安装完成后,您可以根据本章描述,使用远程访问助手。

前提条件

接入物联网边缘计算的设备具有SSH等服务时,需要在您的Windows/macOS/Ubuntu机器上安装第三方远

程桌面软件(例如, Putty/WinSCP/Xshell/SecureCRT/Windows远程桌面),以支持在远程访问助手**调试** 信息输出窗口输出设备信息。

⑦ **说明** 如果这些第三方远程桌面软件需要用户名和密码,则为设备上该服务的用户名和密码,例如 SSH服务所需要的用户名和密码为设备上Linux系统用户名和密码。

下文操作以macOS系统图片为例。

操作步骤

- 1. 创建RAM用户,并为用户配置权限。
 - i. 以阿里云主账号登录RAM控制台。
 - ii. 参考访问控制用户文档, 创建RAM用户。

↓ 注意 其中,访问方式必须选择编程访问,否则无法使用远程访问助手;为便于管理用户,建议不要勾选控制台密码登录。

RAM访问控制		RAM访问控制 / 用户 / 新建用户	
概览		← 新建用户	
人员管理	^	* 用户账号信息	
用户组		登录名称 👔	显示名称 🕜
用户		test_user @ .onaliyun.com	test_user
设置		+ 添加用户	
SSO 管理		访问方式 🕗	
		· 控制台密码登录 用户使用账号密码访问阿里云控制台	
秋 限管埋	^	✓ 编程访问 启用 AccessKey ID 和 AccessKey Secret,支持通过API或其他升	开发工具访问
授权	<	确定 返回	

 iii. 用户创建完成后系统自动跳转到如下图页面,请务必保存好AccessKeyID和 AccessKeySecret。该 两个值会在连接远程访问助手操作中使用到。

↓ 注意 本页面仅在创建完用户后显示一次,即AccessKeyID和 AccessKeySecret仅在此页面获取,若不保存或丢失,只能重新创建新的AccessKey。

RAM访问排	空制 / 用户 / 新建用户					
←新	建用户					
0 若	开通编程访问,请及时保存AccessKey 信息,页面关闭后将无法再次获取	信息。				
用户信息	R.					
下载CS	5V文件					
	用户登录名称	状态	登录密码	AccessKey ID	AccessKeySecret	操作
	test_user@onaliyun.com	● 成功	无	Tests (percention)	distant and services	□ 复制
返回						

iv. 返回用户信息页面,选择权限管理,单击添加权限为用户配置AliyunIOTFullAccess的权限。

(物戦與平台 / 边線注質 / 边線実例 / 実例详情 / 监控详備 ← Rule_Light_Open 运行中 最近 1 小时 ✓	G
触发次数 ● 10 - 9.8 - 9.6 -	异常次数 ③
9.4 - 9.2 - 9 19:08 19:18 19:28 19:38 19:48 19:58 ● 触题之次既次为	暂无数据

- 2. 在相应系统设备中打开已安装的远程访问助手(Remote Access Assistant或remote.access.assistant) 软件。
- 3. 输入步骤1中新建用户时获取的AccessKeyID和 AccessKeySecret。

	0						 Implementation 	程访问助手				
用	户信息											
1	AccessKeyID	T. M	т		Access	KeySecret •••••		•••••				🗹 自动保存密钥
设	备列表											
	请输入DeivceNa	ame			搜索							刷新 启动远程访问
	ProductKey	DeviceName	产品名称	节点类型	数据通道状态	运维通道状态	最后上线时间	创建时间			激活时间	
										চ —ক		到第10页 確定
	欢迎使用阿里	云IoT Link IoT Ed	dge边缘计算远程	访问服务。								
御田			10.00.14-0.4-00									
18 B	版本号: V2.2 请到阿里云官	构建时间: 2018- 网申请 AccessKi	-12-28 11:04:52 evID/AccessSec	。 cretKev: https:	//ram.console.a	livun.com/user	s/new (注意, 访	问方式需勾诜'	编程访问"。同时头	i该用户至少分配loi	的读权限"AlivunIOTReadC)nlvAccess")
编试信	如有问题,请	到 https://githu	b.com/alibaba/i	iot_remote_aci	cess/issues 反馈	或钉钉搜索 231	15744 入群咨询。					

4. 输入正确的用户信息后,单击刷新获取设备列表。

·信息								
cessKeyID LT	т	· · · · · · · · · · · · · · · · · · ·	AccessKeySecret	•••••	•			🗹 自动保存
列表								
输入DeivceName		援索						刷新 启动远程访问
ProductKey	DeviceName	产品名称	节点类型	数据通道状态	运维递道状态	最后上线时间	创建时间	激活时间
a1 8x	aaaaa	eeee	网关	离线	离线	2018-12-20 16:28:16	2018-12-20 16:01:55	2018-12-20 16:02:16
a1 9b	e18512f	dddd	网关	在线	在线	2018-12-27 12:46:26	2018-12-20 10:53:37	2018-12-20 16:28:44
a1 9b	e5a1d12	dddd	网关	在线	离线	2018-12-28 10:14:57	2018-12-20 10:53:37	2018-12-25 13:45:21
a1 9b	ecd8508	dddd	网关	在线	在线	2018-12-25 15:25:13	2018-12-20 10:53:37	2018-12-25 15:25:10
a1 9b	e835718	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	f751f43	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	fdbb354	dddd	网关	离线	离线	2018-12-20 16:35:50	2018-12-20 10:53:37	2018-12-20 16:35:50
a1 9b	f55b6e8	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	e23ff44	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	e185b74	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	eb07e49	dddd	网关	未激活	离线		2018-12-20 10:53:37	
2 a1 9b	eb0e165	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	f7bcc00	dddd	网关	未激活	离线		2018-12-20 10:53:37	

当前用户名下的设备列表,用户可以双击造中设备进行进程访问,也可以通过远程访问按钮进行访问。注意,当前同一时刻仅支持单个设备的进程访问。

列表信息说明

参数	说明
数据通道状态	设备MQTT通道的状态,即在物联网平台控制台上所看到的设备状态。
运维通道状态	设备调试通道的状态。

5. (可选)在搜索栏中,输入具体的设备DeviceName,单击搜索,可以看到该设备的设备详情。

•	•				• 阿里云IoT 远程访问助手					
Acc	essKeyID	MT	Act	cessKeySecret					💟 自动保存密钥	
设备列 aaa	ik a		授案						刷新 启动远程访问	
Ē	ProductKey	DeviceName	产品名称	节点类型	数据通道状态	运维通道状态	最后上线时间	创建时间	激活时间	
1 .	a1 rn	aaaa	gateway_	网关	离线	离线	2018-06-15 11:32:51	2018-06-14 18:00:06	2018-06-14 18:00:53	
		总共1页				上一页 当前第1页	下一页		到第 1 0 页 确定	
11 5 11	I:46:14", "ProductNam and request done, wa 'PageCount":1, "PageS 3:00:53", "Status": "OF - 30"	ne":" test_de iting for response Size":20,"Page":1,"Total" FLINE", "NodeType"1,"C)	vice", "ChildDeviceCoun :1, "Success":true, "Data CreateTime": "2018-06-1	t":0, "StatusRemoteDeb ":{"SimpleDeviceInfo":[(4 18:00:06", "ProductNa	ug":"offline"}]},"Reque ("DeviceName":"aaaa", ame":"gateway_","Child	stid":"58D73E90- "ProductKey":"a1 DeviceCount":1,"Statusf	, rn","LastOnlineTime":"2 temoteDebug":"offline"	A7"} 2018-06-15 11:32:51","A }]], "Requestid":"12BBC4	ctiveTime":"2018-06-14 IEC	
兴动中	白夕下的沿名列来 甲白	っつい双手進由役名が行に利	3.访问 中可以通过运程注	回始短进行访问 注意 业	前同一时刻仅去转首个站。	医的远程访问				

6. 在设备列表中,双击设备所在行,或者选中设备所在行后单击右上角**启动远程访问**,对该设备进行远

程访问。

在系统弹窗中,选择 Yes ,即可看到如下图界面。

cessKeyID LT	MT		AccessKeySecret		•			🔽 自动保存
列表								
输入DeivceName		援索						刷新 关闭远程访问
ProductKey	DeviceName	产品名称	节点类型	数据通道状态	运维通道状态	最后上线时间	创建时间	激活时间
a1 8x	aaaaa	eeee	网关	离线	离线	2018-12-20 16:28:16	2018-12-20 16:01:55	2018-12-20 16:02:16
a1 9b	e18512f	dddd	网关	在线	在线	2018-12-27 12:46:26	2018-12-20 10:53:37	2018-12-20 16:28:44
a1 9b	e5a1d12	dddd	网关	在线	离线	2018-12-28 10:14:57	2018-12-20 10:53:37	2018-12-25 13:45:21
a1 9b	ecd8508	dddd	网关	在线	在线	2018-12-25 15:25:13	2018-12-20 10:53:37	2018-12-25 15:25:10
a1 9b	e835718	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	f751f43	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	fdbb354	dddd	网关	离线	离线	2018-12-20 16:35:50	2018-12-20 10:53:37	2018-12-20 16:35:50
a1 9b	f55b6e8	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	e23ff44	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	e185b74	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	eb07e49	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	eb0e165	dddd	网关	未激活	离线		2018-12-20 10:53:37	
a1 9b	f7bcc00	dddd	网关	未激活	离线		2018-12-20 10:53:37	
	总共 119 页				上一页 当前第1页	下一页		到第 1 🗘 页 确:
使用第三方工具(比如	11Putty/WinSCP/Xshell	等)访问 1270.0.1:61657	以访问远程设备提供的服	3条.				
使用第三方工具(比如	0Putty/WinSCP/Xshell	等)访问 127.0.0.1:61657	以访问远程设备提供的服	孫				
使用第三方工具(比如	Putty/WinSCP/Xshell	等)访问 127.0.0.1:61657	以访问远程设备提供的服	銹。				
1果需要文件上传/下背	载,可以使用 scp 命令,	详细用法请参考: https:/		scp				
;端地址: frontend-io	tx-remote-debug.aliyur							

常见问题

如果设备的运维通道状态是离线,需要怎么处理?

答:如果设备的运维通道状态是离线,将无法远程访问到该设备上,您可以参考远程访问设备端源码内容, 将RemoteTerminalDaemon移植到该设备上,并设置为开机自启动。

12.运维工具

12.1. 设备模拟器

物联网边缘计算(Link IoT Edge)提供了场景联动、函数计算、流数据分析等多种服务,帮助用户构建物联 网应用程序。考虑到用户没有大量可连接的实际设备来测试使用应用程序,物联网边缘计算同时提供一套设 备模拟器(DeviceSimulator)的解决方案。

设备模拟器包含两部分:

- 驱动:在边缘端可以模拟所有类型的设备。设备的属性、事件、服务取决于物模型的定义,物模型相关信息,请参见物模型。
- 控制工具:改变设备模拟器的属性值、触发设备模拟器进行事件上报。

您只需要将设备和设备模拟器的驱动绑定,并部署至边缘端,结合控制工具,便可以脱离真实的物理设备进 行应用程序的测试。本文将会对设备模拟器的使用进行详细介绍。

前提条件

• 下载设备模拟器驱动。

产品规格	边缘版本号	驱动下载地址
专业版、标准版	v1.8.2~最新版本	DeviceSimulator_v0.2

设备模拟器只适用于Link IoT Edge标准版和专业版v1.8.2及以上版本环境中使用,因此开始操作本文内容前,请确保您已完成边缘实例的创建。具体操作,请参见专业版环境搭建或标准版环境搭建。

使用设备模拟器驱动

- 1. 上传驱动。
 - i. 登录边缘计算控制台。
 - ii. 在左侧导航栏单击驱动管理。
 - iii. 在驱动管理页面自研驱动页签下,单击新建驱动。
 - iv. 在**新建驱动**对话框设置驱动参数,上传<mark>前提条件</mark>中下载到本地的设备模拟器驱动文件。具体操作, 请参见云端发布。

本示例中的参数设置如下图所示,其余参数,请参见云端发布内容设置,也可以不设置。

驱动信息			
* 驱动名称 🕐			
DeviceSimulator			
*通信协议类型 📀			
自定义			\sim
* 语言类型:			
Node.js 8			\sim
驱动是否内置 ₍₂₎]	
○ 문 💿 좀			
* 驱动文件 👔			
重新上传			
DeviceSimulator_v0.2.z	ip (4.75KB)		х
* 驱动版本		1	
1.0.0			
* 驱动适配的边缘版本	9		
2.4.1 版本及以上			\sim
驱动版本描述:			
请输入驱动版本描述			

v. 完成参数的设置并上传成功驱动文件后,单击确认。您可以在驱动列表中看到刚刚创建的驱动。

2. 分配驱动到边缘实例。

i. 左侧导航栏单击边缘实例,在前提条件中创建好的边缘实例右侧单击查看。

ii. 在实例详情页面设备与驱动页签下,单击全部驱动右侧的 + 图标,在弹出面板中选择自研驱动,找到步骤1创建的DeviceSimulator驱动,单击对应操作列下的分配,将该驱动分配到边缘实例。

边缘计算 / 边缘实例 / 实例详情	分配驱动	×
← LinkloTEdge_Node	自研驱动 ✓ 済船入驱动名称 Q	C
产品規約 支业成 CPU 使用率 - 查看	驱动名称 通信协议类型 CPU架构 版本	操作
网关 监控信息 设备与驱动 场景联动 边缘应用 流数据分	DeviceSimulator 自定义 - 1.0.0 ~	分配
✓ 全部驱动 + 百元服助分配	4-FF	

iii. (可选) 若您的边缘实例产品规格为专业版, 则需要进行容器配置。

选择已分配的DeviceSimulator驱动,单击**设备列**表右侧的容器配置,设置如下图参数后,单击保存。

← 容器配置			
是否使用宿主机host模式 〇 是 • 否			
网络端口映射 👩			
宿主机端口 💿	容器内端口 💿	类型	操作
8999	9000	TCP	~ 移除
设备映射 ◎ + 新增设备映射			
卷映射 ◎ + 新燈卷映射			
保存 取消			

网络端口映射参数说明如下所示,其他说明,请参见示例驱动。

参数说明

参数	描述
宿主机端口	设置一个宿主机当前没有在使用的端口号,将容器内端口映射到此端口上,例如 设置为8999端口。
容器内端口	容器内端口固定位为9000。
协议类型	此处选择TCP。

- 3. 添加子设备到边缘实例。
 - i. 在**全部驱动**下,单击DeviceSimulator驱动,然后在**设备列表**区域单击**分配子设备**, 在DeviceSimulator驱动下,为边缘实例分配子设备。

ii. 在分配子设备面板中,单击添加子设备。

边缘计算 / 边缘实例 / 实例详情			分配子设备	×
← LinkloTEdge_N 产品原格 专业版 CPU使用率 - 畫者	Node ^{邮票成功}	服务版本 内存使用2	Q Q <	全部 未分配 操作
网关 监控信息 设备与驱	动 场景联动 边缘应用 流数据分析 消息路由	日志服		
✓ 全部驱动 + DeviceSimulator	设备列表 驱动配置 容器配置 分配子设备 了解设备接入?	设备名称	新建子设备可分配 新建子设备	
	□ 设备名称 所置产品	格式配置		
		i未添加任 「Edge 可强 分配子设备		
			分配 关闭 有新	设备? 添加子设备

iii. 在**添加设备**对话框,单击**新建产品**,创建产品。

添加设备	×
针别说明: 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	
10.000	~
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者 务操作。如需配置此产品功能请:前往配置	服
DeviceName	
请输入DeviceName	
确定取消	٤

iv. 在**创建产品**对话框中设置参数后,单击完成。

详细的参数说明,请参见<mark>创建产品</mark>。参数说明

参数	描述
产品名称	设置产品名称。
所属品类	选择标准品类下的 边缘计算 > 其它设备 。
接入网关协议	选择自定义。

v. 在**添加设备**对话框,单击前往配置为产品添加物模型。

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	
温度传感器	~
新建产品 产品中可以增加自定义的物模型配 <u>置,以便</u> 实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能请:前往配置	€
DeviceName	
请输入DeviceName	
确定取消	

为产品添加temperature属性,详细的参数说明,请参见单个添加物模型中的"添加自定义功能"步骤。

添加自定义功能		\times
* 功能类型 ② 属性 服务 事件		
* 功能名称 💿		
温度		
* 标识符 🕜		
temperature		
* 数据类型		
int32 (整数型)		\sim
* 取值范围	_	
0	~ 100	
* 步长		
1		
单位		
摄氏度 / ℃		\sim
* 读写类型		
●读写 ○只读		

■ 为产品添加high_temperature事件,详细的参数说明,请参见单个添加物模型中的"添加自定义功能"步骤。

添加自定义功能	×
* 功能类型 @ 属性 服务 事件	
* 功能名称 ②	
高温报警	
* 标识符 📀	
high_temperature	
* 事件类型 💿	
○ 信息 ● 告警 ○ 故障	
输出参数 +增加参数	
描述	
请输入描述	
	0/100
* 扩展描述 💿	
+新增扩展描述	

为事件增加参数。

新增参数		×
* 参数名称 💿		
温度		
* 标识符 ②		
temperature		
* 数据 类 型		
int32 (整数型)		\sim
* 取值范围		
50	~ 100	
* 步长		
1		
单位		
摄氏度 / ℃		\sim
扩展描述		
* 参数索引: 💿		
0		

vi. 返回添加设备对话框,输入DeviceName的值后单击确定,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	
温度传感器	
新建产品 产品中可以增加自定义的物模型配置,以便实现业务逻辑中的属性、事件或者服 务操作。如需配置此产品功能清:前往配置	
DeviceName	
temperatureSensor01	
确定取消	

- vii. 在**分配子设备**对话框中,选择**温度传感器**产品下的temperatureSensor01设备,单击该设备右侧的分配,将该设备分配到实例中。
- 4. 部署边缘实例。

在**实例详情**页面右上角单击**部署**,部署边缘实例。此时会弹出**部署进程**的对话框,您可以单击对话框 中的查看日志查看具体部署情况。部署成功后边缘实例名称后显示**部署成功**。

使用设备模拟器控制工具

设备模拟器控制工具的功能和真实物理设备的按键类似。完成本文上方使用设备模拟器驱动的操作后,可使 用设备模拟器控制工具。

- 1. 登录您的远程终端。登录操作详情,请参见远程服务访问。
- 2. (可选)若您使用Link IoT Edge专业版软件,则需要在远程终端执行docker ps | grep 驱动名,查看 是否有驱动容器。

如果有驱动容器,则需要执行如下命令,修改设备模拟器工具配置。否则无需执行此步骤。

echo 宿主机IP:宿主机端口 > /linkedge/run/ds_ctrl.conf

⑦ 说明 宿主机IP为您边缘网关的宿主机IP;宿主机端口为您在本文上方使用设备模拟器驱动中的步骤2进行容器配置时设置的端口(本示例中为8999)。

3. 使用设备模拟器控制工具控制设备行为。

进入/linkedge/gateway/build/bin 目录,执行 ./ds_ctrl 命令即可看到工具的使用方式。

Usage: ds_ctrl <command>
where <command> is one of:
 property, event
ds_ctrl property productKey deviceName params
Change property
ds_ctrl event productKey deviceName eventCode params
For example(productKey = xxx, deviceName = yyy):
 ds_ctrl property xxx yyy '{"temperature":30}'
 ds_ctrl event xxx yyy alarm '{"temperature":90}'

可改变设备属性:

a. 改变温度。

使用工具将温度传感器温度改为30。

./ds_ctrl property alWuxHr**** temperatureSensor01 '{"temperature":30}'

系统返回如下信息,表示温度设置成功。

Send property success!

- b. 登录物联网平台控制台, 在实例概览页面, 单击公共实例。
- c. 在左侧导航栏选择设备管理 > 设备,在相应设备名称右侧单击查看。

在**设备详情**页面,选择物模型数据 > 运行状态页签,查看设备的运行状态。

物联网平台 / i	设备管理 / 设备	备 / 设备详情	
← tem	peratui	reSensor	r01 在线
产品	温度传感器	查看	
ProductKey	10000	复制	
设备信息	Topic列表	物模型数据	设备影子
运行状态	事件管理	服务调用	
实时刷新 🌑			
温度	_	查看数据	ł
30 °c o			
2020/05/19	14:17:18.693		

d. (可选)模拟多次温度变化。

连续执行如下命令5次,每次将温度值+1。

./ds ctrl property alWuxHr**** temperatureSensor01 '{"temperature":31}'

可在控制台中,设备的物模型数据 > 运行状态页签,单击查看数据查看数据的变化。



- 可触发设备事件:
 - a. 上报高温事件。

```
./ds_ctrl event alWuxHr**** temperatureSensor01 high_temperature '{"temperature":
90}'
```

- b. 登录物联网平台控制台, 在实例概览页面, 单击公共实例。
- c. 在左侧导航栏选择设备管理 > 设备,在相应设备名称右侧单击查看。

在设备详情页面,选择物模型数据 > 事件管理页签,查看设备的事件。

物联网平台 /	设备管理 / 设	备 / 设备详情							
← tem	peratu	reSensoi	·01 在线						
产品	温度传感	器 查看				C	DeviceSecret	***** 查看	
ProductKey	1000	复制							
设备信息	Topic列表	物模型数据	设备影子	文件管理	日志服务	在线调试			
运行状态	事件管理	服务调用							
请输入事件标	识符	Q	全部类型	~	1小时	\sim			
时间		标识符		事件名称			事件类型		输出参数
2020/01/03 1	4:58:05	high_temperature		高温报警			alert		{"temperature":90}
	物联网平台 / ← tem 产品 ProductKey 设备信息 运行状态 请输入事件标 时间 2020/01/03 1	物联网平台 / 设备管理 / 设 ← temperatu 产品 温度传感 ProductKey 设备信息 Topic列表 运行状态 事件管理 博輸入事件标识符 时间 2020/01/03 14:58:05	 物联网平台 / 设备管理 / 设备 / 设备详情 ← temperatureSensou 产品 温度传感器 查看 ProductKey 复制 设备信息 Topic列表 物模型数据 运行状态 事件管理 服务调用 博输入事件标识符 Q 时间 标识符 2020/01/03 14:58:05 high_temperature 	 物联网平台 / 设备管理 / 设备 / 设备伴摘 	 物联网平台 / 设备管理 / 设备 / 设备详情 	物联网平台 / 设备管理 / 设备 / 设备详備 ← temperatureSensor01 在代 产品 温度传感器 查看 ProductKey 夏制 设备信息 Topic列表 物模型数据 设备影子 文件管理 日志服务 运行状态 事件管理 服务调用 博输入事件标识符 Q 全部类型 ∨ 11/b时 时间 标识符 事件名称 2020/01/03 14:58.05 high_temperature 高温报管	物联网平台 / 设备管理 / 设备 / 设备详情 ← temperatureSensor01 在线 产品 温度传感器 查看	物联网平台 / 设备管理 / 设备 / 设备详情 ★ temperatureSensor01 在线 ★ 温度传感器 查看 DeviceSecret ProductKey 复制 设备信息 Topic列表 物模型数据 设备影子 文件管理 日志服务 在线调试 运行状态 事件管理 服务调用 通输入事件标识符 Q 全部类型 ◇ 1小时 ◇ 时间 标识符 事件名称 事件类型 2020/01/03 14:58:05 high_temperature 高温振管 alert	物联网平台 / 设备管理 / 设备 / 设备详備

在线调试设备

 在物联网平台控制台公共实例页面左侧导航栏,选择监控运维 > 在线调试,在在线调试页面,选择调 试产品和调试设备。

在线调试						
请选择设备:	温度传感器	~	tempera	tureSens	~	
调试真实	设备调试。	虚拟设备				
属性调试	服务调用					
调试功能:	请选择	~	方法:	请选择	~	
1						
发送指令	重置					

2. 获取设备属性。

单击**属性调试**,选择模块,然后单击获取,即可在输入框中看到当前温度值为35。

在线调试				
请选择设备:	温度传感器 〜 temperatureSens 〜			
 在线调 	试只支持调试真实设备,虚拟设备调试请使用 设备模拟器	×		
属性调试	服务调用 远程登录			
模块: 默议	模块 イ			
温度(tempe	erature) 🕜			
35		调试 🖌		
获取	设置 设置期望值 重置			
L	1			

3. 设置设备属性。

将温度设为40,单击设置后可在设备运行状态查看温度传感器温度值已变更为40。

4. 其他自定义服务, 会将相关信息打印到驱动日志中, 并返回如下信息。

{"code":0,"message":"success"}

12.2. 设备批量处理工具

物联网边缘计算提供多个跨平台的命令行工具(即批量处理工具),支持您批量处理设备和设备相关操作。

Modbus设备批量导入工具

您只需依照已获取工具中的Excel模板,填写设备信息,Modbus设备批量导入工具即可帮助您自动完成产品 和设备的配置,将多个Modbus协议的产品或设备自动接入网关,并通过官方Modbus驱动将设备属性上传到 云端。

批量设备操作工具

支持您批量创建产品、批量创建或删除设备、批量授权或取消授权设备、批量创建指定物模型TSL(Thing Specification Language)的产品及设备等操作。

批量导入或者导出设备功能定义工具

支持您使用Excel批量导入或者导出产品的物模型,适用于为一个产品设置多个设备属性、事件、服务,或者为多个产品设置设备属性、事件、服务的场景。

工具获取方式

请按如下模板发送信息,联系我们获取设备批量操作工具以及详细的使用说明。

```
主题:获取物联网边缘计算产品<设备批量工具名称>操作说明文档
公司名称:
公司所属行业信息:
产品和设备信息:(简要描述贵公司产品和设备信息,以获取更多定制化服务)
联系人:
联系电话:
工具名称及使用需求描述:
```

12.3. 管理运维工具lectl

lectl管理运维工具(以下统称lectl)是物联网边缘计算提供的管理Link IoT Edge及其资源的命令行工具,可帮助您更好的使用Link IoT Edge。

您可以在{LINKEDGE_ROOT}/gateway/build/bin目录下找到lectl工具。

(?) 说明 {LINKEDGE_ROOT)指Link IoT Edge软件包根目录所在的路径,通常为/linkedge/。

获取帮助

在 lectl 及子命令后添加--help命令,可获取帮助信息。顶层的帮助信息如下所示。

```
$ lectl --help
Usage:
       lectl [OPTIONS] COMMAND
Manage Link IoT Edge services and facilitate troubleshooting.
Options:
 -v, --version Print version information and quit
Commands:
 completion Output shell completion code
 config Manage configs
 console
           Manage console backend
 dbus
            Manage D-Bus message bus
 deployment Manage deployments
 device Manage devices
 diagnose Check if requirements meet
 driver Manage drivers
           Manage function compute
 fc
 logger
           Manage log service and logs
           Manage MQTT service
 mqtt
 ota
            Manage OTA service
 rom
           Manage remote maintenance
 router
           Manage message routing
           Manage stream compute
 SC
 scene
            Manage scenes
Run 'lectl COMMAND --help' for more information on a command.
```

自动补全

lectl支持bash和zsh的自动补全功能,提升输入效率。

bash

通过在网关上执行**lectl completion bash**命令,生成lectl的bash自动补全脚本,并将脚本导入到Shell 中,即可启动lectl的bash自动补全功能。

但由于bash自动补全脚本依赖bash-completion,且各种包管理系统提供的bash-completion不同,因此 您需要按照如下步骤启动lectl的bash自动补全功能。

- i. 在网关上执行type_init_completion命令,检查系统是否安装了bash自动补全脚本依赖的bash-completion。
 - 已安装:请跳转到步骤iv。
 - 未安装: 请跳转到步骤ii。
- ii. 通过系统自带的包管理系统,安装bash-completion。
 - RedHat系列操作系统: yum inst all bash-completion
 - Debian系列操作系统: apt-get install bash-completion

关于更多包管理系统提供的bash-completion相关信息,请参见开源库。

iii. 将安装bash-completion后生成的/usr/share/bash-completion/bash_completion文件添加到.bashr c文件中。

source /usr/share/bash-completion/bash_completion

iv. 启用自动补全功能。

有如下两种启用方法:

■ 直接导入bash自动补全脚本到当前的Shell。(推荐使用)

source <(lectl completion bash)</pre>

 ⑦ 说明 使用此方法后,在退出Shell时自动补全功能失效。若需要自动补全功能一直生效, 请执行 echo 'source <(lectl completion bash)' >> ~/.bashrc ,将此命令添加到.bashr c文件中。

■ 将bash自动补全脚本添加到 bash_completion.d目录。

lectl completion bash > /etc/bash_completion.d/lectl

• zsh

通过**lectl completion zsh**命令,生成lectl的zsh自动补全脚本,并将生成的脚本导入到Shell中,即可启用zsh自动补全功能。

在网关上执行如下命令,一步实现生成zsh自动补全脚本和启用自动补全功能。

source <(lectl completion zsh)</pre>

⑦ 说明 如果出现 complete:13: command not found: compdef 错误,请将如下命令复制粘贴到 .zshrc文件里。

autoload -Uz compinit compinit

连接诊断

当Link IoT Edge网关无法连接到云端时,您可以通过lectl工具,使用diagnose命令做初步诊断。

lectl diagnose

系统显示类似如下运行信息。

字段说明

字段	描述
Get Gateway Triple	获取网关设备的设备证书。
Get IoT Region	获取地域配置。
Check Network	检查网络连接状态。
Check DNS Service	检查DNS服务状态。
Check MQTT Port	检查MQTT协议端口。
Check MQTT Cert	检查MQTT协议证书。
Check MQTT Connection	检查MQTT连接。

管理日志

您可以使用lectl工具,打包日志或临时改变日志服务行为,方便在出问题时上传日志或临时排查。

• 打包日志

执行如下命令打包日志。

lectl logger pack

⑦ 说明 日志文件被默认打包到当前目录并命名为logs.zip。您也可以通过-n或--name选项指定 输出文件名,-o或--output选项指定输出路径。

● 管理日志服务

lectl允许获取日志服务配置,并临时改变日志服务行为。例如,Link IoT Edge出现问题时可以调整全局日 志等级到debug,以显示更多详细信息。

获取日志服务配置和改变日志服务行为主要通过如下命令实现。

lectl logger config [OPTIONS]

⑦ 说明 lectl logger config命令支持一次配置多个日志选项。

[OPTIONS] 选项说明

选项名称	描述
-d,disable	关闭日志系统、日志等级、模块日志、模块日志等级、输出目标、显示格式 等。不能一次关闭多项内容。
-e,enable	开启日志系统、日志等级、模块日志、模块日志等级、输出目标、显示格式 等。不能一次开启多项内容。
show	显示所有配置或指定配置信息。此选项只能以show= 形式指定。 指定配置项包括: o dir:日志文件路径 o level:全局日志等级 o module:模块日志配置 o format:日志显示格式 o target:日志输出目标 o size:日志文件大小
-g,single-file-mb	设置单个日志文件大小(单位为MB)。
-t,total-files-mb	设置总体日志文件大小(单位为MB)。

常见的lectl logger config命令示例如下。

```
$ lectl logger config --show
$ lectl logger config --show=level
$ lectl logger config -g 10
$ lectl logger config -t 1024
$ lectl logger config -e system
$ lectl logger config -e warn
$ lectl logger config -e credential
$ lectl logger config -e credential:info
$ lectl logger config -e stdout
$ lectl logger config -e stdout
$ lectl logger config -e full
$ lectl logger config -e full
```

管理配置

配置管理主要通过config子命令实现。

● 获取配置

使用如下命令获取配置。

lectl config get [OPTIONS] KEY

[OPTIONS]用于提供常见复杂用法的快捷方式,通常拥有独立的参数。

[OPTIONS] 选项说明

选项名称	描述
-d,d3	获取所有设备的设备证书信息,无独立参数。
-v,driver	获取给定驱动的配置,要求1个参数,表示驱动名。
-g,g3	获取网关设备的设备证书信息,无独立参数。
-s,tsl	获取给定产品的TSL(物模型),要求1个参数,表示产品的ProductKey。

常见的获取配置命令示例如下。

\$ lectl	config	get	foc)
\$ lectl	config	get	-g	
\$ lectl	config	get	-d	
\$ lectl	config	get	-s	foo
\$ lectl	config	get	-v	foo

● 设置配置

使用如下命令设置设备配置。

lectl config set [OPTIONS] KEY VALUE

[OPTIONS]用于提供常见复杂用法的快捷方式,通常拥有独立的参数。

[OPTIONS] 选项说明

选项名称	描述
-d,d3	设置设备证书信息,要求3个参数,分别表示ProductKey、DeviceName和 DeviceSecret。
-p,d3f	从给定的文件读取并设置设备证书信息,要求1个参数,表示文件路径。
-v,driver	设置驱动配置,要求2个参数,分别表示驱动名和驱动配置。
-f,file	从给定的文件读取并设置键、值,要求2个参数,分别表示键和文件路径。
-g,g3	设置网关的设备证书信息,要求3个参数,分别表示ProductKey、 DeviceName和DeviceSecret。
-s,tsl	设置产品的TSL,要求2个参数,分别表示产品的ProductKey和TSL文件路 径。

常见的设备配置命令示例如下。
```
$ lectl config set foo bar
$ lectl config set -f foo /path/to/bar
$ lectl config set -g foo bar baz
$ lectl config set -d foo bar baz
$ lectl config set -p /path/to/foo
$ lectl config set -s foo /path/to/bar
$ lectl config set -v foo /path/to/bar
```

• 移除配置

使用如下命令移除配置。

lectl config unset KEY

• 强制写入

为确保配置操作执行成功,您可以强制配置,并保存数据到磁盘。命令如下。

lectl config flush

管理D-BUS

Link IoT Edge内部组件使用D-Bus通信,可以通过**dbus**子命令来管理Link IoT Edge内D-Bus消息总线。 通过如下命令可以监听D-Bus总线上的消息。

lectl dbus monitor

管理部署

您可以通过deployment子命令,管理边缘实例的部署服务。

可通过lectl deployment show命令,显示边缘实例的部署信息。

管理驱动

您可以通过driver子命令,管理Link IoT Edge的驱动。

通过如下命令,查看驱动详细信息。

⑦ 说明 此命令的返回信息依赖被操作的驱动,如果驱动没有对外显示接口,则无法显示相关信息。

lectl driver show [OPTIONS] DRIVER

参数说明

选项名称	描述
[OPTIONS]	详细说明请参见本表格下方"[OPTIONS]选项说明"表。
DRIVER	被操作驱动的ID。

[OPTIONS] 选项说明

选项名称	描述
-c,connections	只显示驱动连接信息。默认显示驱动所有信息。
devices	只显示设备信息。默认显示驱动所有信息。您可以指定设备的ProductKey和 DeviceName过滤设备。

常见的管理驱动命令示例如下。

```
$ lect1 driver show 44c0****78204641a4b9bcf1024176c2
$ lect1 driver show -c 44c0****78204641a4b9bcf1024176c2
$ lect1 driver show --devices 44c0****78204641a4b9bcf1024176c2
$ lect1 driver show --devices=alt8***n42K/GjCb****gcKYaluGJLN 44c0****78204641a4b9bcf10241
76c2
```

管理设备

您可以通过device子命令来管理Link IoT Edge的设备相关信息。

• 显示信息

当前设备的状态信息,可通过lectl device show子命令查看。

一个显示信息片段如下。

lectl device show
{
"DeviceStats": [
{
"ProductKey": "alt9****42K",
"DeviceName": "GjCb****gcKXeluGJLNp",
"LocalId": "GjCb9****cKXeluGJLNp",
"IsLocal": false,
"DriverId": "e0bb****964e4519b4a52f9b719e1dd3",
"Activation": "activated",
"LocalState": "online",
"CloudState": "online",
"LastSeenTime": "2020-02-27T14:16:52+08:00",
"LastRequestOnlineTime": "2020-02-27T14:16:52+08:00",
"LastLocalOnlineTime": "2020-02-27T14:16:52+08:00",
"LastCloudOnlineTime": "2020-02-27T14:16:52+08:00",
"LastCloudOnlineFinishTime": "2020-02-27T14:16:52+08:00",
"LastRequestOfflineTime": "2020-02-26T12:04:45+08:00",
"LastLocalOfflineTime": "2020-02-26T12:04:45+08:00",
"LastCloudOfflineTime": "2020-02-26T12:04:45+08:00",
"LastCloudOfflineFinishTime": "2020-02-26T12:04:45+08:00",
"RequestOnlineTimes": 17,
"LocalOnlineTimes": 17,
"CloudOnlineTimes": 252,
"RequestOfflineTimes": 1,
"LocalOfflineTimes": 1,
"CloudOfflineTimes": 1
}
]
}

字段说明

选项名称	描述
ProductKey	产品唯一标识。
DeviceName	设备名称。
DriverId	关联该设备的驱动ID。
Activation	设备状态是否为已激活。
LocalState	设备与网关的连接状态。
CloudState	设备与云端的连接状态。

● 获取设备属性

可通过get properties子命令,获取指定设备的属性值。

lectl device getproperties [OPTIONS]

[OPTIONS] 选项说明

选项名称	是否必选	描述
-i,identifier	是	用于标识要获取属性的设备。格式为 ProductKey/De viceName 。
key	是	用于标识要获取的属性的键。格式为JSON字符串数组。

常用的获取设备属性命令示例如下。

lectl device getproperties --identifier=alt8****42K/GjCb****gcKYaluGJLNp --keys='["foo"]'

● 设置设备属性

可通过set properties子命令,设置指定设备的属性值。

lectl device setproperties [OPTIONS]

[OPTIONS] 选项说明

选项名称	是否必选	描述
-i,identifier	是	用于标识要设置属性的设备。格式为 ProductKey/De viceName 。
properties	是	用于标识要设置的属性的键。格式为JSON字符串对象。

常用的设置设备属性命令示例如下。

```
$ lectl device setproperties --identifier=alt8****42K/GjCb****gcKYaluGJLNp --properties='
["foo": "bar"]'
```

• 调用设备服务

可通过callservices子命令,调用指定设备上的服务。

lectl device callservice [OPTIONS]

[OPTIONS] 选项说明

选项名称	是否必选	描述
-i,identifier	是	用于标识被调服务所在的设备。格式为 ProductKey/ DeviceName 。
name	是	用于标识被调服务的名称。
args	否	用于标识被调服务的传入参数,格式由被调服务定义。

常用的调用设备服务命令示例如下。

```
$ lectl device callservice --identifier=alt8****42K/GjCb****gcKYaluGJLNp --name=foo
$ lectl device callservice --identifier=alt8****42K/GjCb****gcKYaluGJLNp --name=foo --arg
s=bar
$ lectl device callservice --identifier=alt8****42K/GjCb****gcKYaluGJLNp --name=foo --arg
s='["bar"]'
```

管理控制台

您可以通过console子命令来管理Link IoT Edge提供的边缘网关控制台,包括设置证书和设置登录密码等。

● 设置证书

您可以通过如下命令设置边缘网关控制台证书:

lectl console certificate <SUBJECT>

其中, <SUBJECT>格式为 /type0=value0/type1=value1... 。

⑦ 说明 type 取值有C (Country)、ST (State)、L (Localcity)、O (Organization)、
 OU (Organization Unit)、CN (Common Name)。更多信息,请参见OpenSSL文档。

默认证书的*<SUBJECT>*为 /C=CN/ST=Zhejiang/L=Hangzhou/O=Link IoT Edge/OU=Link IoT Edge/CN=Link IoT Edge 。

● 设置密码

您可以通过如下命令设置或更新边缘网关控制台用户密码:

```
lectl console password [OPTIONS]
```

[OPTIONS] 选项说明

选项名称	是否必选	描述
-u,username	是	要更新密码的用户名。
-p,password	是	新的密码。

管理场景联动

您可以通过lectl来管理Link IoT Edge上运行的场景联动,包括场景信息的显示、启用、停用、触发场景联动等。

• 显示信息

您可以通过lectl scene show命令查看场景的基本信息、执行信息、统计信息、部署信息等。 执行命令后的结果示例如下。

```
lectl scene show
{
   "Scenes": [
       {
            "Times": 8,
            "FailureTimes": 0,
            "LastState": {
               "State": "success",
               "Cause": "",
               "Time": "2020-02-27T14:25:00+08:00",
               "Duration": 3
            },
            "Id": "d141****2aaf4d1d8e3259637e489528",
            "Name": "scene_test_types",
            "Running": true,
            "Time": "2020-02-27T14:17:53+08:00",
            "Uptime": 466184,
            "Actions": {
               "Times": 8,
               "FailureTimes": 0,
                "LastState": {
                    "State": "success",
                   "Cause": "",
                   "Time": "2020-02-27T14:25:00+08:00",
                   "Duration": 2
               }
            },
            "Conditions": {
               "Times": 8,
               "FailureTimes": 0,
                "LastState": {
                   "State": "success",
                   "Cause": "",
                   "Time": "2020-02-27T14:25:00+08:00",
                   "Duration": 0
                }
            }
       }
   ],
   "Deployment": null
}
```

● 启动场景联动

执行如下命令,启动一个或多个场景联动。

lectl scene enable SCENE [SCENE...]

其中, SCENE为场景联动的ID, 多个场景联动ID之间以空格间隔。

● 停止场景联动

执行如下命令,停止一个或多个场景联动。

lectl scene disable SCENE [SCENE...]

其中, SCENE为场景联动的ID, 多个场景联动ID之间以空格间隔。

• 触发场景联动

执行如下命令,触发一个或多个场景联动。

lectl scene invoke SCENE [SCENE...]

其中, SCENE为场景联动的ID, 多个场景联动ID之间以空格间隔。

• 重置场景联动

执行如下命令,移除当前网关上的所有场景联动。

lectl scene reset

管理函数计算

可以通过lectl管理Link IoT Edge上的函数计算。包括显示信息、部署函数、移除函数、重置部署、调用函数、重启函数等。

• 显示信息

当函数运行异常时,可通过lectlfcshow命令查看函数的运行状态及统计信息。

执行命令后的结果如下。

```
$ lectl fc show
{
    "Functions": [
        {
            "FunctionId": "90fe****d2a5466993117e58c1ae3f7f",
            "ServiceName": "helloworld",
            "FunctionName": "helloworld",
            "Runtime": "nodejs8",
            "MemorySize": 1073741824,
            "Timeout": 5,
            "RunMode": "OnDemand",
            "Status": "Running",
            "InvocationTimes": 157,
            "ProcessStats": {
                "Pid": 254,
                "MemoryUsage": 32988,
                "StartTime": "2020-02-27 14:17:14.412906 ",
                "LastStopTime": "1970-01-01 08:00:00.000000 ",
                "LastInvokeTime": "2020-02-27 14:22:26.547315 ",
                "ExitTimes": 0
            },
            "TaskStats": {
                "HandledTimes": 157,
                "HandledErrorTimes": 0,
                "UnhandledTimes": 0,
                "HandledAverageDuration": 3025,
                "WaitingAverageDuration": 0,
                "LastHandledDuration": 2006
            },
            "RuntimeStats": {
                "RequestUnhandledTimes": 0,
                "RequestHandledTimes": 0,
                "RequestHandledErrorTimes": 0,
                "RequestHandledAverageDuration": 0
            },
            "DeploymentStats": {
                "State": "Success"
            }
       }
   ]
}
```

• 调用函数

使用如下命令调用一个函数。

lectl fc invoke [OPTIONS] FUNCTION

参数说明

选项名称

描述

选项名称	描述
[OPTIONS]	OPTIONS选项,包含如下选项。 <i>-p,payload</i> :Base64编码后的参数内容。该参数内容将被原封不动的传 给函数。
FUNCTION	函数ID或者函数ARN(资源名称, Aliyun Resource Name)。函数ARN格式 为 acs:fc: <region>:<accountid>:service:<servicename>:func tion:<functionname> 。</functionname></servicename></accountid></region>

• 重启函数

使用如下命令重启一个或多个函数。

lectl fc restart FUNCTION [FUNCTION...]

其中,FUNCTION为函数ID,多个函数ID以空格间隔。

• 移除函数

使用如下命令移除一个或多个函数。

lectl fc remove FUNCTION [FUNCTION...]

其中,FUNCTION为函数ID,多个函数ID以空格间隔。

● 重置部署

使用如下命令重置部署。

lectl fc reset

管理流数据分析

您可以在Link IoT Edge专业版(LE Pro)中使用lect L工具管理流数据分析。

• 显示信息

可通过lectl sc show命令查看当前系统流数据任务状态。

[root@lib.id_lib bin]#./lectl sc show
======================================
01
ID: sc4996c 5f75abc7fcf30b
Name: MySQL1
Running: true
SQLPath: /linkedge/run/stream-compute/sql/sc4996c 5f75abc7fcf30b.sql
SQLMd5: 237284d90a7146927a28ff666143e994
LogPath: /linkedge/run/logger/stream-compute/sc4996c
BlinkJobId: d7b020005f29057e5debcbefd2028781
CheckpointDir: /linkedge/run/stream-compute/blink_storage/checkpoint/sc4996c 5f75abc7fcf30t
字段说明

选项名称	描述
ID	流数据任务ID。

选项名称	描述
Name	流数据任务名称。
Running	流数据任务是否正在运行。
SQLPath	SQL语句文件路径。
SQLMd5	SQL语句文件MD5值。
LogPath	流数据任务日志路径。
BlinkJobId	流数据任务在Blink中对应的任务ID。

● 启动流数据任务

使用如下命令启动一个或多个流数据任务。

lectl sc start STREAM [STREAM...]

其中, STREAM为流数据任务ID, 多个流数据任务ID以空格间隔。

● 停止流数据任务

使用如下命令停止一个或多个流数据任务。

lectl sc stop STREAM [STREAM...]

其中, STREAM为流数据任务ID, 多个流数据任务ID以空格间隔。

● 验证SQL

将SQL语句保存到指定文件,并执行如下命令验证流数据任务的SQL语句是否正确。

lectl sc checksql FILE

管理远程服务

您可以通过rom子命令,管理Link loT Edge的远程服务相关信息。

执行如下命令,可以显示远程服务当前状态。

lectl rom show

字段说明

选项名称	描述
ChannelState	远程服务的通道连接状态。
ThreadState	远程服务线程状态。
Services	远程服务列表及其状态。

管理MQTT

您可以通过mqtt子命令,管理Link IoT Edge组件的MQTT消息订阅。

执行如下命令,可以显示Link loT Edge组件的消息订阅。

lectl mqtt show

字段说明

选项名称	描述
Торіс	消息订阅的主题。
QoS	服务质量,对应MQTT协议QoS。
Subscriber	消息的订阅者。
State	消息的订阅状态。
Time	消息的订阅时间。

13.边缘网关控制台

13.1. 概述

边缘网关控制台是Link IoT Edge提供的边缘端Web服务,为物联网用户提供了网关、设备、驱动的管理能力,同时也为设备物模型、流数据分析、消息路由、日志、账户等资源提供管理能力。

适用场景

边缘网关控制台适用于局域网环境,低延时需求的项目。例如某工厂需要实时关注生产线设备的状态,如果 从云端获取数据,则要求具有公网环境,且因为网络问题可能会有数据延迟。此时可通过边缘网关控制台, 实时查看设备数据和状态,也可以调试设备。

使用流程

1. 根据实际情况,选择Link IoT Edge产品规格,搭建Link IoT Edge运行环境。详细操作,请参见环境搭建。

↓ 注意 边缘网关控制台文档内容,仅适用于v2.4.7及以上版本的Link IoT Edge,因此选择产品规格时,边缘版本请选择v2.4.7及以上版本。

2. 设置证书、密码,登录边缘网关控制台。详细操作,请参见登录边缘网关控制台。

3. 根据需求查看或管理网关、驱动、设备、物模型等。

13.2. 登录边缘网关控制台

首次使用边缘网关控制台,需要设置服务端证书和控制台登录密码,同时需要访问符合规范格式的URL来登录边缘网关控制台。

< ○ 注意

- v2.4.7及以上版本的Link IoT Edge: 按照本文内容, 登录边缘网关控制台。
- 低于v2.4.7版本的Link IoT Edge: 请直接访问 https://<Your_Gateway_IP>:9999 , 输入初始用 户名 admin 、密码 admin1234 , 登录边缘网关控制台。

步骤一:设置证书

第一次使用边缘网关控制台时,需要设置服务端证书。您可以通过Link IoT Edge提供的,位 于/linkedge/gateway/build/bin目录的管理运维工具lectl来设置服务端证书。

lectl console certificate <SUBJECT>

其中, <SUBJECT>格式为 /type0=value0/type1=value1... 。

⑦ 说明 type 取值有C (Country)、ST (State)、L (Localcity)、O (Organization)、
 OU (Organization Unit)、CN (Common Name)。更多信息,请参见OpenSSL文档。

默认证书的*<SUBJECT>*为 /C=CN/ST=Zhejiang/L=Hangzhou/O=Link IoT Edge/OU=Link IoT Edge/CN=Link IoT Edge 。

步骤二:设置密码

第一次使用边缘网关控制台时,需要配置默认用户admin的密码。您可以通过lectl工具来设置用户密码。

lectl console password -u admin -p <Your_Password>

⑦ 说明 密码需要由英文字母和数字组成,以英文字母开头,长度为5~20个字符。

步骤三:登录控制台

1. 通过浏览器访问如下格式的URL来登录边缘网关控制台。

https://<Your_Gateway_IP>:9999

其中, <Your_Gateway_IP>为您边缘端网关的地址。

⑦ 说明 支持Chrome浏览器、Safari浏览器、IE 8及以上版本的浏览器。

2. 以使用Chrome浏览器访问边缘网关控制台为例,首次访问URL时,需要信任服务端证书。

因为<mark>步骤一:设置证书</mark>中设置的服务端证书,属于自签发证书,所以浏览器会提示服务端证书签发机构 无效。请单击**高级**,然后单击**继续前往**,信任该服务端证书。



您的连接不是私密连接

攻击者可能会试图从 **localhost** 窃取您的信息(例如:密码、通讯内容或信用卡信息)。 <u>了解详情</u>

NET::ERR_CERT_AUTHORITY_INVALID

□ 将<u>您访问的部分网页的网址、有限的系统信息以及部分网页内容</u>发送给 Google,以帮助我们提升 Chrome 的安全性。<u>隐私权政策</u>

高级

返回安全连接

3. 进入边缘网关控制台登录页面, 输入用户名和密码, 登录边缘网关控制台。

参数	说明
初始用户名	初始用户名为 admin 。

参数	说明
用户密码	步骤二:设置密码中设置的 admin 用户密码。

常见问题

忘记边缘网关控制台的登录密码时怎么办?

答:请参考本文上方步骤二:设置密码中的操作,重新设置边缘网关控制台的登录密码。

13.3. 网关管理

通过网关管理,可查看网关基本信息和网关的CPU、内存、磁盘等多种监控项,也可以重置网关。

查看网关信息

在边缘网关控制台左侧导航栏单击网关管理,进入网关管理页面后,显示网关信息。

你可以看到网关设备的连接状态、设备证书信息、上线时间等。也可以看到网关的CPU数量、UPC使用率、 内存使用率等多种监控项和运行信息。

初始化网关

若您的Link IoT Edge尚未配置网关设备证书(Product Key、DeviceName、DeviceSecret),则需要初始化网关。

1. 登录边缘网关控制台后,系统跳转到网关初始化页面,单击新网关初始化。

网关管理		
	新网关初始化	
	点此进行新的网关设置	

2. 根据系统提示,输入网关设备的ProductKey、DeviceName、DeviceSecret。

3. 单击设置, 使设置的参数生效。

设置网关成功后,根据系统提示,重新登录边缘网关控制台。

重置网关

↓ 注意 重置网关将会清除网关的所有配置,包括网关设备证书、边缘网关控制台用户信息、网关部署信息等,请谨慎操作。

1. 在边缘网关控制台左侧导航栏,单击网关管理,单击重置网关。

- 2. 根据系统提示,为网关设备配置新的ProductKey、DeviceName、DeviceSecret。
- 3. 单击设置, 使设置的参数生效。

重置网关后,根据系统提示,重新登录边缘网关控制台。

13.4. 驱动管理

驱动管理展示了关联到网关的所有驱动及其状态,同时也支持在本地调试驱动。

配置驱动

- 1. 在边缘网关控制台左侧导航栏单击驱动管理。
- 2. 在目标驱动右侧单击配置。

驱动管理			
驱动列表			
请输入驱动名称 搜索			
驱动名称	驱动语言	状态	操作
demo_driver	C	• 运行中	配置 查看子设备 调试运行
LightSensor	Node is 8	● 运行中	配置 查看子设备 调试运行

3. 根据系统提示配置驱动,然后单击保存。

驱动配置页面有默认配置和自定义配置两种显示方式:

- 如果您有自定义驱动的配置页面,则会显示您已定义的配置页面。
- 如果您没有自定义驱动的配置页面,则显示默认的驱动配置页面,默认驱动配置页面参数说明如下: 驱动配置参数说明

参数	描述
配置格式	配置格式有如下三种: ■ 键值对配置 ■ JSON格式
配置格式为键值对配置 时	单击 新增配置 ,通过设置配置名称、值、注释来配置驱动。 最多添加100个键值对。
配置格式为JSON格式时	编写JSON格式配置内容,内容不超过1 KB。编写内容时,系统自动校验格式。若 格式错误,可根据系统提示编写正确格式的内容。

调试驱动

↓ 注意 只有驱动配置模式为键值对的驱动,允许开启调试功能。

- 1. 在边缘网关控制台左侧导航栏单击驱动管理。
- 2. 在目标驱动右侧单击调试运行。
- 3. 在调试运行页面,打开调试开关。
- 4. (可选)在驱动调试页面,更新驱动配置或新增配置,驱动配置完成后单击保存。

驱动配置相关参数说明,请参见本文上方驱动配置参数说明。

驱动管理 > 调试运行			
LED			
是否开启调试: 已开启 🔵			
配置格式: 键值对配置	~ 0		
键值对配置表: + 新增配置			
保存 调试数据展示			自动刷新
Кеу	Value	时间	操作
brightness	50	2020-05-08 10:19:42	读取 写入
power	10	2020-05-08 10:19:42	读取 写入
voltage	25	2020-05-08 10:19:42	读取 写入

- 5. 在驱动已有的键值对右侧单击读取或写入。
 - 读取:读取当前驱动的键值对信息。
 - 写入: 更改当前驱动键值对的配置。

查看子设备

在**驱动管理**页面,单击某个驱动右侧的**查看子设备**,可查看该驱动下关联的所有子设备信息。子设备管理 详情请参见<mark>设备管理</mark>。

13.5. 设备管理

通过设备管理,可查看当前网关下的所有子设备信息和运行状态。

查看设备信息

在边缘网关控制台左侧导航栏单击设备管理,进入端设备管理页面后,显示网关子设备信息。

概览信息说明

数据	描述
设备总数	网关下的子设备总数。
本地在线数	通过驱动与网关连接正常的子设备总数。

设备列表页签下显示所有网关子设备。单击子设备右侧的查看,可查看该子设备的详细信息、功能和运行 状态。

调试设备功能

⑦ 说明 只有读写权限的子设备属性或服务,才支持调试功能。

1. 在边缘网关控制台左侧导航栏单击设备管理。

- 2. 在目标子设备名称右侧单击查看。
- 3. 在设备详情页面,单击设备能力页签,找到需要调试的属性或服务,单击调试。
- 4. 根据系统提示,设置调试参数,然后单击调试,查看调试结果。

⑦ 说明 设备调试参数,会根据您实际设置的设备属性或服务参数显示。设备属性或服务设置, 以及参数说明请参见单个添加物模型。

端设备管理 > 设备详情						
tflite_detector 本地连接 ● 衣 设备调	试			×		
产品名称: detector					uctKey: 复制	
驱动名称:	设备能力:	object				
设备标签: -	* 属性值:	"41"				
设 <mark>备船力</mark> 运行状态	运行结果:	2020-05-15 10:48:21 INFO: 雇性调试成功, 属性值: 41				
1 功能模板						
功能类型					操作	
属性					查看 派	试
事件					查看	
属性					查看 派	试
			调试 取消			

13.6. TSL管理

通过TSL管理,您可以管理TSL(物模型)相关资源,包括查看TSL、导入TSL和删除TSL等。

查看TSL信息

在边缘网关控制台左侧导航栏单击TSL管理,可在TSL管理页面显示网关子设备的TSL(物模型)信息。 在每个TSL右侧,单击相应操作按钮,可管理TSL:

- 查看: 查看该TSL的标准物模型和扩展物模型。
- 删除: 删除子设备下的该物模型。

导入TSL

为网关子设备所属产品,导入TSL步骤如下所示。

- 1. 在边缘网关控制台左侧导航栏单击TSL管理。
- 在TSL管理页面右上角单击导入TSL文件。
 根系统提示,上传TSL文件。

↓ 注意 TSL文件中的ProductKey信息,必须与物联网平台中控制台上的产品ProductKey一致。
 TSL文件编写,请参见什么是物模型。

TSL管理			
TSL列表			导入TSL文件
任务编号	productKey	产品名称	操作
1	导入TSL文件	×	查看 删除
2		1	查看 删除
3	请选择TSL文件	parts -	查看 删除
4	请选择扩展TSL文件		查看 删除
5	产品名称		查看 删除
	确定	取消	

参数说明

参数	描述
请选择TSL文件	此处上传JSON格式的标准物模型文件。
请选择扩展TSL文件	此处上传JSON格式的扩展物模型文件。
产品名称	输入一个产品名称。您可以输入与物联网平台控制台上产品名称一致的名称,也可以设置产品别名,方便您后续识别该TSL文件属于哪个产品。

3. 单击确定,保存设置。

13.7. OTA管理

OTA(Over-the-Air Technology)即空中下载技术。通过边缘网关控制台中的OTA管理,您可以升级子设备固件,监控并查看升级状态。

查看设备信息

在边缘网关控制台左侧导航栏单击OTA管理,进入OTA升级管理页面后,显示可升级的子设备信息。

概览信息说明

数据	描述
设备总数	网关下的子设备总数。
本地在线数	通过驱动与网关连接正常的子设备总数。

升级固件

- 1. 在边缘网关控制台左侧导航栏单击设备管理。
- 2. 在OTA升级管理页面,单击新增升级。
- 3. 在新增升级对话框,单击上传文件,上传升级包文件。

OTA 升级管理				
概览				
设备总数 📀	• 本地在线数 🥐			
5	新增升级	×		
升级状态监控	* 升级包文件: 上传文件			
请输入设备名称 搜索 全部	* 升级设备: 请选择升级设备	~		新增升级
升级时间↓↓ 设备名称		4	前版本	升级状态
		确定 取消		

参数说明

参数	描述
升级包文件	单击 上传文件 ,上传固件文件。
升级设备	选择待升级的设备。 单击下拉框,系统弹出 升级设备 页面

4. 在**新增升级**对话框升级设备参数右侧单击下拉框,系统弹出**升级设备**页面。

从下拉框中选择子设备关联的驱动,然后选中目标:	子设备前的复选框,	单击确定。
-------------------------	-----------	-------

升级设备				\times
LED	~			
	设备名称	所属产品	创建时间	
	TASN_DEVICE_Goods	a1sFeOaQgoe	2020-05-08 09:43:01	
已选择 0 ·	个设备		确定取消	Ĭ

5. 在新增升级对话框中单击确定。

您可在**升级状态监控**区域框中查看升级进度。

13.8. 场景联动

通过场景联动页面,可以查看网关上运行的所有场景联动规则,也可以对场景联动规则进行启用、停用、触 发操作。

查看场景联动规则

在边缘网关控制台左侧导航栏单击**场景联动**,可在**场景联动**页面显示已关联到网关的所有场景联动规则。

操作
启用
停用 触发
停用 触发

在每个规则名称右侧,单击相应操作按钮,可管理该场景联动规则:

• 启动:启动该场景联动规则。运行状态为未启动的规则,会显示该操作按钮。

- 停止:停止该场景联动规则。运行状态为运行中的规则,会显示该操作按钮。
- 触发:无视场景联动规则触发条件,手动触发一次规则。运行状态为运行中的规则,会显示该操作按钮。

13.9. 日志管理

通过日志管理,您可以查看Link loT Edge所有服务或进程的日志,也可以更改日志配置。

查看日志信息

在边缘网关控制台左侧导航栏单击日志管理,可在日志管理页面查看网关日志信息。

在**日志目录**下方,单击要查看的服务或进程名称,左侧区域框显示该服务或进程的日志文件。单击日志文件 右侧的**查看日志**,可查看详细日志。

日志管理				
日志详情				日志配置
日志目录				
- /	文件名	文件大小	文件日期	操作
DebuggerProxy	log_2020-05-19-09-39-59-1.txt	541.52KB	2020-05-22 18:04:44	查看日志
binary-ota	log_2020-05-14-15-56-00-1.txt	316.06KB	2020-05-18 23:40:57	查看日志
cloud-proxy + config-manager	log_2020-05-14-09-49-46-1.txt	56.71KB	2020-05-14 15:47:47	查看日志
console	log_2020-05-12-09-47-38-1.txt	238.71KB	2020-05-13 18:14:25	查看日志
credential	log_2020-05-11-18-18-34-1.txt	572Byte	2020-05-11 19:40:45	查看日志
device-ota	log_2020-05-09-16-08-41-1.txt	895Byte	2020-05-09 16:12:40	查看日志
dimu	log_2020-04-28-20-15-05-1.txt	279.79KB	2020-04-30 22:07:03	查看日志
+ tc-base gw-cascade	log_2020-04-28-20-01-22-1.txt	1.82KB	2020-04-28 20:11:22	查看日志
high-availability				
. €				

配置日志

- 1. 在边缘网关控制台左侧导航栏单击日志管理。
- 2. 在日志管理页面右上角单击日志配置。
- 3. 根据系统提示,设置参数。

日志管理	!								
日志详情									日志配置
日志目易	R.								
	<u>^</u>		日志配置			×			
- /	Data	文件名				文件	+日期	操作	
	DebuggerProxy	log_2020	* 日志级别:	INFO 🗸		202	20-05-22 18:04:44	查看日志	
	binary-ota	log_2020	* 日志空间:	1000	MB V	202	20-05-18 23:40:57	查看日志	
+	cloud-proxy config-manager	log_2020				202	20-05-14 15:47:47	查看日志	
	console	log_2020	日志路径:	/linkedge/run/logger /logger		202	20-05-13 18:14:25	查看日志	
	credential	log_2020			_	202	20-05-11 19:40:45	查看日志	
	default device-ota	log_2020			确认取消	202	20-05-09 16:12:40	查看日志	

参数说明

参数	描述
日志级别	选择需要更改存储空间或路径的日志类型。 • DEBUG:在服务或进程的开发、调试过程中打印的运行信息。 • INFO:服务或进程的运行过程信息。 • WARN:服务或进程运行时,具有告警提示的信息。 • ERROR:服务或进程运行时,发生错误的信息。 • FATAL:服务或进程运行时,发生严重错误,导致服务或进程退出的信息。
日志空间	设置日志的存储空间。最大可设置为20 GB。
日志路径	配置存储日志的路径。您可以使用系统默认的路径,也可以自定义设置路径。

4. 单击确认,保存配置。

13.10. 账户管理

边缘网关控制台,支持为多个用户角色,分配不同权限来管理网关。您可以通过账户管理查看、新增、删除 和编辑用户信息。

查看用户信息

在边缘网关控制台左侧导航栏单击**账户管理**,可在**账户管理**页面显示各个用户信息。

通过用户名右侧操作列中的按钮,可管理该用户。

操作按钮	说明
编辑	修改该账户密码和权限。参数说明如下: • 用户名:该账户对应的用户名,该参数不可更改。 • 密码:为用户设置新密码。支持英文字母和数字,长度为5~20个字符。 • 密码确认:确认新密码。 • 账户权限:选中允许该用户操作的功能模块。
删除	删除该用户。删除用户后,不可再通过该用户名登录边缘网关控制台,请谨慎操作。

添加用户

- 1. 在边缘网关控制台左侧导航栏单击账户管理。
- 2. 在账户管理页面右上角单击添加。
- 3. 根据系统提示,设置参数。

账户管理					
账户列表					
请输入用户名,字母或数字,5-20位 搜索	新增账户		×		添加
账户名	* 用户名:	调输入用户名		最近登录时间	操作
admin	* 密码:	请输入密码		2020-05-15 11:17:57	编辑
super	* 密码确认:	两次输入密码保持一致		2020-05-25 17:24:44	编辑删除
	★账户权限:	 ○ 风关管理 ○ 成功管理 → (株)の管理 ○ TA管理 ○ KA管理 ○ KAE ○ KAE			
		除存	取消		

参数说明

参数	描述
用户名	设置新的用户名。支持英文字母和数字,长度为5~20个字符。
密码	为用户设置密码。支持英文字母和数字,长度为5~20个字符。
密码确认	确认新用户密码。
账户权限	选中允许该用户操作的功能模块。

4. 单击**保存**,保存配置。

14.边缘安全服务

本章主要介绍Link IoT Edge中的安全服务功能。详细介绍边缘安全服务主要功能、在Link IoT Edge中使用边 缘安全服务的方法以及边缘安全服务的基本结构。

概述

边缘安全服务,即Link IoT Edge在边缘端集成了设备保护服务(Device Protection Service),简称DPS服务。该服务将边缘连接到物联网安全运营中心Link SOC(Security Operations Center),帮助管理员识别和 消除边缘系统潜在的安全风险,保障边缘系统运行过程中的安全。更多Link SOC信息请参见Link SOC文档。

名词解释

名词	说明
安全基线	即系统安全范围,超出此范围则被认定为安全性异常。其数据主要来自于取证和行为规 则。
取证	对边缘系统的数据完整性进行检测,用于发现数据的异常变化。主要用于检测"系统对象异常"情况。
行为规则	对边缘系统的程序行为进行检测,用于发现程序的异常行为。主要用于检测"进程异常行为"和"网络异常行为"。
基线发布	将安全基线同步到边缘端,作为边缘端基本参照,用于发现异常。
安全策略	边缘端的安全基线,用于判断系统行为和取证信息是否异常。
组件漏洞	边缘端第三方组件(可执行文件、动态库等)的安全漏洞。

功能介绍

边缘安全服务主要功能如下所示。

- 安全基线制定: 自动构建边缘系统的安全基线。
- 评估安全等级: 根据系统多维度信息评估当前设备的安全等级。
- 安全风险识别: 识别和阻断安全基线之外的异常行为, 异常行为包括如下。
 - 系统对象异常:系统中异常的应用加载执行或对象的未知改动,例如篡改可执行文件。
 - 进程异常行为: 应用中异常的执行行为, 例如访问重要文件。
 - 网络异常行为: 设备异常的网络通信行为, 例如对未知目标发送数据。
- 漏洞修复: 修复存在的组件漏洞, 防止威胁入侵。

安全服务架构图如下所示。





- 边缘安全服务将采集的取证和行为规则信息上报云端,形成安全基线。
- 用户安全基线发布,则将云端安全基线同步到所有适配的边缘端产品中形成安全策略。
- 边缘端通过运行时(Runtime)搜集的取证和行为信息,会根据安全策略判断异常事件,并上报到Link SOC。
- 组件漏洞在边缘端检查可执行文件和依赖库的版本漏洞并上报到云端,用户可以选择进行修复。云端下发补丁版本,边缘安全服务通过热升级进行版本替换。
- 用户设置通知策略, 异常事件和组件漏洞的上报消息会以邮件的方式通知到用户。
- 安全日志在边缘端搜集安全操作历史记录并上报到云端,方便用户查看。

操作步骤

- 启用边缘安全服务
 - i. 参考<mark>环境搭建</mark>内容,启动Link loT Edge并登录后台。可以通过<mark>远程连接</mark>功能登录后台。
 - ii. 执行以下命令启动边缘安全服务。

/linkedge/gateway/build/bin/lectl config set gw_dps_switch 1

iii. 执行以下命令判断是否启动成功。

/linkedge/gateway/build/bin/dps/script/statusDPS.sh

系统显示如下信息,表示启动成功。



- 停止边缘安全服务
 - i. 执行以下命令停止边缘安全服务。

/linkedge/gateway/build/bin/lectl config set gw_dps_switch 0

ii. 执行以下命令判断是否停止成功。

/linkedge/gateway/build/bin/dps/script/statusDPS.sh

系统显示如下信息,表示停止成功。

15.日志服务

Link IoT Edge 日志服务是通过集成阿里云日志服务云产品来完成边缘端日志的管理功能,包括边缘日志自动 同步至云端存储、日志分级别显示等。

物联网边缘计算中的日志服务使用的是阿里云日志服务产品功能。阿里云日志服务在超过免费额度后会产生 一定的费用,在使用前请您仔细阅读该服务的计费方式。

前提条件

在开始操作本章内容前,请您确保已根据<mark>环境搭建</mark>内容完成了边缘实例的创建,并根据云资源访问内容为创建 的实例添加了服务角色。

操作步骤

- 1. 在边缘计算控制台左侧导航栏中单击边缘实例,找到要查看日志的实例,单击对应操作栏中的查看。
- (可选)若您第一次使用日志服务功能,则需要开通服务。若已开通服务,则跳过本步骤进行下一步操作。

← LinkloTEdge_Node 成功											Ĩ	医看部署日志	重置	部署	
产品规格	专业	扳					服务版本号	v2.4	4.0						
CPU 使用率	- 查看	5					内存使用率	- 査	语						
网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置							
]			日志服务 ^{产品,才可以使用}	边缘实例的日志3	力能。								

i. 在**实例详情**页面,选择日志服务,单击立即开通。

在系统跳出的页面中单击创建简单日志服务。

- ii. 在云产品开通页, 阅读《日志服务服务协议》并勾选我已阅读并同意, 单击**立即开通**。
- iii. 返回边缘计算控制台实例详情页面,选择设置页签,请参考云资源访问内容,为边缘实例添加管理
 日志服务(Log)的权限。
- iv. 在**实例详情**页面选择日志服务页签, 打开启用状态按钮。

网关	监控信息	设备与驱动	场景联动	边缘应用	流数据分析	消息路由	日志服务	设置			
日志服务											② 启用状态
日志存儲	は筆略 了第	旧志服务?	全部级别 X	∼ Fe	eb 5, 2020 16:02:32		Feb 12, 2020 16:0	2:32 🛱	请输入日志关键字 C	ł	
序号	日志类	型	级别		内容					时间	

3. 在实例详情页面日志服务页签下,单击日志存储策略,设置您的日志存储策略。

日志存储使用LOG(阿里云日志服务)产品能力,因此日志存储时间越长所需空间就越多,请按需设置存储时间。

用户指南·日志服务

网关	监控信息 设备与驱动	场景联动 边缘	 日志存储策略	×		
日志服务	ui 了解日志服务?	全部限别 🗙 💙	1 日志存儲需要便用SLS产品能力,日志存儲时间越长所需空间 请按需设置存储时间	侈,	送嫁字 Q	◎ 肩用状态
序号	日志美型	級別	* 日志保存时间 7天	\sim		BJIE
			* 日志級别 @	~		
			19k			
			利加速	取消		

参数说明

参数	描述
日志保存时间	网关日志被保存的时间长度。
日志级别	选择存储哪一个级别及以上的日志,日志级别优先级如下: 错误>警告>信息>调试 例如,选择信息,表示存储信息、警告、错误三个级别的日志。

4. 单击确定完成日志存储策略的设置。

当网关产生日志后,物联网边缘计算会根据您日志存储策略,过滤日志的内容并在**日志服务**页面中展示。

5. (可选)您还可以根据日志级别和时间范围来搜索查询日志服务中的日志,展示在日志服务页面中。

16.OTA升级

物联网平台提供OTA升级服务,您只需在控制台上传新的固件,即可将升级消息推送到设备端,完成设备在 线升级。

物联网边缘计算使用OTA升级时,有以下限制条件。

准备工作

您需要在下载地址中选择需要升级的Link IoT Edge标准版(LE Standard)软件包并下载到本地。
 例如,下载的是X86-64系统v2.0.0版本的软件包,包名为*link-iot-edge-x86-64-v2.0.0.tar.gz*。

⑦ 说明 仅支持v2.0.0及以上版本的Link IoT Edge标准版(LE Standard)软件包使用OTA升级服务。

2. 在下载好Link IoT Edge标准版软件包的路径下,执行如下命令生成 config.json文件。

本示例中,在X86-64系统环境中link-iot-edge-x86-64-v2.0.0.tar.gz文件所在路径下执行命令。

```
echo '{"ApiVersion":"1.0","Type":"LinkEdge"}' > config.json
tar -czf upgrade-package.tar.gz config.json link-iot-edge-x86-64-v2.0.0.tar.gzlink-iot-
edge-x86-64-v2.7.0.tar.gz
```

3. 将生成的*upgrade-package.tar.gz*文件上传到物联网平台控制台个人实例或公共实例的监控运维 > OTA升级。

设备端软件包要求

- 要求/linkedge/run/binary-ota/的剩余空间不少于Link IoT Edge自包含软件包大小及解压缩后的大小之和。各版本自包含软件包大小请见下载地址。
- 要求/linkedge/gateway/build/的剩余空间不少于自包含包解压缩后的大小。

控制台升级设置

• 新增固件时: 参数设置要求

参数	描述
升级包版本号	必须使用v2.0.0及以上版本的Link loT Edge,请在 <mark>下载地址</mark> 中获取Link loT Edge版本 号。
签名算法	仅支持MD5方式。

• 验证升级包、批量升级时: 参数设置要求

参数	描述
升级包类型	仅支持 整包 升级。

17.OTA升级SDK

物联网边缘计算提供OTA升级SDK,您可基于该SDK进行二次设备开发,以支持升级您运行Link loT Edge的 操作系统。

请按如下模板发送信息,联系我们获取OTA升级SDK的详细使用说明。

主题:获取物联网边缘计算产品OTA升级SDK和操作说明文档 公司名称: 公司所属行业信息: 使用OTA升级SDK的产品和设备信息:(请简单描述例如设备型号、CPU、通信协议等产品或设备信息) OTA升级需求说明:(请简单描述待升级设备的操作系统和目标操作系统、升级的设备数量、或其他OTA升级需求) 联系人: 联系电话:

18.产线烧录

18.1. 产线烧录概览

本文主要介绍物联网边缘计算产品的产线烧录流程,帮助您快速完成批量生产和产线激活。

背景信息

物联网平台使用ProductKey标识产品,DeviceName标识设备,设备证书(ProductKey、DeviceName和 DeviceSecret)校验设备合法性。设备通过协议接入物联网平台之前,需依据不同的认证方法,上报产品与 设备信息,认证通过后,方可接入物联网平台。针对不同的使用环境,物联网平台提供了多种认证方案。

物联网边缘计算服务的产品管理能力复用自物联网平台,支持两种设备认证方式。

- 一机一密: 逐台为每台设备烧录单独的设备证书。安全级别高。
- 一型一密:批量为同一产品类型的设备烧录相同的产品证书。批量烧录,操作方便。

详细信息请参见物联网平台设备安全认证文档。

在做产线烧录操作前,您需要引导程序

引导程序用于为边缘计算产品配置设备证书信息。一机一密和一型一密两种方案都需使用。

该程序必须开机启动。获取设备证书信息后,需做简单的合法性校验,例如是否为空,长度是否符合要求 (ProductKey长度11位, ProductSecret长度32位)等。

程序逻辑示例如下。



18.2. 一机一密产线烧录

一机一密认证方法,即预先为每个设备烧录其唯一的设备证书(Product Key、DeviceName和 DeviceSecret)。当设备与物联网平台建立连接时,物联网平台对其携带的设备证书信息进行认证。



操作步骤

- 1. 参考物联网平台创建产品和批量创建设备章节,批量创建边缘计算/边缘网关类型的网关设备,并记录设备证书信息(包含ProductKey, DeviceName和DeviceSecret)。
- 2. 在产线上将设备证书信息烧录至实际设备中。
 - i. 将引导程序烧录到设备的某一分区。
 - ii. 将设备证书信息(ProductKey、DeviceName和DeviceSecret)烧录到设备的其它分区,确保安全存储。
- 3. 为设备插上网线,开机后确保设备可以访问公网。

⑦ 说明 您可以在设备上执行以下命令,确认网络环境。

- \$ping iot-as-mqtt.cn-shanghai.aliyuncs.com : 确认外部网络状态。
- \$ping 127.0.0.1 : 确认内部网络状态。
- \$/linkedge/gateway/build/scripts/iot_gateway_status.sh
 :确认引导程序运行正常,物联网边缘计算服务开启。
- 4. 云端查看设备状态。若设备状态变为在线,表明烧录成功。

物联网平台 / 设备管理 / 设备 / 设备详情													
← gateway_01													
产品 LinkloTEdge_Gateway 查看 DeviceSecret ****** 查看													
ProductKey	ProductKey 复制												
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	子设备管理			
设备信息													
产品名称	产品合称 LinkloTEdge_Gateway ProductKey 复制 区域												
节点类型	市点地型 内关 DeviceName gateway_01 観制 认证方式 设备密钥												

18.3. 一型一密产线烧录

一型一密安全认证方式,即为同一产品下所有设备烧录相同固件(固件中写入产品证书,即ProductKey和 ProductSecret)。选用此种方式,您也需要预先在物联网平台上,开启动态注册开关,保存产品证书。

设备基于产品证书发送激活请求时,物联网平台根据产品证书进行认证。认证通过,物联网平台下发该设备 对应的DeviceSecret,设备便可以使用设备证书信息接入物联网平台。



操作步骤

- 1. 参考物联网平台创建产品和批量创建设备章节,批量创建边缘计算/边缘网关类型的网关设备,并记录产 品证书信息(包含ProductKey和ProductSecret)。
- 2. 手动开启产品动态注册按钮。

? 说明

- 开启动态注册需通过手机验证。
- · 若设备发出激活请求时,系统校验发现该开关未开启,将拒绝新设备的动态激活请求。已激
 活设备不受影响。
- 。 建议所有设备完成激活后,关闭动态注册开关。

物联网平台 /	物联网平台 / 设备管理 / 产品 / 产品详惯												
← LinkloTEdge_Gateway													
ProductKey 设备数	ProductKey 复刻 ProductSecret ******* 查音 设备数 4 前往管理												
产品信息	B Topic类列表 功能定义 服务端订阅 日志服务 在线询试												
产品信息	∠ 编辑												
产品名称	LinkloTEdge	_Gateway			节点类型	网关设备	创建时间	2019/11/04 10:24:06					
所属品类	自定义品类				数据格式	ICA 标论性数据格式 (Alink JSON)	认证方式	设备密明					
动态注册 📀	已开启 🧲	2			状态	 ● 开发中 	连网协议	WiFi					
产品描述													

- 3. 在产线上将产品证书信息(ProductKey和ProductSecret)烧录至实际设备中。
 - i. 将烧录工具和引导程序烧录到设备的某一分区。
 - ii. 将产品证书信息, 烧录到设备的其它分区, 确保安全存储。
 - iii. 运行烧录工具,通过产品证书信息获取设备证书信息。即云端下发DeviceSecret至设备端。

\$/linkedge/gateway/build/bin/tool_generate_3triple Your_ProductKey Your_DeviceName
Your_ProductSecret

iv. 将设备证书信息存储到烧录工具和引导程序以外的其它分区,确保安全存储。

4. 为设备插上网线,开机后确保设备可以访问公网。

⑦ 说明 您可以在设备上执行以下命令,确认网络环境。

- \$ping iot-as-mqtt.cn-shanghai.aliyuncs.com : 确认外部网络状态。
- \$ping 127.0.0.1 : 确认内部网络状态。
- \$/linkedge/gateway/build/scripts/iot_gateway_status.sh
 :确认引导程序运行正常,物联网边缘计算服务开启。
- 5. 云端查看设备状态。若设备状态变为在线,表明烧录成功。

物銀屑平台 / 设备管理 / 设备 / 设备详病												
← gateway_01												
产品 ProductKey	产品 LinkoTEdge_Gateway 直合 DeviceSecret ******* 直音 Productkey 复刻											
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	理	日志服务	在线调试	子设备管理		
设备信息												
产品名称	产品名称 LinkloTEdge_Gateway ProductKey 复制 区域											
节点类型	节点类型 网关 DeviceName gateway_01 复制 认证方式 设备密钥											

常见问题

一型一密方案获取设备证书信息失败怎么办?

答:应用程序获取设备证书成功后,须妥善保存至设备的安全存储区域,谨防丢失。

若获取设备证书失败,需要在物联网平台上,手动删除该设备后,方能重新获取。删除设备需经过手机信息 验证。