多功能网关soe事件修改方案

# 申电云设备事件上报格式如下

* 功能：设备按产品物模型中定义的格式将事件信息上报给平台。
* Topic：$v5/{productKey}/{deviceSN}/sys/event/up
* 方向：上行；
* 请求数据格式（示例）：

{

"msgid": "123456",

"eventid": "event.{tsl.event.identifier}"

"params": {

"value": {

"Temperature": "high",

"Battery": "low"

},

"ts": 1524448722000

}

}

* 请求参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 是否必选 | 类型 | 说明 |
| msgid | 必选 | String | 消息ID号。String类型的数字，取值范围0~4294967295。方便平台对消息的跟踪 |
| eventid | 必选 | String | 物模型中定义的事件标识（event.标识符） |
| params | 必选 | Object | 上报事件中的输出参数 |
| value | 必选 | Object | 物模型事件中的输出参数字段信息 |
| ts | 可选 | Long | 可选字段，属性上报时间戳，类型为UTC毫秒时间，不带时由物联网平台自动生成，建议设备携带防止平台时序问题 |

# 阿里云：事件案例

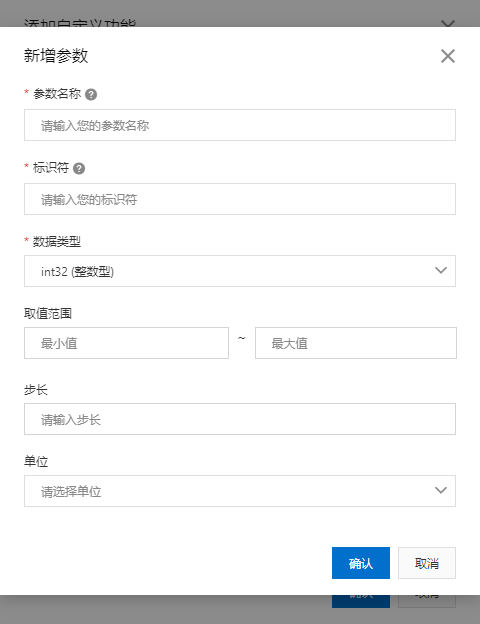
阿里云有专属事件页面

## 创建事件示例：

创建事件页面：



新增输出参数页面：



示例如下：

阿里云自带示例：



## 阿里云事件上传格式：

* 功能：设备按产品物模型中定义的格式将事件信息上报给平台。
* Topic：/sys/{productKey}/${deviceName}/thing/event/${tsl.event.identifier}/post
* （{productKey}表示产品key, {deviceName}表示设备标识符，{tsl.event.identifier}表示事件标识符）
* 方向：上行；
* 请求数据格式（示例）：

{

"id": 3536123,

"version": "1.0",

"params": {

"入参key1": "入参value1",

"入参key2": "入参value2"

},

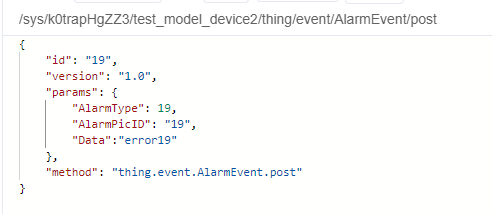
"method": "thing.event.{tsl.event.identifier}.post"

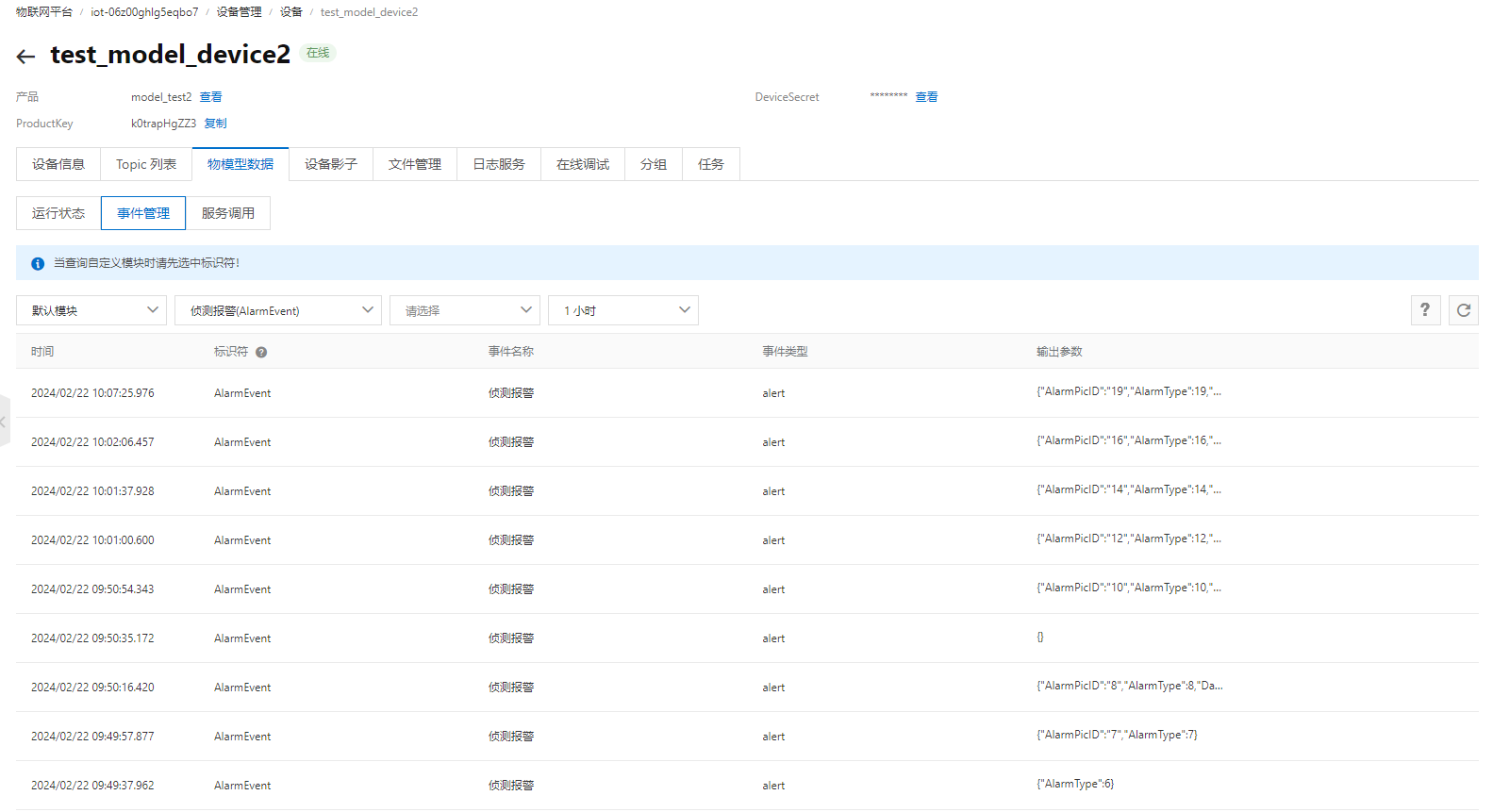
}

* 请求参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 是否必选 | 类型 | 说明 |
| id | 必选 | String | 消息ID号。String类型的数字，取值范围0~4294967295。方便平台对消息的跟踪 |
| version | 可选 | String | 版本号 |
| params | 必选 | Object | 上报事件中的输出参数 |
| 入参key1 | 可选 | Object | 物模型事件中的输出参数字段信息 |
| method | 可选 | String | 事件消息标识符（thing.event.{tsl.event.identifier}.post） |

真实示例：





其中：

标识符：对应{tsl.event.identifier}。

事件名称：对应创建事件时的功能名称。

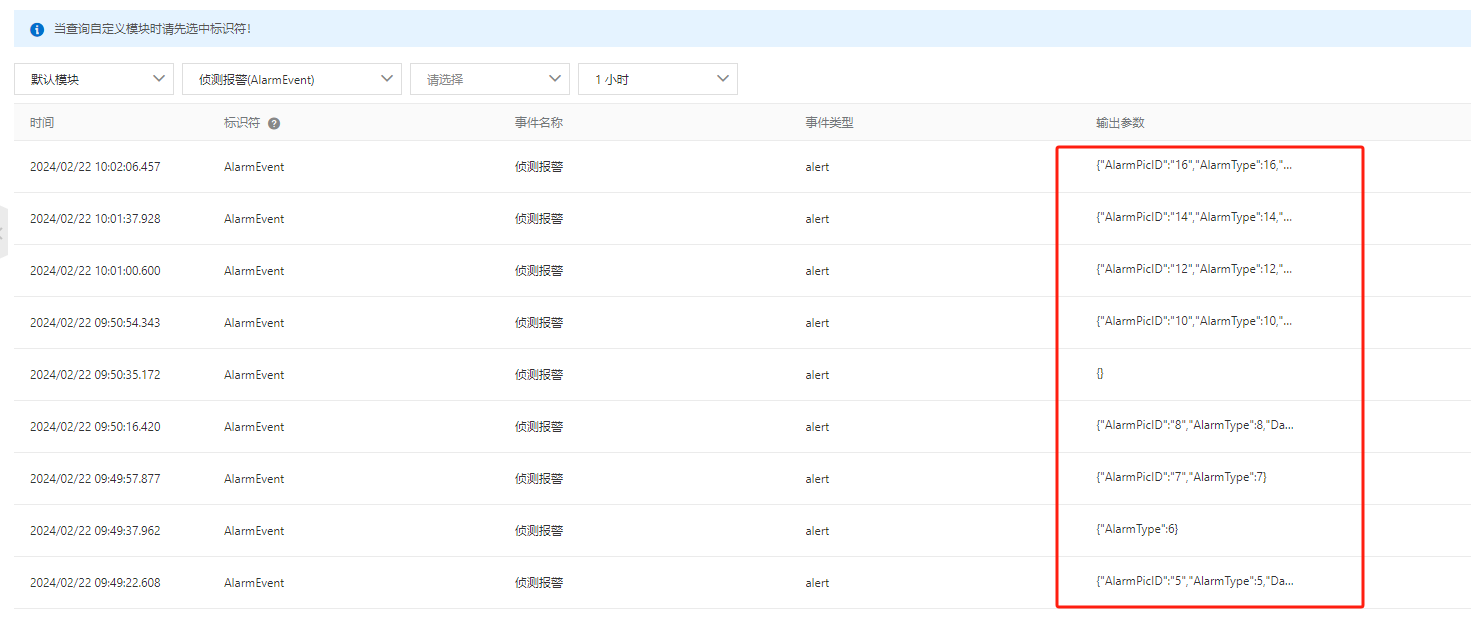
事件类型：对应创建事件时的事件类型。

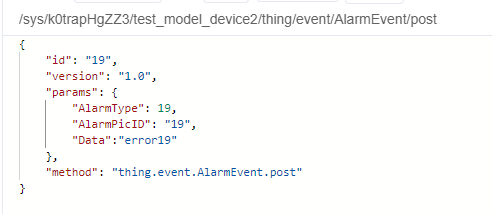
输出参数：对应设备上报事件中的输出参数信息。

## 阿里云事件测试

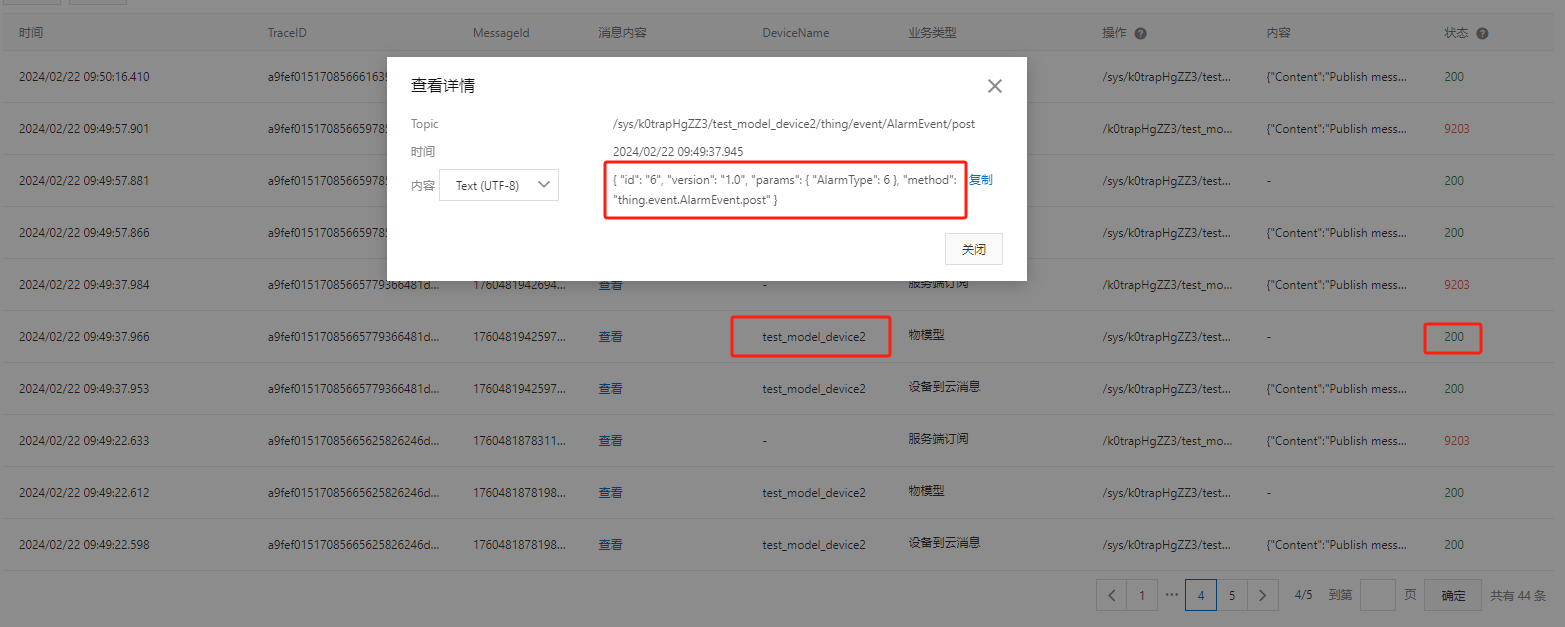
阿里云测试，输出参数可以少写和不写，这种情况会记录在数据库中；但如果输出参数错误，则没有写入；

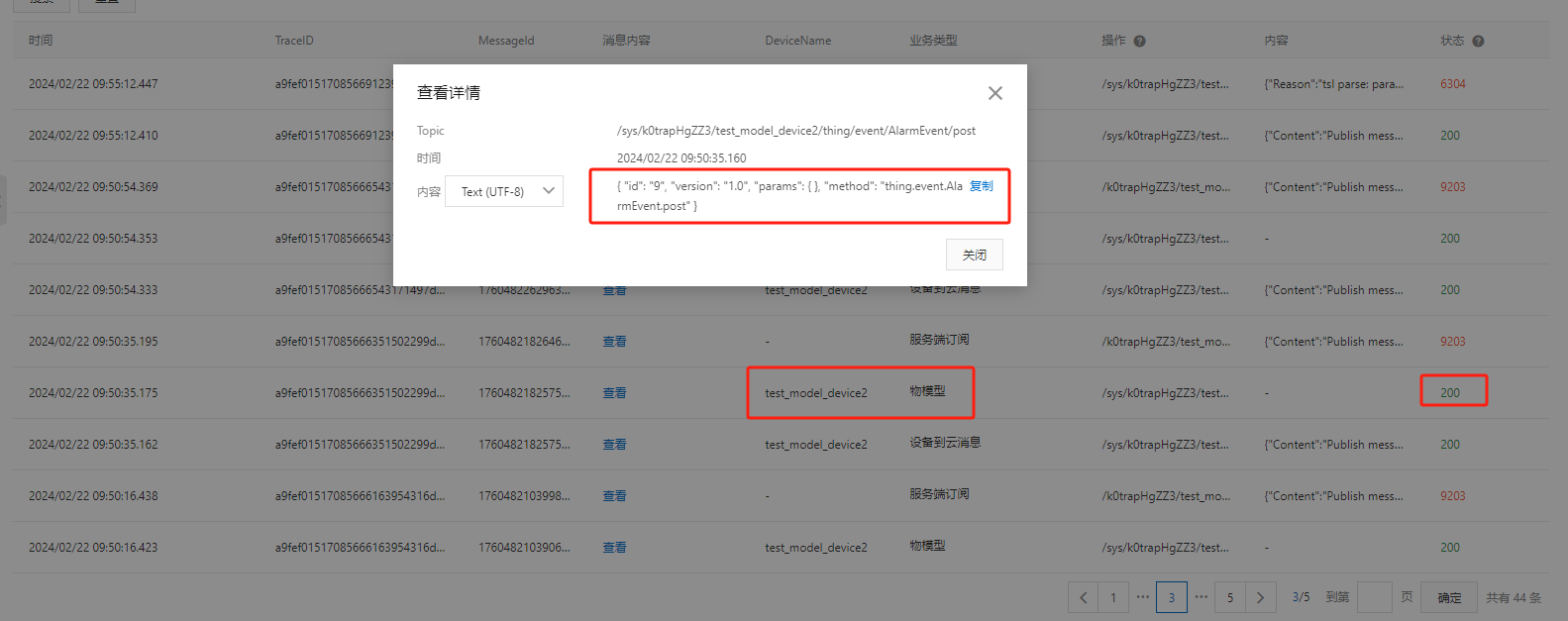
发布的事件如下，id是属性增加，这个事件有三个参数；在输出参数标识符正确情况下，缺少输出参数或者不写输出参数，阿里云依然正确记录，日志显示正确；在输出参数标识符有一个错误的情况下没有记录，日志显示错误；

MQTTX事件发布示例：

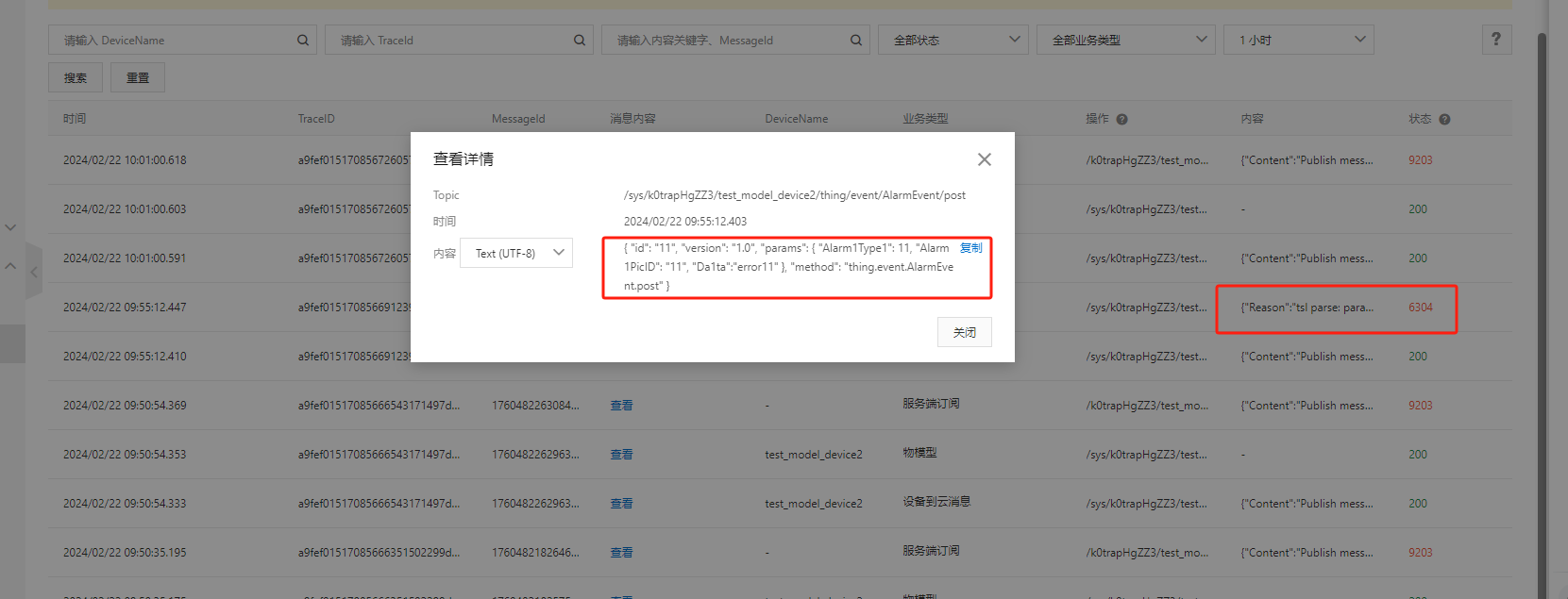


输出参数缺失、空，日志正确：





输出参数不正确，日志错误：

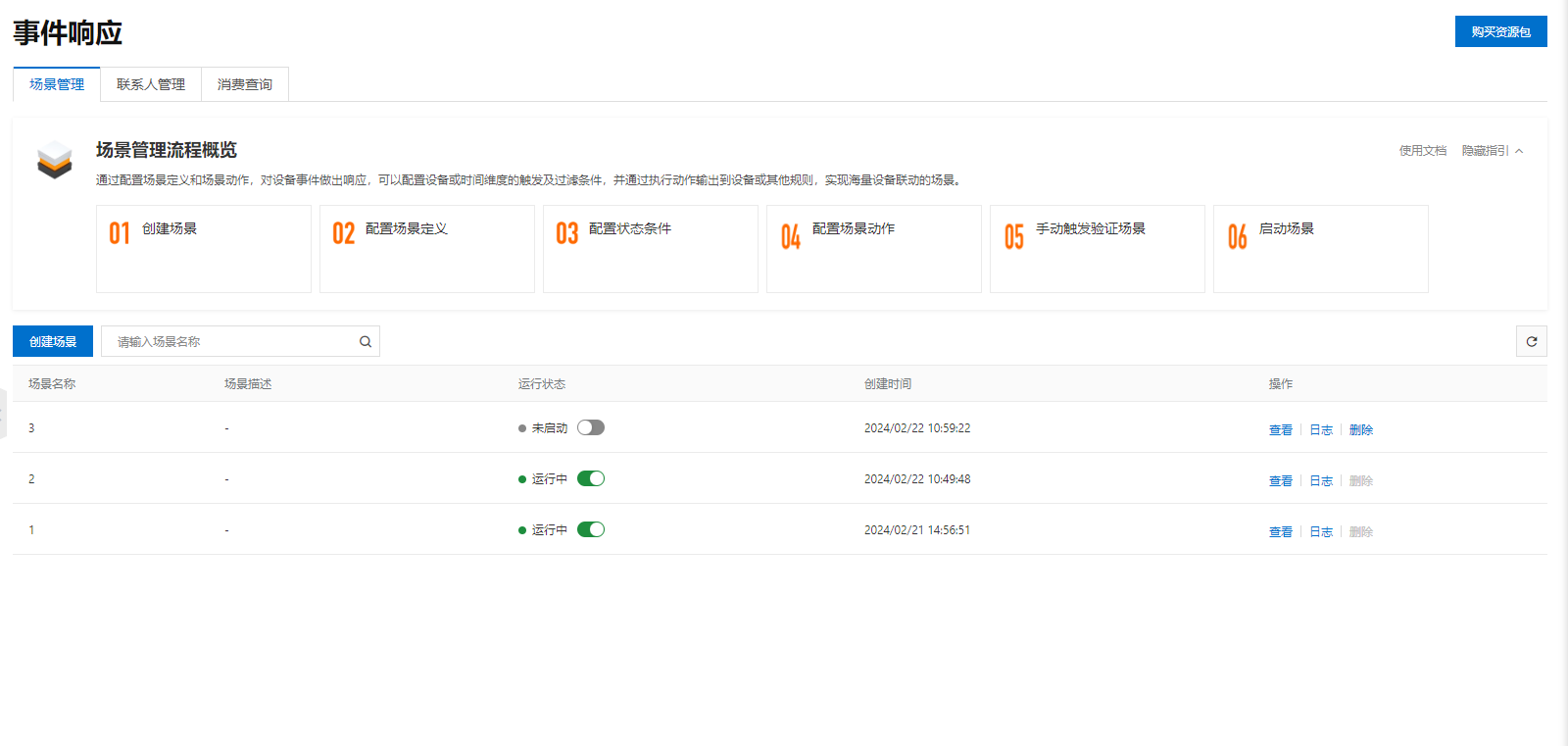




日志报错：{"Reason":"tsl parse: params not exist -> Alarm1Type1"}

## 阿里云事件响应

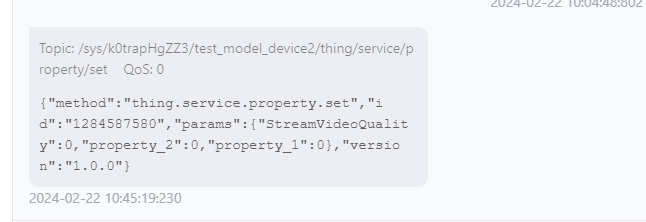
阿里云有 事件响应 这个功能类似规则引擎；



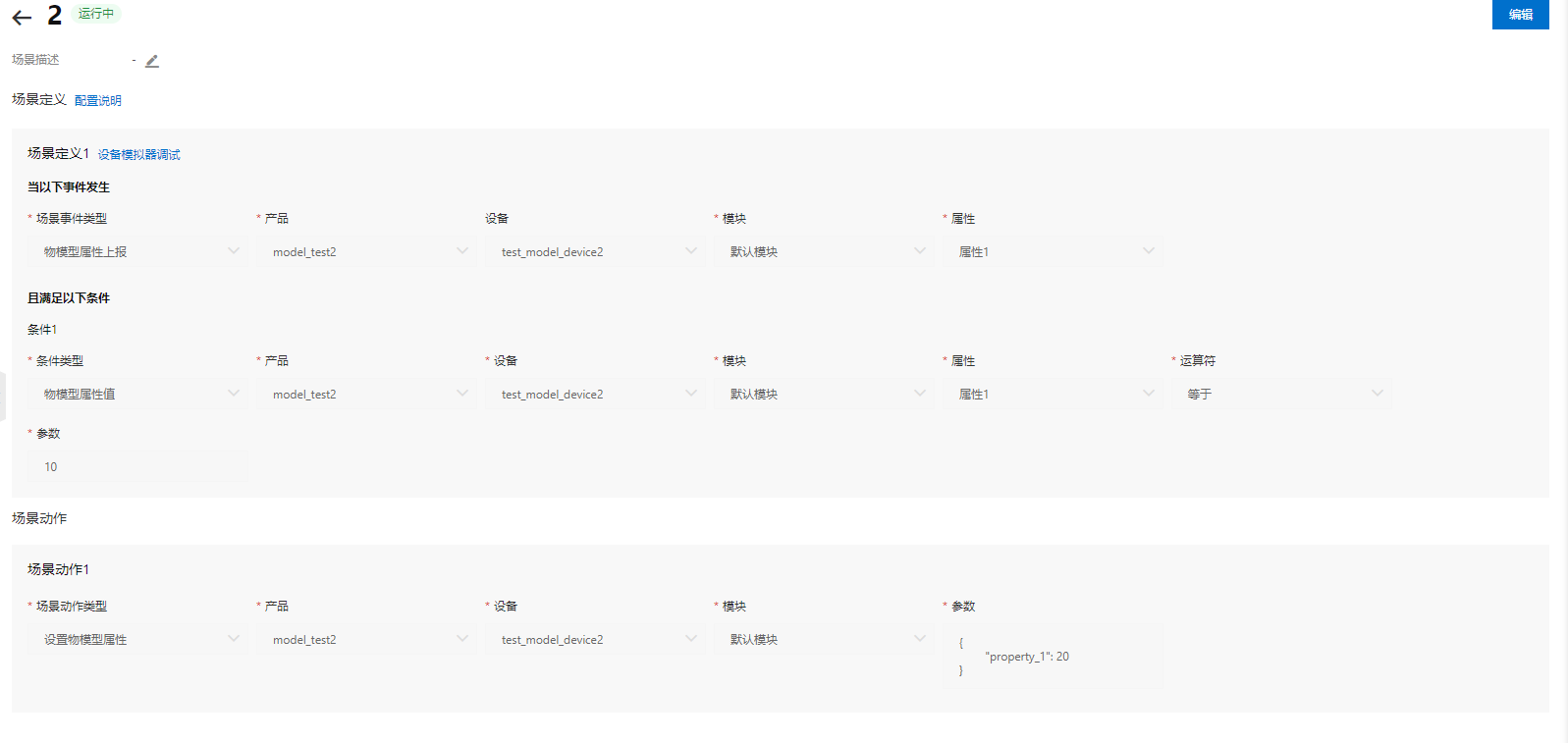
### 示例1：



设备上线后接收到控制topic：



### 示例2：

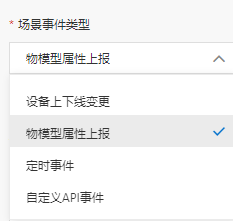


发布属性1为10值，接收到属性设置20的属性设置topic；（即使当前属性1为10值，再次接收属性1为10的值依旧触发）



### 事件响应相关设置参数

场景事件触发：



场景事件触发动作：



# 多功能网关soe事件修改方案

修改主要方向：

多功能网关fep\_gateway获取soe队列数据后，如何转换到IOT支持的格式；

IOT简易版页面如何创建事件，如何展示事件信息，如何接收推送事件格式

多功能网关fep\_mqttiotc如何分辨处理soe事件；

## 具体修改方案：

### IOT简易版展示如阿里云类似展示页面：

页面类似阿里云设计；前端自设计页面；

### IOT简易版页面如何创建事件及接收事件格式：

方案

事件是分类型的，soe事件为事件的一种类型，标识符固定如：event.soe;(一个产品中可能有多个属性都能产生soe事件，但都往一个固定的事件名 event.soe发送)

事件与属性code是隔离的，而创建soe的事件需要与对应的di属性有对应关系，所以用输出参数来关联di属性code）

比如输出参数中有di属性code的参数，整体事件json如下：

{

"msgid": "123456",

"eventid": "event.soe"(事件标识符)

"params": {

"value": {

" di属性code ":值，

" di属性code2 ":值

},

"ts": 1524448722000

}

}

* 请求参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 是否必选 | 类型 | 说明 |
| msgid | 必选 | String | 消息ID号。String类型的数字，取值范围0~4294967295。方便平台对消息的跟踪 |
| eventid | 必选 | String | 物模型中定义的事件标识 |
| params | 必选 | Object | 上报事件中的输出参数 |
| value | 必选 | Object | 物模型事件中的输出参数字段信息 |
| ts | 可选 | Long | 可选字段，属性上报时间戳，类型为UTC毫秒时间，不带时由物联网平台自动生成，建议设备携带防止平台时序问题 |

页面创建格式，就是依照阿里云正常创建事件格式设计，soe事件由部署人员去配置输出参数和标识符（标识符固定）；

需要讨论：输出参数的格式，人员配置soe事件的输出参数时，是产品中哪个属性code有soe事件就配置进去（可能一个产品有多个事件），还是类似固定输出参数将属性code和值放到固定输出参数中？ 这个需要确定；

### IOT简易版接收事件格式，推送时间格式及处理

方案：

简易版IOT需要新增订阅事件的topic，无需处理事件内的信息（可以直接使用事件json包放到params中），使用新格式推送；

{

"events":{

"devKey":"46tcj289afcz",

"params":{

"msgid": "1",

"eventid": "event.soe"(事件标识符)

"params": {

"value": {

" di属性code ":值，

" di属性code2 ":值

},

"ts": 1524448722000

}

},

"productKey":"46sz5222lb2b"

},

"fs":1696993741028,

"msgId":566580597607109088

}

推送处理的topic无需新增，使用推送属性变化的topic即可：推送格式新加参数（events的新格式发事件，与dataParams同级），如下方 推送格式展示；

### 多功能网关fep\_gateway获取soe队列数据后，如何转换到IOT支持的格式；

方案：

Fep\_gateway需要新增配置事件的信息并加载到内存中；（是否配置限制属性转换soe事件，还是只要收到soe就转json发送事件，需要讨论）

Fep\_gateway获取soe队列后得到di值，根据rtu-no-type查找到对应的 di属性code组包发送；

* Topic：$v5/{productKey}/{deviceSN}/sys/event/up

{

"msgid": "123456",

"eventid": "event.soe"(事件标识符)

"params": {

"value": {

" di属性code ":值，

" di属性code2 ":值

},

"ts": 1524448722000

}

}

### 多功能网关fep\_mqttiot处理简易版推送的soe事件

方案：

新增事件处理模块；

接收事件时先解包看事件的类型，如果是soe事件，获取包中的di属性code,根据redis存储di属性code的相应rtu和点号，写入对应soe队列

# 推送格式展示

现在推送格式（简易版）

{

"dataParams":[

{

"devKey":"46tcj289afcz",

"params":{

"ai1":{"ts":1696993740645,"value":""},

"di1":{"ts":1696993740645,"value":""}

},

"productKey":"46sz5222lb2b"

},

{

"devKey":"46tcqpymaedf",

"params":{

"paradi1":{"ts":1696993740645,"value":""},

"paraai1":{"ts":1696993740645,"value":"0.000000"}

},

"productKey":"46tcmx0k6xxf"}

] ,

"fs":1696993741028,

"msgId":566580597607109088

}

事件新增：

推送格式要修改：推送格式新加events参数发事件，与dataParams同级；

合并如下：

方案：

{

"dataParams":[

{

"devKey":"464msya2iob1",

\*\*\*（简易版）

},

{

"devKey":"464msya2iob1",

\*\*\*

}

] ,

"events":{

"devKey":"46tcj289afcz",

"params":{

"msgid": "1",

"eventid": "event.soe"(事件标识符)

"params": {

"value": {

" di属性code ":值，

" di属性code2 ":值

},

"ts": 1524448722000

}

},

"productKey":"46sz5222lb2b"

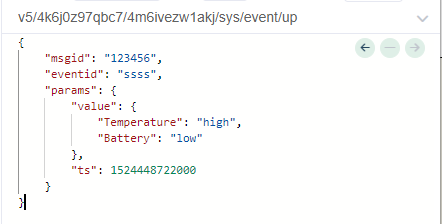
},

"fs":1696993741028,

"msgId":566580597607109088

}

Soe数据上传格式图



Soe数据上传推送

