修改方案

# 方案目的：

大平台端需要接受网关层的物联设备数据，使用标准版IOT则数据层数过多效率和配置都有问题。

目的：可以一键化生成配置；可以做到多功能网关IOT新建的物联设备和大平台数据自动连通，可以自动修改点表配置减少人员配置时间；

# 具体修改方案

## 改造

大平台绕过FEP-HA端新建一个规约直接连接多功能网关端的emqx部分，接收多功能网关IOT推送的全数据上传数据和使用多功能网关IOT的api下发控制；



**需要改动两个功能：**

**一键生成功能**：

多功能网关端IOT推送的数据是物联设备类型的数据格式，这种数据格式无法直接写入大平台的实时库中；需要写一个一键生成功能程序，调用多功能网关IOT的接口获取物联设备信息自动生成点代码数据和物联信息数据与点代码数据的对应关系写入数据库；

**新规约功能**：

**上传**：多功能网关IOT的推送信息是物联信息，需要规约自己转换数据处理；

**控制**：规约下发的控制格式是rtu-no-type-value，无法直接使用多功能网关IOT的控制接口，需要规约接收点控制数据后转换为模型数据控制；

## 一键生成部分

### 方案描述

程序需要调用多功能网关IOT的接口获取设备-模型数据，根据标准版IOT的生成逻辑生成点数据和物联设备-点号映射的数据，调用libddbc写入大平台数据库中；

### 具体方案：

#### 通知机制

程序写成一个服务，大平台主采集启动后会拉起服务，可以让工程人员使用工具（如postman）调用来通知执行一键生成功能。

#### 一键生成顺序

调用多功能网关IOT的三个接口，产品列表接口、产品属性列表接口、查询设备列表接口获取相关物联设备数据；获取大平台数据库中的表数据（iot\_rtu\_config）获取相关参数用于一键生成；

将数据按照java的逻辑，写入大平台的mysql中，写入ai点表（tb4007\_anapoint），di点表（tb4009\_digpoint），ao点表（tb4008\_ao），do点表（tb4011\_do），acc点表（tb4014\_accpoint），现场设备表（tb1004\_device），物联设备映射表（iot\_to\_platform）；（由于libddbc没有事务功能，为提高效率c端程序使用insert插入时sql语句写成多条插入，一次性插入几百条数据用以提高效率）

数据库表iot\_rtu\_config，iot\_to\_platform为新增的两张表，注意使用建表脚本创建；

##### 一键生成参数获取

以下是需要调用的三个接口和获取配置表的相关信息，用于一键生成写入。

###### 产品列表接口信息

接口：/iot/rest/api/product/queryProductList 获取产品列表（获取全部产品信息）

无入参

接收数据

{

 "code": 200,

 "msg": "成功",

 "data": [

 {

 "appSysId": 3,

 "appSysDesc": "环控",

 "childSysId": 3,

 "childSysDesc": "车站空调通风系统",

 "commodityId": 1,

 "commodityType": 1,

 "commodityCode": "dljkdkzq",

 "commodityName": "多联机空调控制器",

 "commodityDesc": "多联机空调控制器",

 "commodityKey": "4iiu3pd8lmtf",

 "commoditySecret": "4iiu3pd8lmtg",

 "platformType": 213,

 "createTime": "2023-12-26 17:01:03",

 "createBy": 1,

 "state": 0

 }

 ],

 "time": "2024-03-11 16:05:41",

 "success": true

}

* 参数说明

|  |  |
| --- | --- |
| 参数 | 说明 |
| appSysId | 系统 |
| childSysId | 子系统 |
| commodityCode | 产品代码 |
| platformType | 设备类型 |
| 等 |  |

###### 产品属性列表信息

接口：/iot/rest/api/thingmodel/queryProductPropertys 产品属性列表（根据产品代码获取此产品属性信息）

入参

{

 "commodityCode": "dljkdkzq"

}

接收数据

{

 "code": 200,

 "msg": "成功",

 "data": [

 {

 "commodityPropertyId": 1,

 "commodityPropertyCode": "1\_ZT",

 "commodityPropertyName": "1号VRV空调运行状态",

 "commodityPropertyDesc": "1号VRV空调运行状态",

 "dataType": 0,

 "dataBody": "{\"range\":{\"min\":\"0\",\"max\":\"1\"},\"unit\":\"\"}",

 "authorizationType": 1,

 "pointType": 3,

 "measType": "800",

 "almpriority": "10",

 "normalState": "-2",

 "ctrlBody": "[{\"ctrlValue\":\"\",\"descValue\":\"\"}]",

 "createTime": "2023-11-15 06:13:20",

 "createBy": 1,

 "state": 0

 }

],

 "time": "2024-03-11 16:06:39",

 "success": true

}

* 参数说明

|  |  |
| --- | --- |
| 参数 | 说明 |
| commodityPropertyCode | 产品属性代码 |
| pointType | 属性类型（1:ai,2:ao,3:di,4:do,5:acc） |
| authorizationType | 产品代码 |
| measType | 设备类型 |
| ctrlBody | 控制信息（do类型使用） |
| almpriority | 告警等级 |

###### 查询这边列表信息

接口：/iot/rest/api/device/queryDeviceList 查询设备列表（根据产品代码获取此产品下所有设备信息）

入参

{

 "commodityCode": "dljkdkzq",

 "keyWord": "",

 "tagWord": ""

}

接收数据

{

 "code": 200,

 "msg": "成功",

 "data": [

 {

 "appSysId": 3,

 "appSysDesc": "环控",

 "childSysId": 3,

 "childSysDesc": "车站空调通风系统",

 "commodityId": 1,

 "commodityType": 1,

 "commodityCode": "dljkdkzq",

 "commodityName": "多联机空调控制器",

 "commodityDesc": "多联机空调控制器",

 "commodityKey": "4iiu3pd8lmtf",

 "commoditySecret": "4iiu3pd8lmtg",

 "platformType": 213,

 "facilityId": 1,

 "facilityCode": "ZJS\_VRV",

 "facilityName": "多联机空调控制器",

 "facilityDesc": "多联机空调控制器",

 "facilityKey": "4iiv1c1wwqmb",

 "facilitySecret": "4iiv1c1wwqmc",

 "authorityArea": "3",

 "bayareaNo": "21",

 "createTime": "2023-12-26 17:11:31",

 "createBy": 1,

 "state": 0

 }

 ],

 "time": "2024-03-11 16:08:33",

 "success": true

}

* 参数说明

|  |  |
| --- | --- |
| 参数 | 说明 |
| facilityCode | 设备代码 |
| appSysId | 系统 |
| childSysId | 子系统 |
| commodityCode | 产品代码 |
| platformType | 设备类型 |
| authorityArea | 权限区域 |
| bayareaNo | 间隔区域号 |

###### rtu映射表信息(iot\_rtu\_config）

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段长度 | 字段描述 |
| group\_code | Varchar | 50 | 组代码 |
| app\_sys\_id | Int | 11 | 系统 |
| rtu\_no | Int | 11 | RTU服务号（RTUNO） |

一键生成时，根据设备所属的系统，使用对应的组代码和RTU服务号；

例如：



右侧为rtu映射表信息(iot\_rtu\_config），左侧为系统表；XCS,1,5表示物联设备的系统（appSysId）为1时（TASK系统），插入的组代码为XCS,生成的rtu为5；

##### 一键生成写入

接口参数获取中pointType是属性类型（1:ai,2:ao,3:di,4:do,5:acc），当属性是AI,DI,ACC时，分别插入插入AI,DI,ACC表；属性是AO时,同时插入AI,AO表且表点代码相同；当属性是DO时，同时插入DI,DO表且表点代码相同。

 不要求每次一键生成保留之前的rtu-no信息（直接写大平台且只有相关规约处理，不用强制要求点号不变）。插入AI时直接获取pointType等于1,2的值，产生rtu-no插入。插入AO时获取pointType等于2的值，产生rtu-no插入。AO类型产生的AI点号和AO点号要求一样，易于联系AI,AO。（DO类似）

下方为插入数据库规则，IOT来源参数名称为插入的数据来源，若为空则插入语句时可以不插入（组sql语句时可以不组为空的字段）；POINTNO自动生成，其他都有参数依赖；

###### 设备表TB1004\_DEVICE插入规则

|  |  |  |  |
| --- | --- | --- | --- |
| 大平台表字段名 | 表字段类型 | 表字段长度 | IOT来源参数名称 |
| F1004\_DEVCODE | char | 64 | facilityCode |
| F1004\_DEVDESC | char | 128 | facilityDesc |
| F1001\_GROUPCODE | char | 24 | 组代码（表参数） |
| F1010\_APPSYSID | tinyint | 3 | appSysId |
| F1011\_CHILDSYSID | smallint | 6 | childSysId |
| F1005\_DEVTYPEID | int | 11 | platformType |
| F1000\_AOJ | int | 11 | authorityArea |
| F1004\_PRODUCER | char | 120 | 　 |
| F1004\_PRODUCTDATE | int | 11 | 默认0 |
| F1004\_RUNDATE | int | 11 | 当前时间(示例:20240101) |
| F1004\_LONGEVITY | int | 11 | 默认0 |
| F1004\_GRAPHCODE | char | 80 | 　 |
| F1004\_LOCATION | char | 80 | 　 |
| F1004\_DEVNUM | char | 64 | 默认"" |
| F1004\_ONLINESTATE | tinyint | 3 | 默认0 |
| F1004\_FAULTSTATE | tinyint | 3 | 默认0 |
| F1004\_EARLYALMSTATE | tinyint | 3 | 默认0 |
| F1004\_TAGSTATE | tinyint | 3 | 默认0 |
| F1004\_DEVORDER | int | 11 | 默认0 |
| F1004\_MODEL | char | 40 | 　 |
| F1004\_ASSETSCODE | char | 64 | 　 |
| F1004\_MANAGEMENT | char | 64 | 　 |
| F1004\_LINENAME | char | 16 | 默认"" |
| F1004\_INSTALLDATE | int | 11 | 　 |
| F1004\_INSTALLER | char | 120 | 　 |
| F1004\_RATEDCURRENT | double | 0 | 　 |
| F1004\_RATEDVOLTAGE | double | 0 | 　 |
| F1004\_SUMSWITCHCOUNT | int | 11 | 　 |
| F1004\_RUNDURATION | int | 11 | 　 |
| F1004\_OVERRIDESTATE | tinyint | 3 | 　 |
| F1004\_RESERVED | double | 0 | 　 |
| F1004\_FAULTTOTAL | int | 11 | 　 |

###### AI点表TB4007\_ANAPOINT插入规则

|  |  |  |  |
| --- | --- | --- | --- |
| 大平台表字段名 | 表字段类型 | 表字段长度 | IOT来源参数名称 |
| F4007\_POINTCODE | char | 64 | facilityCode + "\_" + commodityPropertyCode |
| F4007\_POINTDESC | char | 128 | facilityDesc + " " + commodityPropertyDesc |
| F1001\_GROUPCODE | char | 24 | 组代码 |
| F1004\_DEVCODE | char | 64 | facilityCode |
| F4004\_MEASTYPEID | smallint | 6 | measType |
| F4007\_POINTTYPE | int | 11 | 默认1 |
| F4007\_AOJ | int | 11 | authorityArea |
| F4007\_ALMPRIO | tinyint | 3 | almpriority |
| F4007\_MANENTRYINHIB | tinyint | 3 | 默认0 |
| F4007\_CONTROLLABLE | tinyint | 3 |  如该ai有对应的ao点，则1，否则0 |
| F1002\_BAYAREANO | tinyint | 3 | bayareaNo |
| F2001\_RTUNO | smallint | 6 | RTUNO |
| F4007\_FTUNO | smallint | 6 | 默认-1 |
| F4007\_POINTNO | int | 11 | POINTNO |
| F4007\_ZERODBAND | double | 0 | 默认0 |
| F4007\_DACDBAND | double | 0 | 默认0 |
| F4007\_CONVFACTA1 | double | 0 | 默认1 |
| F4007\_CONVFACTA2 | double | 0 | 默认0 |
| F4007\_CONVFACTB | double | 0 | 默认0 |
| F4007\_BREAKPOINT | int | 11 | 默认0 |
| F4007\_HIREASON | double | 0 | 默认0 |
| F4007\_LOREASON | double | 0 | 默认0 |
| F4007\_ROCLIMIT | double | 0 | 默认0 |
| F4007\_ALARMPERMPT | char | 64 | 　 |
| F4007\_ALARMDBAND | double | 0 | 默认0 |
| F4007\_ALARMDELAY | tinyint | 3 | 默认0 |
| F4007\_ALARMGRAPH | char | 40 | 　 |
| F4007\_NORMALALMMODE | smallint | 6 | 默认0 |
| F4007\_EMERGALMMODE | smallint | 6 | 默认0 |
| F4007\_SAVESTYLE | tinyint | 3 | 默认0 |
| F4007\_SAVEINTERVAL | smallint | 6 | 默认15 |
| F4007\_PLANINTERVAL | smallint | 6 | 默认0 |
| F4007\_CONVMODE | tinyint | 3 | 默认0 |
| F1010\_APPSYSID | tinyint | 3 | appSysId |
| F4007\_VALDEFAULT | double | 0 | 默认0 |
| F4007\_UNITS | char | 8 | dataBody值中的unit |

###### AO点表TB4008\_AO插入规则

|  |  |  |  |
| --- | --- | --- | --- |
| 大平台表字段名 | 表字段类型 | 表字段长度 | IOT来源参数名称 |
| F1001\_GROUPCODE | char | 24 | 组代码 |
| F4007\_POINTCODE | char | 64 | facilityCode + "\_" + commodityPropertyCode |
| F4008\_AOJ | int | 11 | authorityArea |
| F2001\_RTUNO | smallint | 6 | RTUNO |
| F4008\_POINTNO | smallint | 6 | POINTNO |
| F4008\_CONVFACTA | double | 0 | 默认1 |
| F4008\_CONVFACTB | double | 0 | 默认0 |
| F4008\_EXETIMELIM | smallint | 6 | 默认25 |
| F4008\_RETTIMELIM | smallint | 6 | 默认20 |
| F4008\_SUPERVISE | tinyint | 3 | 默认0 |
| F4008\_CTRLACTION | tinyint | 3 | 默认0 |
| F4008\_CTLPERMSVPT | char | 64 | 　 |
| F1010\_APPSYSID | tinyint | 3 | appSysId |

###### DI点表TB4009\_DIGPOINT插入规则

|  |  |  |  |
| --- | --- | --- | --- |
| 大平台表字段名 | 表字段类型 | 表字段长度 | IOT来源参数名称 |
| F4009\_POINTCODE | char | 64 | facilityCode + "\_" + commodityPropertyCode |
| F4009\_POINTDESC | char | 128 | facilityDesc + " " + commodityPropertyDesc |
| F1001\_GROUPCODE | char | 24 | 组代码 |
| F1004\_DEVCODE | char | 64 | facilityCode |
| F4004\_MEASTYPEID | smallint | 6 | measType |
| F4009\_POINTTYPE | int | 11 | 默认1 |
| F4009\_DATACAT | char | 24 | 默认0 |
| F4009\_AOJ | int | 11 | authorityArea |
| F4009\_ALMPRIO | tinyint | 3 | almpriority |
| F4009\_MANENTRYINHIB | tinyint | 3 | 默认0 |
| F4009\_DISTURBANCE | tinyint | 3 | 默认0 |
| F4039\_DISTURBGROUPNO | smallint | 6 | 默认0 |
| F4009\_CONTROLLABLE | tinyint | 3 | 如该di有对应的do点，则1，否则0 |
| F1002\_BAYAREANO | tinyint | 3 | bayareaNo |
| F2001\_RTUNO | smallint | 6 | RTUNO |
| F4009\_FTUNO | smallint | 6 | 默认-1 |
| F4009\_DOUBLENO | tinyint | 3 | 默认0 |
| F4009\_POINTNO1 | int | 11 | POINTNO |
| F4009\_POINTNO2 | int | 11 | 默认-1 |
| F4009\_NORMALSTATE | smallint | 6 | normalState |
| F4009\_COMPLEMENT | tinyint | 3 | 默认0 |
| F4009\_DELAY | tinyint | 3 | 默认0 |
| F4009\_ALARMMODE | smallint | 6 | 默认0 |
| F4009\_ALARMPERMPT | char | 64 | 　 |
| F4009\_ALARMGRAPH | char | 40 | 　 |
| F4009\_INCIDENTMODE | tinyint | 3 | 默认0 |
| F4009\_INCIDENTLIM | smallint | 6 | 默认0 |
| F4009\_SWITCHLIM | smallint | 6 | 默认0 |
| F4009\_BYPASSPT | char | 64 | 　 |
| F4009\_INCIDENTCODE | char | 64 | 　 |
| F4009\_VALDEFAULT | tinyint | 3 | 默认0 |
| F1010\_APPSYSID | tinyint | 3 | appSysId |

###### DO点表TB4011\_DO插入规则

属性字段中ctrlBody中可能有多个ctrlValue和descValue，就相当于写多条信息；例如ctrlValue有0,1两个值，do包就插入两条且其他字段都一致；

|  |  |  |  |
| --- | --- | --- | --- |
| 大平台表字段名 | 表字段类型 | 表字段长度 | IOT来源参数名称 |
| F1001\_GROUPCODE | char | 24 | 组代码 |
| F4011\_POINTCODE | char | 64 | facilityCode + "\_" + commodityPropertyCode |
| F4011\_POINTDESC | char | 128 | facilityDesc + " " + commodityPropertyDesc |
| F4011\_CTRLVALUE | tinyint | 3 | ctrlBody字段值中的ctrlValue |
| F4025\_DESCVALUE | tinyint | 3 | ctrlBody字段值中的descValue |
| F4011\_AOJ | int | 11 | authorityArea |
| F2001\_RTUNO | smallint | 6 | RTUNO |
| F4011\_POINTNO | smallint | 6 | POINTNO (备注:F4011\_POINTCODE相同F4011\_CTRLVALUE不同,F4011\_POINTNO需要设置为一样) |
| F4011\_EXETIMELIM | smallint | 6 | 默认25 |
| F4011\_RETTIMELIM | smallint | 6 | 默认20 |
| F4011\_SUPERVISE | tinyint | 3 | 默认0 |
| F4011\_CTRLACTION | tinyint | 3 | 默认0 |
| F4011\_CTLPERMSVPT | char | 64 | 　 |
| F1010\_APPSYSID | tinyint | 3 | appSysId |
| F4011\_COMPLEMENT | tinyint | 3 | 默认0 |
| F4011\_CHECKRTUNO | smallint | 6 | 默认0 |
| F4011\_CHECKPOINTNO | smallint | 6 | 默认0 |

###### ACC点表TB4014\_ACCPOINT插入规则

|  |  |  |  |
| --- | --- | --- | --- |
| 大平台表字段名 | 表字段类型 | 表字段长度 | IOT来源参数名称 |
| F4014\_POINTCODE | char | 64 | facilityCode + "\_" + commodityPropertyCode |
| F4014\_POINTDESC | char | 128 | facilityDesc + " " + commodityPropertyDesc |
| F1001\_GROUPCODE | char | 24 | 组代码 |
| F1004\_DEVCODE | char | 64 | facilityCode |
| F4004\_MEASTYPEID | smallint | 6 | measType |
| F4014\_POINTTYPE | int | 11 | 默认1 |
| F4014\_AOJ | int | 11 | authorityArea |
| F4014\_ALMPRIO | tinyint | 3 | almpriority |
| F4014\_ALARMGRAPH | char | 40 | 　 |
| F4014\_MANENTRYINHIB | tinyint | 3 | 默认0 |
| F1002\_BAYAREANO | tinyint | 3 | bayareaNo |
| F2001\_RTUNO | smallint | 6 | RTUNO |
| F4014\_FTUNO | smallint | 6 | 默认-1 |
| F4014\_POINTNO | int | 11 | POINTNO |
| F4014\_BASECODE | float | 0 | 默认0 |
| F4014\_METER | float | 0 | 默认1 |
| F4014\_FULLCODE | float | 0 | 默认10000000 |
| F4014\_FACTOR | float | 0 | 默认1 |
| F4014\_EXTEND | float | 0 | 默认10 |
| F4014\_ALMMODE | smallint | 6 | 默认0 |
| F4014\_SAVE | tinyint | 3 | 默认0 |
| F4014\_SAVEINTERVAL | smallint | 6 | 默认15 |
| F1010\_APPSYSID | tinyint | 3 | appSysId |

###### 物联设备映射表（iot\_to\_platform）插入规则

物联设备映射表（iot\_to\_platform）：

这张表主要是插入上边ai,ao,di,do,dd表的集合，插入这五张表的记录数和本表的记录数一致；

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段长度 | 字段描述 | IOT来源参数名称 |
| commodity\_name | char | 50 | 产品名称 | commodityName |
| commodity\_key | char | 50 | 产品键 | commodityKey |
| facility\_code | char | 50 | 设备代码 | facilityCode |
| facility\_desc | char | 50 | 设备描述 | facilityDesc |
| facility\_key | char | 50 | 设备键 | facilityKey |
| commodity\_property\_name | char | 50 | 产品属性名称 | commodityPropertyName |
| commodity\_property\_code | char | 50 | 产品属性代码 | commodityPropertyCode |
| commodity\_property\_desc | char | 50 | 产品属性描述 | commodityPropertyDesc |
| ctrl\_value | int | 11 | Do控制值 | 属性为DO类型时，ctrlBody中ctrlValue的值。其他类型填写-10 |
| point\_code | char | 64 | 点代码 | facilityCode + "\_" + commodityPropertyCode |
| point\_desc | char | 128 | 点描述 | facilityDesc + " " + commodityPropertyDesc |
| point\_type | int | 11 | 点类型 | pointType（1:ai,2:ao,3:di,4:do,5:acc） |
| rtu\_no | int | 11 | Rtu服务号 | RTUNO |
| point\_no | int | 11 | 点号 | POINTNO |

## 大平台规约逻辑修改

### 方案描述

规约初始化时根据物联设备映射表（iot\_to\_platform）创建map队列，创建数据上传的map和数据下控的map两种类型，以支持点号和模型属性之间的转换；规约初始化后需要监控rtu号来获取命令，根据物联设备映射表（iot\_to\_platform）获取对应的所有rtu号来监视rtu服务下控；（大平台物联规约（mqttiotc规约）是处理标准版IOT数据的规约，可以对照着看；）

### 修改方案

#### 物联数据上传

订阅多功能网关IOT的全数据topic，使用数据上传map查找出点号关系，然后写入实时库；（数据上传映射map： facilitykey:propertycode – 点号信息（rtu,no,type ））

多功能网关IOT推送数据格式：

Topic: /sys/device/allData

{

"dataParams":[

{

"devKey":"46tcj289afcz",

"params":{

"Temperature": {"value": "37.0","ts": 1524448722000},

"Battery": {"value": 23.6,"ts": 1524448722000}

},

"productKey":"46sz5222lb2b"

}

],

"fs":1524448722000,

"msgId":123456

}

转换后写入大平台的实时库中；

#### 物联数据下控

根据获取物联设备映射表（iot\_to\_platform）的所有rtu号，监视这些rtu的控制命令；

收到控制命令后根据ao,do命令使用控制map来查找对应的物模型数据，使用多功能网关IOT的控制接口下发控制；（数据下控的map：分为ao,do的map，rtu:no:type – 模型信息（facilitykey,commoditykey,propertycode））

多功能网关IOT控制接口格式：

Api: {IP}:{PORT}/api/facilityProperty/setUpFacilityPro

{

 "deviceKey": "46sz5222lb2b ",

 "pointCode": " Temperature ",

 "productKey": "46tcj289afcz ",

 "value": "1"

}

#### SOE事件功能：

支持事件部分功能处理；根据多功能网关的soe事件处理（多功能网关独立程序fep\_mqttiotc的soe事件处理），修改到规约中；

# 附录

## 标准版IOT的物联数据流程

### 数据流程

标准版IOT+大平台的物联数据上传和下控流程；



### 一键生成

首先要先执行一键生成功能，标准版IOT的一键生成是根据已经在标准版IOT端配置好的物联设备信息，根据分组以点代码（类似rtu服务，点号，类型信息）维度写入大平台数据库中，同时生成大平台点维度数据和物联设备信息的映射；

### 物联数据上传

（标准版IOT一键生成：每次一键生成都会全部重写点代码对应的rtu-no-type不会保留之前的点号顺序但点代码保持不变；大平台的前端使用的是点代码来获取值，所以即使点号索引改变但不影响前端使用）

物联设备连接emqx服务上传数据，标准版IOT订阅物联设备数据根据映射转换成大平台可读取的点数据发送给emqx服务，大平台规约订阅标准版IOT平台数据解包出rtu-no-type-value使用接口写入大平台实时库；

格式如下：

物联设备数据上传格式：

Topic:$v5/{productKey}/{deviceSN}/sys/property/up

{

"msgid": "123456",

"params": {

"Temperature": {

"value": "37.0",

"ts": 1524448722000

 },

"Battery": {

"value": 23.6,

"ts": 1524448722000

 }

 }

}

标准版IOT推送数据格式：

Topic: /sys/device/data/{groupid}

{

"dataParams":[

{

"devKey":"470oxpvobnku",

"groupId":"5",

"pointCode":"Temperature",

"pointNo":"1",

"rtu":"5",

"type":"1",

"value":"37.0"

},

{

"devKey":"470oxpvobnku",

"groupId":"5",

"pointCode":"Battery",

"pointNo":"2",

"rtu":"5",

"type":"1",

"value":"23.6"

}

],

"fs":1524448722000,

"msgId":123456

}

### 物联数据下控

大平台接收控制命令（rtu-no-type-value），大平台规约读取到命令后调用标准版IOT的控制接口（属性下控）,标准版IOT接收接口命令转换为物联格式数据下发控制topic给具体的物联设备：

格式如下：

大平台主采集调用IOT控制接口格式：

Api: {IP}:{PORT}/api/facilityProperty/setUpDevPro

{

"rtu": "5",

"pointNo": "1",

"type": "2",

"value": "1"

}

标准版IOT下控控制格式：

Topic:$v5/{productKey}/{deviceSN}/sys/property/down

{

"msgid": "123456",

"params": {

"Temperature": "1"

 }

}

## 简易版IOT（多功能网关IOT）物联数据流程

多功能网关的物联数据上传和下控流程；



### 一键生成

一键生成功能，简易版IOT的一键生成通知FEP-HA端，FEP-HA的fep\_server程序会使用简易版IOT提供的接口获取设备和模型的信息，生成模型属性和点的对应关系写入db表和reids中；

### 物联数据上传

（简易版IOT一键生成：FEP-HA每次一键生成有对应的映射表，保证之前的模型属性映射的点号索引不变）

物联设备连接emqx服务上传数据，简易版IOT订阅物联设备数据整合发送给emqx服务，fep\_mqttiotc程序订阅简易版IOT推送的物联数据使用映射转换为点号写入实时库+redis中；

格式如下：

物联设备数据上传格式：

Topic:$v5/{productKey}/{deviceSN}/sys/property/up

{

"msgid": "123456",

"params": {

"Temperature": {

"value": "37.0",

"ts": 1524448722000

 },

"Battery": {

"value": 23.6,

"ts": 1524448722000

 }

 }

}

简易版IOT推送数据格式：

Topic: /sys/device/allData

{

"dataParams":[

{

"devKey":"46tcj289afcz",

"params":{

"Temperature": {"value": "37.0","ts": 1524448722000},

"Battery": {"value": 23.6,"ts": 1524448722000}

},

"productKey":"46sz5222lb2b"

}

],

"fs":1524448722000,

"msgId":123456

}

### 物联数据下控

Fep\_mqttiotc程序接收控制命令（rtu-no-type-value），fep\_mqttiotc使用映射转换成模型数据调用简易版IOT的控制接口（属性下控）,简易版IOT接收接口数据后下发控制topic给具体的物联设备：

格式如下：

Fep\_mqttiotc用简易版IOT控制接口格式：

Api: {IP}:{PORT}/api/facilityProperty/setUpFacilityPro

{

 "deviceKey": "46sz5222lb2b ",

 "pointCode": " Temperature ",

 "productKey": "46tcj289afcz ",

 "value": "1"

}

简易版IOT下控控制格式：

Topic:$v5/{productKey}/{deviceSN}/sys/property/down

{

"msgid": "123456",

"params": {

"Temperature": "1"

 }

}