



华为 IoT 平台

## 北向 SDK API 参考

文档版本 01

发布日期 2017-12-07

版权所有 © 华为技术有限公司 2017。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

## 华为技术有限公司

地址：深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址：<http://www.huawei.com>

客户服务邮箱：[support@huawei.com](mailto:support@huawei.com)

客户服务电话：4008302118

# 目 录

<b>1 前言</b>	<b>1</b>
<b>2 接口列表</b>	<b>2</b>
2.1 应用安全接入	3
2.1.1 Auth(鉴权)	3
2.1.2 Refresh Token(刷新 token)	4
2.2 设备管理	6
2.2.1 注册直连设备	6
2.2.2 发现非直连设备	7
2.2.3 查询设备激活状态	9
2.2.4 删除设备	10
2.2.5 修改设备信息	11
2.3 数据采集	12
2.3.1 按条件批量查询设备信息列表	12
2.3.2 查询单个设备信息	15
2.3.3 查询设备历史数据	16
2.3.4 查询设备能力	18
2.4 设备服务调用	19
2.5 规则	20
2.5.1 创建规则	20
2.5.2 更新规则	24
2.5.3 修改单个规则状态	29
2.5.4 删除规则	30
2.5.5 查找规则	30
2.6 订阅平台数据	31
2.6.1 北向应用订阅平台数据	31
<b>3 常用数据结构体定义</b>	<b>33</b>
3.1 DeviceInfo 结构体	33
3.2 DeviceService 结构体	35
3.3 QueryDeviceDTOCloud2NA 结构体	35
3.4 ServiceCommand 结构体	35
3.5 ServiceProperty 结构体	36

# 1 前言

本文档系统化描述IoT Platform对北向应用开放的能力全集、集成原理和集成参考样例等信息，帮助集成开发者快速而准确的掌握集成方法从而高效实现特定的业务需求。本文档主要包含如下几个部分：

一接口列表。主要描述能力开放的接口集合，详细介绍每个接口的功能、输入参数、输出参数、和调用示例等信息。

二常用数据结构定义。是对多个接口都会用到的数据结构做一个详细介绍，方便开发者集中查看。

## 修改记录

修改记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

文档版本	发布日期	修改说明	修改影响
1	2016/11/28	第一次发布	

## 2 接口列表

注意:北向提供的SDK API 是平台与Application之间的http接口的封装。

调用过程中，都要先初始化上下文调用信息：

```
String APP_ID = "ZpM****kca";
String APP_SECRET = "10m****g0a";
ConfigService.init("https", " *.*.*.", "8743"); //server info
// init ssl info      UrlConfiguration.initSSL(SSLUtil.SSL_TWO_AUTH_WITH_NO_DNS_CHECK,
      "/**", "**", "/**", "**");
// init login
UrlConfiguration.setIotIp(ConfigService.getIotServerIp());
      UrlConfiguration.setIotPort(ConfigService.getIotServerPort());
UrlConfiguration.setProtocolType(ConfigService.getProtocolType());
      AuthHttpClient.getInstance().setKey(APP_ID);
AuthHttpClient.getInstance().setSecret(APP_SECRET);
```

其中，初始化SSL信息中（若不需要使用制定的证书认证，不需调用SSL初始化），参数说明如下：

参数	必选/可选	类型	描述
sslType	必选	int	SSL检查的类型，当前有两个值： SSLUtil.SSL_TWO_AUTH_WITH_DNS_CHECK 双向认证检查DNS认证 SSLUtil.SSL_TWO_AUTH_WITH_NO_DNS_CHECK 双向认证不检查DNS认证
caServerFilePath	必选	String	服务器jks证书地址（相对于当前项目CLASSPATH的地址），需包含在开发的项目中
caServerFilePWD	必选	String	服务器jks证书密码
caClientFilePath	必选	String	客户端pkcs12证书地址（相对于当前项目CLASSPATH的地址），需包含在开发的项目中

参数	必选/可选	类型	描述
caClientFilePWD	必选	String	客户端pkcs12证书密码



注意

应用开发语言若是JAVA请使用JDK1.8。

应用和IoT平台之间请使用安全传输协议TLS 1.1/1.2。

[2.1 应用安全接入](#)

[2.2 设备管理](#)

[2.3 数据采集](#)

[2.4 设备服务调用](#)

[2.5 规则](#)

[2.6 订阅平台数据](#)

## 2.1 应用安全接入

Application携带在IoT平台产生的app ID 和密码过来，调用鉴权接口，获取鉴权token。  
请参考本文档 2.1.1鉴权章节进行开发。

SDK中封装类：com.huawei.iom.sdk.api.service.token.AccessTokenService

### 2.1.1 Auth(鉴权)

接口功能

实现第三方系统在访问开放API之前的认证。

调用方法

```
public static AppLoginDTONa2Cloud loginFromNA(AppLoginDTONa2Cloud  
appLoginDTONa2Cloud) throws BaseException
```

注意事项

鉴权接口是调用其他API的前提，北向接口除了鉴权接口（Auth），其他接口调用都需要在request header中携带参数app\_key和Authorization:Bearer {accessToken}。app\_key为参数中的appId，Authorization中{accessToken}的值即为调用Auth接口获取到的accessToken。

参数说明

参数	必选/可选	类型	描述
appLoginDTONa2Cloud	必选	AppLoginDTONa2Cloud	包含北向登录信息的对象

AppLoginDTONA2Cloud对象:

参数	必选/可选	类型	描述
appId	必选	String	用户名, 填写应用程序ID
secret	必选	String	登录用户口令, 填写应用程序密码

返回结果

字段	必选/可选	类型	描述
AppLoginDTOCloud2NA	必选	AppLoginDTOCloud2NA	返回北向的认证后对象

AppLoginDTOCloud2NA对象:

字段	必选/可选	类型	描述
scope	必选	String	范围, 默认值default
tokenType	必选	String	鉴权token类型, 默认值bearer
expiresIn	必选	integer	平台生成并返回accessToken的有效时间, 单位秒
accessToken	必选	String	Oauth 2.0 鉴权参数
refreshToken	必选	String	刷新Token, 用于刷新已有token

调用示例

```
String APP_ID = "ZpM****kca";
String APP_SECRET = "10m****g0a";
ConfigService.init("https", " *.*.*.", "8743");
AppLoginDTONA2Cloud appLoginDTONA2Cloud = new AppLoginDTONA2Cloud();
appLoginDTONA2Cloud.setAppId(APP_ID);
appLoginDTONA2Cloud.setSecret(APP_SECRET);
// login
AppLoginDTOCloud2NA appLoginDTOCloud2NA = AccessTokenService.loginFromNA(appLoginDTONA2Cloud);
System.out.println("login response: " + appLoginDTOCloud2NA);
```

输出:

```
login response: AppLoginDTOCloud2NA [accessToken:7d8****fec, tokenType:bearer, scope:[read, write], expiresIn:43199, refreshToken:a3e****b28]
```

## 2.1.2 Refresh Token(刷新 token)

接口功能

accessToken快过期时，第三方系统通过调用此接口，重新获取可用token。

### 调用方法

```
public static AppLoginDTOCloud2NA  
refreshTokenFromNA(AppRefreshTokenDTONA2Cloud appRefreshTokenDTONA2Cloud)  
throws BaseException
```

### 注意事项

### 参数说明

参数	必选/可选	类型	描述
appRefreshTokenDTONA2Cloud	必选	AppRefreshTokenDTONA2Cloud	刷新token的请求对象

AppRefreshTokenDTONA2Cloud对象：

参数	必选/可选	类型	描述
appId	必选	String	用户名，填写应用程序ID
secret	必选	String	登录用户口令，填写应用程序密码
refreshToken	必选	String	鉴权时获取到的refreshToken

### 返回结果

字段	必选/可选	类型	描述
AppLoginDTOCloud2NA	必选	AppLoginDTOCloud2NA	返回北向的认证后对象

AppLoginDTOCloud2NA对象：

字段	必选/可选	类型	描述
scope	必选	String	范围，默认值default
tokenType	必选	String	鉴权token类型，默认值bearer
expiresIn	必选	integer	平台生成并返回accessToken的有效时间，单位秒



字段	必选/可选	类型	描述
accessToken	必选	String	Oauth 2.0 鉴权参数
refreshToken	必选	String	刷新Token，会生成个新的

### 调用示例

```
String APP_ID = "ZpM***kca";
String APP_SECRET = "10m***g0a";
ConfigService.init("https", "*,*,*,*", "8743");
AppLoginDTONA2Cloud appLoginDTONA2Cloud = new AppLoginDTONA2Cloud();
appLoginDTONA2Cloud.setAppId(APP_ID);
appLoginDTONA2Cloud.setSecret(APP_SECRET);
System.out.println("login request: " + appLoginDTONA2Cloud);
// login
AppLoginDTOCLOUD2NA appLoginDTOCLOUD2NA =
AccessTokenService.loginFromNA(appLoginDTONA2Cloud);
// refresh token
AppRefreshTokenDTONA2Cloud appRefreshTokenDTONA2Cloud = new AppRefreshTokenDTONA2Cloud();
appRefreshTokenDTONA2Cloud.setAppId(APP_ID);
appRefreshTokenDTONA2Cloud.setSecret(APP_SECRET);
appRefreshTokenDTONA2Cloud.setRefreshToken(appLoginDTOCLOUD2NA.getRefreshToken());
appLoginDTOCLOUD2NA = AccessTokenService.refreshTokenFromNA(appRefreshTokenDTONA2Cloud);
System.out.println("refresh token response: " + appLoginDTOCLOUD2NA);
```

输出:

```
refresh token response: AppLoginDTOCLOUD2NA [accessToken:dab***20b, tokenType:bearer, scope:
[read, write], expiresIn:43199, refreshToken:c50***b9d]
```

## 2.2 设备管理

设备管理提供了Application申请设备的增，删，改，查接口。

提供修改设备或者设备所属传感器基本信息的功能。

app向IoT平台申请新的设备,IoT平台分配对应的设备验证码，待设备携带验证码请求接入平台后，分配其id,密码，允许其使用。平台提供了增，删，改，查接口，实现对新设备接入灵活操作。对设备的管理请参考本文档进行开发。

SDK中封装类：com.huawei.iom.sdk.api.service.device.DeviceService

### 2.2.1 注册直连设备

#### 接口功能

应用程序添加设备，获取设备的验证码，并在设备访问南向接口时携带验证码，获取设备唯一标识和密码。

#### 调用方法

```
public static AddDeviceDTOCLOUD2NA registerDevice(String verifyCode, String nodeId)
```

```
throws IOException, HttpResponseException, BaseException
```

#### 参数说明

字段	必选/可选	类型	描述
verifyCode	可选	String	客户端给出verifyCode则返回的就是这个verifyCode，即使用客户端给出的verifyCode。如果不指定，系统自动生成。
nodeId	必选	String	设备唯一标识, 如：MAC或SIM卡号或设备esn号等。与设备对接时，必须与设备上报的nodeId一致。

### 返回结果

字段	必选/可选	类型	描述
AddDeviceDTOCloud2NA	必选	AddDeviceDTOCloud2NA	注册设备时的返回对象，内部属性值为null则注册失败

AddDeviceDTOCloud2NA对象：

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识，1-64个字节
verifyCode	必选	String	申请的临时验证码，设备可以通过验证码获取id和密码
timeout	必选	Integer	验证码有效时间，单位秒，设备需要在有效时间内接入平台
psk	必选	String	psk码，用于生成设备鉴权参数

### 调用示例

```
// set nodeId
String nodeId = "****"
// register device
AddDeviceDTOCloud2NA addDeviceDTOCloud2NA =
DeviceService.registerDevice(nodeId, nodeId);
```

## 2.2.2 发现非直连设备

### 接口功能

Application发送DISCOVERY命令给网关下的设备，实质是设备服务调用。

此接口应用场景一般用于通过网关添加传感器，若没有网关，该接口无效，请勿使用。

### 调用方法

```
public static CommandDT0Device2CloudReply sendDeviceCommand(String deviceId, String serviceId,
    CommandDTONA2Cloud request)
    throws IOException, HttpRequestException, HttpResponseException, BaseException
```

### URL参数说明：

字段	必选/可选	类型	描述
deviceId	必选	String	网关设备唯一标识，1-64个字节
serviceId	必选	String	取值"Discovery"
request	必选	CommandDTONA2Cloud	请求的命令信息

### CommandDTONA2Cloud对象：

字段	必选/可选	类型	描述
header	必选	CommandNA2CloudHeader	见CommandNA2CloudHeader对象说明
body	必选	Map	消息的消息体

### CommandNA2CloudHeader对象：

字段	必选/可选	类型	描述
requestId	可选	String	如果填写该requestId，就是用填写的requestId标识一个命令，如果不填写就由平台生成。1-32个字节
mode	必选	String	是否要确认消息，不需要确认消息："NOACK"，需要确认消息："ACK"，其它值无效。
from	必选	String	(1-128个字节)表示消息发布者的地址： Application发起的请求：/{serviceName}
method	必选	String	发现命令名称："DISCOVERY"
callbackURL	可选	String	命令返回的回调地址。回调地址中必须指定回调地址的端口。

返回结果:

字段	必选/可选	类型	描述
CommandDTODevice2CloudReply	必选	CommandDTODevice2CloudReply	返回对象

CommandDTODevice2CloudReply对象:

字段	必选/可选	类型	描述
status	必选	String	命令状态: 1-32字节 sent --- 已发送 delivered --- 已送达 executed --- 已执行
timestamp	必选	String	发送命令的时间戳, 1-32字节 时间格式: yyyyMMdd' T' HHmmss' Z' 如: 20151212T121212Z
requestId	必选	String	1-32个字节, 平台分配的序列号, 标识一个命令 需要requestId关联对应命令执行结果

调用示例

```
// info
String deviceId = "5df****afc";
String serviceId = "RawData";
CommandDTONA2Cloud commandDTONA2Cloud = new CommandDTONA2Cloud();
CommandNA2CloudHeader header = new CommandNA2CloudHeader();
header.setMode("ACK");
header.setFrom("/users/**");
header.setMethod("rawData");
header.setCallbackURL("****");
Map<String, String> body = new HashMap<String, String>();
body.put("**", "**");
commandDTONA2Cloud.setHeader(header);
commandDTONA2Cloud.setBody(body);
// send command
CommandDTODevice2CloudReply reply = DeviceService.sendDeviceCommand(deviceId, serviceId, commandDTONA2Cloud);
```

## 2.2.3 查询设备激活状态

接口功能

给定网关设备的设备唯一标识查询网关设备激活状态。

### 调用方法

```
public static QueryDeviceRegStatusDTOCloud2NA queryDeviceActiveStatus(String deviceId)
```

throws IOException, HttpRequestException, HttpResponseException, URISyntaxException, BaseException

### 参数说明

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识，1-64个字节

### 返回结果

字段	必选/可选	类型	描述
QueryDeviceRegStatusDTOCloud2NA	必选	QueryDeviceRegStatusDTOCloud2NA	返回对象

QueryDeviceRegStatusDTOCloud2NA对象：

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识，1-64个字节
activated	必选	boolean	网关是否通过验证码获取密码的状态标识
name	必选	String	设备名称

### 调用示例

```
// device info
String deviceId = "7f7d***e67";
// query device status info
QueryDeviceRegStatusDTOCloud2NA queryDeviceRegStatusDTOCloud2NA =
DeviceService.queryDeviceActiveStatus(deviceId);
```

## 2.2.4 删除设备

### 接口功能

用来删除设备。

### 调用方法

```
public static void deleteDevice(String deviceId)
throws IOException, HttpRequestException, HttpResponseException, URISyntaxException,
BaseException
```

### 参数说明

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识，1-64个字节

### 返回结果

无，失败时抛出异常。

### 调用示例

```
// device info
String deviceId = "7f7d***e67";
// delete device info
DeviceService.deleteDevice(deviceId);
```

## 2.2.5 修改设备信息

### 接口功能

修改设备信息的接口，目前用于设备改名，预留修改设备信息的能力。

### 调用方法

```
public static void renameDevice(String deviceId, UpdateDeviceInfoDTONA2Cloud request)
    throws IOException, HttpRequestException, HttpResponseException, URISyntaxException,
    BaseException
```

### 参数说明

字段	必选/可选	类型	描述
deviceId	可选	String	设备唯一标识，1-64个字节
request	可选	UpdateDeviceInfoDTONA2Cloud	自定义结构体（见下表）

UpdateDeviceInfoDTONA2Cloud对象：

字段	必选/可选	类型	描述
name	可选	String	设备名称
endUser	可选	String	所有者标识
mute	可选	Enum	表示设备是否处于冻结状态，即是否上报数据（处于冻结状态，则不上报数据），取值有"TRUE"，"FALSE"
manufacturerId	可选	String	厂商ID

字段	必选/可选	类型	描述
manufacturerName	可选	String	厂商名
location	可选	String	设备的位置
deviceType	可选	String	设备类型: 大驼峰命名方式 MultiSensor ContactSensor Camera Gateway
model	可选	String	型号 z-wave: ProductType + ProductId 16 进制: XXXX-XXXX 补0 对齐 如: 001A-0A12 其他协议再定
protocolType	可选	String	设备协议类型

### 返回结果

无，失败时抛出异常

### 调用示例

```
// device info
String deviceId = "7f7d***e67";
UpdateDeviceInfoDTONA2Cloud updateDeviceInfoDTONA2Cloud = new UpdateDeviceInfoDTONA2Cloud();
updateDeviceInfoDTONA2Cloud.setName("***");
updateDeviceInfoDTONA2Cloud.setEndUser("***");
// update device info
DeviceService.renameDevice(deviceId, updateDeviceInfoDTONA2Cloud);
```

## 2.3 数据采集

平台支持对网关事件的订阅，同时还支持查看传感器上报的历史数据，能按时、天、月等维度查看传感器上报的历史数据。数据采集实现了对网关数据收集。

SDK中封装类: com.huawei.iom.sdk.api.service.data.DataService

### 2.3.1 按条件批量查询设备信息列表

#### 接口功能

按条件批量查询设备信息列表，如查询指定网关下的所有设备信息列表。

#### 调用方法

```
public static QueryDevicesDTOCcloud2NA queryDevicesInfo(String appId, String gatewayId, String status, String nodeType,
```

Integer pageNo, Integer pageSize, String startTime, String endTime, String sort)  
throws IOException, HttpRequestException, URISyntaxException, ServiceException,  
HttpResponseException

#### 参数说明

字段	必选/可选	类型	描述
appId	可选	String	应用唯一标识
gatewayId	可选	String	设备唯一标识（含网关的设备唯一标识）。当设备是直连设备时，gatewayId为设备本身的deviceId。当设备是非直连设备时，gatewayId为设备所关联的直连设备（即网关）的deviceId。 gatewayId和pageNo不能同时为null，二者必选其一。
nodeType	可选	String	指定节点的类：ENDPOINT: 传感器. GATEWAY: 网关.
pageNo	可选	Integer	分页查询参数，pageNo=null时查询内容不分页；取值大于等于0的整数时分页查询，等于0时查询第一页。 gatewayId和pageNo不能同时为null，二者必选其一。
pageSize	可选	Integer	分页查询参数，取值大于等于1的整数，缺省：1
status	可选	String	设备状态,取值:ONLINE在线;OFFLINE 不在线;ABNORMAL异常状态
startTime	可选	String	查询注册设备信息时间在startTime之后的记录。时间格式： yyyyMMdd' T' HHmmss' Z' 如： 20151212T121212Z
endTime	可选	String	查询注册设备信息时间在endTime之前的记录。时间格式： yyyyMMdd' T' HHmmss' Z' 如： 20151212T121212Z
sort	可选	String	指定返回记录的排序，可取值ASC按注册设备的时间升序排列；DESC按注册设备的时间降序排列 缺省：DESC

注：gatewayId和pageNo不能同时为空。

#### 结果QueryDevicesDTOCloud2NA对象说明



字段	必选/可选	类型	描述
totalCount	可选	Long	查询的记录数量
pageNo	必选	Long	返回的页码序号(大于等于0的整数,0代表第一页)
pageSize	必选	Long	页码大小,当不分页查询时,取值等于totalCount
devices	必选	List<QueryDeviceDTOCloud2NA>	自定义结构, 见下表

#### QueryDeviceDTOCloud2NA结构

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识, 1-64个字节
gatewayId	可选	String	设备唯一标识。当设备是直连设备时, gatewayId为设备本身的deviceId。当设备是非直连设备时, gatewayId为设备所关联的直连设备(即网关)的deviceId。
nodeType	必选	String	指定节点的类型 ENDPOINT: 传感器. GATEWAY: 网关.
creationTime	必选	String	设备注册时间; 时间格式: yyyyMMdd' T' HHmmss' Z' 如: 20151212T121212Z
lastModifiedTime	必选	String	最新修改时间; 时间格式: yyyyMMdd' T' HHmmss' Z' 如: 20151212T121212Z
deviceInfo	必选	DeviceInfo	DeviceInfo的定义参见3.1
services	可选	List<DeviceService>	DeviceService的定义参见3.2

#### 调用示例

```
// query parameter deviceId没给值时，返回对象的属性值都为null(deviceId = null;)
String deviceId = "7f7****e67";//必选
String gatewayId = deviceId;//必选
String serviceId = null;
int pageNo = 0;
int pageSize = 1;
String startTime = null;
String endTime = null;
// query
QueryDeviceDataHistoryDTOCloud2NA queryDeviceDataHistoryDTOCloud2NA =
DataService.getDeviceHistory(deviceId, gatewayId, serviceId, pageNo, pageSize, startTime, endTime);
```

## 2.3.2 查询单个设备信息

### 接口功能

查询指定条件的设备信息。

### 调用方法

```
public static QueryDeviceDTOCloud2NA queryOneDeviceInfo(String deviceId, String appId)
    throws IOException, HttpRequestException, HttpResponseException, URISyntaxException,
    ServiceException
```

### 参数说明

字段	必选/可选	类型	描述
appId	可选	String	应用唯一标识
deviceId	必选	String	设备唯一标识（含网关的设备唯一标识），1-64个字节

### 返回结果QueryDeviceDTOCloud2NA对象说明

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识，1-64个字节
gatewayId	可选	String	网关唯一标识。当设备是直连设备时，gatewayId为设备本身的deviceId。当设备是非直连设备时，gatewayId为设备所关联的直连设备（即网关）的deviceId。
nodeType	必选	String	指定节点的类型ENDPOINT: 传感器，GATEWAY: 网关。
creationTime	必选	String	设备注册时间 时间格式: yyyyMMdd' T' HHmmss' Z' 如: 20151212T121212Z
lastModifiedTime	必选	String	最新修改时间 时间格式: yyyyMMdd' T' HHmmss' Z' 如: 20151212T121212Z

字段	必选/可选	类型	描述
deviceInfo	必选	DeviceInfo	DeviceInfo的定义参见3.1章节 <a href="#">3.1 DeviceInfo结构体</a>
services	可选	List<DeviceService>	Service的定义参见3.2章节 <a href="#">3.2 DeviceService结构体</a>

#### 调用示例

```
// query parameter deviceId没给值时，返回对象的属性值都为null(deviceId = null;)
String deviceId = "7f7d48ac-f7a1-4b36-b2df-a5d56633ee67";
// query
QueryDeviceDT0Cloud2NA queryDeviceDT0Cloud2NA = DataService.queryOneDeviceInfo(deviceId,
null);
```

## 2.3.3 查询设备历史数据

#### 接口功能

Application查询设备历史数据。

#### 调用方法

```
public static QueryDeviceDataHistoryDT0Cloud2NA getDeviceHistory(String deviceId,String gatewayId,
String serviceId, Integer pageNo, Integer pageSize,
String startTime, String endTime)
throws IOException, HttpRequestException, HttpResponseException, URISyntaxException,
ServiceException
```

#### 参数说明

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识，1-64个字节
gatewayId	必选	String	查询参数，网关的设备唯一标识。当设备是直连设备时，gatewayId为设备本身的deviceId。当设备是非直连设备时，gatewayId为设备所关联的直连设备（即网关）的deviceId。
serviceId	可选	String	服务Id
pageNo	可选	Integer	分页查询参数，pageNo=null时查询内容不分页；取值大于等于0的整数时分页查询,等于0时查询第一页
pageSize	可选	Integer	分页查询参数，取值大于等于1的整数，缺省值为1 设备历史数据最多保存10万条，这个数值有可能被运营商修改

字段	必选/可选	类型	描述
startTime	可选	String	查询参数，根据时间段查询的起始时间; 时间格式: yyyyMMdd' T' HHmmss' Z' 如: 20151212T121212Z 设备历史数据最多保存90天，这个数值有可能被运营商修改
endTime	可选	String	查询参数，根据时间段查询的结束时间; 时间格式: yyyyMMdd' T' HHmmss' Z' 如: 20151212T121212Z

返回结果QueryDeviceDataHistoryDTOCloud2NA对象:

字段	必选/可选	类型	描述
totalCount	可选	Long	查询的记录数量
pageNo	必选	Long	返回的页码序号(大于等于0的整数,0代表第一页)
pageSize	必选	Long	页码大小,当不分页查询时,取值等于totalCount
list	必选	List<DeviceDataHistoryDTO>	历史记录列表

DeviceDataHistoryDTO对象:

字段	必选/可选	类型	描述
serviceId	必选	String	服务ID
deviceId	必选	String	设备唯一标识，1-64个字节
gatewayId	必选	String	网关的设备唯一标识，1-64个字节。当设备是直连设备时，gatewayId为设备本身的deviceId。当设备是非直连设备时，gatewayId为设备所关联的直连设备（即网关）的deviceId。
appId	必选	String	应用程序ID
data	必选	JsonObject	设备上报的数据，数据是键值对，取值请参见Profile文件中服务能力表里面的propertyties。

字段	必选/可选	类型	描述
timestamp	必选	String	上报数据的时间戳 时间格式： yyyyMMdd' T' HHmmss' Z' 如： 20151212T121212Z

#### 调用示例

```
// query parameter
String deviceId = "7f7d48ac-f7a1-4b36-b2df-a5d56633ee67";//必选
String gatewayId = deviceId;//必选
String serviceId = null;
int pageNo = 0;
int pageSize = 1;
String startTime = null;
String endTime = null;
// query
QueryDeviceDataHistoryDTOCloud2NA queryDeviceDataHistoryDTOCloud2NA =
DataService.getDeviceHistory(deviceId, gatewayId, serviceId, pageNo, pageSize, startTime, endTime);
```

## 2.3.4 查询设备能力

### 接口功能

### 查询设备能力

### 调用方法

```
public static
QueryDeviceCapabilitiesDTOCloud2NA queryDeviceServiceCapabilities(String deviceId,
String gatewayId)
    throws IOException,
HttpRequestException, HttpResponseException, URISyntaxException, ServiceException
```

### 参数说明

字段	必选/可选	类型	描述
deviceId	条件可选	String	设备唯一标识，1-64个字节， deviceId与gatewayId必须有一个存在
gatewayId	条件可选	String	网关标识，1-64个字节， deviceId与gatewayId必须有一个存在。当设备是直连设备时，gatewayId为设备本身的deviceId。当设备是非直连设备时，gatewayId为设备所关联的直连设备（即网关）的deviceId。

返回QueryDeviceCapabilitiesDTOCloud2NA对象，结构如下：

字段	必选/可选	类型	描述
deviceCapabilities	必选	List<DeviceCapabilityDTO>	自定义结构体，见下表

#### DeviceCapabilityDTO

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识，1-64个字节
serviceCapabilities	必选	List<DeviceServiceCapabilityDTO>	设备的服务能力列表

#### DeviceServiceCapabilityDTO

字段	必选/可选	类型	描述
serviceId	必选	String	服务标识
serviceType	必选	String	服务类型
description	可选	String	描述
commands	必选	List<ServiceCommand>	支持的命令名称列表,参见结构体3.4
properties	必选	List<ServiceProperty>	支持的属性名称列表，参见结构体3.5

#### 调用示例

```
// query parameter
String deviceId = "7f7***e67";
String gatewayId = null;
// query
QueryDeviceCapabilitiesDTOCloud2NA queryDeviceCapabilitiesDTOCloud2NA =
DataService.queryDeviceServiceCapabilities(deviceId, gatewayId);
```

## 2.4 设备服务调用

Application给设备发送命令消息，例如实现对传感器的实时控制。平台分别提供了下发至设备或者具体某传感器的控制命令接口，下发消息的具体格式需要Application与设备自定义，平台在接口中进行封装，透传。

具体调用与“2.2.2发现非直连设备”一致，请参考该章节。

## 2.5 规则

Application能够根据具体的业务需要，制定规则（条件）和动作，当网关的状态达到指定规则后，以邮件或者短信的方式对用户进行实时提醒。根据业务需要，编排规则，指定场景触发。

SDK中封装类：com.huawei.iom.sdk.api.service.rule.RuleService

### 2.5.1 创建规则

接口功能

创建规则。

调用方法

```
public static RuleCreationResultDTO createRule(RuleDTO ruleInfo)
    throws BaseException
```

参数RuleDTO对象结构说明

字段	必选/可选	类型	描述
ruleId	可选	String	规则实例的id，仅在规则更新时有效，此处不带。
appKey	可选	String	应用实例的key值(0-128)
name	必选	String	规则名（并且接受1~128个字符）
description	可选	String	规则描述(0-256)
author	必选	String	创建此规则的用户id（0-128）
conditions	必选	List<ObjectNode>	条件列表，自定义结构体（见下表Condition-ObjectNode结构说明）
logic	可选	String	多条件之间的逻辑关系，支持and和or，默认为and
timeRange	可选	TimeRange	条件场景的时间段。TimeRange为自定义结构体（见下表）
actions	定时场景必选	List<ObjectNode>	规则的动作。Action为自定义结构体。（见下表Action-DeviceCMD结构说明）
matchNow	可选	String	表示是否立即触发，即是否理解进行条件判断，条件符合的话执行动作，取值有” yes”，” no”，默认为” no”。
status	可选	String	规则的状态，“active”代表激活状态，“inactive”代表未激活，默认为” active”状态

TimeRange结构体说明:

字段	必选/可选	类型	描述
startTime	必选	String	开始时间，格式为 "HH:mm"。
endTime	必选	String	结束时间，格式为 "HH:mm"，开始时间与结束时间一致，则认为是全天。
daysOfWeek	可选	String	星期列表，以逗号分隔。1代表周日，2代表周一，以此类推，默认每天。

Condition-ObjectNode结构说明:

字段	必选/可选	类型	描述
type	必选	String	条件触发condition类型，取值为 "DEVICE_DATA"
id	可选	String	对condition的编号，在一条规则范围内唯一，不指定时由IoT自动编号。
deviceInfo	必选	DeviceInfo	设备数据的定位信息，自定义结构，根据定位信息读取的数据作为比较表达式的左值
operator	必选	String	数据比较的操作符
value	必选	String	比较表达式的右值
transInfo	可选	Json	不需要平台理解的信息
duration	可选	Integer	规则执行延时时间，单位为分钟，默认为0不延时

DeviceInfo结构体说明:

字段	必选/可选	类型	描述
deviceId	必选	String	设备id
path	必选	String	以"/"分割的信息路径，按照当前的数据模型，路径为ServiceId/DataProperty，例如门磁状态为 "DoorWindow/status"

Condition-DailyTimer结构说明:



字段	必选/可选	类型	描述
type	必选	String	定时触发condition类型， “DAILY_TIMER”
id	可选	String	对condition的编号，在一条规则范围内唯一，不指定时由系统自动编号
time	必选	String	指定执行时间点，格式为 "HH:mm"
daysOfWeek	可选	String	星期列表，以逗号分隔，默认每天

Condition-CycleTimer结构说明：

字段	必选/可选	类型	描述
type	必选	String	周期性触发condition类型，取值为 “CYCLE_TIMER”
id	可选	String	对condition的编号，在一条规则范围内唯一，不指定时由系统自动编号
timeRange	必选	timeRange	限定的时间范围
interval	必选	Integer	周期，单位为分钟

Action-DeviceCMD结构说明：

字段	必选/可选	类型	描述
type	必选	String	下发控制命令action类型，取值为 “DEVICE_CMD”
id	可选	String	对action的编号，在一条规则范围内唯一，不指定时由系统自动编号
appKey	可选	String	应用实例的key值
deviceId	必选	String	设备id
cmd	必选	CMD	命令内容
transInfo	可选	Json	不需要平台理解的信息

CMD结构说明：

字段	必选/可选	类型	描述
messageType	必选	String	命令名字
messageBody	必选	JSON	下发命令的消息内容
serviceId	必选	String	服务id

Action-SMS结构说明：

字段	必选/可选	类型	描述
type	必选	String	发送短信action类型，取值为”SMS”
id	可选	String	对action的编号，在一条规则范围内唯一，不指定时由系统自动编号
msisdn	必选	String	短信通知地址
content	必选	String	通知内容
subject	必选	String	通知主题

Action-email结构说明：

字段	必选/可选	类型	描述
type	必选	String	发送邮件action类型，取值为“EMAIL”
id	可选	String	对action的编号，在一条规则范围内唯一，不指定时由系统自动编号
email	必选	String	邮箱地址
content	必选	String	通知内容
subject	必选	String	通知主题

返回结果RuleCreationResultDTO对象

字段	必选/可选	类型	描述
ruleId	必选	String	规则实例id

调用示例

```
String deviceId = "7f7***e67";
// set create info
RuleDTO ruleInfo = new RuleDTO();
ruleInfo.setName("***");
ruleInfo.setAuthor("***");
List<ObjectNode> conditions = new ArrayList<ObjectNode>();
DeviceInfo deviceInfo = new DeviceInfo();
deviceInfo.setDeviceId(deviceId);
deviceInfo.setPath("DoorWindow/status");
ObjectNode condition1 = JsonUtil.createObjectNode();
condition1.put("type", "DEVICE_DATA");
condition1.put("deviceInfo", JsonUtil.jsonObj2Sting(deviceInfo));
condition1.put("operator", "=");
condition1.put("value", "OPEN");
conditions.add(condition1);
ruleInfo.setConditions(conditions);
List<ObjectNode> actions = new ArrayList<ObjectNode>();
ObjectNode action1 = JsonUtil.createObjectNode();
action1.put("type", "DEVICE_CMD");
action1.put("deviceId", deviceId);
CMD cmd1 = new CMD();
cmd1.setServiceId("RawData");
cmd1.setMessageType("rawData");
ObjectNode msgBody1 = JsonUtil.createObjectNode();
msgBody1.put("rawData", "MLAZ");
cmd1.setMessageBody(msgBody1);
action1.put("cmd", JsonUtil.jsonObj2Sting(cmd1));
actions.add(action1);
ruleInfo.setActions(actions);
// create
RuleCreationResultDTO response = RuleService.createRule(ruleInfo);
```

## 2.5.2 更新规则

### 接口功能

更新规则。实际过程是删除原有规则，创建新的规则，会产生新的规则ID。

### 调用方法

```
public static RuleCreationResultDTO updateRule(RuleDTO ruleInfo)
    throws BaseException
```

### 参数RuleDTO对象说明

字段	必选/可选	类型	描述
ruleId	必选	String	需要更新的规则实例的id（创建规则成功后平台返回的ruleId）
appKey	必选	String	应用实例的key值(0-128)
name	必选	String	规则名（并且接受1-128个字符）
description	可选	String	规则描述(0-256)
author	必选	String	创建此规则的用户id（0-128）
conditions	必选	List<ObjectNode>	条件列表，自定义结构体（见下表）
logic	可选	String	多条件之间的逻辑关系，支持and和or，默认为and

字段	必选/可选	类型	描述
timeRange	可选	TimeRange	条件场景的时间段。TimeRange为自定义结构体（见下表）
actions	定时场景必选	List<ObjectNode>	规则的动作。Action为自定义结构体。（见下表）
matchNow	可选	String	表示是否立即触发，即是否理解进行条件判断，条件符合的话执行动作，取值有"yes"，"no"，默认为"no"
status	可选	String	规则的状态，“active”代表激活状态，“inactive”代表未激活，默认为"active"状态

TimeRange结构体说明：

字段	必选/可选	类型	描述
startTime	必选	String	开始时间，格式为"HH:mm"。
endTime	必选	String	结束时间，格式为"HH:mm"，开始时间与结束时间一致，则认为是全天。
daysOfWeek	可选	String	星期列表，以逗号分隔。1代表周日，2代表周一，以此类推，week周列表需与开始时间保持一致，默认每天。

Condition- ObjectNode结构说明：

字段	必选/可选	类型	描述
type	必选	String	条件触发condition类型，取值为"DEVICE_DATA"
id	可选	String	对condition的编号，在一条规则范围内唯一，不指定时由系统自动编号
deviceInfo	必选	DeviceInfo	设备数据的定位信息，自定义结构，根据定位信息读取的数据作为比较表达式的左值
operator	必选	String	数据比较的操作符
value	必选	String	比较表达式的右值

字段	必选/可选	类型	描述
transInfo	必选	Json	不需要平台理解的信息，规则触发之后通过规则事件返回给NA
duration	可选	Integer	规则执行延时时间，单位为分钟，默认为0不延时

DeviceInfo结构体说明：

字段	必选/可选	类型	描述
deviceId	必选	String	设备id
path	必选	String	以"/"分割的信息路径，按照当前的数据模型，路径为ServiceId/DataProperty，例如门磁状态为"DoorWindow/status"

Condition-DailyTimer结构说明：

字段	必选/可选	类型	描述
type	必选	String	定点触发condition类型，“DAILY_TIMER”
id	可选	String	对condition的编号，在一条规则范围内唯一，不指定时由系统自动编号
time	必选	String	指定执行时间点，格式为"HH:MM"
daysOfWeek	必选	String	星期列表，以逗号分隔，默认每天

Condition-CycleTimer结构说明：

字段	必选/可选	类型	描述
type	必选	String	周期性触发condition类型，取值为“CYCLE_TIMER”
id	可选	String	对condition的编号，在一条规则范围内唯一，不指定时由系统自动编号
timeRange	必选	timeRange	限定的时间范围

字段	必选/可选	类型	描述
interval	必选	Integer	周期，单位为分钟

Action-DeviceCMD结构说明：

字段	必选/可选	类型	描述
type	必选	String	下发控制命令action类型，取值为“DEVICE_CMD”
id	可选	String	对action的编号，在一条规则范围内唯一，不指定时由系统自动编号
appKey	必选	String	应用实例的key值
deviceId	必选	String	设备id
cmd	必选	CMD	命令内容
transInfo	必选	Json	不需要平台理解的信息，规则触发之后通过规则事件返回给NA

CMD结构说明：

字段	必选/可选	类型	描述
messageType	必选	String	命令名字
messageBody	必选	Map<String, String>	命令参数列表
serviceId	必选	String	服务id

Action-SMS结构说明：

字段	必选/可选	类型	描述
type	必选	String	发送短信action类型，取值为“SMS”
id	可选	String	对action的编号，在一条规则范围内唯一，不指定时由系统自动编号
msisdn	必选	String	短信通知地址
content	必选	String	通知内容

字段	必选/可选	类型	描述
subject	必选	String	通知主题

Action-email结构说明：

字段	必选/可选	类型	描述
type	必选	String	发送邮件action类型，取值为“EMAIL”
id	可选	String	对action的编号，在一条规则范围内唯一，不指定时由系统自动编号
email	必选	String	邮箱地址
content	必选	String	通知内容
subject	必选	String	通知主题

返回结果RuleCreationResultDTO对象：

字段	必选/可选	类型	描述
ruleId	必选	String	规则实例id

### 消息示例

```
Method:
PUT
request:
https://server:port/iocm/app/rule/v1.2.0/rules
Content-Type:application/json
Header:
"app_key: a0pVJb06fLzRShwOMh0ywFvGsIga "
"Authorization:Bearer f9a2cbbdabefae2b91b21c4172c0d534"
Body:
{
  "ruleId": "*****",
  "appKey": "a0pVJb06fLzRShwOMh0ywFvGsIga",
  "name": "scene condx",
  "description": "scene condx",
  "author": "23558",
  "conditions": [
    {
      "duration": 0,
      "transInfo": {
        "rightValue": "OPEN"
      },
      "id": "0",
      "type": "DEVICE_DATA",
      "value": "OPEN",
      "deviceInfo": {
        "path": "DoorWindow/status",
```

```

        "deviceId": "2770f874-dec4-468e-bcb1-66425aa9d546"
    },
    "operator": "="
  }
],
"logic": "and",
"timeRange": null,
"actions": [
  {
    "transInfo": {},
    "appKey": "aOpVJb06fLzRShwOMh0ywFvGsIga",
    "cmd": {
      "messageBody": {
        "mode": "ON",
        "format": "AVI",
        "time": 60
      },
      "messageType": "VIDEO_RECORD",
      "serviceId": "Camera"
    },
    "id": "8",
    "type": "DEVICE_CMD",
    "deviceId": "17c0d895-370f-4a97-8989-1ff07af9ec15"
  }
],
"matchNow": "no",
"status": "active"
}
response:
Status Code: 200 OK
Content-Type:application/json
Body:
{
  "ruleId": "576e51aa45706d1584f7c06f"
}

```

## 2.5.3 修改单个规则状态

### 接口功能

修改单个规则状态。

### 调用方法

```

public static int changeStatus(String ruleId, String status)
    throws BaseException

```

### 参数说明

字段	必选/可选	类型	描述
ruleId	必选	String	需要修改的规则id
status	必选	String	规则状态，active代表激活，inactive代表未激活

### 返回结果

返回执行状态，成功时返回200

### 调用示例

```
// init
```



```
String ruleId = "583***50c";
String status = "active"; //inactive active
// change status
int responseStatus = RuleService.changeStatus(ruleId, status);
```

## 2.5.4 删除规则

### 接口功能

删除规则。

### 调用方法

```
public static int deleteRule(String ruleId)
    throws BaseException, URISyntaxException
```

### 参数说明

字段	必选/可选	类型	描述
ruleId	必选	String	需要删除的规则id

### 返回结果

返回执行状态，成功时返回204

### 调用示例

```
// init
String ruleId = "583***531";
// delete
int responseStatus = RuleService.deleteRule(ruleId);
```

## 2.5.5 查找规则

### 接口功能

查找规则。

### 接口路径

```
public static List<RuleDTO> findRuleByNameAndAuthor(String ruleName, String author)
    throws BaseException, URISyntaxException
```

### 参数说明

字段	必选/可选	类型	描述
author	必选	String	用户id
name	可选	String	规则名称

### 返回结果

Rule列表，同创建/更新规则结构体

### 消息示例

```
// init
String ruleName = null;//
String author = "***";
// query
List<RuleDTO> ruleDTOList = RuleService.findRuleByNameAndAuthor(ruleName, author);
```

## 2.6 订阅平台数据

北向应用向平台订阅设备变更信息，一个APP一个事件对应一个订阅回调地址。

SDK中封装类：com.huawei.iom.sdk.api.service.subscribe.SubscribeService

备注说明：调用后返回的数据格式参见“华为IoT\_Platform\_API参考(北向)”文档。

### 2.6.1 北向应用订阅平台数据

接口功能

北向应用订阅设备变更通知，当设备发生变更时平台会推送给北向应用。

调用方法

```
public static int subscribeDeviceEvent(SubscribeDTONA2Cloud request)
    throws BaseException
```

参数说明

字段	必选/可选	类型	描述
notifyType	必选	String	通知类型，如： 1、bindDevice（设备激活通知） 2、deviceAdded（添加新设备） 3、deviceInfoChanged（设备信息变化） 4、deviceDataChanged（设备数据变化） 5、deviceDeleted（删除设备） 6、deviceEvent（设备事件） 7、messageConfirm（消息确认） 8、commandRsp（响应命令） 9、serviceInfoChanged（服务信息） 10、ruleEvent（规则事件）
callbackurl	必选	String	回调的url地址。回调地址中必须指定回调地址的端口。

返回结果

返回执行状态，成功时返回201

### 调用示例

```
// set parameter
SubscribeDTONA2Cloud request = new SubscribeDTONA2Cloud();
request.setNotifyType("bindDevice");
request.setCallbackurl("http://185.12.26.50:9999");
// subscribe event
int responseStatus = SubscribeService.subscribeDeviceEvent(request);
```

# 3 常用数据结构体定义

- 3.1 DeviceInfo结构体
- 3.2 DeviceService结构体
- 3.3 QueryDeviceDTOCloud2NA结构体
- 3.4 ServiceCommand结构体
- 3.5 ServiceProperty结构体

## 3.1 DeviceInfo 结构体

字段	必选/可选	类型	描述
nodeId	必选	String	设备唯一标识
name	可选	String	设备名称
description	可选	String	设备描述
manufactureId	必选	String	供应商标识
manufacturerName	可选	String	供应商名字
mac	可选	String	物理地址
location	可选	String	设备位置
deviceType	必选	String	设备种类： MultiSensor ContactSensor Camera Gateway

字段	必选/可选	类型	描述
model	必选	String	设备型号. 如果是Z-Wave协议, 型号就是ProductType + ProductId (用零填充, 如果需要) 格式的十六进制数,例如 001A-0A12. 如果是其他协议, 型号需要另一种格式表示
swVersion	可选	String	使用Z-Wave协议时的软件版本号。其格式是 major version number.minor version number, 例如, 1.1.
fwVersion	可选	String	固件版本号
hwVersion	可选	String	硬件版本号
protocolType	必选	String	协议类型 (Z-Wave).
signalStrength	可选	String	信号强度
bridgeId	可选	String	网桥标识
supportedSecurity	可选	String	是否支持安全模式, 其取值为TRUE或FALSE
isSecurity	可选	String	设备是否安全, 在安全模式为TRUE的情况下有效, 其取值为TRUE或FALSE
sigVersion	可选	String	信令版本
runningStatus	可选	String	设备运行状态: NORMAL: 正常 ABNORMAL: 异常 FAULT: 故障
status	可选	String	设备状态. ONLINE: 在线 OFFLINE: 下线 INBOX: 停用.
statusDetail	可选	String	设备状态的详细情况
mute	可选	String	设备是否进行屏蔽 TRUE: 被屏蔽. FALSE: 没被屏蔽.

## 3.2 DeviceService 结构体

字段	必选/可选	类型	描述
serviceType	必选	String	服务类型
serviceId	必选	String	服务ID标识
data	必选	JSON String	指定一个属性值对(AVP).
eventTime	必选	DateTime	时间格式 yyyyymmddThhmissZ , 例如, 20151212T121212Z.
serviceInfo	可选	ServiceInfo	业务信息, 此字段用于内部控制功能, 不用关心此字段

## 3.3 QueryDeviceDTOCloud2NA 结构体

字段	必选/可选	类型	描述
deviceId	必选	String	设备唯一标识
gatewayId	必选	String	网关唯一标识。当设备是非直连设备时, gatewayId为设备所关联的直连设备(即网关)的deviceId。当设备是直连设备时, gatewayId为设备本身的deviceId。
nodeType	必选	String	指定节点的类型 ENDPOINT: 传感器. GATEWAY: 网关.
createTime	必选	String	设备注册时间
lastModifiedTime	必选	String	设备信息最后修改时间
deviceInfo	必选	DeviceInfo	详见" <a href="#">3.1 DeviceInfo结构体</a> "
services	可选	List<DeviceService>	详见 " <a href="#">3.2 DeviceService结构体</a> "

## 3.4 ServiceCommand 结构体

字段	必选/可选	类型	描述
commandName	可选	String	命令

字段	必选/可选	类型	描述
paras	可选	List<ServiceCommandPara>	属性列表，参考下表

ServiceCommandPara结构体

字段	必选/可选	类型	描述
paraName	可选	String	参数名
dataType	可选	String	数据类型
required	可选	boolean	是否必选
min	可选	int	最小
max	可选	int	最大
step	可选	Double	步长
maxLength	可选	int	最大长度
unit	可选	String	单位（符号）
enumList	可选	List<String>	枚举类型列表

### 3.5 ServiceProperty 结构体

字段	必选/可选	类型	描述
propertyName	可选	String	属性名
dataType	可选	String	数据类型
required	可选	boolean	是否必选
min	可选	int	最小
max	可选	int	最大
step	可选	double	步长
maxLength	可选	int	最大长度
method	可选	String	访问方法：RW 可读R，可写W
unit	可选	String	单位（符号）
enumList	可选	List<String>	枚举类型列表