

# IoT 平台增强版

## API 参考

文档版本

07

发布日期

2019-05-28



版权所有 © 华为技术有限公司 2019。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。

# 目 录

<b>1 IoT 平台北向 API 参考.....</b>	<b>1</b>
1.1 总体说明.....	1
1.1.1 网络结构.....	1
1.1.2 修订记录.....	2
1.1.3 兼容性说明.....	2
1.2 接口列表.....	2
1.2.1 应用安全接入.....	2
1.2.1.1 鉴权.....	3
1.2.1.2 刷新 token.....	5
1.2.2 设备管理.....	7
1.2.2.1 注册直连设备.....	8
1.2.2.2 发现非直连设备.....	14
1.2.2.3 刷新设备密钥.....	14
1.2.2.4 修改设备信息.....	19
1.2.2.5 删除直连设备.....	24
1.2.2.6 删除非直连设备.....	27
1.2.2.7 查询设备激活状态.....	27
1.2.2.8 修改设备影子.....	29
1.2.2.9 查询设备影子.....	33
1.2.2.10 查询设备实时位置信息.....	39
1.2.3 批量处理.....	44
1.2.3.1 创建批量任务.....	44
1.2.3.2 查询指定批量任务信息.....	49
1.2.3.3 查询批量任务的子任务信息.....	54
1.2.4 订阅管理.....	58
1.2.4.1 订阅平台业务数据.....	59
1.2.4.2 订阅平台管理数据.....	62
1.2.4.3 查询单个订阅.....	65
1.2.4.4 批量查询订阅.....	67
1.2.4.5 删除单个订阅.....	70
1.2.4.6 批量删除订阅.....	72
1.2.5 消息推送.....	75
1.2.5.1 注册设备通知.....	75

1.2.5.2 绑定设备通知.....	80
1.2.5.3 设备信息变化通知.....	84
1.2.5.4 设备数据变化通知.....	88
1.2.5.5 批量设备数据变化通知.....	90
1.2.5.6 设备服务信息变化通知.....	92
1.2.5.7 删除设备通知.....	94
1.2.5.8 设备消息确认通知.....	95
1.2.5.9 设备命令响应通知.....	97
1.2.5.10 命令状态变化通知.....	100
1.2.5.11 设备事件通知.....	101
1.2.5.12 增加设备模型通知.....	103
1.2.5.13 删除设备模型通知.....	105
1.2.5.14 设备影子状态变更通知.....	107
1.2.5.15 软件升级状态变更通知.....	108
1.2.5.16 软件升级结果通知.....	110
1.2.5.17 固件升级状态变更通知.....	112
1.2.5.18 固件升级结果通知.....	114
1.2.6 命令下发.....	116
1.2.6.1 创建设备命令.....	116
1.2.6.2 查询设备命令.....	122
1.2.6.3 修改设备命令.....	129
1.2.6.4 批量创建设备命令.....	135
1.2.6.5 创建设备命令撤销任务.....	135
1.2.6.6 查询设备命令撤销任务.....	141
1.2.6.7 设备服务调用.....	148
1.2.7 数据采集.....	153
1.2.7.1 查询单个设备信息.....	153
1.2.7.2 批量查询设备信息.....	160
1.2.7.3 查询设备历史数据.....	169
1.2.7.4 查询设备影子历史数据.....	173
1.2.7.5 查询设备服务能力.....	177
1.2.8 设备组管理.....	184
1.2.8.1 创建设备组.....	184
1.2.8.2 删除设备组.....	188
1.2.8.3 修改设备组.....	190
1.2.8.4 查询设备组列表.....	193
1.2.8.5 查询指定设备组.....	196
1.2.8.6 查询指定设备组成员.....	198
1.2.8.7 增加设备组成员.....	201
1.2.8.8 删除设备组成员.....	205
1.2.9 设备升级.....	208
1.2.9.1 查询版本包列表.....	208

1.2.9.2 查询指定版本包.....	212
1.2.9.3 删除指定版本包.....	215
1.2.9.4 创建软件升级任务.....	216
1.2.9.5 创建固件升级任务.....	221
1.2.9.6 查询指定任务信息.....	226
1.2.9.7 查询指定任务的子任务详情.....	231
1.2.9.8 查询任务列表.....	235

## 2 IoT 平台北向 JAVA SDK API 参考.....**242**

2.1 SDK Demo 总体架构及使用说明.....	242
2.2 SDK 初始化配置及测试.....	243
2.2.1 NorthApiClient 类方法.....	243
2.2.2 Clientinfo 类方法.....	244
2.2.3 NorthApiException 类方法.....	244
2.2.4 SSLConfig 类方法.....	245
2.2.5 BasicTest 类方法.....	245
2.3 业务接口列表.....	246
2.3.1 应用安全接入.....	246
2.3.1.1 鉴权.....	246
2.3.1.2 刷新 token.....	247
2.3.1.3 定时刷新 token.....	249
2.3.1.4 停止定时刷新 token.....	250
2.3.2 设备管理.....	250
2.3.2.1 注册直连设备.....	251
2.3.2.2 刷新设备密钥.....	256
2.3.2.3 修改设备信息.....	260
2.3.2.4 删除直连设备.....	265
2.3.2.5 查询设备激活状态.....	267
2.3.2.6 查询设备实时位置信息.....	269
2.3.2.7 查询设备影子.....	273
2.3.2.8 修改设备影子.....	277
2.3.3 批量处理.....	280
2.3.3.1 创建批量任务.....	281
2.3.3.2 查询指定批量任务信息.....	285
2.3.3.3 查询批量任务的子任务信息.....	289
2.3.4 订阅管理.....	292
2.3.4.1 订阅平台业务数据.....	292
2.3.4.2 订阅平台管理数据.....	295
2.3.4.3 查询单个订阅.....	297
2.3.4.4 批量查询订阅.....	298
2.3.4.5 删除单个订阅.....	302
2.3.4.6 批量删除订阅.....	303
2.3.5 消息推送.....	305

2.3.5.1 注册设备通知.....	305
2.3.5.2 绑定设备通知.....	310
2.3.5.3 设备信息变化通知.....	313
2.3.5.4 设备数据变化通知.....	318
2.3.5.5 批量设备数据变化通知.....	320
2.3.5.6 设备服务信息变化通知.....	322
2.3.5.7 删除设备通知.....	324
2.3.5.8 设备消息确认通知.....	326
2.3.5.9 设备命令响应通知.....	328
2.3.5.10 设备事件通知.....	330
2.3.5.11 增加设备模型通知.....	332
2.3.5.12 删除设备模型通知.....	334
2.3.5.13 设备影子状态变更通知.....	336
2.3.5.14 软件升级状态变更通知.....	338
2.3.5.15 软件升级结果变更通知.....	340
2.3.5.16 固件升级状态变更通知.....	342
2.3.5.17 固件升级结果变更通知.....	344
2.3.5.18 NB 设备命令状态变化通知.....	346
2.3.6 命令下发（NB 命令）.....	348
2.3.6.1 创建设备命令.....	348
2.3.6.2 查询设备命令.....	353
2.3.6.3 修改设备命令.....	357
2.3.6.4 创建设备命令撤销任务.....	362
2.3.6.5 查询设备命令撤销任务.....	366
2.3.7 命令下发（非 NB 命令）.....	371
2.3.7.1 设备服务调用.....	371
2.3.8 数据采集.....	375
2.3.8.1 查询单个设备信息.....	375
2.3.8.2 批量查询设备信息列表.....	380
2.3.8.3 查询设备历史数据.....	385
2.3.8.4 查询设备影子历史数据.....	389
2.3.8.5 查询设备服务能力.....	392
2.3.9 设备组管理.....	396
2.3.9.1 创建设备组.....	396
2.3.9.2 删除设备组.....	399
2.3.9.3 修改设备组.....	400
2.3.9.4 查询设备组详情.....	403
2.3.9.5 查询指定设备组.....	405
2.3.9.6 查询指定设备组成员.....	407
2.3.9.7 增加设备组成员.....	409
2.3.9.8 删除设备组成员.....	412
2.3.10 设备升级.....	414

2.3.10.1 查询版本包列表.....	414
2.3.10.2 查询指定版本包.....	417
2.3.10.3 删除指定版本包.....	419
2.3.10.4 创建软件升级任务.....	420
2.3.10.5 创建固件升级任务.....	424
2.3.10.6 查询指定升级任务结果.....	428
2.3.10.7 查询指定升级任务子任务详情.....	432
2.3.10.8 查询升级任务列表.....	434

## 3 IoT 平台北向 PHP SDK API 参考..... 440

3.1 SDK Demo 总体架构及使用说明.....	440
3.2 SDK 初始化配置及测试.....	441
3.2.1 NorthApiClient 类方法.....	441
3.2.2 Clientinfo 类方法.....	442
3.2.3 NorthApiException 类方法.....	442
3.2.4 SSLConfig 类.....	442
3.3 业务接口列表.....	442
3.3.1 应用安全接入.....	442
3.3.1.1 鉴权.....	442
3.3.1.2 刷新 token.....	444
3.3.2 设备管理.....	446
3.3.2.1 注册直连设备.....	446
3.3.2.2 刷新设备密钥.....	451
3.3.2.3 修改设备信息.....	455
3.3.2.4 删除直连设备.....	460
3.3.2.5 查询设备激活状态.....	462
3.3.2.6 查询设备实时位置信息.....	464
3.3.2.7 查询设备影子.....	468
3.3.2.8 修改设备影子.....	473
3.3.3 批量处理.....	476
3.3.3.1 创建批量任务.....	476
3.3.3.2 查询指定批量任务信息.....	480
3.3.3.3 查询批量任务的子任务信息.....	484
3.3.4 订阅管理.....	487
3.3.4.1 订阅平台业务数据.....	487
3.3.4.2 订阅平台管理数据.....	489
3.3.4.3 查询单个订阅.....	491
3.3.4.4 批量查询订阅.....	493
3.3.4.5 删除单个订阅.....	495
3.3.4.6 批量删除订阅.....	497
3.3.5 消息推送.....	499
3.3.5.1 注册设备通知.....	499
3.3.5.2 绑定设备通知.....	503

3.3.5.3 设备信息变化通知	507
3.3.5.4 设备数据变化通知	511
3.3.5.5 批量设备数据变化通知	513
3.3.5.6 设备服务信息变化通知	515
3.3.5.7 删除设备通知	517
3.3.5.8 设备消息确认通知	518
3.3.5.9 设备命令响应通知	520
3.3.5.10 设备事件通知	522
3.3.5.11 增加设备模型通知	525
3.3.5.12 删除设备模型通知	526
3.3.5.13 设备影子状态变更通知	528
3.3.5.14 软件升级状态变更通知	530
3.3.5.15 软件升级结果变更通知	532
3.3.5.16 固件升级状态变更通知	534
3.3.5.17 固件升级结果变更通知	536
3.3.5.18 NB 设备命令状态变化通知	538
3.3.6 命令下发（NB 命令）	540
3.3.6.1 创建设备命令	540
3.3.6.2 查询设备命令	545
3.3.6.3 修改设备命令	549
3.3.6.4 创建设备命令撤销任务	553
3.3.6.5 查询设备命令撤销任务	557
3.3.7 命令下发（非 NB 命令）	562
3.3.7.1 设备服务调用	562
3.3.8 数据采集	566
3.3.8.1 查询单个设备信息	566
3.3.8.2 批量查询设备信息列表	570
3.3.8.3 查询设备历史数据	575
3.3.8.4 查询设备影子历史数据	578
3.3.8.5 查询设备服务能力	581
3.3.9 设备组管理	585
3.3.9.1 创建设备组	585
3.3.9.2 删除设备组	587
3.3.9.3 修改设备组	589
3.3.9.4 查询设备组详情	591
3.3.9.5 查询指定设备组	593
3.3.9.6 查询指定设备组成员	595
3.3.9.7 增加设备组成员	597
3.3.9.8 删除设备组成员	600
3.3.10 设备升级	602
3.3.10.1 查询版本包列表	602
3.3.10.2 查询指定版本包	604

3.3.10.3 删除指定版本包.....	606
3.3.10.4 创建软件升级任务.....	607
3.3.10.5 创建固件升级任务.....	611
3.3.10.6 查询指定升级任务结果.....	616
3.3.10.7 查询指定升级任务子任务详情.....	619
3.3.10.8 查询升级任务列表.....	622
<b>4 IoT 平台北向 Python SDK API 参考.....</b>	<b>627</b>
4.1 SDK Demo 总体架构及使用说明.....	627
4.2 SDK 初始化配置及测试.....	628
4.2.1 NorthApiClient 类.....	628
4.2.2 Clientinfo 类.....	628
4.3 业务接口列表.....	629
4.3.1 应用安全接入.....	629
4.3.1.1 鉴权.....	629
4.3.1.2 刷新 token.....	631
4.3.2 设备管理.....	633
4.3.2.1 注册直连设备.....	633
4.3.2.2 刷新设备密钥.....	641
4.3.2.3 修改设备信息.....	645
4.3.2.4 删除直连设备.....	650
4.3.2.5 查询设备激活状态.....	652
4.3.2.6 查询设备实时位置信息.....	654
4.3.2.7 查询设备影子.....	658
4.3.2.8 修改设备影子.....	662
4.3.3 批量处理.....	665
4.3.3.1 创建批量任务.....	666
4.3.3.2 查询指定批量任务信息.....	670
4.3.3.3 查询批量任务的子任务信息.....	674
4.3.4 订阅管理.....	677
4.3.4.1 订阅平台业务数据.....	677
4.3.4.2 订阅平台管理数据.....	680
4.3.4.3 查询单个订阅.....	682
4.3.4.4 批量查询订阅.....	683
4.3.4.5 删除单个订阅.....	686
4.3.4.6 批量删除订阅.....	687
4.3.5 消息推送.....	689
4.3.5.1 注册设备通知.....	689
4.3.5.2 绑定设备通知.....	694
4.3.5.3 设备信息变化通知.....	698
4.3.5.4 设备数据变化通知.....	702
4.3.5.5 批量设备数据变化通知.....	704
4.3.5.6 设备服务信息变化通知.....	707

4.3.5.7 删除设备通知.....	709
4.3.5.8 设备消息确认通知.....	710
4.3.5.9 设备命令响应通知.....	713
4.3.5.10 设备事件通知.....	715
4.3.5.11 增加设备模型通知.....	718
4.3.5.12 删除设备模型通知.....	720
4.3.5.13 设备影子状态变更通知.....	722
4.3.5.14 软件升级状态变更通知.....	724
4.3.5.15 软件升级结果变更通知.....	726
4.3.5.16 固件升级状态变更通知.....	728
4.3.5.17 固件升级结果变更通知.....	730
4.3.5.18 NB 设备命令状态变化通知.....	732
4.3.6 命令下发（NB 命令）.....	735
4.3.6.1 创建设备命令.....	735
4.3.6.2 查询设备命令.....	740
4.3.6.3 修改设备命令.....	744
4.3.6.4 创建设备命令撤销任务.....	748
4.3.6.5 查询设备命令撤销任务.....	753
4.3.7 命令下发（非 NB 命令）.....	758
4.3.7.1 设备服务调用.....	758
4.3.8 数据采集.....	762
4.3.8.1 查询单个设备信息.....	762
4.3.8.2 批量查询设备信息列表.....	766
4.3.8.3 查询设备历史数据.....	772
4.3.8.4 查询设备影子历史数据.....	775
4.3.8.5 查询设备服务能力.....	778
4.3.9 设备组管理.....	782
4.3.9.1 创建设备组.....	782
4.3.9.2 删除设备组.....	785
4.3.9.3 修改设备组.....	786
4.3.9.4 查询设备组详情.....	789
4.3.9.5 查询指定设备组.....	790
4.3.9.6 查询指定设备组成员.....	792
4.3.9.7 增加设备组成员.....	795
4.3.9.8 删除设备组成员.....	797
4.3.10 设备升级.....	799
4.3.10.1 查询版本包列表.....	799
4.3.10.2 查询指定版本包.....	802
4.3.10.3 删除指定版本包.....	804
4.3.10.4 创建软件升级任务.....	805
4.3.10.5 创建固件升级任务.....	809
4.3.10.6 查询指定升级任务结果.....	813

4.3.10.7 查询指定升级任务子任务详情.....	816
4.3.10.8 查询升级任务列表.....	819
<b>5 Agent Lite API 参考(Android).....</b>	<b>824</b>
5.1 开发者必读.....	824
5.2 接口列表.....	825
5.2.1 广播机制.....	825
5.2.2 直连设备接入.....	825
5.2.2.1 初始化 Agent Lite 资源.....	825
5.2.2.2 释放 Agent Lite 资源.....	826
5.2.2.3 绑定配置.....	827
5.2.2.4 设备绑定.....	828
5.2.2.5 设备解绑定命令接收.....	829
5.2.2.6 参数配置.....	830
5.2.2.6.1 配置业务参数.....	830
5.2.2.6.2 配置加密算法类型（可选）.....	832
5.2.2.7 设备登录.....	833
5.2.2.8 设备登出.....	835
5.2.3 网关管理非直连设备.....	836
5.2.3.1 设备添加.....	836
5.2.3.2 设备状态更新.....	837
5.2.3.3 设备删除.....	843
5.2.4 设备服务数据上报.....	844
5.2.5 设备命令接收.....	846
5.2.6 数据发布.....	847
5.3 常用数据结构定义.....	848
5.3.1 IotaDeviceInfo 类说明.....	849
5.3.2 BIND_IE_RESULT 信元参数说明.....	850
5.3.3 LOGIN_IE_REASON 信元参数说明.....	850
5.3.4 HUB_IE_RESULT 参数枚举说明.....	851
<b>6 Agent Lite API 参考(C).....</b>	<b>852</b>
6.1 开发者必读.....	852
6.2 接口列表.....	853
6.2.1 广播机制介绍.....	853
6.2.2 直连设备接入.....	853
6.2.2.1 初始化 Agent Lite 资源.....	854
6.2.2.2 释放 Agent Lite 资源.....	854
6.2.2.3 绑定配置.....	855
6.2.2.4 设备绑定.....	855
6.2.2.5 设备解绑定命令接收.....	857
6.2.2.6 参数配置.....	858
6.2.2.6.1 设置业务参数.....	858
6.2.2.6.2 设置加密算法类型参数（可选）.....	859

6.2.2.7 设备登录.....	859
6.2.2.8 设备登出.....	861
6.2.3 网关管理非直连设备.....	861
6.2.3.1 设备添加.....	861
6.2.3.2 设备状态更新.....	862
6.2.3.3 设备删除.....	865
6.2.4 设备服务数据上报.....	866
6.2.5 设备命令接收.....	867
6.2.6 数据发布.....	868
6.2.7 Json 组件使用说明.....	869
6.3 常用数据结构定义.....	872
6.3.1 ST_IOTA_DEVICE_INFO 结构体说明.....	872
6.3.2 EN_IOTA_BIND_IE_TYPE 消息信元枚举说明.....	874
6.3.3 EN_IOTA_BIND_RESULT_TYPE 参数枚举说明.....	875
6.3.4 EN_IOTA_LGN_IE_TYPE 消息信元枚举说明.....	875
6.3.5 EN_IOTA_LGN_REASON_TYPE 参数枚举说明.....	875
6.3.6 EN_IOTA_CFG_TYPE 参数枚举说明.....	876
6.3.7 EN_IOTA_HUB_IE_TYPE 消息信元枚举说明.....	877
6.3.8 EN_IOTA_HUB_RESULT_TYPE 参数枚举说明.....	877
6.3.9 EN_IOTA_DATATRANS_IE_TYPE 消息信元枚举说明.....	877
6.3.10 EN_IOTA_DEVUPDATE_IE_TYPE 消息信元枚举说明.....	878
6.4 数据类型定义.....	878
<b>7 Agent Lite API 参考(Java).....</b>	<b>881</b>
7.1 开发者必读.....	881
7.2 接口列表.....	882
7.2.1 观察者模式.....	882
7.2.2 直连设备接入.....	882
7.2.2.1 初始化 AgentLite 资源.....	882
7.2.2.2 释放 AgentLite 资源.....	883
7.2.2.3 绑定配置.....	884
7.2.2.4 设备绑定.....	885
7.2.2.5 设备解绑定命令接收.....	886
7.2.2.6 参数配置.....	887
7.2.2.6.1 配置业务参数.....	887
7.2.2.6.2 配置加密算法类型（可选）.....	889
7.2.2.7 设备登录.....	889
7.2.2.8 设备登出.....	890
7.2.3 网关管理非直连设备.....	891
7.2.3.1 设备添加.....	892
7.2.3.2 设备状态更新.....	893
7.2.3.3 设备删除.....	894
7.2.4 设备服务数据上报.....	895

7.2.5 设备命令接收.....	897
7.2.6 数据发布.....	898
7.3 常用数据结构定义.....	899
7.3.1 IotaDeviceInfo 类说明.....	899
7.3.2 IotaMessage 类说明.....	901
<b>8 IoT 平台 MQTT 接口参考.....</b>	<b>903</b>
8.1 概述.....	903
8.1.1 MQTT 介绍.....	903
8.1.2 消息格式.....	903
8.1.3 流程说明.....	904
8.1.3.1 业务流程.....	904
8.1.3.2 设备注册（一机一密）.....	905
8.2 接口列表.....	905
8.2.1 Topic 定义.....	905
8.2.2 MQTT CONNECT 连接鉴权.....	906
8.2.3 设备数据上报.....	907
8.2.4 平台命令下发.....	908
8.2.5 设备响应命令.....	909

# 1 IoT 平台北向 API 参考

总体说明  
接口列表

## 1.1 总体说明

### 1.1.1 网络结构



物联网平台提供了海量API接口给应用开发者。通过调用平台的接口，开发者可以开发出基于多种行业设备的应用，如公共事业、智慧家庭等，从而实现对设备的管理（包括设备的增、删、查、改）、数据采集、命令下发和消息推送等功能。

## 1.1.2 修订记录

配套物联网 平台版本	修改说明
19.2.RC1	<p>修改如下接口：</p> <ul style="list-style-type: none"><li>● 设备管理<ul style="list-style-type: none"><li>- 注册直连设备：新增请求参数deviceInfo, productId。</li><li>- 修改设备信息：新增请求参数customFields, imsi, ip, tags。</li></ul></li><li>● 批量处理<ul style="list-style-type: none"><li>- 创建批量任务：请求参数param结构体增加参数groupTag；请求参数taskType新增取值DeviceLocation、DeviceMod、DeviceDel和DeviceLicense；请求参数param结构体下的type参数新增取值GroupIdList。</li><li>- 查询指定批量任务信息：响应参数param结构体增加参数groupTag；响应参数taskType新增取值DeviceLocation、DeviceMod、DeviceDel和DeviceLicense；响应参数param结构体下的type参数新增取值GroupIdList。</li></ul></li><li>● 订阅管理 订阅平台业务数据：新增请求参数channel。</li></ul>

## 1.1.3 兼容性说明

对于接口版本做了演进或者接口url做了修改的接口，其对应老版本的接口依然可以继续使用，但接口的功能不再做增强。

建议使用新版本接口。

## 1.2 接口列表

参数位置说明

参数	描述
header	http头部的参数。
path	url中path部分的参数。
query	url中?后面的参数。
body	http body部分的参数。

## 1.2.1 应用安全接入

应用服务器获取鉴权信息，接入物联网平台，随后携带鉴权信息调用其他API接口。

### 1.2.1.1 鉴权

#### 典型场景

应用服务器首次访问物联网平台的开放API时，需调用此接口完成接入认证；应用服务器在物联网平台的认证过期后，需调用此接口重新进行认证，才能继续访问物联网平台的开放API。

#### 接口功能

支持应用服务器在首次访问物联网平台的开放API之前，完成在物联网平台的接入认证。

#### 注意事项

鉴权接口是调用其他API的前提，除了调用鉴权接口（Auth），其他接口调用都需要在request header中携带参数app\_key和Authorization。app\_key取值与请求参数中appId相同，Authorization的格式为Authorization: Bearer {accessToken}，其中accessToken通过调用Auth接口获取到的。

如果多次获取accessToken，则之前的accessToken失效，最后一次获取的accessToken才有效。请勿并发获取accessToken。

#### 接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/sec/v1.1.0/login
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
appId	必选	String(256)	body	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
secret	必选	String(256)	body	访问物联网平台的应用密钥，与appId对应，在物联网平台创建应用时由平台分配获得。

#### 响应参数

Status Code: 200 OK

参数	类型	描述
scope	String(256)	申请权限范围，即accessToken所能访问物联网平台资源的范围，参数值固定为default。
tokenType	String(256)	accessToken的类型，参数值固定为Bearer。
expiresIn	Integer(256)	accessToken的有效时间，参数值固定为3600秒。
accessToken	String(256)	鉴权参数，访问物联网平台API接口的凭证。
refreshToken	String(256)	鉴权参数，有效时间为1个月，用于“刷新Token”接口。当accessToken即将过期时，可通过“刷新Token”接口来获取新的accessToken。

## 请求示例

Method: POST

request:

`https://server:port/iocm/app/sec/v1.1.0/login`

`Content-Type:application/x-www-form-urlencoded`

`appId=*****&secret=*****`

## 响应示例

response:

Status Code: 200 OK

`Content-Type: application/json`

Body:

```
{  
    "scope": "default",  
    "tokenType": "Bearer ",  
    "expiresIn": "*****",  
    "accessToken": "*****",  
    "refreshToken": "*****"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	100449	The device is freezed cant operate.	用户无操作权限。 处理建议：请检查appId所对应用户是否有接口调用权限。
400	102202	Required Parameter is null or empty.	必选参数为空。 处理建议：请检查请求参数中的必选参数是否已填写。
401	100208	AppId or secret is not right.	appId或secret错误。 处理建议： <ul style="list-style-type: none"><li>● 请检查appId和secret是否有误，是否新增字符或缺少字符。</li><li>● 请检查请求路径中的IP地址是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.1.2 刷新 token

#### 典型场景

应用服务器通过鉴权接口获取到的accessToken是有有效时间的，在accessToken快过期时，应用服务器通过调用此接口，获取新的accessToken。

#### 接口功能

支持应用服务器在accessToken到期前，从物联网平台获取新的accessToken。

#### 接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/sec/v1.1.0/refreshToken
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
appId	必选	String(256)	body	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
secret	必选	String(256)	body	访问物联网平台的应用密钥，与appId对应，在物联网平台创建应用时由平台分配获得。
refreshToken	必选	String(256)	body	刷新token，用来获取一个新的accessToken。refreshToken在调用鉴权接口或刷新token接口时获得。

## 响应参数

Status Code: 200 OK

参数	类型	描述
scope	String(256)	申请的权限范围，参数值固定为default。
tokenType	String(256)	鉴权token的类型，参数值固定为Bearer。
expiresIn	Integer(256)	accessToken的有效时间，参数值固定为3600秒。
accessToken	String(256)	鉴权参数，访问物联网平台API接口的凭证。
refreshToken	String(256)	鉴权参数，有效时间为1个月，用于“刷新Token”接口。当accessToken即将过期时，可通过“刷新Token”接口来获取新的accessToken。

## 请求示例

Method: POST

request:

`https://server:port/iocm/app/sec/v1.1.0/refreshToken`

`Content-Type:application/x-www-form-urlencoded`

`appId=*****&secret=*****&refreshToken=*****`

## 响应样例

response:

Status Code: 200 OK

`Content-Type: application/json`

Body:

```
{  
    "accessToken": "*****",  
    "tokenType": "*****",  
    "expiresIn": "*****",  
    "refreshToken": "*****",  
    "scope": "*****"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	100449	The device is freezed cant operate.	用户无操作权限。 处理建议：请检查appId所对应用用户是否有接口调用权限。
400	102202	Required Parameter is null or empty.	必选参数为空。 处理建议：请检查请求参数中的必选参数是否已填写。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100208	AppId or secret is not right.	appId或secret错误。 处理建议： <ul style="list-style-type: none"><li>● 请检查appId和secret是否有误，是否新增字符或缺少字符。</li><li>● 请检查请求路径中的IP地址是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 1.2.2 设备管理

应用服务器向物联网平台添加设备，获取设备的ID和验证码，待设备完成接入物联网平台流程后，设备与应用服务器建立从属关系。

### 1.2.2.1 注册直连设备

#### 典型场景

在直连设备接入物联网平台前，应用服务器需要调用此接口在物联网平台注册设备，并把设备的唯一标识码（如IMEI）设置为设备接入平台的验证码。在设备接入物联网平台时携带设备唯一标识，完成设备的接入认证。

#### 接口功能

支持应用服务器在物联网平台注册直连设备，完成注册后设备方可接入物联网平台。

#### 接口原型

请求方法	POST
请求地址	<a href="https://server:port/iocm/app/reg/v1.1.0/deviceCredentials?appId={appId}">https://server:port/iocm/app/reg/v1.1.0/deviceCredentials?appId={appId}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceInfo	可选	<a href="#">DeviceInfoD TO</a>	Body	设备信息。
endUserId	可选	String(256)	Body	终端用户ID。 在NB-IoT方案中，endUserId设置为设备的IMSI号。在Smarthome解决方案中，endUserId设置为App账号。
imsi	可选	String(1-64)	Body	NB-IoT终端的IMSI。

参数	必选/可选	类型	位置	描述
isSecure	可选	Boolean	Body	<p>指定设备是否为安全设备，默认值为false。在NB-IoT场景下，注册的设备为加密设备时，isSecure要设置为true。</p> <ul style="list-style-type: none"><li>● true: 安全设备</li><li>● false: 非安全设备</li></ul>
nodeId	必选	String(256)	Body	<p>设备唯一标识码，必须与设备上报的设备标识一致。通常使用MAC地址，Serial No或IMEI作为nodeId。</p> <p><b>说明</b></p> <p>使用IMEI作为nodeId时，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
psk	可选	String(8-32)	Body	请求中指定psk，则平台使用指定的psk；请求中不指定psk，则由平台生成psk。取值范围是a-f、A-F、0-9组成的字符串。
timeout	可选	Integer(>=0)	Body	<p>超时时间。当调用北向接口对设备开户，在超时时间内可绑定设备，若超过timeout时间且未绑定设备，则会删除超时的开户信息。</p> <p>取值范围：0~2147483647。若填写为“0”，则表示设备验证码不会失效（建议填写为“0”）。</p> <p>默认值：180(默认值可配置，具体配置值请咨询物联网平台运维人员。)</p> <p>单位：秒。</p>
verifyCode	可选	String(256)	body	<p>设备验证码，若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。</p> <p>在NB-IoT方案中，verifyCode为必填参数，且必须与nodeId设置成相同值。</p>

参数	必选/可选	类型	位置	描述
productId	可选	String(256)	Body	设备所属的产品ID。

DeviceInfoDTO:

参数	必选/可选	类型	位置	描述
manufacturerId	可选	String(256)	Body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	Body	厂商名称。
deviceType	可选	String(256)	Body	设备类型，大驼峰命名方式，如 MultiSensor、ContactSensor、CameraGateway。
model	必选	String(256)	Body	设备的型号。 Z-Wave: productType + productId, 16进制格式XXXX-XXXX 补0对齐, 如: 001A-0A12, 其他协议的格式待定。
protocolType	可选	String(256)	Body	设备使用的协议类型，当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
psk	String(32)	随机psk参数，若请求中携带了psk，则使用请求中的psk，否则由平台生成随机psk参数。
timeout	Integer	验证码有效时间，单位秒，设备需要在有效时间内接入物联网平台。

参数	类型	描述
verifyCode	String(256)	设备验证码，设备可以通过验证码完成物联网平台的接入认证。若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。

## 请求示例

Method: POST

request:

`https://server:port/iocm/app/reg/v1.1.0/deviceCredentials?appId=*****`

Header:

`"app_key: *****"`

`"Authorization:Bearer *****"`

`Content-Type:application/json`

Body:

```
{  
    "endUserId": "*****",  
    "verifyCode": "*****",  
    "nodeId": "*****",  
    "deviceInfo": {  
        "manufacturerName": "*****",  
        "manufacturerId": "*****",  
        "deviceType": "*****",  
        "model": "*****",  
        "protocolType": "*****"  
    },  
    "psk": "*****",  
    "timeout": 0  
}
```

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "deviceId": "*****",  
    "verifyCode": "*****",  
    "psk": "*****",  
    "timeout": 300  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	103028	The license pool resources.	License资源用尽。
400	100003	Invalid verify code.	验证码无效。 处理建议：请检查接口请求中的 verifyCode 是否有误。若请求参数中未带 verifyCode，请联系物联网平台维护人员处理。
400	100007	Bad request message.	参数不合法。 处理建议：deviceId未赋值，请参考请求参数说明填写请求。
400	100416	The device has already been binded.	设备已经绑定。 处理建议：请检查设备是否已经注册。
400	100426	The nodeId is duplicated.	nodeId重复。 处理建议：请检查接口请求中nodeId是否有误。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100025	AppId for auth not exist.	获取不到appId鉴权信息。 处理建议： <ul style="list-style-type: none"><li>● 请检查请求结构体的Header头域中是否给字段app_key赋值。</li><li>● 若通过HTTP调用，请联系物联网平台维护人员确认Header头域中放置appId的字段名是app_key还是x-app-key。</li></ul>

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
403	600002	The product not existed.	产品不存在。 处理建议：物联网平台未找到 productId对应的产品，请检查 productId是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100203	The application is not existed.	授权应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	100412	The amount of device has reached the limit.	当前应用下设备数量达到上限。 处理建议：请检查当前应用下的设备数量是否已达到所申请资源的上限。
500	100441	The amount of nonSecure device has reached the limit.	非安全设备数量已达到上限。

Http状态码	错误码	错误描述	说明
500	103026	The license is not exist.	License不存在。 处理建议：物联网平台内部License问题，请联系物联网平台维护人员处理。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.2.2 发现非直连设备

在网关接入物联网平台后，应用服务器可调用此接口为网关绑定非直连设备，实现非直连设备接入物联网平台。

此功能通过[设备服务调用](#)接口向网关下发命令实现，“serviceId”参数设置为“Discovery”，“method”参数设置为“DISCOVERY”时，为发现非直连设备命令。

### 1.2.2.3 刷新设备密钥

#### 典型场景

已在物联网平台注册的设备，若设备的唯一标识码发生变更（如更换设备等场景），应用服务器需要调用此接口在物联网平台更新设备的唯一标识，并重新绑定设备。



仅当设备处于离线状态时，才能进行刷新设备密钥操作。

#### 接口功能

支持应用服务器在物联网平台更新已注册设备的nodeId，在设备deviceId不变的情况下，重新绑新设备。

#### 接口原型

请求方法	PUT
请求地址	<a href="https://server:port/iocm/app/reg/v1.1.0/deviceCredentials/{deviceId}?appId={appId}">https://server:port/iocm/app/reg/v1.1.0/deviceCredentials/{deviceId}?appId={appId}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceId	必选	String(256)	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
nodeId	可选	String(256)	body	<p>设备唯一标识码，通常使用MAC地址，Serial No或IMEI作为nodeId。 若需要更新设备的nodeId，则填写此参数，否则不需要填写。</p> <p><b>说明</b> 若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>海思芯片设备的唯一标识为IMEI号</li><li>其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
timeout	可选	Integer	body	<p>超时时间。在超时时间内可绑定设备，若超过timeout时间且未绑定设备，则会删除超时的开户信息。</p> <p>取值范围：0~2147483647。若填写为“0”，则表示设备验证码不会失效（建议填写为“0”）。</p> <p>默认值：180（默认值可配置，具体配置值请咨询物联网平台运维人员。）</p> <p>单位：秒。</p>
verifyCode	可选	String(256)	body	<p>设备验证码，若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。建议和nodeId设置成一样。</p> <p>在NB-IoT方案中，verifyCode为必填参数，且必须与nodeId设置成相同值。</p>

## 响应参数

Status Code: 200 OK

参数	类型	描述
verifyCode	String(256)	设备验证码，设备可以通过验证码完成物联网平台的接入认证。若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。
timeout	Integer	验证码有效时间，单位秒，设备需要在有效时间内接入物联网平台。

## 请求示例

Method: PUT

request:

`https://server:port/iocm/app/reg/v1.1.0/deviceCredentials/**********?  
appId=*****`

Header:

`"app_key: *****"`

`"Authorization:Bearer *****"`

`Content-Type:application/json`

Body:

```
{  
    "verifyCode": "*****",  
    "nodeId": "*****",  
    "timeout": 300  
}
```

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "verifyCode": "AE10-12424-12414",  
    "timeout": 300  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
400	100003	Invalid verify code.	验证码无效。 处理建议：请检查接口请求中的verifyCode是否有误。若请求参数中未带verifyCode，请联系物联网平台维护人员处理。
400	100007	Bad request message.	参数不合法。 处理建议：deviceId未赋值，请参考请求参数说明填写请求。
400	100426	The nodeId is duplicated.	nodeId重复。 处理建议：请检查接口请求中nodeId是否有误。
400	100610	Device is not active.	设备未激活。 处理建议：请检查该设备是否已接入物联网平台并激活。
400	100611	Device is online.	设备在线。 处理建议：请下线该设备或者断开设备与物联网平台的连接。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100025	AppId for auth not exist.	获取不到appId鉴权信息。 处理建议： <ul style="list-style-type: none"><li>● 请检查请求结构体的Header头域中是否给字段app_key赋值。</li><li>● 若通过HTTP调用，请联系物联网平台维护人员确认Header头域中放置appId的字段名是app_key还是x-app-key。</li></ul>
403	100203	The application is not existed.	授权应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.2.4 修改设备信息

#### 典型场景

应用服务器在物联网平台注册设备后，若设备的基本信息发生变更，可调用此接口在物联网平台修改设备的信息。

#### 接口功能

支持应用服务器修改设备的基本信息，包括设备类型、设备模型、厂商信息、接入协议等。

#### 接口原型

请求方法	PUT
请求地址	<a href="https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?appId={appId}">https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?appId={appId}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
customFields	可选	List< <a href="#">CustomField</a> >	Body	自定义字段列表，用户可设置自定义字段。
deviceConfig	可选	<a href="#">DeviceConfigDTO</a>	body	设备配置信息。
deviceType	可选	String(1~256)	body	设备类型，大驼峰命名方式，例如：MultiSensor、ContactSensor、CameraGateway。 注册设备后必须修改设备类型，且要与profile中定义的保持一致。

参数	必选/可选	类型	位置	描述
endUser	可选	String(1~256)	body	终端用户，若为直连设备，则endUser可选；若为非直连设备，则endUser可以为null。
location	可选	String(1~1024)	body	设备位置。
manufacturerId	可选	String(1~256)	body	厂商ID，唯一标识一个厂商。 注册设备后必须修改厂商ID，且要与profile中定义的保持一致。
manufacturerName	可选	String(1~256)	body	厂商名称。
model	可选	String(1~256)	body	设备型号，由厂商定义。 注册设备后必须修改设备型号，且要与profile中定义的保持一致。
mute	可选	Enum	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>
name	可选	String(1~256)	body	设备名称。
organization	可选	String(1~256)	body	设备所属的组织信息。
protocolType	可选	String(1~256)	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。 注册设备后必须修改协议类型，且要与profile中定义的保持一致。
region	可选	String(1~256)	body	设备区域信息。
timezone	可选	String(1~256)	body	设备所在时区信息，使用时区编码，如Asia/Shanghai, America/New_York。
imsi	可选	String(1-4)	Body	NB-IoT终端的IMSI。
ip	可选	String(128)	Body	设备的IP地址。

参数	必选/可选	类型	位置	描述
isSecure	可选	Boolean	body	指定设备是否为安全设备，默认值为 false。在NB-IoT场景下，注册的设备为加密设备时，isSecure要设置为 true。 ● true: 安全设备 ● false: 非安全设备
psk	可选	String(8~32)	body	psk参数，取值范围是a-f、A-F、0-9组成的字符串。
tags	可选	List< <a href="#">Tag2</a> >	Body	设备的标签信息。

CustomField:

参数	必选/可选	类型	位置	描述
fieldName	可选	String(256)	Body	字段名字。
fieldType	可选	String(256)	Body	字段类型。
fieldValue	可选	String(256)	Body	字段值。

DeviceConfigDTO:

参数	必选/可选	字段类型	描述
dataConfig	可选	<a href="#">DataConfigDTO</a>	数据配置信息。

DataConfigDTO:

参数	必选/可选	字段类型	描述
dataAgingTime	可选	Integer	数据老化时长，取值范围：0-90，单位：天。

Tag2:

参数	必选/可选	类型	位置	描述
tagName	必选	String(1-128)	body	标签名称。
tagValue	必选	String(1-1024)	body	标签值。
tagType	可选	Integer	body	标签类型。

## 响应参数

Status Code: 204 No Content

## 请求示例

Method: PUT

request:

`https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?appId={appId}`

Body:

```
{  
    "name": "*****",  
    "endUser": "*****",  
    "mute": "*****",  
    "deviceConfig": {  
        "dataConfig": {  
            "dataAgingTime": 30  
        }  
    }  
}
```

## 响应示例

response:

Status Code: 204 No Content

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	100440	The isSecure is invalid.	isSecure参数值有误。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
403	500004	The amount of frozen devices has reached the limit.	被冻结设备已达到上限。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。

Http状态码	错误码	错误描述	说明
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	100441	The amount of nonSecure device has reached the limit.	非安全设备数量已达到上限。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.2.5 删除直连设备

#### 典型场景

已在物联网平台注册的直连设备，若不再需要接入平台时，应用服务器可调用此接口在物联网平台删除设备。后续设备若需要再次接入平台，则需重新注册。

#### 接口功能

支持应用服务器在物联网平台删除已注册的直连设备，使设备无法再接入平台。

#### 接口原型

请求方法	DELETE
请求地址	https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?appId={appId}&cascade={cascade}
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
cascade	可选	Boolean	query	仅当设备下连接了非直连设备时生效。默认值为true。 <ul style="list-style-type: none"><li>● true，级联删除，即删除直连设备和其下的非直连设备。</li><li>● false，删除直连设备，但是不删其下的非直连设备，并将非直连设备的属性变为直连设备属性。</li></ul>

## 响应参数

Status Code: 204 No Content

## 请求示例

Method: DELETE

request:

`https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?appId={appId}&cascade={cascade}`

Header:

app\_key: \*\*\*\*\*

Authorization: Bearer \*\*\*\*\*

## 响应示例

response:

Status Code: 204 No Content

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.2.6 删除非直连设备

若非直连设备不再需要接入物联网平台，应用服务器可调用此接口解绑网关下的非直连设备。

此功能通过[设备服务调用](#)接口向网关设备下发命令实现，“serviceId”参数设置为“Remove”，“method”参数设置为“REMOVE”时，为删除非直连设备命令。

### 1.2.2.7 查询设备激活状态

#### 典型场景

应用服务器通过“注册直连设备”接口在物联网平台注册设备后，在设备首次接入平台之前，设备的激活状态为false；当设备首次接入平台后，设备的激活状态为true（无论设备接入后是处于在线、离线或异常状态）。应用服务器可调用此接口查询设备在物联网平台的激活状态，确定设备是否已接入平台。

#### 接口功能

支持应用服务器通过设备ID查询设备在物联网平台的激活状态，确认设备是否已接入平台。

#### 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iocm/app/reg/v1.1.0/deviceCredentials/{deviceId}?appId={appId}">https://server:port/iocm/app/reg/v1.1.0/deviceCredentials/{deviceId}?appId={appId}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。

## 响应参数

Status Code: 200 OK

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
activated	Boolean	激活状态，设备是否通过验证码获取密码的状态标识。 <ul style="list-style-type: none"><li>● true: 已激活</li><li>● false: 未激活</li></ul>
name	String(256)	设备名称。

## 请求示例

Method: GET

request:

`https://server:port/iocm/app/reg/v1.1.0/deviceCredentials/{deviceId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer *****"`

`Content-Type:application/json;`

## 响应示例

response:

Status Code: 200 OK

`Content-Type: application/json`

Body:

```
{  
    "deviceId": "*****",  
    "activated": "*****",  
    "name": "*****"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.2.8 修改设备影子

#### 典型场景

物联网平台支持创建设备的“影子”。设备影子用于存储设备最新上报的服务属性数据，以及应用服务器对服务属性的配置（服务属性为设备Profile文件中定义的service属

性）。若设备处于离线或异常状态时，应用服务器无法通过命令下发的方式向设备下发配置，此时可把要下发的配置设置到设备影子，当设备重新上线时，设备影子会把配置下发给设备。应用服务器可调用此接口在设备影子上配置要下发给设备的配置信息。

每个设备有且只有一个设备影子，设备影子上有desired和report区。

- desired区用于存储对设备服务属性的配置。若设备当前在线，desired区的配置会即时下发给设备；若设备当前不在线，desired区的配置会在设备上线时再下发给设备。
- report区用于存储设备最新上报的服务属性数据。当设备上报数据时，平台会把数据同步到设备影子的report区。

## 接口功能

支持应用服务器配置设备影子desired区的信息，用于设备上线时把配置下发给设备。

## 注意事项

当前仅使用LWM2M协议的设备支持设备影子功能，且仅支持修改LWM2M协议定义的属性信息，用户自定义的属性暂不支持修改。

## 接口原型

请求方法	PUT
请求地址	<a href="https://server:port/iocm/app/shadow/v1.5.0/devices/{deviceId}?appId={appId}">https://server:port/iocm/app/shadow/v1.5.0/devices/{deviceId}?appId={appId}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。

参数	必选/可选	类型	位置	描述
serviceDesireds	必选	List<Service DesiredDTO>	body	需要修改的设备配置或状态信息。

ServiceDesiredDTO:

参数	必选/可选	类型	位置	描述
serviceId	可选	String(1-256 )	body	服务ID，用于标识一个服务。
desired	可选	ObjectNode	body	设备的服务属性配置。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: PUT

request:

`https://server:port/iocm/app/shadow/v1.5.0/devices/devices/{deviceId}?appId={appId}`

Header:

`app_key: *****`

`Authorization:Bearer *****`

`Content-Type:application/json`

Body:

```
{  
  "serviceDesireds": [  
    {  
      "serviceId" : "Temperature",  
      "desired" : {  
        "targetTemperature" : 35  
      }  
    }  
  ]  
}
```

{

## 响应示例

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
200	100425	The special deviceCapability is not exist.	设备模板不存在。 处理建议：请检查设备模板是否已在物联网平台上传。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	107002	The properties is empty in database.	设备属性不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107003	The request properties is unknown.	设备状态为未知。 处理建议：请检查设备与平台的连接是否正常。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	100443	The property is forbidden to write.	设备属性不可写。

Http状态码	错误码	错误描述	说明
403	10100 04	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	10100 05	pp_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	10040 3	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	10041 8	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	10002 3	The data in dataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	10020 3	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.2.9 查询设备影子

#### 典型场景

若设备处于离线或异常状态时，应用服务器无法通过命令下发的方式向设备下发配置。此时可把要下发的配置设置到设备影子，当设备重新上线时，设备影子会把配置下发给设备。应用服务器可调用此接口查看设备影子上的设备配置信息及最新的设备上报数据信息。

## 接口功能

支持应用服务器查询单个设备的设备影子信息，包括对设备的配置信息（desired区）和设备最新上报的数据信息（reported区）。

## 注意事项

当前仅使用LWM2M协议的设备支持设备影子功能，且仅支持修改LWM2M协议定义的属性信息，用户自定义的属性暂不支持修改。

## 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iocm/app/shadow/v1.5.0/devices/{deviceId}?appId={appId}">https://server:port/iocm/app/shadow/v1.5.0/devices/{deviceId}?appId={appId}</a>
传输协议	HTTPS

## 请求参数

字段	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceId	必选	String(36)	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。

## 响应参数

Status Code: 200 OK

参数	类型	描述
deviceId	String(36)	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	String(36)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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。

参数	类型	描述
nodeType	Enum	设备的类型。
createTime	String(256)	设备创建的时间。
lastModifiedTi me	String(256)	设备最后一次修改信息的时间。
deviceInfo	<b>DeviceInfo</b>	设备详细信息。
services	List< <b>DeviceServ iceB</b> >	设备的服务属性信息。

DeviceInfo:

参数	类型	描述
nodeId	String(256)	设备唯一标识码。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacturerId	String(256)	厂商ID，唯一标识一个厂商。
manufacturerN ame	String(256)	厂商名称。
mac	String(256)	设备的MAC地址。
location	String(2048)	设备的位置信息。
deviceType	String(256)	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号。 Z-Wave: productType + productId, 16进制格式XXXX-XXXX补0对齐, 如: 001A-0A12, 其他协议的格式待定。
swVersion	String(256)	设备的软件版本。 Z-Wave: 主版本号.次版本号, 如: 1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
protocolType	String(256)	设备使用的协议类型, 当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	String(256)	Bridge标识, 表示设备通过哪个Bridge接入物联网平台。

参数	类型	描述
status	String	设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。 若设备超过25（默认值）小时未向平台上报数据或发送消息，则设备状态为ABNORMAL（默认值）；若设备超过49小时未向平台上报数据或发送消息，则设备状态为OFFLINE。
statusDetail	String(256)	设备的状态详情，跟status取值对应。 <ul style="list-style-type: none"> <li>● status为ONLINE时，取值NONE COMMUNICATION_ERROR, CONFIGURATION_ERROR BRIDGE_OFFLINE或 FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE</li> <li>● status为OFFLINE时，取值NONE CONFIGURATION_PENDING</li> </ul>
mute	String	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"> <li>● TRUE: 冻结状态</li> <li>● FALSE: 非冻结状态</li> </ul>
supportedSecurity	String	表示设备是否支持安全模式。 <ul style="list-style-type: none"> <li>● TRUE: 支持安全模式</li> <li>● FALSE: 不支持安全模式</li> </ul>
isSecurity	String	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"> <li>● TRUE: 启用</li> <li>● FALSE: 未启用</li> </ul>
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。
batteryLevel	String(256)	设备的电池电量。

 **说明**

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceServiceB:

参数	类型	描述
serviceId	String(256)	服务ID，用于标识一个服务。

参数	类型	描述
reportedProps	ObjectNode	设备最新上报的数据。
desiredProps	ObjectNode	要向设备下发的配置信息。
eventTime	String(256)	事件发生的时间。
serviceType	String(256)	服务类型。

## 请求示例

Method: GET

request:

`https://server:port/iocm/app/shadow/v1.5.0/devices/{deviceId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer *****"`

`Content-Type:application/json;`

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "nodeType": "*****",  
    "createTime": "*****",  
    "lastModifiedTime": "*****",  
    "deviceInfo": "*****"  
    "services": [  
        {  
            "serviceId": "*****",  
            "reportedProps": "*****",  
            "desiredProps": "*****",  
            "eventTime": "*****",  
            "serviceType": "*****"  
        }  
    ]  
}
```

```
"serviceType": "*****"  
,  
]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的 appId 是否有误。</li><li>请检查请求路径（url）中传入的 appId 是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	pp_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数 appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的 appId 是否有误。</li><li>请检查请求路径（url）中传入的 appId 是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.2.10 查询设备实时位置信息

#### 典型场景

应用服务器若需要获取设备当前最新的位置信息时，可调用此接口查询设备的实时位置信息，当前仅支持查询NB-IoT设备的实时位置信息。

#### 接口功能

物联网平台通过和位置服务器对接，向应用服务器提供查询NB-IoT设备实时位置信息的能力。

##### 注意

根据位置服务器的能力和配置情况，查询设备实时位置信息接口会存在较大的时延，而且并发能力有限。具体情况建议咨询物联网平台维护人员。

#### 接口原型

请求方法	POST
请求地址	<a href="https://server:port/iocm/app/location/v1.1.0/queryDeviceRealtimeLocation?appId={appId}">https://server:port/iocm/app/location/v1.1.0/queryDeviceRealtimeLocation?appId={appId}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。

参数	必选/可选	类型	位置	描述
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceId	必选	String(256)	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
horAcc	必选	Integer[0, 1800000]	body	水平误差，单位：米，如果不携带，则默认1000米。
geoInfo	可选	GeoInfoDTO	body	地理坐标信息，默认WGS84，其他坐标暂不支持。

GeoInfoDTO:

参数	必选/可选	类型	位置	描述
coordinateReferenceSystem	必选	CoordinateReferenceSystemDTO	body	坐标参考系统。

CoordinateReferenceSystemDTO:

参数	必选/可选	类型	位置	描述
identifier	必选	IdentifierDTO	body	坐标参考系统标识。

IdentifierDTO:

参数	必选/可选	类型	位置	描述
code	必选	String	body	坐标参考系统编号，默认4326，当前只支持这个WGS84数字坐标。
codeSpace	必选	String	body	编码空间，当前只支持EPSG。
edition	必选	String	body	编辑版本，当前只支持6.1。

## 响应参数

Status Code: 200 OK

参数	类型	描述
deviceId	String	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
pd	PdIoMDTO	定位结果，定位成功时携带。
poserr	PoserrIoMDTO	定位错误信息，定位失败时携。

PdIoMDTO:

参数	类型	描述
time	String	定位时间，格式“yyyyMMddhhmmss”。
utcOff	String	时间时区，格式“hhmm”。
srsName	String	坐标参考，固定取值：“www.epsg.org#4326”。
X	String	X轴，纬度，格式：“DD MM SS.hhhN”或“DD MM SS.hhhS”，最后的N指北纬，S指南纬。
Y	String	Y轴，经度，格式：“DD MM SS.hhhE”或“DD MM SS.hhhW”，最后的E指东经，W指西经。
radius	String	误差半径，单位：米。

PoserrIoMDTO:

参数	类型	描述
time	String	定位时间，格式“yyyyMMddhhmmss”。
resid	String	定位结果值。
add_info	String	定位详细结果。

## 请求示例

Method: POST

request:

`https://server:port/iocm/app/location/v1.1.0/queryDeviceRealtimeLocation?appId={appId}`

Content-Type:application/json

Header:

```
"app_key": "*****"  
"Authorization:Bearer *****"  
  
Body:  
  
{  
    "deviceId": "123",  
    "horAcc": "1000",  
    "geoInfo": {  
        "coordinateReferenceSystem": {  
            "identifier": {  
                "code": "4326",  
                "codeSpace": "EPSG",  
                "edition": "6.1"  
            }  
        }  
    }  
}
```

响应示例

## response

Status Code: 200 OK

Content-Type: application/json

### Body:

{

"deviceId": "\*\*\*\*\*",

"pd": {

"time": "\*\*\*",

"utcOff": "\*\*\*",

"srsName": "www.epsg.org#4326",

"X": "\*\*\*",

"Y". "\*\*\*"

"radius": "\*\*\*"

3

"poserr": {

"time": "\*\*\*"

```
"resid": "2",  
"add_info": "***"  
}  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	<p>设备信息不存在。</p> <p>处理建议:</p> <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。 请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	50400	The input is invalid.	<p>输入参数无效。</p> <p>处理建议: 请检查接口调用请求中携带参数的合法性。</p>
403	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议:</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	<p>应用未被授权。</p> <p>处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。</p>
403	1010004	app throttle exceed.	<p>应用调用接口过于频繁, 超过流控值(默认值是100次/60s)。</p> <p>处理建议: 联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。</p>
403	1010005	App_key or access_token is invalid.	<p>错误的token信息。</p> <p>处理建议: 请检查接口请求中所携带的accessToken是否有误。</p>

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	103801	Exceed the system concurrency restriction.	超过系统并发限制。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 1.2.3 批量处理

应用服务器通过批量处理接口，对已接入物联网平台的设备进行批量操作。

### 1.2.3.1 创建批量任务

#### 典型场景

应用服务器若需要对多个设备进行批量操作时，可调用此接口创建设备的批量操作任务，当前仅支持给批量设备下发缓存命令。

#### 接口功能

支持应用服务器在物联网平台创建设备的批量操作任务。

#### 接口原型

请求方法	POST
请求地址	<a href="https://server:port/iocm/app/batchtask/v1.1.0/tasks">https://server:port/iocm/app/batchtask/v1.1.0/tasks</a>
传输协议	HTTPS

## 请求参数

参数	必选/ 可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	必选	String(6-4)	body	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
param	必选	Object Node	body	任务详细参数，根据taskType任务类型的不同对应不同类型参数。
taskName	必选	String	body	任务名称，最大长度256字符。
taskType	必选	String	body	任务类型，取值范围：DeviceCmd。
timeout	必选	Integer	body	任务超时时长，单位分钟，范围为10到2880。
tags	可选	List<TagDTO2>	body	标签列表。

ObjectNode:

参数	必选/ 可选	类型	位置	描述
type	必选	String	body	批量命令类型，取值范围：DeviceList/DeviceType/DeviceArea/GroupList/Broadcast/GroupIdList。
deviceList	条件必选	List<String>	body	设备ID列表，type为DeviceList时需要填写。
deviceType	条件必选	String	body	设备类型，type为DeviceType时需要填写，其值应当与profile中定义的一致。
manufacturerId	条件可选	String	body	厂商ID，type为DeviceType时可填写，其值应当与profile中定义的一致。
model	条件可选	String	body	设备型号，type为DeviceType时可填写，其值应当与profile中定义的一致。

参数	必选/可选	类型	位置	描述
deviceLocation	条件必选	String	body	设备位置, type为DeviceArea时需要填写。
groupList	条件必选	List<String>	body	群组Id列表或设备组名称列表, type为GroupIdList时需要填写群组Id, type为GroupList时需要填写设备组名称。
command	必选	CommandDTO	body	命令信息。
callbackUrl	可选	String	body	命令执行结果的推送地址。
maxRetransmit	可选	Integer(0~3)	body	命令下发最大重传次数, 取值范围: 0-3。
groupTag	可选	String	body	群组标签。

CommandDTO:

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-4)	body	命令对应的服务ID, 要与profile中定义的serviceId保持一致。
method	必选	String(1-28)	body	服务下具体的命令名称, 要与profile中定义的命令名保持一致。
paras	可选	ObjectNode	body	命令参数, jsonString格式, 里面是一个个键值对, 每个键都是profile中命令的参数名 (paraName), 具体格式需要应用和设备约定。

TagDTO2:

参数	必选/可选	类型	位置	描述
tagName	必选	String(1-128)	body	标签名称。
tagValue	必选	String(1-1024)	body	标签值。

## 响应参数

Status Code: 200 OK

参数	类型	描述
taskID	String	批量任务的ID。

## 请求示例

Method: POST

request:

<https://server:port/iocm/app/batchtask/v1.1.0/tasks>

Content-Type:application/json

Body:

```
{  
    "appId": "*****",  
    "timeout": 1000,  
    "taskName": "*****",  
    "taskType": "DeviceCmd",  
    "param": {  
        "type": "*****",  
        "deviceList": [  
            "*****",  
            "*****",  
            "*****",  
        ]  
        "command": {  
            "serviceId": "*****",  
            "method": "*****",  
            "paras": {  
                "paraName1": "paraValue1",  
                "paraName2": "paraValue2"  
            }  
        },  
    }  
}
```

{

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

{  
  "taskID": "\*\*\*\*\*"  
}

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
200	101001	Resource doesn't exist.	资源不存在。
200	105001	The batchTask count has reached the limit.	未完成的任务数大于等于10个时，返回任务达到数量限制。
200	105002	The batchTask name has exist.	任务名字已存在。 处理建议：修改任务名称。
400	105201	The tagName and tagValue has been used on the platform.	平台已使用tagName和tagValue。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100028	The user has no right.	用户没有操作权限。
403	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105202	The tag is not existed.	标签不存在。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.3.2 查询指定批量任务信息

#### 典型场景

应用服务器通过“创建批量任务”接口创建了设备的批量操作任务后，可调用此接口查询该批量任务的信息，包括任务的当前状态、子任务（指批量任务中一个设备执行的任务）完成情况等。

#### 接口功能

支持应用服务器通过任务ID查询单个批量任务的信息。

#### 接口原型

请求方法	GET
请求地址	https://server:port/iocm/app/batchtask/v1.1.0/tasks/{taskId}?appId={appId}&select={select}

传输协议	HTTPS
------	-------

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
taskId	必选	String	path	批量任务ID，创建批量任务后获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
select	可选	String	query	指定可选的返回值，可取值：tag。

## 响应参数

Status Code: 200 OK

参数	类型	描述
appId	String	批量任务所归属的appId。
taskId	String	批量任务ID。
taskName	String	批量任务名称。
operator	String	下发该批量任务的操作员。
taskFrom	String	批量任务的来源。 <ul style="list-style-type: none"><li>● Portal: 通过SP Portal创建的。</li><li>● Northbound: 调用北向API接口创建的。</li></ul>
taskType	String	批量任务的类型，取值范围：DeviceCmd。
status	String	批量任务的状态，取值范围：Pending/Running/Complete/Timeout。
startTime	String	批量任务的创建时间。
timeout	Integer	批量任务的超时时间，单位秒。

参数	类型	描述
progress	Integer	批量任务的进度，单位：千分比，范围：0-1000，向下取整。
totalCnt	Integer	任务的总数。
successCnt	Integer	成功的任务数。
failCnt	Integer	失败的任务数。
timeoutCnt	Integer	超时的任务数。
expiredCnt	Integer	未执行的失效任务数。
completeCnt	Integer	完成的任务数，完成的任务数=成功的任务数+失败的任务数+超时的任务数。
successRate	Integer	任务成功率，单位：千分比，范围：0-1000，向下取整。
param	ObjectNode	不同任务类型的具体参数。
tags	List<TagDT O2>	批量任务的标签列表。

ObjectNode:

参数	类型	描述
type	String	批量命令类型，取值范围：DeviceList/DeviceType/ DeviceArea/GroupList/Broadcast/GroupIdList。
deviceList	List<String>	设备ID列表，type为DeviceList时返回值。
deviceType	String	设备类型，type为DeviceType时返回值，其值应当与 profile中定义的一致。
manufacturer Id	String	厂商ID，type为DeviceType时返回值，其值应当与 profile中定义的一致。
model	String	设备型号，type为DeviceType时返回值，其值应当与 profile中定义的一致。
deviceLocation	String	设备位置，type为DeviceArea时返回值。
groupList	List<String>	群组名称列表，type为GroupList时返回值。
command	CommandD TO	命令信息。
callbackUrl	String	命令执行结果的推送地址。
maxRetransmit	Integer(0~3)	命令下发最大重传次数，取值范围：0-3。

参数	类型	描述
groupTag	String	群组标签。

CommandDTO:

参数	类型	描述
serviceId	String(1-64)	命令对应的服务ID，要与profile中定义的serviceId保持一致。
method	String(1-128)	服务下具体的命令名称，要与profile中定义的命令名保持一致。
paras	ObjectNode	命令参数，jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。

TagDTO2:

参数	类型	描述
tagName	String(1-128 )	标签名称。
tagValue	String(1-102 4)	标签值。

## 请求示例

Method: GET

request:

`https://server:port/iocm/app/batchtask/v1.1.0/tasks/********?appId=*****`

Header:

`"app_key: *****"`

`"Authorization:Bearer *****"`

`Content-Type:application/json;`

## 响应示例

response:

Status Code: 200 OK

`Content-Type: application/json`

Body:

```
{  
    "taskId" : "*****",  
    "taskName" : "*****",  
    "appId" : "*****",  
    "operator" : "*****",  
    "taskFrom" : "*****",  
    "taskType" : "*****",  
    "status" : "*****",  
    "startTime" : "*****",  
    "timeout" : 1000,  
    "progress" : 100,  
    "totalCnt" : 100,  
    "successCnt" : 70,  
    "failCnt" : 10,  
    "timeoutCnt" : 10,  
    "expiredCnt" : 10,  
    "completeCnt" : 100,  
    "successRate" : 70,  
    "param" : {  
        "fileId" : "*****"  
    }  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	100019	Illegal request.	非法请求。 处理建议：请检查接口请求中的必须参数是否为空。
400	100022	The input is invalid	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105005	The batchTask is not existed.	批量任务不存在。 处理建议：请检查接口请求中的taskId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

### 1.2.3.3 查询批量任务的子任务信息

#### 典型场景

应用服务器通过“创建批量任务”接口创建了设备的批量操作任务后，可调用此接口查询该批量任务中子任务的详情信息，如任务的执行状态、任务内容等。

#### 接口功能

支持应用服务器根据条件查询批量任务中子任务的详情信息。

## 接口原型

请求方法	GET
请求地址	https://server:port/iocm/app/batchtask/v1.1.0/taskDetails? appId={appId}&taskId={taskId}&status={status}&index={index}&de viceId={deviceID}&commandId={commandId}&pageNo={pageNo}& pageSize={pageSize}
传输协议	HTTP/HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
taskId	必选	String	query	批量任务的ID。
status	可选	String	query	任务的详情状态，Pending/WaitResult/Success/Fail/Timeout。
device_id	可选	String	query	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
commandId	可选	String	query	设备命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。
pageNo	可选	Integer	query	分页查询参数。 ● 值为空时，查询内容不分页。 ● 值为大于等于0的整数时，分页查询。 ● 值为0时查询第一页。
pageSize	可选	Integer	query	分页查询参数，取值大于等于1的整数，缺省值：1。

## 响应参数

Status Code: 200 OK

参数	类型	描述
pageNo	Long	分页查询参数。 ● 值为空时，查询内容不分页。 ● 值为大于等于0的整数时，分页查询。 ● 值为0时查询第一页。
pageSize	Long	分页查询参数，取值大于等于1的整数，缺省值：1。
totalCount	Long	查询到的任务总数。
taskDetails	List<QueryTaskDetailDTOCloud2NA>	任务详情信息列表。

QueryTaskDetailDTOCloud2NA:

参数	类型	描述
status	String	任务执行状态，取值范围：Pending/WaitResult/Success/Fail/Timeout。
output	String	批量下发命令的输出信息。
error	String	任务的错误原因，格式为：{"error_code": "*****", "error_desc": "*****"}。
param	ObjectNode	不同任务类型的具体参数。

ObjectNode:

参数	类型	描述
deviceId	String	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
commandId	String	下发的命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。

## 请求示例

Method: GET

request:

`https://server:port/iocm/app/batchtask/v1.1.0/taskDetails?  
appId=*****&taskId=*****&status=*****&pageNo=*****&pageSize=*****`

Header:

```
"app_key: *****"  
"Authorization:Bearer *****"  
Content-Type:application/json;
```

## 响应示例

response:

Status Code: 200 OK  
Content-Type: application/json

Body:

```
{  
    "totalCount": 1,  
    "pageNo": 0,  
    "pageSize": 1,  
    "taskDetails": [  
        {  
            "status": "WaitResult",  
            "output": "{\"requestId\":\"*****\",\"commandResult\":null}",  
            "error": null,  
            "param": {  
                "deviceId": "*****",  
                "commandId": "*****"  
            }  
        },  
        {  
            "status": "WaitResult",  
            "output": "{\"requestId\":\"*****\",\"commandResult\":null}",  
            "error": null,  
            "param": {  
                "deviceId": "*****",  
                "commandId": "*****"  
            }  
        }  
    ]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105005	The batchTask is not existed.	批量任务不存在。 处理建议：请检查接口请求中的 taskId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 1.2.4 订阅管理

物联网平台支持应用服务器对设备数据的订阅，若订阅的设备信息有变更，平台会推送给应用服务器。消息订阅接口需要与[消息推送](#)接口配合使用。

### 1.2.4.1 订阅平台业务数据

#### 典型场景

应用服务器可调用此接口在物联网平台订阅设备相关的业务信息，当物联网平台中设备的业务信息发生变化时（如设备注册、设备数据上报、设备状态变更等），平台会向应用服务器发送通知消息。

#### 接口功能

支持应用服务器向物联网平台订阅设备的业务变更信息，当设备的状态、数据等信息发生变化时，平台会向应用服务器发送通知消息。

#### 接口原型

请求方法	POST
请求地址	<a href="https://server:port/iocm/app/sub/v1.2.0/subscriptions?ownerFlag={ownerFlag}">https://server:port/iocm/app/sub/v1.2.0/subscriptions?ownerFlag={ownerFlag}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
ownerFlag	可选	String(256)	query	callbackUrl的所有者标识。 <ul style="list-style-type: none"><li>● ownerflag为false时，表示callbackUrl的owner是授权应用。</li><li>● ownerflag为true时，表示callbackUrl的owner为被授权应用。</li></ul>
appId	可选	String(256)	body	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。

参数	必选/ 可选	类型	位置	描述
notifyType	必选	String	body	<p>通知类型，应用可以根据通知类型接收物联网平台推送的对应通知消息。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备，订阅后推送绑定设备通知）</li><li>● deviceAdded（添加新设备，订阅后推送注册设备通知）</li><li>● deviceInfoChanged（设备信息变化，订阅后推送设备信息变化通知）</li><li>● deviceDataChanged（设备数据变化，订阅后推送设备数据变化通知）</li><li>● deviceDatasChanged（设备数据批量变化，订阅后推送批量设备数据变化通知）</li><li>● deviceDeleted（删除设备，订阅后推送删除设备通知）</li><li>● messageConfirm（消息确认，订阅后推送设备消息确认通知）</li><li>● commandRsp（命令响应，订阅后推送设备命令响应通知）</li><li>● deviceEvent（设备事件，订阅后推送设备事件通知）</li><li>● serviceInfoChanged（服务信息变化，订阅后推送设备服务信息变化通知）</li><li>● deviceModelAdded（添加设备模型，订阅后推送增加设备模型通知）</li><li>● deviceModelDeleted（删除设备模型，订阅后推送删除设备模型通知）</li><li>● deviceDesiredPropertiesModifyStatusChanged（设备影子状态变更，订阅后推送设备影子状态变更通知）</li></ul>
callbackUrl	必选	String(1024)	body	<p>订阅的回调地址，用于接收对应类型的通知消息。</p> <p>可使用HTTPS或HTTP回调地址，建议使用HTTPS。同时回调地址中必须指定回调地址的端口。例如：<a href="https://XXX.XXX.XXX.XXX:443/">https://XXX.XXX.XXX.XXX:443/</a> callbackurltest</p>
channel	可选	String(32)	Body	传输通道，若是MQTT客户端订阅，则取值为MQTT，其他情况为HTTP。

## 响应参数

Status Code: 201 Created

参数	类型	描述
subscriptionId	String	订阅ID，用于唯一标识一个订阅，在创建订阅时由物联网平台分配获得。
notifyType	String	通知的类型。
callbackUrl	String	订阅的回调地址。
clientIds	List<String>	MQTT客户端ID，只有MQTT订阅时该字段有返回值。

## 请求示例

Method: POST

request: https://server:port/iocm/app/sub/v1.2.0/subscriptions

Header:

"app\_key : \*\*\*\*\*"

"Authorization: Bearer \*\*\*\*\*"

Content-Type:application/json

Body:

```
{  
    "appId": "*****",  
    "notifyType": "deviceInfoChanged",  
    "callbackUrl": "https://*****"  
}
```

## 响应示例

response:

Status Code: 201 Created

Body:

```
{  
    "subscriptionId": "*****",  
    "notifyType": "*****",  
    "callbackUrl": "https://*****"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	100222	The request callbackurl is illegal.	回调地址非法。 处理建议：请检查请求体中的回调地址是否有误。
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
409	100227	The resource is conflicted.	资源冲突，通知类型已经被订阅。 处理建议：请检查要订阅的通知类型是否已被订阅。

### 1.2.4.2 订阅平台管理数据

#### 典型场景

应用服务器可调用此接口向物联网平台订阅设备相关的管理数据，当物联网平台中设备进行版本升级等相关操作时，平台会向应用服务器发送通知消息，知会其具体的操作状态或操作结果信息。

#### 接口功能

支持应用服务器向物联网平台订阅设备的升级操作信息，当设备进行版本升级等相关操作时，平台会向应用服务器发送通知消息。

#### 接口原型

请求方法	POST
请求地址	https://server:port/iodm/app/sub/v1.1.0/subscribe
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
notifyType	必选	String	body	通知类型。 <ul style="list-style-type: none"><li>● swUpgradeStateChangeNotify（软件升级状态变更通知，订阅后推送软件升级状态变更通知）</li><li>● swUpgradeResultNotify（软件升级结果通知，订阅后推送软件升级结果通知）</li><li>● fwUpgradeStateChangeNotify（固件升级状态变更通知，订阅后推送固件升级状态变更通知）</li><li>● fwUpgradeResultNotify（固件升级结果通知，订阅后推送固件升级结果通知）</li></ul>
callbackUrl	必选	String	body	订阅的回调地址，用于接收对应类型的的通知消息。 可使用HTTPS或HTTP回调地址，建议使用HTTPS。同时回调地址中必须指定回调地址的端口。例如：https://XXX.XXX.XXX.XXX:443/callbackurltest。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: https://server:port/iodm/app/sub/v1.1.0/subscribe

Header:

"app\_key: \*\*\*\*\*"

"Authorization:Bearer\*\*\*\*\*"

Content-Type:application/json

Body:

{

```
"notifyType": "swUpgradeStateChangeNotify",
"callbackUrl": "http://*****"
}
```

## 响应示例

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
400	100222	Internal server error.	回调地址非法。 处理建议：请检查请求体中的回调地址是否有误。
400	100228	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100229	Get AppKey from header failed.	从消息头获取AppKey失败。
500	100244	register out route fail.	注册路由失败。 处理建议：联系物联网平台维护人员。

### 1.2.4.3 查询单个订阅

#### 典型场景

应用服务器可在物联网平台上创建多个订阅，以订阅不同类型的设备变更通知。应用服务器若需要查看某个订阅的配置信息，可调用此接口进行查询。

#### 接口功能

支持应用服务器在物联网平台根据订阅ID查询指定订阅的配置信息。

#### 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iocm/app/sub/v1.2.0/subscriptions/{subscriptionId}?appId={appId}">https://server:port/iocm/app/sub/v1.2.0/subscriptions/{subscriptionId}?appId={appId}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
subscriptionId	必选	String(256)	path	订阅ID，用于唯一标识一个订阅，在创建订阅时由物联网平台分配获得。
appId	可选	String(256)	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。

#### 响应参数

Status Code: 200 OK

参数	类型	描述
subscriptionId	String	订阅ID，用于唯一标识一个订阅，在创建订阅时由物联网平台分配获得。
notifyType	String	通知的类型。

参数	类型	描述
callbackUrl	String	订阅的回调地址。

## 请求示例

Method: GET

request: https://server:port/iocm/app/sub/v1.2.0/subscriptions/{subscriptionId}?  
appId={appId}

Header:

"app\_key: \*\*\*\*\*"

"Authorization:Bearer\*\*\*\*\*"

Content-Type:application/json

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "subscriptionId": "*****",  
    "notifyType": "*****",  
    "callbackUrl": "*****"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 1.2.4.4 批量查询订阅

#### 典型场景

应用服务器可在物联网平台上创建多个订阅，以订阅不同类型的设备变更通知。应用服务器若需要查看本应用下所有订阅配置，或者查看某种订阅类型下的所有订阅配置，可调用此接口进行查询。

#### 接口功能

支持应用服务器在物联网平台查询本应用下的所有订阅信息，或者某种订阅类型下的所有订阅信息。

#### 接口原型

请求方法	GET
请求地址	<code>https://server:port/iocm/app/sub/v1.2.0/subscriptions? appId={appId}&amp;notifyType={notifyType}&amp;pageNo={pageNo}&amp;pageSize={pageSize}</code>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“ <code>Bearer {accessToken}</code> ”，其中 <code>{accessToken}</code> 为调用鉴权接口返回的accessToken。
appId	可选	String(256)	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。

参数	必选/可选	类型	位置	描述
notifyType	可选	String(256)	query	<p>通知类型，应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备）</li><li>● deviceAdded（添加新设备）</li><li>● deviceInfoChanged（设备信息变化）</li><li>● deviceDataChanged（设备数据变化）</li><li>● deviceDatasChanged（设备数据批量变化）</li><li>● deviceDeleted（删除设备）</li><li>● messageConfirm（消息确认）</li><li>● commandRsp（命令响应）</li><li>● deviceEvent（设备事件）</li><li>● serviceInfoChanged（服务信息变化）</li><li>● deviceModelAdded（添加设备模型）</li><li>● deviceModelDeleted（删除设备模型）</li><li>● deviceDesiredPropertiesModifyStatusChanged（修改设备影子状态变更）</li><li>● swUpgradeStateChangeNotify（软件升级状态变更通知）</li><li>● swUpgradeResultNotify（软件升级结果通知）</li><li>● fwUpgradeStateChangeNotify（固件升级状态变更通知）</li><li>● fwUpgradeResultNotify（固件升级结果通知）</li></ul>
pageNo	可选	Integer	query	<p>分页查询参数。</p> <ul style="list-style-type: none"><li>● 值为空时，查询内容不分页</li><li>● 值大于等于0的时分页查询</li><li>● 值等于0时查询第一页</li></ul>
pageSize	可选	Integer	query	分页查询参数，取值大于等于1的整数，缺省值：10。

## 响应参数

Status Code: 200 OK

参数	类型	描述
totalCount	Long(>=0)	查询到的订阅记录总数。
pageNo	Long(>=0)	查询的页码。
pageSize	Long(>=0)	查询每页信息的数量。
subscriptions	List< <a href="#">QuerySubscriptionDTOCloud2NA</a> >	订阅信息列表。

QuerySubscriptionDTOCloud2NA:

参数	类型	描述
subscriptionId	String	订阅ID号，用于唯一标识一个订阅，在创建订阅时由物联网平台分配获得。
notifyType	String	通知类型。
callbackUrl	String	订阅的回调地址。

## 请求示例

Method: GET

request:[https://server:port/iocm/app/sub/v1.2.0/subscriptions?  
appId={appId}&notifyType={notifyType}&pageNo={pageNo}&pageSize={pageSize}](https://server:port/iocm/app/sub/v1.2.0/subscriptions?appId={appId}&notifyType={notifyType}&pageNo={pageNo}&pageSize={pageSize})

Header:

"app\_key: \*\*\*\*\*"  
"Authorization: Bearer\*\*\*\*\*"  
Content-Type: application/json

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "pageNo": 0,  
    "pageSize": 100,  
    "totalCount": 10,
```

```
"subscriptions": [  
    {  
        "subscriptionId": "*****",  
        "notifyType": "*****",  
        "callbackUrl": "*****"  
    }  
]
```

## 错误码

Http状态码	错误码	错误描述	说明
400	100224	The resource exceeds 1000, please refinement query conditions.	查到端资源超出1000条，请重新设定查询条件。 处理建议：请缩小查询条件范围。
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 1.2.4.5 删除单个订阅

#### 典型场景

应用服务器若不再需要接收物联网平台推送的某个订阅通知消息时，可调用此接口删除指定的订阅配置，取消订阅。

#### 接口功能

支持应用服务器在物联网平台根据订阅ID删除指定的订阅配置。

## 接口原型

请求方法	DELETE
请求地址	<a href="https://server:port/iocm/app/sub/v1.2.0/subscriptions/{subscriptionId}?appId={appId}">https://server:port/iocm/app/sub/v1.2.0/subscriptions/{subscriptionId}?appId={appId}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
subscriptionId	必选	String(2-56)	path	订阅ID号，用于唯一标识一个订阅，在创建订阅时由物联网平台分配获得。
appId	可选	String(2-56)	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。

## 响应参数

Status Code: 204 No Content

## 请求示例

Method: DELETE

request: <https://server:port/iocm/app/sub/v1.2.0/subscriptions/{subscriptionId}?appId={appId}>

Header:

"app\_key: \*\*\*\*\*"

"Authorization: Bearer\*\*\*\*\*"

Content-Type:application/json

## 响应示例

response:

Status Code: 204 No Content

## 错误码

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100225	The resource is not found	找不到资源。 处理建议：请检查订阅ID是否有误。

### 1.2.4.6 批量删除订阅

#### 典型场景

应用服务器若不再需要接收物联网平台推送的订阅通知消息，或者不再需要接收某类订阅通知消息，可调用此接口批量删除订阅配置，取消订阅。

#### 接口功能

支持应用服务器在物联网平台批量删除订阅配置，可删除全部订阅，或删除指定订阅类型的所有订阅，或删除指定回调URL地址的所有订阅。

#### 接口原型

请求方法	DELETE
请求地址	<a href="https://server:port/iocm/app/sub/v1.2.0/subscriptions?appId={appId}&amp;notifyType={notifyType}&amp;callbackUrl={callbackUrl}&amp;channel={channel}">https://server:port/iocm/app/sub/v1.2.0/subscriptions? appId={appId}&amp;notifyType={notifyType}&amp;callbackUrl={callbackUrl}&amp;channel={channel}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	可选	String(256 )	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
notifyType	可选	String(256 )	query	<p>通知类型，应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备）</li><li>● deviceAdded（添加新设备）</li><li>● deviceInfoChanged（设备信息变化）</li><li>● deviceDataChanged（设备数据变化）</li><li>● deviceDatasChanged（设备数据批量变化）</li><li>● deviceDeleted（删除设备）</li><li>● messageConfirm（消息确认）</li><li>● commandRsp（命令响应）</li><li>● deviceEvent（设备事件）</li><li>● serviceInfoChanged（服务信息变化）</li><li>● deviceModelAdded（添加设备模型）</li><li>● deviceModelDeleted（删除设备模型）</li><li>● deviceDesiredPropertiesModifyStatus Changed（修改设备影子状态变更）</li><li>● swUpgradeStateChangeNotify（软件升级状态变更通知）</li><li>● swUpgradeResultNotify（软件升级结果通知）</li><li>● fwUpgradeStateChangeNotify（固件升级状态变更通知）</li><li>● fwUpgradeResultNotify（固件升级结果通知）</li></ul>

参数	必选/可选	类型	位置	描述
callbackUrl	可选	String(256 )	query	订阅回调的URL地址。
channel	可选	String(32)	query	传输通道。 <ul style="list-style-type: none"><li>当channel为空时，删除“传输通道”为“MQTTS”和“HTTP”的订阅。</li><li>当channel为“MQTTS”时，删除“传输通道”为“MQTTS”的订阅。</li><li>当channel为“HTTP”时，删除“传输通道”为“HTTP”的订阅。</li><li>当channel取其他值时，不会删除订阅。</li></ul>

## 响应参数

Status Code: 204 No Content

## 请求示例

Method: DELETE

request: https://server:port/iocm/app/sub/v1.2.0/subscriptions?  
appId={appId}&notifyType={notifyType}&callbackUrl={callbackUrl}&channel={channel}

Header:

"app\_key: \*\*\*\*\*"

"Authorization:Bearer\*\*\*\*\*"

Content-Type:application/json

## 响应示例

response:

Status Code: 204 No Content

## 错误码

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100225	The resource is not found	找不到资源。 处理建议：请检查订阅类型是否有误。

## 1.2.5 消息推送

应用服务器向物联网平台订阅设备信息，当设备信息发生变更时，物联网平台向应用服务器推送此消息，应用服务器根据通知类型对消息分派处理。需要配合[消息订阅](#)接口使用。

### 1.2.5.1 注册设备通知

#### 典型场景

应用服务器在物联网平台订阅了注册设备通知后（订阅的通知类型为deviceAdded），当应用服务器通过“注册直连设备”接口在平台注册设备时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了注册设备通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceAdded。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	可选	String	body	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	body	设备类型。 <ul style="list-style-type: none"><li>● ENDPOINT</li><li>● GATEWAY</li><li>● UNKNOWN</li></ul>
deviceInfo	必选	DeviceInfo	body	设备信息，具体参见 <a href="#">DeviceInfo结构体</a> 。

DeviceInfo结构体说明：

参数	必选/可选	类型	位置	描述
nodeId	必选	String(256)	body	设备唯一标识码，通常使用MAC地址，Serial No或IMEI作为nodeId。 <b>说明</b> 若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。 <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
name	可选	String(256)	body	设备名称。
description	可选	String(2048)	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。

参数	必选/ 可选	类型	位置	描述
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048 )	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型，大驼峰命名方式，如 MultiSensor、ContactSensor、 CameraGateway。
model	可选	String(256)	body	设备的型号。 Z-Wave: productType + productId, 16进制格式XXXX-XXXX补0对齐，如：001A-0A12，其他协议的格式待定。
swVersion	可选	String(256)	body	设备的软件版本。 Z-Wave: 主版本号.次版本号，如：1.1。
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	可选	String(256)	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
status	可选	String	body	设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。 若设备超过25（默认值）小时未向平台上报数据或发送消息，则设备状态为ABNORMAL（默认值）；若设备超过49小时未向平台上报数据或发送消息，则设备状态为OFFLINE。
statusDetail	可选	String(256)	body	设备的状态详情，跟status取值对应。 <ul style="list-style-type: none"><li>● status为ONLINE时，取值NONE COMMUNICATION_ERROR, CONFIGURATION_ERROR BRIDGE_OFFLINE或 FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE</li><li>● status为OFFLINE时，取值NONE CONFIGURATION_PENDING</li></ul>

参数	必选/ 可选	类型	位置	描述
mute	可选	String	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 ● TRUE: 冻结状态 ● FALSE: 非冻结状态
supporte dSecurity	可选	String	body	表示设备是否支持安全模式。 ● TRUE: 支持安全模式 ● FALSE: 不支持安全模式
isSecurit y	可选	String	body	表示设备当前是否启用安全模式。 ● TRUE: 启用 ● FALSE: 未启用
signalStr ength	可选	String(256)	body	设备的信号强度。
sigVersio n	可选	String(256)	body	设备的sig版本。
serialNu mber	可选	String(256)	body	设备的序列号。
batteryLe vel	可选	String(256)	body	设备的电池电量。

### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceAdded",  
    "deviceId": "*****",  
    "status": "online",  
    "statusDetail": "ok",  
    "signalStrength": -50,  
    "batteryLevel": 100,  
    "sigVersion": "1.0",  
    "serialNumber": "12345678901234567890",  
    "isSecurity": true,  
    "supportedsSecurity": true,  
    "mute": false}
```

```
"gatewayId": "*****",
"nodeType": "GATEWAY",
"deviceInfo": {
    "nodeId": "*****",
    "name": null,
    "description": null,
    "manufacturerId": null,
    "manufacturerName": null,
    "mac": null,
    "location": null,
    "deviceType": null,
    "model": null,
    "swVersion": null,
    "fwVersion": null,
    "hwVersion": null,
    "protocolType": null,
    "bridgeId": null,
    "status": "OFFLINE",
    "statusDetail": "NOT_ACTIVE",
    "mute": null,
    "supportedSecurity": null,
    "isSecurity": null,
    "signalStrength": null,
    "sigVersion": null,
    "serialNumber": null,
    "batteryLevel": null
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.2 绑定设备通知

#### 典型场景

应用服务器在物联网平台订阅了绑定设备通知后（订阅的通知类型为bindDevice），当直连设备接入并绑定物联网平台时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了绑定设备通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：bindDevice。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
resultCode	必选	String	body	绑定结果，取值：expired/succeeded。
deviceInfo	可选	DeviceInfo	body	设备信息，具体参见 <a href="#">DeviceInfo结构体</a> 。

DeviceInfo结构体说明：

参数	必选/ 可选	类型	位置	描述
nodeId	必选	String(256)	body	<p>设备唯一标识码，通常使用MAC地址，Serial No或IMEI作为nodeId。</p> <p><b>说明</b></p> <p>若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
name	可选	String(256)	body	设备名称。
description	可选	String(2048 )	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048 )	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	可选	String(256)	body	<p>设备的型号。</p> <p>Z-Wave: productType + productId, 16进制格式XXXX-XXXX补0对齐，如：001A-0A12，其他协议的格式待定。</p>
swVersion	可选	String(256)	body	<p>设备的软件版本。</p> <p>Z-Wave: 主版本号.次版本号，如：1.1。</p>
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型：Z-Wave，ZigBee，WPS。
bridgeId	可选	String(256)	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。

参数	必选/ 可选	类型	位置	描述
status	可选	String	body	<p>设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。</p> <p>若设备超过25（默认值）小时未向平台上报数据或发送消息，则设备状态为ABNORMAL（默认值）；若设备超过49小时未向平台上报数据或发送消息，则设备状态为OFFLINE。</p>
statusDetail	可选	String(256)	body	<p>设备的状态详情，跟status取值对应。</p> <ul style="list-style-type: none"><li>● status为ONLINE时，取值NONE COMMUNICATION_ERROR, CONFIGURATION_ERROR BRIDGE_OFFLINE或 FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE</li><li>● status为OFFLINE时，取值NONE CONFIGURATION_PENDING</li></ul>
mute	可选	String	body	<p>表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。</p> <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>
supporte dSecurity	可选	String	body	<p>表示设备是否支持安全模式。</p> <ul style="list-style-type: none"><li>● TRUE: 支持安全模式</li><li>● FALSE: 不支持安全模式</li></ul>
isSecurit y	可选	String	body	<p>表示设备当前是否启用安全模式。</p> <ul style="list-style-type: none"><li>● TRUE: 启用</li><li>● FALSE: 未启用</li></ul>
signalStr ength	可选	String(256)	body	设备的信号强度。
sigVersio n	可选	String(256)	body	设备的sig版本。
serialNu mber	可选	String(256)	body	设备的序列号。
batteryLe vel	可选	String(256)	body	设备的电池电量。

 **说明**

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "bindDevice",  
    "deviceId": "*****",  
    "resultCode": "succeeded",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturer": "wulian",  
        "deviceType": 90,  
        "model": "90",  
        "mac": "*****",  
        "swVersion": "...",  
        "fwVersion": "...",  
        "hwVersion": "...",  
        "protocolType": "zigbee",  
        "description": "smockdetector",  
        "nodeType": "GATEWAY"  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.3 设备信息变化通知

#### 典型场景

应用服务器在物联网平台订阅了设备信息变化通知后（订阅的通知类型为 deviceInfoChanged），当设备的配置或状态信息（如设备厂商信息、位置信息、版本信息、在线状态信息等）发生变化时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备信息变化通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceInfoChanged。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	必选	String	body	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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。
deviceInfo	必选	DeviceInfo	body	设备信息，具体参见 <a href="#">DeviceInfo结构体</a> 。

DeviceInfo结构体说明：

参数	必选/ 可选	类型	位置	描述
nodeId	必选	String(256)	body	<p>设备唯一标识码，通常使用MAC地址，Serial No或IMEI作为nodeId。</p> <p><b>说明</b></p> <p>若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
name	可选	String(256)	body	设备名称。
description	可选	String(2048 )	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048 )	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	可选	String(256)	body	<p>设备的型号。</p> <p>Z-Wave: productType + productId, 16进制格式XXXX-XXXX补0对齐，如：001A-0A12，其他协议的格式待定。</p>
swVersion	可选	String(256)	body	<p>设备的软件版本。</p> <p>Z-Wave: 主版本号.次版本号，如：1.1。</p>
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	可选	String(256)	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。

参数	必选/ 可选	类型	位置	描述
status	可选	String	body	<p>设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。</p> <p>若设备超过25（默认值）小时未向平台上报数据或发送消息，则设备状态为ABNORMAL（默认值）；若设备超过49小时未向平台上报数据或发送消息，则设备状态为OFFLINE。</p>
statusDetail	可选	String(256)	Body	<p>设备的状态详情，跟status取值对应。</p> <ul style="list-style-type: none"><li>● status为ONLINE时，取值NONE COMMUNICATION_ERROR, CONFIGURATION_ERROR BRIDGE_OFFLINE或 FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE</li><li>● status为OFFLINE时，取值NONE CONFIGURATION_PENDING</li></ul>
mute	可选	String	body	<p>表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。</p> <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>
supporte dSecurity	可选	String	body	<p>表示设备是否支持安全模式。</p> <ul style="list-style-type: none"><li>● TRUE: 支持安全模式</li><li>● FALSE: 不支持安全模式</li></ul>
isSecurit y	可选	String	body	<p>表示设备当前是否启用安全模式。</p> <ul style="list-style-type: none"><li>● TRUE: 启用</li><li>● FALSE: 未启用</li></ul>
signalStr ength	可选	String(256)	body	设备的信号强度。
sigVersio n	可选	String(256)	body	设备的sig版本。
serialNu mber	可选	String(256)	body	设备的序列号。
batteryLe vel	可选	String(256)	body	设备的电池电量。

 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceInfoChanged",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturer": "wulian",  
        "type": 90,  
        "model": "90",  
        "mac": "*****",  
        "swVersion": "...",  
        "fwVersion": "...",  
        "hwVersion": "...",  
        "protocolType": "zigbee",  
        "description": "smock detector"  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.4 设备数据变化通知

#### 典型场景

应用服务器在物联网平台订阅了设备数据变化通知后（订阅的通知类型为 deviceDataChanged），当设备上报单个服务属性数据时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备数据变化通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceDataChanged。
requestId	可选	String(1-128)	body	消息的序列号，唯一标识该消息。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	必选	String	body	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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。
service	必选	DeviceServiceData	body	设备的服务数据，具体参见 <a href="#">DeviceServiceData结构体</a> 。

DeviceServiceData结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String	body	服务ID。
serviceType	必选	String	body	服务的类型。
data	必选	ObjectNode	body	服务数据信息。
eventTime	必选	String	body	事件发生时间，时间格式 yyyyMMddThhmmssZ，例如 20151212T121212Z。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDataChanged",  
    "requestId": "*****",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "service": {  
        "serviceId": "Brightness",  
        "serviceType": "Brightness",  
        "data": {  
            "brightness": 80  
        },  
        "eventTime": "20170311T163657Z"  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.5 批量设备数据变化通知

#### 典型场景

应用服务器在物联网平台订阅了设备数据批量变化通知后（订阅的通知类型为 deviceDatasChanged），当设备同时上报多个服务属性数据时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备数据批量变化通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceDatasChanged。
requestId	必选	String	body	消息的序列号，唯一标识该消息。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	必选	String	body	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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。

参数	必选/可选	类型	位置	描述
services	必选	List<DeviceServiceData>	body	服务列表信息，具体参见 <a href="#">DeviceServiceData结构体</a> 。

DeviceServiceData结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String	body	服务ID。
serviceType	必选	String	body	服务类型。
data	必选	ObjectNode	body	服务数据信息。
eventTime	必选	String	body	事件上报时间，时间格式：yyyymmddThhmmssZ，例如20151212T121212Z。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDatasChanged",  
    "requestId": "*****",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "service": [  
        {"serviceId": "Brightness",  
         "serviceType": "Brightness",  
         "data": {}  
    }  
]}  
Content-Type: application/json
```

```
"data": {  
    "brightness": 80  
},  
    "eventTime": "20170311T163657Z"  
},  
{  
    "serviceId": "Color",  
    "serviceType": "Color",  
    "data": {  
        "value": "red"  
    },  
    "eventTime": "20170311T163657Z"  
}  
]
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 1.2.5.6 设备服务信息变化通知

#### 典型场景

应用服务器在物联网平台订阅了设备服务信息变化通知后（订阅的通知类型为 serviceInfoChanged），当平台向设备下发命令修改设备服务信息时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备服务信息变化通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	Enum	body	通知类型, 取值: serviceInfoChanged。
deviceId	必选	String	body	设备ID, 用于唯一标识一个设备, 在注册设备时由物联网平台分配获得。
gatewayId	必选	String	body	网关ID, 用于标识一个网关设备。当设备是直连设备时, gatewayId与设备的deviceId一致。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	必选	String	body	设备服务标识。
serviceType	必选	String	body	设备服务类型。
serviceInfo	必选	ServiceInfo	body	设备服务信息, 增量上报, 具体参见 <a href="#">ServiceInfo结构体</a> 。

ServiceInfo结构体说明:

参数	必选/可选	类型	位置	描述
muteCmds	可选	List<String>	body	屏蔽的设备控制命令列表

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "serviceInfoChanged",  
    "deviceId": "*****",  
    "serviceId": "*****",  
    "serviceInfo": {  
        "muteCmds": ["  
            ""  
        ]  
    }  
}
```

```
"serviceType": "*****",
"gatewayId": "*****",
"serviceInfo":
{
  "muteCmds": "VIDEO_RECORD"
}
```

## 响应示例

response:  
Status Code: 200 OK

### 1.2.5.7 删除设备通知

#### 典型场景

应用服务器在物联网平台订阅了删除设备通知后（订阅的通知类型为deviceDeleted），当设备在物联网平台中被删除后，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了删除设备通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceDeleted。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。

参数	必选/可选	类型	位置	描述
gatewayId	必选	String	body	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDeleted",  
    "deviceId": "*****",  
    "gatewayId": "*****"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.8 设备消息确认通知

#### 典型场景

应用服务器在物联网平台订阅了设备消息确认通知后（订阅的通知类型为messageConfirm），当平台向设备下发命令，并收到设备返回的命令确认消息（如命令已送达、已执行等）时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备消息确认通知的应用服务器推送通知消息。

## 注意事项

应用收到推送消息，请务必返回响应码。

## 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：messageConfirm。
header	必选	Header	body	具体参见 <a href="#">Header字段</a> 。
body	必选	ObjectNode	body	根据业务具体定义，确认消息可以携带的状态变化等消息。

Header字段说明：

参数	必选/可选	类型	位置	描述
requestId	必选	String(1-128)	body	消息的序列号，唯一标识该消息。
from	必选	String(1-128)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● 设备发起的请求：/devices/{deviceId}</li><li>● 设备服务发起的请求：/devices/{deviceId}/services/{serviceId}</li></ul>
to	必选	String(1-128)	body	表示消息接收者的地址，To就是订阅请求中的From，如应用服务器的userId。
status	必选	String(1-32)	body	命令状态。 <ul style="list-style-type: none"><li>● sent: 已发送</li><li>● delivered: 已送达</li><li>● executed: 已执行</li></ul>
timestamp	必选	String(1-32)	body	时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "messageConfirm",  
    "header": {  
        "requestId": "*****",  
        "from": "*****",  
        "to": "*****",  
        "status": "delivered",  
        "timestamp": "20151212T121212Z"  
    },  
    "body": {}  
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.9 设备命令响应通知

#### 典型场景

应用服务器在物联网平台订阅了设备命令响应通知后（订阅的通知类型为 commandRsp），当平台向设备下发命令，并收到设备返回的命令响应消息（如命令执行成功或者执行失败，设备要向平台返回指定内容响应）时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备命令响应通知的应用服务器推送通知消息。

## 注意事项

应用收到推送消息，请务必返回响应码。

## 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：commandRsp。
header	必选	Header	body	具体参见 <a href="#">Header字段</a> 。
body	必选	ObjectNode	body	响应命令的消息内容。

Header字段说明：

参数	必选/可选	类型	位置	描述
requestId	必选	String(1-128)	body	消息的序列号，唯一标识该消息。
from	必选	String(1-128)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>设备发起的请求：/devices/{deviceId}</li><li>设备服务发起的请求：/devices/{deviceId}/services/{serviceId}</li></ul>
to	必选	String(1-128)	body	表示消息接收者的地址，To就是订阅请求中的From，如应用服务器的userId。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
serviceType	必选	String	body	命令所属的服务类型。

参数	必选/可选	类型	位置	描述
method	必选	String(1-128) )	body	存放的响应命令，如：INVITE-RSP。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "commandRsp",  
    "header": {  
        "requestId": "*****",  
        "from": "*****",  
        "to": "*****",  
        "deviceId": "*****",  
        "serviceType": "Camera",  
        "method": "MUTE_COMMANDS"  
    },  
    "body": {}  
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.10 命令状态变化通知

#### 典型场景

应用服务器通过物联网平台创建设备命令时，如果设置了callbackurl回调地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达），平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台在命令状态发生变化时向应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器下发命令请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
commandId	必选	string	body	设备命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。
result	必选	CommandResultForDevice	body	命令状态信息，具体参见 <a href="#">CommandResultForDevice结构体</a> 。

CommandResultForDevice结构体说明：

参数	必选/可选	类型	位置	描述
resultCode	必选	String	body	命令状态。
resultDetail	必选	ObjectNode	body	命令结果详细信息。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request:

Body

```
{  
    "deviceId": "92d3f8da-200a-4143-8d0d-591a7e11de6c",  
    "commandId": "108a9c71462a48e09426e06e844d47ba",  
    "result": {  
        "resultCode": "SENT",  
        "resultDetail": null  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.11 设备事件通知

#### 典型场景

应用服务器在物联网平台订阅了设备事件通知后（订阅的通知类型为deviceEvent），当平台收到设备上报的事件消息时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备事件通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
------	------

请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceEvent。
header	必选	Header	body	具体参见 <a href="#">Header字段</a> 。
body	必选	ObjectNode	body	响应命令的消息内容。

Header字段说明：

参数	必选/可选	类型	位置	描述
eventType	必选	String(1-32)	body	事件类型。
from	必选	String(1-28)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● 设备发起的请求：/devices/{deviceId}</li><li>● 设备服务发起的请求：/devices/{deviceId}/services/{serviceId}</li></ul>
timestamp	必选	String(1-32)	body	时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
eventTime	必选	String(1-32)	body	事件上报时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
serviceType	必选	String	body	命令所属的服务类型。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
  "notifyType": "deviceEvent",  
  "header": {  
    "eventType": "*****",  
    "from": "/devices/{deviceId}/services/{serviceId}",  
    "deviceId": "*****",  
    "serviceType": "*****",  
    "timestamp": "20151212T121212Z",  
    "eventTime": "20151212T121212Z"  
  },  
  "body": {  
    "usedPercent": 80  
  }  
}
```

## 响应示例

response:  
Status Code: 200 OK

### 1.2.5.12 增加设备模型通知

#### 典型场景

应用服务器在物联网平台订阅了设备模型增加通知后（订阅的通知类型为 deviceModelAdded），在物联网平台上新增设备Profile文件时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备模型增加通知的应用服务器推送通知消息。

## 注意事项

应用收到推送消息，请务必返回响应码。

## 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceModelAdded。
appId	必选	String	body	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceType	必选	String	body	设备的类型。
manufacturerName	必选	String	body	设备模型的厂商名称。
manufacturerId	必选	String	body	设备模型的厂商ID。
model	必选	String	body	设备型号。
protocolType	必选	String	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceModelAdded",  
    "appId": "*****",  
    "deviceType": "*****",  
    "manufacturerName": "wulian",  
    "manufacturerId": "*****",  
    "model": "*****",  
    "protocolType": "zigbee"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.13 删除设备模型通知

#### 典型场景

应用服务器在物联网平台订阅了设备模型删除通知后（订阅的通知类型为 deviceModelDeleted），在物联网平台上删除设备Profile文件时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备模型删除通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型, 取值: deviceModelDeleted。
appId	必选	String	body	访问物联网平台的应用ID, 在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceType	必选	String	body	设备的类型。
manufacturerName	必选	String	body	设备模型的厂商名称。
manufacturerId	必选	String	body	设备模型的厂商ID。
model	必选	String	body	设备型号。
protocolType	必选	String	body	设备使用的协议类型, 当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceModelAdded",  
    "appId": "*****",  
    "deviceTyp": "*****",  
    "manufacturerName": "*****",  
    "manufacturerId": "*****",  
    "model": "*****",
```

```
"protocolType": "*****  
}
```

## 响应示例

response:  
Status Code: 200 OK

### 1.2.5.14 设备影子状态变更通知

#### 典型场景

应用服务器在物联网平台订阅了设备影子状态变更通知后（订阅的通知类型为 deviceDesiredPropertiesModifyStatusChanged），当平台中的设备影子向设备同步数据成功或失败时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备影子状态变更通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceDesiredPropertiesModifyStatusChanged。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
serviceId	必选	String	body	服务ID，用于标识一个服务。
properties	必选	ObjectNode	body	设备影子数据属性
status	必选	String	body	状态，取值DELIVERED/FAILED。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

{

"notifyType":"deviceDesiredPropertiesModifyStatusChanged",

"deviceId":"\*\*\*\*\*",

"serviceId":"Device",

"properties":{

"Model Number":1,

"Serial Number":2,

"Firmware Version":"v1.1.0"

},

"status":"DELIVERED"

}

## 响应示例

response:

Status Code: 200 OK

### 1.2.5.15 软件升级状态变更通知

#### 典型场景

应用服务器在物联网平台订阅了软件升级状态变更通知后（订阅的通知类型为swUpgradeStateChangeNotify），当设备进行软件升级，升级任务的状态发生变化时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了软件升级状态变更通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

## 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值： swUpgradeStateChangeNotify。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	软件升级任务ID。
subOperationId	必选	String	body	软件升级子任务ID。
swUpgradeState	必选	String	body	软件升级状态。 <ul style="list-style-type: none"><li>● downloading: 设备正在下载软件包</li><li>● downloaded: 设备下载软件包完成</li><li>● updating: 设备正在进行升级</li><li>● idle: 设备处于空闲状态</li></ul>

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

{

```
"notifyType": "swUpgradeStateChangeNotify",
"deviceId": "*****",
"appId": "*****",
"operationId": "*****",
"subOperationId": "*****",
"swUpgradeState": "downloading"
}
```

## 响应示例

```
response:
Status Code: 200 OK
```

### 1.2.5.16 软件升级结果通知

#### 典型场景

应用服务器在物联网平台订阅了软件升级结果通知后（订阅的通知类型为 swUpgradeResultNotify），当设备的软件升级任务执行完成时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了软件升级结果通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：swUpgradeResultNotify。

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	软件升级任务ID。
subOperationId	必选	String	body	软件升级子任务ID。
curVersion	必选	String	body	设备当前的软件版本。
targetVersion	必选	String	body	设备要升级的目标软件版本。
sourceVersion	必选	String	body	设备的源软件版本。
swUpgradeResult	必选	String	body	软件升级结果。 ● SUCCESS: 设备升级成功 ● FAIL: 设备升级失败
upgradeTime	必选	String	body	升级时长。
resultDesc	必选	String	body	升级结果描述。
errorCode	必选	String	body	设备上报的状态错误码。
description	必选	String	body	错误原因描述。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "swUpgradeResultNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",
```

```
"subOperationId":"*****",
"curVersion":"1.3",
"targetVersion":"1.5",
"sourceVersion":"1.0",
"swUpgradeResult":"SUCCESS",
"upgradeTime":"***",
"resultDesc":"***",
"errorCode":"***",
"description":"***"
}
```

## 响应示例

```
response:
Status Code: 200 OK
```

### 1.2.5.17 固件升级状态变更通知

#### 典型场景

应用服务器在物联网平台订阅了固件升级状态变更通知后（订阅的通知类型为 fwUpgradeStateChangeNotify），当设备进行固件升级，升级任务的状态发生变化时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了固件升级状态变更通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型, 取值: fwUpgradeStateChangeNotify。
deviceId	必选	String	body	设备ID, 用于唯一标识一个设备, 在注册设备时由物联网平台分配获得。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	固件升级任务ID。
subOperationId	必选	String	body	固件升级子任务ID。
step	必选	String	body	固件升级状态, 可取值为0、1、2、3。
stepDesc	必选	String	body	升级状态描述。 <ul style="list-style-type: none"><li>● 1: downloading: 设备正在下载软件包</li><li>● 2: downloaded: 设备下载软件包完成</li><li>● 3: updating: 设备正在进行升级</li><li>● 0: idle: 设备处于空闲状态</li></ul>

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "fwUpgradeStateChangeNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "step": 0,  
    "stepDesc": ""  
}
```

```
"subOperationId":"*****",
"step":"1",
"stepDesc":"downloading"
}
```

## 响应示例

response:  
Status Code: 200 OK

### 1.2.5.18 固件升级结果通知

#### 典型场景

应用服务器在物联网平台订阅了固件升级结果通知后（订阅的通知类型为 fwUpgradeResultNotify），当设备的固件升级任务执行完成时，平台会向应用服务器推送通知消息。

#### 接口功能

支持物联网平台向订阅了固件升级结果通知的应用服务器推送通知消息。

#### 注意事项

应用收到推送消息，请务必返回响应码。

#### 接口原型

请求方法	POST
请求地址	由应用服务器订阅请求消息中的callbackurl决定。
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：fwUpgradeResultNotify。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	固件升级任务ID。

参数	必选/可选	类型	位置	描述
subOperationId	必选	String	body	固件升级子任务ID。
curVersion	必选	String	body	设备当前的固件版本。
targetVersion	必选	String	body	设备要升级的目标固件版本。
sourceVersion	必选	String	body	设备的源固件版本。
status	必选	String	body	升级结果。 <ul style="list-style-type: none"><li>● SUCCESS</li><li>● FAIL</li></ul>
statusDesc	必选	String	body	升级结果描述。 <ul style="list-style-type: none"><li>● SUCCESS: 设备升级成功</li><li>● FAIL: 设备升级失败</li></ul>
upgradeTime	必选	String	body	固件升级时长。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "fwUpgradeResultNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "curVersion": "1.6",  
    "targetVersion": "1.6",  
    "sourceVersion": "1.3",  
    "status": "SUCCESS",  
}
```

```
"statusDesc":"****",  
"upgradeTime":"****"  
}
```

## 响应示例

response:  
Status Code: 200 OK

## 1.2.6 命令下发

### 1.2.6.1 创建设备命令

#### 典型场景

设备Profile文件中定义了物联网平台可向设备下发的命令，应用服务器可调用此接口向设备下发命令，配置或修改设备的服务属性，以实现对设备的控制。

物联网平台有两种命令下发机制：

- 立即下发：平台收到命令后立即下发给设备，保证实时性，不保证串行性。
- 缓存下发：平台收到命令后缓存起来，在设备可达时排队串行下发，平台收到前一个命令的响应（模组自动回复的ACK）才会下发后一个命令，保证串行性，不保证实时性。

#### 接口功能

支持应用服务器向设备下发命令，实现对设备的控制。支持物联网平台立即下发命令和缓存下发命令。

#### 接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/cmd/v1.4.0/deviceCommands? appId={appId}
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。

参数	必选/可选	类型	位置	描述
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	可选	String(1-64)	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceId	必选	String(64)	body	下发命令的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
command	必选	CommandDTO	body	下发命令的信息。
callbackUrl	可选	String(1024)	body	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知应用服务器。
expireTime	可选	Integer(>=0)	body	下发命令的有效时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。 如果expireTime设置为0，则无论物联网平台上设置的命令下发模式是什么，该命令都会立即下发给设备（如果设备处于休眠状态或者链路已老化，则设备收不到命令，平台没收到设备的响应，该命令最终会超时）。
maxRetransmit	可选	Integer(0~3)	body	命令下发最大重传次数。

CommandDTO:

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-64)	body	命令对应的服务ID，用于标识一个服务。要与profile中定义的serviceId保持一致。
method	必选	String(1-128)	body	命令服务下具体的命令名称，要与profile中定义的命令名保持一致。

参数	必选/可选	类型	位置	描述
paras	可选	ObjectNode	body	命令参数, jsonString格式, 里面是一个个键值对, 每个键都是profile中命令的参数名 (paraName), 具体格式需要应用和设备约定。

## 响应参数

StatusCode: 201 Created

参数	类型	描述
commandId	String(1-64)	设备命令ID, 用于唯一标识一条命令, 在下发设备命令时由物联网平台分配获得。
appId	String(1-64)	设备命令所属应用ID。
deviceId	String(1-64)	下发命令的设备ID, 用于唯一标识一个设备, 在注册设备时由物联网平台分配获得。
command	CommandID TO	下发命令的信息。
callbackUrl	String(1024)	命令状态变化通知地址, 当命令状态变化时 (执行失败, 执行成功, 超时, 发送, 已送达) 会通知应用服务器。
expireTime	Integer(>=0)	下发命令的超时时间, 单位为秒, 表示设备命令在创建后expireTime秒内有效, 超过这个时间范围后命令将不再下发, 如果未设置则默认为48小时 (86400s*2)。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● PENDING表示缓存未下发</li><li>● EXPIRED表示命令已经过期</li><li>● SUCCESSFUL表示命令已经成功执行</li><li>● FAILED表示命令执行失败</li><li>● TIMEOUT表示命令下发执行超时</li><li>● CANCELED表示命令已经被撤销执行</li><li>● DELIVERED表示命令已送达设备</li><li>● SENT表示命令正在下发</li></ul>
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。

参数	类型	描述
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

## 请求示例

Method: POST

request: https://server:port/iocm/app/cmd/v1.4.0/deviceCommands

Header:

"app\_key: \*\*\*\*\*"

"Authorization:Bearer\*\*\*\*\*"

Content-Type:application/json

Body:

```
{  
    "deviceId": "*****",  
    "command": {  
        "serviceId": "*****",  
        "method": "*****",  
        "paras": {  
            "paraName1": "paraValue1",  
            "paraName2": "paraValue2"  
        }  
    },  
    "callbackUrl": "http://127.0.0.1:9999/cmd/callbackUrl",  
    "maxRetransmit": "*****",  
}
```

## 响应示例

response:

Status Code: 201 OK

Content-Type:application/json

Body:

```
{  
    "commandId": "*****",  
    "appId": "*****",  
    "deviceId": "*****",  
    "command": {  
        "serviceId": "*****",  
        "method": "*****",  
        "paras": {  
            "paraName1": "paraValue1",  
            "paraName2": "paraValue2"  
        }  
    },  
    "callbackUrl": "http://127.0.0.1:9999/cmd/callbackUrl",  
    "expireTime": null,  
    "status": "PENDDING",  
    "creationTime": "20170222T164000Z",  
    "executeTime": null,  
    "platformIssuedTime": null,  
    "deliveredTime": null,  
    "issuedTimes": null,  
    "maxRetransmit": *****  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>请检查该设备的Profile文件是否已上传到物联网平台。</li><li>请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	100223	Command counts has reached the upLimit.	缓存的命令数已到达限制。处于PENDING状态的命令条数不超过限定期。默认限定期为20。 处理建议：如当前平台缓存的命令还需执行，请让设备上报数据触发缓存命令的下发；如当前平台缓存的命令有不需要执行的命令，请调用“修改设备命令V4”接口，将命令的状态从PENDING修改为CANCELED。
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	100612	Device is zombie.	设备为僵尸设备。（距离设备上次上线时间超过阈值，默认7天） 处理建议：请把设备上线后再重新下发命令。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 1.2.6.2 查询设备命令

#### 典型场景

应用服务器向设备下发命令后，可调用此接口在物联网平台查询下发命令的状态及内容信息，以了解命令的执行情况。

#### 接口功能

支持应用服务器在物联网平台查询下发命令的状态和内容信息，可查询指定时间段内本应用下的所有下发命令，或者指定设备的所有下发命令。

## 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iocm/app/cmd/v1.4.0/deviceCommands?pageNo={pageNo}&amp;pageSize={pageSize}&amp;deviceId={deviceId}&amp;startTime={startTime}&amp;endTime={endTime}&amp;appId={appId}">https://server:port/iocm/app/cmd/v1.4.0/deviceCommands? pageNo={pageNo}&amp;pageSize={pageSize}&amp;deviceId={deviceId}&amp;start tTime={startTime}&amp;endTime={endTime}&amp;appId={appId}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
pageNo	可选	Integer(>=0)	query	查询的页码，大于等于0，默认值：0。
pageSize	可选	Integer[1,1000]	query	查询每页信息的数量，大于等于1，最大值1000，默认值：1000。
deviceId	可选	String(64)	query	指定查询命令的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
startTime	可选	String	query	查询开始时间，查询下发命令时间在startTime之后的记录。时间格式： yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。
endTime	可选	String	query	查询结束时间，查询下发命令时间在endTime之前的记录。时间格式： yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。

## 响应参数

Status Code: 200 OK

参数	类型	描述
pagination	<b>Pagination</b>	页面信息。
data	List< <b>DeviceCommandResp</b> >	设备命令列表。

Pagination:

参数	类型	描述
pageNo	long	页码。
pageSize	long	每页信息数量。
totalSize	long	记录总数。

DeviceCommandResp:

参数	类型	描述
commandId	String(1-64)	设备命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。
appId	String(1-64)	设备命令所属的应用ID。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
command	<b>CommandDTO</b>	下发命令的信息。
callbackUrl	String(1024)	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知应用服务器。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。

参数	类型	描述
status	String	下发命令的状态。 <ul style="list-style-type: none"> <li>● PENDING表示缓存未下发</li> <li>● EXPIRED表示命令已经过期</li> <li>● SUCCESSFUL表示命令已经成功执行</li> <li>● FAILED表示命令执行失败</li> <li>● TIMEOUT表示命令下发执行超时</li> <li>● CANCELED表示命令已经被撤销执行</li> <li>● DELIVERED表示命令已送达设备</li> <li>● SENT表示命令正在下发中</li> </ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTO:

参数	必选/可选	类型	描述
serviceId	必选	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	必选	String(1-128)	命令服务下具体的命令名称，要与profile中定义的命令名保持一致。
paras	可选	ObjectNode	命令参数， jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。

## 请求示例

Method: GET

request:

```
https://server:port/iocm/app/cmd/v1.4.0/deviceCommands?  

pageNo=0&pageSize=10&deviceId=*****&startTime=20151112T101012Z&endTime=201  

51212T121212Z
```

Header:

```
"app_key: *****"  
"Authorization:Bearer*****"  
Content-Type:application/json
```

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
  "pagination": {  
    "pageNo": 0,  
    "pageSize": 20,  
    "totalSize": 100  
  },  
  "data": [  
    {  
      "commandId": "*****",  
      "appId": "*****",  
      "deviceId": "*****",  
      "command": {  
        "serviceId": "*****",  
        "method": "*****",  
        "paras": {  
          "paraName1": "paraValue1",  
          "paraName2": "paraValue2"  
        }  
      },  
      "callbackUrl": "http://127.0.0.1:9999/cmd/callbackUrl",  
      "expireTime": null,  
      "status": "PENDING",  
      "result": null,  
      "creationTime": "20170222T164000Z",  
    }  
  ]}
```

```
"executeTime": null,  
"platformIssuedTime": null,  
"deliveredTime": null,  
"issuedTimes": null,  
"maxRetransmit":*****  
},  
{  
"commandId": "*****",  
"appId": "*****",  
"deviceId": "*****",  
"command": {  
"serviceId": "*****",  
"method": "*****",  
"paras": {  
"paraName1": "paraValue1",  
"paraName2": "paraValue2"  
}  
},  
"callbackUrl": "http://127.0.0.1:9999/cmd/callbackUrl",  
"expireTime": null,  
"status": "PENDDING",  
"result": null,  
"creationTime": "20170222T164000Z",  
"executeTime": null,  
"platformIssuedTime": null,  
"deliveredTime": null,  
"issuedTimes": null,  
"maxRetransmit":*****  
}  
]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>请检查该设备的Profile文件是否已上传到物联网平台。</li><li>请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议： <ul style="list-style-type: none"><li>请确认startTime和endTime都不为空，且endTime比startTime晚。</li><li>请确认pageNo不为空，且pageNo大于0。</li><li>请检查pageSize不为空，且pageSize大于1。</li></ul>

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 1.2.6.3 修改设备命令

#### 典型场景

应用服务器向设备下发命令后，如果由于命令排队、设备不在线等原因，物联网平台还未把命令下发给设备执行，此时应用服务器可调用此接口修改该命令的状态信息。当前仅支持把命令状态修改为CANCELED状态，即撤销命令的执行。

## 接口功能

支持应用服务器修改指定命令的状态信息，仅能修改还处于PENDING状态的命令，且当前仅支持修改的命令状态为CANCELED，即撤销命令。

## 接口原型

请求方法	PUT
请求地址	https://server:port/iocm/app/cmd/v1.4.0/deviceCommands/{deviceCommandId}?appId={appId}
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceCommandId	必选	String	path	要修改的命令ID，在调用创建设备命令接口后获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
status	必选	String	body	命令的状态，可选值：CANCELED，撤销命令。

## 响应参数

Status Code: 200 OK

参数	类型	描述
commandId	String(1-64)	设备命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。
appId	String(1-64)	设备命令所属的应用ID。

参数	类型	描述
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
command	<b>CommandDTO</b>	下发命令的信息。
callbackUrl	String(1024)	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知应用服务器。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● PENDING表示缓存未下发</li><li>● EXPIRED表示命令已经过期</li><li>● SUCCESSFUL表示命令已经成功执行</li><li>● FAILED表示命令执行失败</li><li>● TIMEOUT表示命令下发执行超时</li><li>● CANCELED表示命令已经被撤销执行</li><li>● DELIVERED表示命令已送达设备</li><li>● SENT表示命令正在下发中</li></ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTO:

参数	类型	描述
serviceId	String(1-64)	命令对应的服务ID，用于标识一个服务。

参数	类型	描述
method	String(1-128)	命令服务下具体的命令名称，要与profile中定义的命令名保持一致。
paras	ObjectNode	命令参数， jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。

## 请求示例

Method: PUT

request:

`https://server:port/iocm/app/cmd/v1.4.0/deviceCommands/{deviceCommandId}?appId={appId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

`Content-Type: application/json`

Body:

{

`"status":"CANCELED"`

}

## 响应示例

response:

`Status Code: 200 OK`

`Content-Type: application/json`

Body:

{

`"commandId": "*****",`

`"appId": "*****",`

`"deviceId": "*****",`

`"command": {`

`"serviceId": "*****",`

`"method": "*****",`

`"paras": {`

```
"paraName1": "paraValue1",
"paraName2": "paraValue2"
},
},
"callbackUrl": "http://127.0.0.1:9999/cmd/callbackUrl",
"expireTime": null,
"status": "PENDING",
"result": null,
"creationTime": "20170222T164000Z",
"executeTime": null,
"platformIssuedTime": null,
"deliveredTime": null,
"issuedTimes": null,
"maxRetransmit": *****
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
200	100434	The device command is not existed.	设备命令不存在。 处理建议：请检查接口请求中的设备命令ID是否正确。
200	100435	The device command already canceled, expired or executed, Cannot cancel.	设备命令已经取消，过期或执行，不能取消。
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。

Http状态码	错误码	错误描述	说明
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

#### 1.2.6.4 批量创建设备命令

应用服务器若需要向多个设备下发相同的设备命令时，可调用此接口向批量设备下发命令。

此功能通过[创建批量任务](#)接口向设备下发批量任务实现，“taskType”参数设置为“DeviceCmd”时，为向批量设备下发命令。

#### 1.2.6.5 创建设备命令撤销任务

##### 典型场景

应用服务器向设备下发命令后，如果由于命令排队、设备不在线等原因，物联网平台还未把命令下发给设备执行（PENDING状态），此时应用服务器可调用此接口撤销指定设备的所有未下发的命令，对于已下发成功的命令不可撤销。

##### 接口功能

支持应用服务器创建设备命令撤销任务，该任务用于撤销物联网平台中指定设备ID下所有未下发的命令（命令处于PENDING状态）。

##### 接口原型

请求方法	POST
请求地址	<a href="https://server:port/iocm/app/cmd/v1.4.0/deviceCommandCancelTasks?appId={appId}">https://server:port/iocm/app/cmd/v1.4.0/deviceCommandCancelTasks? appId={appId}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceId	必选	String(1-64)	body	待撤销设备命令的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。撤销任务将会撤销所有向该设备下发的命令。

## 响应参数

Status Code: 201 Created

参数	类型	描述
taskId	String(1-64)	设备命令撤销任务的任务ID。
appId	String(1-64)	设备命令撤销任务所属的应用ID。
deviceId	String(1-64)	执行设备命令撤销任务的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
status	String	撤销任务的任务状态。 <ul style="list-style-type: none"><li>● WAITTING表示等待执行中</li><li>● RUNNING表示撤销任务正在执行</li><li>● SUCCESS表示撤销任务执行成功</li><li>● FAILED表示撤销任务执行失败</li><li>● PART_SUCCESS表示撤销任务部分执行成功。</li></ul>
totalCount	Integer	撤销的设备命令总数。
deviceCommands	List<DeviceCommandResp>	撤销的设备命令列表。

DeviceCommandResp:

参数	类型	描述
commandId	String(1-64)	设备命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。
appId	String(1-64)	设备命令撤销任务所属的应用ID。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
command	CommandDTO	下发命令的信息。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● DEFAULT表示未下发</li><li>● EXPIRED表示命令已经过期</li><li>● SUCCESSFUL表示命令已经成功执行</li><li>● FAILED表示命令执行失败</li><li>● TIMEOUT表示命令下发执行超时</li><li>● CANCELED表示命令已经被撤销执行</li></ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTO:

参数	类型	描述
serviceId	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	String(1-128)	命令服务下具体的命令名称，要与profile中定义的命令名保持一致。

参数	类型	描述
paras	ObjectNode	命令参数, jsonString格式, 里面是一个个键值对, 每个键都是profile中命令的参数名( paraName ), 具体格式需要应用和设备约定。

## 请求示例

Method: POST

request:

`https://server:port/iocm/app/cmd/v1.4.0/deviceCommandCancelTasks?appId={appId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

`Content-Type: application/json`

Body:

{

`"deviceId": "*****"`

}

## 响应示例

response:

Status Code: 201 Created

`Content-Type: application/json`

Body:

{

`"taskId": "*****",`

`"appId": "*****",`

`"deviceId": "*****",`

`"status": "WAITTING",`

`"totalCount": 1,`

`"deviceCommands": [`

{

`"commandId": "*****",`

`"appId": "*****",`

```
"deviceId": "*****",
"command": {
    "serviceId": "*****",
    "method": "*****",
    "paras": {
        "paraName1": "paraValue1",
        "paraName2": "paraValue2"
    }
},
"callbackUrl": "http://127.0.0.1:9999/cmd/callbackUrl",
"expireTime": null,
"status": "PENDDING",
"result": null,
"creationTime": "20170222T164000Z",
"executeTime": null,
"platformIssuedTime": null,
"deliveredTime": null,
"issuedTimes": null,
"maxRetransmit": *****
}
]
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

## 1.2.6.6 查询设备命令撤销任务

### 典型场景

应用服务器创建了设备命令撤销任务后，可调用此接口查询设备命令撤销任务的详细信息和执行状态，以了解撤销任务的执行情况。

### 接口功能

支持应用服务器在物联网平台查询设备命令撤销任务信息和状态，可指定条件查询单个或多个撤销任务。

### 接口原型

请求方法	GET
请求地址	<code>https://server:port/iocm/app/cmd/v1.4.0/deviceCommandCancelTasks? pageNo={pageNo}&amp;pageSize={pageSize}&amp;taskId={taskId}&amp;deviceId={deviceId}&amp;status={status}&amp;startTime={startTime}&amp;endTime={endTime}&amp;appId={appId}</code>
传输协议	HTTPS

### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
pageNo	可选	Integer(>=0)	query	查询的页码，大于等于0，默认值：0。
pageSize	可选	Integer[1,1000]	query	查询每页信息的数量，大于等于1，最大值1000，默认值：1000。
taskId	可选	String	query	撤销任务的任务ID。
deviceId	可选	String	query	执行设备命令撤销任务的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
status	可选	String	query	设备命令撤销任务的状态。

参数	必选/可选	类型	位置	描述
startTime	可选	String	query	查询开始时间，查询创建撤销设备命令任务时间在 startTime 之后的记录。 时间格式： yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。
endTime	可选	String	query	查询结束时间，查询创建撤销设备命令任务时间在 endTime 之前的记录。 时间格式： yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的 appId。

## 响应参数

Status Code: 200 OK

参数	类型	描述
pagination	<a href="#">Pagination</a>	页码信息。
data	List< <a href="#">DeviceCommandCancelTaskResp</a> >	设备命令列表。

Pagination:

参数	类型	描述
pageNo	long	查询结果页码。
pageSize	long	每页信息数量。
totalSize	long	记录总数，即查询到的撤销任务中的命令总数。

DeviceCommandCancelTaskResp:

参数	类型	描述
taskId	String(1-64)	设备命令撤销任务的任务ID。
appId	String(1-64)	设备命令撤销任务所属的应用ID。

参数	类型	描述
deviceId	String(1-64)	设备命令撤销任务指定撤销命令的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
status	String	撤销任务的任务状态。 <ul style="list-style-type: none"><li>● WAITTING表示等待执行中</li><li>● RUNNING表示撤销任务正在执行</li><li>● SUCCESS表示撤销任务执行成功</li><li>● FAILED表示撤销任务执行失败</li><li>● PART_SUCCESS表示撤销任务部分执行成功。</li></ul>
totalCount	Integer	撤销的设备命令总数。
deviceCommands	List< <a href="#">DeviceCommandAndResp</a> >	设备命令撤销任务信息列表。

DeviceCommandResp:

参数	类型	描述
commandId	String(1-64)	设备命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。
appId	String(1-64)	设备命令撤销任务所属的应用ID。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
command	<a href="#">CommandDTO</a>	下发命令的信息。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● DEFAULT表示未下发</li><li>● EXPIRED表示命令已经过期</li><li>● SUCCESSFUL表示命令已经成功执行</li><li>● FAILED表示命令执行失败</li><li>● TIMEOUT表示命令下发执行超时</li><li>● CANCELED表示命令已经被撤销执行</li></ul>

参数	类型	描述
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTO:

参数	类型	描述
serviceId	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	String(1-128)	命令服务下具体的命令名称，要与profile中定义的命令名保持一致。
paras	ObjectNode	命令参数， jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。

## 请求示例

Method: GET

request:

```
https://server:port/iocm/app/cmd/v1.4.0/deviceCommandCancelTasks?  
pageNo={pageNo}&pageSize={pageSize}&taskId={taskId}&deviceId={deviceId}&status={  
status}&startTime={startTime}&endTime={endTime}&appId={appId}
```

Header:

"app\_key: \*\*\*\*\*"

"Authorization:Bearer\*\*\*\*\*"

Content-Type: application/json

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
    "pagination": {  
        "pageNo": 0,  
        "pageSize": 20,  
        "totalSize": 100  
    },  
    "data": [  
        {  
            "taskId": "*****",  
            "appId": "*****",  
            "deviceId": "*****",  
            "status": "WAITTING",  
            "totalCount": 1,  
            "deviceCommands": [  
                {  
                    "commandId": "*****",  
                    "appId": "*****",  
                    "deviceId": "*****",  
                    "command": {  
                        "serviceId": "*****",  
                        "method": "*****",  
                        "paras": {  
                            "paraName1": "paraValue1",  
                            "paraName2": "paraValue2"  
                        }  
                    },  
                    "callbackUrl": "http://127.0.0.1:9999/cmd/callbackUrl",  
                    "expireTime": null,  
                    "status": "PENDDING",  
                    "result": null,  
                    "creationTime": "20170222T164000Z",  
                    "executeTime": null,  
                }  
            ]  
        }  
    ]  
}
```

```
"platformIssuedTime": null,  
"deliveredTime": null,  
"issuedTimes": null,  
"maxRetransmit": "*****  
}  
]  
},  
{  
"taskId": "*****",  
"appId": "*****",  
"deviceId": "*****",  
"status": "WAITTING",  
"totalCount": 1,  
"deviceCommands": [  
{  
"commandId": "*****",  
"appId": "*****",  
"deviceId": "*****",  
"command": {  
"serviceId": "*****",  
"method": "*****",  
"paras": {  
"paraName1": "paraValue1",  
"paraName2": "paraValue2"  
}  
},  
}  
,"callbackUrl": "http://127.0.0.1:9999/cmd/callbackUrl",  
"expireTime": null,  
"status": "PENDING",  
"result": null,  
"creationTime": "20170222T164000Z",  
"executeTime": null,  
"platformIssuedTime": null,
```

```
"deliveredTime": null,  
"issuedTimes": null,  
"maxRetransmit":*****  
}  
]  
}  
]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 1.2.6.7 设备服务调用

#### 典型场景

设备Profile文件中定义了物联网平台可向设备下发的命令，应用服务器可调用此接口向设备下发命令，配置或修改设备的服务属性，以实现对设备的控制。

物联网平台不对设备命令进行缓存而是直接下发命令，设备不在线时命令下发失败。下发命令的具体格式需要应用服务器与设备自定义，物联网平台在接口中直接进行封装，透传。



##### 说明

目前该接口仅用于安装了IoT Agent/AgentLite的网关设备，通过向网关设备下发命令，可对网关下的非直连设备进行控制。

#### 接口功能

支持应用服务器向设备立即下发命令（当前仅支持安装了IoT Agent/AgentLite的网关设备），实现对设备的控制。

## 接口原型

请求方法	POST
请求地址	<a href="https://server:port/iocm/app/signaltrans/v1.1.0/devices/{deviceId}/services/{serviceId}/sendCommand?appId={appId}">https://server:port/iocm/app/signaltrans/v1.1.0/devices/{deviceId}/services/{serviceId}/sendCommand?appId={appId}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceId	必选	String(1-64)	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
serviceId	必选	String(1-64)	path	服务ID，唯一标识一个服务。如：“Discovery”为发现非直连设备，“Remove”为删除非直连设备。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
header	必选	CommandNA2CloudHeader	body	消息的消息头。
body	可选	ObjectNode	body	消息的消息体。JsonObject里面是一个个键值对。每个键都是profile中命令的参数名（paraName）。

CommandNA2CloudHeader:

参数	必选/可选	类型	位置	描述
requestId	可选	String(0 -128)	body	用于标识一个命令，不能重复。

参数	必选/可选	类型	位置	描述
mode	必选	Enum	body	是否要确认消息， 默认为ACK模式。 <ul style="list-style-type: none"> <li>● NOACK：不需要确认消息</li> <li>● ACK：需要确认消息</li> <li>● 其它值无效</li> </ul>
from	可选	String(1-28)	body	表示消息发布者的地址。 <ul style="list-style-type: none"> <li>● App发起的请求： /users/{userId}</li> <li>● 应用发起的请求： /{serviceName}</li> <li>● 物联网平台发起的请求： /cloud/{serviceName}</li> </ul>
toType	可选	Enum	body	消息接受者的类型， CLOUD/GATEWAY， 默认为GATEWAY。
to	可选	String(1-28)	body	消息接收者的地址。
method	必选	String(1-32)	body	命令名称， 如：“DISCOVERY”为发现非直连设备，“REMOVE”为删除非直连设备。
callbackURL	可选	String(1-024)	body	命令的回调路径。

## 响应参数

Status Code: 200 OK

参数	类型	描述
status	String(128)	命令状态。 <ul style="list-style-type: none"> <li>● sent：已发送</li> <li>● delivered：已投递， toType为CLOUD的请求才有该返回值</li> <li>● failed：失败， toType为CLOUD的请求才有该返回值</li> </ul>
timestamp	String(128)	发送命令的时间戳， 时间格式： <code>yyyyMMdd'T'HHmmss'Z'</code> ， 如 <code>20151212T121212Z</code> 。
requestId	String(128)	设备命令ID。 <ul style="list-style-type: none"> <li>● toType为GATEWAY时， 如果请求有requestId则同请求的requestId一致， 如果请求无requestId则平台分配一个序列号。</li> <li>● toType为CLOUD时，则为空。</li> </ul>

## 请求示例

Method: POST

request:

`https://server:port/iocm/app/signaltrans/v1.1.0/devices/{deviceId}/services/{serviceId}/sendCommand`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

`Content-Type:application/json`

Body:

```
{  
  "header": {  
    "mode": "ACK",  
    "from": "/users/23212121",  
    "method": "INVITE-INIT",  
    "callbackURL": "http://10.10.10.10:8043/na/iocm/message/confirm"  
  },  
  "body": {  
    "from": "*****",  
    "sessionID": "*****",  
    "sdp": "*****"  
  }  
}
```

## 响应示例

response:

Status Code: 202 Accepted.

Content-Type: application/json

Body:

```
{  
  "requestId": "*****",  
  "status": "sent",  
  "timestamp": "*****"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与网关的连接是否正常。
200	100432	The device command is muted.	设备命令已被禁用。 处理建议：请检查接口请求参数method中的命令是否有误。
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	102203	CommandName is invalid.	命令名称无效。 处理建议：请检查接口请求参数method中携带的命令是否有误。
403	100450	The gateway is not online.	网关不在线。 处理建议：请检查网关与物联网平台的连接是否正常。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	100444	The serviceType is not exist.	服务类型不存在。 处理建议：请检查接口请求参数toType中的服务类型是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	50252	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

## 1.2.7 数据采集

物联网平台支持应用服务器查询设备的基本信息，同时还支持查看设备上报的历史数据，能按时、天，月等维度查看设备上报的历史数据。

### 1.2.7.1 查询单个设备信息

#### 典型场景

已在物联网平台注册的设备，应用服务器若需要查看某个设备的详细信息（如厂商、型号、版本、状态和服务属性等信息），可调用此接口在物联网平台查询获取。

#### 接口功能

支持应用服务器在物联网平台根据设备ID查询指定设备的详细信息，包括配置信息、状态信息、服务属性信息等。

#### 接口原型

请求方法	GET
------	-----

请求地址	<a href="https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?appId={appId}&amp;select={select}">https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?appId={appId}&amp;select={select}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
select	可选	String	query	指定查询条件，可选值：imsi。

## 响应参数

Status Code: 200 OK

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	String(256)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	Enum	节点类型，取值：ENDPOINT/GATEWAY/UNKNOW。
createTime	String(256)	创建设备的时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。
lastModifiedTime	String(256)	最后修改设备的时间。
deviceInfo	<a href="#">DeviceInfoQuery DTO</a>	设备信息。

参数	类型	描述
services	List< <a href="#">DeviceService</a>  e>	设备服务列表。
alarmInfo	<a href="#">AlarmInfoDTO</a>	设备告警信息。

DeviceInfoQueryDTO:

参数	类型	描述
nodeId	String(256)	设备唯一标识码。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacturerId	String(256)	厂商ID，唯一标识一个厂商。
manufacturerName	String(256)	厂商名称。
mac	String(256)	设备的MAC地址。
location	String(2048)	设备的位置信息。
deviceType	String(256)	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号。 Z-Wave: productType + productId, 16进制格式 XXXX-XXXX补0对齐, 如: 001A-0A12, 其他协议的格式待定。
swVersion	String(256)	设备的软件版本。 Z-Wave: 主版本号.次版本号, 如: 1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
imsi	String	NB-IoT终端的IMSI。
protocolType	String(256)	设备使用的协议类型, 当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
radiusIp	String	Radius地址。
bridgeId	String(256)	Bridge标识, 表示设备通过哪个Bridge接入物联网平台。

参数	类型	描述
status	String	设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。 若设备超过25（默认值）小时未向平台上报数据或发送消息，则设备状态为ABNORMAL（默认值）；若设备超过49小时未向平台上报数据或发送消息，则设备状态为OFFLINE。
statusDetail	String(256)	设备的状态详情，跟status取值对应。 <ul style="list-style-type: none"><li>● status为ONLINE时，取值NONE COMMUNICATION_ERROR, CONFIGURATION_ERROR BRIDGE_OFFLINE或 FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE</li><li>● status为OFFLINE时，取值NONE CONFIGURATION_PENDING</li></ul>
mute	String	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>
supportedSecurity	String	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● TRUE: 支持安全模式</li><li>● FALSE: 不支持安全模式</li></ul>
isSecurity	String	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● TRUE: 启用</li><li>● FALSE: 未启用</li></ul>
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。
batteryLevel	String(256)	设备的电池电量。

### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceService:

参数	类型	描述
serviceId	String(256)	设备的服务标识。

参数	类型	描述
serviceType	String(256)	设备的服务类型。
serviceInfo	ServiceInfo	设备的服务信息。
data	ObjectNode(2097152)	属性值对。
eventTime	String(256)	时间格式: yyyyMMdd'T'HHmmss'Z', 如 20151212T121212Z。

ServiceInfo:

参数	类型	描述
muteCmds	List<String>	屏蔽的设备控制命令列表

AlarmInfoDTO:

参数	类型	描述
alarmSeverity	String	告警级别。
alarmStatus	Boolean	告警状态。
alarmTime	String	告警上报时间。

## 请求示例

Method: GET

request:

`https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?select=imsi`

Header:

`"app_key: **"`

`"Authorization:Bearer**"`

`Content-Type:application/json`

## 响应示例

response:

Status Code: 200 OK

`Content-Type: application/json`

Body:

```
{  
    "deviceId": "xxxxx",  
    "gatewayId": "xxxxx",  
    "nodeType": "xxxxx",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturerName": "wulian",  
        "deviceType": "gateway",  
        "model": "90",  
        "mac": "C7EA1904004B1204",  
        "swVersion": "th",  
        "fwVersion": "seu",  
        "hwVersion": "sru",  
        "protocolType": "zigbee",  
        "description": "smockdetector",  
        "imsi": "xxxxx"  
    },  
    "services": [  
        {  
            "serviceType": "air_conditioner",  
            "serviceId": "1",  
            "data": {  
                "battery_low": 1  
            }  
        },  
        {  
            "serviceType": "air_conditioner",  
            "serviceId": "jkh",  
            "data": {  
                "battery_low": "jhj"  
            }  
        }  
    ]  
}
```

{}

## 错误码

Http状态码	错误码	错误描述	说明
400	100405	The request parameter is invalid.	请求参数不合法。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.7.2 批量查询设备信息

#### 典型场景

已在物联网平台注册的设备，应用服务器若需要根据条件查看多个设备的详细信息（如厂商、型号、版本、状态和服务属性等信息），可调用此接口在物联网平台查询获取。

#### 接口功能

支持应用服务器在物联网平台根据条件查询多个设备的详细信息，包括配置信息、状态信息、服务属性信息等。

#### 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iocm/app/dm/v1.4.0/devices?appId={appId}&amp;deviceType={deviceType}&amp;endTime={endTime}&amp;gatewayId={gatewayId}&amp;location={location}&amp;name={name}&amp;nodeType={nodeType}&amp;pageNo={pageNo}&amp;pageSize={pageSize}&amp;select={select}&amp;sort={sort}&amp;startTime={startTime}&amp;status={status}">https://server:port/iocm/app/dm/v1.4.0/devices? appId={appId}&amp;deviceType={deviceType}&amp;endTime={endTime}&amp;ga tewayId={gatewayId}&amp;location={location}&amp;name={name}&amp;nodeTyp e={nodeType}&amp;pageNo={pageNo}&amp;pageSize={pageSize}&amp;select={s elect}&amp;sort={sort}&amp;startTime={startTime}&amp;status={status}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。

参数	必选/可选	类型	位置	描述
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
gatewayId	可选	String	query	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	可选	String	query	节点类型，取值：ENDPOINT/GATEWAY/UNKNOW。
deviceType	可选	String	query	设备类型。
location	可选	String	query	设备位置信息。
name	可选	String	query	设备名称。
pageNo	可选	Integer	query	查询的页码。 <ul style="list-style-type: none"><li>● 值为空时查询内容不分页</li><li>● 值大于等于0的整数时分页查询</li><li>● 值等于0时查询第一页</li></ul>
pageSize	可选	Integer	query	查询每页信息的数量，缺省值：1。
status	可选	String	query	设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。 若设备超过25（默认值）小时未向平台上报数据或发送消息，则设备状态为ABNORMAL（默认值）；若设备超过49小时未向平台上报数据或发送消息，则设备状态为OFFLINE。
startTime	可选	String	query	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	query	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	query	指定返回记录的排序。 <ul style="list-style-type: none"><li>● ASC：按注册设备的时间升序排列</li><li>● DESC：按注册设备的时间降序排列</li></ul> 缺省值：DESC。

参数	必选/可选	类型	位置	描述
select	可选	String	query	指定返回记录，可取值：imsi。

## 响应参数

Status Code: 200 OK

参数	类型	描述
totalCount	Long	查询的记录数量。
pageNo	Long	查询的页码。
pageSize	Long	查询每页信息的数量。
devices	List< <a href="#">QueryDeviceID TO4Cloud2NA</a> >	设备分页列表信息。

[QueryDeviceDTO4Cloud2NA](#):

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	String(256)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	Enum	节点类型，取值：ENDPOINT/GATEWAY/UNKNOW。
createTime	String(256)	创建设备的时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。
lastModifiedTime	String(256)	最后修改设备的时间。
deviceInfo	<a href="#">DeviceInfoQueryDTO</a>	设备信息。
services	List< <a href="#">DeviceService</a> >	设备服务列表。
alarmInfo	<a href="#">AlarmInfoDTO</a>	设备告警信息。

[DeviceInfoQueryDTO](#):

参数	类型	描述
nodeId	String(256)	设备唯一标识码。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacture rId	String(256)	厂商ID，唯一标识一个厂商。
manufacture rName	String(256)	厂商名称。
mac	String(256)	设备的MAC地址。
location	String(2048)	设备的位置信息。
deviceType	String(256)	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号。 Z-Wave: productType + productId, 16进制格式 XXXX-XXXX补0对齐, 如: 001A-0A12, 其他协议 的格式待定。
swVersion	String(256)	设备的软件版本。 Z-Wave: 主版本号.次版本号, 如: 1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
imsi	String	NB-IoT终端的IMSI。
protocolTyp e	String(256)	设备使用的协议类型, 当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
radiusIp	String	Radius地址。
bridgeId	String(256)	Bridge标识, 表示设备通过哪个Bridge接入物联网平台。
status	String	设备的状态, 表示设备是否在线, 取值范围: ONLINE、OFFLINE、ABNORMAL。 若设备超过25（默认值）小时未向平台上报数据或 发送消息, 则设备状态为ABNORMAL（默认值）; 若设备超过49小时未向平台上报数据或发送消息, 则设备状态为OFFLINE。

参数	类型	描述
statusDetail	String(256)	设备的状态详情，跟status取值对应。 <ul style="list-style-type: none"><li>● status为ONLINE时，取值NONE COMMUNICATION_ERROR, CONFIGURATION_ERROR BRIDGE_OFFLINE或 FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE</li><li>● status为OFFLINE时，取值NONE CONFIGURATION_PENDING</li></ul>
mute	String	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>
supportedSecurity	String	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● TRUE: 支持安全模式</li><li>● FALSE: 不支持安全模式</li></ul>
isSecurity	String	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● TRUE: 启用</li><li>● FALSE: 未启用</li></ul>
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。
batteryLevel	String(256)	设备的电池电量。

 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceService:

参数	类型	描述
serviceId	String(256)	设备的服务标识。
serviceType	String(256)	设备的服务类型。
serviceInfo	ServiceInfo	设备的服务信息。
data	ObjectNode(2097152)	属性值对（Attribute-value pair）。

参数	类型	描述
eventTime	String(256)	时间格式: yyyyMMdd'T'HHmmss'Z', 如 20151212T121212Z。

ServiceInfo:

参数	类型	描述
muteCmds	List<String>	屏蔽的设备控制命令列表

AlarmInfoDTO:

参数	类型	描述
alarmSeverity	String	告警级别。
alarmStatus	Boolean	告警状态。
alarmTime	String	告警上报时间。

## 请求示例

Method: GET

request:

`https://server:port/iocm/app/dm/v1.4.0/devices?gatewayId={gatewayId}&select=imsi`

Header:

`"app_key: **"`

`"Authorization:Bearer**"`

`Content-Type:application/json`

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "totalCount": "****",  
    "pageNo": "*****",  
}
```

```
"pageSize": "*****",
"devices": [
{
  "deviceId": "xxxxx",
  "gatewayId": "xxxxx",
  "nodeType": "xxxxx",
  "deviceInfo": {
    "name": "Sensor_12",
    "manufacturerName": "wulian",
    "deviceType": "gateway",
    "model": "90",
    "mac": "C7EA1904004B1204",
    "swVersion": "th",
    "fwVersion": "seu",
    "hwVersion": "sru",
    "protocolType": "zigbee",
    "description": "smockdetector",
    "imsi": "xxxxx"
  },
  "services": [
    {
      "serviceType": "air_conditioner",
      "serviceId": "1",
      "data": {
        "battery_low": "1"
      }
    },
    {
      "serviceType": "air_conditioner",
      "serviceId": "jkh",
      "data": {
        "battery_low": "jhj"
      }
    }
  ]
}
```

```
        }
    ],
},
{
  "deviceId": "xxxxx",
  "gatewayId": "xxxxx",
  "nodeType": "xxxxx",
  "deviceInfo": {
    "name": "Sensor_12",
    "manufacturerName": "wulian",
    "type": 90,
    "model": "90",
    "mac": "C7EA1904004B1204",
    "swVersion": "...",
    "fwVersion": "...",
    "hwVersion": "...",
    "protocolType": "zigbee",
    "description": "smockdetector",
    "imsi": "xxxxx"
  },
  "services": [
    {
      "serviceType": "air_conditioner",
      "serviceId": "1",
      "data": {
        "battery_low": "1"
      }
    },
    {
      "serviceType": "air_conditioner",
      "serviceId": "1",
      "data": {
        "battery_low": "1"
      }
    }
  ]
}
```

```
    }  
    }  
]  
}  
]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。
400	100218	The gatewayId and pageNo can't be both null.	网关ID和pageNo不能同时为空。 处理建议：请检查请求参数中 gatewayId 或 pageNo 是否填写。
400	100405	The request parameter is invalid.	请求参数不合法。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.7.3 查询设备历史数据

#### 典型场景

在日常运行中，物联网平台会收到并保存设备上报业务数据（设备数据的保存时间可通过“修改设备信息”接口配置，最长保存90天），应用服务器若需要查看某个设备上报到平台的历史数据，可调用此接口查询获取。

#### 接口功能

支持应用服务器根据设备ID，查询指定设备上报到物联网平台的历史数据。

#### 接口原型

请求方法	GET
请求地址	https://server:port/iocm/app/data/v1.2.0/deviceDataHistory? deviceId={deviceId}&gatewayId={gatewayId}&appId={appId}&page No={pageNo}&pageSize={pageSize}&startTime={startTime}&endTi me={endTime}
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。

参数	必选/可选	类型	位置	描述
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceId	必选	String	query	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	必选	String	query	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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。
pageNo	可选	Integer	query	分页查询参数，默认0，查询第一页，建议通过设置pageNo和pageSize做分页查询。
pageSize	可选	Integer	query	分页查询参数，默认1，最大值2000，建议根据设备上报数据的平均大小来设置，保证单次查询返回的包大小不超过16Mb，同时根据单次查询返回的包大小以及网络带宽设置合理的超时时间。
startTime	可选	String	query	查询产生时间在startTime之后的历史数据。 时间格式：yyyy-MMdd'T'HHmmss'Z'，如20151212T121212Z。
endTime	可选	String	query	查询产生时间在endTime之前的历史数据。 时间格式：yyyy-MMdd'T'HHmmss'Z'，如20151212T121212Z。

## 响应参数

StatusCode: 200 OK

参数	类型	描述
totalCount	Long	查询的记录数量。
pageNo	Long	查询的页码。
pageSize	Long	查询每页信息的数量。
deviceDataHistoryDTOs	List< <a href="#">DeviceDataHistoryDTO</a> >	设备历史数据列表。

DeviceDataHistoryDTO:

参数	类型	描述
serviceId	String(256)	设备的服务标识。
deviceId	String(256)	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	String(256)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	String(256)	设备所属的应用ID。
data	JsonObject	设备上报的数据。
timestamp	String(256)	上报数据的时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。

## 请求示例

Method: GET

request:

```
https://server:port/iocm/app/data/v1.2.0/deviceDataHistory?  
deviceId={deviceId}&gatewayId={gatewayId}&appId={appId}&pageNo={pageNo}&pageSize={pageSize}&startTime={startTime}&endTime={endTime}
```

Content-Type:application/json

Header:

"app\_key:\*\*\*\*\*"

"Authorization:Bearer \*\*\*\*\*"

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
    "totalCount": "*****",  
    "pageNo": "*****",  
    "pageSize": "*****",  
    "deviceDataHistoryDTOs": [  
        {
```

```
"serviceId": "*****",
"deviceId": "*****",
"gatewayId": "*****",
"appId": "*****",
"data": "*****",
"timestamp": "*****"
}
]
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	<p>设备信息不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。 请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100216	The application input is invalid.	<p>应用输入无效。</p> <p>处理建议：请参考请求参数说明检查接口请求中的参数是否有误。例如，pageSize是否超过2000。</p>
400	100419	The deviceId and gatewayId can't be both null.	<p>deviceId和gatewayId不能同时为空。</p> <p>处理建议：请检查请求参数中deviceId和gatewayId是否填写。</p>
403	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	<p>应用未被授权。</p> <p>处理建议：若非应用授权场景，请确认请求参数中的appId为空。</p>

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

#### 1.2.7.4 查询设备影子历史数据

##### 典型场景

应用服务器通过“修改设备影子”接口修改设备影子的配置时，物联网平台会保存修改记录。当应用服务器需要查看设备影子的历史配置记录时，可调用此接口查询获取。

##### 接口功能

支持应用服务器根据设备ID，在物联网平台查询指定设备影子的历史配置数据。

##### 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iocm/app/shadow/v1.5.0/deviceDesiredHistory?deviceId={deviceId}&amp;gatewayId={gatewayId}&amp;appId={appId}&amp;serviceId={serviceId}&amp;property={property}&amp;pageNo={pageNo}&amp;pageSize={pageSize}&amp;startTime={startTime}&amp;endTime={endTime}">https://server:port/iocm/app/shadow/v1.5.0/deviceDesiredHistory? deviceId={deviceId}&amp;gatewayId={gatewayId}&amp;appId={appId}&amp;serviceId={serviceId}&amp;property={property}&amp;pageNo={pageNo}&amp;pageSize={pageSize}&amp;startTime={startTime}&amp;endTime={endTime}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
appId	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceId	必选	String	query	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	必选	String	query	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	query	设备的服务标识。
property	可选	String	query	服务属性数据。
pageNo	可选	Integer	query	查询的页码。 <ul style="list-style-type: none"><li>● 值为空时查询内容不分页</li><li>● 值大于等于0的整数时分页查询</li><li>● 值等于0时查询第一页</li></ul>
pageSize	可选	Integer	query	查询每页信息的数量，缺省值：1。
startTime	可选	String	query	查询产生时间在startTime之后的历史数据。时间格式：yyyy-MMdd'T'HHmmss'Z'，如20151212T121212Z。
endTime	可选	String	query	查询产生时间在endTime之前的历史数据。时间格式：yyyy-MMdd'T'HHmmss'Z'，如20151212T121212Z。

## 响应参数

StatusCode: 200 OK

参数	类型	描述
totalCount	Long	查询的记录数量。

参数	类型	描述
pageNo	Long	查询的页码。
pageSize	Long	查询每页信息的数量。
DeviceDesiredHistoryDTO	List<DeviceDesiredHistoryDTO>	设备影子历史配置数据列表。

DeviceDesiredHistoryDTO:

参数	类型	描述
serviceId	String(256)	设备的服务标识。
deviceId	String(256)	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
gatewayId	String(256)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	String(256)	设备所属的应用ID。
desired	JsonObject	要向设备下发的配置信息。
timestamp	String(256)	数据配置的时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。

## 请求示例

Method: GET

request:

```
https://server:port/iocm/app/shadow/v1.5.0/deviceDesiredHistory?  
deviceId={deviceId}&gatewayId={gatewayId}&appId={appId}&serviceId={serviceId}&pro  
perty={property}&pageNo={pageNo}&pageSize={pageSize}&startTime={startTime}&endT  
ime={endTime}
```

Content-Type:application/json

Header:

"app\_key:\*\*\*\*\*"

"Authorization:Bearer \*\*\*\*\*"

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
    "totalCount": "*****",  
    "pageNo": "*****",  
    "pageSize": "*****",  
    "DeviceDesiredHistoryDTO": [  
        {  
            "serviceId": "*****",  
            "deviceId": "*****",  
            "gatewayId": "*****",  
            "appId": "*****",  
            "desired": "*****",  
            "timestamp": "*****"  
        }  
    ]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。
400	100419	The deviceId and gatewayId can't be both null.	deviceId和gatewayId不能同时为 空。 处理建议：请检查请求参数中 deviceId和gatewayId是否填写。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.7.5 查询设备服务能力

#### 典型场景

应用服务器若需要了解设备可上报哪些服务属性数据，以及设备支持下发哪些命令，可调用此接口在物联网平台查询设备的Profile文件中定义的设备服务能力信息。

#### 接口功能

支持应用服务器在物联网平台查询设备的服务属性、设备命令等服务能力信息。

## 接口原型

请求方法	GET
请求地址	https://server:port/iocm/app/data/v1.1.0/deviceCapabilities? appId={appId}&gatewayId={gatewayId}&deviceId={deviceId}
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
gatewayId	可选	String	query	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	可选	String	query	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。此处填写授权应用的appId。
deviceId	可选	String	query	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。

## 响应参数

参数	类型	描述
deviceCapabilities	List< <a href="#">DeviceCapabilityDTO</a> >	查询结果列表。

DeviceCapabilityDTO:

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。

参数	类型	描述
serviceCapa bilities	List< <a href="#">ServiceCapa bilityDTO</a> >	设备的服务能力列表。

ServiceCapabilityDTO:

参数	类型	描述
serviceId	String(256)	设备的服务标识。
serviceType	String(256)	设备的服务类型。
option	String(256)	服务选项。
description	String(10240)	设备服务描述信息。
commands	List< <a href="#">ServiceCom mand</a> >	支持的命令名称列表。
properties	List< <a href="#">ServiceProp erty</a> >	支持的属性名称列表。

ServiceCommand:

参数	类型	描述
commandN ame	String(256)	命令名称。
paras	List< <a href="#">ServiceComm andPara</a> >	属性列表。
responses	List< <a href="#">ServiceComm andResponse</a> >	响应列表。

ServiceCommandPara:

参数	类型	描述
paraName	String(256)	参数名称。
dataType	String(256)	数据类型。
required	Boolean	是否必选。
min	String	属性最小值。
max	String	属性最大值。

参数	类型	描述
step	Double	步长。
maxLength	Integer	最大长度。
unit	String	单位（符号）。
enumList	List<String>	枚举类型列表。

ServiceCommandResponse:

参数	类型	描述
responseName	String(256)	响应名称。
paras	List< <a href="#">ServiceCommandPara</a> >	属性列表。

ServiceProperty:

参数	类型	描述
propertyName	String(256)	属性名称。
dataType	String(256)	数据类型。
required	Boolean	是否必选。
min	String	属性最小值。
max	String	属性最大值。
step	Double	步长。
maxLength	Integer	最大长度。
method	String(256)	访问方法。 <ul style="list-style-type: none"><li>● R: 可读</li><li>● W: 可写</li><li>● E: 可观察</li></ul>
unit	String	单位（符号）。
enumList	List<String>	枚举类型列表。

## 请求示例

Method: GET

request:

```
https://server:port/iocm/app/data/v1.1.0/deviceCapabilities?  
appId={appId}&gatewayId={gatewayId}&deviceId={deviceId}
```

Content-Type:application/json

Header:

"app\_key:\*\*\*\*\*"

"Authorization:Bearer \*\*\*\*\*"

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
  "deviceCapabilities": [  
    {  
      "deviceId": "****",  
      "serviceCapabilities": [  
        {  
          "serviceId": "****",  
          "serviceType": "****",  
          "option": "****",  
          "description": "****",  
          "commands": [  
            {  
              "commandName": "*****",  
              "paras": [{  
                "paraName": "*****",  
                "dataType": "*****",  
                "required": "True",  
                "min": "*****",  
                "max": "*****",  
                "step": "*****",  
                "maxLength": 1111111,  
                "unit": "*****"  
              }]  
            }]  
          ]  
        }]  
      ]  
    }]  
}
```

```
"unit": "*****",
"enumList": [
],
},
"responses": [
{
"responseName": "***",
"paras": [
{
"paraName": "*****",
"dataType": "*****",
"required": "Ture",
"min": "*****",
"max": "*****",
"step": "*****",
"maxLength": 1111111,
"unit": "*****",
"enumList": [
}
]
},
}
],
"properties": [
{
"propertyName": "*****",
"dataType": "*****",
"required": "Ture",
"min": "*****",
"max": "*****",
"step": "*****",
"maxLength": 1111111,
"method": "*****",
"unit": "*****",
"enumList": [
}
]
```

```
    }  
]  
}  
]  
}  
]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 1.2.8 设备组管理

### 1.2.8.1 创建设备组

#### 典型场景

应用服务器可调用此接口在物联网平台上创建设备组，并把设备归类到不同的设备组内，进行分组管理。一个设备可以归属到多个设备组内。

在对设备进行某些操作时（如升级设备软固件、批量下发命令等），可通过设备组来指定要进行操作的设备。

#### 接口功能

支持应用服务器在物联网平台创建设备组，用于对设备进行分组管理。

#### 接口原型

请求方法	POST
请求地址	<a href="https://server:port/iocm/app/devgroup/v1.3.0/devGroups">https://server:port/iocm/app/devgroup/v1.3.0/devGroups</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
name	必选	String(1-50)	body	设备组名称，仅限大小写字母和数字。

参数	必选/可选	类型	位置	描述
description	可选	String(1024)	body	设备组的描述信息。
appId	可选	String(50)	body	当在授权应用下增加设备组时需要填写, 填写授权应用的应用ID。
maxDevNum	可选	Integer (>=0)	body	设备组设备最大数量, 默认最小值为0。当值为0时, 表示对设备数量不做限制。
deviceIds	可选	List<String>	body	添加到设备组的设备ID列表。

## 响应参数

Status Code: 200 ok

参数	类型	描述
name	String(50)	设备组名称, 仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String(50)	设备组所属的应用ID。
maxDevNum	Integer (>=0)	设备组设备最大数量限制, 当值为0时, 表示对设备数量不做限制。
curDevNum	Integer	当前设备组内的设备数量。
deviceIds	List<String>	添加到设备组的设备ID列表。

## 请求示例

Method: POST

request:

<https://server:port/iocm/app/devgroup/v1.3.0/devGroups>

Header:

"app\_key: \*\*\*\*\*"

"Authorization: Bearer\*\*\*\*\*"

Content-Type: application/json

Body:

{

```
"name": "*****",
"description": "*****",
"appId": "*****",
"maxDevNum": "*****",
"deviceIds": [
    "*****",
    "*****",
    "*****",
    ],
}
```

## 响应示例

response:

Status Code: 201 OK

Content-Type:application/json

Body:

```
{
"name": "*****",
"description": "*****",
"id": "*****",
"appId": "*****",
"maxDevNum": "*****",
"curDevNum": "*****",
"deviceIds": [
    "*****",
    "*****",
    "*****",
    ],
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100602	The device group name has been used.	设备组名字已经存在。 处理建议：请修改接口请求中的设备组名称。
200	100607	The devGroup has reached the limit.	设备组数目达到限制。 处理建议：请检查已创建的设备组数量是否已达到License限定的数量上限。
400	100609	Too much devices to add.	添加太多设备至设备组。 处理建议：请确认deviceIds中的设备ID数量在maxDevNum设置值的范围内。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 1.2.8.2 删除设备组

### 典型场景

若因分组变更，应用服务器不再需要使用某个设备组，且不想在物联网平台上继续保存该设备组信息时，可调用此接口在物联网平台删除指定设备组。

### 接口功能

支持应用服务器在物联网平台根据设备组ID删除指定设备组。

### 接口原型

请求方法	DELETE
请求地址	<a href="https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}">https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}</a>
传输协议	HTTPS

### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
devGroupId	必选	String	path	设备组ID，在增加设备组后由物联网平台返回获得。
accessAppId	可选	String	query	当删除授权应用下的设备组时需要填写，填写授权应用的应用ID。

### 响应参数

Status Code: 200 ok

### 请求示例

Method: DELETE

request:

<https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}>

Header:

```
"app_key: *****"  
"Authorization: Bearer*****"  
Content-Type: application/json
```

## 响应示例

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
200	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100603	The device group is not existed	设备组不存在。 处理建议：请检查设备组ID是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.8.3 修改设备组

#### 典型场景

若因业务变更需要修改设备组的信息，如设备组名称、设备组的设备数量限制等，应用服务器可调用此接口修改指定设备组的信息。

#### 接口功能

支持应用服务器在物联网平台修改指定设备组的信息。

#### 接口原型

请求方法	PUT
请求地址	<a href="https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}">https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
accessAppId	可选	String	query	当修改授权应用下的设备组时需要填写，填写授权应用的应用ID。
devGroupId	必选	String (50)	path	设备组ID，在增加设备组后由物联网平台返回获得。
name	必选	String (1-50)	body	设备组名称，仅限大小写字母和数字。
description	可选	String (1024)	body	设备组的描述信息。
maxDevNum	可选	Integer (>=0)	body	设备组设备最大数量，默认值为0。当值为0时，表示对设备数量不做限制。

#### 响应参数

Status Code: 200 ok

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String(50)	设备组所属的应用ID。
maxDevNum	Integer (>=0)	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。
curDevNum	Integer	当前设备组内设备数量。

## 请求示例

Method: PUT

request:

`https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

`Content-Type: application/json`

Body:

```
{  
    "name": "*****",  
    "description": "*****",  
    "maxDevNum": "*****",  
}
```

## 响应示例

response:

Status Code: 200 OK

`Content-Type:application/json`

Body:

```
{  
    "name": "*****",  
    "description": "*****",  
}
```

```
"id": "*****",
"appId": "*****",
"maxDevNum": "*****",
"curDevNum": "*****"
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议: 请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100602	The device group name has been used.	设备组的名字已经存在。 处理建议: 请修改接口请求中的设备组名称。
200	100603	The device group is not existed.	设备组不存在。 处理建议: 请检查设备组ID是否正确。
400	50400	The input is invalid.	输入参数无效。 处理建议: 请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁, 超过流控值(默认值是100次/60s)。 处理建议: 联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议: 请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

#### 1.2.8.4 查询设备组列表

##### 典型场景

应用服务器可调用此接口查询当前已创建的所有设备组列表信息，以了解当前设备组的分组和使用情况。

##### 接口功能

支持应用服务器在物联网平台查询已创建的所有设备组信息。

##### 接口原型

请求方法	GET
请求地址	<code>https://server:port/iocm/app/devgroup/v1.3.0/devGroups?accessAppId={accessAppId}&amp;pageNo={pageNo}&amp;pageSize={pageSize}&amp;name={name}</code>
传输协议	HTTPS

##### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“ <code>Bearer {accessToken}</code> ”，其中 <code>{accessToken}</code> 为调用鉴权接口返回的accessToken。
accessAppId	可选	String	query	当查询授权应用下的设备组时需要填写，填写授权应用的应用ID。

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer	query	分页查询参数。 ● 值为空时，查询内容不分页。 ● 值为大于等于0的整数时，分页查询。 ● 值为0时查询第一页。
pageSize	可选	Integer	query	每页设备组记录数量，默认值为1。
name	可选	String	query	设备组名称。

## 响应参数

Status Code: 200 ok

参数	类型	描述
totalCount	long	设备组总数。
pageNo	long	查询结果页码。
pageSize	long	每页设备组记录数量。
list	List<QueryDevGroupD TOCloud2NA>	设备组信息详情。

QueryDevGroupDTOCloud2NA:

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String(50)	设备组所属的应用ID。
maxDevNum	Integer(>=0)	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。
curDevNum	Integer	当前设备组内的设备数量。
creator	String(1-50)	创建设备组的用户名。

## 请求示例

Method: GET

request:

[https://server:port/iocm/app/devgroup/v1.3.0/devGroups?  
accessAppId={accessAppId}&pageNo={pageNo}&pageSize={pageSize}&name={name}](https://server:port/iocm/app/devgroup/v1.3.0/devGroups?accessAppId={accessAppId}&pageNo={pageNo}&pageSize={pageSize}&name={name})

Header:

"app\_key: \*\*\*\*\*"

"Authorization: Bearer\*\*\*\*\*"

Content-Type: application/json

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "totalCount": "*****",  
    " pageNo": "*****",  
    " pageSize": "*****",  
    "list": [ "object" ]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.8.5 查询指定设备组

#### 典型场景

应用服务器若需要查看某个设备组的信息，以了解该设备组的使用情况，可调用此接口查询指定设备组的信息。

#### 接口功能

支持应用服务器在物联网平台根据设备组ID查询指定设备组的信息。

#### 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}">https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
devGroupId	必选	String	path	设备组ID，在增加设备组后由物联网平台返回获得。
accessAppId	可选	String	query	当查询授权应用下的设备组时需要填写，填写授权应用的应用ID。

## 响应参数

Status Code: 200 ok

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String (50)	设备组所属的应用ID。
maxDevNum	Integer (>=0)	设备组设备最大数量限制。
curDevNum	Integer	当前设备组内的设备数量。
creator	String(1-50)	创建设备组的用户名称。

## 请求示例

Method: GET

request:

`https://server:port/iocm/app/devgroup/v1.3.0/devGroups/{devGroupId}?accessAppId={accessAppId}`

Header:

`"app_key: *****"`

`"Authorization: Bearer*****"`

`Content-Type: application/json`

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "name": "*****",  
    "description": "*****",  
    "id": "*****",  
    "appId":  
    "maxDevNum": "*****",  
    "curDevNum": "*****",  
}
```

```
"creator": "*****"  
}  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.8.6 查询指定设备组成员

#### 典型场景

应用服务器若需要查看某个设备组中的设备成员分布情况，可调用此接口查询指定设备组的设备成员列表信息。

#### 接口功能

支持应用服务器在物联网平台根据设备组ID查询指定设备组内设备列表信息。

## 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iocm/app/dm/v1.2.0/devices/ids?devGroupId={devGroupId}&amp;accessAppId={accessAppId}&amp;pageNo={pageNo}&amp;pageSize={pageSize}">https://server:port/iocm/app/dm/v1.2.0/devices/ids? devGroupId={devGroupId}&amp;accessAppId={accessAppId}&amp;pageNo={ pageNo}&amp;pageSize={pageSize}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
devGrou pId	必选	String	query	设备组ID，在增加设备组后由物联网平台返回获得。
appId	可选	String	query	当查询授权应用下的设备组时需要填写，填写授权应用的应用ID。
pageNo	可选	Integer	query	分页查询参数，默认0，查询第一页，建议通过设置pageNo和pageSize做分页查询。
pageSize	可选	Integer(1 000)	query	每页设备记录数量，默认值为10。

## 响应参数

StatusCode: 200 ok

参数	类型	描述
totalCount	long	设备组内设备总数。
pageNo	long	查询结果页码。
pageSize	long	每页设备记录数量。
deviceIds	List<String>	设备组内设备ID列表。

## 请求示例

Method: GET

request:

```
https://server:port/iocm/app/dm/v1.2.0/devices/ids?  
devGroupId={devGroupId}&accessAppId={accessAppId}&pageNo={pageNo}&pageSize={  
pageSize}
```

Header:

"app\_key: \*\*\*\*\*"

"Authorization:Bearer\*\*\*\*\*"

Content-Type: application/json

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
    "totalCount": "*****",  
    "pageNo": "*****",  
    "pageSize": "*****",  
    "deviceIds": [  
        "*****",  
        "*****",  
        "*****"  
    ]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	107001	The serviceId is not exist.	服务ID不存在。 处理建议：请检查接口请求中的serviceId是否有误。

Http状态码	错误码	错误描述	说明
400	107002	The properties is empty in database.	设备属性不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107003	The request properties is unknown.	设备状态为未知。 处理建议：请检查设备与平台的连接是否正常。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.8.7 增加设备组成员

#### 典型场景

若需要把新增的设备或已有设备增加到某个设备组中，应用服务器可调用此接口向指定设备组添加设备成员。在向设备组添加设备前，建议通过“查询指定设备组”接口

查询该设备组的当前设备数量及最大设备数量限制，确保设备组中还有足够的可添加成员数量。

## 接口功能

支持应用服务器在物联网平台向指定设备组添加设备。

## 接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/dm/v1.1.0/devices/addDevGroupTagToDevices?accessAppId={accessAppId}
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
accessAppId	可选	String	query	当向授权应用下的设备组添加设备时需要填写，填写授权应用的应用ID。
devGroupId	必选	String(1-50)	body	设备组ID，在增加设备组后由物联网平台返回获得。
deviceIds	必选	List<String>(1000)	body	要添加到设备组的设备ID列表。

## 返回参数

Status Code: 200 ok

参数	类型	描述
devGroupId	String(1-50)	设备组ID。
deviceIds	List<String>	添加到设备组的设备ID列表。

## 请求示例

Method: POST

request:

`https://server:port/iocm/app/dm/v1.1.0/devices/addDevGroupTagToDevices?  
accessAppId={accessAppId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

`Content-Type: application/json`

Body:

```
{  
    "devGroupId": "*****",  
    "deviceIds": [  
        "*****",  
        "*****",  
        "*****",  
        "*****"  
    ]  
}
```

## 响应示例

response:

Status Code: 200 OK

`Content-Type:application/json`

Body:

```
{  
    "devGroupId": "*****",  
    "deviceIds": [  
        "*****",  
        "*****",  
        "*****"  
    ]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议： <ul style="list-style-type: none"><li>● 请检查接口请求中的设备ID是否有误。</li><li>● 请检查设备组是否达到最大设备数量限制。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 1.2.8.8 删除设备组成员

#### 典型场景

若设备组中的一个或多个设备不再归属于该设备组，应用服务器可调用此接口从设备组删除设备成员。

#### 接口功能

支持应用服务器在物联网平台从指定设备组删除设备。

#### 接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/dm/v1.1.0/devices/deleteDevGroupTagFromDevices?accessAppId={accessAppId}
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。

参数	必选/可选	类型	位置	描述
accessAppId	可选	String	query	当在授权应用下的设备组删除设备时需要填写，填写授权应用的应用ID。
devGroupId	必选	String(1-50)	body	设备组ID，在增加设备组后由物联网平台返回获得。
deviceIds	必选	List<String>(1000)	body	要从设备组删除的设备ID列表。

## 响应参数

Status Code: 200 ok

## 请求示例

Method: POST

request:

`https://server:port/iocm/app/dm/v1.1.0/devices/deleteDevGroupTagFromDevices?  
accessAppId={accessAppId}`

Header:

`"app_key: *****"`

`"Authorization: Bearer*****"`

`Content-Type: application/json`

Body:

{

`"devGroupId": "*****",`

`"deviceIds": [`

`"*****",`

`"*****",`

`"*****"`

`]`

}

## 响应示例

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议： <ul style="list-style-type: none"><li>● 请检查接口请求中的设备ID是否有误。</li><li>● 请检查设备组是否达到最大设备数量限制。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 1.2.9 设备升级

### 1.2.9.1 查询版本包列表

#### 典型场景

在进行设备版本升级前，应用服务器可调用此接口查询已经上传到物联网平台的版本升级包列表信息，确保设备要升级的版本包已上传到平台。

#### 接口功能

支持应用服务器在物联网平台查询已上传的版本包列表信息，可根据条件查询满足要求的版本包。

#### 接口原型

请求方法	GET
请求地址	https://server:port/iodm/northbound/v1.5.0/category? fileType={fileType}&deviceType={deviceType}&model={model}&m anufactureName={manufactureName}&version={version}
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。

参数	必选/可选	类型	位置	描述
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
fileType	可选	String(256)	query	版本包类型。 ● firmwarePackage: 固件包 ● softwarePackage: 软件包
deviceType	可选	String(256)	query	版本包适用的设备类型。
model	可选	String(256)	query	版本包适用的设备型号。
manufacturerName	可选	String(256)	query	版本包适用的设备厂商名称。
version	可选	String(256)	query	版本包的版本号。
pageNo	可选	Integer	query	分页查询参数，默认0，查询第一页，建议通过设置pageNo和pageSize做分页查询。
pageSize	可选	Integer	query	查询结果分页时的每页结果数量，取值范围1-100，默认值10。

## 响应参数

Status Code: 200 OK

参数	类型	描述
data	List<CategoryInfo>	版本包列表信息。
pageNo	Integer	查询结果的页码。
pageSize	Integer	查询结果每页的记录数量。
totalCount	Integer	查询结果的总记录数。

CategoryInfo:

参数	类型	描述
fileId	String	版本包ID。

参数	类型	描述
name	String	版本包名称。
version	String	版本包版本号。
fileType	String	版本包文件类型。 <ul style="list-style-type: none"><li>● firmwarePackage: 固件包</li><li>● softwarePackage: 软件包</li></ul>
deviceType	String	版本包适用的设备类型。
model	String	版本包适用的设备型号。
manufacturerName	String	版本包适用的设备厂商名称。
protocolType	String	版本包适用的设备协议类型
description	String	版本包的描述信息。
date	String	版本包的生成时间。
uploadTime	String	版本包的上传日期。

## 请求示例

Method: GET

request:

`https://server:port/iodm/northbound/v1.5.0/category?  
fileType={fileType}&deviceType={deviceType}&model={model}&manufactureName={ma  
nufactureName}&version={version}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

`Content-Type: application/json`

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

{

`"data": [`

{

```
"fileId": "*****",
"name": "*****",
"version": "V1.1.10",
"fileType": "softwarePackage",
"deviceType": "*****",
"model": "*****",
"manufacturerName": "***",
"protocolType": "CoAP",
"description": "Test software package made by WYH",
"date": "2017-08-11",
"uploadTime": ""20151212T121212Z"""

},
{
"fileId": "*****",
"name": "*****",
"version": "1.0",
"fileType": "firmwarePackage",
"deviceType": "WaterMeter",
"model": "17",
"manufacturerName": "*****",
"protocolType": "CoAP",
"description": null,
"date": "2017-11-11",
"uploadTime": ""20151212T121212Z"""

}
],
"pageNo": 0,
"pageSize": 2,
"totalCount": 2
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中pageNo和pageSize的值是否在正常范围内。
400	123029	pageNo or pageSize beyond the limit.	pageNo或pageSize超出上限。 处理建议：修改pageNo或pageSize为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 1.2.9.2 查询指定版本包

#### 典型场景

在进行设备版本升级前，应用服务器可调用此接口在物联网平台查询某个版本升级包的信息，确保设备要升级的版本包已上传到平台。

#### 接口功能

支持应用服务器在物联网平台根据版本包ID查询指定版本包信息，在调用该接口前需要通过“查询版本包列表”接口查询获得版本包ID。

#### 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iodm/northbound/v1.5.0/category/{fileId}">https://server:port/iodm/northbound/v1.5.0/category/{fileId}</a>
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。

参数	必选/可选	类型	位置	描述
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
fileId	必选	String	path	版本包ID，通过“查询版本包列表”接口查询获得。

## 响应参数

Status Code: 200 OK

参数	类型	描述
fileId	String	版本包ID。
name	String	版本包名称。
version	String	版本包的版本号。
fileType	String	版本包类型。 <ul style="list-style-type: none"><li>● firmwarePackage: 固件包</li><li>● softwarePackage: 软件包</li></ul>
deviceType	String	版本包适用的设备类型。
model	String	版本包适用的设备型号。
manufacturerName	String	版本包适用的设备厂商名称。
protocolType	String	版本包适用的设备协议类型。
description	String	版本包的描述信息。
date	String	版本包的生成时间。
uploadTime	String	版本包的上传日期。

## 请求示例

Method: GET

request:

`https://server:port/iodm/northbound/v1.5.0/category/{fileId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

Content-Type: application/json

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
    "fileId": "*****",  
    "name": "*****",  
    "version": "V1.1.10",  
    "fileType": "softwarePackage",  
    "deviceType": "*****",  
    "model": "*****",  
    "manufacturerName": "***",  
    "protocolType": "CoAP",  
    "description": "Test software package made by WYH",  
    "date": "2015-2-2 ",  
    "uploadTime": "20151212T121212Z"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。

Http状态码	错误码	错误描述	说明
404	123002	Device or package file not found.	设备或包不存在。 处理建议：请检查接口请求中的fileId是否正确。

### 1.2.9.3 删除指定版本包

#### 典型场景

对于不需要继续使用和保留的设备版本包，应用服务器可调用此接口在物联网平台上删除指定的版本包。

#### 接口功能

支持应用服务器在物联网平台根据版本包ID删除指定的版本包文件，在调用该接口前需要通过“查询版本包列表”接口查询获得要删除

#### 接口原型

请求方法	DELETE
请求地址	https://server:port/iodm/northbound/v1.5.0/category/{fileId}
传输协议	HTTPS

#### 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
fileId	必选	String	path	版本包ID，通过“查询版本包列表”接口查询获得。

#### 响应参数

Status Code: 200 OK

## 请求示例

Method: DELETE

request:

`https://server:port/iodm/northbound/v1.5.0/category/{fileId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

Content-Type: application/json

## 响应示例

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 1.2.9.4 创建软件升级任务

#### 典型场景

若需要对设备进行软件版本升级，应用服务器可调用此接口为多个设备创建软件升级任务，升级前请确保目标版本包已上传到物联网平台。当前仅支持对NB-IoT设备进行软件版本升级。

## 接口功能

支持应用服务器在物联网平台为多个设备进行软件版本升级，当前仅支持对NB-IoT设备进行软件版本升级。

## 接口原型

请求方法	POST
请求地址	<a href="https://server:port/iodm/northbound/v1.5.0/operations/softwareUpgrade">https://server:port/iodm/northbound/v1.5.0/operations/softwareUpgrade</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
fileId	必选	String	body	要升级的目标版本包ID。
targets	必选	<b>OperateDevices</b>	body	要进行升级的目标。
policy	可选	<b>OperatePolicy</b>	body	升级任务的执行策略。

OperateDevices:

参数	必选/可选	类型	位置	描述
deviceGroups	可选	List<String>	body	设备组名称列表，最多支持256个设备组。 与指定设备列表方式二选一。
deviceType	可选	String(256)	body	设备类型。 指定设备组时，此字段必选。
model	可选	String(256)	body	设备型号。 指定设备组时，此字段必选。

参数	必选/可选	类型	位置	描述
manufacturerName	可选	String(256)	body	设备厂商名称。 指定设备组时，此字段必选。
devices	可选	List<String>	body	设备ID列表，最多支持256个设备。 与指定设备组方式二选一。

OperatePolicy:

参数	必选/可选	类型	位置	描述
executeType	必选	String	body	执行类型，默认值为now。 <ul style="list-style-type: none"> <li>● now: 现在执行</li> <li>● device_online: 设备上线时执行</li> <li>● custom: 自定义</li> </ul>
startTime	可选	String	body	任务执行时间，executeType为custom时必选，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
endTime	可选	String	body	任务停止时间，executeType为custom时必选，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
retryType	可选	Boolean	body	执行失败是否进行重试，默认不重试。 <ul style="list-style-type: none"> <li>● true: 重试</li> <li>● false: 不重试</li> </ul>
retryTimes	可选	Integer[1, 5]	body	重试次数，取值范围1-5，retryType为true时必选。

## 响应参数

Status Code: 200ok

参数	类型	描述
operationId	String	操作任务ID。

## 请求示例

Method: POST

request:

https://server:port/iodm/northbound/v1.5.0/operations/softwareUpgrade

Header:

"app\_key: \*\*\*\*\*"

"Authorization:Bearer\*\*\*\*\*"

Content-Type: application/json

Body:

```
{  
  "fileId": "*****",  
  "targets":  
  {  
    "devices": ["*****"]  
  }  
}
```

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
  "operationId": "*****"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查请求参数的是否有误。

Http状态码	错误码	错误描述	说明
400	123016	The parameter is error, targetversion not match with device.	参数不正确，目标版本与指定设备不匹配。 处理建议：请检查接口请求中deviceType、manufacturerName、model，与参数fileId指定的目标版本包信息是否一致。
400	123019	manufacturerName is null.	厂商名称内容为空。 处理建议：请检查接口请求中的manufacturerName是否为空。
400	123020	deviceType is null	设备类型内容为空。 处理建议：请检查接口请求中的deviceType是否为空。
400	123021	model is null.	设备型号内容为空。 处理建议：请检查接口请求中的model是否为空。
400	123022	deviceGroups and devices cannot be null together	deviceGroups和devices不能同时为空。 处理建议：deviceGroups与devices必须要选择一个进行赋值。
400	123023	deviceGroups and devices cannot be exist together	deviceGroups和devices不能同时存在。 处理建议：deviceGroups与devices只能选择一个进行赋值。
400	123024	The number of deviceGroups or devices reached upper limit	deviceGroups或devices内容数量达到上限。 处理建议：请检查deviceGroups或devices的内容，数量不能超过256个。
400	123025	executeType is error or can not to be null.	executeType有误或不能为空。 处理建议：请检查接口请求中的executeType是否为空或者是否有误。
400	123026	startTime or endTime is null or error.	startTime或endTime为空或有误。 处理建议：请检查接口请求中的startTime和endTime是否为空或者格式有误。

Http状态码	错误码	错误描述	说明
400	123028	retryTimes is null or beyond the limit.	retryTimes为空或超出上限。 处理建议：请检查接口请求中的retryTimes是否为空，或者不能小于1或者大于5。
400	123032	startTime can not be later than the endTime.	startTime的时间不能比endTime晚。 处理建议：请检查接口请求中的startTime是否晚于endTime。
400	123033	startTime can not be earlier than the now.	startTime的时间不能比当前时间早。 处理建议：请检查接口请求中的startTime是否早于当前时间。
400	123034	endtime must be greater than 5 minutes.	endtime要比startTime大5分钟。 处理建议：请检查接口请求中的startTime和endTime间隔时间必须大于5分钟。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 1.2.9.5 创建固件升级任务

#### 典型场景

若需要对设备进行固件版本升级，应用服务器可调用此接口为多个设备创建固件升级任务，升级前请确保目标版本包已上传到物联网平台。当前仅支持对NB-IoT设备进行固件版本升级。

## 接口功能

支持应用服务器在物联网平台为多个设备进行固件版本升级，当前仅支持对NB-IoT设备进行固件版本升级。

## 接口原型

请求方法	POST
请求地址	https://server:port/iodm/northbound/v1.5.0/operations/firmwareUpgrade
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
fileId	必选	String	body	要升级的目标版本包ID。
targets	必选	OperateDevices	body	要进行升级的目标。
policy	可选	OperatePolicy	body	升级任务的执行策略。

OperateDevices:

参数	必选/可选	类型	位置	描述
deviceGroups	可选	List<String>	body	设备组名称列表，最多支持256个设备组。 与指定设备列表方式二选一。
deviceType	可选	String(256)	body	设备类型。 指定设备组时，此字段必选。
model	可选	String(256)	body	设备型号。 指定设备组时，此字段必选。

参数	必选/可选	类型	位置	描述
manufacturerName	可选	String(256)	body	设备厂商名称。 指定设备组时，此字段必选。
devices	可选	List<String>	body	设备ID列表，最多支持256个设备。 与指定设备组方式二选一。

OperatePolicy:

参数	必选/可选	类型	位置	描述
executeType	必选	String	body	执行类型，默认值为now。 <ul style="list-style-type: none"> <li>● now: 现在执行</li> <li>● device_online: 设备上线时执行</li> <li>● custom: 自定义</li> </ul>
startTime	可选	String	body	任务执行时间，executeType为custom时必选，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
endTime	可选	String	body	任务停止时间，executeType为custom时必选，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
retryType	可选	Boolean	body	执行失败是否进行重试，默认不重试。 <ul style="list-style-type: none"> <li>● true: 重试</li> <li>● false: 不重试</li> </ul>
retryTimes	可选	Integer[1,5]	body	重试次数, 取值范围1-5, retryType为true时必选。

## 响应参数

Status Code: 200ok

参数	类型	描述
operationId	String	操作任务ID。

## 请求示例

Method: POST

request:

<https://server:port/iodm/northbound/v1.5.0/operations/firmwareUpgrade>

Header:

"app\_key: \*\*\*\*\*"

"Authorization: Bearer\*\*\*\*\*"

Content-Type: application/json

Body:

```
{  
    "fileId": "*****",  
    "targets":  
    {  
        "devices": ["*****"]  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

```
{  
    "operationId": "*****"  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查请求参数的是否有误。

Http状态码	错误码	错误描述	说明
400	123016	The parameter is error, targetversion not match with device.	参数不正确，目标版本与指定设备不匹配。 处理建议：请检查接口请求中deviceType、manufacturerName、model，与参数fileId指定的目标版本包信息是否一致。
400	123019	manufacturerName is null.	厂商名称内容为空。 处理建议：请检查接口请求中的manufacturerName是否为空。
400	123020	deviceType is null	设备类型内容为空。 处理建议：请检查接口请求中的deviceType是否为空。
400	123021	model is null.	设备型号内容为空。 处理建议：请检查接口请求中的model是否为空。
400	123022	deviceGroups and devices cannot be null together	deviceGroups和devices不能同时为空。 处理建议：deviceGroups与devices必须要选择一个进行赋值。
400	123023	deviceGroups and devices cannot be exist together	deviceGroups和devices不能同时存在。 处理建议：deviceGroups与devices只能选择一个进行赋值。
400	123024	The number of deviceGroups or devices reached upper limit	deviceGroups或devices内容数量达到上限。 处理建议：请检查deviceGroups或devices的内容，数量不能超过256个。
400	123025	executeType is error or can not to be null.	executeType有误或不能为空。 处理建议：请检查接口请求中的executeType是否为空或者是是否有误。
400	123026	startTime or endTime is null or error.	startTime或endTime为空或有误。 处理建议：请检查接口请求中的startTime和endTime是否为空或者格式有误。

Http状态码	错误码	错误描述	说明
400	123028	retryTimes is null or beyond the limit.	retryTimes为空或超出上限。 处理建议：请检查接口请求中的retryTimes是否为空，或者不能小于1或者大于5。
400	123032	startTime can not be later than the endTime.	startTime的时间不能比endTime晚。 处理建议：请检查接口请求中的startTime是否晚于endTime。
400	123033	startTime can not be earlier than the now.	startTime的时间不能比当前时间早。 处理建议：请检查接口请求中的startTime是否早于当前时间。
400	123034	endtime must be greater than 5 minutes.	endtime要比startTime大5分钟。 处理建议：请检查接口请求中的startTime和endTime间隔时间必须大于5分钟。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 1.2.9.6 查询指定任务信息

#### 典型场景

在创建了设备软件升级任务或固件升级任务后，应用服务器可调用此接口查询某个升级任务详细信息，以查看升级任务的配置信息和执行情况等。

## 接口功能

支持应用服务器在物联网平台查询单个固件或软件升级任务的详细信息，包括升级任务的配置信息和执行情况等。

## 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iodm/northbound/v1.5.0/operations/{operationId}">https://server:port/iodm/northbound/v1.5.0/operations/{operationId}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
operationId	必选	String	path	操作任务ID，在创建操作任务后由物联网平台返回获得。

## 响应参数

Status Code: 200 OK

参数	类型	描述
operationId	String	操作任务ID。
createTime	String	操作任务的创建时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。。
startTime	String	操作任务的启动时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。。
stopTime	String	操作任务的停止时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。。

参数	类型	描述
operateType	String	操作类型。 <ul style="list-style-type: none"><li>● firmware_upgrade</li><li>● software_upgrade</li></ul>
targets	<b>OperateDevices</b>	执行操作的目标设备。
policy	<b>OperatePolicy</b>	操作执行策略。
status	String	操作任务的状态。 <ul style="list-style-type: none"><li>● wait: 等待</li><li>● processing: 正在执行</li><li>● failed: 失败</li><li>● success: 成功</li><li>● stop: 停止</li></ul>
staResult	<b>OperationStaResult</b>	操作结果统计。
extendPara	Map<String, String>	操作扩展参数，视不同类型的操作不同。

OperateDevices:

参数	类型	描述
deviceGroups	List<String>	设备组名称列表，最多支持256个设备组。 与指定设备列表方式二选一。
deviceType	String	设备类型。 指定设备组时，此字段必选。
model	String	设备型号。 指定设备组时，此字段必选。
manufacturerName	String	设备厂商名称。 指定设备组时，此字段必选。
devices	List<String>	设备ID列表，最多支持256个设备。 与指定设备组方式二选一。

OperatePolicy:

参数	类型	描述
executeType	String	执行类型, 默认值为now。 ● now: 现在执行 ● device_online: 设备上线时执行 ● custom: 自定义
startTime	String	任务执行时间, executeType=custom时必选, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
endTime	String	任务停止时间, executeType=custom时必选, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
retryType	Boolean	执行失败是否进行重试, 默认不重试。 ● true: 重试 ● false: 不重试
retryTimes	Integer	重试次数, 取值范围1-5, retryType=true时必选。

OperationStaResult:

参数	类型	描述
total	Integer(64)	操作设备总数。
wait	Integer(64)	等待操作的设备个数。
processing	Integer(64)	正在操作的设备个数。
success	Integer(64)	操作成功的设备个数。
fail	Integer(64)	操作失败的设备个数。
stop	Integer(64)	停止操作的设备个数。
timeout	Integer(64)	操作超时失败的设备个数。

## 请求示例

Method: GET

request:

`https://server:port/iodm/northbound/v1.5.0/operations/{operationId}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

Content-Type: application/json

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
    "operationId": "*****",  
    "createTime": "20151212T121212Z",  
    "startTime": "20151212T121212Z",  
    "stopTime": null,  
    "operateType": "software_upgrade",  
    "targets": {  
        "deviceGroups": null,  
        "deviceType": "*****",  
        "model": "*****",  
        "manufacturerName": "***",  
        "devices": [  
            "*****"  
        ]  
    },  
    "policy": null,  
    "status": "FAIL",  
    "staResult": {  
        "wait": 0,  
        "processing": 0,  
        "success": 0,  
        "fail": 1,  
        "stop": 0,  
        "timeout": 0  
    },  
    "extendPara": {  
        "fileVersion": "V1.1.10"  
    }  
}
```

```
    }  
}  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken 是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的 operationId 内容是否正确。

### 1.2.9.7 查询指定任务的子任务详情

#### 典型场景

在创建了设备软件升级任务或固件升级任务后，任务中涉及的每个设备的升级是一个子任务（即升级任务中涉及多少个设备，就有多少个子任务）。应用服务器可调用此接口查询某个升级任务中各个子任务的详细信息，以查看子任务的具体执行情况。

#### 接口功能

支持应用服务器在物联网平台查询单个固件或软件升级任务中每个设备的升级执行情况。

#### 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iodm/northbound/v1.5.0/operations/{operationId}/subOperations?subOperationStatus={subOperationStatus}&amp;pageNo={pageNo}&amp;pageSize={pageSize}">https://server:port/iodm/northbound/v1.5.0/operations/{operationId}/subOperations?subOperationStatus={subOperationStatus}&amp;pageNo={pageNo}&amp;pageSize={pageSize}</a>
传输协议	HTTPS

## 请求参数

参数	必选/ 可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
operationId	必选	String	path	操作任务ID，在创建操作任务后由物联网平台返回获得。
subOperation Status	可选	String	query	子任务状态，不指定，则查询该任务下所有子任务执行详情。 <ul style="list-style-type: none"> <li>● wait: 等待</li> <li>● processing: 正在执行</li> <li>● fail: 失败</li> <li>● success: 成功</li> <li>● stop: 停止</li> </ul>
pageNo	可选	Integer(>=0)	query	分页查询参数，默认0，查询第一页，建议通过设置pageNo和pageSize做分页查询。
pageSize	可选	Integer[1, 100]	query	查询结果分页时的每页结果数量，取值范围1-100，默认值10。

## 响应参数

Status Code: 200 OK

参数	类型	描述
data	List< <a href="#">SubOperation Info</a> >	子任务列表信息。
pageNo	Integer(>=0)	查询结果的页码。
pageSize	Integer[1,100]	查询结果每页的记录数量。
totalCount	Integer(>=0)	查询结果的记录总数。

SubOperationInfo:

参数	类型	描述
subOperationId	String	子任务ID。
createTime	String	子任务创建时间，时间格式: yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。。
startTime	String	子任务启动时间，时间格式: yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。。
stopTime	String	子任务停止时间，时间格式: yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。。
operateType	String	操作类型。 <ul style="list-style-type: none"><li>● firmware_upgrade</li><li>● software_upgrade</li></ul>
deviceId	String	操作设备的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
status	String	子任务状态。 <ul style="list-style-type: none"><li>● wait: 等待</li><li>● processing: 正在执行</li><li>● fail: 失败</li><li>● success: 成功</li><li>● stop: 停止</li></ul>
detailInfo	String	任务状态的详细描述，对于失败场景下为失败原因。
extendPara	Map<String, String>	任务扩展信息，视不同类型的操作不同。

## 请求示例

Method: GET

request:

`https://server:port/iodm/northbound/v1.5.0/operations/{operationId}/subOperations?  
subOperationStatus={subOperationStatus}&pageNo={pageNo}&pageSize={pageSize}`

Header:

`"app_key: *****"`

`"Authorization:Bearer*****"`

`Content-Type: application/json`

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
    "totalCount": 1,  
    "pageNo": 0,  
    "pageSize": 1,  
    "date": [  
        {  
            "subOperationId": "*****",  
            "createTime": "20151212T121212Z",  
            "startTime": "20151212T121212Z",  
            "stopTime": null,  
            "operateType": "software_upgrade",  
            "deviceId": "*****",  
            "status": "FAIL",  
            "detailInfo": "The task failed to start, unable to find protocol service based on device information",  
            "extendInfo": {  
                "upgradeTime": null,  
                "sourceVersion": null,  
                "curVersion": "V1.1.10",  
                "downloadTime": null,  
                "targetVersion": null  
            }  
        }  
    ]  
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
400	123029	pageNo or pageSize beyond the limit.	pageNo 或 pageSize 超出上限。 处理建议：修改 pageNo 或 pageSize 为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的 token 信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的 operationId 内容是否正确。

### 1.2.9.8 查询任务列表

#### 典型场景

应用服务器可调用此接口查询已创建的升级任务列表信息，以了解当前已有的升级任务信息及各个任务的执行情况。

#### 接口功能

支持应用服务器在物联网平台查询升级任务的列表信息，可根据条件查询满足要求的升级任务信息。

#### 接口原型

请求方法	GET
请求地址	<a href="https://server:port/iodm/northbound/v1.5.0/operations?operationType={operationType}&amp;operationStatus={operationStatus}&amp;deviceType={deviceType}&amp;manufacturerName={manufacturerName}&amp;model={model}&amp;deviceId={deviceId}">https://server:port/iodm/northbound/v1.5.0/operations? operationType={operationType}&amp;operationStatus={operationStatus}&amp; deviceType={deviceType}&amp;manufacturerName={manufacturerName}&amp; model={model}&amp;deviceId={deviceId}</a>
传输协议	HTTPS

## 请求参数

参数	必选/可选	类型	位置	描述
app_key	必选	String	header	访问物联网平台的应用ID，在物联网平台创建应用时由平台分配获得。
Authorization	必选	String	header	访问物联网平台的认证信息，值为“Bearer {accessToken}”，其中{accessToken}为调用鉴权接口返回的accessToken。
operationType	可选	String(256)	query	操作类型。 <ul style="list-style-type: none"><li>● firmware_upgrade</li><li>● software_upgrade</li></ul>
operationStatus	可选	String(256)	query	操作任务的状态。 <ul style="list-style-type: none"><li>● wait: 等待</li><li>● processing: 正在执行</li><li>● failed: 失败</li><li>● success: 成功</li><li>● stop: 停止</li></ul>
deviceType	可选	String(256)	query	操作任务针对的设备类型。
model	可选	String(256)	query	操作任务针对的设备型号。
manufacturerName	可选	String(256)	query	操作任务针对的设备厂家名称。
deviceId	可选	String(256)	query	操作任务针对的设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
pageNo	可选	Integer(>=0)	query	分页查询参数，默认0，查询第一页，建议通过设置pageNo和pageSize做分页查询。
pageSize	可选	Integer[1, 100]	query	查询结果分页时的每页结果数量，取值范围1-100，默认值10。

## 响应参数

Status Code: 200 OK

参数	类型	描述
data	List< <b>OperationInfo</b> <b>o</b> >	任务列表信息。
pageNo	Integer(>=0)	查询结果的页码。
pageSize	Integer[1,100]	查询结果每页的记录数量。
totalCount	Integer(>=0)	查询结果的总记录数。

OperationInfo:

参数	类型	描述
operationId	String	操作任务ID。
createTime	String	操作任务的创建时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。。
startTime	String	操作任务的启动时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。。
stopTime	String	操作任务的停止时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。。
operateType	String	操作类型。 <ul style="list-style-type: none"><li>● firmware_upgrade</li><li>● software_upgrade</li></ul>
targets	<b>OperateDevices</b>	执行操作的目标设备。
policy	<b>OperatePolicy</b>	操作执行策略。
status	String	操作任务的状态。 <ul style="list-style-type: none"><li>● wait: 等待</li><li>● processing: 正在执行</li><li>● failed: 失败</li><li>● success: 成功</li><li>● stop: 停止</li></ul>
staResult	<b>OperationStaResu lt</b>	操作结果统计。
extendPara	Map<String, String>	操作扩展参数，视不同类型的操作不同。

OperateDevices:

参数	类型	描述
deviceGroups	List<String>	设备组名称列表，最多支持256个设备组。 与指定设备列表方式二选一。
deviceType	String	设备类型。 指定设备组时，此字段必选。
model	String	设备型号。 指定设备组时，此字段必选。
manufacturerName	String	设备厂商名称。 指定设备组时，此字段必选。
devices	List<String>	设备ID列表，最多支持256个设备。 与指定设备组方式二选一。

## OperatePolicy:

参数	类型	描述
executeType	String	执行类型，默认值为now。 <ul style="list-style-type: none"><li>● now: 现在执行</li><li>● device_online: 设备上线时执行</li><li>● custom: 自定义</li></ul>
startTime	String	任务执行时间，executeType=custom时必选， 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
endTime	String	任务停止时间，executeType=custom时必选， 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
retryType	Boolean	执行失败是否进行重试，默认不重试。 <ul style="list-style-type: none"><li>● true: 重试</li><li>● false: 不重试</li></ul>
retryTimes	Integer	重试次数, 取值范围1-5, retryType=true时必选。

## OperationStaResult:

参数	类型	描述
total	Integer(64)	操作设备总数。
wait	Integer(64)	等待操作的设备个数。

参数	类型	描述
processing	Integer(64)	正在操作的设备个数。
success	Integer(64)	操作成功的设备个数。
fail	Integer(64)	操作失败的设备个数。
stop	Integer(64)	停止操作的设备个数。
timeout	Integer(64)	操作超时失败的设备个数。

## 请求示例

Method: GET

request:

```
https://server:port/iodm/northbound/v1.5.0/operations?  
operationType={operationType}&operationStatus={operationStatus}&deviceType={deviceT  
ype}&manufacturerName={manufacturerName}&model={model}&deviceId={deviceId}
```

Header:

"app\_key: \*\*\*\*\*"

"Authorization:Bearer\*\*\*\*\*"

Content-Type: application/json

## 响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

```
{  
    "totalCount": 1,  
    "pageNo": 0,  
    "pageSize": 1,  
    "data": [  
        {  
            "operationId": "*****",  
            "createTime": "20151212T121212Z",  
            "startTime": "20151212T121212Z",  
            "stopTime": null,  
            "status": "Success",  
            "deviceType": "DeviceA",  
            "manufacturerName": "Huawei",  
            "model": "ModelA",  
            "deviceId": "DeviceID1",  
            "processing": 1,  
            "success": 1,  
            "fail": 0,  
            "stop": 0,  
            "timeout": 0  
        }  
    ]  
}
```

```
"operateType": "software_upgrade",
"targets": {
    "deviceGroups": null,
    "deviceType": "*****",
    "model": "*****",
    "manufacturerName": "***",
    "devices": [
        "*****"
    ],
},
"policy": null,
"status": "FAIL",
"staResult": {
    "wait": 0,
    "processing": 0,
    "success": 0,
    "fail": 1,
    "stop": 0,
    "timeout": 0
},
"extendPara": {
    "fileVersion": "V1.1.10"
}
}
```

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。

Http状态码	错误码	错误描述	说明
400	123029	pageNo or pageSize beyond the limit.	pageNo或pageSize超出上限。 处理建议：修改pageNo或pageSize为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的operationId内容是否正确。

# 2 IoT 平台北向 JAVA SDK API 参考

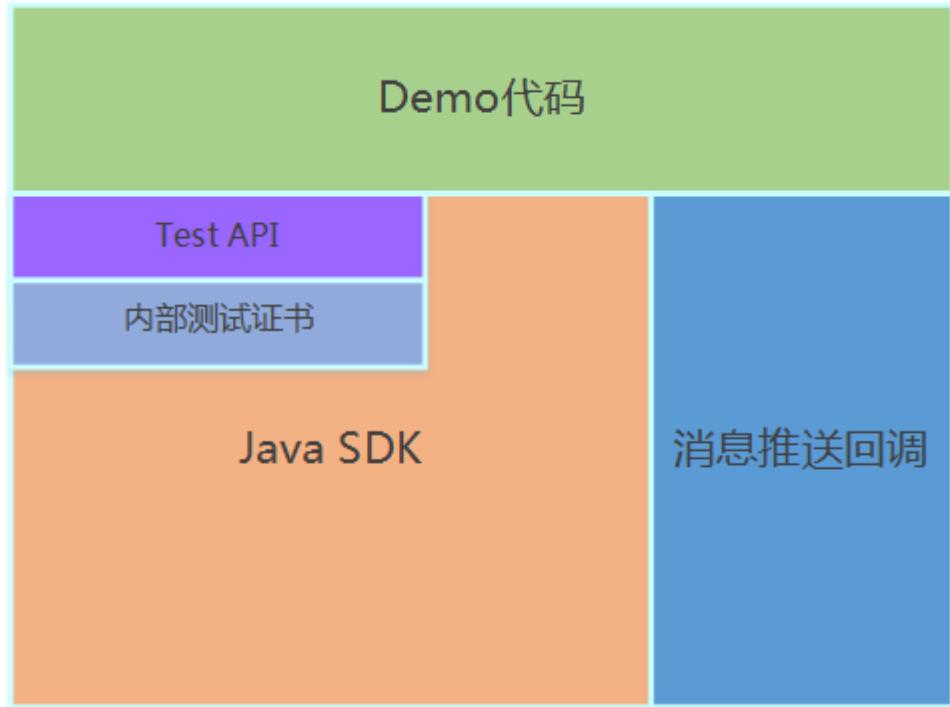
[SDK Demo 总体架构及使用说明](#)

[SDK 初始化配置及测试](#)

[业务接口列表](#)

## 2.1 SDK Demo 总体架构及使用说明

- Demo代码是调用SDK接口的样例代码，包括初始化及各个接口调用，仅供开发者参考。
- SDK提供JAVA方法调用平台北向Restful接口与平台通信。
- 消息推送回调使用JAVA代码实现回调接口，供平台推送消息给应用服务器时调用，应用通过继承PushMessageReceiver类并重写类中的方法接收推送消息的内容。
- Test API提供测试SDK接口的功能，主要测试SDK与平台之间的基本功能是否可用，并输出测试结果。测试时会分别使用内部测试证书和开发者设置的证书，以便排查证书问题。



## 2.2 SDK 初始化配置及测试

### 2.2.1 NorthApiClient 类方法

用于创建应用实例，是调用SDK其他接口的前提。主要方法如下：

方法	说明
<code>public void setClientInfo( ClientInfo clientInfo)</code>	初始化NorthApiClient中的参数， ClientInfo的定义参考 <a href="#">Clientinfo类方法</a> 。
<code>public void setHttpConnection(int httpSocketTimeout, int httpConnectionTimeout, int maxConnectionAmount, int maxConnectionAmountPerRoute)</code>	HTTP连接池设置，可设置socket超时时间（单位毫秒）、连接超时时间（单位毫秒）、最大连接数和每路由最大连接数。这4个参数的默认值分别为30000、30000、200、200。 需要在 <a href="#">initSSLConfig()</a> 或 <a href="#">initSSLConfig(SSLConfig sslConfig)</a> 前调用。
<code>public void initSSLConfig()</code>	初始化双向认证的配置，调用后才可以使用其他方法。 <b>注意</b> 本方法使用了测试证书，非正式证书，且不做主机名校验，故仅在集成对接的调试阶段使用。

方法	说明
public void initSSLConfig(SSLConfig sslConfig)	<p>初始化双向认证的配置，调用后才可以使用其他方法。SSLConfig的定义参考<a href="#">SSLConfig类方法</a>。</p> <p><b>注意</b>            本方法用于导入证书，可以在商用阶段使用。在调用本方法前可调用setHostnameVerifier(HostnameVerifier hostnameVerifier)方法进行主机名校验，否则默认使用严格主机名校验。</p>
public void setHostnameVerifier(HostnameVerifier hostnameVerifier)	<p>自定义主机名校验方法，在<a href="#">initSSLConfig(SSLConfig sslConfig)</a>方法前调用，否则默认使用严格主机名校验。</p> <p><b>注意</b></p> <ul style="list-style-type: none"> <li>如果商用证书中的主机名与平台环境的主机名一致，可以不调用本方法，直接默认使用严格主机名校验。</li> <li>主机名校验方法应以安全为原则，不应该直接返回true。</li> </ul>
public String getVersion()	查询SDK版本号。

## 2.2.2 Clientinfo 类方法

用于设置与物联网平台对接的基本信息。主要方法（省略get方法）如下：

方法	说明
public void setPlatformIp(String platformIp)	设置物联网平台的IP地址。
public void setPlatformPort(String platformPort)	设置物联网平台的端口，如8743。
public void setAppId(String appId)	设置应用的appId。appId是物联网平台为第三方应用分配的身份标识，用于唯一标识一个应用。
public void setSecret(String secret)	设置应用的secret。secret是与平台appId配套的密码。

## 2.2.3 NorthApiException 类方法

当SDK的处理有异常，或者物联网平台对SDK的请求处理有异常时，抛出NorthApiException类方法。主要方法（省略set方法）如下：

方法	说明
public String geterror_code()	获取异常的错误码，SDK自身处理异常会抛出四位数字的错误码，物联网平台处理异常会抛出六位数字的错误码。

方法	说明
public String getError_desc()	获取异常错误码对应的错误描述。
public String getHttpStatusCode()	获取异常http状态码。 本SDK封装了http restful API，因此当SDK在调用restful API异常时，会得到异常的http状态码，如404、200等。
public String getHttpReasonPhrase()	获取异常http状态码对应的原因解释（ReasonPhrase）。 如http状态码为400时，原因解释为“Not Found”；StatusCode为200时，原因解释为“OK”。
public String getHttpMessage()	当物联网平台处理请求异常，并且异常中未包含错误码时，将该异常的内容通过本方法返回。

## 2.2.4 SSLConfig 类方法

用于设置证书路径及密码。主要方法（省略get方法）如下：

方法	说明
public String setSelfCertPath(String selfCertPath)	设置客户端的证书位置的绝对路径。
public String setSelfCertPwd(String selfCertPwd)	设置客户端证书的密码。
public String setTrustCAPath(String trustCAPath)	设置服务端CA证书位置的绝对路径。
public String setTrustCAPwd(String trustCAPwd)	设置服务端CA证书的密码。

## 2.2.5 BasicTest 类方法

用于测试证书、登录、注册设备等基本功能是否正常。方法如下：

方法	说明
public static void beginBasicTest(String platformIp, String platformPort, String appId, String secret, SSLConfig sslConfig)	用户设置好平台IP、端口、appId、密码、商用证书（可选参数）后，可使用该方法测试证书、登录、注册设备等基本功能是否正常，如有异常则给出原因及建议（在控制台打印输出）。

## 2.3 业务接口列表

### 2.3.1 应用安全接入

第三方应用获取鉴权信息，接入物联网平台，随后携带鉴权信息调用其他API接口。

#### 2.3.1.1 鉴权

##### 典型场景

第三方应用首次访问物联网平台的开放API时，需调用此接口完成接入认证；第三方应用在物联网平台的认证过期后，需调用此接口重新进行认证，才能继续访问物联网平台的开放API。

##### 接口功能

支持第三方应用在首次访问物联网平台的开放API之前，完成在物联网平台的接入认证。

##### 注意事项

鉴权接口是调用其他API的前提，除了调用鉴权接口（Auth），其他接口调用都需要使用鉴权接口获取的accessToken。

如果多次获取accessToken，则之前的accessToken失效，后一次获取的accessToken才有效。请勿并发获取accessToken。

##### 接口描述

```
AuthOutDTO getAuthToken() throws NorthApiException
```

##### 接口所属类

Authentication

##### 参数说明

appId和secret使用[NorthApiClient](#)成员变量的[Clientinfo类方法](#)中的值。

## 返回值

AuthOutDTO

参数	类型	描述
scope	String(256)	申请权限范围，即accessToken所能访问物联网平台资源的范围，参数值固定为default。
tokenType	String(256)	accessToken的类型，参数值固定为bearer。
expiresIn	Integer(256)	平台生成并返回accessToken的有效时间，单位为秒。
accessToken	String(256)	鉴权参数，访问物联网平台API接口的凭证。
refreshToken	String(256)	鉴权参数，用来刷新accessToken，refreshToken的有效时间为1个月。

## 错误码

Http状态码	错误码	错误描述	说明
400	100449	The device is freezed can't operate.	用户无操作权限。 处理建议：请检查appId所对应用是否有接口调用权限。
400	102202	Required Parameter is null or empty.	必选参数为空。 处理建议：请检查请求参数中的必选参数是否已填写。
401	100208	AppId or secret is not right.	appId或secret错误。 处理建议： <ul style="list-style-type: none"><li>请检查appId和secret是否有误，是否新增字符或缺少字符。</li><li>请检查请求路径中的IP地址是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.1.2 刷新 token

#### 典型场景

第三方应用通过鉴权接口获取到的accessToken是有有效时间的，在accessToken快过期时，第三方应用通过调用此接口，获取新的accessToken。

## 接口功能

支持第三方应用在accessToken到期前，从物联网平台获取新的accessToken。

## 接口描述

```
AuthRefreshOutDTO refreshAuthToken(AuthRefreshInDTO arInDTO) throws NorthApiException
```

## 接口所属类

Authentication

## 参数说明

AuthRefreshInDTO

参数	必选/可选	类型	位置	描述
appId	必选	String(256)	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。appid在物联网平台的SP Portal上创建应用时获得。
secret	必选	String(256)	body	第三方应用的密码，与appId对应，用于登录访问物联网平台。secret在物联网平台的SP Portal上创建应用时获得。
refreshToken	必选	String(256)	body	刷新token，用来获取一个新的accessToken。refreshToken在调用鉴权接口时获得。

## 返回值

AuthRefreshOutDTO

参数	类型	描述
scope	String(256)	申请的权限范围，参数值固定为default。
tokenType	String(256)	鉴权token的类型，参数值固定为bearer。
expiresIn	Integer(256)	平台生成并返回accessToken的有效时间，单位为秒。
accessToken	String(256)	鉴权参数，访问物联网平台API接口的凭证。
refreshToken	String(256)	鉴权参数，用来刷新accessToken，refreshToken的有效时间为1个月。

## 错误码

Http状态码	错误码	错误描述	说明
400	100449	The device is freezed cant operate.	用户无操作权限。 处理建议：请检查appId所对应用是否有接口调用权限。
400	102202	Required Parameter is null or empty.	必选参数为空。 处理建议：请检查请求参数中的必选参数是否已填写。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100208	AppId or secret is not right.	appId或secret错误。 处理建议： <ul style="list-style-type: none"><li>● 请检查appId和secret是否有误，是否新增字符或缺少字符。</li><li>● 请检查请求路径中的IP地址是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.1.3 定时刷新 token

#### 典型场景

第三方应用通过调用此接口，定期刷新accessToken，刷新周期为[鉴权](#)方法返回值中的expiresIn，以保证accessToken不过期。accessToken由SDK管理，调用其他业务接口，accessToken可以直接传入null。

#### 接口功能

在accessToken到期前，自动获取新的accessToken。

#### 注意事项

1. 调用此接口后，SDK会启动一个线程自动刷新accessToken，且accessToken只在SDK内部管理，第三方应用不必关心；调用其他业务接口时，accessToken直接传入null即可（如果传入的accessToken不是null，则使用传入的accessToken）。
2. 可以调用[stopRefreshTokenTimer\(\)](#)方法停止线程，一旦调用了stopRefreshTokenTimer()方法，调用其他业务接口，accessToken不可为null，accessToken转由第三方应用自行管理。

## 接口描述

```
void startRefreshTokenTimer() throws NorthApiException
```

## 接口所属类

Authentication

## 参数说明

appId和secret直接使用[NorthApiClient](#)成员变量的[ClientInfo](#)中的值

## 返回值

void

### 2.3.1.4 停止定时刷新 token

## 典型场景

第三方应用通过调用此接口，停止定期刷新accessToken，一旦调用了stopRefreshTokenTimer方法，调用其他业务接口，accessToken不可为null。

## 接口功能

停止自动获取新的accessToken，并关闭stopRefreshTokenTimer方法启动的线程。

## 注意事项

一旦调用了stopRefreshTokenTimer方法，调用其他业务接口，accessToken不可为null。

## 接口描述

```
void stopRefreshTokenTimer()
```

## 接口所属类

Authentication

## 参数说明

无

## 返回值

void

### 2.3.2 设备管理

第三方应用向物联网平台添加设备，获取设备的ID和验证码，待设备完成接入物联网平台流程后，设备与第三方应用建立从属关系。

### 2.3.2.1 注册直连设备

#### 典型场景

在直连设备接入物联网平台前，第三方应用需要调用此接口在物联网平台注册设备，并把设备的唯一标识码（如IMEI）设置为设备接入平台的验证码。在设备接入物联网平台时携带设备唯一标识，完成设备的接入认证。

#### 接口功能

支持第三方应用在物联网平台注册直连设备，完成注册后设备方可接入物联网平台。

#### 接口描述

```
RegDirectDeviceOutDTO regDirectDevice(RegDirectDeviceInDTO2 rddInDto, String appId, String accessToken) throws NorthApiException
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。
rddInDto	必选	RegDirectDeviceInDTO2	body	具体参见 <a href="#">RegDirectDeviceInDTO2</a> 结构体的描述。

RegDirectDeviceInDTO2结构体：

参数	必选/可选	类型	位置	描述
deviceInfo	可选	DeviceInfoDTO	Body	设备信息。见下方DeviceInfo结构体说明。
imsi	可选	String(1-64)	Body	NB-IoT终端的IMSI。

参数	必选/可选	类型	位置	描述
isSecure	可选	Boolean	Body	<p>指定设备是否为安全设备， 默认值为“false”。</p> <ul style="list-style-type: none"><li>● “true”：安全设备</li><li>● “false”：非安全设备</li></ul> <p><b>说明</b> 如果用户需要注册安全设备， isSecure参数必须填写。</p>
verifyCode	可选	String(256)	body	<p>设备验证码，若在请求中指定 verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。</p> <p>在NB-IoT方案中，verifyCode为必填参数，且必须与nodeId设置成相同值。</p>
nodeId	必选	String(256)	body	<p>设备的唯一标识，必须与设备上报的设备标识一致。通常使用 MAC地址，Serial No或IMEI作为nodeId。</p> <p><b>说明</b> 使用IMEI作为nodeId时，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为 urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为 IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
endUserId	可选	String(256)	body	<p>终端用户ID。</p> <p>在NB-IoT方案中，endUserId设置为设备的IMSI号。在 Smarthome解决方案中，endUserId设置为App账号。</p>
psk	可选	String(8~32)	body	<p>请求中指定psk，则平台使用指定的psk；请求中不指定psk，则由平台生成psk。取值范围是a-f、A-F、0-9组成的字符串。</p>

参数	必选/可选	类型	位置	描述
timeout	可选	Integer(>=0)	Body	<p>超时时间。当调用北向接口对设备开户，在超时时间内可绑定设备，若超过timeout时间且未绑定设备，则会删除超时的开户信息。</p> <p>取值范围：0~2147483647。若填写为“0”，则表示设备验证码不会失效（建议填写为“0”）。</p> <p>默认值：“180”（默认值可配置，具体配置值请咨询物联网平台运维人员。）</p> <p>单位：秒。</p>
productId	可选	String(256)	Body	设备所属的产品ID。

DeviceInfoDTO:

参数	必选/可选	类型	位置	描述
manufacturerId	可选	String(256)	Body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	Body	厂商名称。
deviceType	可选	String(256)	Body	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	必选	String(256)	Body	<p>设备的型号。</p> <p>Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。</p>
protocolType	可选	String(256)	Body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 返回值

RegDirectDeviceOutDTO

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
verifyCode	String(256)	设备验证码，设备可以通过验证码完成物联网平台的接入认证。若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。
timeout	Integer	验证码有效时间，单位秒，设备需要在有效时间内接入物联网平台。若为“0”，则表示设备验证码不会失效。
psk	String(32)	随机psk参数，若请求中携带了psk，则使用请求中的psk，否则由平台生成随机psk参数。

## 错误码

Http状态码	错误码	错误描述	说明
200	103028	The license pool resources.	License资源用尽。
400	100003	Invalid verify code.	验证码无效。 处理建议：请检查接口请求中的verifyCode是否有误。若请求参数中未带verifyCode，请联系物联网平台维护人员处理。
400	100007	Bad request message.	参数不合法。 处理建议：deviceId未赋值，请参考请求参数说明填写请求。
400	100416	The device has already been binded.	设备已经绑定。 处理建议：请检查设备是否已经注册。
400	100426	The nodeId is duplicated.	nodeId重复。 处理建议：请检查接口请求中nodeId是否有误。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
401	100025	AppId for auth not exist.	获取不到appId鉴权信息。 处理建议： <ul style="list-style-type: none"><li>● 请检查请求结构体的Header头域中是否给字段app_key赋值。</li><li>● 若通过HTTP调用，请联系物联网平台维护人员确认Header头域中放置appId的字段名是app_key还是x-app-key。</li></ul>
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
403	600002	The product not existed.	产品不存在。 处理建议：物联网平台未找到productId对应的产品，请检查productId是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100203	The application is not existed.	授权应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	100412	The amount of device has reached the limit.	当前应用下设备数量达到上限。 处理建议：请检查当前应用下的设备数量是否已达到所申请资源的上限。
500	100441	The amount of nonSecure device has reached the limit.	非安全设备数量已达到上限。
500	103026	The license is not exist.	License不存在。 处理建议：物联网平台内部License问题，请联系物联网平台维护人员处理。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.2.2 刷新设备密钥

#### 典型场景

已在物联网平台注册的设备，若设备的唯一标识码发生变更（如更换设备等场景），第三方应用需要调用此接口在物联网平台更新设备的唯一标识，并重新绑定设备。



##### 说明

仅当设备处于离线状态时，才能进行刷新设备密钥操作。

#### 接口功能

支持第三方应用在物联网平台更新已注册设备的nodeId，在设备deviceId不变的情况下，重新绑新设备。

#### 接口描述

```
RefreshDeviceKeyOutDTO refreshDeviceKey(RefreshDeviceKeyInDTO rdkInDTO, String deviceId, String appId, String accessToken) throws NorthApiException
```

#### 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String(256)	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。
rdkInDTO	必选	RefreshDeviceKeyInDTO	body	具体参见 <a href="#">RefreshDeviceKeyInDTO</a> 结构体的描述。

RefreshDeviceKeyInDTO结构体：

参数	必选/可选	类型	位置	描述
verifyCode	可选	String(256)	body	备验证码，若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。建议和nodeId设置成一样。
nodeId	可选	String(256)	body	<p>设备的唯一标识，通常使用MAC，MAC地址，Serial No或IMEI作为nodeId。</p> <ul style="list-style-type: none"> <li>● 值空时，nodeId不变。</li> <li>● 值不为空时，更新nodeId。</li> </ul> <p><b>说明</b></p> <p>若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"> <li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li> <li>● 海思芯片设备的唯一标识为IMEI号</li> <li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li> </ul>
timeout	可选	Integer	body	<p>验证码超时时间，单位秒，取值大于等于0的整数。</p> <ul style="list-style-type: none"> <li>● 值为空时，使用默认值（180s）。</li> <li>● 值为0时，永不过期。</li> <li>● 值为非0，指定时间。</li> </ul>

## 返回值

RefreshDeviceKeyOutDTO

参数	类型	描述
verifyCode	String(256)	设备验证码，设备可以通过验证码完成物联网平台的接入认证。若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。
timeout	Integer	验证码有效时间，单位秒，设备需要在有效时间内接入物联网平台。

## 错误码

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
400	100003	Invalid verify code.	验证码无效。 处理建议：请检查接口请求中的verifyCode是否有误。若请求参数中未带verifyCode，请联系物联网平台维护人员处理。
400	100007	Bad request message.	参数不合法。 处理建议：deviceId未赋值，请参考请求参数说明填写请求。
400	100426	The nodeId is duplicated.	nodeId重复。 处理建议：请检查接口请求中nodeId是否有误。
400	100610	Device is not active.	设备未激活。 处理建议：请检查该设备是否已接入物联网平台并激活。
400	100611	Device is online.	设备在线。 处理建议：请下线该设备或者断开设备与物联网平台的连接。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
401	100025	AppId for auth not exist.	获取不到appId鉴权信息。 处理建议： <ul style="list-style-type: none"><li>● 请检查请求结构体的Header头域中是否给字段app_key赋值。</li><li>● 若通过HTTP调用，请联系物联网平台维护人员确认Header头域中放置appId的字段名是app_key还是x-app-key。</li></ul>
403	100203	The application is not existed.	授权应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.2.3 修改设备信息

#### 典型场景

第三方应用在物联网平台注册设备后，若设备的基本信息发生变更，也可调用此接口在物联网平台修改设备的信息。

#### 接口功能

支持第三方应用修改设备的基本信息，包括设备类型、设备模型、厂商信息、接入协议等。

#### 接口描述

```
void modifyDeviceInfo(ModifyDeviceInforInDTO mdiInDTO, String deviceId, String appId, String accessToken) throws NorthApiException
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。

参数	必选/可选	类型	位置	描述
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。
mdiInDto	必选	ModifyDeviceInforInDTO	body	具体参见 <a href="#">ModifyDeviceInforInDTO</a> 结构体的描述。

ModifyDeviceInforInDTO结构体：

参数	必选/可选	类型	位置	描述
customFields	可选	List< <a href="#">CustomField</a> >	Body	自定义字段列表，用户可设置自定义字段。
imsi	可选	String(1~64)	Body	NB-IoT终端的IMSI。
name	可选	String(1~256)	body	设备名称。
endUser	可选	String(1~256)	body	终端用户，若为直连设备，则endUser可选；若为非直连设备，则endUser可以为null。
mute	可选	Enum	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● TRUE：冻结状态</li><li>● FALSE：非冻结状态</li></ul>
manufacturerId	可选	String(1~256)	body	厂商ID，唯一标识一个厂商。 注册设备后必须修改厂商ID，且要与profile中定义的保持一致。
manufacturerName	可选	String(1~256)	body	厂商名称。
deviceType	可选	String(1~256)	body	设备类型，大驼峰命名方式，例如：MultiSensor、ContactSensor、CameraGateway。 在NB-IoT方案中，注册设备后必须修改deviceType，且要与profile中定义的保持一致。

参数	必选/可选	类型	位置	描述
model	可选	String(1~256)	body	设备型号，由厂商定义。 注册设备后必须修改设备型号，且要与profile中定义的保持一致。
location	可选	String(1~1024)	body	设备位置。
protocolType	可选	String(1~256)	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。 注册设备后必须修改协议类型，且要与profile中定义的保持一致。
deviceConfig	可选	DeviceConfigDTO	body	设备配置信息，具体参见 <b>DeviceConfigDTO</b> 结构体。
region	可选	String(1~256)	body	设备区域信息。
organization	可选	String(1~256)	body	设备所属的组织信息。
timezone	可选	String(1~256)	body	设备所在时区信息，使用时区编码，如上海时区对应的时区编码为Asia/Shanghai。
ip	可选	String(128)	Body	设备的IP地址。
isSecure	可选	Boolean	body	设备的安全状态，默认值为“false”。 ● “true”：安全 ● “false”：非安全
psk	可选	String(8~32)	body	psk参数，取值范围是“a-f、A-F、0-9”组成的字符串。
tags	可选	List< <b>Tag2</b> >	Body	设备的标签信息。

CustomField:

参数	必选/可选	类型	位置	描述
fieldName	可选	String(256)	Body	字段名字。
fieldType	可选	String(256)	Body	字段类型。
fieldValue	可选	String(256)	Body	字段值。

DeviceConfigDTO结构体说明：

参数	必选/可选	字段类型	描述
dataConfig	可选	DataConfigDTO	数据配置信息，具体参见 <a href="#">DataConfigDTO</a> 结构体。

DataConfigDTO结构体说明：

参数	必选/可选	字段类型	描述
dataAgingTime	可选	Integer	数据老化时长，取值范围：0-90，单位：天。

Tag2:

参数	必选/可选	类型	位置	描述
tagName	必选	String(1-128)	body	标签名称。
tagValue	必选	String(1-1024)	body	标签值。
tagType	可选	Integer	body	标签类型。

## 返回值

void

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	100440	The isSecure is invalid.	isSecure参数值有误。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
403	500004	The amount of frozen devices has reached the limit.	被冻结设备已达到上限。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	100441	The amount of nonSecure device has reached the limit.	非安全设备数量已达到上限。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.2.4 删除直连设备

#### 典型场景

已在物联网平台注册的直连设备，若不再需要接入平台时，第三方应用可调用此接口在物联网平台删除设备。后续若设备要再次接入平台，第三方应用需要在物联网平台重新注册设备。

#### 接口功能

支持第三方应用在物联网平台删除已注册的直连设备，使设备无法再接入平台。

#### 接口描述

```
void deleteDirectDevice(String deviceId, Boolean cascade, String appId, String accessToken) throws NorthApiException
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。

参数	必选/可选	类型	位置	描述
cascade	必选	Boolean	query	仅当设备下连接了非直连设备时生效，不设置时可填写null。 <ul style="list-style-type: none"><li>● true，级联删除，即删除直连设备和其下的非直连设备。</li><li>● false，删除直连设备，但是不删其下的非直连设备，并将非直连设备的属性变为直连设备属性。</li></ul>
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 返回值

void

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.2.5 查询设备激活状态

#### 典型场景

第三方应用通过“注册直连设备”接口在物联网平台注册设备后，在设备首次接入平台之前，设备的激活状态为false；当设备首次接入平台后，设备的激活状态为true（无论设备接入后是处于在线、离线或异常状态）。第三方应用可调用此接口查询设备在物联网平台的激活状态，确定设备是否已接入平台。

#### 接口功能

支持第三方应用通过设备ID查询设备在物联网平台的激活状态，确认设备是否已接入平台。

#### 接口描述

```
QueryDeviceStatusOutDTO queryDeviceStatus(String deviceId, String appId, String accessToken)  
throws NorthApiException
```

#### 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 返回值

QueryDeviceStatusOutDTO

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
activated	Boolean	激活状态，设备是否通过验证码获取密码的状态标识。 ● true: 已激活 ● false: 未激活
name	String(256)	设备名称。

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： ● 请检查HTTP请求头域中携带的appId是否有误。 ● 请检查请求路径(url)中传入的appId是否有误。
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.2.6 查询设备实时位置信息

#### 典型场景

第三方应用若需要获取设备当前最新的位置信息时，可调用此接口查询设备的实时位置信息，当前仅支持查询NB-IoT设备的实时位置信息。

#### 接口功能

物联网平台通过和位置服务器对接，向第三方应用提供查询NB-IoT设备实时位置信息的能力。

##### 注意

根据位置服务器的能力和配置情况，查询设备实时位置信息接口会存在较大的时延，而且并发能力有限。具体情况建议咨询物联网平台维护人员。

## 接口描述

```
QueryDeviceRealtimeLocationOutDTO queryDeviceRealtimeLocation(QueryDeviceRealtimeLocationInDTO  
qdrlInDTO, String appId, String accessToken) throws NorthApiException
```

## 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。
qdrlInDTO	必选	QueryDeviceRealtimeLocationInDTO	body	具体参见 <a href="#">QueryDeviceRealtimeLocationInDTO</a> 结构体的描述。

QueryDeviceRealtimeLocationInDTO结构体：

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
horAcc	必选	Integer	body	水平误差，单位：米，如果不携带，则默认1000米。
geoInfo	可选	CoordinateReferenceSystem	body	地理坐标信息要求，默认WGS84，其他坐标暂不支持。具体参见 <a href="#">CoordinateReferenceSystem</a> 结构体。

CoordinateReferenceSystem结构体说明：

参数	必选/可选	类型	位置	描述
identifier	必选	Identifier	body	坐标参考系统标识。具体参见 <a href="#">Identifier</a> 结构体。

Identifier结构体说明：

参数	必选/可选	类型	位置	描述
code	必选	String	body	坐标参考系统编号， 默认4326， 当前只支持这个WGS84数字坐标。
codeSpace	必选	String	body	编码空间， 当前只支持EPSG。
edition	必选	String	body	编辑版本， 当前只支持6.1。

## 返回值

QueryDeviceRealtimeLocationOutDTO

参数	类型	描述
deviceId	String	设备ID， 用于唯一标识一个设备。
pd	Pd	定位结果， 定位成功时携带， 具体参见 <a href="#">Pd结构体</a> 。
poserr	Poserr	定位错误信息， 定位失败时携， 具体参见 <a href="#">Poserr结构体</a> 。

Pd结构体说明：

参数	类型	描述
time	String	定位时间， 格式“yyyyMMddhhmmss”。
utcOff	String	时间时区， 格式“hhmm”。
srsName	String	坐标参考， 固定取值：“www.epsg.org#4326”。
X	String	X轴， 纬度， 格式：“DD MM SS.hhhN”或“DD MM SS.hhhS”， 最后的N指北纬， S指南纬。
Y	String	Y轴， 经度， 格式：“DD MM SS.hhhE”或“DD MM SS.hhhW”， 最后的E指东经， W指西经。
radius	String	误差半径， 单位：米。

Poserr结构体说明：

参数	类型	描述
time	String	定位时间， 格式“yyyyMMddhhmmss”。

参数	类型	描述
resid	String	定位结果值。
add_info	String	定位详细结果。

## 错误码

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	103801	Exceed the system concurrency restriction.	超过系统并发限制。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.2.7 查询设备影子

#### 典型场景

若设备处于离线或异常状态时，第三方应用无法通过命令下发的方式向设备下发配置。此时可把要下发的配置设置到设备影子，当设备重新上线时，设备影子会把配置下发给设备。第三方应用可调用此接口查看设备影子上的设备配置信息及最新的设备上报数据信息。

#### 接口功能

支持第三方应用查询单个设备的设备影子信息，包括对设备的配置信息（desired区）和设备最新上报的数据信息（reported区）。

#### 接口描述

```
QueryDeviceShadowOutDTO queryDeviceShadow(String deviceId, String appId, String accessToken)  
throws NorthApiException
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 返回值

QueryDeviceShadowOutDTO

参数	类型	描述
deviceId	String(36)	设备ID，用于唯一标识一个设备。
gatewayId	String(36)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	Enum	设备的类型。
createTime	String(256)	设备创建的时间。
lastModifiedTime	String(256)	设备最后一次修改信息的时间。
deviceInfo	DeviceInfo	设备详细信息，具体参见 <a href="#">DeviceInfo结构体</a> 。
services	List<DeviceServiceB>	设备的服务属性信息，具体参见 <a href="#">DeviceServiceB</a> 。

DeviceInfo结构体说明：

参数	类型	描述
nodeId	String(256)	设备的唯一标识。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacturerId	String(256)	厂商ID，唯一标识一个厂商，与设备Profile文件中的信息保持一致。
manufacturerName	String(256)	厂商名称，与设备Profile文件中的信息保持一致。
mac	String(256)	设备的MAC地址。

参数	类型	描述
location	String(2048)	设备的位置信息。
deviceType	String(256)	设备类型，大驼峰命名方式，与设备Profile文件中的信息保持一致，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号，与设备Profile文件中的信息保持一致。 Z-Wave: ProductType + ProductId, 16 进制格式XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
swVersion	String(256)	设备的软件版本。 Z-Wave: 主版本号.次版本号，如：1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
protocolType	String(256)	设备使用的协议类型，与设备Profile文件中的信息保持一致，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	String(256)	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
status	String	设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。
statusDetail	String(256)	设备的状态详情，跟status取值对应，取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	String	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>
supportedSecurity	String	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● TRUE: 支持安全模式</li><li>● FALSE: 不支持安全模式</li></ul>
isSecurity	String	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● TRUE: 启用</li><li>● FALSE: 未启用</li></ul>
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。

参数	类型	描述
batteryLevel	String(256)	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceServiceB说明:

参数	类型	描述
serviceId	String(256)	服务ID，用于标识一个服务。
reportedProps	ObjectNode	设备最新上报的数据。
desiredProps	ObjectNode	要向设备下发的配置信息。
eventTime	String(256)	事件发生的时间。
serviceType	String(256)	服务类型。

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	pp_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.2.8 修改设备影子

#### 典型场景

物联网平台支持创建设备的“影子”。设备影子用于存储设备最新上报的服务属性数据，以及第三方应用对服务属性的配置（服务属性为设备Profile文件中定义的service属性）。若设备处于离线或异常状态时，第三方应用无法通过命令下发的方式向设备下发配置，此时可把要下发的配置设置到设备影子，当设备重新上线时，设备影子会把配置下发给设备。第三方应用可调用此接口在设备影子上配置要下发给设备的配置信息。

每个设备有且只有一个设备影子，设备影子上有desired和report区。

- desired区用于存储对设备服务属性的配置。若设备当前在线，desired区的配置会即时下发给设备；若设备当前不在线，desired区的配置会在设备上线时再下发给设备。

- report区用于存储设备最新上报的服务属性数据。当设备上报数据时，平台会把数据同步到设备影子的report区。

## 接口功能

支持第三方应用配置设备影子desired区的信息，用于设备上线时把配置下发给设备。

## 接口描述

```
void modifyDeviceShadow(ModifyDeviceShadowInDTO mdsInDTO, String deviceId, String appId, String accessToken) throws NorthApiException
```

## 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String(256)	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。
mdsInDTO	必选	ModifyDeviceShadowInDTO	body	具体参见 <a href="#">ModifyDeviceShadowInDTO</a> 结构体的描述。

ModifyDeviceShadowInDTO结构体：

参数	必选/可选	类型	位置	描述
serviceDesireds	必选	List<ServiceDesiredDTO>	body	需要修改的设备配置或状态信息。具体参见 <a href="#">ServiceDesiredDTO</a> 结构体。

ServiceDesiredDTO结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	可选	String(1-256 )	body	服务ID，用于标识一个服务。
desired	可选	ObjectNode	body	设备的服务属性配置。

## 返回值

void

## 错误码

Http状态码	错误码	错误描述	说明
200	100425	The special deviceCapability is not exist.	设备模板不存在。 处理建议：请检查设备模板是否已在物联网平台上传。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	107002	The properties is empty in database.	设备属性不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107003	The request properties is unknown.	设备状态为未知。 处理建议：请检查设备与平台的连接是否正常。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	10044 3	The property is forbidden to write.	设备属性不可写。
403	10100 04	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	10100 05	pp_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken是否有误。
404	10040 3	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	10041 8	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	10002 3	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	10020 3	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.3 批量处理

第三方应用通过批量处理接口对已接入物联网平台的设备进行批量操作。

### 2.3.3.1 创建批量任务

#### 典型场景

第三方应用若需要对多个设备进行批量操作时，可调用此接口创建设备的批量操作任务，当前仅支持给批量设备下发缓存命令。

#### 接口功能

支持第三方应用在物联网平台创建设备的批量操作任务。

#### 接口描述

```
BatchTaskCreateOutDTO createBatchTask(BatchTaskCreateInDTO2 btcInDTO, String accessToken) throws NorthApiException
```

#### 接口所属类

BatchProcess

#### 参数说明

参数	必选/可选	类型	位置	描述
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。
btcInDTO	必选	BatchTaskCreateInDTO2	body	详细参见 <a href="#">BatchTaskCreateInDTO2</a> 结构体的描述。

BatchTaskCreateInDTO2结构体：

参数	必选/可选	类型	位置	描述
appId	必选	String(6~4)	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。 <code>appid</code> 在物联网平台的SP Portal上创建应用时获得。
timeout	必选	Integer	body	任务超时时长，单位分钟，范围为10到2880。
taskName	必选	String	body	任务名称，最大长度256字符。
taskType	必选	String	body	任务类型，DeviceCmd。

参数	必选/ 可选	类型	位置	描述
param	必选	ObjectN ode	body	任务详细参数，根据taskType任务类型的不同对应不同类型参数，具体参见 <a href="#">ObjectNode结构体</a> 。
tags	可选	List<Ta gDTO2 >	body	标签列表，请参见 <a href="#">TagDTO2结构体</a> 。

ObjectNode:

参数	必选/ 可选	类型	位置	描述
type	必选	String	body	批量命令类型，取值范围： DeviceList/DeviceType/DeviceArea/GroupList/Broadcast/GroupIdList。
device List	条件必 选	List<Str ing>	body	设备ID列表， type为DeviceList时需要填写。
device Type	条件必 选	String	body	设备类型， type为DeviceType时需要填写，其值应当与profile中定义的一致。
manufa cturerI d	条件可 选	String	body	厂商ID， type为DeviceType时可填写，其值应当与profile中定义的一致。
model	条件可 选	String	body	设备型号， type为DeviceType时可填写，其值应当与profile中定义的一致。
device Locatio n	条件必 选	String	body	设备位置， type为DeviceArea时需要填写。
groupL ist	条件必 选	List<Str ing>	body	群组Id列表或设备组名称列表， type为 GroupIdList时需要填写群组Id， type为 GroupList时需要填写设备组名称。
comma nd	必选	<a href="#">Comma ndDTO</a>	body	命令信息。
callbac kUrl	可选	String	body	命令执行结果的推送地址。
maxRe transmi t	可选	Integer(0~3)	body	命令下发最大重传次数，取值范围： 0-3。
groupT ag	可选	String	body	群组标签。

CommandDTO:

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-64)	body	命令对应的服务ID，要与profile中定义的serviceId保持一致。
method	必选	String(1-28)	body	服务下具体的命令名称，要与profile中定义的命令名保持一致。
paras	可选	ObjectNode	body	命令参数， jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。

TagDTO2结构体说明:

参数	必选/可选	类型	位置	描述
tagName	必选	String(1-128)	body	标签名称。
tagValue	必选	String(1-1024)	body	标签值。

## 返回值

BatchTaskCreateOutDTO

参数	类型	描述
taskID	String	批量任务的ID。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	<p>应用不存在。 处理建议：</p> <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径（url）中传入的appId是否有误。</li> </ul>

Http状态码	错误码	错误描述	说明
200	101001	Resource doesn't exist.	资源不存在。
200	105001	The batchTask count has reached the limit.	未完成的任务数大于等于10个时，返回任务达到数量限制。
200	105002	The batchTask name has exist.	任务名字已存在。 处理建议：修改任务名称。
400	105201	The tagName and tagValue has been used on the platform.	平台已使用tagName和tagValue。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100028	The user has no right.	用户没有操作权限。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105202	The tag is not existed.	标签不存在。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.3.2 查询指定批量任务信息

#### 典型场景

第三方应用通过“创建批量任务”接口创建了设备的批量操作任务后，可调用此接口查询该批量任务的信息，包括任务的当前状态、子任务完成情况等。

#### 接口功能

支持第三方应用通过任务ID查询单个批量任务的信息。

#### 接口描述

```
QueryOneTaskOutDTO queryOneTask(String taskId, String select, String appId, String accessToken)  
throws NorthApiException
```

#### 接口所属类

BatchProcess

#### 参数说明

参数	必选/可选	类型	位置	描述
taskId	必选	String	path	批量任务ID，创建批量任务后获得。
select	必选	String	query	指定可选的返回值，可取值：tag。不指定时可填写null。
appId	必选	String	query	如果是本应用的任务，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 返回值

QueryOneTaskOutDTO

参数	类型	描述
appId	String	批量任务所归属的appId。
taskId	String	批量任务ID。
taskName	String	批量任务名称。
operator	String	下发该批量任务的操作员。
taskFrom	String	批量任务的来源。 ● Portal: 通过SP Portal创建的。 ● Northbound: 调用北向API接口创建的。
taskType	String	批量任务的类型, 取值范围: DeviceCmd。
status	String	批量任务的状态, 取值范围: Pending/Running/Complete/Timeout。
startTime	String	批量任务的创建时间。
timeout	Integer	批量任务的超时时间, 单位秒。
progress	Integer	批量任务的进度, 单位: 千分比, 范围: 0-1000, 向下取整。
totalCnt	Integer	任务的总数。
successCnt	Integer	成功的任务数。
failCnt	Integer	失败的任务数。
timeoutCnt	Integer	超时的任务数。
expiredCnt	Integer	未执行的失效任务数。
completeCnt	Integer	完成的任务数, 完成的任务数=成功的任务数+失败的任务数+超时的任务数。
successRate	Integer	任务成功率, 单位: 千分比, 范围: 0-1000, 向下取整。
param	ObjectNode	不同任务类型的具体参数。
tags	List<TagDT O2>	批量任务的标签列表。

ObjectNode:

参数	类型	描述
type	String	批量命令类型, 取值范围: DeviceList/DeviceType/DeviceArea/GroupList/Broadcast/GroupIdList。
deviceList	List<String>	设备ID列表, type为DeviceList时返回值。

参数	类型	描述
deviceType	String	设备类型, type为DeviceType时返回值, 其值应当与profile中定义的一致。
manufacturer Id	String	厂商ID, type为DeviceType时返回值, 其值应当与profile中定义的一致。
model	String	设备型号, type为DeviceType时返回值, 其值应当与profile中定义的一致。
deviceLocation	String	设备位置, type为DeviceArea时返回值。
groupList	List<String>	群组名称列表, type为GroupList时返回值。
command	CommandDTO	命令信息。
callbackUrl	String	命令执行结果的推送地址。
maxRetransmit	Integer(0~3)	命令下发最大重传次数, 取值范围: 0-3。
groupTag	String	群组标签。

CommandDTO:

参数	类型	描述
serviceId	String(1-64)	命令对应的服务ID, 要与profile中定义的serviceId保持一致。
method	String(1-128)	服务下具体的命令名称, 要与profile中定义的命令名保持一致。
paras	ObjectNode	命令参数, jsonString格式, 里面是一个个键值对, 每个键都是profile中命令的参数名 (paraName), 具体格式需要应用和设备约定。

TagDTO2:

参数	类型	描述
tagName	String(1-128)	标签名称。
tagValue	String(1-1024)	标签值。

## 错误码

Http状态码	错误码	错误描述	说明
400	100019	Illegal request.	非法请求。 处理建议：请检查接口请求中的必须参数是否为空。
400	100022	The input is invalid	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105005	The batchTask is not existed.	批量任务不存在。 处理建议：请检查接口请求中的taskId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

### 2.3.3.3 查询批量任务的子任务信息

#### 典型场景

第三方应用通过“创建批量任务”接口创建了设备的批量操作任务后，可调用此接口查询该批量任务中子任务的详情信息，如任务的执行状态、任务内容等。

#### 接口功能

支持第三方应用根据条件查询批量任务中子任务的详情信息，支持查询本应用创建的批量任务信息。

#### 接口描述

```
QueryTaskDetailsOutDTO queryTaskDetails(QueryTaskDetailsInDTO qtdInDTO, String accessToken) throws NorthApiException
```

#### 接口所属类

BatchProcess

#### 参数说明

参数	必选/可选	类型	位置	描述
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。
qtdInDTO	必选	QueryTaskDetailsInDTO	query	详细参见 <a href="#">QueryTaskDetailsInDTO</a> 结构体的描述。

QueryTaskDetailsInDTO结构体：

参数	必选/可选	类型	位置	描述
appId	可选	String	query	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。appid在物联网平台的SP Portal上创建应用时获得。此处填写授权应用的appid。
taskId	必选	String	query	批量任务的ID。
status	可选	String	query	任务的详情状态，Pending/WaitResult/Success/Fail/Timeout。

参数	必选/可选	类型	位置	描述
device Id	可选	String	query	设备Id, 查询批量命令任务时使用。
comm andId	可选	String	query	命令Id, 查询批量命令任务时使用。
pageNo	可选	Integer	query	分页查询参数。 ● 值为空时, 查询内容不分页。 ● 值为大于等于0的整数时, 分页查询。 ● 值为0时查询第一页。
pageSize	可选	Integer	query	分页查询参数, 取值大于等于1的整数, 缺省值: 1。

## 返回值

QueryTaskDetailsOutDTO

参数	类型	描述
pageNo	Integer	分页查询参数。 ● 值为空时, 查询内容不分页。 ● 值为大于等于0的整数时, 分页查询。 ● 值为0时查询第一页。
pageSize	Integer	分页查询参数, 取值大于等于1的整数, 缺省值: 1。
totalCount	Integer	查询到的任务总数。
taskDetails	List<QueryTaskDetailDTOCloud2NA>	任务详情信息列表, 具体参见 <a href="#">QueryTaskDetailDTOCloud2NA结构体</a> 。

QueryTaskDetailDTOCloud2NA结构体说明:

参数	类型	描述
status	String	任务执行状态, 取值范围: Pending/WaitResult/Success/Fail/Timeout。
output	String	批量下发命令的输出信息。
error	String	任务的错误原因, 格式为: {"error_code": "****", "error_desc": "*****"}。
param	<a href="#">ObjectNode</a>	不同任务类型的具体参数。

ObjectNode:

参数	类型	描述
deviceId	String	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
commandId	String	下发的命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。

## 错误码

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105005	The batchTask is not existed.	批量任务不存在。 处理建议：请检查接口请求中的taskId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 2.3.4 订阅管理

物联网平台支持第三方应用对设备数据的订阅，若订阅的设备信息有变更，平台会推送给第三方应用。消息订阅接口需要与[消息推送](#)接口配合使用。

### 2.3.4.1 订阅平台业务数据

#### 典型场景

第三方应用可在物联网平台订阅设备相关的业务数据，当物联网平台中设备的业务信息发生变化时（如设备注册、设备数据上报、设备状态变更等），平台会向第三方应用发送通知消息，通知其具体的变化信息。第三方应用可以调用此接口向物联网平台订阅不同类型的设备业务变更通知。

#### 接口功能

支持第三方应用向物联网平台订阅设备的业务变更信息，当设备的状态、数据等信息发生变化时，平台会向第三方应用发送通知消息。

#### 接口描述

```
SubscriptionDTO subDeviceData(SubDeviceDataInDTO sddInDTO, String ownerFlag, String accessToken)  
throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
sddInDTO	必选	SubDeviceDataInDTO	body	见下表SubDeviceDataInDTO的描述。
ownerFlag	必选	String(256)	query	callbackUrl的所有者标识，不指定该标识可填写null。 <ul style="list-style-type: none"><li>● ownerflag为false时，表示callbackUrl的owner是授权应用。</li><li>● ownerflag为true时，表示callbackUrl的owner为被授权应用。</li></ul>

参数	必选/可选	类型	位置	描述
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## SubDeviceDataInDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	<p>通知类型，第三方应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备，订阅后推送绑定设备通知）</li><li>● deviceAdded（添加新设备，订阅后推送注册设备通知）</li><li>● deviceInfoChanged（设备信息变化，订阅后推送设备信息变化通知）</li><li>● deviceDataChanged（设备数据变化，订阅后推送设备数据变化通知）</li><li>● deviceDatasChanged（设备数据批量变化，订阅后推送批量设备数据变化通知）</li><li>● deviceDeleted（删除设备，订阅后推送删除设备通知）</li><li>● messageConfirm（消息确认，订阅后推送设备消息确认通知）</li><li>● commandRsp（命令响应，订阅后推送设备命令响应通知）</li><li>● deviceEvent（设备事件，订阅后推送设备事件通知）</li><li>● serviceInfoChanged（服务信息变化，订阅后推送设备服务信息变化通知）</li><li>● deviceModelAdded（添加设备模型，订阅后推送增加设备模型通知）</li><li>● deviceModelDeleted（删除设备模型，订阅后推送删除设备模型通知）</li><li>● deviceDesiredPropertiesModifyStatusChanged（设备影子状态变更，订阅后推送设备影子状态变更通知）</li></ul>

参数	必选/ 可选	类型	位置	描述
callbackUrl	必选	String(1024)	body	<p>订阅的回调地址，用于接收对应类型的通知消息。</p> <p>必须使用HTTPS信道回调地址，同时回调地址中必须指定回调地址的端口。例如：https://XXX.XXX.XXX.XXX:443/callbackurltest</p> <p><b>说明</b> HTTP信道只可用于调测。</p>
appId	可选	String(256)	body	设备或者规则等订阅内容主体所属的appId。
channel	可选	String(32)	Body	传输通道，若是MQTT客户端订阅，则取值为MQTT，其他情况为HTTP。

## 响应参数

SubscriptionDTO

参数	类型	描述
subscriptionId	String	订阅ID号，用于标识一个订阅。
notifyType	String	通知的类型。
callbackUrl	String	订阅的回调地址。
clientIds	List<String>	MQTT客户端ID，只有MQTT订阅时该字段有返回值。

## 错误码

Http状态码	错误码	错误描述	说明
400	100222	The request callbackurl is illegal.	<p>回调地址非法。</p> <p>处理建议：请检查请求体中的回调地址是否有误。</p>
403	100217	The application hasn't been authorized.	<p>应用未被授权。</p> <p>处理建议：若非应用授权场景，请确认请求参数中的appId为空。</p>

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
409	100227	The resource is conflicted.	资源冲突，通知类型已经被订阅。 处理建议：请检查要订阅的通知类型是否已被订阅。

### 2.3.4.2 订阅平台管理数据

#### 典型场景

第三方应用可向物联网平台订阅设备相关的管理数据，当物联网平台中设备进行版本升级等相关操作时，平台会向第三方应用发送通知消息，知会其具体的操作状态或操作结果信息。第三方应用可以调用此接口向物联网平台订阅不同类型的设备升级操作通知。

#### 接口功能

支持第三方应用向物联网平台订阅设备的升级操作信息，当设备进行版本升级等相关操作时，平台会向第三方应用发送通知消息。

#### 接口描述

```
void subDeviceData(SubDeviceManagementDataInDTO smdInDTO, String accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
smdInDTO	必选	SubDeviceManagementDataInDTO	body	见下表 SubDeviceManagementDataInDTO 的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

## SubDeviceManagementDataInDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	<p>通知类型。</p> <ul style="list-style-type: none"> <li>● swUpgradeStateChangeNotify（软件升级状态变更通知，订阅后推送软件升级状态变更通知）</li> <li>● swUpgradeResultNotify（软件升级结果通知，订阅后推送软件升级结果通知）</li> <li>● fwUpgradeStateChangeNotify（固件升级状态变更通知，订阅后推送固件升级状态变更通知）</li> <li>● fwUpgradeResultNotify（固件升级结果通知，订阅后推送固件升级结果通知）</li> </ul>
callbackurl	必选	String	body	<p>订阅的回调地址，用于接收对应类型的的通知消息。</p> <p>必须使用HTTPS信道回调地址，同时回调地址中必须指定回调地址的端口。例如： <a href="https://XXX.XXX.XXX.XXX:443/callbackurltest">https://XXX.XXX.XXX.XXX:443/callbackurltest</a></p> <p><b>说明</b> HTTP信道只可用于调测。</p>

## 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
400	100222	Internal server error.	<p>回调地址非法。</p> <p>处理建议：请检查请求体中的回调地址是否有误。</p>
400	100228	The application input is invalid.	<p>应用输入无效。</p> <p>处理建议：请参考请求参数说明检查接口请求中的参数是否有误。</p>
403	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的 appId是否有误。</li> <li>● 请检查请求路径（url）中传入的 appId是否有误。</li> </ul>

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100229	Get AppKey from header failed.	从消息头获取AppKey失败。
500	100244	register out route fail.	注册路由失败。 处理建议：联系物联网平台维护人员。

### 2.3.4.3 查询单个订阅

#### 典型场景

第三方应用可在物联网平台上创建多个订阅，以订阅不同类型的设备变更通知。第三方应用若需要查看某个订阅的配置信息，可调用此接口进行查询。

#### 接口功能

支持第三方应用在物联网平台根据订阅ID查询指定订阅的配置信息。

#### 接口描述

```
SubscriptionDTO querySingleSubscription(String subscriptionId, String appId, String accessToken)  
throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
subscriptionId	必选	String	path	订阅ID号，通过调用订阅接口返回或者查询订阅接口获得。
appId	必选	String	query	设备或者规则等订阅内容主体所属的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

SubscriptionDTO

参数	类型	描述
subscriptionId	String	订阅ID号，用于标识一个订阅。
notifyType	String	通知的类型。
callbackUrl	String	订阅的回调地址。

## 错误码

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。

### 2.3.4.4 批量查询订阅

#### 典型场景

第三方应用可在物联网平台上创建多个订阅，以订阅不同类型的设备变更通知。第三方应用若需要查看本应用下所有订阅配置，或者查看某种订阅类型下的所有订阅配置，可调用此接口进行查询。

#### 接口功能

支持第三方应用在物联网平台查询本应用下的所有订阅信息，或者某种订阅类型下的所有订阅信息。

#### 接口描述

```
QueryBatchSubOutDTO queryBatchSubscriptions(QueryBatchSubInDTO qbsInDTO, String accessToken)  
throws NorthApiException
```

## 参数说明

参数	必选/可选	类型	位置	描述
qbsInDTO	必选	QueryBatchSubInDTO	query	见下表QueryBatchSubInDTO的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

### QueryBatchSubInDTO

参数	必选/可选	类型	位置	描述
appId	可选	String(256)	query	设备或者规则等订阅内容主体所属的appId。

参数	必选/可选	类型	位置	描述
notifyType	可选	String(256)	query	<p>通知类型，第三方应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备）</li><li>● deviceAdded（添加新设备）</li><li>● deviceInfoChanged（设备信息变化）</li><li>● deviceDataChanged（设备数据变化）</li><li>● deviceDatasChanged（设备数据批量变化）</li><li>● deviceDeleted（删除设备）</li><li>● messageConfirm（消息确认）</li><li>● commandRsp（命令响应）</li><li>● deviceEvent（设备事件）</li><li>● serviceInfoChanged（服务信息变化）</li><li>● deviceModelAdded（添加设备模型）</li><li>● deviceModelDeleted（删除设备模型）</li><li>● deviceDesiredPropertiesModifyStatusChanged（修改设备影子状态变更）</li><li>● swUpgradeStateChangeNotify（软件升级状态变更通知）</li><li>● swUpgradeResultNotify（软件升级结果通知）</li><li>● fwUpgradeStateChangeNotify（固件升级状态变更通知）</li><li>● fwUpgradeResultNotify（固件升级结果通知）</li></ul>
pageNo	可选	Integer	query	<p>分页查询参数。</p> <ul style="list-style-type: none"><li>● 值为空时，查询内容不分页</li><li>● 值大于等于0时分页查询</li><li>● 值等于0时查询第一页</li></ul>
pageSize	可选	Integer	query	分页查询参数，取值大于等于1的整数，缺省值：10。

## 响应参数

QueryBatchSubOutDTO

参数	类型	描述
totalCount	long	查询到的订阅记录总数。
pageNo	long	查询的页码。
pageSize	long	查询每页信息的数量。
subscriptions	List<Subscription DTO>	订阅信息列表，具体参见 <a href="#">SubscriptionDTO结构体</a> 。

SubscriptionDTO结构体说明：

参数	类型	描述
subscriptionId	String	订阅ID号。
notifyType	String	通知类型。
callbackUrl	String	订阅的回调地址。

## 错误码

Http状态码	错误码	错误描述	说明
400	100224	The resource exceeds 1000, please refinement query conditions.	查到端资源超出1000条，请重新设定查询条件。 处理建议：请缩小查询条件范围。
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 2.3.4.5 删除单个订阅

#### 典型场景

第三方应用若不再需要接收物联网平台推送的某个订阅通知消息时，可调用此接口删除指定的订阅配置，取消订阅。

#### 接口功能

支持第三方应用在物联网平台根据订阅ID删除指定的订阅配置。

#### 接口描述

```
void deleteSingleSubscription(String subscriptionId, String appId, String accessToken) throws  
NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
subscriptionId	必选	String(2-56)	path	订阅ID号。
appId	可选	String(2-56)	query	设备或者规则等订阅内容主体所属的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

void

#### 错误码

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100225	The resource is not found	找不到资源。 处理建议：请检查订阅ID是否有误。

### 2.3.4.6 批量删除订阅

#### 典型场景

第三方应用若不再需要接收物联网平台推送的订阅通知消息，或者不再需要接收某类订阅通知消息，可调用此接口批量删除订阅配置，取消订阅。

#### 接口功能

支持第三方应用在物联网平台批量删除订阅配置，可删除全部订阅，或删除指定订阅类型的所有订阅，或删除指定回调URL地址的所有订阅。

#### 接口描述

```
void deleteBatchSubscriptions(DeleteBatchSubInDTO dbsInDTO, String accessToken) throws  
NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
dbsInDTO	必选	DeleteBatchSubInDTO	body	见下表DeleteBatchSubInDTO的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### DeleteBatchSubInDTO

参数	必选/可选	类型	位置	描述
appId	可选	String(256 )	query	设备或者规则等订阅内容主体所属的appId。

参数	必选/可选	类型	位置	描述
notifyType	可选	String(256 )	query	<p>通知类型，第三方应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice (绑定设备)</li><li>● deviceAdded (添加新设备)</li><li>● deviceInfoChanged (设备信息变化)</li><li>● deviceDataChanged (设备数据变化)</li><li>● deviceDatasChanged (设备数据批量变化)</li><li>● deviceDeleted (删除设备)</li><li>● messageConfirm (消息确认)</li><li>● commandRsp (命令响应)</li><li>● deviceEvent (设备事件)</li><li>● serviceInfoChanged (服务信息变化)</li><li>● deviceModelAdded (添加设备模型)</li><li>● deviceModelDeleted (删除设备模型)</li><li>● deviceDesiredPropertiesModifyStatus Changed (修改设备影子状态变更)</li><li>● swUpgradeStateChangeNotify (软件升级状态变更通知)</li><li>● swUpgradeResultNotify (软件升级结果通知)</li><li>● fwUpgradeStateChangeNotify (固件升级状态变更通知)</li><li>● fwUpgradeResultNotify (固件升级结果通知)</li></ul>
callbackU rl	可选	String(256 )	query	订阅回调的URL地址。

## 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100225	The resource is not found	找不到资源。 处理建议：请检查订阅类型是否有误。

## 2.3.5 消息推送

第三方应用向物联网平台订阅设备信息，当设备信息发生变更时，物联网平台向第三方应用推送此消息，第三方应用根据通知类型对消息分派处理。需要配合[订阅管理](#)接口使用。

### 2.3.5.1 注册设备通知

#### 典型场景

第三方应用在物联网平台订阅了注册设备通知后（订阅的通知类型为deviceAdded），当第三方应用通过“注册直连设备”接口在平台注册设备时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了注册设备通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT 平台北向 API 参考](#)中的[消息推送](#)章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleDeviceAdded(NotifyDeviceAddedDTO body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceAddedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceAdded。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	可选	String	body	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	body	设备类型。 <ul style="list-style-type: none"><li>● ENDPOINT</li><li>● GATEWAY</li><li>● UNKNOWN</li></ul>
deviceInfo	必选	DeviceInfo	body	设备信息，具体参见 <a href="#">DeviceInfo结构体</a> 。

DeviceInfo结构体说明：

参数	必选/可选	类型	位置	描述
nodeId	必选	String(256)	body	设备的唯一标识，通常使用MAC地址，Serial No或IMEI作为nodeId。 <b>说明</b> 若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。 <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
name	可选	String(256)	body	设备名称。

参数	必选/ 可选	类型	位置	描述
description	可选	String(2048) )	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048) )	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型，大驼峰命名方式，如 MultiSensor、ContactSensor、CameraGateway。
model	可选	String(256)	body	设备的型号。 Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
swVersion	可选	String(256)	body	设备的软件版本。 Z-Wave: 主版本号.次版本号，如：1.1。
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	可选	String(256)	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
status	可选	String	body	设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。
statusDetail	可选	String(256)	body	设备的状态详情，根据status取值对应，取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	可选	String	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>

参数	必选/ 可选	类型	位置	描述
supporte dSecurity	可选	String	body	表示设备是否支持安全模式。 ● TRUE: 支持安全模式 ● FALSE: 不支持安全模式
isSecurit y	可选	String	body	表示设备当前是否启用安全模式。 ● TRUE: 启用 ● FALSE: 未启用
signalStr ength	可选	String(256)	body	设备的信号强度。
sigVersio n	可选	String(256)	body	设备的sig版本。
serialNu mber	可选	String(256)	body	设备的序列号。
batteryLe vel	可选	String(256)	body	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceAdded",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "nodeType": "GATEWAY",  
    "deviceInfo": {  
        "nodeId": "*****",  
        "name": null,  
        "description": null,  
        "manufacturerId": null,  
        "manufacturerName": null,  
        "mac": null,  
        "location": null,  
        "deviceType": null,  
        "model": null,  
        "swVersion": null,  
        "fwVersion": null,  
        "hwVersion": null,  
        "protocolType": null,  
        "bridgeId": null,  
        "status": "OFFLINE",  
        "statusDetail": "NOT_ACTIVE",  
        "mute": null,  
        "supportedSecurity": null,  
        "isSecurity": null,  
        "signalStrength": null,  
        "sigVersion": null,  
        "serialNumber": null,  
        "batteryLevel": null  
    }  
}
```

## 响应示例

response:  
Status Code: 200 OK

### 2.3.5.2 绑定设备通知

#### 典型场景

第三方应用在物联网平台订阅了绑定设备通知后（订阅的通知类型为bindDevice），当直连设备接入并绑定物联网平台时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了绑定设备通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleBindDevice(NotifyBindDeviceDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyBindDeviceDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：bindDevice。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
resultCode	必选	String	body	绑定结果，取值：expired/succeeded。
deviceInfo	可选	DeviceInfo	body	设备信息，具体参见 <a href="#">DeviceInfo结构体</a> 。

DeviceInfo结构体说明：

参数	必选/ 可选	类型	位置	描述
nodeId	必选	String(256)	body	标识设备的唯一ID。
name	可选	String(256)	body	设备名称。
description	可选	String(2048 )	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048 )	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型，大驼峰命名方式，如 MultiSensor、ContactSensor、 CameraGateway。
model	可选	String(256)	body	设备的型号。 Z-Wave：ProductType + ProductId，16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
swVersion	可选	String(256)	body	设备的软件版本。 Z-Wave：主版本号.次版本号，如：1.1。
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型： Z-Wave， ZigBee， WPS。
bridgeId	可选	String(256)	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
status	可选	String	body	设备的状态，表示设备是否在线，取值范围： ONLINE、OFFLINE、 ABNORMAL。
statusDetail	可选	String(256)	body	设备的状态详情，跟status取值对应，取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。

参数	必选/ 可选	类型	位置	描述
mute	可选	String	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 ● TRUE: 冻结状态 ● FALSE: 非冻结状态
supporte dSecurity	可选	String	body	表示设备是否支持安全模式。 ● TRUE: 支持安全模式 ● FALSE: 不支持安全模式
isSecurit y	可选	String	body	表示设备当前是否启用安全模式。 ● TRUE: 启用 ● FALSE: 未启用
signalStr ength	可选	String(256)	body	设备的信号强度。
sigVersio n	可选	String(256)	body	设备的sig版本。
serialNu mber	可选	String(256)	body	设备的序列号。
batteryLe vel	可选	String(256)	body	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

```
request: {callbackUrl}  
  
Header:  
Content-Type:application/json  
  
Body:  
{  
    "notifyType": "bindDevice",  
    "deviceId": "*****",  
    "resultCode": "succeeded",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturer": "wulian",  
        "deviceType": 90,  
        "model": "90",  
        "mac": "*****",  
        "swVersion": "...",  
        "fwVersion": "...",  
        "hwVersion": "...",  
        "protocolType": "zigbee",  
        "description": "smockdetector",  
        "nodeType": "GATEWAY"  
    }  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 2.3.5.3 设备信息变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备信息变化通知后（订阅的通知类型为 deviceInfoChanged），当设备的配置或状态信息（如设备厂商信息、位置信息、版本信息、在线状态信息等）发生变化时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备信息变化通知的第三方应用推送通知消息。

## 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleDeviceInfoChanged(NotifyDeviceInfoChangedDTO body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceInfoChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceInfoChanged。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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。
deviceInfo	必选	DeviceInfo	body	设备信息，具体参见 <a href="#">DeviceInfo结构体</a> 。

DeviceInfo结构体说明：

参数	必选/ 可选	类型	位置	描述
nodeId	必选	String(256)	body	<p>设备的唯一标识，通常使用MAC地址，Serial No或IMEI作为nodeId。</p> <p><b>说明</b></p> <p>若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
name	可选	String(256)	body	设备名称。
description	可选	String(2048 )	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048 )	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	可选	String(256)	body	<p>设备的型号。</p> <p>Z-Wave：ProductType + ProductId，16进制格式XXXX-XXXX补0对齐，如：001A-0A12，其他协议的格式待定。</p>
swVersion	可选	String(256)	body	<p>设备的软件版本。</p> <p>Z-Wave：主版本号.次版本号，如：1.1。</p>
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型，当前支持的协议类型：CoAP，huaweiM2M，Z-Wave，ONVIF，WPS，Hue，WiFi，J808，Gateway，ZigBee，LWM2M。
bridgeId	可选	String(256)	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。

参数	必选/ 可选	类型	位置	描述
status	可选	String	body	设备的状态，表示设备是否在线，取值范围：ONLINE、OFFLINE、ABNORMAL。
statusDetail	可选	String(256)	body	设备的状态详情，跟status取值对应，取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	可选	String	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>
supporte dSecurity	可选	String	body	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● TRUE: 支持安全模式</li><li>● FALSE: 不支持安全模式</li></ul>
isSecurit y	可选	String	body	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● TRUE: 启用</li><li>● FALSE: 未启用</li></ul>
signalStr ength	可选	String(256)	body	设备的信号强度。
sigVersio n	可选	String(256)	body	设备的sig版本。
serialNu mber	可选	String(256)	body	设备的序列号。
batteryLe vel	可选	String(256)	body	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE



#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceInfoChanged",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturer": "wulian",  
        "type": 90,  
        "model": "90",  
        "mac": "*****",  
        "swVersion": "...",  
        "fwVersion": "...",  
        "hwVersion": "...",  
        "protocolType": "zigbee",  
        "description": "smock detector"  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.4 设备数据变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备数据变化通知后（订阅的通知类型为 deviceDataChanged），当设备上报单个服务属性数据时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备数据变化通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleDeviceDataChanged(NotifyDeviceDataChangedDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyDeviceDataChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceDataChanged。
requestId	可选	String(1-128)	body	消息的序列号，唯一标识该消息。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID，用于唯一标识一个网关设备。
service	必选	DeviceService	body	设备的服务数据，具体参见下表 <a href="#">DeviceService</a> 结构体说明。

DeviceService结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String	body	服务ID。
serviceType	必选	String	body	服务的类型。
data	必选	ObjectNode	body	服务数据信息。
eventTime	必选	String	body	事件发生时间，时间格式 yyyyMMddThhmmssZ，例如 20151212T121212Z。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDataChanged",  
    "requestId": "*****",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "service": {  
        "serviceId": "Brightness",  
        "serviceType": "Brightness",  
        "data": {  
            "brightness": 80  
        },  
        "eventTime": "20170311T163657Z"  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.5 批量设备数据变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备数据批量变化通知后（订阅的通知类型为 deviceDatasChanged），当设备同时上报多个服务属性数据时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备数据批量变化通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleDeviceDatasChanged(NotifyDeviceDatasChangedDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyDeviceDatasChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceDataChanged。
requestId	可选	String(1-128)	body	消息的序列号，唯一标识该消息。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。

参数	必选/可选	类型	位置	描述
gatewayId	必选	String	body	网关ID，用于唯一标识一个网关设备。
services	必选	List<DeviceService>	body	服务列表信息，具体参见 <a href="#">DeviceService</a> 结构体说明。

DeviceService结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String	body	服务ID。
serviceType	必选	String	body	服务类型。
data	必选	ObjectNode	body	服务数据信息。
eventTime	必选	String	body	事件上报时间，时间格式：yyyymmddThhmmssZ，例如20151212T121212Z。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDatasChanged",  
    "requestId": "*****",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "service": [{  
        "serviceId": "Brightness",  
        "value": 100  
    }]  
}
```

```
"serviceType": "Brightness",
"data": {
  "brightness": 80
},
"eventTime": "20170311T163657Z"
},
{
  "serviceId": "Color",
  "serviceType": "Color",
  "data": {
    "value": "red"
  },
  "eventTime": "20170311T163657Z"
}
]
```

## 响应示例

```
response:
Status Code: 200 OK
```

### 2.3.5.6 设备服务信息变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备服务信息变化通知后（订阅的通知类型为 serviceInfoChanged），当平台向设备下发命令修改设备服务信息时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备服务信息变化通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleServiceInfoChanged(NotifyServiceInfoChangedDTO body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyServiceInfoChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	Enum	body	通知类型，取值： serviceInfoChanged。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID，用于标识一个网关设备。 当设备是直连设备时，gatewayId与设备的deviceId一致。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	必选	String	body	设备服务标识。
serviceType	必选	String	body	设备服务类型。
serviceInfo	必选	ServiceInfo	body	屏蔽的设备服务信息，增量上报， 具体参见 <a href="#">ServiceInfo结构体</a> 。

ServiceInfo结构体说明：

参数	必选/可选	类型	位置	描述
muteCmds	可选	List<String>	body	屏蔽的设备控制命令列表。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "serviceInfoChanged",  
    "deviceId": "*****",  
    "serviceId": "*****",  
    "serviceType": "*****",  
    "gatewayId": "*****",  
    "serviceInfo":  
    {  
        "muteCmds": "VIDEO_RECORD"  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.7 删除设备通知

#### 典型场景

第三方应用在物联网平台订阅了删除设备通知后（订阅的通知类型为deviceDeleted），当设备在物联网平台中被删除后，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了删除设备通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
------	--

回调接口	void handleDeviceDeleted(NotifyDeviceDeletedDTO body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceDeletedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型, 取值: deviceDeleted。
deviceId	必选	String	body	设备ID, 用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID, 用于标识一个网关设备。当设备是直连设备时, gatewayId与设备的deviceId一致。 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。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDeleted",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.8 设备消息确认通知

#### 典型场景

第三方应用在物联网平台订阅了设备消息确认通知后（订阅的通知类型为 messageConfirm），当平台向设备下发命令，并收到设备返回的命令确认消息（如命令已送达、已执行等）时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备消息确认通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleMessageConfirm(NotifyMessageConfirmDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyMessageConfirmDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	header	通知类型，取值：messageConfirm。
header	必选	MessageConfirmHeader	header	具体参见下表 <a href="#">MessageConfirmHeader结构</a> 。
body	必选	ObjectNode	body	根据业务具体定义，确认消息可以携带的状态变化等消息。

MessageConfirmHeader结构说明：

参数	必选/可选	类型	位置	描述
requestId	必选	String(1-128)	body	消息的序列号，唯一标识该消息。
from	必选	String(1-128)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● 设备发起的请求: /devices/{deviceId}</li><li>● 设备服务发起的请求: /devices/{deviceId}/services/{serviceId}</li></ul>
to	必选	String(1-128)	body	表示消息接收者的地址，To就是订阅请求中的From，如第三方应用的userId。
status	必选	String(1-32)	body	命令状态。 <ul style="list-style-type: none"><li>● sent: 已发送</li><li>● delivered: 已送达</li><li>● executed: 已执行</li></ul>
timestamp	必选	String(1-32)	body	时间戳，时间格式: yyyyMMdd'T'HHmmss'Z'，如: 20151212T121212Z。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "messageConfirm",  
    "header": {  
        "requestId": "*****",  
        "from": "*****",  
        "to": "*****",  
        "status": "delivered",  
        "timestamp": "20151212T121212Z"  
    },  
}
```

```
"body": {  
}  
}  
}
```

## 响应示例

```
response:  
  
Status Code: 200 OK
```

### 2.3.5.9 设备命令响应通知

#### 典型场景

第三方应用在物联网平台订阅了设备命令响应通知后（订阅的通知类型为 commandRsp），当平台向设备下发命令，并收到设备返回的命令响应消息（如命令执行成功或者执行失败，设备要向平台返回指定内容响应）时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备命令响应通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleCommandRsp(NotifyCommandRspDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyCommandRspDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：commandRsp。

参数	必选/可选	类型	位置	描述
header	必选	CommandRspHeader	body	具体参见下表 <a href="#">CommandRspHeader</a> 结构说明。
body	必选	ObjectNode	body	响应命令的消息内容。

CommandRspHeader结构说明：

参数	必选/可选	类型	位置	描述
requestId	必选	String(1-128)	body	消息的序列号，唯一标识该消息。
from	必选	String(1-128)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● 设备发起的请求：/devices/{deviceId}</li><li>● 设备服务发起的请求：/devices/{deviceId}/services/{serviceId}</li></ul>
to	必选	String(1-128)	body	表示消息接收者的地址，To就是订阅请求中的From，如第三方应用的userId。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
serviceType	必选	String	body	命令所属的服务类型。
method	必选	String(1-128)	body	存放的响应命令，如：INVITE-RSP。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

{

```
"notifyType": "commandRsp",
"header": {
  "requestId": "*****",
  "from": "*****",
  "to": "*****",
  "deviceId": "*****",
  "serviceType": "Camera",
  "method": "MUTE_COMMANDS"
},
"body": {}
```

## 响应示例

response:  
Status Code: 200 OK

### 2.3.5.10 设备事件通知

#### 典型场景

第三方应用在物联网平台订阅了设备事件通知后（订阅的通知类型为deviceEvent），当平台收到设备上报的事件消息时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备事件通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleDeviceEvent(NotifyDeviceEventDTO body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceEventDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型, 取值: deviceEvent。
header	必选	DeviceEventHeader	body	具体参见下表 <b>DeviceEventHeader</b> 结构说明。
body	必选	ObjectNode	body	响应命令的消息内容。

DeviceEventHeader结构说明:

参数	必选/可选	类型	位置	描述
eventType	必选	String(1-2)	body	事件类型。
from	必选	String(1-28)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● 设备发起的请求: /devices/{deviceId}</li><li>● 设备服务发起的请求: /devices/{deviceId}/services/{serviceId}</li></ul>
timestamp	必选	String(1-32)	body	时间戳, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
eventTime	必选	String(1-32)	body	事件上报时间, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
deviceId	必选	String	body	设备ID, 用于唯一标识一个设备。
serviceType	必选	String	body	命令所属的服务类型。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceEvent",  
    "header": {  
        "eventType": "*****",  
        "from": "/devices/{deviceId}/services/{serviceId}",  
        "deviceId": "*****",  
        "serviceType": "*****",  
        "timestamp": "20151212T121212Z",  
        "eventTime": "20151212T121212Z",  
    },  
    "body": {  
        "usedPercent": 80  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.11 增加设备模型通知

#### 典型场景

第三方应用在物联网平台订阅了设备模型增加通知后（订阅的通知类型为 deviceModelAdded），在物联网平台上新增设备Profile文件时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备模型增加通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容

3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleDeviceModelAdded(NotifyDeviceModelAddedDTO body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceModelAddedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：deviceModelAdded。
appId	必选	String	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceType	必选	String	body	设备的类型。
manufacturerName	必选	String	body	设备模型的厂商名称。
manufacturerId	必选	String	body	设备模型的厂商ID。
model	必选	String	body	设备型号。
protocolType	必选	String	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceModelAdded",  
    "appId": "*****",  
    "deviceType": "*****",  
    "manufacturerName": "wulian",  
    "manufacturerId": "*****",  
    "model": "*****",  
    "protocolType": "zigbee"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.12 删除设备模型通知

#### 典型场景

第三方应用在物联网平台订阅了设备模型删除通知后（订阅的通知类型为 deviceModelDeleted），在物联网平台上删除设备Profile文件时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备模型删除通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleDeviceModelDeleted(NotifyDeviceModelDeletedDTO body)

接口所属类	PushMessageReceiver
-------	---------------------

## 参数说明

NotifyDeviceModelDeletedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型, 取值: deviceModelDeleted。
appId	必选	String	body	第三方应用的身份标识, 用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceType	必选	String	body	设备的类型。
manufacturerName	必选	String	body	设备模型的厂商名称。
manufacturerId	必选	String	body	设备模型的厂商ID。
model	必选	String	body	设备型号。
protocolType	必选	String	body	设备使用的协议类型, 当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

{

"notifyType": "deviceModelAdded",

"appId": "\*\*\*\*\*",

```
"deviceTyp": "*****",
"manufacturerName": "*****",
"manufacturerId": "*****",
"model": "*****",
"protocolType": "*****"
}
```

## 响应示例

```
response:
Status Code: 200 OK
```

### 2.3.5.13 设备影子状态变更通知

#### 典型场景

第三方应用在物联网平台订阅了设备影子状态变更通知后（订阅的通知类型为 deviceDesiredPropertiesModifyStatusChanged），当平台中的设备影子向设备同步数据成功或失败时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备影子状态变更通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleDeviceDesiredStatusChanged(NotifyDeviceDesiredStatusChangedDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyDeviceDesiredStatusChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型, 取值: deviceDesiredPropertiesModifyStatusChanged。
deviceId	必选	String	body	设备ID, 用于唯一标识一个设备。
serviceId	必选	String	body	服务ID, 用于标识一个服务。
properties	必选	ObjectNode	body	设备影子数据属性
status	必选	String	body	状态, 取值DELIVERED/FAILED。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDesiredPropertiesModifyStatusChanged",  
    "deviceId": "*****",  
    "serviceId": "Device",  
    "properties": {  
        "Model Number" : 1,  
        "Serial Number" : 2,  
        "Firmware Version" : "v1.1.0"  
    },  
    "status": "DELIVERED"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.14 软件升级状态变更通知

#### 典型场景

第三方应用在物联网平台订阅了软件升级状态变更通知后（订阅的通知类型为 swUpgradeStateChangeNotify），当设备进行软件升级，升级任务的状态发生变化时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了软件升级状态变更通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleSwUpgradeStateChanged(NotifySwUpgradeStateChangedDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifySwUpgradeStateChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值： swUpgradeStateChangeNotify。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	软件升级任务ID。
subOperationId	必选	String	body	软件升级子任务ID。

参数	必选/可选	类型	位置	描述
swUpgradeState	必选	String	body	软件升级状态。 <ul style="list-style-type: none"><li>● downloading: 设备正在下载软件包</li><li>● downloaded: 设备下载软件包完成</li><li>● updating: 设备正在进行升级</li><li>● idle: 设备处于空闲状态</li></ul>

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "swUpgradeStateChangeNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "swUpgradeState": "downloading"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.15 软件升级结果变更通知

#### 典型场景

第三方应用在物联网平台订阅了软件升级结果通知后（订阅的通知类型为 swUpgradeResultNotify），当设备的软件升级任务执行完成时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了软件升级结果通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleSwUpgradeResult(NotifySwUpgradeResultDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifySwUpgradeResultDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：swUpgradeResultNotify。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	软件升级任务ID。
subOperationId	必选	String	body	软件升级子任务ID。
curVersion	必选	String	body	设备当前的软件版本。
targetVersion	必选	String	body	设备要升级的目标软件版本。

参数	必选/可选	类型	位置	描述
sourceVersion	必选	String	body	设备的源软件版本。
swUpgradeResult	必选	String	body	软件升级结果。 ● SUCCESS: 设备升级成功 ● FAIL: 设备升级失败
upgradeTime	必选	String	body	升级时长。
resultDesc	必选	String	body	升级结果描述。
errorCode	必选	String	body	设备上报的状态错误码。
description	必选	String	body	错误原因描述。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "swUpgradeResultNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "curVersion": "1.3",  
    "targetVersion": "1.5",  
    "sourceVersion": "1.0",  
    "swUpgradeResult": "SUCCESS",  
    "upgradeTime": "***",  
    "resultDesc": "***",  
    "errorCode": "***",
```

```
"description": "****",  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 2.3.5.16 固件升级状态变更通知

#### 典型场景

第三方应用在物联网平台订阅了固件升级状态变更通知后（订阅的通知类型为 fwUpgradeStateChangeNotify），当设备进行固件升级，升级任务的状态发生变化时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了固件升级状态变更通知的第三方应用推送通知消息。

#### 注意事项

- 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
- 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
- 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleFwUpgradeStateChanged(NotifyFwUpgradeStateChangedDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyFwUpgradeStateChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值： fwUpgradeStateChangeNotify。

参数	必选/可选	类型	位置	描述
deviceid	必选	String	body	设备ID，用于唯一标识一个设备。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	固件升级任务ID。
subOperationId	必选	String	body	固件升级子任务ID。
step	必选	String	body	固件升级状态，可取值为0、1、2、3。
stepDesc	必选	String	body	升级状态描述。 <ul style="list-style-type: none"><li>● 1: downloading: 设备正在下载软件包</li><li>● 2: downloaded: 设备下载软件包完成</li><li>● 3: updating: 设备正在进行升级</li><li>● 0: idle: 设备处于空闲状态</li></ul>

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "fwUpgradeStateChangeNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "step": "1",  
    "stepDesc": "downloading"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 2.3.5.17 固件升级结果变更通知

#### 典型场景

第三方应用在物联网平台订阅了固件升级结果通知后（订阅的通知类型为 fwUpgradeResultNotify），当设备的固件升级任务执行完成时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了固件升级结果通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口
2. 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的[消息推送](#)章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	void handleFwUpgradeResult(NotifyFwUpgradeResultDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyFwUpgradeResultDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：fwUpgradeResultNotify。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	固件升级任务ID。
subOperationId	必选	String	body	固件升级子任务ID。

参数	必选/可选	类型	位置	描述
curVersion	必选	String	body	设备当前的固件版本。
targetVersion	必选	String	body	设备要升级的目标固件版本。
sourceVersion	必选	String	body	设备的源固件版本。
status	必选	String	body	升级结果。 <ul style="list-style-type: none"><li>● SUCCESS</li><li>● FAIL</li></ul>
statusDesc	必选	String	body	升级结果描述。 <ul style="list-style-type: none"><li>● SUCCESS: 设备升级成功</li><li>● FAIL: 设备升级失败</li></ul>
upgradeTime	必选	String	body	固件升级时长。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "fwUpgradeResultNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "curVersion": "1.6",  
    "targetVersion": "1.6",  
    "sourceVersion": "1.3",  
    "status": "SUCCESS",  
    "statusDesc": "****",
```

```
"upgradeTime":"****"  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 2.3.5.18 NB 设备命令状态变化通知

#### 典型场景

第三方应用通过物联网平台创建设备命令时，如果设置了callbackurl回调地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达），平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台在命令状态发生变化时向第三方应用推送通知消息。

#### 注意事项

- 应用在[创建NB设备命令](#)需要设置回调地址为接口描述中的回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口
- 应用通过继承PushMessageReceiver类并重写回调接口接收推送消息的内容
- 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver/cmd
回调接口	void handleNBCommandStateChanged(NotifyNBCommandStatusChanged DTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyNBCommandStatusChangedDTO

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
commandId	必选	String	body	命令ID，在创建设备命令时由平台生成的。

参数	必选/可选	类型	位置	描述
result	必选	NBCommandResult	body	详见下表 <b>NBCommandResult</b> 结构说明

NBCommandResult结构说明

参数	必选/可选	类型	位置	描述
resultCode	必选	String	body	命令状态结果 <ul style="list-style-type: none"><li>● SENT, 平台已下发给设备, 但还未收到设备响应</li><li>● DELIVERED, 平台收到设备响应, 命令已到达设备</li><li>● SUCCESS, 平台收到命令结果上报, 且上报的结果是成功</li><li>● FAIL, 平台收到命令结果上报, 且上报的结果是失败</li></ul>
resultDetail	必选	Object Node	body	设备上报命令结果上报时, 携带的自定义字段

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "deviceId": "*****",  
    "commandId": "*****",  
    "result": {  
        "resultCode": "DELIVERED",  
        "resultDetail": null  
    }  
}
```

{}

## 响应示例

```
response:  
Status Code: 200 OK
```

## 2.3.6 命令下发（NB 命令）

### 2.3.6.1 创建设备命令

#### 典型场景

设备Profile文件中定义了物联网平台可向设备下发的命令，第三方应用可调用此接口向设备下发命令，配置或修改设备的服务属性，以实现对设备的控制。

物联网平台有两种命令下发机制：

- 立即下发：平台收到命令后立即下发给设备，保证实时性，不保证串行性。
- 缓存下发：平台收到命令后缓存起来，在设备可达时排队串行下发，平台收到前一个命令的响应（模组自动回复的ACK）才会下发后一个命令，保证串行性，不保证实时性。

#### 接口功能

支持第三方应用向设备下发命令，实现对设备的控制。支持物联网平台立即下发命令和缓存下发命令。

#### 接口描述

```
PostDeviceCommandOutDTO2 postDeviceCommand(PostDeviceCommandInDTO2 pdcInDTO, String appId, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
pdcInDTO	必选	PostDeviceCommandInDTO2	body	见下表 PostDeviceCommandInDTO2 的描述。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

## PostDeviceCommandInDTO2

参数	必选/可选	类型	位置	描述
deviceId	必选	String(64)	body	下发命令的设备ID，用于唯一标识一个设备。
command	必选	CommandDTOV4	body	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
callbackUrl	可选	String(1024)	body	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	可选	Integer(>=0)	body	下发命令的有效时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。 如果expireTime设置为0，则无论物联网平台上设置的命令下发模式是什么，该命令都会立即下发给设备（如果设备处于眠状态或者链路已老化，则设备收不到命令，平台没收到设备的响应，该命令最终会超时）。
maxRetransmit	可选	Integer(0~3)	body	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-64)	body	命令对应的服务ID，用于标识一个服务。要与profile中定义的serviceId保持一致。
method	必选	String(1-128)	body	命令服务下具体的命令名称，服务属性等。要与profile中定义的命令名保持一致。
paras	必选	ObjectNode	body	命令参数，jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。 若profile中命令未定义参数，则paras值中的参数可不填写，即设置为"paras": {}。

## 响应参数

PostDeviceCommandOutDTO2

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4结构体</a> 。
callbackUrl	String(1024)	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● PENDING表示未下发</li><li>● EXPIRED表示命令已经过期</li><li>● SUCCESSFUL表示命令已经成功执行</li><li>● FAILED表示命令执行失败</li><li>● TIMEOUT表示命令下发执行超时</li><li>● CANCELED表示命令已经被撤销执行</li><li>● DELIVERED表示命令已送达设备</li><li>● SENT表示命令正在下发</li></ul>
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	100223	Command counts has reached the upLimit.	缓存的命令数已到达限制。处于PENDING状态的命令条数不超过限定期。默认限定期为20。 处理建议：如当前平台缓存的命令还需执行，请让设备上报数据触发缓存命令的下发；如当前平台缓存的命令有不需要执行的命令，请调用“修改设备命令V4”接口，将命令的状态从PENDING修改为CANCELED。

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	100612	Device is zombie.	设备为僵尸设备。（距离设备上次上线时间超过阈值，默认7天） 处理建议：请把设备上线后再重新下发命令。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 2.3.6.2 查询设备命令

#### 典型场景

第三方应用向设备下发命令后，可调用此接口在物联网平台查询下发命令的状态及内容信息，以了解命令的执行情况。

#### 接口功能

支持第三方应用在物联网平台查询下发命令的状态和内容信息，可查询指定时间段内本应用下的所有下发命令，或者指定设备的所有下发命令。

#### 接口描述

```
QueryDeviceCommandOutDTO2 queryDeviceCommand(QueryDeviceCommandInDTO2 qdcInDTO, String  
accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qdcInDTO	必选	QueryDeviceCommandInDTO2	query	见下表QueryDeviceCommandInDTO2的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### QueryDeviceCommandInDTO2

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer(>=0)	query	查询的页码，大于等于0，默认值：0。
pageSize	可选	Integer(>=1&& <=1000)	query	查询每页信息的数量，大于等于1，最大值1000，默认值：1000。
deviceId	可选	String(64)	query	指定查询命令的设备ID。
startTime	可选	String	query	查询开始时间，查询下发命令时间在startTime之后的记录。时间格式： yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。

参数	必选/可选	类型	位置	描述
endTime	可选	String	query	查询结束时间，查询下发命令时间在endTime之前的记录。时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。
appId	可选	String	query	如果是本应用的设备命令，此参数值可以填写null，否则填写授权应用的appId。

## 响应参数

QueryDeviceCommandOutDTO2

参数	类型	描述
pagination	Pagination	页面信息，具体参见 <a href="#">Pagination结构体</a> 。
data	List<DeviceCommandRespV4>	设备命令列表，具体参见 <a href="#">DeviceCommandRespV4结构体</a> 。

Pagination结构体说明：

参数	类型	描述
pageNo	long	页码。
pageSize	long	每页信息数量。
totalSize	long	记录总数。

DeviceCommandRespV4结构体说明：

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4结构体</a> 。

参数	类型	描述
callbackUrl	String(1024)	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● PENDING表示未下发</li><li>● EXPIRED表示命令已经过期</li><li>● SUCCESSFUL表示命令已经成功执行</li><li>● FAILED表示命令执行失败</li><li>● TIMEOUT表示命令下发执行超时</li><li>● CANCELED表示命令已经被撤销执行</li><li>● DELIVERED表示命令已送达设备</li><li>● SENT表示命令正在下发中</li></ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	类型	描述
serviceId	必选	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	必选	String(1-128)	命令服务下具体的命令名称，服务属性等。
paras	可选	Object	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议： <ul style="list-style-type: none"><li>● 请确认startTime和endTime都不为空，且endTime比startTime晚。</li><li>● 请确认pageNo不为空，且pageNo大于0。</li><li>● 请检查pageSize不为空，且pageSize大于1。</li></ul>

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 2.3.6.3 修改设备命令

#### 典型场景

第三方应用向设备下发命令后，如果由于命令排队、设备不在线等原因，物联网平台还未把命令下发给设备执行，此时第三方应用可调用此接口修改该命令的状态信息。当前仅支持把命令状态修改为CANCELED状态，即撤销命令的执行。

## 接口功能

支持第三方应用修改指定命令的状态信息，仅能修改还处于PENDING状态的命令，且当前仅支持修改的命令状态为CANCELED，即撤销命令。

## 接口描述

```
UpdateDeviceCommandOutDTO updateDeviceCommand(UpdateDeviceCommandInDTO udcInDTO, String deviceCommandId, String appId, String accessToken) throws NorthApiException
```

## 参数说明

参数	必选/可选	类型	位置	描述
udcInDTO	必选	UpdateDeviceCommandInDTO	body	见下表 UpdateDeviceCommandInDTO的描述。
deviceCommandId	必选	String	path	要修改的命令ID，在调用创建设备命令接口后获得。
appId	必选	String	query	如果是本应用的设备命令，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

### UpdateDeviceCommandInDTO

参数	必选/可选	类型	描述
status	必选	String	命令执行结果，可选值： CANCELED， 撤销命令。

## 响应参数

### UpdateDeviceCommandOutDTO

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。

参数	类型	描述
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4结构体</a> 。
callbackUrl	String(1024)	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"> <li>● PENDING表示未下发</li> <li>● EXPIRED表示命令已经过期</li> <li>● SUCCESSFUL表示命令已经成功执行</li> <li>● FAILED表示命令执行失败</li> <li>● TIMEOUT表示命令下发执行超时</li> <li>● CANCELED表示命令已经被撤销执行</li> <li>● DELIVERED表示命令已送达设备</li> <li>● SENT表示命令正在下发中</li> </ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-64)	body	命令对应的服务ID，用于标识一个服务。

参数	必选/可选	类型	位置	描述
method	必选	String(1-128)	body	命令服务下具体的命令名称，服务属性等。
paras	可选	Object	body	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>请检查该设备的Profile文件是否已上传到物联网平台。</li><li>请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>

Http状态码	错误码	错误描述	说明
200	100434	The device command is not existed.	设备命令不存在。 处理建议：请检查接口请求中的设备命令ID是否正确。
200	100435	The device command already canceled, expired or executed, Cannot cancel.	设备命令已经取消，过期或执行，不能取消。
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 2.3.6.4 创建设备命令撤销任务

#### 典型场景

第三方应用向设备下发命令后，如果由于命令排队、设备不在线等原因，物联网平台还未把命令下发给设备执行（DEFAULT状态），此时第三方应用可调用此接口撤销指定设备的所有未下发的命令，对于已下发成功的命令不可撤销。

#### 接口功能

支持第三方应用创建设备命令撤销任务，该任务用于撤销物联网平台中指定设备ID下所有未下发的命令（命令处于DEFAULT状态）。

#### 接口描述

```
CreateDeviceCmdCancelTaskOutDTO createDeviceCmdCancelTask(CreateDeviceCmdCancelTaskInDTO  
cdccInDTO, String appId, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
cdccInDTO	必选	CreateDeviceCmdCancelTaskInDTO	body	见下表 CreateDeviceCmdCancelTaskInDTO 的描述。
appId	必选	String	query	如果是本应用的设备命令，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

CreateDeviceCmdCancelTaskInDTO

参数	必选/可选	类型	位置	描述
deviceId	必选	String(1-64)	body	待撤销设备命令的设备ID，撤销任务将会撤销所有向该设备下发的命令。

#### 响应参数

CreateDeviceCmdCancelTaskOutDTO

参数	类型	描述
taskId	String(1-64)	设备命令撤销任务的任务ID。
appId	String(1-64)	设备命令撤销任务所属的应用ID。
deviceId	String(1-64)	执行设备命令撤销任务的设备ID。
status	String	撤销任务的任务状态。 <ul style="list-style-type: none"><li>● WAITTING表示等待执行中</li><li>● RUNNING表示撤销任务正在执行</li><li>● SUCCESS表示撤销任务执行成功</li><li>● FAILED表示撤销任务执行失败</li><li>● PART_SUCCESS表示撤销任务部分执行成功。</li></ul>
totalCount	Integer	撤销的设备命令总数。
deviceCommands	List<DeviceCommandRespV4>	撤销的设备命令列表，具体参见 <a href="#">DeviceCommandRespV4结构体</a> 。

DeviceCommandRespV4结构体说明：

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4结构体</a> 。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。

参数	类型	描述
status	String	下发命令的状态。 <ul style="list-style-type: none"> <li>● DEFAULT表示未下发</li> <li>● EXPIRED表示命令已经过期</li> <li>● SUCCESSFUL表示命令已经成功执行</li> <li>● FAILED表示命令执行失败</li> <li>● TIMEOUT表示命令下发执行超时</li> <li>● CANCELED表示命令已经被撤销执行</li> </ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTOV4结构体说明:

参数	必选/可选	类型	描述
serviceId	必选	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	必选	String(1-128)	命令服务下具体的命令名称，服务属性等。
paras	可选	Object	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径（url）中传入的appId是否有误。</li> </ul>

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。

Http状态码	错误码	错误描述	说明
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 2.3.6.5 查询设备命令撤销任务

#### 典型场景

第三方应用创建了设备命令撤销任务后，可调用此接口查询设备命令撤销任务的详细信息和执行状态，以了解撤销任务的执行情况。

#### 接口功能

支持第三方应用在物联网平台查询设备命令撤销任务信息和状态，可指定条件查询单个或多个撤销任务。

#### 接口描述

```
QueryDeviceCmdCancelTaskOutDTO queryDeviceCmdCancelTask(QueryDeviceCmdCancelTaskInDTO
qdcctInDTO, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qdccetInDT O	必选	QueryDevice CmdCancelT askInDTO2	query	见下表 QueryDeviceCmdCancelTaskInDT O2的描述。
accessToke n	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接 口，此参数值可以填写null，否 则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

#### QueryDeviceCmdCancelTaskInDTO2

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer(>=0 )	query	查询的页码，大于等于0，默认值： 0。
pageSize	可选	Integer(>=1 &&≤1000)	query	查询每页信息的数量，大于等于1， 最大值1000，默认值：1000。
taskId	可选	String	query	撤销任务的任务ID。

参数	必选/可选	类型	位置	描述
deviceId	可选	String	query	执行设备命令撤销任务的设备ID。
status	可选	String	query	设备命令撤销任务的状态。
startTime	可选	String	query	查询开始时间, 查询创建撤销设备命令任务时间在 startTime之后的记录。 时间格式: yyyyMMdd'T'HHmmss'Z', 如 20151212T121212Z。
endTime	可选	String	query	查询结束时间, 查询创建撤销设备命令任务时间在 endTime之前的记录。 时间格式: yyyyMMdd'T'HHmmss'Z', 如 20151212T121212Z。
appId	可选	String	query	如果是本应用的设备命令, 此参数值可以填写null, 否则填写授权应用的 appId。

## 响应参数

QueryDeviceCmdCancelTaskOutDTO2

参数	类型	描述
pagination	Pagination	页码信息, 具体参见 <a href="#">Pagination结构体</a> 。
data	List<DeviceCommandCancelTaskRespV4>	设备命令列表, 具体参见 <a href="#">DeviceCommandCancelTaskRespV4结构体</a> 。

Pagination结构体说明:

参数	类型	描述
pageNo	long	查询结果页码。
pageSize	long	每页信息数量。
totalSize	long	记录总数, 即查询到的撤销任务中的命令总数。

DeviceCommandCancelTaskRespV4结构体说明:

参数	类型	描述
taskId	String(1-64)	设备命令撤销任务的任务ID。
appId	String(1-64)	设备命令撤销任务所属的应用ID。
deviceId	String(1-64)	设备命令撤销任务指定撤销命令的设备ID。
status	String	撤销任务的任务状态。 <ul style="list-style-type: none"><li>● WAITTING表示等待执行中</li><li>● RUNNING表示撤销任务正在执行</li><li>● SUCCESS表示撤销任务执行成功</li><li>● FAILED表示撤销任务执行失败</li><li>● PART_SUCCESS表示撤销任务部分执行成功。</li></ul>
totalCount	Integer	撤销的设备命令总数。
deviceCommands	List<DeviceCommandRespV4>	设备命令撤销任务信息列表，具体参见 <a href="#">DeviceCommandRespV4结构体</a> 。

DeviceCommandRespV4结构体说明：

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4结构体</a> 。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。

参数	类型	描述
status	String	下发命令的状态。 <ul style="list-style-type: none"> <li>● DEFAULT表示未下发</li> <li>● EXPIRED表示命令已经过期</li> <li>● SUCCESSFUL表示命令已经成功执行</li> <li>● FAILED表示命令执行失败</li> <li>● TIMEOUT表示命令下发执行超时</li> <li>● CANCELED表示命令已经被撤销执行</li> </ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTOV4结构体说明:

参数	必选/可选	类型	描述
serviceId	必选	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	必选	String(1-128)	命令服务下具体的命令名称，服务属性等。
paras	可选	Object	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径（url）中传入的appId是否有误。</li> </ul>

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。

Http状态码	错误码	错误描述	说明
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

## 2.3.7 命令下发（非 NB 命令）

### 2.3.7.1 设备服务调用

#### 典型场景

设备Profile文件中定义了物联网平台可向设备下发的命令，第三方应用可调用此接口向设备下发命令，配置或修改设备的服务属性，以实现对设备的控制。

物联网平台不对设备命令进行缓存而是直接下发命令，设备不在线时命令下发失败。下发命令的具体格式需要第三方应用与设备自定义，物联网平台在接口中直接进行封装，透传。



#### 说明

目前该接口仅用于安装了IoT Agent/AgentLite的网关设备，通过向网关设备下发命令，可对网关下的非直连设备进行控制。

#### 接口功能

支持第三方应用向设备立即下发命令（当前仅支持安装了IoT Agent/AgentLite的网关设备），实现对设备的控制，支持给本应用的设备下发命令。

#### 接口描述

```
InvokeDeviceServiceOutDTO invokeDeviceService(String deviceId, String serviceId, CommandDTO2  
commandDTO, String appId, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String(1-64)	path	设备ID，用于唯一标识一个设备。
serviceId	必选	String(1-64)	path	服务ID，唯一标识一个服务。
commandDTO	必选	CommandDTO2	body	见下表 <b>CommandDTO2</b> 的描述。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。

参数	必选/可选	类型	位置	描述
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## CommandDTO2

参数	必选/可选	类型	位置	描述
header	必选	CommandNA2CloudHeader	body	具体参见 <a href="#">CommandNA2CloudHeader</a> 结构体。
body	可选	Object	body	消息的消息体。JsonObject里面是一个个键值对。每个键都是profile中命令的参数名（paraName）。

## CommandNA2CloudHeader结构体说明：

参数	必选/可选	类型	位置	描述
requestId	可选	String(0-128)	body	用于标识一个命令，不能重复。
mode	必选	Enum	body	是否要确认消息。 <ul style="list-style-type: none"><li>● NOACK：不需要确认消息</li><li>● ACK：需要确认消息</li><li>● 其它值无效</li></ul>
from	可选	String(1-28)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● App发起的请求：/users/{userId}</li><li>● 第三方应用发起的请求：/{serviceName}</li><li>● 物联网平台发起的请求：/cloud/{serviceName}</li></ul>
toType	可选	Enum	body	消息接受者的类型，CLOUD/GATEWAY。
to	可选	String(1-28)	body	消息接收者的地址。

参数	必选/可选	类型	位置	描述
method	必选	String(1-32)	body	命令名称，如：“DISCOVERY”为发现非直连设备，“REMOVE”为删除非直连设备。
callbackURL	可选	String(1024)	body	命令的回调路径。

## 响应参数

InvokeDeviceServiceOutDTO

参数	类型	描述
status	String(128)	命令状态。 <ul style="list-style-type: none"> <li>● sent: 已发送</li> <li>● delivered: 已投递, toType为CLOUD的请求才有该返回值</li> <li>● failed: 失败, toType为CLOUD的请求才有该返回值</li> </ul>
timestamp	String(128)	发送命令的时间戳, 时间格式: yyyyMMdd'T'HHmmss'Z', 如20151212T121212Z。
requestId	String(128)	设备命令ID。 <ul style="list-style-type: none"> <li>● toType为GATEWAY时, 如果请求有requestId则同请求的requestId一致, 如果请求无requestId则平台分配一个序列号。</li> <li>● toType为CLOUD时, 则为空。</li> </ul>

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径(url)中传入的appId是否有误。</li> </ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与网关的连接是否正常。
200	100432	The device command is muted.	设备命令已被禁用。 处理建议：请检查接口请求参数method中的命令是否有误。
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	102203	CommandName is invalid.	命令名称无效。 处理建议：请检查接口请求参数method中携带的命令是否有误。
403	100450	The gateway is not online.	网关不在线。 处理建议：请检查网关与物联网平台的连接是否正常。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100444	The serviceType is not exist.	服务类型不存在。 处理建议：请检查接口请求参数toType中的服务类型是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

Http状态码	错误码	错误描述	说明
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	50252	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

## 2.3.8 数据采集

物联网平台支持第三方应用查询设备的基本信息，同时还支持查看设备上报的历史数据，能按时、天、月等维度查看设备上报的历史数据。

### 2.3.8.1 查询单个设备信息

#### 典型场景

已在物联网平台注册的设备，第三方应用若需要查看某个设备的详细信息（如厂商、型号、版本、状态和服务属性等信息），可调用此接口在物联网平台查询获取。

#### 接口功能

支持第三方应用在物联网平台根据设备ID查询指定设备的详细信息，包括配置信息、状态信息、服务属性信息等。

#### 接口描述

```
QuerySingleDeviceInfoOutDTO querySingleDeviceInfo(String deviceId, String select, String appId,  
String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
select	可选	String	query	指定查询条件，可选值：imsi。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。

参数	必选/可选	类型	位置	描述
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

QuerySingleDeviceInfoOutDTO

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
gatewayId	String(256)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	Enum	节点类型，取值：ENDPOINT/GATEWAY/UNKNOW。
createTime	String(256)	创建设备的时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。
lastModifiedTime	String(256)	最后修改设备的时间。
deviceInfo	DeviceInfo	设备信息，具体参见 <a href="#">DeviceInfo结构体</a> 。
services	List<DeviceService>	设备服务列表，具体参见 <a href="#">DeviceService结构体</a> 。

DeviceInfo结构体说明：

参数	类型	描述
nodeId	String(256)	设备的唯一标识。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacturerId	String(256)	厂商ID，唯一标识一个厂商。
manufacturerName	String(256)	厂商名称。
mac	String(256)	设备的MAC地址。

参数	类型	描述
location	String(2048)	设备的位置信息。
deviceType	String(256)	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号。 Z-Wave: ProductType + ProductId, 16 进制格式XXXX-XXXX 补0对齐, 如: 001A-0A12, 其他协议的格式待定。
swVersion	String(256)	设备的软件版本。 Z-Wave: 主版本号.次版本号, 如: 1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
protocolType	String(256)	设备使用的协议类型, 当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	String(256)	Bridge标识, 表示设备通过哪个Bridge接入物联网平台。
status	String	设备的状态, 表示设备是否在线, 取值范围: ONLINE、OFFLINE、ABNORMAL。
statusDetail	String(256)	设备的状态详情, 跟status取值对应, 取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	String	表示设备是否处于冻结状态, 即设备上报数据时, 平台是否会管理和保存。 <ul style="list-style-type: none"><li>● TRUE: 冻结状态</li><li>● FALSE: 非冻结状态</li></ul>
supportedSecurity	String	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● TRUE: 支持安全模式</li><li>● FALSE: 不支持安全模式</li></ul>
isSecurity	String	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● TRUE: 启用</li><li>● FALSE: 未启用</li></ul>
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。
batteryLevel	String(256)	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceService结构体说明:

参数	类型	描述
serviceId	String(256)	设备的服务标识。
serviceType	String(256)	设备的服务类型。
serviceInfo	ServiceInfo	屏蔽的设备服务信息，具体参见 <a href="#">ServiceInfo结构体</a> 。
data	ObjectNode(2097152)	属性值对。
eventTime	String(256)	时间格式: yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。

ServiceInfo结构体说明:

参数	类型	描述
muteCmds	List<String>	屏蔽的设备控制命令列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100405	The request parameter is invalid.	请求参数不合法。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.8.2 批量查询设备信息列表

#### 典型场景

已在物联网平台注册的设备，第三方应用若需要根据条件查看多个设备的详细信息（如厂商、型号、版本、状态和服务属性等信息），可调用此接口在物联网平台查询获取。

#### 接口功能

支持第三方应用在物联网平台根据条件查询多个设备的详细信息，包括配置信息、状态信息、服务属性信息等。

#### 接口描述

```
QueryBatchDevicesInfoOutDTO queryBatchDevicesInfo(QueryBatchDevicesInfoInDTO qbdiInDTO, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qbdiInDTO	必选	QueryBatchDevicesInfoInDTO	query	见下表 <a href="#">PostDeviceCommandInDTO2</a> 的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### PostDeviceCommandInDTO2

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
gatewayId	可选	String	query	网关ID，用于标识一个网关设备。
nodeType	可选	String	query	节点类型，取值：ENDPOINT/GATEWAY/UNKNOW。
deviceType	可选	String	query	设备类型。

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer	query	查询的页码。 <ul style="list-style-type: none"> <li>● 值为空时查询内容不分页</li> <li>● 值大于等于0的整数时分页查询</li> <li>● 值等于0时查询第一页</li> </ul>
pageSize	可选	Integer	query	查询每页信息的数量，缺省值：1。
status	可选	String	query	查询设备的状态。 <ul style="list-style-type: none"> <li>● ONLINE：在线</li> <li>● OFFLINE：不在线</li> <li>● ABNORMAL：异常状态</li> </ul>
startTime	可选	String	query	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	query	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	query	指定返回记录的排序。 <ul style="list-style-type: none"> <li>● ASC：按注册设备的时间升序排列</li> <li>● DESC：按注册设备的时间降序排列</li> </ul> 缺省值：DESC。
select	可选	String	query	指定返回记录，可取值：imsi。

## 响应参数

QueryBatchDeviceInfoOutDTO

参数	类型	描述
totalCount	long	查询的记录数量。
pageNo	long	查询的页码。
pageSize	long	查询每页信息的数量。
devices	List<QuerySingleDeviceInfoOutDTO>	设备分页列表信息，具体参见 <a href="#">QuerySingleDeviceInfoOutDTO</a> 结构体。

QuerySingleDeviceInfoOutDTO结构体说明：

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
gatewayId	String(256)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	Enum	节点类型，取值：ENDPOINT/GATEWAY/UNKNOW。
createTime	String(256)	创建设备的时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。
lastModifiedTime	String(256)	最后修改设备的时间。
deviceInfo	DeviceInfoQueryDTO	设备信息，具体参见 <a href="#">DeviceInfo结构体</a> 。
services	List<DeviceService>	设备服务列表，具体参见 <a href="#">DeviceService结构体</a> 。

DeviceInfo结构体说明：

参数	类型	描述
nodeId	String(256)	标识设备的唯一ID。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacturerId	String(256)	厂商ID，唯一标识一个厂商。
manufacturerName	String(256)	厂商名称。
mac	String(256)	设备的MAC地址。
location	String(2048)	设备的位置信息。
deviceType	String(256)	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号。 Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。

参数	类型	描述
swVersion	String(256)	设备的软件版本。 Z-Wave: 主版本号.次版本号, 如: 1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
protocolType	String(256)	设备使用的协议类型, 当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	String(256)	Bridge标识, 表示设备通过哪个Bridge接入物联网平台。
status	String	设备的状态, 表示设备是否在线, 取值范围: ONLINE、OFFLINE、ABNORMAL。
statusDetail	String(256)	设备的状态详情, 跟status取值对应, 取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	String	表示设备是否处于冻结状态, 即设备上报数据时, 平台是否会管理和保存。 <ul style="list-style-type: none"> <li>● TRUE: 冻结状态</li> <li>● FALSE: 非冻结状态</li> </ul>
supportedSecurity	String	表示设备是否支持安全模式。 <ul style="list-style-type: none"> <li>● TRUE: 支持安全模式</li> <li>● FALSE: 不支持安全模式</li> </ul>
isSecurity	String	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"> <li>● TRUE: 启用</li> <li>● FALSE: 未启用</li> </ul>
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。
batteryLevel	String(256)	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING

status	statusDetail
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceService结构体说明：

参数	类型	描述
serviceId	String(256)	设备的服务标识。
serviceType	String(256)	设备的服务类型。
serviceInfo	ServiceInfo	屏蔽的设备服务信息，具体参见 <a href="#">ServiceInfo结构体</a> 。
data	ObjectNode(209 7152)	属性值对（Attribute-value pair）。
eventTime	String(256)	时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。

ServiceInfo结构体说明：

参数	类型	描述
muteCmds	List<String>	屏蔽的设备控制命令列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。
400	100218	The gatewayId and pageNo can't be both null.	网关ID和pageNo不能同时为空。 处理建议：请检查请求参数中 gatewayId或pageNo是否填写。

Http状态码	错误码	错误描述	说明
400	100405	The request parameter is invalid.	请求参数不合法。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景， 请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.8.3 查询设备历史数据

#### 典型场景

在日常运行中，物联网平台会收到并保存设备上报业务数据（设备数据的保存时间可通过“修改设备信息”接口配置，最长保存90天），第三方应用若需要查看某个设备上报到平台的历史数据，可调用此接口查询获取。

## 接口功能

支持第三方应用根据设备ID，查询指定设备上报到物联网平台的历史数据。

## 接口描述

```
QueryDeviceDataHistoryOutDTO queryDeviceDataHistory(QueryDeviceDataHistoryInDTO qddhInDTO, String accessToken) throws NorthApiException
```

## 参数说明

参数	必选/可选	类型	位置	描述
qddhInDTO	必选	QueryDeviceDataHistoryInDTO	query	见下表 QueryDeviceDataHistoryInDTO的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

### QueryDeviceDataHistoryInDTO

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
deviceId	必选	String	query	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	query	网关ID，用于标识一个网关设备。
serviceId	可选	String	query	设备的服务标识。
property	可选	String	query	服务属性数据。
pageNo	可选	Integer	query	查询的页码。 <ul style="list-style-type: none"><li>● 值为空时查询内容不分页</li><li>● 值大于等于0的整数时分页查询</li><li>● 值等于0时查询第一页</li></ul>
pageSize	可选	Integer	query	查询每页信息的数量，缺省值：1。
startTime	可选	String	query	查询产生时间在startTime之后的历史数据。时间格式：yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。

参数	必选/可选	类型	位置	描述
endTime	可选	String	query	查询产生时间在endTime之前的历史数据。 时间格式: yyyyMMdd'T'HHmmss'Z', 如 20151212T121212Z。

## 响应参数

QueryDeviceDataHistoryOutDTO

参数	类型	描述
totalCount	Long	查询的记录数量。
pageNo	Long	查询的页码。
pageSize	Long	查询每页信息的数量。
deviceDataHistoryDTOS	List<DeviceDataHistoryDTO>	设备历史数据列表，具体参见 <a href="#">DeviceDataHistoryDTO结构体</a> 。

DeviceDataHistoryDTO结构体说明:

参数	类型	描述
serviceId	String(256)	设备的服务标识。
deviceId	String(256)	设备ID，用于唯一标识一个设备。
gatewayId	String(256)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	String(256)	第三方应用的身份标识，用于唯一标识一个应用。
data	JsonObject	设备上报的数据。
timestamp	String(256)	上报数据的时间戳，时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。

## 错误码

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	<p>设备信息不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100216	The application input is invalid.	<p>应用输入无效。</p> <p>处理建议：请参考请求参数说明检查接口请求中的参数是否有误。例如，pageSize是否超过2000。</p>
400	100419	The deviceId and gatewayId can't be both null.	<p>deviceId和gatewayId不能同时为空。</p> <p>处理建议：请检查请求参数中deviceId和gatewayId是否填写。</p>
403	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	<p>应用未被授权。</p> <p>处理建议：若非应用授权场景，请确认请求参数中的appId为空。</p>
403	1010004	app throttle exceed.	<p>应用调用接口过于频繁，超过流控值（默认值是100次/60s）。</p> <p>处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。</p>
403	1010005	App_key or access_token is invalid.	<p>错误的token信息。</p> <p>处理建议：请检查接口请求中所携带的accessToken是否有误。</p>

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId 是否有误。</li><li>● 请检查请求路径（url）中传入的 appId 是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.8.4 查询设备影子历史数据

#### 典型场景

第三方应用通过“修改设备影子”接口修改设备影子的配置时，物联网平台会保存修改记录。当第三方应用需要查看设备影子的历史配置记录时，可调用此接口查询获取。

#### 接口功能

支持第三方应用根据设备ID，在物联网平台查询指定设备影子的历史配置数据。

#### 接口描述

```
QueryDeviceDesiredHistoryOutDTO queryDeviceDesiredHistory(QueryDeviceDesiredHistoryInDTO  
qddhInDTO, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qddhInDTO	必选	QueryDeviceDesiredHistoryInDTO	query	见下表 QueryDeviceDesiredHistoryInDTO 的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

QueryDeviceDesiredHistoryInDTO

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
deviceId	必选	String	query	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	query	网关ID，用于标识一个网关设备。
serviceId	可选	String	query	设备的服务标识。
property	可选	String	query	服务属性数据。
pageNo	可选	Integer	query	查询的页码。 ● 值为空时查询内容不分页 ● 值大于等于0的整数时分页查询 ● 值等于0时查询第一页
pageSize	可选	Integer	query	查询每页信息的数量，缺省值：1。
startTime	可选	String	query	查询产生时间在 startTime 之后的历史数据。时间格式： yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。
endTime	可选	String	query	查询产生时间在 endTime 之前的历史数据。时间格式： yyyyMMdd'T'HHmmss'Z'，如 20151212T121212Z。

## 响应参数

QueryDeviceDesiredHistoryOutDTO

参数	类型	描述
totalCount	Long	查询的记录数量。
pageNo	Long	查询的页码。
pageSize	Long	查询每页信息的数量。
DeviceDesiredHistoryDTO	List<DeviceDesiredHistoryDTO>	设备影子历史配置数据列表，具体参见 <a href="#">DeviceDesiredHistoryDTO结构体</a> 。

DeviceDesiredHistoryDTO结构体说明：

参数	类型	描述
serviceId	String(256)	设备的服务标识。

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
gatewayId	String(256)	网关ID，用于标识一个网关设备。当设备是直连设备时，gatewayId与设备的deviceId一致。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	String(256)	第三方应用的身份标识，用于唯一标识一个应用。
desired	JsonObject	要向设备下发的配置信息。
timestamp	String(256)	数据配置的时间戳，时间格式：yyyyMMdd'THHmmss'Z'，如 20151212T121212Z。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。
400	100419	The deviceId and gatewayId can't be both null.	deviceId和gatewayId不能同时为空。 处理建议：请检查请求参数中 deviceId和gatewayId是否填写。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.8.5 查询设备服务能力

#### 典型场景

第三方应用若需要了解设备可上报哪些服务属性数据，以及设备支持下发哪些命令，可调用此接口在物联网平台查询设备的Profile文件中定义的设备服务能力信息。

#### 接口功能

支持第三方应用在物联网平台查询设备的服务属性、设备命令等服务能力信息。

#### 接口描述

```
QueryDeviceCapabilitiesOutDTO queryDeviceCapabilities(QueryDeviceCapabilitiesInDTO qdcInDTO,  
String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qdcInDTO	必选	QueryDeviceCapabilitiesInDTO	query	见下表QueryDeviceCapabilitiesInDTO的描述。

参数	必选/ 可选	类型	位置	描述
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

## QueryDeviceCapabilitiesInDTO

参数	是否必 须	类型	位置	描述
gatewayId	可选	String	query	网关ID，用于标识一个网关设备。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的 appId。
deviceId	可选	String	query	设备ID，用于唯一标识一个设备。

## 响应参数

## QueryDeviceCapabilitiesOutDTO

参数	类型	描述
deviceCapabilit ies	List<DeviceCapabil ityDTO>	查询结果列表，具体参见 <a href="#">DeviceCapabilityDTO结构体</a> 。

DeviceCapabilityDTO结构体说明：

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
serviceCapabi lities	List<ServiceCapabi lityDTO>	设备的服务能力列表，具体参见 <a href="#">ServiceCapabilityDTO结构体</a> 。

ServiceCapabilityDTO结构体说明：

参数	类型	描述
serviceId	String(256)	设备的服务标识。
serviceType	String(256)	设备的服务类型。

参数	类型	描述
option	String(256)	服务选项。
description	String(10240)	设备服务描述信息。
commands	List<ServiceCommand>	支持的命令名称列表，具体参见 <a href="#">ServiceCommand 结构体</a> 。
properties	List<ServiceProperty>	支持的属性名称列表，具体参见 <a href="#">ServiceProperty 结构体</a> 。

ServiceCommand结构体说明：

参数	类型	描述
commandName	String(256)	命令名称。
paras	List<ServiceCommandPara>	属性列表，具体参见 <a href="#">ServiceCommandPara 结构体</a> 。
responses	List<ServiceCommandResponse>	响应列表，具体参见 <a href="#">ServiceCommandResponse 结构体</a> 。

ServiceCommandPara结构体说明：

参数	类型	描述
paraName	String(256)	参数名称。
dataType	String(256)	数据类型。
required	Boolean	是否必选。
min	String	属性最小值。
max	String	属性最大值。
step	Double	步长。
maxLength	Integer	最大长度。
unit	String	单位（符号）。
enumList	List<String>	枚举类型列表。

ServiceCommandResponse结构体说明：

参数	类型	描述
responseName	String(256)	响应名称。
paras	List<ServiceCommandPara>	属性列表，具体参见 <a href="#">ServiceCommandPara结构体</a> 。

ServiceProperty结构体说明：

参数	类型	描述
propertyName	String(256)	属性名称。
dataType	String(256)	数据类型。
required	Boolean	是否必选。
min	String	属性最小值。
max	String	属性最大值。
step	Double	步长。
maxLength	Integer	最大长度。
method	String(256)	访问方法。 <ul style="list-style-type: none"><li>● R: 可读</li><li>● W: 可写</li><li>● E: 可观察</li></ul>
unit	String	单位（符号）。
enumList	List<String>	枚举类型列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 2.3.9 设备组管理

### 2.3.9.1 创建设备组

#### 典型场景

第三方应用可调用此接口在物联网平台上创建设备组，并把设备归类到不同的设备组内，进行分组管理。一个设备可以归属到多个设备组内。

在对设备进行某些操作时（如升级设备软固件、批量下发命令等），可通过设备组来指定要进行操作的设备。

## 接口功能

支持第三方应用在物联网平台创建设备组，用于对设备进行分组管理。

## 接口描述

```
CreateDeviceGroupOutDTO createDeviceGroup(CreateDeviceGroupInDTO cdgInDTO, String accessToken)
throws NorthApiException
```

## 参数说明

参数	必选/可选	类型	位置	描述
cdgInDTO	必选	CreateDeviceGroupInDTO	body	见下表 <a href="#">CreateDeviceGroupInDTO</a> 的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

CreateDeviceGroupInDTO

参数	必选/可选	类型	位置	描述
name	必选	String(1-50)	body	设备组名称，仅限大小写字母和数字。
description	可选	String(1024)	body	设备组的描述信息。
appId	可选	String(50)	body	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
maxDevNum	可选	Integer(>=0)	body	设备组设备最大数量，默认最小值为0。当值为0时，表示对设备数量不做限制。
deviceIds	可选	List<String>	body	添加到设备组的设备ID列表。
id	可选	String(1-50)	Body	设备组ID。

## 响应参数

CreateDeviceGroupOutDTO

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID，由平台自动生成。
appId	String(50)	第三方应用的身份标识，用于唯一标识一个应用。
maxDevNum	Integer (>=0)	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。
curDevNum	Integer	当前设备组内的设备数量。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100602	The device group name has been used.	设备组名字已经存在。 处理建议：请修改接口请求中的设备组名称。
200	100607	The devGroup has reached the limit.	设备组数目达到限制。 处理建议：请检查已创建的设备组数量是否已达到License限定的数量上限。
400	100609	Too much devices to add.	添加太多设备至设备组。 处理建议：请确认deviceIds中的设备ID数量在maxDevNum设置值的范围内。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.9.2 删除设备组

#### 典型场景

若因分组变更，第三方应用不再需要使用某个设备组，且不想在物联网平台上继续保存该设备组信息时，可调用此接口在物联网平台删除指定设备组。

#### 接口功能

支持第三方应用在物联网平台根据设备组ID删除指定设备组。

#### 接口描述

```
void deleteDeviceGroup(String devGroupId, String accessAppId, String accessToken) throws  
NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
devGroupId	必选	String	path	设备组ID，在增加设备组后由物联网平台返回获得。
accessAppId	可选	String	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId 是否有误。</li><li>● 请检查请求路径（url）中传入的 appId 是否有误。</li></ul>
200	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100603	The device group is not existed	设备组不存在。 处理建议：请检查设备组ID是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.9.3 修改设备组

#### 典型场景

若因业务变更需要修改设备组的信息，如设备组名称、设备组的设备数量限制等，第三方应用可调用此接口修改指定设备组的信息。

#### 接口功能

支持第三方应用在物联网平台修改指定设备组的信息。

#### 接口描述

```
ModifyDeviceGroupOutDTO modifyDeviceGroup(ModifyDeviceGroupInDTO mdgInDTO, String devGroupId,  
String accessAppId, String accessToken) throws NorthApiException
```

## 参数说明

参数	必选/ 可选	类型	位置	描述
mdgInDTO	必选	ModifyDeviceGroupInDTO	body	见下表 <b>ModifyDeviceGroupInDTO</b> 的描述。
devGroupId	必选	String (50)	path	设备组ID，在增加设备组后由物联网平台返回获得。
accessAppId	必选	String	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

### ModifyDeviceGroupInDTO

参数	必选/ 可选	类型	位置	描述
name	必选	String (1-50)	body	设备组名称，仅限大小写字母和数字。
description	可选	String (1024)	body	设备组的描述信息。
maxDevNum	可选	Integer (>=0)	body	设备组设备最大数量，默认值为0。当值为0时，表示对设备数量不做限制。

## 响应参数

### ModifyDeviceGroupOutDTO

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String(50)	设备组所属的应用ID。
maxDevNum	Integer (>=0)	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。

参数	类型	描述
curDevNum	Integer	当前设备组内设备数量。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100602	The device group name has been used.	设备组的名字已经存在。 处理建议：请修改接口请求中的设备组名称。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.9.4 查询设备组详情

#### 典型场景

第三方应用可调用此接口查询当前已创建的所有设备组列表信息，以了解当前设备组的分组和使用情况。

#### 接口功能

支持第三方应用在物联网平台查询已创建的所有设备组信息。

#### 接口描述

```
QueryDeviceGroupsOutDTO queryDeviceGroups(QueryDeviceGroupsInDTO qdgInDTO, String accessToken)  
throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qdgInDTO	必选	QueryDeviceGroupsInDTO	query	见下表 <b>QueryDeviceGroupsInDTO</b> 的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### QueryDeviceGroupsInDTO

参数	必选/可选	类型	位置	描述
accessAppId	可选	String	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer	query	分页查询参数。 ● 值为空时，查询内容不分页。 ● 值为大于等于0的整数时，分页查询。 ● 值为0时查询第一页。
pageSize	可选	Integer	query	每页设备组记录数量，默认值为1。
name	可选	String	query	设备组名称。

## 响应参数

QueryDeviceGroupsOutDTO

参数	类型	描述
totalCount	long	设备组总数。
pageNo	long	查询结果页码。
pageSize	long	每页设备组记录数量。
list	List<QuerySingleDeviceGroupOutDTO>	设备组信息详情，具体参见 <a href="#">QuerySingleDeviceGroupOutDTO</a> 结构体。

QuerySingleDeviceGroupOutDTO结构体说明：

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String(50)	设备组所属的应用ID。
maxDevNum	Integer(>=0)	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。
curDevNum	Integer	当前设备组内的设备数量。
creator	String(1-50)	创建设备组的用户名。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.9.5 查询指定设备组

#### 典型场景

第三方应用若需要查看某个设备组的信息，以了解该设备组的使用情况，可调用此接口查询指定设备组的信息。

#### 接口功能

支持第三方应用在物联网平台根据设备组ID查询指定设备组的信息。

#### 接口描述

```
QuerySingleDeviceGroupOutDTO querySingleDeviceGroup(String devGroupId, String accessAppId, String accessToken) throws NorthApiException
```

## 参数说明

参数	必选/ 可选	类型	位置	描述
devGrou pId	必选	String	path	设备组ID，在增加设备组后由物联网平台返 回获得。
accessA ppId	可选	String	query	如果是本应用的设备组，此参数值可以填写 null，否则填写授权应用的appId。
accessT oken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值 可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

## 响应参数

QuerySingleDeviceGroupOutDTO

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String (50)	设备组所属的应用ID。
maxDevNum	Integer (>=0)	设备组设备最大数量限制。
curDevNum	Integer	当前设备组内的设备数量。
creator	String(1-50)	创建设备组的用户名称。

## 错误码

Http状态 码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的 appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.9.6 查询指定设备组成员

#### 典型场景

第三方应用若需要查看某个设备组中的设备成员分布情况，可调用此接口查询指定设备组的设备成员列表信息。

#### 接口功能

支持第三方应用在物联网平台根据设备组ID查询指定设备组内设备列表信息。

#### 接口描述

```
QueryDeviceGroupMembersOutDTO queryDeviceGroupMembers(QueryDeviceGroupMembersInDTO qdgmInDTO,  
String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qdgmInDTO	必选	QueryDeviceGroupMembersInDTO	query	见下表 <b>QueryDeviceGroupMembersInDTO</b> 的描述。
accessToken	必选	String	header	如果调用了 <b>定时刷新token</b> 接口，此参数值可以填写null，否则要填写 <b>鉴权</b> 接口获取的accessToken。

## QueryDeviceGroupMembersInDTO

参数	必选/可选	类型	位置	描述
devGrou pId	必选	String	query	设备组ID，在增加设备组后由物联网平台返回获得。
accessAp pId	可选	String	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
pageNo	可选	Integer	query	分页查询参数，默认值为0。 ● 值为空时，查询内容不分页。 ● 值为大于等于0的整数时，分页查询。 ● 值为0时查询第一页。
pageSize	可选	Integer(1 000)	query	每页设备记录数量，默认值为10。

## 响应参数

## QueryDeviceGroupMembersOutDTO

参数	类型	描述
totalCount	long	设备组内设备总数。
pageNo	long	查询结果页码。
pageSize	long	每页设备记录数量。
deviceIds	List<String>	设备组内设备ID列表。

## 错误码

Http状 态码	错误码	错误描述	说明
400	107001	The serviceId is not exist.	服务ID不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107002	The properties is empty in database.	设备属性不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107003	The request properties is unknown.	设备状态为未知。 处理建议：请检查设备与平台的连接是否正常。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁, 超过流控值（默认值是100次/60s）。 处理建议: 联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议: 请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议: 物联网平台内部错误, 请联系物联网平台维护人员处理。

### 2.3.9.7 增加设备组成员

#### 典型场景

若需要把新增的设备或已有设备增加到某个设备组中, 第三方应用可调用此接口向指定设备组添加设备成员。在向设备组添加设备前, 建议通过“查询指定设备组”接口查询该设备组的当前设备数量及最大设备数量限制, 确保设备组中还有足够的可添加成员数量。

#### 接口功能

支持第三方应用在物联网平台向指定设备组添加设备。

## 接口描述

```
DeviceGroupWithDeviceListDTO addDevicesToGroup(DeviceGroupWithDeviceListDTO dgwdlDTO, String accessAppId, String accessToken) throws NorthApiException
```

## 参数说明

参数	必选/可选	类型	位置	描述
dgwdlDTO	必选	DeviceGroupWithDeviceListDTO	body	见下表 DeviceGroupWithDeviceListDTO 的描述。
accessAppId	可选	String	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

DeviceGroupWithDeviceListDTO

参数	必选/可选	类型	位置	描述
devGroupId	必选	String(1-50)	body	设备组ID，在增加设备组后由物联网平台返回获得。
deviceIds	必选	List<String>(1000)	body	要添加到设备组的设备ID列表。

## 返回参数

DeviceGroupWithDeviceListDTO

参数	类型	描述
devGroupId	String(1-50)	设备组ID。
deviceIds	List<String>	添加到设备组的设备ID列表。

## 错误码

Http状态码	错误码	错误描述	说明
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议： <ul style="list-style-type: none"><li>● 请检查接口请求中的设备ID是否有误。</li><li>● 请检查设备组是否达到最大设备数量限制。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 2.3.9.8 删除设备组成员

#### 典型场景

若设备组中的一个或多个设备不再归属于该设备组，第三方应用可调用此接口从设备组删除设备成员。

#### 接口功能

支持第三方应用在物联网平台从指定设备组删除设备。

#### 接口描述

```
void deleteDevicesFromGroup(DeviceGroupWithDeviceListDTO dgwdlDTO, String accessAppId, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
dgwdlDTO	必选	DeviceGroupWithDeviceListDTO	body	见下表 DeviceGroupWithDeviceListDTO 的描述。
accessAppId	可选	String	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

## DeviceGroupWithDeviceListDTO

参数	必选/可选	类型	位置	描述
devGroupId	必选	String(1-50)	body	设备组ID，在增加设备组后由物联网平台返回获得。
deviceIds	必选	List<String>(1000)	body	要从设备组删除的设备ID列表。

## 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议： <ul style="list-style-type: none"><li>● 请检查接口请求中的设备ID是否有误。</li><li>● 请检查设备组是否达到最大设备数量限制。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 2.3.10 设备升级

### 2.3.10.1 查询版本包列表

#### 典型场景

在进行设备版本升级前，第三方应用可调用此接口查询已经上传到物联网平台的版本升级包列表信息，确保设备要升级的版本包已上传到平台。

#### 接口功能

支持第三方应用在物联网平台查询已上传的版本包列表信息，可根据条件查询满足要求的版本包。

#### 接口描述

```
QueryUpgradePackageListOutDTO queryUpgradePackageList(QueryUpgradePackageListInDTO quplInDTO,  
String accessToken) throws NorthApiException
```

## 参数说明

参数	必选/可选	类型	位置	描述
quplInDTO	必选	QueryUpgradePackageListInDTO	query	见下表 QueryUpgradePackageListInDTO 的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的 accessToken。

### QueryUpgradePackageListInDTO

参数	必选/可选	类型	位置	描述
fileType	可选	String(256)	query	版本包类型。 ● firmwarePackage: 固件包 ● softwarePackage: 软件包
deviceType	可选	String(256)	query	版本包适用的设备类型。
model	可选	String(256)	query	版本包适用的设备型号。
manufacturerName	可选	String(256)	query	版本包适用的设备厂商名称。
version	可选	String(256)	query	版本包的版本号。
pageNo	可选	Integer	query	分页查询参数，默认值0。 ● 值为空时，查询内容不分页。 ● 值为大于等于0的整数时，分页查询。 ● 值为0时查询第一页。
pageSize	可选	Integer	query	查询结果分页时的每页结果数量，取值范围1-100，默认值10。

## 响应参数

### QueryUpgradePackageListOutDTO

参数	类型	描述
data	List<QueryUpgradePackageOutDTO>	版本包列表信息，具体参见 <a href="#">QueryUpgradePackageOutDTO</a> 结构体。
pageNo	Integer	查询结果的页码。
pageSize	Integer	查询结果每页的记录数量。
totalCount	Integer	查询结果的总记录数。

QueryUpgradePackageOutDTO结构体说明：

参数	类型	描述
fileId	String	版本包ID。
name	String	版本包名称。
version	String	版本包版本号。
fileType	String	版本包文件类型。 ● firmwarePackage： 固件包 ● softwarePackage： 软件包
deviceType	String	版本包适用的设备类型。
model	String	版本包适用的设备型号。
manufacturerName	String	版本包适用的设备厂商名称。
protocolType	String	版本包适用的设备协议类型
description	String	版本包的描述信息。
date	String	版本包的生成时间。
uploadTime	String	版本包的上传日期。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中pageNo和pageSize的值是否在正常范围内。

Http状态码	错误码	错误描述	说明
400	123029	pageNo or pageSize beyond the limit.	pageNo或pageSize超出上限。 处理建议：修改pageNo或pageSize为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 2.3.10.2 查询指定版本包

#### 典型场景

在进行设备版本升级前，第三方应用可调用此接口在物联网平台查询某个版本升级包的信息，确保设备要升级的版本包已上传到平台。

#### 接口功能

支持第三方应用在物联网平台根据版本包ID查询指定版本包信息，在调用该接口前需要通过“查询版本包列表”接口查询获得版本包ID。

#### 接口描述

```
QueryUpgradePackageOutDTO queryUpgradePackage(String fileId, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
fileId	必选	String	path	版本包ID，在上传版本包后获得。
access Token	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

```
QueryUpgradePackageOutDTO
```

参数	类型	描述
fileId	String	版本包ID。
name	String	版本包名称。
version	String	版本包的版本号。
fileType	String	版本包类型。 ● firmwarePackage: 固件包 ● softwarePackage: 软件包
deviceType	String	版本包适用的设备类型。
model	String	版本包适用的设备型号。
manufacturerName	String	版本包适用的设备厂商名称。
protocolType	String	版本包适用的设备协议类型。
description	String	版本包的描述信息。
date	String	版本包的生成时间。
uploadTime	String	版本包的上传日期。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	123002	Device or package file not found.	设备或包不存在。 处理建议：请检查接口请求中的 fileId 是否正确。

### 2.3.10.3 删除指定版本包

#### 典型场景

对于不需要继续使用和保留的设备版本包，第三方应用可调用此接口在物联网平台上删除指定的版本包。

#### 接口功能

支持第三方应用在物联网平台根据版本包ID删除指定的版本包文件，在调用该接口前需要通过“查询版本包列表”接口查询获得要删除

#### 接口描述

```
void deleteUpgradePackage(String fileId, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
fileId	必选	String	path	版本包ID，在上传版本包后获得。
access Token	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

void

#### 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。

Http状态码	错误码	错误描述	说明
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 2.3.10.4 创建软件升级任务

#### 典型场景

若需要对设备进行软件版本升级，第三方应用可调用此接口为多个设备创建软件升级任务，升级前请确保目标版本包已上传到物联网平台。当前仅支持对NB-IoT设备进行软件版本升级。

#### 接口功能

支持第三方应用在物联网平台为多个设备进行软件版本升级，当前仅支持对NB-IoT设备进行软件版本升级。

#### 接口描述

```
CreateUpgradeTaskOutDTO createSoftwareUpgradeTask(CreateUpgradeTaskInDTO cutInDTO, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
cutInDTO	必选	CreateUpgradeTaskInDTO	body	见下表CreateUpgradeTaskInDTO的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### CreateUpgradeTaskInDTO

参数	必选/可选	类型	位置	描述
fileId	必选	String	body	要升级的目标版本包ID。
targets	必选	OperateDevices	body	要进行升级的目标，具体参见 <a href="#">OperateDevices结构体</a> 。

参数	必选/可选	类型	位置	描述
policy	可选	OperatePolicy	body	升级任务的执行策略, 具体参见 <a href="#">OperatePolicy结构体</a> 。

OperateDevices结构体说明:

参数	必选/可选	类型	位置	描述
deviceGroups	可选	List<String>	body	设备组名称列表, 最多支持256个设备组。 与指定设备列表方式二选一。
deviceType	可选	String	body	设备类型。 指定设备组时, 此字段必选。
model	可选	String	body	设备型号。 指定设备组时, 此字段必选。
manufacturerName	可选	String	body	设备厂商名称。 指定设备组时, 此字段必选。
devices	可选	List<String>	body	设备ID列表, 最多支持256个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明:

参数	必选/可选	类型	位置	描述
executeType	必选	String	body	执行类型, 默认值为now。 <ul style="list-style-type: none"> <li>● now: 现在执行</li> <li>● device_online: 设备上线时执行</li> <li>● custom: 自定义</li> </ul>
startTime	可选	String	body	任务执行时间, executeType为custom时必选, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
endTime	可选	String	body	任务停止时间, executeType为custom时必选, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。

参数	必选/可选	类型	位置	描述
retryType	可选	Boolean	body	执行失败是否进行重试， 默认不重试。 ● true: 重试 ● false: 不重试
retryTimes	可选	Integer	body	重试次数, 取值范围1-5, retryType为true时必选。

## 响应参数

CreateUpgradeTaskOutDTO

参数	类型	描述
operationId	String	操作任务ID。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查请求参数的是否有误。
400	123016	The parameter is error, targetversion not match with device.	参数不正确，目标版本与指定设备不匹配。 处理建议：请检查接口请求中deviceType、 manufacturerName、 model，与参数fileId指定的目标版本包信息是否一致。
400	123019	manufacturerName is null.	厂商名称内容为空。 处理建议：请检查接口请求中的manufacturerName是否为空。
400	123020	deviceType is null	设备类型内容为空。 处理建议：请检查接口请求中的deviceType是否为空。
400	123021	model is null.	设备型号内容为空。 处理建议：请检查接口请求中的model是否为空。

Http状态码	错误码	错误描述	说明
400	123022	deviceGroups and devices cannot be null together	deviceGroups和devices不能同时为空。 处理建议：deviceGroups与devices必须要选择一个进行赋值。
400	123023	deviceGroups and devices cannot be exist together	deviceGroups和devices不能同时存在。 处理建议：deviceGroups与devices只能选择一个进行赋值。
400	123024	The number of deviceGroups or devices reached upper limit	deviceGroups或devices内容数量达到上限。 处理建议：请检查deviceGroups或devices的内容，数量不能超过256个。
400	123025	executeType is error or can not to be null.	executeType有误或不能为空。 处理建议：请检查接口请求中的executeType是否为空或者是否有误。
400	123026	startTime or endTime is null or error.	startTime或endTime为空或者有误。 处理建议：请检查接口请求中的startTime和endTime是否为空或者格式有误。
400	123028	retryTimes is null or beyond the limit.	retryTimes为空或超出上限。 处理建议：请检查接口请求中的retryTimes是否为空，或者不能小于1或者大于5。
400	123032	startTime can not be later than the endTime.	startTime的时间不能比endTime晚。 处理建议：请检查接口请求中的startTime是否晚于endTime。
400	123033	startTime can not be earlier than the now.	startTime的时间不能比当前时间早。 处理建议：请检查接口请求中的startTime是否早于当前时间。
400	123034	endtime must be greater than 5 minutes.	endtime要比startTime大5分钟。 处理建议：请检查接口请求中的startTime和endTime间隔时间必须大于5分钟。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 2.3.10.5 创建固件升级任务

#### 典型场景

若需要对设备进行固件版本升级，第三方应用可调用此接口为多个设备创建固件升级任务，升级前请确保目标版本包已上传到物联网平台。当前仅支持对NB-IoT设备进行固件版本升级。

#### 接口功能

支持第三方应用在物联网平台为多个设备进行固件版本升级，当前仅支持对NB-IoT设备进行固件版本升级。

#### 接口描述

```
CreateUpgradeTaskOutDTO createFirmwareUpgradeTask(CreateUpgradeTaskInDTO cutInDTO, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/可选	类型	位置	描述
cutInDTO	必选	CreateUpgradeTaskInDTO	body	见下表CreateUpgradeTaskInDTO的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## CreateUpgradeTaskInDTO

参数	必选/可选	类型	位置	描述
fileId	必选	String	body	要升级的目标版本包ID。
targets	必选	OperateDevices	body	要进行升级的目标，具体参见 <a href="#">OperateDevices结构体</a> 。
policy	可选	OperatePolicy	body	升级任务的执行策略，具体参见 <a href="#">OperatePolicy结构体</a> 。

## OperateDevices结构体说明：

参数	必选/可选	类型	位置	描述
deviceGroups	可选	List<String>	body	设备组名称列表，最多支持256个设备组。 与指定设备列表方式二选一。
deviceType	可选	String	body	设备类型。 指定设备组时，此字段必选。
model	可选	String	body	设备型号。 指定设备组时，此字段必选。
manufacturerName	可选	String	body	设备厂商名称。 指定设备组时，此字段必选。
devices	可选	List<String>	body	设备ID列表，最多支持256个设备。 与指定设备组方式二选一。

## OperatePolicy结构体说明：

参数	必选/可选	类型	位置	描述
executeType	必选	String	body	执行类型，默认值为now。 <ul style="list-style-type: none"><li>● now：现在执行</li><li>● device_online：设备上线时执行</li><li>● custom：自定义</li></ul>

参数	必选/可选	类型	位置	描述
startTime	可选	String	body	任务执行时间, executeType为custom时必选, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
endTime	可选	String	body	任务停止时间, executeType为custom时必选, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
retryType	可选	Boolean	body	执行失败是否进行重试, 默认不重试。 ● true: 重试 ● false: 不重试
retryTimes	可选	Integer	body	重试次数, 取值范围1-5, retryType为true时必选。

## 响应参数

CreateUpgradeTaskOutDTO

参数	类型	描述
operationId	String	操作任务ID。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议: 请检查请求参数的是否有误。
400	123016	The parameter is error, targetversion not match with device.	参数不正确, 目标版本与指定设备不匹配。 处理建议: 请检查接口请求中deviceType、manufacturerName、model, 与参数fileId指定的目标版本包信息是否一致。

Http状态码	错误码	错误描述	说明
400	123019	manufacturerName is null.	厂商名称内容为空。 处理建议：请检查接口请求中的manufacturerName是否为空。
400	123020	deviceType is null	设备类型内容为空。 处理建议：请检查接口请求中的deviceType是否为空。
400	123021	model is null.	设备型号内容为空。 处理建议：请检查接口请求中的model是否为空。
400	123022	deviceGroups and devices cannot be null together	deviceGroups和devices不能同时为空。 处理建议：deviceGroups与devices必须要选择一个进行赋值。
400	123023	deviceGroups and devices cannot be exist together	deviceGroups和devices不能同时存在。 处理建议：deviceGroups与devices只能选择一个进行赋值。
400	123024	The number of deviceGroups or devices reached upper limit	deviceGroups或devices内容数量达到上限。 处理建议：请检查deviceGroups或devices的内容，数量不能超过256个。
400	123025	executeType is error or can not to be null.	executeType有误或不能为空。 处理建议：请检查接口请求中的executeType是否为空或者是是否有误。
400	123026	startTime or endTime is null or error.	startTime或endTime为空或者有误。 处理建议：请检查接口请求中的startTime和endTime是否为空或者格式有误。
400	123028	retryTimes is null or beyond the limit.	retryTimes为空或超出上限。 处理建议：请检查接口请求中的retryTimes是否为空，或者不能小于1或者大于5。

Http状态码	错误码	错误描述	说明
400	123032	startTime can not be later than the endTime.	startTime的时间不能比endTime晚。 处理建议：请检查接口请求中的startTime是否晚于endTime。
400	123033	startTime can not be earlier than the now.	startTime的时间不能比当前时间早。 处理建议：请检查接口请求中的startTime是否早于当前时间。
400	123034	endtime must be greater than 5 minutes.	endtime要比startTime大5分钟。 处理建议：请检查接口请求中的startTime和endTime间隔时间必须大于5分钟。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 2.3.10.6 查询指定升级任务结果

#### 典型场景

在创建了设备软件升级任务或固件升级任务后，第三方应用可调用此接口查询某个升级任务详细信息，以查看升级任务的配置信息和执行情况等。

#### 接口功能

支持第三方应用在物联网平台查询单个固件或软件升级任务的详细信息，包括升级任务的配置信息和执行情况等。

## 接口描述

```
QueryUpgradeTaskOutDTO queryUpgradeTask(String operationId, String accessToken) throws  
NorthApiException
```

## 参数说明

参数	必选/可选	类型	位置	描述
operationId	必选	String	path	操作任务ID，在创建操作任务后由物联网平台返回获得。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

QueryUpgradeTaskOutDTO

参数	类型	描述
operationId	String	操作任务ID。
createTime	String	操作任务的创建时间。
startTime	String	操作任务的启动时间。
stopTime	String	操作任务的停止时间。
operateType	String	操作类型。 <ul style="list-style-type: none"><li>● firmware_upgrade</li><li>● software_upgrade</li></ul>
targets	OperateDevices	执行操作的目标设备，具体参见 <a href="#">OperateDevices结构体</a> 。
policy	OperatePolicy	操作执行策略，具体参见 <a href="#">OperatePolicy结构体</a> 。
status	String	操作任务的状态。 <ul style="list-style-type: none"><li>● wait: 等待</li><li>● processing: 正在执行</li><li>● failed: 失败</li><li>● success: 成功</li><li>● stop: 停止</li></ul>
staResult	OperationStaResult	操作结果统计，具体参见 <a href="#">OperationStaResult结构体</a> 。

参数	类型	描述
extendPara	JsonString	操作扩展参数，视不同类型的操作不同，具体参见 <a href="#">extendPara请求参数</a> 。

OperateDevices结构体说明：

参数	类型	描述
deviceGroups	List<String>	设备组名称列表，最多支持256个设备组。 与指定设备列表方式二选一。
deviceType	String	设备类型。 指定设备组时，此字段必选。
model	String	设备型号。 指定设备组时，此字段必选。
manufacturerName	String	设备厂商名称。 指定设备组时，此字段必选。
devices	List<String>	设备ID列表，最多支持256个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明：

参数	类型	描述
executeType	String	执行类型，默认值为now。 <ul style="list-style-type: none"><li>● now: 现在执行</li><li>● device_online: 设备上线时执行</li><li>● custom: 自定义</li></ul>
startTime	String	任务执行时间，executeType=custom时必选，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
endTime	String	任务停止时间，executeType=custom时必选，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
retryType	Boolean	执行失败是否进行重试，默认不重试。 <ul style="list-style-type: none"><li>● true: 重试</li><li>● false: 不重试</li></ul>
retryTimes	Integer	重试次数，取值范围1-5，retryType=true时必选。

OperationStaResult结构体说明：

参数	类型	描述
total	Integer(64)	操作设备总数。
wait	Integer(64)	等待操作的设备个数。
processing	Integer(64)	正在操作的设备个数。
success	Integer(64)	操作成功的设备个数。
fail	Integer(64)	操作失败的设备个数。
stop	Integer(64)	停止操作的设备个数。
timeout	Integer(64)	操作超时失败的设备个数。

对于操作类型为softwareUpgrade和firmwareUpgrade，extendPara请求参数如下：

参数	类型	描述
fileVersion	String	升级的目标版本号。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的operationId内容是否正确。

### 2.3.10.7 查询指定升级任务子任务详情

#### 典型场景

在创建了设备软件升级任务或固件升级任务后，任务中涉及的每个设备的升级是一个子任务（即升级任务中涉及多少个设备，就有多少个子任务）。第三方应用可调用此接口查询某个升级任务中各个子任务的详细信息，以查看子任务的具体执行情况。

#### 接口功能

支持第三方应用在物联网平台查询单个固件或软件升级任务中每个设备的升级执行情况。

#### 接口描述

```
QueryUpgradeSubTaskOutDTO queryUpgradeSubTask(QueryUpgradeSubTaskInDTO qustInDTO, String  
operationId, String accessToken) throws NorthApiException
```

#### 参数说明

参数	必选/ 可选	类型	位置	描述
qustInDTO	必选	QueryUpgradeSubTaskInDTO	query	见下表 QueryUpgradeSubTaskInDTO的描述。
operationId	必选	String	path	操作任务ID，在创建操作任务后由物联网平台返回获得。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### QueryUpgradeSubTaskInDTO

参数	必选/ 可选	类型	位置	描述
subOperationStatus	可选	String	query	子任务状态，不指定，则查询该任务下所有子任务执行详情。 <ul style="list-style-type: none"><li>● wait: 等待</li><li>● processing: 正在执行</li><li>● fail: 失败</li><li>● success: 成功</li><li>● stop: 停止</li></ul>

参数	必选/ 可选	类型	位置	描述
pageNo	可选	Integer	query	分页查询参数， 默认值0。 <ul style="list-style-type: none"> <li>● 值为空时， 查询内容不分页。</li> <li>● 值为大于等于0的整数时， 分页查询。</li> <li>● 值为0时查询第一页。</li> </ul>
pageSize	可选	Integer	query	查询结果分页时的每页结果数量， 取值范围1-100， 默认值10。

## 响应参数

QueryUpgradeSubTaskOutDTO

参数	类型	描述
data	list<SubOperationInfo>	子任务列表信息， 具体参见 <a href="#">SubOperationInfo结构体</a> 。
pageNo	long	查询结果的页码。
pageSize	long	查询结果每页的记录数量。
totalCount	long	查询结果的记录总数。

SubOperationInfo结构体说明：

参数	类型	描述
subOperationId	String	子任务ID。
createTime	String	子任务创建时间。
startTime	String	子任务启动时间。
stopTime	String	子任务停止时间。
operateType	String	操作类型。 <ul style="list-style-type: none"> <li>● firmware_upgrade</li> <li>● software_upgrade</li> </ul>
deviceId	String	操作设备的设备ID。

参数	类型	描述
status	String	子任务状态。 ● wait: 等待 ● processing: 正在执行 ● fail: 失败 ● success: 成功 ● stop: 停止
detailInfo	String	任务状态的详细描述，对于失败场景下为失败原因。
extendInfo	JsonString	任务扩展信息，视不同类型的操作不同。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
400	123029	pageNo or pageSize beyond the limit.	pageNo 或 pageSize 超出上限。 处理建议：修改 pageNo 或 pageSize 为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的 token 信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的 operationId 内容是否正确。

### 2.3.10.8 查询升级任务列表

#### 典型场景

第三方应用可调用此接口查询已创建的升级任务列表信息，以了解当前已有的升级任务信息及各个任务的执行情况。

## 接口功能

支持第三方应用在物联网平台查询升级任务的列表信息，可根据条件查询满足要求的升级任务信息。

## 接口描述

```
QueryUpgradeTaskListOutDTO queryUpgradeTaskList(QueryUpgradeTaskListInDTO qutlInDTO, String accessToken) throws NorthApiException
```

## 参数说明

参数	必选/可选	类型	位置	描述
qutlInDTO	必选	QueryUpgradeTaskListInDTO	query	见下表 QueryUpgradeTaskListInDTO的描述。
accessToken	必选	String	header	如果调用了 <a href="#">定时刷新token</a> 接口，此参数值可以填写null，否则要填写 <a href="#">鉴权</a> 接口获取的accessToken。

### QueryUpgradeTaskListInDTO

参数	必选/可选	类型	位置	描述
operationType	可选	String(256)	query	操作类型。 <ul style="list-style-type: none"><li>● firmware_upgrade</li><li>● software_upgrade</li></ul>
operationStatus	可选	String(256)	query	操作任务的状态。 <ul style="list-style-type: none"><li>● wait: 等待</li><li>● processing: 正在执行</li><li>● failed: 失败</li><li>● success: 成功</li><li>● stop: 停止</li></ul>
deviceType	可选	String(256)	query	操作任务针对的设备类型。
model	可选	String(256)	query	操作任务针对的设备型号。
manufacturerName	可选	String(256)	query	操作任务针对的设备厂家名称。
deviceId	可选	String(256)	query	操作任务针对的设备ID。

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer	query	分页查询参数，默认值0。 <ul style="list-style-type: none"> <li>● 值为空时，查询内容不分页。</li> <li>● 值为大于等于0的整数时，分页查询。</li> <li>● 值为0时查询第一页。</li> </ul>
pageSize	可选	Integer	query	查询结果分页时的每页结果数量，取值范围1-100，默认值10。

## 响应参数

QueryUpgradeTaskListOutDTO

参数	类型	描述
data	List<OperationInfo>	任务列表信息，具体参见 <a href="#">OperationInfo结构体</a> 。
pageNo	Integer	查询结果的页码。
pageSize	Integer	查询结果每页的记录数量。
totalCount	Integer	查询结果的总记录数。

OperationInfo结构体说明：

参数	类型	描述
operationId	String	操作任务ID。
createTime	String	操作任务的创建时间。
startTime	String	操作任务的启动时间。
stopTime	String	操作任务的停止时间。
operateType	String	操作类型。 <ul style="list-style-type: none"> <li>● firmware_upgrade</li> <li>● software_upgrade</li> </ul>
targets	OperateDevices	执行操作的目标设备，具体参见 <a href="#">OperateDevices结构体</a> 。
policy	OperatePolicy	操作执行策略，具体参见 <a href="#">OperatePolicy结构体</a> 。

参数	类型	描述
status	String	操作任务的状态。 <ul style="list-style-type: none"> <li>● wait: 等待</li> <li>● processing: 正在执行</li> <li>● failed: 失败</li> <li>● success: 成功</li> <li>● stop: 停止</li> </ul>
staResult	OperationStaResult	操作结果统计, 具体参见 <a href="#">OperationStaResult 结构体</a> 。
extendPara	JsonString	操作扩展参数, 视不同类型的操作不同。

OperateDevices结构体说明:

参数	类型	描述
deviceGroups	List<String>	设备组名称列表, 最多支持256个设备组。 与指定设备列表方式二选一。
deviceType	String	设备类型。 指定设备组时, 此字段必选。
model	String	设备型号。 指定设备组时, 此字段必选。
manufacturerName	String	设备厂商名称。 指定设备组时, 此字段必选。
devices	List<String>	设备ID列表, 最多支持256个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明:

参数	类型	描述
executeType	String	执行类型, 默认值为now。 <ul style="list-style-type: none"> <li>● now: 现在执行</li> <li>● device_online: 设备上线时执行</li> <li>● custom: 自定义</li> </ul>
startTime	String	任务执行时间, executeType=custom时必选, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。

参数	类型	描述
endTime	String	任务停止时间, executeType=custom时必选, 时间格式: yyyyMMdd'T'HHmmss'Z', 如: 20151212T121212Z。
retryType	Boolean	执行失败是否进行重试, 默认不重试。 ● true: 重试 ● false: 不重试
retryTimes	Integer	重试次数, 取值范围1-5, retryType=true时必选。

OperationStaResult结构体说明:

参数	类型	描述
total	Integer(64)	操作设备总数。
wait	Integer(64)	等待操作的设备个数。
processing	Integer(64)	正在操作的设备个数。
success	Integer(64)	操作成功的设备个数。
fail	Integer(64)	操作失败的设备个数。
stop	Integer(64)	停止操作的设备个数。
timeout	Integer(64)	操作超时失败的设备个数。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议: 请检查接口请求中 fileId 的格式是否正确。
400	123029	pageNo or pageSize beyond the limit.	pageNo或pageSize超出上限。 处理建议: 修改pageNo或pageSize为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁, 超过流控值(默认值是100次/60s)。 处理建议: 联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的operationId内容是否正确。

# 3 IoT 平台北向 PHP SDK API 参考

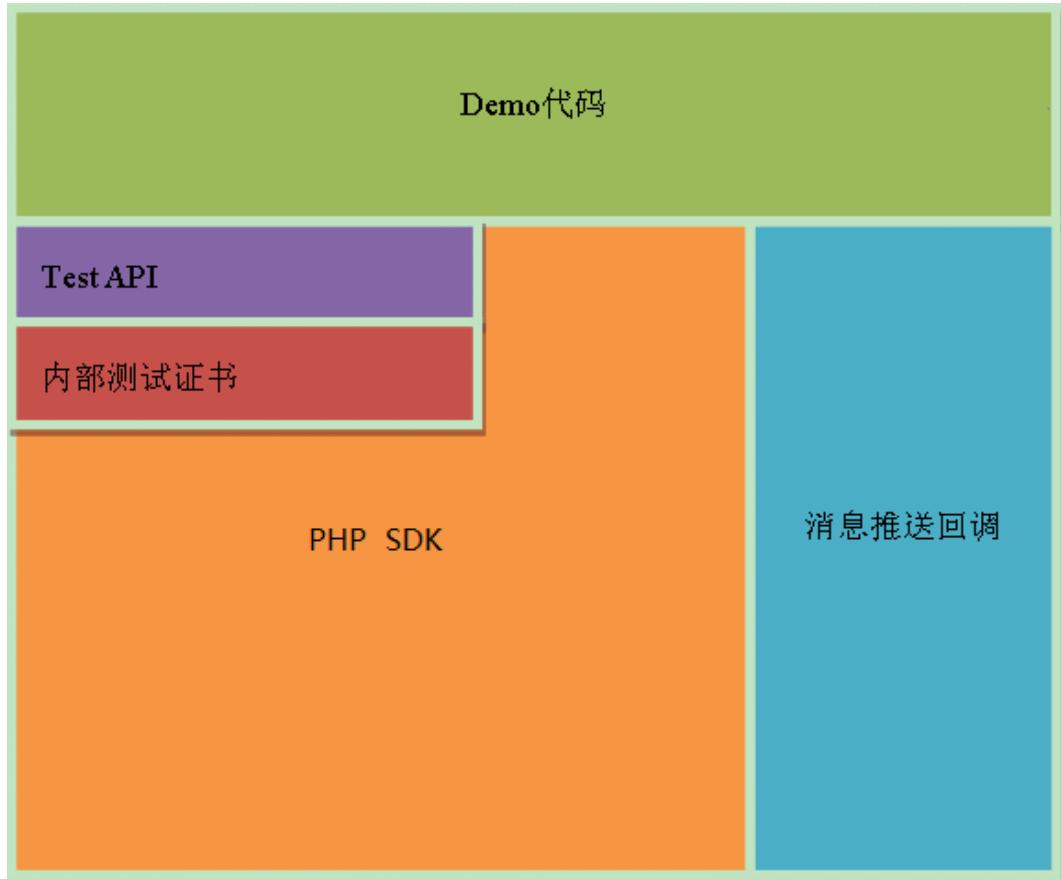
[SDK Demo 总体架构及使用说明](#)

[SDK 初始化配置及测试](#)

[业务接口列表](#)

## 3.1 SDK Demo 总体架构及使用说明

- Demo代码是调用SDK接口的样例代码，包括初始化及各个接口调用，仅供开发者参考。
- SDK提供PHP方法调用平台北向Restful接口与平台通信。
- 消息推送回调使用PHP代码实现回调接口，供平台推送消息给应用服务器时调用，应用通过继承**PushMessageReceiver**类并重写类中的方法接收推送消息的内容。
- Test API提供测试SDK接口的功能，主要测试SDK与平台之间的基本功能是否可用，并输出测试结果。测试时会分别使用内部测试证书和开发者设置的证书，以便排查证书问题。



## 3.2 SDK 初始化配置及测试

### 3.2.1 NorthApiClient 类方法

用于创建应用实例，是调用SDK其他接口的前提。主要方法如下：

方法	说明
public function __set(\$name, \$value)	魔术方法，初始化 <b>NorthApiClient</b> 中的属性，例如： \$northClient->\$clientInfo(\$clientInfo)。
public function initSSLConfig0()	初始化双向认证的配置，调用后才可以使用其他方法。 <b>注意</b> 本方法使用了测试证书，非正式证书，且不做主机名校验，故仅用于集成对接的调试阶段。
public function initSSLConfig1(\$sslConfig)	初始化双向认证的配置，调用后才可以使用其他方法。SSLConfig的定义参考 <a href="#">2.4 SSLConfig类</a> 。 <b>注意</b> 本方法用于导入证书，可以用于商用与正式使用阶段。
public function getVersion()	查询SDK版本号

### 3.2.2 Clientinfo 类方法

用于设置对接的基本信息。主要方法（省略get方法）如下：

方法	说明
public function __set(\$name, \$value)	设置平台的IP、端口、appId、secret，例如： \$clientInfo->appId = ‘zxcxxxxxxxx12’。

### 3.2.3 NorthApiException 类方法

当SDK处理或请求被平台处理有异常时，抛出**NorthApiException**类对象。主要方法（省略set方法）如下：

方法	说明
public function __get(\$name)	获取异常的各个属性信息，例如，获取异常错误码： \$e->error_code。

### 3.2.4 SSLConfig 类

用于设置证书路径及密码。主要方法（省略get方法）如下：

方法	说明
public function __set(\$name, \$value)	设置客户端的证书位置的绝对路径，例如： \$sslConfig->selfCertPath = ‘./client.pem’。

## 3.3 业务接口列表

### 3.3.1 应用安全接入

第三方应用获取鉴权信息，接入物联网平台，随后携带鉴权信息调用其他API接口。

#### 3.3.1.1 鉴权

##### 典型场景

第三方应用首次访问物联网平台的开放API时，需调用此接口完成接入认证；第三方应用在物联网平台的认证过期后，需调用此接口重新进行认证，才能继续访问物联网平台的开放API。

##### 接口功能

支持第三方应用在首次访问物联网平台的开放API之前，完成在物联网平台的接入认证。

## 注意事项

鉴权接口是调用其他API的前提，除了调用鉴权接口（Auth），其他接口调用都需要使用鉴权接口获取的accessToken。

如果多次获取accessToken，则之前的accessToken失效，后一次获取的accessToken才有效。请勿并发获取accessToken。

## 接口描述

```
public function getAuthToken()
```

## 接口所属类

Authentication

## 参数说明

appId和secret使用[NorthApiClient类方法](#)成员变量的[Clientinfo类方法](#)中的值。

## 返回值

AuthOutDTO

参数	描述
\$scope	申请权限范围，即accessToken所能访问物联网平台资源的范围，参数值固定为“default”。
\$tokenType	accessToken的类型，参数值固定为“bearer”。
\$expiresIn	平台生成并返回accessToken的有效时间，单位为秒。
\$accessToken	鉴权参数，访问物联网平台API接口的凭证。
\$refreshToken	鉴权参数，用来刷新accessToken，refreshToken的有效时间为“1”个月。

## 错误码

Http状态码	错误码	错误描述	说明
400	100449	The device is freezed cant operate.	用户无操作权限。 处理建议：请检查appId所对应用户是否有接口调用权限。
400	102202	Required Parameter is null or empty.	必选参数为空。 处理建议：请检查请求参数中的必选参数是否已填写。

Http状态码	错误码	错误描述	说明
401	100208	AppId or secret is not right.	appId或secret错误。 处理建议： <ul style="list-style-type: none"><li>● 请检查appId和secret是否有误，是否新增字符或缺少字符。</li><li>● 请检查请求路径中的IP地址是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.1.2 刷新 token

#### 典型场景

第三方应用通过鉴权接口获取到的accessToken是有有效时间的，在accessToken快过期时，第三方应用通过调用此接口，获取新的accessToken。

#### 接口功能

支持第三方应用在accessToken到期前，从物联网平台获取新的accessToken。

#### 接口描述

```
public function refreshAuthToken(AuthRefreshInDTO $arInDTO)
```

#### 接口所属类

Authentication

#### 参数说明

AuthRefreshInDTO

参数	必选/ 可选	位置	描述
\$appId	必选	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。appid在物联网平台的SP Portal上创建应用时获得。
\$secret	必选	body	第三方应用的密码，与appId对应，用于登录访问物联网平台。secret在物联网平台的SP Portal上创建应用时获得。

参数	必选/ 可选	位置	描述
\$refreshToken	必选	body	刷新token，用来获取一个新的 accessToken。refreshToken在调用 <a href="#">鉴权</a> 接口时获得。

## 返回值

AuthRefreshOutDTO

参数	描述
\$scope	申请的权限范围，参数值固定为“default”。
\$tokenType	鉴权token的类型，参数值固定为“bearer”。
\$expiresIn	平台生成并返回accessToken的有效时间，单位为秒。
\$accessToken	鉴权参数，访问物联网平台API接口的凭证。
\$refreshToken	鉴权参数，用来刷新accessToken，refreshToken的有效时间为“1”个月。

## 错误码

Http状态码	错误码	错误描述	说明
400	100449	The device is freezed cant operate.	用户无操作权限。 处理建议：请检查appId所对应用户是否有接口调用权限。
400	102202	Required Parameter is null or empty.	必选参数为空。 处理建议：请检查请求参数中的必选参数是否已填写。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100208	AppId or secret is not right.	appId或secret错误。 处理建议： <ul style="list-style-type: none"><li>● 请检查appId和secret是否有误，是否新增字符或缺少字符。</li><li>● 请检查请求路径中的IP地址是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 3.3.2 设备管理

第三方应用向物联网平台添加设备，获取设备的ID和验证码，待设备完成接入物联网平台流程后，设备与第三方应用建立从属关系。

### 3.3.2.1 注册直连设备

#### 典型场景

在直连设备接入物联网平台前，第三方应用需要调用此接口在物联网平台注册设备，并把设备的唯一标识码（如IMEI）设置为设备接入平台的验证码。在设备接入物联网平台时携带设备唯一标识，完成设备的接入认证。

#### 接口功能

支持第三方应用在物联网平台注册直连设备，完成注册后设备方可接入物联网平台。

#### 接口描述

```
public function regDirectDevice($rddInDto, $appId, $accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	位置	描述
\$rddInDto	必选	body	见下表 <b>RegDirectDeviceInDTO</b> 的描述。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

RegDirectDeviceInDTO

参数	必选/可选	位置	描述
deviceInfo	可选	Body	设备信息，见下方 <b>DeviceInfo</b> 结构体说明。。
endUserId	可选	Body	终端用户ID。 在NB-IoT方案中，endUserId设置为设备的IMSI号。在Smarthome解决方案中，endUserId设置为App账号。
imsi	可选	Body	NB-IoT终端的IMSI。
isSecure	可选	Body	<p>指定设备是否为安全设备， 默认值为“false”。</p> <ul style="list-style-type: none"><li>● “true”： 安全设备</li><li>● “false”： 非安全设备</li></ul> <p><b>说明</b> 如果用户需要注册安全设备， isSecure参数必须填写。</p>
nodeId	必选	Body	<p>设备的唯一标识，必须与设备上报的设备标识一致。通常使用MAC地址，Serial No或IMEI作为nodeId。</p> <p><b>说明</b> 使用IMEI作为nodeId时，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx， xxxx为IMEI号。</li><li>● 海思芯片设备的唯一标识为IMEI号。</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
psk	可选	Body	请求中指定psk，则平台使用指定的psk；请求中不指定psk，则由平台生成psk。取值范围是“a-f、A-F、0-9”组成的字符串。
timeout	可选	Body	<p>超时时间。当调用北向接口对设备开户，在超时时间内可绑定设备，若超过timeout时间且未绑定设备，则会删除超时的开户信息。</p> <p>取值范围：0~2147483647。若填写为“0”，则表示设备验证码不会失效（建议填写为“0”）。</p> <p>默认值：“180”（默认值可配置，具体配置值请咨询物联网平台运维人员。）</p> <p>单位：秒。</p>

参数	必选/可选	位置	描述
verifyCode	条件可选 必选	body	设备验证码。若请求中指定 verifyCode，则响应中返回请求中指定的verifyCode。若请求中不指定 verifyCode，则由物联网平台自动生成。在NB-IoT方案中，verifyCode为必填参数，且必须与nodeId设置成相同值。
productId	可选	Body	设备所属的产品ID。

DeviceInfo:

参数	必选/可选	位置	描述
manufacture rId	可选	Body	厂商ID，唯一标识一个厂商。
manufacture rName	可选	Body	厂商名称。
deviceType	可选	Body	设备类型，大驼峰命名方式，如 MultiSensor、ContactSensor、 CameraGateway。
model	必选	Body	设备的型号。 Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
protocolType	可选	Body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 返回值

RegDirectDeviceOutDTO

参数	描述
\$deviceId	设备ID，用于唯一标识一个设备。
\$verifyCode	验证码，设备可以通过验证码获取设备ID和密码。若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。
\$timeout	验证码有效时间，单位秒，设备需要在有效时间内接入物联网平台。

参数	描述
\$psk	随机psk参数，若请求中携带了psk，则使用请求中的psk，否则由平台生成随机psk参数。

## 错误码

Http状态码	错误码	错误描述	说明
200	103028	The license pool resources.	License资源用尽。
400	100003	Invalid verify code.	验证码无效。 处理建议：请检查接口请求中的verifyCode是否有误。若请求参数中未带verifyCode，请联系物联网平台维护人员处理。
400	100007	Bad request message.	参数不合法。 处理建议：deviceId未赋值，请参考请求参数说明填写请求。
400	100416	The device has already been binded.	设备已经绑定。 处理建议：请检查设备是否已经注册。
400	100426	The nodeId is duplicated.	nodeId重复。 处理建议：请检查接口请求中nodeId是否有误。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100025	AppId for auth not exist.	获取不到appId鉴权信息。 处理建议： <ul style="list-style-type: none"><li>● 请检查请求结构体的Header头域中是否给字段app_key赋值。</li><li>● 若通过HTTP调用，请联系物联网平台维护人员确认Header头域中放置appId的字段名是app_key还是x-app-key。</li></ul>

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
403	600002	The product not existed.	产品不存在。 处理建议：物联网平台未找到 productId对应的产品，请检查 productId是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100203	The application is not existed.	授权应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	100412	The amount of device has reached the limit.	当前应用下设备数量达到上限。 处理建议：请检查当前应用下的设备数量是否已达到所申请资源的上限。
500	100441	The amount of nonSecure device has reached the limit.	非安全设备数量已达到上限。

Http状态码	错误码	错误描述	说明
500	103026	The license is not exist.	License不存在。 处理建议：物联网平台内部License问题，请联系物联网平台维护人员处理。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.2.2 刷新设备密钥

#### 典型场景

已在物联网平台注册的设备，若设备的唯一标识码发生变更（如更换设备等场景），第三方应用需要调用此接口在物联网平台更新设备的唯一标识，并重新绑定设备。



#### 说明

仅当设备处于离线状态时，才能进行刷新设备秘钥操作。

#### 接口功能

支持第三方应用在物联网平台更新已注册设备的nodeId，在设备deviceId不变的情况下，重新绑新设备。

#### 接口描述

```
public function refreshDeviceKey($rdkInDTO, $deviceId, $appId, $accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	位置	描述
\$rdkInDTO	必选	body	见下表 <b>RegDirectDeviceInDTO</b> 的描述。
\$deviceId	必选	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

## RefreshDeviceKeyInDTO

参数	必选/可选	位置	描述
\$verifyCode	可选	body	备验证码，若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。建议和nodeId设置成一样。
\$nodeId	可选	body	<p>设备的唯一标识，通常使用MAC，MAC地址，Serial No或IMEI作为nodeId。</p> <ul style="list-style-type: none"><li>● 值为空时， nodeId不变。</li><li>● 值不为空时， 更新nodeId。</li></ul> <p><b>说明</b></p> <p>若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx， xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
\$timeout	可选	body	<p>验证码超时时间，单位秒，取值大于等于“0”的整数。</p> <ul style="list-style-type: none"><li>● 值为空时， 使用默认值（“180” s）。</li><li>● 值为“0”时， 永不过期。</li><li>● 值为非“0”， 指定时间。</li></ul>

## 返回值

## RefreshDeviceKeyOutDTO

参数	描述
\$verifyCode	验证码，设备可以通过验证码获取设备ID和密码。若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。
\$timeout	验证码有效时间，单位秒，设备需要在有效时间内接入物联网平台。

## 错误码

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
400	100003	Invalid verify code.	验证码无效。 处理建议：请检查接口请求中的verifyCode是否有误。若请求参数中未带verifyCode，请联系物联网平台维护人员处理。
400	100007	Bad request message.	参数不合法。 处理建议：deviceId未赋值，请参考请求参数说明填写请求。
400	100426	The nodeId is duplicated.	nodeId重复。 处理建议：请检查接口请求中nodeId是否有误。
400	100610	Device is not active.	设备未激活。 处理建议：请检查该设备是否已接入物联网平台并激活。
400	100611	Device is online.	设备在线。 处理建议：请下线该设备或者断开设备与物联网平台的连接。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100025	AppId for auth not exist.	获取不到appId鉴权信息。 处理建议： <ul style="list-style-type: none"><li>● 请检查请求结构体的Header头域中是否给字段app_key赋值。</li><li>● 若通过HTTP调用，请联系物联网平台维护人员确认Header头域中放置appId的字段名是app_key还是x-app-key。</li></ul>
403	100203	The application is not existed.	授权应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.2.3 修改设备信息

#### 典型场景

第三方应用在物联网平台注册设备后，若设备的基本信息发生变更，也可调用此接口在物联网平台修改设备的信息。

#### 接口功能

支持第三方应用修改设备的基本信息，包括设备类型、设备模型、厂商信息、接入协议等。

#### 接口描述

```
public function modifyDeviceInfo($mdiInDto, $deviceId, $appId, $accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	位置	描述
\$mdiInDto	必选	body	见下表 <b>ModifyDeviceInforInDTO</b> 的描述。
\$deviceId	必选	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### ModifyDeviceInforInDTO

参数	必选/可选	位置	描述
\$customFields	可选	body	自定义字段 <b>CustomField</b> 列表，用户可设置自定义字段。
\$deviceConfig	可选	body	设备配置信息，具体参见 <b>DeviceConfigDTO</b> 结构体。

参数	必选/可选	位置	描述
\$deviceType	可选	body	设备类型，大驼峰命名方式，例如：MultiSensor、ContactSensor、CameraGateway。 在NB-IoT方案中，注册设备后必须修改deviceType，且要与profile中定义的保持一致。
\$endUser	可选	body	终端用户，若为直连设备，则endUser可选；若为非直连设备，则endUser可以为null。
\$location	可选	body	设备位置。
\$manufacturerId	可选	body	厂商ID，唯一标识一个厂商。 在NB-IoT方案中，注册设备后必须修改manufacturerId，且要与profile中定义的保持一致。
\$manufacturerName	可选	body	厂商名称。
\$model	可选	body	设备型号。 在NB-IoT方案中，注册设备后必须修改model，且要与profile中定义的保持一致。
\$mute	可选	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
\$name	可选	body	设备名称。
\$organization	可选	body	设备所属的组织信息。
\$protocolType	可选	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
\$region	可选	body	设备区域信息。
\$timezone	可选	body	设备所在时区信息，使用时区编码，如北京时区对应的时区编码为Asia/Beijing。
\$imsi	可选	body	NB-IoT终端的IMSI。
\$ip	可选	body	设备的IP地址。

参数	必选/可选	位置	描述
\$isSecure	可选	body	指定设备的安全状态， 默认值为“false”。 ● “true”： 安全 ● “false”： 非安全
\$psk	可选	body	psk参数， 取值范围是“a-f、 A-F、 0-9”组成的字符串。
\$tags	可选	body	设备的标签信息 <a href="#">Tag2</a> 列表。

CustomField:

参数	必选/可选	位置	描述
fieldName	可选	Body	字段名字。
fieldType	可选	Body	字段类型。
fieldValue	可选	Body	字段值。

DeviceConfigDTO 结构体说明:

参数	必选/可选	描述
\$dataConfig	可选	数据配置信息， 具体参见 <a href="#">DataConfigDTO</a> 结构体。

DataConfigDTO 结构体说明:

参数	必选/可选	字段类型	描述
\$dataAgingTime	可选	Integer	数据老化时长， 取值范围： 0-90， 单位： 天。

Tag2:

参数	必选/可选	类型	位置	描述
tagName	必选	String(1-128)	body	标签名称。

参数	必选/可选	类型	位置	描述
tagValue	必选	String(1-1024)	body	标签值。
tagType	可选	Integer	body	标签类型。

## 返回值

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议: 请检查接口调用请求中携带参数的合法性。
400	100440	The isSecure is invalid.	isSecure参数值有误。
400	50400	The input is invalid.	输入参数无效。 处理建议: 请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
403	500004	The amount of frozen devices has reached the limit.	被冻结设备已达到上限。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	100441	The amount of nonSecure device has reached the limit.	非安全设备数量已达到上限。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.2.4 删除直连设备

#### 典型场景

已在物联网平台注册的直连设备，若不再需要接入平台时，第三方应用可调用此接口在物联网平台删除设备。后续若设备要再次接入平台，第三方应用需要在物联网平台重新注册设备。

#### 接口功能

支持第三方应用在物联网平台删除已注册的直连设备，使设备无法再接入平台。

#### 接口描述

```
public function deleteDirectDevice($deviceId, $cascade, $appId, $accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/ 可选	位置	描述
\$deviceId	必选	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
\$cascade	必选	query	仅当设备下连接了非直连设备时生效，不设置时可填写null。 <ul style="list-style-type: none"><li>● “true”，级联删除，即删除直连设备和其下的非直连设备。</li><li>● “false”，删除直连设备，但是不删其下的非直连设备，并将非直连设备的属性变为直连设备属性。</li></ul>
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 返回值

Status Code: 204 No Content

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.2.5 查询设备激活状态

#### 典型场景

第三方应用通过“注册直连设备”接口在物联网平台注册设备后，在设备首次接入平台之前，设备的激活状态为false；当设备首次接入平台后，设备的激活状态为true（无论设备接入后是处于在线、离线或异常状态）。第三方应用可调用此接口查询设备在物联网平台的激活状态，确定设备是否已接入平台。

#### 接口功能

支持第三方应用通过设备ID查询设备在物联网平台的激活状态，确认设备是否已接入平台。

#### 接口描述

```
public function queryDeviceStatus($deviceId, $appId, $accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	位置	描述
\$deviceId	必选	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 返回值

QueryDeviceStatusOutDTO

参数	描述
\$deviceId	设备ID，用于唯一标识一个设备。
\$activated	激活状态，设备是否通过验证码获取密码的状态标识。 <ul style="list-style-type: none"><li>● “true”：已激活</li><li>● “false”：未激活</li></ul>
\$name	设备名称。

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.2.6 查询设备实时位置信息

#### 典型场景

第三方应用若需要获取设备当前最新的位置信息时，可调用此接口查询设备的实时位置信息，当前仅支持查询NB-IoT设备的实时位置信息。

#### 接口功能

物联网平台通过和位置服务器对接，向第三方应用提供查询NB-IoT设备实时位置信息的能力。

##### 注意

根据位置服务器的能力和配置情况，查询设备实时位置信息接口会存在较大的时延，而且并发能力有限。具体情况建议咨询物联网平台维护人员。

#### 接口描述

```
public function queryDeviceRealtimeLocation($qdrlInDTO, $appId, $accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	位置	描述
\$qdrlInDTO	必选	body	见下表 <b>QueryDeviceRealtimeLocat...</b> 的描述。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### QueryDeviceRealtimeLocationInDTO

参数	必选/可选	位置	描述
\$deviceId	必选	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
\$horAcc	必选	body	水平误差，单位：米，如果不携带，则默认“1000”米。

参数	必选/可选	位置	描述
\$geoInfo	可选	body	地理坐标信息要求， 默认 WGS84， 其他坐标暂不支持。具体参见 <a href="#">GeoInfoDTO</a> 结构体。

GeoInfoDTO:

参数	必选/可选	位置	描述
coordinateReferenceSystem	必选	body	坐标参考系统。具体参见 <a href="#">CoordinateReferenceSystemDTO</a> 结构体

CoordinateReferenceSystemDTO结构体说明:

参数	必选/可选	位置	描述
\$identifier	必选	body	坐标参考系统标识。具体参见 <a href="#">IdentifierDTO</a> 结构体。

IdentifierDTO结构体说明:

参数	必选/可选	位置	描述
\$code	必选	body	坐标参考系统编号， 默认“4326”， 当前只支持这个WGS84数字坐标。
\$codeSpace	必选	body	编码空间， 当前只支持EPSG。
\$edition	必选	body	编辑版本， 当前只支持6.1。

## 返回值

Status Code: 200 OK

QueryDeviceRealtimeLocationOutDTO

参数	描述
\$deviceId	设备ID， 用于唯一标识一个设备。
\$pd	定位结果， 定位成功时携带， 具体参见 <a href="#">PdIoMDTO</a> 结构体。

参数	描述
\$poserr	定位错误信息，定位失败时携，具体参见 <a href="#">PoserrIoMDTO</a> 结构体。

PdIoMDTO结构体说明：

参数	描述
\$time	定位时间，格式“yyyyMMddhhmmss”。
\$utcOff	时间时区，格式“hhmm”。
\$srsName	坐标参考，固定取值：“www.epsg.org#4326”。
\$X	X轴，纬度，格式：“DD MM SS.hhhN”或“DD MM SS.hhhS”，最后的N指北纬，S指南纬。
\$Y	Y轴，经度，格式：“DD MM SS.hhhE”或“DD MM SS.hhhW”，最后的E指东经，W指西经。
\$radius	误差半径，单位：米。

PoserrIoMDTO结构体说明：

参数	描述
\$time	定位时间，格式“yyyyMMddhhmmss”。

参数	描述
\$resid	定位结果值。 <ul style="list-style-type: none"><li>● “2”：定位成功</li><li>● “104”：基站信息获取失败</li><li>● “225”：用户停机</li><li>● “226”：用户关机</li><li>● “227”：用户不在服务区</li><li>● “228”：定位失败</li><li>● “229”：定位失败</li><li>● “230”：定位失败</li><li>● “231”：定位失败</li><li>● “232”：定位失败</li><li>● “233”：定位失败</li><li>● “245”：用户没有在定位平台开户</li><li>● “247”：定位失败</li><li>● “248”：定位失败</li><li>● “249”：定位失败</li><li>● “250”：定位失败</li><li>其他值：定位失败</li></ul>
\$add_info	定位详细结果。

## 错误码

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	103801	Exceed the system concurrency restriction.	超过系统并发限制。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.2.7 查询设备影子

#### 典型场景

若设备处于离线或异常状态时，第三方应用无法通过命令下发的方式向设备下发配置。此时可把要下发的配置设置到设备影子，当设备重新上线时，设备影子会把配置下发给设备。第三方应用可调用此接口查看设备影子上的设备配置信息及最新的设备上报数据信息。

## 接口功能

支持第三方应用查询单个设备的设备影子信息，包括对设备的配置信息（desired区）和设备最新上报的数据信息（reported区）。

## 接口描述

```
public function queryDeviceShadow($deviceId, $appId, $accessToken)
```

## 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	位置	描述
\$deviceId	必选	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 返回值

Status Code: 200 OK

QueryDeviceShadowOutDTO

参数	描述
\$deviceId	设备ID，用于唯一标识一个设备。
\$gatewayId	网关ID，用于标识一个网关。
\$nodeType	设备的类型。
\$createTime	设备创建的时间。
\$lastModifiedTime	设备最后一次修改信息的时间。
\$deviceInfo	设备详细信息，具体参见 <a href="#">DeviceInfo</a> 结构体。
\$services	设备的服务能力， <a href="#">DeviceServiceB</a> 列表

DeviceInfo结构体说明：

参数	描述
\$nodeId	设备的唯一标识。
\$name	设备名称。
\$description	设备的描述信息。
\$manufacturerId	厂商ID，唯一标识一个厂商，与设备Profile文件中的信息保持一致。
\$manufacturerName	厂商名称，与设备Profile文件中的信息保持一致。
\$mac	设备的MAC地址。
\$location	设备的位置信息。
\$deviceType	设备类型，大驼峰命名方式，与设备Profile文件中的信息保持一致，如MultiSensor、ContactSensor、CameraGateway。
\$model	设备的型号，与设备Profile文件中的信息保持一致。 Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
\$swVersion	设备的软件版本。 Z-Wave: 主版本号.次版本号，如：1.1。
\$fwVersion	设备的固件版本。
\$hwVersion	设备的硬件版本。
\$protocolType	设备使用的协议类型，与设备Profile文件中的信息保持一致，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
\$bridgeId	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
\$status	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
\$statusDetail	设备的状态详情，status条件取值，取值查看下表 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
\$mute	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
\$supportedSecurity	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>

参数	描述
\$isSecurity	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
\$signalStrength	设备的信号强度。
\$sigVersion	设备的sig版本。
\$serialNumber	设备的序列号。
\$batteryLevel	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceServiceB说明:

参数	描述
\$serviceId	服务ID，用于标识一个服务。
\$reportedProps	设备上报的信息。
\$desiredProps	设备下发的信息。
\$eventTime	事件发生的时间。
\$serviceType	服务类型。

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId 是否有误。</li><li>● 请检查请求路径（url）中传入的 appId 是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的 appId 为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	pp_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的 deviceId 错误。请确认该 deviceId 是否属于该 appId 或者 deviceId 是否写错。</li><li>● Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。</li><li>● 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId 是否有误。</li><li>● 请检查请求路径（url）中传入的 appId 是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.2.8 修改设备影子

#### 典型场景

物联网平台支持创建设备的“影子”。设备影子用于存储设备最新上报的服务属性数据，以及第三方应用对服务属性的配置（服务属性为设备Profile文件中定义的service属性）。若设备处于离线或异常状态时，第三方应用无法通过命令下发的方式向设备下发配置，此时可把要下发的配置设置到设备影子，当设备重新上线时，设备影子会把配置下发给设备。第三方应用可调用此接口在设备影子上配置要下发给设备的配置信息。

每个设备有且只有一个设备影子，设备影子上有desired和report区。

- desired区用于存储对设备服务属性的配置。若设备当前在线，desired区的配置会即时下发给设备；若设备当前不在线，desired区的配置会在设备上线时再下发给设备。
- report区用于存储设备最新上报的服务属性数据。当设备上报数据时，平台会把数据同步到设备影子的report区。

#### 接口功能

支持第三方应用配置设备影子desired区的信息，用于设备上线时把配置下发给设备。

#### 接口描述

```
public function modifyDeviceShadow($mdsInDTO, $deviceId, $appId, $accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	位置	描述
\$mdsInDTO	必选	body	见下表 <b>ModifyDeviceShadowInDTO</b> 的描述。
\$deviceId	必选	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

ModifyDeviceShadowInDTO

参数	必选/可选	位置	描述
\$serviceDesireds	必选	body	ServiceDesiredDTO列表，需要修改的设备配置或状态信息。 <b>ServiceDesiredDTO</b> 见下表。

ServiceDesiredDTO结构体说明：

参数	必选/可选	位置	描述
\$serviceId	可选	body	服务ID，用于标识一个服务。
\$desired	可选	body	设备的状态。

## 返回值

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
200	100425	The special deviceCapability is not exist.	设备模板不存在。 处理建议：请检查设备模板是否已在物联网平台上传。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者 serviceId在Profile文件中是否存在。</li></ul>
400	107002	The properties is empty in database.	设备属性不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107003	The request properties is unknown.	设备状态为未知。 处理建议：请检查设备与平台的连接是否正常。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议:</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	<p>应用未被授权。</p> <p>处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。</p>
403	100443	The property is forbidden to write.	设备属性不可写。
403	101004	app throttle exceed.	<p>应用调用接口过于频繁, 超过流控值（默认值是100次/60s）。</p> <p>处理建议: 联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。</p>
403	101005	pp_key or access_token is invalid.	<p>错误的token信息。</p> <p>处理建议: 请检查接口请求中所携带的accessToken是否有误。</p>
404	100403	The device is not existed.	<p>设备不存在。</p> <p>处理建议: 物联网平台未找到deviceId对应的设备, 请检查deviceId是否有误。</p>
404	100418	The deviceData is not existed.	<p>设备信息不存在。</p> <p>处理建议:</p> <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误, 请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId, 请检查该appId是否有误。</li></ul>
500	100023	The data in DataBase is abnormal.	<p>数据库异常。</p> <p>处理建议: 物联网平台内部错误, 请联系物联网平台维护人员处理。</p>
500	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议:</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.3 批量处理

第三方应用通过批量处理接口对已接入物联网平台的设备进行批量操作。

#### 3.3.3.1 创建批量任务

##### 典型场景

第三方应用若需要对多个设备进行批量操作时，可调用此接口创建设备的批量操作任务，当前仅支持给批量设备下发缓存命令。

##### 接口功能

支持第三方应用在物联网平台创建设备的批量操作任务。

##### 接口描述

```
public function createBatchTask($btcInDTO, $accessToken)
```

##### 接口所属类

BatchProcess

##### 参数说明

参数	必选/可选	位置	描述
\$btcInDTO	必选	body	见下表 <b>BatchTaskCreateInDTO</b> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

BatchTaskCreateInDTO

参数	必选/可选	位置	描述
\$appId	必选	body	如果是本应用的任务，此参数值可以填写null，否则填写授权应用的appId。
\$timeout	必选	body	任务超时时长，单位秒，范围为10-2880。

参数	必选/可选	位置	描述
\$taskName	必选	body	任务名称，最大长度“256”字符。
\$taskType	必选	body	任务类型，取值范围：“DeviceReg/DeviceCmd/DeviceLocation/DeviceMod/DeviceDel/DeviceLicense”。
\$param	必选	body	任务详细参数，根据taskType任务类型的不同对应不同类型参数，具体参见 <a href="#">DeviceCmd</a> 。
\$tags	可选	body	标签列表，具体参见 <a href="#">TagDTO2</a> 。

DeviceCmd:

参数	必选/可选	类型	位置	描述
type	必选	String	body	批量命令类型，取值范围：DeviceList/DeviceType/DeviceArea/GroupList/Broadcast/GroupIdList。
deviceList	条件必选	List<String>	body	设备ID列表，type为DeviceList时需要填写。
deviceType	条件必选	String	body	设备类型，type为DeviceType时需要填写，其值应当与profile中定义的一致。
manufacturerId	条件可选	String	body	厂商ID，type为DeviceType时可填写，其值应当与profile中定义的一致。
model	条件可选	String	body	设备型号，type为DeviceType时可填写，其值应当与profile中定义的一致。
deviceLocation	条件必选	String	body	设备位置，type为DeviceArea时需要填写。
groupList	条件必选	List<String>	body	群组Id列表或设备组名称列表，type为GroupIdList时需要填写群组Id，type为GroupList时需要填写设备组名称。
command	必选	<a href="#">CommandDTO</a>	body	命令信息。
callbackUrl	可选	String	body	命令执行结果的推送地址。
maxRetransmit	可选	Integer(0~3)	body	命令下发最大重传次数，取值范围：0~3。

参数	必选/可选	类型	位置	描述
groupTag	可选	String	body	群组标签。

CommandDTO:

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-64)	body	命令对应的服务ID，要与profile中定义的serviceId保持一致。
method	必选	String(1-128)	body	服务下具体的命令名称，要与profile中定义的命令名保持一致。
paras	可选	ObjectNode	body	命令参数， jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。

TagDTO2结构体说明:

参数	必选/可选	位置	描述
\$tagName	必选	body	标签名称。
\$tagValue	必选	body	标签值。

## 返回值

BatchTaskCreateOutDTO

参数	描述
\$taskID	批量任务的ID。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	101001	Resource doesn't exist.	资源不存在。
200	105001	The batchTask count has reached the limit.	未完成的任务数大于等于10个时，返回任务达到数量限制。
200	105002	The batchTask name has exist.	任务名字已存在。 处理建议：修改任务名称。
400	105201	The tagName and tagValue has been used on the platform.	平台已使用tagName和tagValue。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100028	The user has no right.	用户没有操作权限。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105202	The tag is not existed.	标签不存在。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.3.2 查询指定批量任务信息

#### 典型场景

第三方应用通过“创建批量任务”接口创建了设备的批量操作任务后，可调用此接口查询该批量任务的信息，包括任务的当前状态、子任务完成情况等。

#### 接口功能

支持第三方应用通过任务ID查询单个批量任务的信息。

#### 接口描述

```
public function queryOneTask($taskId, $select, $appId, $accessToken)
```

#### 接口所属类

BatchProcess

#### 参数说明

参数	必选/可选	位置	描述
\$taskId	必选	path	批量任务ID，创建批量任务后获得。
\$select	必选	query	指定可选的返回值，可取值：“tag”。不指定时可填写null。
\$appId	必选	query	如果是本应用的任务，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 返回值

QueryOneTaskOutDTO

参数	描述
\$taskId	批量任务ID。
\$taskName	批量任务名称。
\$appId	批量任务所归属的appId。
\$operator	下发该批量任务的操作员。
\$taskFrom	批量任务的来源。 <ul style="list-style-type: none"><li>● “Portal”：通过SP Portal创建的。</li><li>● “Northbound”：调用北向API接口创建的。</li></ul>
\$taskType	批量任务的类型，取值范围：“DeviceReg/DeviceCmd”。
\$status	批量任务的状态，取值范围：“Pending/Running/Complete/Timeout”。
\$startTime	批量任务的创建时间。
\$timeout	批量任务的超时时间，单位秒。
\$progress	批量任务的进度，单位：千分比，范围：0-1000，向下取整。
\$totalCnt	任务的详情总数。
\$successCnt	成功的任务详情数。
\$failCnt	失败的任务详情数
\$timeoutCnt	超时的任务详情数。
\$expiredCnt	未执行的失效任务详情数。
\$completeCnt	完成的任务详情数，完成的任务详情数=成功的任务详情数+失败的任务详情数+超时的任务详情数。
\$successRate	任务成功率，单位：千分比，范围：0-1000，向下取整。
\$param	不同任务类型的具体参数，具体参见 <a href="#">DeviceCmd</a> 结构。
\$tags	批量任务的标签列表，具体参见 <a href="#">TagDTO2</a> 结构体。

DeviceCmd:

参数	描述
type	批量命令类型，取值范围：DeviceList/DeviceType/DeviceArea/GroupList/Broadcast/GroupIdList。
deviceList	设备ID列表，type为DeviceList时返回值。

参数	描述
deviceType	设备类型, type为DeviceType时返回值, 其值应当与profile中定义的一致。
manufacturerId	厂商ID, type为DeviceType时返回值, 其值应当与profile中定义的一致。
model	设备型号, type为DeviceType时返回值, 其值应当与profile中定义的一致。
deviceLocation	设备位置, type为DeviceArea时返回值。
groupList	群组名称列表, type为GroupList时返回值。
command	命令信息。具体参见 <a href="#">CommandDTO</a> 。
callbackUrl	命令执行结果的推送地址。
maxRetransmit	命令下发最大重传次数, 取值范围: 0-3。
groupTag	群组标签。

CommandDTO:

参数	描述
serviceId	命令对应的服务ID, 要与profile中定义的serviceId保持一致。
method	服务下具体的命令名称, 要与profile中定义的命令名保持一致。
paras	命令参数, jsonString格式, 里面是一个个键值对, 每个键都是profile中命令的参数名 (paraName), 具体格式需要应用和设备约定。

TagDTO2结构体说明:

参数	描述
\$tagName	标签名称。
\$tagValue	标签值。

## 错误码

Http状态码	错误码	错误描述	说明
400	100019	Illegal request.	非法请求。 处理建议：请检查接口请求中的必须参数是否为空。
400	100022	The input is invalid	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105005	The batchTask is not existed.	批量任务不存在。 处理建议：请检查接口请求中的taskId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

### 3.3.3.3 查询批量任务的子任务信息

#### 典型场景

第三方应用通过“创建批量任务”接口创建了设备的批量操作任务后，可调用此接口查询该批量任务中子任务的详情信息，如任务的执行状态、任务内容等。

#### 接口功能

支持第三方应用根据条件查询批量任务中子任务的详情信息，支持查询本应用创建的批量任务信息。

#### 接口描述

```
public function queryTaskDetails($qtdInDTO, $accessToken)
```

#### 接口所属类

BatchProcess

#### 参数说明

参数	必选/可选	位置	描述
\$qtdInDTO	必选	query	见下表 <b>QueryTaskDetailsInDTO</b> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### QueryTaskDetailsInDTO

参数	必选/可选	位置	描述
\$appId	可选	query	如果是本应用的任务，此参数值可以填写null，否则填写授权应用的appId。
\$taskId	必选	query	批量任务的ID。
\$status	可选	query	任务的详情状态，“Pending/Success/Fail/Timeout”。
\$index	可选	query	批量任务文件里第几行的任务，查询批量注册任务时使用。
\$nodeId	可选	query	设备nodeId，查询批量注册任务时使用。
\$deviceId	可选	query	设备Id，查询批量命令任务时使用。
\$commandId	可选	query	命令Id，查询批量命令任务时使用。

参数	必选/ 可选	位置	描述
\$pageNo	可选	query	分页查询参数。 ● 值为空时，查询内容不分页。 ● 值为大于等于0的整数时，分页查询。 ● 值为0时查询第一页。
\$pageSize	可选	query	分页查询参数，取值大于等于1的整数，缺省值：“1”。

## 返回值

QueryTaskDetailsOutDTO

参数	描述
\$pageNo	分页查询参数。 ● 值为空时，查询内容不分页。 ● 值为大于等于0的整数时，分页查询。 ● 值为0时查询第一页。
\$pageSize	分页查询参数，取值大于等于1的整数，缺省值：“1”。
\$totalCount	查询到的任务总数。
\$taskDetails	任务详情信息（ <a href="#">QueryTaskDetailDTOCloud2NA</a> ）列表。

QueryTaskDetailDTOCloud2NA:

参数	描述
status	任务执行状态，取值范围：Pending/WaitResult/Success/Fail/Timeout。
output	批量下发命令的输出信息。
error	任务的错误原因，格式为： {"error_code": "****", "error_desc": "****"}。
param	不同任务类型的具体参数，具体参见 <a href="#">ObjectNode</a> 。

ObjectNode:

参数	描述
deviceId	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
commandId	下发的命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。

## 错误码

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105005	The batchTask is not existed.	批量任务不存在。 处理建议：请检查接口请求中的 taskId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.4 订阅管理

物联网平台支持第三方应用对设备数据的订阅，若订阅的设备信息有变更，平台会推送给第三方应用。消息订阅接口需要与[消息推送](#)接口配合使用。

#### 3.3.4.1 订阅平台业务数据

##### 典型场景

第三方应用可在物联网平台订阅设备相关的业务数据，当物联网平台中设备的业务信息发生变化时（如设备注册、设备数据上报、设备状态变更等），平台会向第三方应用发送通知消息，通知其具体的变化信息。第三方应用可以调用此接口向物联网平台订阅不同类型的设备业务变更通知。

##### 接口功能

支持第三方应用向物联网平台订阅设备的业务变更信息，当设备的状态、数据等信息发生变化时，平台会向第三方应用发送通知消息。

##### 接口描述

```
public function subDeviceData($sddInDTO, $ownerFlag, $accessToken)
```

##### 参数说明

参数	必选/可选	位置	描述
\$sddInDTO	必选	body	见下表 <b>SubDeviceDataInDTO</b> 的描述。
\$ownerFlag	必选	query	callbackUrl的所有者标识，不指定该标识可填写null。 <ul style="list-style-type: none"><li>● ownerflag为false时，表示callbackUrl的owner是授权应用。</li><li>● ownerflag为true时，表示callbackUrl的owner为被授权应用。</li></ul>
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

SubDeviceDataInDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	<p>通知类型，第三方应用可以根据通知类型接收物联网平台推送的对应通知消息。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备，订阅后推送绑定设备通知）</li><li>● deviceAdded（添加新设备，订阅后推送注册设备通知）</li><li>● deviceInfoChanged（设备信息变化，订阅后推送设备信息变化通知）</li><li>● deviceDataChanged（设备数据变化，订阅后推送设备数据变化通知）</li><li>● deviceDatasChanged（设备数据批量变化，订阅后推送批量设备数据变化通知）</li><li>● deviceDeleted（删除设备，订阅后推送删除设备通知）</li><li>● messageConfirm（消息确认，订阅后推送设备消息确认通知）</li><li>● commandRsp（命令响应，订阅后推送设备命令响应通知）</li><li>● deviceEvent（设备事件，订阅后推送设备事件通知）</li><li>● serviceInfoChanged（服务信息变化，订阅后推送设备服务信息变化通知）</li><li>● deviceModelAdded（添加设备模型，订阅后推送增加设备模型通知）</li><li>● deviceModelDeleted（删除设备模型，订阅后推送删除设备模型通知）</li><li>● deviceDesiredPropertiesModifyStatusChanged（设备影子状态变更，订阅后推送设备影子状态变更通知）</li></ul>
\$callbackUrl	必选	body	<p>订阅的回调地址，用于接收对应类型的通知消息。</p> <p>必须使用HTTPS信道回调地址，同时回调地址中必须指定回调地址的端口。例如： <a href="https://XXX.XXX.XXX.XXX:443/callbackurltest">https://XXX.XXX.XXX.XXX:443/callbackurltest</a></p> <p><b>说明</b> HTTP信道只可用于调测。</p>
\$appId	可选	body	设备或者规则等订阅内容主体所属的appId。
\$channel	可选	body	传输通道，若是MQTT客户端订阅，则取值为MQTT，其他情况为HTTP。

## 响应参数

Status Code: 201 Created

SubscriptionDTO

参数	描述
subscriptionId	订阅ID号，用于标识一个订阅。
notifyType	通知的类型。
callbackUrl	订阅的回调地址。
clientIds	MQTT客户端ID，只有MQTT订阅时该字段有返回值。

## 错误码

Http状态码	错误码	错误描述	说明
400	100222	The request callbackurl is illegal.	回调地址非法。 处理建议：请检查请求体中的回调地址是否有误。
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
409	100227	The resource is conflicted.	资源冲突，通知类型已经被订阅。 处理建议：请检查要订阅的通知类型是否已被订阅。

### 3.3.4.2 订阅平台管理数据

#### 典型场景

第三方应用可向物联网平台订阅设备相关的管理数据，当物联网平台中设备进行版本升级等相关操作时，平台会向第三方应用发送通知消息，知会其具体的操作状态或操

作结果信息。第三方应用可以调用此接口向物联网平台订阅不同类型的设备升级操作通知。

## 接口功能

支持第三方应用向物联网平台订阅设备的升级操作信息，当设备进行版本升级等相关操作时，平台会向第三方应用发送通知消息。

## 接口描述

```
public function subDeviceData2($smdInDTO, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$smdInDTO	必选	body	见下表 <b>SubDeviceManagementDataInDTO</b> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### SubDeviceManagementDataInDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型。 <ul style="list-style-type: none"><li>● swUpgradeStateChangeNotify（软件升级状态变更通知，订阅后推送软件升级状态变更通知）</li><li>● swUpgradeResultNotify（软件升级结果通知，订阅后推送软件升级结果通知）</li><li>● fwUpgradeStateChangeNotify（固件升级状态变更通知，订阅后推送固件升级状态变更通知）</li><li>● fwUpgradeResultNotify（固件升级结果通知，订阅后推送固件升级结果通知）</li></ul>
\$callbackurl	必选	body	订阅的回调地址，用于接收对应类型的通知消息。 必须使用HTTPS信道回调地址，同时回调地址中必须指定回调地址的端口。例如： <a href="https://XXX.XXX.XXX.XXX:443/callbackurltest">https://XXX.XXX.XXX.XXX:443/callbackurltest</a> <b>说明</b> HTTP信道只可用于调测。

## 响应参数

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
400	100222	Internal server error.	回调地址非法。 处理建议：请检查请求体中的回调地址是否有误。
400	100228	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100229	Get AppKey from header failed.	从消息头获取AppKey失败。
500	100244	register out route fail.	注册路由失败。 处理建议：联系物联网平台维护人员。

### 3.3.4.3 查询单个订阅

#### 典型场景

第三方应用可在物联网平台上创建多个订阅，以订阅不同类型的设备变更通知。第三方应用若需要查看某个订阅的配置信息，可调用此接口进行查询。

## 接口功能

支持第三方应用在物联网平台根据订阅ID查询指定订阅的配置信息。

## 接口描述

```
public function querySingleSubscription($subscriptionId, $appId, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$subscriptionId	必选	path	订阅ID号，通过调用订阅接口返回或者查询订阅接口获得。
\$appId	必选	query	设备或者规则等订阅内容主体所属的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

SubscriptionDTO

参数	描述
\$subscriptionId	订阅ID号，用于标识一个订阅。
\$notifyType	通知的类型。
\$callbackUrl	订阅的回调地址。

## 错误码

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 3.3.4.4 批量查询订阅

#### 典型场景

第三方应用可在物联网平台上创建多个订阅，以订阅不同类型的设备变更通知。第三方应用若需要查看本应用下所有订阅配置，或者查看某种订阅类型下的所有订阅配置，可调用此接口进行查询。

#### 接口功能

支持第三方应用在物联网平台查询本应用下的所有订阅信息，或者某种订阅类型下的所有订阅信息。

#### 接口描述

```
public function queryBatchSubscriptions($qbsInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$qbsInDTO	必选	query	见下表 <b>QueryBatchSubInDTO</b> 的描述。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### QueryBatchSubInDTO

参数	必选/可选	位置	描述
\$appId	可选	query	设备或者规则等订阅内容主体所属的appId。

参数	必选/可选	位置	描述
\$notifyType	可选	query	<p>通知类型，第三方应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备）</li><li>● deviceAdded（添加新设备）</li><li>● deviceInfoChanged（设备信息变化）</li><li>● deviceDataChanged（设备数据变化）</li><li>● deviceDatasChanged（设备数据批量变化）</li><li>● deviceDeleted（删除设备）</li><li>● messageConfirm（消息确认）</li><li>● commandRsp（命令响应）</li><li>● deviceEvent（设备事件）</li><li>● serviceInfoChanged（服务信息变化）</li><li>● deviceModelAdded（添加设备模型）</li><li>● deviceModelDeleted（删除设备模型）</li><li>● deviceDesiredPropertiesModifyStatusChanged（修改设备影子状态变更）</li><li>● swUpgradeStateChangeNotify（软件升级状态变更通知）</li><li>● swUpgradeResultNotify（软件升级结果通知）</li><li>● fwUpgradeStateChangeNotify（固件升级状态变更通知）</li><li>● fwUpgradeResultNotify（固件升级结果通知）</li></ul>
\$pageNo	可选	query	<p>分页查询参数。</p> <ul style="list-style-type: none"><li>● 值为空时，查询内容不分页</li><li>● 值大于等于0时分页查询</li><li>● 值等于0时查询第一页</li></ul>
\$pageSize	可选	query	分页查询参数，取值大于等于1的整数，缺省值：“10”。

## 响应参数

QueryBatchSubOutDTO

参数	描述
\$totalCount	查询到的订阅记录总数。

参数	描述
\$pageNo	查询的页码。
\$pageSize	查询每页信息的数量。
\$subscriptions	订阅信息列表，具体参见 <a href="#">SubscriptionDTO</a> 结构体。

SubscriptionDTO结构体说明：

参数	描述
\$subscriptionId	订阅ID号。
\$notifyType	通知类型。
\$callbackUrl	订阅的回调地址。

## 错误码

Http状态码	错误码	错误描述	说明
400	100224	The resource exceeds 1000, please refinement query conditions.	查到端资源超出1000条，请重新设定查询条件。 处理建议：请缩小查询条件范围。
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 3.3.4.5 删除单个订阅

#### 典型场景

第三方应用若不再需要接收物联网平台推送的某个订阅通知消息时，可调用此接口删除指定的订阅配置，取消订阅。

## 接口功能

支持第三方应用在物联网平台根据订阅ID删除指定的订阅配置。

## 接口描述

```
public function deleteSingleSubscription($subscriptionId, $appId, $accessToken)
```

## 参数说明

参数	必选/ 可选	位置	描述
\$subscriptionId	必选	path	订阅ID号。
\$appId	可选	query	设备或者规则等订阅内容主体所属的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

Status Code: 204 No Content

## 错误码

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100225	The resource is not found	找不到资源。 处理建议：请检查订阅ID是否有误。

### 3.3.4.6 批量删除订阅

#### 典型场景

第三方应用若不再需要接收物联网平台推送的订阅通知消息，或者不再需要接收某类订阅通知消息，可调用此接口批量删除订阅配置，取消订阅。

#### 接口功能

支持第三方应用在物联网平台批量删除订阅配置，可删除全部订阅，或删除指定订阅类型的所有订阅，或删除指定回调URL地址的所有订阅。

#### 接口描述

```
public function deleteBatchSubscriptions($dbsInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$dbsInDTO	必选	body	见下表 <b>DeleteBatchSubInDTO</b> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### DeleteBatchSubInDTO

参数	必选/可选	位置	描述
\$appId	可选	query	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。appid在物联网平台的SP Portal上创建应用时获得。此处填写授权应用的appid。

参数	必选/可选	位置	描述
\$notifyType	可选	query	<p>通知类型，第三方应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备）</li><li>● deviceAdded（添加新设备）</li><li>● deviceInfoChanged（设备信息变化）</li><li>● deviceDataChanged（设备数据变化）</li><li>● deviceDatasChanged（设备数据批量变化）</li><li>● deviceDeleted（删除设备）</li><li>● messageConfirm（消息确认）</li><li>● commandRsp（命令响应）</li><li>● deviceEvent（设备事件）</li><li>● serviceInfoChanged（服务信息变化）</li><li>● deviceModelAdded（添加设备模型）</li><li>● deviceModelDeleted（删除设备模型）</li><li>● deviceDesiredPropertiesModifyStatusChanged（修改设备影子状态变更）</li><li>● swUpgradeStateChangeNotify（软件升级状态变更通知）</li><li>● swUpgradeResultNotify（软件升级结果通知）</li><li>● fwUpgradeStateChangeNotify（固件升级状态变更通知）</li><li>● fwUpgradeResultNotify（固件升级结果通知）</li></ul>
\$callbackUrl	可选	query	订阅回调的URL地址。
\$channel	可选	query	<p>传输通道。</p> <ul style="list-style-type: none"><li>● 当channel为空时，删除“传输通道”为“MQTT”和“HTTP”的订阅。</li><li>● 当channel为“MQTT”时，删除“传输通道”为“MQTT”的订阅。</li><li>● 当channel为“HTTP”时，删除“传输通道”为“HTTP”的订阅。</li><li>● 当channel取其他值时，不会删除订阅。</li></ul>

## 响应参数

Status Code: 204 No Content

## 错误码

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100225	The resource is not found	找不到资源。 处理建议：请检查订阅类型是否有误。

## 3.3.5 消息推送

第三方应用向物联网平台订阅设备信息，当设备信息发生变更时，物联网平台向第三方应用推送此消息，第三方应用根据通知类型对消息分派处理。需要配合[订阅管理](#)接口使用。

### 3.3.5.1 注册设备通知

#### 典型场景

第三方应用在物联网平台订阅了注册设备通知后（订阅的通知类型为deviceAdded），当第三方应用通过“注册直连设备”接口在平台注册设备时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了注册设备通知的第三方应用推送通知消息。

#### 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleDeviceAdded(NotifyDeviceAddedDTO \$body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceAddedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“deviceAdded”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$gatewayId	可选	body	网关ID，用于唯一标识一个网关设备。
\$nodeType	必选	body	设备类型。 <ul style="list-style-type: none"><li>● “ENDPOINT”</li><li>● “GATEWAY”</li><li>● “UNKNOWN”</li></ul>
\$DeviceInfo	必选	body	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。

DeviceInfo结构体说明：

参数	必选/可选	位置	描述
\$nodeId	必选	body	设备的唯一标识，通常使用MAC，MAC地址，Serial No或IMEI作为nodeId。 <b>说明</b> 若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。 <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
\$name	可选	body	设备名称。
\$description	可选	body	设备的描述信息。
\$manufacturerId	可选	body	厂商ID，唯一标识一个厂商。
\$manufacturerName	可选	body	厂商名称。

参数	必选/可选	位置	描述
\$mac	可选	body	设备的MAC地址。
\$location	可选	body	设备的位置信息。
\$deviceType	可选	body	设备类型，大驼峰命名方式，如 MultiSensor、ContactSensor、CameraGateway。
\$model	可选	body	设备的型号。
\$swVersion	可选	body	设备的软件版本。 Z-Wave：主版本号.次版本号，如：1.1。
\$fwVersion	可选	body	设备的固件版本。
\$hwVersion	可选	body	设备的硬件版本。
\$protocolType	可选	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
\$bridgeId	可选	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
\$status	可选	body	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
\$statusDetail	可选	body	设备的状态详情，status条件取值，取值查看 <a href="#">\$status</a> 和 <a href="#">\$statusDetail</a> 。
\$mute	可选	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
\$supportedSecurity	可选	body	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
\$isSecurity	可选	body	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
\$signalStrength	可选	body	设备的信号强度。
\$sigVersion	可选	body	设备的sig版本。

参数	必选/可选	位置	描述
\$serialNumber	可选	body	设备的序列号。
\$batteryLevel	可选	body	设备的电池电量。

\$status和\$statusDetail:

\$status	\$statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceAdded",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "nodeType": "GATEWAY",  
    "deviceInfo": {  
        "nodeId": "*****",  
        "name": null,  
        "description": null,  
    }  
}
```

```
"manufacturerId": null,  
"manufacturerName": null,  
"mac": null,  
"location": null,  
"deviceType": null,  
"model": null,  
"swVersion": null,  
"fwVersion": null,  
"hwVersion": null,  
"protocolType": null,  
"bridgeId": null,  
"status": "OFFLINE",  
"statusDetail": "NOT_ACTIVE",  
"mute": null,  
"supportedSecurity": null,  
"isSecurity": null,  
"signalStrength": null,  
"sigVersion": null,  
"serialNumber": null,  
"batteryLevel": null  
}  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 3.3.5.2 绑定设备通知

#### 典型场景

第三方应用在物联网平台订阅了绑定设备通知后（订阅的通知类型为bindDevice），当直连设备接入并绑定物联网平台时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了绑定设备通知的第三方应用推送通知消息。

## 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过继承[PushMessageReceiver](#)类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleBindDevice(NotifyBindDeviceDTO \$body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyBindDeviceDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“bindDevice”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$resultCode	必选	body	绑定结果，取值：“expired/succeeded”。
\$DeviceInfo	可选	body	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。

DeviceInfo结构体说明：

参数	必选/可选	位置	描述
\$nodeId	必选	body	标识设备的唯一ID。
\$name	可选	body	设备名称。
\$description	可选	body	设备的描述信息。
\$manufacturerId	可选	body	厂商ID，唯一标识一个厂商。
\$manufacturerName	可选	body	厂商名称。
\$mac	可选	body	设备的MAC地址。
\$location	可选	body	设备的位置信息。

参数	必选/可选	位置	描述
\$deviceType	可选	body	设备类型，大驼峰命名方式，如 MultiSensor、ContactSensor、CameraGateway。
\$model	可选	body	设备的型号。
\$swVersion	可选	body	设备的软件版本。 Z-Wave：主版本号.次版本号，如：1.1。
\$fwVersion	可选	body	设备的固件版本。
\$hwVersion	可选	body	设备的硬件版本。
\$protocolType	可选	body	设备使用的协议类型： Z-Wave, ZigBee, WPS。
\$bridgeId	可选	body	Bridge标识，表示设备通过哪个 Bridge接入物联网平台。
\$status	可选	body	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
\$statusDetail	可选	body	设备的状态详情，status条件取值，取值查看 <a href="#">\$status和\$statusDetail</a> 。
\$mute	可选	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
\$supportedSecurity	可选	body	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
\$isSecurity	可选	body	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
\$signalStrength	可选	body	设备的信号强度。
\$sigVersion	可选	body	设备的sig版本。
\$serialNumber	可选	body	设备的序列号。
\$batteryLevel	可选	body	设备的电池电量。

\$status和\$statusDetail:

\$status	\$statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

```
request: {callbackUrl}  
Header:  
Content-Type:application/json  
Body:  
{  
    "notifyType": "bindDevice",  
    "deviceId": "*****",  
    "resultCode": "succeeded",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturer": "wulian",  
        "deviceType": 90,  
        "model": "90",  
        "mac": "*****",  
        "swVersion": "...",  
        "fwVersion": "...",  
        "hwVersion": "...",  
        "protocolType": "zigbee",  
        "description": "smockdetector",  
        "nodeType": "GATEWAY"
```

```
}
```

```
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 3.3.5.3 设备信息变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备信息变化通知后（订阅的通知类型为 deviceInfoChanged），当设备的配置或状态信息（如设备厂商信息、位置信息、版本信息、在线状态信息等）发生变化时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备信息变化通知的第三方应用推送通知消息。

#### 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleDeviceInfoChanged(NotifyDeviceInfoChangedDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyDeviceInfoChangedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“deviceInfoChanged”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$gatewayId	必选	body	网关ID，用于唯一标识一个网关设备。

参数	必选/可选	位置	描述
\$deviceInfo	必选	body	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。

DeviceInfo结构体说明：

参数	必选/可选	位置	描述
\$nodeId	必选	body	设备的唯一标识，通常使用MAC，MAC地址，Serial No或IMEI作为nodeId。 <b>说明</b> 若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。 <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
\$name	可选	body	设备名称。
\$description	可选	body	设备的描述信息。
\$manufacturerId	可选	body	厂商ID，唯一标识一个厂商。
\$manufacturerName	可选	body	厂商名称。
\$mac	可选	body	设备的MAC地址。
\$location	可选	body	设备的位置信息。
\$deviceType	可选	body	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
\$model	可选	body	设备的型号。
\$swVersion	可选	body	设备的软件版本。 Z-Wave：主版本号.次版本号，如：1.1。
\$fwVersion	可选	body	设备的固件版本。
\$hwVersion	可选	body	设备的硬件版本。
\$protocolType	可选	body	设备使用的协议类型，当前支持的协议类型：CoAP，huaweiM2M，Z-Wave，ONVIF，WPS，Hue，WiFi，J808，Gateway，ZigBee，LWM2M。

参数	必选/可选	位置	描述
\$bridgeId	可选	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
\$status	可选	body	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
\$statusDetail	可选	body	设备的状态详情，status条件取值，取值查看 <a href="#">\$status</a> 和 <a href="#">\$statusDetail</a> 。
\$mute	可选	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
\$supportedSecurity	可选	body	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
\$isSecurity	可选	body	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
\$signalStrength	可选	body	设备的信号强度。
\$sigVersion	可选	body	设备的sig版本。
\$serialNumber	可选	body	设备的序列号。
\$batteryLevel	可选	body	设备的电池电量。

\$status和\$statusDetail:

\$status	\$statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceInfoChanged",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturer": "wulian",  
        "type": 90,  
        "model": "90",  
        "mac": "*****",  
        "swVersion": "...",  
        "fwVersion": "...",  
        "hwVersion": "...",  
        "protocolType": "zigbee",  
        "description": "smock detector"  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.4 设备数据变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备数据变化通知后（订阅的通知类型为 deviceDataChanged），当设备上报单个服务属性数据时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备数据变化通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承[PushMessageReceiver](#)类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleDeviceDataChanged(NotifyDeviceDataChangedDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyDeviceDataChangedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“deviceDataChanged”。
\$requestId	可选	body	消息的序列号，唯一标识该消息。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$gatewayId	必选	body	网关ID，用于唯一标识一个网关设备。
\$service	必选	body	设备的服务数据，具体参见下表 <a href="#">DeviceService</a> 结构体说明。

DeviceService结构体说明：

参数	必选/可选	位置	描述
\$serviceId	必选	body	服务ID。
\$serviceType	必选	body	服务的类型。
\$data	必选	body	服务数据信息。
\$eventTime	必选	body	时间发生时间，时间格式 yyyymmddThhmissZ，例如 “20151212T121212Z”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDataChanged",  
    "requestId": "*****",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "service": {  
        "serviceId": "Brightness",  
        "serviceType": "Brightness",  
        "data": {  
            "brightness": 80  
        },  
        "eventTime": "20170311T163657Z"  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.5 批量设备数据变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备数据批量变化通知后（订阅的通知类型为 deviceDatasChanged），当设备同时上报多个服务属性数据时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备数据批量变化通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承[PushMessageReceiver](#)类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleDeviceDatasChanged(NotifyDeviceDatasChangedDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyDeviceDatasChangedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“deviceDatasChanged”。
\$requestId	必选	body	消息的序列号，唯一标识该消息。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$gatewayId	必选	body	网关ID，用于唯一标识一个网关设备。
\$services	必选	body	服务列表信息，具体参见 <a href="#">DeviceService</a> 结构体说明。

DeviceService结构体说明:

参数	必选/可选	位置	描述
\$serviceId	必选	body	服务ID。
\$serviceType	必选	body	服务类型。
\$data	必选	body	服务数据信息。
\$eventTime	必选	body	事件上报时间, 时间格式: yyyyymmddThhmissZ, 例如“20151212T121212Z”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDatasChanged",  
    "requestId": "*****",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "service": [  
        {"  
            "serviceId": "Brightness",  
            "serviceType": "Brightness",  
            "data": {  
                "brightness": 80  
            },  
            "eventTime": "20170311T163657Z"  
        },  
        {  
    }
```

```
"serviceId": "Color",
"serviceType": "Color",
"data": {
  "value": "red"
},
"eventTime": "20170311T163657Z"
}]}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 3.3.5.6 设备服务信息变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备服务信息变化通知后（订阅的通知类型为 serviceInfoChanged），当平台向设备下发命令修改设备服务信息时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备服务信息变化通知的第三方应用推送通知消息。

#### 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleServiceInfoChanged(NotifyServiceInfoChangedDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyServiceInfoChangedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型, 取值: “serviceInfoChanged”。
\$deviceId	必选	body	设备ID, 用于唯一标识一个设备。
\$gatewayId	必选	body	网关ID, 用于唯一标识一个网关设备。
\$serviceId	必选	body	设备服务标识。
\$serviceType	必选	body	设备服务类型。
\$serviceInfo	必选	body	设备服务信息, 增量上报, 具体参见 <a href="#">ServiceInfo</a> 结构体。

ServiceInfo结构体说明:

参数	必选/可选	位置	描述
\$muteCmds	可选	body	屏蔽的设备控制命令列表。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "serviceInfoChanged",  
    "deviceId": "*****",  
    "serviceId": "*****",  
    "serviceType": "*****",  
    "gatewayId": "*****",  
    "serviceInfo":  
    {  
        "muteCmds": "VIDEO_RECORD"
```

```
    }  
}  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 3.3.5.7 删除设备通知

#### 典型场景

第三方应用在物联网平台订阅了删除设备通知后（订阅的通知类型为deviceDeleted），当设备在物联网平台中被删除后，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了删除设备通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleDeviceDeleted(NotifyDeviceDeletedDTO body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyDeviceDeletedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“deviceDeleted”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$gatewayId	必选	body	网关ID，用于唯一标识一个网关设备。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDeleted",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.8 设备消息确认通知

## 典型场景

第三方应用在物联网平台订阅了设备消息确认通知后（订阅的通知类型为 messageConfirm），当平台向设备下发命令，并收到设备返回的命令确认消息（如命令已送达、已执行等）时，平台会向第三方应用推送通知消息。

## 接口功能

支持物联网平台向订阅了设备消息确认通知的第三方应用推送通知消息。

## 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
------	--

回调接口	function handleMessageConfirm(NotifyMessageConfirmDTO \$body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyMessageConfirmDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“messageConfirm”。
\$header	必选	body	具体参见下表 <b>MessageConfirmHeader</b> 结构说明。
\$body	必选	body	根据业务具体定义，确认消息可以携带的状态变化等消息。

MessageConfirmHeader结构说明：

参数	必选/可选	位置	描述
\$requestId	必选	body	消息的序列号，唯一标识该消息。
\$from	必选	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● 设备发起的请求：“/devices/{deviceId}”</li><li>● 设备服务发起的请求：“/devices/{deviceId}/services/{serviceId}”</li></ul>
\$to	必选	body	表示消息接收者的地址，To就是订阅请求中的From，如第三方应用的userId。
\$status	必选	body	命令状态。 <ul style="list-style-type: none"><li>● “sent”：已发送</li><li>● “delivered”：已送达</li><li>● “executed”：已执行</li></ul>
\$timestamp	必选	body	时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。

## 响应参数

Status Code: 200 OK

## 请求示例

```
Method: POST
request: {callbackUrl}
Header:
Content-Type:application/json
Body:
{
  "notifyType": "messageConfirm",
  "header": {
    "requestId": "*****",
    "from": "*****",
    "to": "*****",
    "status": "delivered",
    "timestamp": "20151212T121212Z"
  },
  "body": {}
}
```

## 响应示例

```
response:
Status Code: 200 OK
```

### 3.3.5.9 设备命令响应通知

#### 典型场景

第三方应用在物联网平台订阅了设备命令响应通知后（订阅的通知类型为 commandRsp），当平台向设备下发命令，并收到设备返回的命令响应消息（如命令执行成功或者执行失败，设备要向平台返回指定内容响应）时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备命令响应通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server 与 port 为应用服务器公网地址与服务器配置中指定的端口。

2. 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleCommandRsp(NotifyCommandRspDTO \$body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyCommandRspDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“commandRsp”。
\$header	必选	body	具体参见下表 <b>CommandRspHeader</b> 结构说明。
\$body	必选	body	响应命令的消息内容。

CommandRspHeader结构说明：

参数	必选/可选	位置	描述
\$requestId	必选	body	消息的序列号，唯一标识该消息。
\$from	必选	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● 设备发起的请求：“/devices/{deviceId}”</li><li>● 设备服务发起的请求：“/devices/{deviceId}/services/{serviceId}”</li></ul>
\$to	必选	body	表示消息接收者的地址，To就是订阅请求中的From，如第三方应用的userId。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$serviceType	必选	body	命令所属的服务类型。
\$method	必选	body	存放的响应命令，如：“INVITE-RSP”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "commandRsp",  
    "header": {  
        "requestId": "*****",  
        "from": "*****",  
        "to": "*****",  
        "deviceId": "*****",  
        "serviceType": "Camera",  
        "method": "MUTE_COMMANDS"  
    },  
    "body": {}  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.10 设备事件通知

#### 典型场景

第三方应用在物联网平台订阅了设备事件通知后（订阅的通知类型为deviceEvent），当平台收到设备上报的事件消息时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备事件通知的第三方应用推送通知消息。

## 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleDeviceEvent(NotifyDeviceEventDTO \$body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceEventDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“deviceEvent”。
\$header	必选	body	具体参见下表 <b>DeviceEventHeader</b> 结构说明。
\$body	必选	body	响应命令的消息内容。

DeviceEventHeader结构说明：

参数	必选/可选	位置	描述
\$eventType	必选	body	事件类型。
\$from	必选	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>设备发起的请求：“/devices/{deviceId}”</li><li>设备服务发起的请求：“/devices/{deviceId}/services/{serviceId}”</li></ul>
\$timestamp	必选	body	时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。
\$eventTime	必选	body	事件上报时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。

参数	必选/可选	位置	描述
\$serviceType	必选	body	命令所属的服务类型。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceEvent",  
    "header": {  
        "eventType": "*****",  
        "from": "/devices/{deviceId}/services/{serviceId}",  
        "deviceId": "*****",  
        "serviceType": "*****",  
        "timestamp": "20151212T121212Z",  
        "eventTime": "20151212T121212Z",  
    },  
    "body": {  
        "usedPercent": 80  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.11 增加设备模型通知

#### 典型场景

第三方应用在物联网平台订阅了设备模型增加通知后（订阅的通知类型为 deviceModelAdded），在物联网平台上新增设备Profile文件时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备模型增加通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承[PushMessageReceiver](#)类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleDeviceModelAdded(NotifyDeviceModelAddedDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyDeviceModelAddedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“deviceModelAdded”。
\$appId	必选	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
\$deviceType	必选	body	设备的类型。
\$manufacturerName	必选	body	增加设备模型的操作者名称。
\$manufacturerId	必选	body	增加设备模型的操作者ID。
\$model	必选	body	设备型号。

参数	必选/可选	位置	描述
\$protocolType	必选	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceModelAdded",  
    "appId": "*****",  
    "deviceType": "*****",  
    "manufacturerName": "wulian",  
    "manufacturerId": "*****",  
    "model": "*****",  
    "protocolType": "zigbee"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.12 删除设备模型通知

## 典型场景

第三方应用在物联网平台订阅了设备模型删除通知后（订阅的通知类型为deviceModelDeleted），在物联网平台上删除设备Profile文件时，平台会向第三方应用推送通知消息。

## 接口功能

支持物联网平台向订阅了设备模型删除通知的第三方应用推送通知消息。

## 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleDeviceModelDeleted(NotifyDeviceModelDeletedDTO body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceModelDeletedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“deviceModelDeleted”。
\$appId	必选	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
\$deviceType	必选	body	设备的类型。
\$manufacturerName	必选	body	增加设备模型的操作者名称。
\$manufacturerId	必选	body	增加设备模型的操作者ID。
\$model	必选	body	设备型号。
\$protocolType	必选	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceModelAdded",  
    "appId": "*****",  
    "deviceTyp": "*****",  
    "manufacturerName": "*****",  
    "manufacturerId": "*****",  
    "model": "*****",  
    "protocolType": "*****"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.13 设备影子状态变更通知

#### 典型场景

第三方应用在物联网平台订阅了设备影子状态变更通知后（订阅的通知类型为 deviceDesiredPropertiesModifyStatusChanged），当平台中的设备影子向设备同步数据成功或失败时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备影子状态变更通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承[PushMessageReceiver](#)类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	funciton handleDeviceDesiredStatusChanged(NotifyDeviceDesiredStatusChangedDTO \$body)
接口所属类	PushMessageReceiver

## 参数说明

NotifyDeviceDesiredStatusChangedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值： “deviceDesiredPropertiesModifyStatusChanged”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$serviceId	必选	body	服务ID，用于标识一个服务。
\$properties	必选	body	设备影子数据属性
\$status	必选	body	状态，取值“DELIVERED/FAILED”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{
  "notifyType": "deviceDesiredPropertiesModifyStatusChanged",
  "deviceId": "*****",
  "serviceId": "Device",
  "properties": {
```

```
"Model Number" : 1,  
"Serial Number" : 2,  
"Firmware Version" : "v1.1.0"  
},  
"status":"DELIVERED"  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 3.3.5.14 软件升级状态变更通知

#### 典型场景

第三方应用在物联网平台订阅了软件升级状态变更通知后（订阅的通知类型为 swUpgradeStateChangeNotify），当设备进行软件升级，升级任务的状态发生变化时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了软件升级状态变更通知的第三方应用推送通知消息。

#### 注意事项

- 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过继承[PushMessageReceiver](#)类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考[IoT平台北向API参考](#)中的消息推送章节。

#### 接口描述

回调地址	<a href="https://server:port/v1.0.0/messageReceiver">https://server:port/v1.0.0/messageReceiver</a>
回调接口	function handleSwUpgradeStateChanged(NotifySwUpgradeStateChangedDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifySwUpgradeStateChangedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“swUpgradeStateChangeNotify”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$appId	必选	body	设备所属应用的应用ID。
\$operationId	必选	body	软件升级任务ID。
\$subOperationId	必选	body	软件升级子任务ID。
\$swUpgradeState	必选	body	软件升级状态。 <ul style="list-style-type: none"><li>● “downloading”：设备正在下载软件包</li><li>● “downloaded”：设备下载软件包完成</li><li>● “updating”：设备正在进行升级</li><li>● “idle”：设备处于空闲状态</li></ul>

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "swUpgradeStateChangeNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "swUpgradeState": "downloading"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.15 软件升级结果变更通知

#### 典型场景

第三方应用在物联网平台订阅了软件升级结果通知后（订阅的通知类型为 swUpgradeResultNotify），当设备的软件升级任务执行完成时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了软件升级结果通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleSwUpgradeResult(NotifySwUpgradeResultDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifySwUpgradeResultDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“swUpgradeResultNotify”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$appId	必选	body	设备所属应用的应用ID。
\$operationId	必选	body	软件升级任务ID。
\$subOperationId	必选	body	软件升级子任务ID。

参数	必选/可选	位置	描述
\$curVersion	必选	body	设备当前的软件版本。
\$targetVersion	必选	body	设备要升级的目标软件版本。
\$sourceVersion	必选	body	设备的源软件版本。
\$swUpgradeResult	必选	body	软件升级结果。 ● “SUCCESS”：设备升级成功 ● “FAIL”：设备升级失败
\$upgradeTime	必选	body	升级时长。
\$resultDesc	必选	body	升级结果描述。
\$errorCode	必选	body	设备上报的状态错误码。
\$description	必选	body	错误原因描述。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "swUpgradeResultNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "curVersion": "1.3",  
    "targetVersion": "1.5",  
    "sourceVersion": "1.0",  
    "swUpgradeResult": "SUCCESS",  
    "upgradeTime": "***",  
}
```

```
"resultDesc":"***",
"errorCode":"***",
"description":"***",
}
```

## 响应示例

```
response:
Status Code: 200 OK
```

### 3.3.5.16 固件升级状态变更通知

#### 典型场景

第三方应用在物联网平台订阅了固件升级状态变更通知后（订阅的通知类型为 fwUpgradeStateChangeNotify），当设备进行固件升级，升级任务的状态发生变化时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了固件升级状态变更通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleFwUpgradeStateChanged(NotifyFwUpgradeStateChangedDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyFwUpgradeStateChangedDTO

参数	必选/可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“fwUpgradeStateChangeNotify”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$appId	必选	body	设备所属应用的应用ID。
\$operationId	必选	body	固件升级任务ID。
\$subOperationId	必选	body	固件升级子任务ID。
\$step	必选	body	固件升级状态，可取值为“0”、“1”、“2”、“3”。
\$stepDesc	必选	body	升级状态描述。 <ul style="list-style-type: none"><li>● “downloading”：设备正在下载软件包</li><li>● “downloaded”：设备下载软件包完成</li><li>● “updating”：设备正在进行升级</li><li>● “idle”：设备处于空闲状态</li></ul>

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "fwUpgradeStateChangeNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "step": "1",  
    "stepDesc": "downloading"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 3.3.5.17 固件升级结果变更通知

#### 典型场景

第三方应用在物联网平台订阅了固件升级结果通知后（订阅的通知类型为 fwUpgradeResultNotify），当设备的固件升级任务执行完成时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了固件升级结果通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	function handleFwUpgradeResult(NotifyFwUpgradeResultDTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyFwUpgradeResultDTO

参数	必选/ 可选	位置	描述
\$notifyType	必选	body	通知类型，取值：“fwUpgradeResultNotify”。
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$appId	必选	body	设备所属应用的应用ID。
\$operationId	必选	body	固件升级任务ID。
\$subOperationId	必选	body	固件升级子任务ID。

参数	必选/ 可选	位置	描述
\$curVersion	必选	body	设备当前的固件版本。
\$targetVersion	必选	body	设备要升级的目标固件版本。
\$sourceVersion	必选	body	设备的源固件版本。
\$Status	必选	body	升级结果。 <ul style="list-style-type: none"><li>● “SUCCESS”</li><li>● “FAIL”</li></ul>
\$statusDesc	必选	body	升级结果描述。 <ul style="list-style-type: none"><li>● “SUCCESS”：设备升级成功</li><li>● “FAIL”：设备升级失败</li></ul>
\$upgradeTime	必选	body	固件升级时长。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "fwUpgradeResultNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "curVersion": "1.6",  
    "targetVersion": "1.6",  
    "sourceVersion": "1.3",  
    "status": "SUCCESS",  
    "statusDesc": "****",
```

```
"upgradeTime":"****"  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 3.3.5.18 NB 设备命令状态变化通知

#### 典型场景

第三方应用通过物联网平台创建设备命令时，如果设置了callbackurl回调地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达），平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台在命令状态发生变化时向第三方应用推送通知消息。

#### 注意事项

- 应用在[创建设备命令](#)需要设置回调地址为接口描述中的回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过继承**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver/cmd
回调接口	function handleNBCommandStateChanged(NotifyNBCommandStatusChanged DTO \$body)
接口所属类	PushMessageReceiver

#### 参数说明

NotifyNBCommandStatusChangedDTO

参数	必选/可选	位置	描述
\$deviceId	必选	body	设备ID，用于唯一标识一个设备。
\$commandId	必选	body	命令ID，在创建设备命令时由平台生成的。

参数	必选/可选	位置	描述
\$result	必选	body	详见下表 <b>NBCommandResult</b> 结构说明

NBCommandResult结构说明

参数	必选/可选	位置	描述
\$resultCode	必选	body	命令状态结果 <ul style="list-style-type: none"><li>● “SENT”， 平台已下发给设备，但还未收到设备响应。</li><li>● “DELIVERED”， 平台收到设备响应，命令已到达设备。</li><li>● “SUCCESS”， 平台收到命令结果上报，且上报的结果是成功。</li><li>● “FAIL”， 平台收到命令结果上报，且上报的结果是失败。</li></ul>
\$resultDetail	必选	body	设备上报命令结果上报时，携带的自定义字段

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "deviceId": "*****",  
    "commandId": "*****",  
    "result": {  
        "resultCode": "DELIVERED",  
        "resultDetail": null  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

## 3.3.6 命令下发（NB 命令）

### 3.3.6.1 创建设备命令

#### 典型场景

设备Profile文件中定义了物联网平台可向设备下发的命令，第三方应用可调用此接口向设备下发命令，配置或修改设备的服务属性，以实现对设备的控制。

物联网平台有两种命令下发机制：

- 立即下发：平台收到命令后立即下发给设备，保证实时性，不保证串行性。
- 缓存下发：平台收到命令后缓存起来，在设备可达时排队串行下发，平台收到前一个命令的响应（模组自动回复的ACK）才会下发后一个命令，保证串行性，不保证实时性。

#### 接口功能

支持第三方应用向设备下发命令，实现对设备的控制。支持物联网平台立即下发命令和缓存下发命令。

#### 接口描述

```
public function postDeviceCommand($pdcInDTO, $appId, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$pdcInDTO	必选	body	见下表 <b>PostDeviceCommandInDTO</b> 的描述。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### PostDeviceCommandInDTO

参数	必选/可选	位置	描述
\$deviceId	必选	body	下发命令的设备ID，用于唯一标识一个设备。

参数	必选/可选	位置	描述
\$command	必选	body	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
\$callbackUrl	可选	body	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
\$expireTime	可选	body	下发命令的有效时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。 如果expireTime设置为0，则无论物联网平台上设置的命令下发模式是什么，该命令都会立即下发给设备（如果设备处于眠状态或者链路已老化，则设备收不到命令，平台没收到设备的响应，该命令最终会超时）。
\$maxRetransmit	可选	body	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	位置	描述
\$serviceId	必选	body	命令对应的服务ID，用于标识一个服务。要与profile中定义的serviceId保持一致。
\$method	必选	body	命令服务下具体的命令名称，服务属性等。要与profile中定义的命令名保持一致。
\$paras	必选	ObjectNode	body 命令参数， jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。 若profile中命令未定义参数，则paras值中的参数可不填写，即设置为"paras": {}。

## 响应参数

PostDeviceCommandOutDTO

参数	描述
\$commandId	设备命令ID。
\$appId	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
\$deviceId	下发命令的设备ID，用于唯一标识一个设备。
\$command	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
\$callbackUrl	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
\$expireTime	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
\$status	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
\$result	下发命令执行的详细结果。
\$creationTime	命令的创建时间。
\$executeTime	命令执行的时间。
\$platformIssuedTime	平台发送命令的时间。
\$deliveredTime	平台将命令送达设备的时间。
\$issuedTimes	平台发送命令的次数。
\$maxRetransmit	命令下发最大重传次数。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	100223	Command counts has reached the upLimit.	缓存的命令数已到达限制。处于PENDING状态的命令条数不超过限定期。默认限定期为20。 处理建议：如当前平台缓存的命令还需执行，请让设备上报数据触发缓存命令的下发；如当前平台缓存的命令有不需要执行的命令，请调用“修改设备命令V4”接口，将命令的状态从PENDING修改为CANCELED。

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	100612	Device is zombie.	设备为僵尸设备。（距离设备上次上线时间超过阈值，默认7天） 处理建议：请把设备上线后再重新下发命令。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 3.3.6.2 查询设备命令

#### 典型场景

第三方应用向设备下发命令后，可调用此接口在物联网平台查询下发命令的状态及内容信息，以了解命令的执行情况。

#### 接口功能

支持第三方应用在物联网平台查询下发命令的状态和内容信息，可查询指定时间段内本应用下的所有下发命令，或者指定设备的所有下发命令。

#### 接口描述

```
public function queryDeviceCommand($qdcInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$qdcInDTO	必选	query	见下表 <b>QueryDeviceCommandInDTO</b> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### QueryDeviceCommandInDTO

参数	必选/可选	位置	描述
\$pageNo	可选	query	查询的页码，大于等于0，默认值：“0”。
\$pageSize	可选	query	查询每页信息的数量，大于等于1，最大值1000，默认值：“1000”。
\$deviceId	可选	query	指定查询命令的设备ID。
\$startTime	可选	query	查询开始时间，查询下发命令时间在 startTime之后的记录。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$endTime	可选	query	查询结束时间，查询下发命令时间在 endTime之前的记录。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$appId	可选	query	如果是本应用的设备命令，此参数值可以填写null，否则填写授权应用的appId。

## 响应参数

QueryDeviceCommandOutDTO

参数	描述
\$pagination	页面信息，具体参见 <a href="#">Pagination</a> 结构体。
\$data	设备命令列表，具体参见 <a href="#">DeviceCommandRespV4</a> 结构体。

Pagination结构体说明：

参数	描述
\$pageNo	页码。
\$pageSize	每页信息数量。
\$totalSize	记录总数。

DeviceCommandRespV4结构体说明：

参数	描述
\$commandId	设备命令ID。
\$appId	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
\$deviceId	下发命令的设备ID，用于唯一标识一个设备。
\$command	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
\$callbackUrl	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
\$expireTime	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。

参数	描述
\$status	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
\$result	下发命令执行的详细结果。
\$creationTime	命令的创建时间。
\$executeTime	命令执行的时间。
\$platformIssuedTime	平台发送命令的时间。
\$deliveredTime	平台将命令送达到设备的时间。
\$issuedTimes	平台发送命令的次数。
\$maxRetransmit	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	描述
\$serviceId	必选	命令对应的服务ID，用于标识一个服务。
\$method	必选	命令服务下具体的命令名称，服务属性等。
\$paras	可选	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	<p>应用不存在。 处理建议：</p> <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的 appId是否有误。</li> <li>● 请检查请求路径（url）中传入的appId是否有误。</li> </ul>

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数 appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>请检查该设备的Profile文件是否已上传到物联网平台。</li><li>请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议： <ul style="list-style-type: none"><li>请确认startTime和endTime都不为空，且endTime比startTime晚。</li><li>请确认pageNo不为空，且pageNo大于0。</li><li>请检查pageSize不为空，且pageSize大于1。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 3.3.6.3 修改设备命令

#### 典型场景

第三方应用向设备下发命令后，如果由于命令排队、设备不在线等原因，物联网平台还未把命令下发给设备执行，此时第三方应用可调用此接口修改该命令的状态信息。当前仅支持把命令状态修改为CANCELED状态，即撤销命令的执行。

#### 接口功能

支持第三方应用修改指定命令的状态信息，仅能修改还处于PENDING状态的命令，且当前仅支持修改的命令状态为CANCELED，即撤销命令。

#### 接口描述

```
public function updateDeviceCommand($udcInDTO, $deviceCommandId, $appId, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$udcInDTO	必选	body	见下表 <a href="#">UpdateDeviceCommandInDTO</a> 的描述。
\$deviceCommandId	必选	path	要修改的命令ID，在调用创建设备命令接口后获得。
\$appId	必选	query	如果是本应用的设备命令，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### UpdateDeviceCommandInDTO

参数	必选/可选	描述
\$status	必选	命令执行结果，可选值：“CANCELED”，撤销命令。

## 响应参数

### UpdateDeviceCommandOutDTO

参数	描述
\$commandId	设备命令ID。
\$appId	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
\$deviceId	下发命令的设备ID，用于唯一标识一个设备。
\$command	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
\$callbackUrl	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
\$expireTime	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。

参数	描述
\$status	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
\$result	下发命令执行的详细结果。
\$creationTime	命令的创建时间。
\$executeTime	命令执行的时间。
\$platformIssuedTime	平台发送命令的时间。
\$deliveredTime	平台将命令送达到设备的时间。
\$issuedTimes	平台发送命令的次数。
\$maxRetransmit	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/ 可选	位置	描述
\$serviceId	必选	body	命令对应的服务ID，用于标识一个服务。
\$method	必选	body	命令服务下具体的命令名称，服务属性等。
\$paras	可选	body	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	<p>应用不存在。 处理建议：<ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul></p>

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>请检查该设备的Profile文件是否已上传到物联网平台。</li><li>请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
200	100434	The device command is not existed.	设备命令不存在。 处理建议：请检查接口请求中的设备命令ID是否正确。
200	100435	The device command already canceled, expired or executed, Cannot cancel.	设备命令已经取消，过期或执行，不能取消。
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 3.3.6.4 创建设备命令撤销任务

#### 典型场景

第三方应用向设备下发命令后，如果由于命令排队、设备不在线等原因，物联网平台还未把命令下发给设备执行（DEFAULT状态），此时第三方应用可调用此接口撤销指定设备的所有未下发的命令，对于已下发成功的命令不可撤销。

#### 接口功能

支持第三方应用创建设备命令撤销任务，该任务用于撤销物联网平台中指定设备ID下所有未下发的命令（命令处于DEFAULT状态）。

#### 接口描述

```
public function createDeviceCmdCancelTask($cdccInDTO, $appId, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$cdctInDTO	必选	body	见下表 <a href="#">CreateDeviceCmdCancelTaskInDTO</a> 的描述。
\$appId	必选	query	如果是本应用的设备命令，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### CreateDeviceCmdCancelTaskInDTO

参数	必选/可选	位置	描述
\$deviceId	必选	body	待撤销设备命令的设备ID，撤销任务将会撤销所有向该设备下发的命令。

## 响应参数

### CreateDeviceCmdCancelTaskOutDTO

参数	描述
\$taskId	设备命令撤销任务的任务ID。
\$appId	设备命令撤销任务所属的应用ID。
\$deviceId	执行设备命令撤销任务的设备ID。
\$status	撤销任务的任务状态。 <ul style="list-style-type: none"><li>● “WAITTING” 表示等待执行中</li><li>● “RUNNING” 表示撤销任务正在执行</li><li>● “SUCCESS” 表示撤销任务执行成功</li><li>● “FAILED” 表示撤销任务执行失败</li><li>● “PART_SUCCESS” 表示撤销任务部分执行成功。</li></ul>
\$totalCount	撤销的设备命令总数。
\$deviceCommands	撤销的设备命令列表，具体参见 <a href="#">DeviceCommandRespV4</a> 结构体。

DeviceCommandRespV4结构体说明：

参数	描述
\$commandId	设备命令ID。
\$appId	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
\$deviceId	下发命令的设备ID，用于唯一标识一个设备。
\$command	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
\$callbackUrl	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
\$expireTime	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
\$status	下发命令的状态。 <ul style="list-style-type: none"> <li>● “DEFAULT” 表示未下发</li> <li>● “EXPIRED” 表示命令已经过期</li> <li>● “SUCCESSFUL” 表示命令已经成功执行</li> <li>● “FAILED” 表示命令执行失败</li> <li>● “TIMEOUT” 表示命令下发执行超时</li> <li>● “CANCELED” 表示命令已经被撤销执行</li> </ul>
\$result	下发命令执行的详细结果。
\$creationTime	命令的创建时间。
\$executeTime	命令执行的时间。
\$platformIssuedTime	平台发送命令的时间。
\$deliveredTime	平台将命令送达到设备的时间。
\$issuedTimes	平台发送命令的次数。
\$maxRetransmit	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	描述
\$serviceId	必选	命令对应的服务ID，用于标识一个服务。
\$method	必选	命令服务下具体的命令名称，服务属性等。

参数	必选/可选	描述
\$paras	可选	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 3.3.6.5 查询设备命令撤销任务

#### 典型场景

第三方应用创建了设备命令撤销任务后，可调用此接口查询设备命令撤销任务的详细信息和执行状态，以了解撤销任务的执行情况。

#### 接口功能

支持第三方应用在物联网平台查询设备命令撤销任务信息和状态，可指定条件查询单个或多个撤销任务。

#### 接口描述

```
public function queryDeviceCmdCancelTask($qdcctInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$qdcctInDTO	必选	query	见下表 <a href="#">QueryDeviceCmdCancelTask...</a> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## QueryDeviceCmdCancelTaskInDTO

参数	必选/可选	位置	描述
\$pageNo	可选	query	查询的页码，大于等于0，默认值：“0”。
\$pageSize	可选	query	查询每页信息的数量，大于等于1，最大值1000，默认值：“1000”。
\$taskId	可选	query	撤销任务的任务ID。
\$deviceId	可选	query	执行设备命令撤销任务的设备ID。
\$status	可选	query	设备命令撤销任务的状态。
\$startTime	可选	query	查询开始时间，查询创建撤销设备命令任务时间在 startTime之后的记录。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$endTime	可选	query	查询结束时间，查询创建撤销设备命令任务时间在 endTime之前的记录。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$appId	可选	query	如果是本应用的设备命令，此参数值可以填写null，否则填写授权应用的appId。

## 响应参数

## QueryDeviceCmdCancelTaskOutDTO

参数	描述
\$pagination	页码信息，具体参见 <a href="#">Pagination</a> 结构体。
\$data	设备命令列表，具体参见 <a href="#">CreateDeviceCmdCancelTaskOutDTO</a> 结构体。

Pagination结构体说明：

参数	描述
\$pageNo	查询结果页码。
\$pageSize	每页信息数量。
\$totalSize	记录总数，即查询到的撤销任务中的命令总数。

CreateDeviceCmdCancelTaskOutDTO结构体说明：

参数	描述
\$taskId	设备命令撤销任务的任务ID。
\$appId	设备命令撤销任务所属的应用ID。
\$deviceId	设备命令撤销任务指定撤销命令的设备ID。
\$status	撤销任务的任务状态。 <ul style="list-style-type: none"><li>● “WAITTING” 表示等待执行中</li><li>● “RUNNING” 表示撤销任务正在执行</li><li>● “SUCCESS” 表示撤销任务执行成功</li><li>● “FAILED” 表示撤销任务执行失败</li><li>● “PART_SUCCESS” 表示撤销任务部分执行成功。</li></ul>
\$totalCount	撤销的设备命令总数。
\$deviceCommands	设备命令撤销任务信息列表，具体参见 <a href="#">DeviceCommandRespV4</a> 结构体。

DeviceCommandRespV4结构体说明：

参数	描述
\$commandId	设备命令ID。
\$appId	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
\$deviceId	下发命令的设备ID，用于唯一标识一个设备。
\$command	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
\$callbackUrl	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
\$expireTime	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。

参数	描述
\$status	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
\$result	下发命令执行的详细结果。
\$creationTime	命令的创建时间。
\$executeTime	命令执行的时间。
\$platformIssuedTime	平台发送命令的时间。
\$deliveredTime	平台将命令送达到设备的时间。
\$issuedTimes	平台发送命令的次数。
\$maxRetransmit	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	描述
\$serviceId	必选	命令对应的服务ID，用于标识一个服务。
\$method	必选	命令服务下具体的命令名称，服务属性等。
\$paras	可选	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	<p>应用不存在。 处理建议：</p> <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径（url）中传入的appId是否有误。</li> </ul>

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。

Http状态码	错误码	错误描述	说明
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

## 3.3.7 命令下发（非 NB 命令）

### 3.3.7.1 设备服务调用

#### 典型场景

设备Profile文件中定义了物联网平台可向设备下发的命令，第三方应用可调用此接口向设备下发命令，配置或修改设备的服务属性，以实现对设备的控制。

物联网平台不对设备命令进行缓存而是直接下发命令，设备不在线时命令下发失败。下发命令的具体格式需要第三方应用与设备自定义，物联网平台在接口中直接进行封装，透传。

#### 说明

目前该接口仅用于安装了IoT Agent/AgentLite的网关设备，通过向网关设备下发命令，可对网关下的非直连设备进行控制。

#### 接口功能

支持第三方应用向设备立即下发命令（当前仅支持安装了IoT Agent/AgentLite的网关设备），实现对设备的控制，支持给本应用的设备下发命令。

#### 接口描述

```
public function invokeDeviceService($deviceId, $serviceId, $commandDTO, $appId, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$deviceId	必选	path	设备ID，用于唯一标识一个设备。
\$serviceId	必选	path	服务ID，唯一标识一个服务。
\$commandDTO	必选	body	见下表 <b>CommandDTO</b> 的描述。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

## CommandDTO

参数	必选/可选	位置	描述
\$header	必选	body	具体参见 <a href="#">CommandNA2CloudHeader</a> 结构体。
\$body	可选	body	消息的消息体。JsonObject里面是一个个键值对。每个键都是profile中命令的参数名（paraName）。

CommandNA2CloudHeader结构体说明：

参数	必选/可选	位置	描述
\$requestId	可选	body	用于标识一个命令，不能重复。
\$mode	必选	body	是否要确认消息。 <ul style="list-style-type: none"><li>● “NOACK”：不需要确认消息</li><li>● “ACK”：需要确认消息</li><li>● 其它值无效</li></ul>
\$from	可选	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● App发起的请求：“/users/{userId}”</li><li>● 第三方应用发起的请求：“/ {serviceName}”</li><li>● 物联网平台发起的请求：“/cloud/ {serviceName}”</li></ul>
\$toType	可选	body	消息接受者的类型，“CLOUD” / “GATEWAY”。
\$to	可选	body	消息接收者的地址。
\$method	必选	body	命令名称，如：“DISCOVERY”为发现非直连设备，“REMOVE”为删除非直连设备。
\$callbackURL	可选	body	命令的回调路径。

## 响应参数

## InvokeDeviceServiceOutDTO

参数	描述
\$status	命令状态。 <ul style="list-style-type: none"><li>● “sent”：已发送</li><li>● “delivered”：已投递，toType为CLOUD的请求才有该返回值</li><li>● “failed”：失败，toType为CLOUD的请求才有该返回值</li></ul>
\$timestamp	发送命令的时间戳，时间格式: yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$requestId	设备命令ID。 <ul style="list-style-type: none"><li>● toType为“GATEWAY”时，如果请求有requestId则同请求的requestId一致，如果请求无requestId则平台分配一个序列号。</li><li>● toType为“CLOUD”时，则为空。</li></ul>

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与网关的连接是否正常。

Http状态码	错误码	错误描述	说明
200	100432	The device command is muted.	设备命令已被禁用。 处理建议：请检查接口请求参数method中的命令是否有误。
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	102203	CommandName is invalid.	命令名称无效。 处理建议：请检查接口请求参数method中携带的命令是否有误。
403	100450	The gateway is not online.	网关不在线。 处理建议：请检查网关与物联网平台的连接是否正常。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100444	The serviceType is not exist.	服务类型不存在。 处理建议：请检查接口请求参数toType中的服务类型是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	50252	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

## 3.3.8 数据采集

物联网平台支持第三方应用查询设备的基本信息，同时还支持查看设备上报的历史数据，能按时、天、月等维度查看设备上报的历史数据。

### 3.3.8.1 查询单个设备信息

#### 典型场景

已在物联网平台注册的设备，第三方应用若需要查看某个设备的详细信息（如厂商、型号、版本、状态和服务属性等信息），可调用此接口在物联网平台查询获取。

#### 接口功能

支持第三方应用在物联网平台根据设备ID查询指定设备的详细信息，包括配置信息、状态信息、服务属性信息等。

#### 接口描述

```
public function querySingleDeviceInfo($deviceId, $select, $appId, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$deviceId	必选	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
\$select	可选	query	指定查询条件，可选值：“imsi”。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

QuerySingleDeviceInfoOutDTO

参数	描述
\$deviceId	设备ID，用于唯一标识一个设备。
\$gatewayId	网关ID，用于标识一个网关设备。
\$nodeType	节点类型，取值：“ENDPOINT” / “GATEWAY” / “UNKNOW”。
\$createTime	创建设备的时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$lastModifiedTime	最后修改设备的时间。

参数	描述
\$deviceInfo	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。
\$services	设备服务列表，具体参见 <a href="#">DeviceService</a> 结构体。

DeviceInfo结构体说明：

参数	描述
\$nodeId	设备的唯一标识。
\$name	设备名称。
\$description	设备的描述信息。
\$manufacturerId	厂商ID，唯一标识一个厂商。
\$manufacturerName	厂商名称。
\$mac	设备的MAC地址。
\$location	设备的位置信息。
\$deviceType	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
\$model	设备的型号。 Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
\$swVersion	设备的软件版本，。 Z-Wave: 主版本号.次版本号，如：1.1。
\$fwVersion	设备的固件版本。
\$hwVersion	设备的硬件版本。
\$protocolType	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
\$bridgeId	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
\$status	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
\$statusDetail	设备的状态详情，status条件取值，取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。

参数	描述
\$mute	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
\$supportedSecurity	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
\$isSecurity	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
\$signalStrength	设备的信号强度。
\$sigVersion	设备的sig版本。
\$serialNumber	设备的序列号。
\$batteryLevel	设备的电池电量。

\$status和\$statusDetail:

\$status	\$statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceService结构体说明：

参数	描述
\$serviceId	设备的服务标识。
\$serviceType	设备的服务类型。
\$serviceInfo	设备的服务信息，具体参见 <a href="#">ServiceInfo</a> 结构体。
\$data	属性值对。

参数	描述
\$eventTime	时间格式: yyyyMMdd'T'HHmmss'Z', 如 “20151212T121212Z”。

ServiceInfo结构体说明:

参数	描述
\$muteCmds	屏蔽的设备控制命令列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100405	The request parameter is invalid.	请求参数不合法。 处理建议: 请参考请求参数说明检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁, 超过流控值(默认值是100次/60s)。 处理建议: 联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议: 请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议: 物联网平台未找到deviceId对应的设备, 请检查deviceId是否有误。

Http状态码	错误码	错误描述	说明
404	100418	The deviceData is not existed.	<p>设备信息不存在。 处理建议：</p> <ul style="list-style-type: none"> <li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li> <li>● Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li> <li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li> </ul>
500	100203	The application is not existed.	<p>应用不存在。 处理建议：</p> <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径（url）中传入的appId是否有误。</li> </ul>
500	50252	Internal server error.	<p>服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。</p>

### 3.3.8.2 批量查询设备信息列表

#### 典型场景

已在物联网平台注册的设备，第三方应用若需要根据条件查看多个设备的详细信息（如厂商、型号、版本、状态和服务属性等信息），可调用此接口在物联网平台查询获取。

#### 接口功能

支持第三方应用在物联网平台根据条件查询多个设备的详细信息，包括配置信息、状态信息、服务属性信息等。

#### 接口描述

```
public function queryBatchDevicesInfo($qbdiInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$qbdiInDTO	必选	query	见下表 <a href="#">PostDeviceCommandInDTO</a> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## PostDeviceCommandInDTO

参数	必选/ 可选	位置	描述
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$gatewayId	可选	query	网关ID，用于标识一个网关设备。
\$nodeType	可选	query	节点类型，取值：“ENDPOINT” / “GATEWAY” / “UNKNOW”。
\$deviceType	可选	query	设备类型。
\$pageNo	可选	query	查询的页码。 <ul style="list-style-type: none"><li>● 值为空时查询内容不分页</li><li>● 值大于等于0的整数时分页查询</li><li>● 值等于0时查询第一页</li></ul>
\$pageSize	可选	query	查询每页信息的数量，缺省值：“1”。
\$status	可选	query	查询设备的状态。 <ul style="list-style-type: none"><li>● “ONLINE”：在线</li><li>● “OFFLINE”：不在线</li><li>● “ABNORMAL”：异常状态</li></ul>
\$startTime	可选	query	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	query	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	query	指定返回记录的排序。 <ul style="list-style-type: none"><li>● “ASC”：按注册设备的时间升序排列</li><li>● “DESC”：按注册设备的时间降序排列</li></ul> 缺省值：“DESC”。
\$select	可选	query	指定返回记录，可取值：“imsi”。

## 响应参数

## QueryBatchDevicesInfoOutDTO

参数	描述
\$totalCount	查询的记录数量。
\$pageNo	查询的页码。
\$pageSize	查询每页信息的数量。
\$devices	设备分页列表信息，具体参见 <a href="#">QuerySingleDeviceInfoOutDTO</a> 结构体。

QuerySingleDeviceInfoOutDTO结构体说明：

参数	描述
\$deviceId	设备ID，用于唯一标识一个设备。
\$gatewayId	网关ID，用于标识一个网关设备。
\$nodeType	节点类型，取值：ENDPOINT/GATEWAY/UNKNOW。
\$createTime	创建设备的时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$lastModifiedTime	最后修改设备的时间。
\$deviceInfo	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。
\$services	设备服务列表，具体参见 <a href="#">DeviceService</a> 结构体。

DeviceInfo结构体说明：

参数	描述
\$nodeId	标识设备的唯一ID。
\$name	设备名称。
\$description	设备的描述信息。
\$manufacturerId	厂商ID，唯一标识一个厂商。
\$manufacturerName	厂商名称。
\$mac	设备的MAC地址。
\$location	设备的位置信息。
\$deviceType	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。

参数	描述
\$model	设备的型号。 Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐, 如: 001A-0A12, 其他协议的格式待定。
\$swVersion	设备的软件版本，。 Z-Wave: 主版本号.次版本号, 如: 1.1。
\$fwVersion	设备的固件版本。
\$hwVersion	设备的硬件版本。
\$protocolType	设备使用的协议类型, 当前支持的协议类型: CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
\$bridgeId	Bridge标识, 表示设备通过哪个Bridge接入物联网平台。
\$status	设备的状态, 表示设备是否在线, 取值范围: “ONLINE”、 “OFFLINE”、 “INBOX”、 “ABNORMAL”。
\$statusDetail	设备的状态详情, status条件取值, 取值查看 <a href="#">\$status</a> 和 <a href="#">\$statusDetail</a> 。
\$mute	表示设备是否处于冻结状态, 即设备上报数据时, 平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE” : 冻结状态</li><li>● “FALSE” : 非冻结状态</li></ul>
\$supportedSecurity	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE” : 支持安全模式</li><li>● “FALSE” : 不支持安全模式</li></ul>
\$isSecurity	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE” : 启用</li><li>● “FALSE” : 未启用</li></ul>
\$signalStrength	设备的信号强度。
\$sigVersion	设备的sig版本。
\$serialNumber	设备的序列号。
\$batteryLevel	设备的电池电量。

\$status和\$statusDetail:

\$status	\$statusDetail
OFFLINE	NONE CONFIGURATION_PENDING

\$status	\$statusDetail
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

 **说明**

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceService结构体说明：

参数	描述
\$serviceId	设备的服务标识。
\$serviceType	设备的服务类型。
\$serviceInfo	设备的服务信息，具体参见 <a href="#">ServiceInfo</a> 结构体。
\$data	属性值对（Attribute-value pair）。
\$eventTime	时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

ServiceInfo结构体说明：

参数	描述
\$muteCmds	屏蔽的设备控制命令列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。
400	100218	The gatewayId and pageNo can't be both null.	网关ID和pageNo不能同时为空。 处理建议：请检查请求参数中 gatewayId或pageNo是否填写。

Http状态码	错误码	错误描述	说明
400	100405	The request parameter is invalid.	请求参数不合法。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景， 请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.8.3 查询设备历史数据

#### 典型场景

在日常运行中，物联网平台会收到并保存设备上报业务数据（设备数据的保存时间可通过“修改设备信息”接口配置，最长保存90天），第三方应用若需要查看某个设备上报到平台的历史数据，可调用此接口查询获取。

## 接口功能

支持第三方应用根据设备ID，查询指定设备上报到物联网平台的历史数据。

## 接口描述

```
public function queryDeviceDataHistory($qddhInDTO, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$qddhInDTO	必选	query	见下表 <a href="#">QuerySingleDeviceInfoOutDTO</a> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

QueryDeviceDataHistoryInDTO

参数	必选/可选	位置	描述
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$deviceId	必选	query	设备ID，用于唯一标识一个设备。
\$gatewayId	必选	query	网关ID，用于标识一个网关设备。
\$serviceId	可选	query	设备的服务标识。
\$property	可选	query	服务属性数据。
\$pageNo	可选	query	查询的页码。 <ul style="list-style-type: none"><li>● 值为空时查询内容不分页</li><li>● 值大于等于0的整数时分页查询</li><li>● 值等于0时查询第一页</li></ul>
\$pageSize	可选	query	查询每页信息的数量，缺省值：“1”。
\$startTime	可选	query	查询产生时间在startTime之后的历史数据。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$endTime	可选	query	查询产生时间在endTime之前的历史数据。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

## 响应参数

QueryDeviceDataHistoryOutDTO

参数	描述
\$totalCount	查询的记录数量。
\$pageNo	查询的页码。
\$pageSize	查询每页信息的数量。
\$deviceDataHistoryDTOs	设备历史数据列表，具体参见 <a href="#">DeviceDataHistoryDTO</a> 结构体。

DeviceDataHistoryDTO结构体说明：

参数	描述
\$serviceId	设备的服务标识。
\$deviceId	设备ID，用于唯一标识一个设备。
\$gatewayId	网关ID，用于标识一个网关设备。
\$appId	第三方应用的身份标识，用于唯一标识一个应用。
\$data	设备上报的数据。
\$timestamp	上报数据的时间戳，时间格式：yyyyMMdd'THHmmss'Z'，如：“20151212T121212Z”。

## 错误码

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。例如，pageSize是否超过2000。

Http状态码	错误码	错误描述	说明
400	100419	The deviceId and gatewayId can't be both null.	deviceId和gatewayId不能同时为空。 处理建议：请检查请求参数中deviceId和gatewayId是否填写。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.8.4 查询设备影子历史数据

#### 典型场景

第三方应用通过“修改设备影子”接口修改设备影子的配置时，物联网平台会保存修改记录。当第三方应用需要查看设备影子的历史配置记录时，可调用此接口查询获取。

#### 接口功能

支持第三方应用根据设备ID，在物联网平台查询指定设备影子的历史配置数据。

## 接口描述

```
public function queryDeviceDesiredHistory($qddhInDTO, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$qddhInDTO	必选	query	见下表 <b>QueryDeviceDesiredHistoryInDTO</b> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### QueryDeviceDesiredHistoryInDTO

参数	必选/可选	位置	描述
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$deviceId	必选	query	设备ID，用于唯一标识一个设备。
\$gatewayId	必选	query	网关ID，用于标识一个网关设备。
\$serviceId	可选	query	设备的服务标识。
\$property	可选	query	服务属性数据。
\$pageNo	可选	query	查询的页码。 <ul style="list-style-type: none"><li>● 值为空时查询内容不分页</li><li>● 值大于等于0的整数时分页查询</li><li>● 值等于0时查询第一页</li></ul>
\$pageSize	可选	query	查询每页信息的数量，缺省值：“1”。
\$startTime	可选	query	查询产生时间在startTime之后的历史数据。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
\$endTime	可选	query	查询产生时间在endTime之前的历史数据。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

## 响应参数

### QueryDeviceDesiredHistoryOutDTO

参数	描述
\$totalCount	查询的记录数量。
\$pageNo	查询的页码。
\$pageSize	查询每页信息的数量。
\$DeviceDesiredHistoryDTO	设备历史数据列表，具体参见 <a href="#">DeviceDesiredHistoryDTO</a> 结构体。

DeviceDesiredHistoryDTO结构体说明：

参数	描述
\$serviceId	设备的服务标识。
\$deviceId	设备ID，用于唯一标识一个设备。
\$gatewayId	网关ID，用于标识一个网关设备。
\$appId	第三方应用的身份标识，用于唯一标识一个应用。
\$desired	设备上报的数据。
\$timestamp	上报数据的时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的 appId是否有误。</li><li>请检查请求路径（url）中传入的 appId是否有误。</li></ul>
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。
400	100419	The deviceId and gatewayId can't be both null.	deviceId和gatewayId不能同时为空。 处理建议：请检查请求参数中 deviceId和gatewayId是否填写。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.8.5 查询设备服务能力

#### 典型场景

第三方应用若需要了解设备可上报哪些服务属性数据，以及设备支持下发哪些命令，可调用此接口在物联网平台查询设备的Profile文件中定义的设备服务能力信息。

#### 接口功能

支持第三方应用在物联网平台查询设备的服务属性、设备命令等服务能力信息。

#### 接口描述

```
public function queryDeviceCapabilities($qdcInDTO, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$qdcInDTO	必选	query	见下表 <b>QueryDeviceCapabilitiesInDTO</b> 的描述。
\$accessToken	必选	header	填写3.1.1-鉴权接口获取的accessToken。

### QueryDeviceCapabilitiesInDTO

参数	是否必须	位置	描述
\$gatewayId	可选	query	网关ID，用于标识一个网关设备。
\$appId	必选	query	如果是本应用的设备，此参数值可以填写null，否则填写授权应用的appId。
\$deviceId	可选	query	设备ID，用于唯一标识一个设备。

## 响应参数

### QueryDeviceCapabilitiesOutDTO

参数	描述
\$deviceCapabilities	查询结果列表，具体参见 <b>DeviceCapabilityDTO</b> 结构体。

DeviceCapabilityDTO结构体说明：

参数	描述
\$deviceId	设备ID，用于唯一标识一个设备。
\$serviceCapabilities	设备的服务能力列表，具体参见 <b>ServiceCapabilityDTO</b> 结构体。

ServiceCapabilityDTO结构体说明：

参数	描述
\$serviceId	设备的服务标识。
\$serviceType	设备的服务类型。
\$option	服务选项。

参数	描述
\$description	设备服务描述信息。
\$commands	支持的命令名称列表，具体参见 <a href="#">ServiceCommand</a> 结构体。
\$properties	支持的属性名称列表，具体参见 <a href="#">ServiceProperty</a> 结构体。

ServiceCommand结构体说明：

参数	描述
\$commandName	命令名称。
\$paras	属性列表，具体参见 <a href="#">ServiceCommandPara</a> 结构体。
\$responses	响应列表，具体参见 <a href="#">ServiceCommandResponse</a> 结构体。

ServiceCommandPara结构体说明：

参数	描述
\$paraName	参数名称。
\$dataType	数据类型。
\$required	是否必选。
\$min	属性最小值。
\$max	属性最大值。
\$step	步长。
\$maxLength	最大长度。
\$unit	单位（符号）。
\$enumList	枚举类型列表。

ServiceCommandResponse结构体说明：

参数	描述
\$responseName	响应名称。
\$paras	属性列表，具体参见 <a href="#">ServiceCommandPara</a> 结构体。

ServiceProperty结构体说明：

参数	描述
\$propertyName	属性名称。
\$dataType	数据类型。
\$required	是否必选。
\$min	属性最小值。
\$max	属性最大值。
\$step	步长。
\$maxLength	最大长度。
\$method	访问方法。 <ul style="list-style-type: none"><li>● “R”：可读</li><li>● “W”：可写</li><li>● “E”：可观察</li></ul>
\$unit	单位（符号）。
\$enumList	枚举类型列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 3.3.9 设备组管理

### 3.3.9.1 创建设备组

#### 典型场景

第三方应用可调用此接口在物联网平台上创建设备组，并把设备归类到不同的设备组内，进行分组管理。一个设备可以归属到多个设备组内。

在对设备进行某些操作时（如升级设备软固件、批量下发命令等），可通过设备组来指定要进行操作的设备。

#### 接口功能

支持第三方应用在物联网平台创建设备组，用于对设备进行分组管理。

#### 接口描述

```
public function createDeviceGroup($cdgInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$cdgInDTO	必选	body	见下表 <a href="#">CreateDeviceGroupInDTO</a> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## CreateDeviceGroupInDTO

参数	必选/可选	位置	描述
\$name	必选	body	设备组名称，仅限大小写字母和数字。
\$description	可选	body	设备组的描述信息。
\$appId	可选	body	如果是本应用的设备，此参数值可以填写 null，否则填写授权应用的appId。
\$maxDevNum	可选	body	设备组设备最大数量，默认最小值为0。当值为0时，表示对设备数量不做限制。
\$deviceIds	可选	body	添加到设备组的设备ID列表。
id	可选	body	设备组ID。

## 响应参数

StatusCode: 200 ok

## CreateDeviceGroupOutDTO

参数	描述
\$name	设备组名称，仅限大小写字母和数字。
\$description	设备组的描述信息。
\$id	设备组ID，由平台自动生成。
\$appId	第三方应用的身份标识，用于唯一标识一个应用。
\$maxDevNum	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。
curDevNum	当前设备组内的设备数量。
deviceIds	添加到设备组的设备ID列表。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100602	The device group name has been used.	设备组名字已经存在。 处理建议：请修改接口请求中的设备组名称。
200	100607	The devGroup has reached the limit.	设备组数目达到限制。 处理建议：请检查已创建的设备组数量是否已达到License限定的数量上限。
400	100609	Too much devices to add.	添加太多设备至设备组。 处理建议：请确认deviceIds中的设备ID数量在maxDevNum设置值的范围内。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.9.2 删除设备组

#### 典型场景

若因分组变更，第三方应用不再需要使用某个设备组，且不想在物联网平台上继续保存该设备组信息时，可调用此接口在物联网平台删除指定设备组。

#### 接口功能

支持第三方应用在物联网平台根据设备组ID删除指定设备组。

## 接口描述

```
public function deleteDeviceGroup($devGroupId, $accessAppId, $accessToken)
```

## 参数说明

参数	必选/ 可选	位置	描述
\$devGroupId	必选	path	设备组ID，在增加设备组后由物联网平台返回获得。
\$accessAppId	可选	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

Status Code: 200 ok

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100603	The device group is not existed	设备组不存在。 处理建议：请检查设备组ID是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.9.3 修改设备组

#### 典型场景

若因业务变更需要修改设备组的信息，如设备组名称、设备组的设备数量限制等，第三方应用可调用此接口修改指定设备组的信息。

#### 接口功能

支持第三方应用在物联网平台修改指定设备组的信息。

#### 接口描述

```
public function modifyDeviceGroup($mdgInDTO, $devGroupId, $accessAppId, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$mdgInDTO	必选	body	见下表 <b>ModifyDeviceGroupInDTO</b> 的描述。
\$devGroupId	必选	path	设备组ID，在增加设备组后由物联网平台返回获得。
\$accessAppId	必选	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### ModifyDeviceGroupInDTO

参数	必选/可选	位置	描述
\$name	必选	body	设备组名称，仅限大小写字母和数字。
\$description	可选	body	设备组的描述信息。
\$maxDevNum	可选	body	设备组设备最大数量，默认值为0。当值为0时，表示对设备数量不做限制。

## 响应参数

ModifyDeviceGroupOutDTO

参数	描述
\$name	设备组名称，仅限大小写字母和数字。
\$description	设备组的描述信息。
\$id	设备组ID。
\$appId	设备组所属的应用ID。
\$maxDevNum	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。
\$curDevNum	当前设备组内设备数量。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100602	The device group name has been used.	设备组的名字已经存在。 处理建议：请修改接口请求中的设备组名称。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.9.4 查询设备组详情

#### 典型场景

第三方应用可调用此接口查询当前已创建的所有设备组列表信息，以了解当前设备组的分组和使用情况。

#### 接口功能

支持第三方应用在物联网平台查询已创建的所有设备组信息。

#### 接口描述

```
public function queryDeviceGroups($qdgInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$qdgInDTO	必选	query	见下表 <b>QueryDeviceGroupsInDTO</b> 的描述。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### QueryDeviceGroupsInDTO

参数	必选/可选	位置	描述
\$accessAppId	可选	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。

参数	必选/可选	位置	描述
\$pageNo	可选	query	分页查询参数。 <ul style="list-style-type: none"><li>● 值为空时，查询内容不分页。</li><li>● 值为大于等于0的整数时，分页查询。</li><li>● 值为0时查询第一页。</li></ul>
\$pageSize	可选	query	每页设备组记录数量，默认值为“1”。
\$name	可选	query	设备组名称。

## 响应参数

QueryDeviceGroupsOutDTO

参数	描述
\$totalCount	设备组总数。
\$pageNo	查询结果页码。
\$pageSize	每页设备组记录数量。
\$list	设备组信息详情，具体参见 <a href="#">QuerySingleDeviceGroupOutDTO</a> 结构体。

QuerySingleDeviceGroupOutDTO结构体说明：

参数	描述
\$name	设备组名称，仅限大小写字母和数字。
\$description	设备组的描述信息。
\$id	设备组ID。
\$appId	设备组所属的应用ID。
\$maxDevNum	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。
\$curDevNum	当前设备组内的设备数量。
\$creator	创建设备组的用户名称。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.9.5 查询指定设备组

#### 典型场景

第三方应用若需要查看某个设备组的信息，以了解该设备组的使用情况，可调用此接口查询指定设备组的信息。

#### 接口功能

支持第三方应用在物联网平台根据设备组ID查询指定设备组的信息。

#### 接口描述

```
public function querySingleDeviceGroup($devGroupId, $accessAppId, $accessToken)
```

## 参数说明

参数	必选/ 可选	位置	描述
\$devGroupId	必选	path	设备组ID，在增加设备组后由物联网平台返回获得。
\$accessAppId	可选	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

QuerySingleDeviceGroupOutDTO

参数	描述
\$name	设备组名称，仅限大小写字母和数字。
\$description	设备组的描述信息。
\$id	设备组ID。
\$appId	设备组所属的应用ID。
\$maxDevNum	设备组设备最大数量限制。
\$curDevNum	当前设备组内的设备数量。
\$creator	创建设备组的用户名称。

## 错误码

Http状态 码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.9.6 查询指定设备组成员

#### 典型场景

第三方应用若需要查看某个设备组中的设备成员分布情况，可调用此接口查询指定设备组的设备成员列表信息。

#### 接口功能

支持第三方应用在物联网平台根据设备组ID查询指定设备组内设备列表信息。

#### 接口描述

```
public function queryDeviceGroupMembers($qdgmInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$qdgmInDTO	必选	query	见下表 <a href="#">QueryDeviceGroupMembersInDTO</a> 的 描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

[QueryDeviceGroupMembersInDTO](#)

参数	必选/可选	位置	描述
\$devGroupId	必选	query	设备组ID，在增加设备组后由物联网平台返回获得。
\$accessAppId	可选	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
\$pageNo	可选	query	分页查询参数，默认值为“0”。 <ul style="list-style-type: none"><li>● 值为空时，查询内容不分页。</li><li>● 值为大于等于0的整数时，分页查询。</li><li>● 值为0时查询第一页。</li></ul>
\$pageSize	可选	query	每页设备记录数量，默认值为“10”。

## 响应参数

QueryDeviceGroupMembersOutDTO

参数	描述
\$totalCount	设备组内设备总数。
\$pageNo	查询结果页码。
\$pageSize	每页设备记录数量。
\$deviceIds	设备组内设备ID列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	107001	The serviceId is not exist.	服务ID不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107002	The properties is empty in database.	设备属性不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107003	The request properties is unknown.	设备状态为未知。 处理建议：请检查设备与平台的连接是否正常。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁, 超过流控值(默认值是100次/60s)。 处理建议: 联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议: 请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议: <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议: 物联网平台内部错误, 请联系物联网平台维护人员处理。

### 3.3.9.7 增加设备组成员

#### 典型场景

若需要把新增的设备或已有设备增加到某个设备组中, 第三方应用可调用此接口向指定设备组添加设备成员。在向设备组添加设备前, 建议通过“查询指定设备组”接口查询该设备组的当前设备数量及最大设备数量限制, 确保设备组中还有足够的可添加成员数量。

#### 接口功能

支持第三方应用在物联网平台向指定设备组添加设备。

## 接口描述

```
public function addDevicesToGroup($dgwdlDTO, $accessAppId, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$dgwdlDTO	必选	body	见下表 <b>DeviceGroupWithDeviceListDTO</b> 的描述。
\$accessAppId	可选	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

DeviceGroupWithDeviceListDTO

参数	必选/可选	位置	描述
\$devGroupId	必选	body	设备组ID，在增加设备组后由物联网平台返回获得。
\$deviceIds	必选	body	要添加到设备组的设备ID列表。

## 返回参数

DeviceGroupWithDeviceListDTO

参数	描述
\$devGroupId	设备组ID。
\$deviceIds	添加到设备组的设备ID列表。

## 错误码

Http状态码	错误码	错误描述	说明
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。

Http状态码	错误码	错误描述	说明
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议： <ul style="list-style-type: none"><li>● 请检查接口请求中的设备ID是否有误。</li><li>● 请检查设备组是否达到最大设备数量限制。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.9.8 删除设备组成员

#### 典型场景

若设备组中的一个或多个设备不再归属于该设备组，第三方应用可调用此接口从设备组删除设备成员。

#### 接口功能

支持第三方应用在物联网平台从指定设备组删除设备。

#### 接口描述

```
public function deleteDevicesFromGroup($dgwdlDTO, $accessAppId, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$dgwdlDTO	必选	body	见下表 <b>DeviceGroupWithDeviceListDTO</b> 的描述。
\$accessAppId	可选	query	如果是本应用的设备组，此参数值可以填写null，否则填写授权应用的appId。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### DeviceGroupWithDeviceListDTO

参数	必选/可选	位置	描述
\$devGroupId	必选	body	设备组ID，在增加设备组后由物联网平台返回获得。
\$deviceIds	必选	body	要从设备组删除的设备ID列表。

#### 响应参数

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议： <ul style="list-style-type: none"><li>● 请检查接口请求中的设备ID是否有误。</li><li>● 请检查设备组是否达到最大设备数量限制。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 3.3.10 设备升级

#### 3.3.10.1 查询版本包列表

##### 典型场景

在进行设备版本升级前，第三方应用可调用此接口查询已经上传到物联网平台的版本升级包列表信息，确保设备要升级的版本包已上传到平台。

##### 接口功能

支持第三方应用在物联网平台查询已上传的版本包列表信息，可根据条件查询满足要求的版本包。

##### 接口描述

```
public function queryUpgradePackageList($uplInDTO, $accessToken)
```

##### 参数说明

参数	必选/可选	位置	描述
\$uplInDTO	必选	query	见下表 <b>QueryUpgradePackageListInDTO</b> 的描述。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

QueryUpgradePackageListInDTO

参数	必选/可选	位置	描述
\$fileType	可选	query	版本包类型。 ● “firmwarePackage”： 固件包 ● “softwarePackage”： 软件包
\$deviceType	可选	query	版本包适用的设备类型。
\$model	可选	query	版本包适用的设备型号。
\$manufacturerName	可选	query	版本包适用的设备厂商名称。
\$version	可选	query	版本包的版本号。
\$pageNo	可选	query	分页查询参数， 默认值“0”。 ● 值为空时， 查询内容不分页。 ● 值为大于等于0的整数时， 分页查询。 ● 值为0时查询第一页。
\$pageSize	可选	query	查询结果分页时的每页结果数量， 取值范围1-100， 默认值“10”。

## 响应参数

QueryUpgradePackageListOutDTO

参数	描述
\$data	版本包列表信息， 具体参见 <a href="#">QueryUpgradePackageOutDTO</a> 结构体。
\$pageNo	查询结果的页码。
\$pageSize	查询结果每页的记录数量。
\$totalCount	查询结果的总记录数。

QueryUpgradePackageOutDTO结构体说明：

参数	描述
\$fileId	版本包ID。
\$name	版本包名称。
\$version	版本包版本号。

参数	描述
\$fileType	版本包文件类型。 <ul style="list-style-type: none"><li>● “firmwarePackage”： 固件包</li><li>● “softwarePackage”： 软件包</li></ul>
\$deviceType	版本包适用的设备类型。
\$model	版本包适用的设备型号。
\$manufacturerName	版本包适用的设备厂商名称。
\$protocolType	版本包适用的设备协议类型
\$description	版本包的描述信息。
\$date	版本包的生成时间。
\$uploadTime	版本包的上传日期。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中pageNo和pageSize的值是否在正常范围内。
400	123029	pageNo or pageSize beyond the limit.	pageNo或pageSize超出上限。 处理建议：修改pageNo或pageSize为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 3.3.10.2 查询指定版本包

#### 典型场景

在进行设备版本升级前，第三方应用可调用此接口在物联网平台查询某个版本升级包的信息，确保设备要升级的版本包已上传到平台。

## 接口功能

支持第三方应用在物联网平台根据版本包ID查询指定版本包信息，在调用该接口前需要通过“[查询版本包列表](#)”接口查询获得版本包ID。

## 接口描述

```
public function queryUpgradePackage($fileId, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$fileId	必选	path	版本包ID，在上传版本包后获得。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

QueryUpgradePackageOutDTO

参数	描述
\$fileId	版本包ID。
\$name	版本包名称。
\$version	版本包的版本号。
\$fileType	版本包类型。 <ul style="list-style-type: none"><li>● “firmwarePackage”： 固件包</li><li>● “softwarePackage”： 软件包</li></ul>
\$deviceType	版本包适用的设备类型。
\$model	版本包适用的设备型号。
\$manufacturerName	版本包适用的设备厂商名称。
\$protocolType	版本包适用的设备协议类型。
\$description	版本包的描述信息。
\$date	版本包的生成时间。
\$uploadTime	版本包的上传日期。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	123002	Device or package file not found.	设备或包不存在。 处理建议：请检查接口请求中的 fileId 是否正确。

### 3.3.10.3 删除指定版本包

#### 典型场景

对于不需要继续使用和保留的设备版本包，第三方应用可调用此接口在物联网平台上删除指定的版本包。

#### 接口功能

支持第三方应用在物联网平台根据版本包ID删除指定的版本包文件，在调用该接口前需要通过“查询版本包列表”接口查询获得要删除

#### 接口描述

```
public function deleteUpgradePackage($fileId, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$fileId	必选	path	版本包ID，在上传版本包后获得。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

response:

Status Code: 200 OK

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken 是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 3.3.10.4 创建软件升级任务

#### 典型场景

若需要对设备进行软件版本升级，第三方应用可调用此接口为多个设备创建软件升级任务，升级前请确保目标版本包已上传到物联网平台。当前仅支持对NB-IoT设备进行软件版本升级。

#### 接口功能

支持第三方应用在物联网平台为多个设备进行软件版本升级，当前仅支持对NB-IoT设备进行软件版本升级。

#### 接口描述

```
public function createSoftwareUpgradeTask($cutInDTO, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$cutInDTO	必选	body	见下表 <a href="#">CreateUpgradeTaskInDTO</a> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### CreateUpgradeTaskInDTO

参数	必选/可选	位置	描述
\$fileId	必选	body	要升级的目标版本包ID。
\$targets	必选	body	要进行升级的目标，具体参见 <a href="#">OperateDevices</a> 结构体。
\$policy	可选	body	升级任务的执行策略，具体参见 <a href="#">OperatePolicy</a> 结构体。

### OperateDevices结构体说明：

参数	必选/可选	位置	描述
\$deviceGroups	可选	body	设备组名称列表，最多支持256个设备组。 与指定设备列表方式二选一。
\$deviceType	可选	body	设备类型。 指定设备组时，此字段必选。
\$model	可选	body	设备型号。 指定设备组时，此字段必选。
\$manufacturerName	可选	body	设备厂商名称。 指定设备组时，此字段必选。
\$devices	可选	body	设备ID列表，最多支持256个设备。 与指定设备组方式二选一。

### OperatePolicy结构体说明：

参数	必选/可选	位置	描述
\$executeType	必选	body	执行类型， 默认值为“now”。 <ul style="list-style-type: none"> <li>● “now”：现在执行</li> <li>● “device_online”：设备上线时执行</li> <li>● “custom”：自定义</li> </ul>
\$startTime	可选	body	任务执行时间， executeType=custom时必选， 时间格式： <code>yyyyMMdd'T'HHmmss'Z'</code> , 如： “20151212T121212Z”。
\$endTime	可选	body	任务停止时间， executeType=custom时必选， 时间格式： <code>yyyyMMdd'T'HHmmss'Z'</code> , 如： “20151212T121212Z”。
\$retryType	可选	body	执行失败是否进行重试， 默认不重试。 <ul style="list-style-type: none"> <li>● “true”：重试</li> <li>● “false”：不重试</li> </ul>
\$retryTimes	可选	body	重试次数, 取值范围1-5, retryType=true时必选。

## 响应参数

CreateUpgradeTaskOutDTO

参数	描述
\$operationId	操作任务ID。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查请求参数的是否有误。

Http状态码	错误码	错误描述	说明
400	123016	The parameter is error, targetversion not match with device.	参数不正确，目标版本与指定设备不匹配。 处理建议：请检查接口请求中deviceType、manufacturerName、model，与参数fileId指定的目标版本包信息是否一致。
400	123019	manufacturerName is null.	厂商名称内容为空。 处理建议：请检查接口请求中的manufacturerName是否为空。
400	123020	deviceType is null	设备类型内容为空。 处理建议：请检查接口请求中的deviceType是否为空。
400	123021	model is null.	设备型号内容为空。 处理建议：请检查接口请求中的model是否为空。
400	123022	deviceGroups and devices cannot be null together	deviceGroups和devices不能同时为空。 处理建议：deviceGroups与devices必须要选择一个进行赋值。
400	123023	deviceGroups and devices cannot be exist together	deviceGroups和devices不能同时存在。 处理建议：deviceGroups与devices只能选择一个进行赋值。
400	123024	The number of deviceGroups or devices reached upper limit	deviceGroups或devices内容数量达到上限。 处理建议：请检查deviceGroups或devices的内容，数量不能超过256个。
400	123025	executeType is error or can not to be null.	executeType有误或不能为空。 处理建议：请检查接口请求中的executeType是否为空或者是否有误。
400	123026	startTime or endTime is null or error.	startTime或endTime为空或有误。 处理建议：请检查接口请求中的startTime和endTime是否为空或者格式有误。

Http状态码	错误码	错误描述	说明
400	123028	retryTimes is null or beyond the limit.	retryTimes为空或超出上限。 处理建议：请检查接口请求中的retryTimes是否为空，或者不能小于1或者大于5。
400	123032	startTime can not be later than the endTime.	startTime的时间不能比endTime晚。 处理建议：请检查接口请求中的startTime是否晚于endTime。
400	123033	startTime can not be earlier than the now.	startTime的时间不能比当前时间早。 处理建议：请检查接口请求中的startTime是否早于当前时间。
400	123034	endtime must be greater than 5 minutes.	endtime要比startTime大5分钟。 处理建议：请检查接口请求中的startTime和endTime间隔时间必须大于5分钟。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 3.3.10.5 创建固件升级任务

#### 典型场景

若需要对设备进行固件版本升级，第三方应用可调用此接口为多个设备创建固件升级任务，升级前请确保目标版本包已上传到物联网平台。当前仅支持对NB-IoT设备进行固件版本升级。

## 接口功能

支持第三方应用在物联网平台为多个设备进行固件版本升级，当前仅支持对NB-IoT设备进行固件版本升级。

## 接口描述

```
public function createFirmwareUpgradeTask($cutInDTO, $accessToken)
```

## 参数说明

参数	必选/可选	位置	描述
\$cutInDTO	必选	body	见下表 <b>CreateUpgradeTaskInDTO</b> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### CreateUpgradeTaskInDTO

参数	必选/可选	位置	描述
\$fileId	必选	body	要升级的目标版本包ID。
\$targets	必选	body	要进行升级的目标，具体参见 <b>OperateDevices</b> 结构体。
\$policy	可选	body	升级任务的执行策略，具体参见 <b>OperatePolicy</b> 结构体。

### OperateDevices结构体说明：

参数	必选/可选	位置	描述
\$deviceGroups	可选	body	设备组名称列表，最多支持256个设备组。 与指定设备列表方式二选一。
\$deviceType	可选	body	设备类型。 指定设备组时，此字段必选。
\$model	可选	body	设备型号。 指定设备组时，此字段必选。
\$manufacturerName	可选	body	设备厂商名称。 指定设备组时，此字段必选。

参数	必选/可选	位置	描述
\$devices	可选	body	设备ID列表，最多支持“256”个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明：

参数	必选/可选	位置	描述
\$executeType	必选	body	执行类型，默认值为“now”。 ● “now”：现在执行 ● “device_online”：设备上线时执行 ● “custom”：自定义
\$startTime	可选	body	任务执行时间，executeType=custom时必选，时间格式： yyyyMMdd'T'HHmmss'Z'，如： “20151212T121212Z”。
\$endTime	可选	body	任务停止时间，executeType=custom时必选，时间格式： yyyyMMdd'T'HHmmss'Z'，如： “20151212T121212Z”。
\$retryType	可选	body	执行失败是否进行重试，默认不重试。 ● “true”：重试 ● “false”：不重试
\$retryTimes	可选	body	重试次数，取值范围1-5，retryType=true时必选。

## 响应参数

CreateUpgradeTaskOutDTO

参数	描述
\$operationId	操作任务ID。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查请求参数的是否有误。
400	123016	The parameter is error, targetversion not match with device.	参数不正确，目标版本与指定设备不匹配。 处理建议：请检查接口请求中deviceType、manufacturerName、model，与参数fileId指定的目标版本包信息是否一致。
400	123019	manufacturerName is null.	厂商名称内容为空。 处理建议：请检查接口请求中的manufacturerName是否为空。
400	123020	deviceType is null	设备类型内容为空。 处理建议：请检查接口请求中的deviceType是否为空。
400	123021	model is null.	设备型号内容为空。 处理建议：请检查接口请求中的model是否为空。
400	123022	deviceGroups and devices cannot be null together	deviceGroups和devices不能同时为空。 处理建议：deviceGroups与devices必须要选择一个进行赋值。
400	123023	deviceGroups and devices cannot be exist together	deviceGroups和devices不能同时存在。 处理建议：deviceGroups与devices只能选择一个进行赋值。
400	123024	The number of deviceGroups or devices reached upper limit	deviceGroups或devices内容数量达到上限。 处理建议：请检查deviceGroups或devices的内容，数量不能超过256个。
400	123025	executeType is error or can not to be null.	executeType有误或不能为空。 处理建议：请检查接口请求中的executeType是否为空或者是是否有误。

Http状态码	错误码	错误描述	说明
400	123026	startTime or endTime is null or error.	startTime或endTime为空或者有误。 处理建议：请检查接口请求中的 startTime 和 endTime 是否为空或者格式有误。
400	123028	retryTimes is null or beyond the limit.	retryTimes 为空或超出上限。 处理建议：请检查接口请求中的 retryTimes 是否为空，或者不能小于1或者大于5。
400	123032	startTime can not be later than the endTime.	startTime 的时间不能比 endTime 晚。 处理建议：请检查接口请求中的 startTime 是否晚于 endTime。
400	123033	startTime can not be earlier than the now.	startTime 的时间不能比当前时间早。 处理建议：请检查接口请求中的 startTime 是否早于当前时间。
400	123034	endtime must be greater than 5 minutes.	endtime 要比 startTime 大5分钟。 处理建议：请检查接口请求中的 startTime 和 endTime 间隔时间必须大于5分钟。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的 fileId 是否正确。

### 3.3.10.6 查询指定升级任务结果

#### 典型场景

在创建了设备软件升级任务或固件升级任务后，第三方应用可调用此接口查询某个升级任务详细信息，以查看升级任务的配置信息和执行情况等。

#### 接口功能

支持第三方应用在物联网平台查询单个固件或软件升级任务的详细信息，包括升级任务的配置信息和执行情况等。

#### 接口描述

```
public function queryUpgradeTask($operationId, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$operationId	必选	path	操作任务ID，在创建操作任务后由物联网平台返回获得。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

QueryUpgradeTaskOutDTO

参数	描述
\$operationId	操作任务ID。
\$createTime	操作任务的创建时间。
\$startTime	操作任务的启动时间。
\$stopTime	操作任务的停止时间。
\$operateType	操作类型。 <ul style="list-style-type: none"><li>● “firmware_upgrade”</li><li>● “software_upgrade”</li></ul>
\$targets	执行操作的目标设备，具体参见 <a href="#">OperateDevices</a> 结构体。
\$policy	操作执行策略，具体参见 <a href="#">OperatePolicy</a> 结构体。

参数	描述
\$status	操作任务的状态。 <ul style="list-style-type: none"><li>● “wait”：等待</li><li>● “processing”：正在执行</li><li>● “failed”：失败</li><li>● “success”：成功</li><li>● “stop”：停止</li></ul>
\$staResult	操作结果统计，具体参见 <a href="#">OperationStaResult</a> 结构体。
\$extendPara	操作扩展参数，视不同类型的操作不同，具体参见 <a href="#">extendPara</a> 请求参数。

OperateDevices结构体说明：

参数	描述
\$deviceGroups	设备组名称列表，最多支持“256”个设备组。 与指定设备列表方式二选一。
\$deviceType	设备类型。 指定设备组时，此字段必选。
\$model	设备型号。 指定设备组时，此字段必选。
\$manufacturerName	设备厂商名称。 指定设备组时，此字段必选。
\$devices	设备ID列表，最多支持“256”个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明：

参数	描述
\$executeType	执行类型，默认值为now。 <ul style="list-style-type: none"><li>● now：现在执行</li><li>● device_online：设备上线时执行</li><li>● custom：自定义</li></ul>
\$startTime	任务执行时间，executeType=custom时必选，时间格式： yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。

参数	描述
\$endTime	任务停止时间, executeType=custom时必选, 时间格式: yyyyMMdd'THHmmss'Z', 如: 20151212T121212Z。
\$retryType	执行失败是否进行重试, 默认不重试。 <ul style="list-style-type: none"><li>● true: 重试</li><li>● false: 不重试</li></ul>
\$retryTimes	重试次数, 取值范围1-5, retryType=true时必选。

OperationStaResult结构体说明:

参数	描述
\$total	操作设备总数。
\$wait	等待操作的设备个数。
\$processing	正在操作的设备个数。
\$success	操作成功的设备个数。
\$fail	操作失败的设备个数。
\$stop	停止操作的设备个数。
\$timeout	操作超时失败的设备个数。

对于操作类型为softwareUpgrade和firmwareUpgrade, extendPara请求参数如下:

参数	描述
\$fileVersion	升级的目标版本号。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议: 请检查接口请求中 fileId的格式是否正确。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的operationId内容是否正确。

### 3.3.10.7 查询指定升级任务子任务详情

#### 典型场景

在创建了设备软件升级任务或固件升级任务后，任务中涉及的每个设备的升级是一个子任务（即升级任务中涉及多少个设备，就有多少个子任务）。第三方应用可调用此接口查询某个升级任务中各个子任务的详细信息，以查看子任务的具体执行情况。

#### 接口功能

支持第三方应用在物联网平台查询单个固件或软件升级任务中每个设备的升级执行情况。

#### 接口描述

```
public function queryUpgradeSubTask($qustInDTO, $operationId, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$qustInDTO	必选	query	见下表 <b>QueryUpgradeSubTaskInDTO</b> 的描述。
\$operationId	必选	path	操作任务ID，在创建操作任务后由物联网平台返回获得。
\$accessToken	必选	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### QueryUpgradeSubTaskInDTO

参数	必选/ 可选	位置	描述
\$subOperationStatus	可选	query	<p>子任务状态，不指定，则查询该任务下所有子任务执行详情。</p> <ul style="list-style-type: none"> <li>● “wait”：等待</li> <li>● “processing”：正在执行</li> <li>● “fail”：失败</li> <li>● “success”：成功</li> <li>● “stop”：停止</li> </ul>
\$pageNo	可选	query	<p>分页查询参数，默认值“0”。</p> <ul style="list-style-type: none"> <li>● 值为空时，查询内容不分页。</li> <li>● 值为大于等于0的整数时，分页查询。</li> <li>● 值为0时查询第一页。</li> </ul>
\$pageSize	可选	query	查询结果分页时的每页结果数量，取值范围1-100，默认值“10”。

## 响应参数

QueryUpgradeSubTaskOutDTO

参数	描述
\$data	子任务列表信息，具体参见。
\$pageNo	查询结果的页码。
\$pageSize	查询结果每页的记录数量。
\$totalCount	查询结果的记录总数。

SubOperationInfo结构体说明：

参数	描述
\$subOperationId	子任务ID。
\$createTime	子任务创建时间。
\$startTime	子任务启动时间。
\$stopTime	子任务停止时间。

参数	描述
\$operateType	操作类型。 <ul style="list-style-type: none"><li>● “firmware_upgrade”</li><li>● “software_upgrade”</li></ul>
\$deviceId	操作设备的设备ID。
\$status	子任务状态。 <ul style="list-style-type: none"><li>● “wait”：等待</li><li>● “processing”：正在执行</li><li>● “fail”：失败</li><li>● “success”：成功</li><li>● “stop”：停止</li></ul>
\$detailInfo	任务状态的详细描述，对于失败场景下为失败原因。
\$extendInfo	任务扩展信息，视不同类型的操作不同。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
400	123029	pageNo or pageSize beyond the limit.	pageNo 或 pageSize 超出上限。 处理建议：修改 pageNo 或 pageSize 为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的 token 信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的 operationId 内容是否正确。

### 3.3.10.8 查询升级任务列表

#### 典型场景

第三方应用可调用此接口查询已创建的升级任务列表信息，以了解当前已有的升级任务信息及各个任务的执行情况。

#### 接口功能

支持第三方应用在物联网平台查询升级任务的列表信息，可根据条件查询满足要求的升级任务信息。

#### 接口描述

```
public function queryUpgradeTaskList($qutlInDTO, $accessToken)
```

#### 参数说明

参数	必选/可选	位置	描述
\$qutlInDTO	必选	query	见下表 <b>QueryUpgradeTaskListInDTO</b> 的描述。
\$accessToken	必选	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### QueryUpgradeTaskListInDTO

参数	必选/可选	位置	描述
\$operationType	可选	query	操作类型。 <ul style="list-style-type: none"><li>● “firmware_upgrade”</li><li>● “software_upgrade”</li></ul>
\$operationStatus	可选	query	操作任务的状态。 <ul style="list-style-type: none"><li>● “wait”：等待</li><li>● “processing”：正在执行</li><li>● “failed”：失败</li><li>● “success”：成功</li><li>● “stop”：停止</li></ul>
\$deviceType	可选	query	操作任务针对的设备类型。
\$model	可选	query	操作任务针对的设备型号。
\$manufacturerName	可选	query	操作任务针对的设备厂家名称。
\$deviceId	可选	query	操作任务针对的设备ID。

参数	必选/可选	位置	描述
\$pageNo	可选	query	分页查询参数，默认值“0”。 <ul style="list-style-type: none"><li>● 值为空时，查询内容不分页。</li><li>● 值为大于等于0的整数时，分页查询。</li><li>● 值为0时查询第一页。</li></ul>
\$pageSize	可选	query	查询结果分页时的每页结果数量，取值范围1-100，默认值“10”。

## 响应参数

QueryUpgradeTaskListOutDTO

参数	描述
\$data	任务列表信息，具体参见 <a href="#">OperationInfo</a> 结构体。
\$pageNo	查询结果的页码。
\$pageSize	查询结果每页的记录数量。
\$totalCount	查询结果的总记录数。

OperationInfo结构体说明：

参数	描述
\$operationId	操作任务ID。
\$createTime	操作任务的创建时间。
\$startTime	操作任务的启动时间。
\$stopTime	操作任务的停止时间。
\$operateType	操作类型。 <ul style="list-style-type: none"><li>● “firmware_upgrade”</li><li>● “software_upgrade”</li></ul>
\$targets	执行操作的目标设备，具体参见 <a href="#">OperateDevices</a> 结构体。
\$policy	操作执行策略，具体参见 <a href="#">OperatePolicy</a> 结构体。

参数	描述
\$status	操作任务的状态。 <ul style="list-style-type: none"><li>● “wait”：等待</li><li>● “processing”：正在执行</li><li>● “failed”：失败</li><li>● “success”：成功</li><li>● “stop”：停止</li></ul>
\$staResult	操作结果统计，具体参见 <a href="#">OperationStaResult</a> 结构体。
\$extendPara	操作扩展参数，视不同类型的操作不同。

OperateDevices结构体说明：

参数	描述
\$deviceGroups	设备组名称列表，最多支持“256”个设备组。 与指定设备列表方式二选一。
\$deviceType	设备类型。 指定设备组时，此字段必选。
\$model	设备型号。 指定设备组时，此字段必选。
\$manufacturerName	设备厂商名称。 指定设备组时，此字段必选。
\$devices	设备ID列表，最多支持“256”个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明：

参数	描述
\$executeType	执行类型，默认值为“now”。 <ul style="list-style-type: none"><li>● “now”：现在执行</li><li>● “device_online”：设备上线时执行</li><li>● “custom”：自定义</li></ul>
\$startTime	任务执行时间，executeType=custom时必选，时间格式： yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。
\$endTime	任务停止时间，executeType=custom时必选，时间格式： yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。

参数	描述
\$retryType	执行失败是否进行重试， 默认不重试。 ● “true”：重试 ● “false”：不重试
\$retryTimes	重试次数, 取值范围1-5, retryType=true时必选。

OperationStaResult结构体说明：

参数	描述
\$total	操作设备总数。
\$wait	等待操作的设备个数。
\$processing	正在操作的设备个数。
\$success	操作成功的设备个数。
\$fail	操作失败的设备个数。
\$stop	停止操作的设备个数。
\$timeout	操作超时失败的设备个数。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
400	123029	pageNo or pageSize beyond the limit.	pageNo 或 pageSize 超出上限。 处理建议：修改 pageNo 或 pageSize 为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的 token 信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。

Http状态码	错误码	错误描述	说明
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的operationId内容是否正确。

# 4 IoT 平台北向 Python SDK API 参考

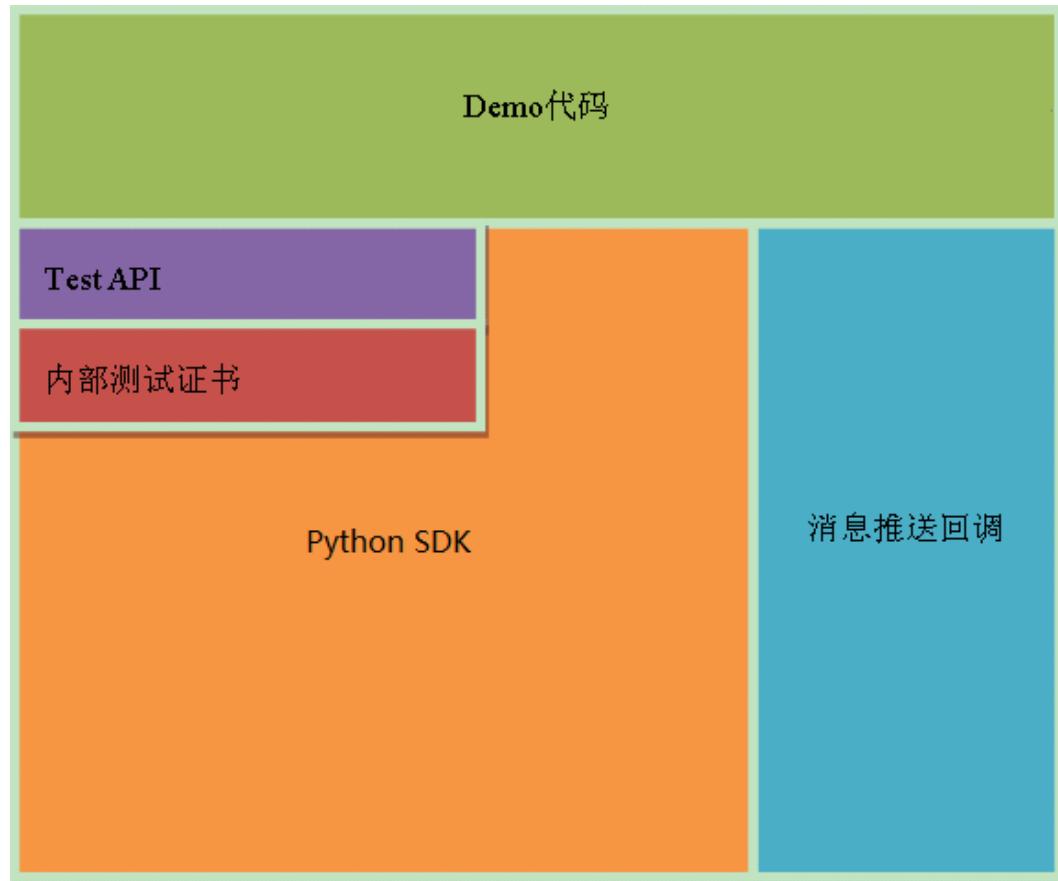
[SDK Demo 总体架构及使用说明](#)

[SDK 初始化配置及测试](#)

[业务接口列表](#)

## 4.1 SDK Demo 总体架构及使用说明

- Demo代码是调用SDK接口的样例代码，包括初始化及各个接口调用，仅供开发者参考。
- SDK提供Python方法调用平台北向Restful接口与平台通信。
- 消息推送回调使用Python代码实现回调接口，供平台推送消息给应用服务器时调用，应用通过继承**PushMessageReceiver**类并重写类中的方法接收推送消息的内容。
- Test API提供测试SDK接口的功能，主要测试SDK与平台之间的基本功能是否可用，并输出测试结果。测试时会分别使用内部测试证书和开发者设置的证书，以便排查证书问题。



## 4.2 SDK 初始化配置及测试

### 4.2.1 NorthApiClient 类

用于创建应用实例，是调用SDK其他接口的前提。主要方法如下：

方法	说明
<code>def invokeAPI(httpMethod, url, headers, payload, clientInfo)</code>	所有接口最终调用的方法，根据传入的参数不同，最后返回的数据不同。

### 4.2.2 Clientinfo 类

用于设置对接的基本信息。主要方法如下：

方法	说明
<code>platformIp</code>	平台的IP。
<code>platformPort</code>	平台的端口，如“8743”。

方法	说明
appId	应用的appId。appId是平台为伙伴服务器分配的ID。
accessToken	鉴权接口获取的accessToken。

## 4.3 业务接口列表

### 4.3.1 应用安全接入

第三方应用获取鉴权信息，接入物联网平台，随后携带鉴权信息调用其他API接口。

#### 4.3.1.1 鉴权

##### 典型场景

第三方应用首次访问物联网平台的开放API时，需调用此接口完成接入认证；第三方应用在物联网平台的认证过期后，需调用此接口重新进行认证，才能继续访问物联网平台的开放API。

##### 接口功能

支持第三方应用在首次访问物联网平台的开放API之前，完成在物联网平台的接入认证。

##### 注意事项

鉴权接口是调用其他API的前提，除了调用鉴权接口（Auth），其他接口调用都需要使用鉴权接口获取的accessToken。

如果多次获取accessToken，则之前的accessToken失效，后一次获取的accessToken才有效。请勿并发获取accessToken。

##### 接口描述

```
def getAuthToken(self, clientInfo)
```

##### 接口所属类

Authentication

##### 参数说明

参数	必选/可选	类型	位置	描述
clientInfo	必选	ClientInfo	-	见下表ClientInfo的描述。

ClientInfo

参数	必选/可选	类型	位置	描述
platformIp	必选	String(256)	path	设置平台的IP。
platformPort	必选	String(256)	path	设置平台的端口，如“8743”。
appId	必选	String(256)	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。appId在物联网平台的SP Portal上创建应用时获得。
secret	必选	String(256)	body	第三方应用的密码，与appId对应，用于登录访问物联网平台。secret在物联网平台的SP Portal上创建应用时获得。

## 返回值

AuthOutDTO

参数	类型	描述
scope	String(256)	申请权限范围，即accessToken所能访问物联网平台资源的范围，参数值固定为default。
tokenType	String(256)	accessToken的类型，参数值固定为bearer。
expiresIn	Number(256)	平台生成并返回accessToken的有效时间，单位为秒。
accessToken	String(256)	鉴权参数，访问物联网平台API接口的凭证。
refreshToken	String(256)	鉴权参数，用来刷新accessToken，refreshToken的有效时间为“1”个月。

## 错误码

Http状态码	错误码	错误描述	说明
400	100449	The device is freezed can't operate.	用户无操作权限。 处理建议：请检查appId所对应用是否有接口调用权限。
400	102202	Required Parameter is null or empty.	必选参数为空。 处理建议：请检查请求参数中的必选参数是否已填写。

Http状态码	错误码	错误描述	说明
401	100208	AppId or secret is not right.	appId或secret错误。 处理建议： <ul style="list-style-type: none"><li>● 请检查appId和secret是否有误，是否新增字符或缺少字符。</li><li>● 请检查请求路径中的IP地址是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.1.2 刷新 token

#### 典型场景

第三方应用通过鉴权接口获取到的accessToken是有有效时间的，在accessToken快过期时，第三方应用通过调用此接口，获取新的accessToken。

#### 接口功能

支持第三方应用在accessToken到期前，从物联网平台获取新的accessToken。

#### 接口描述

```
def refreshAuthToken(self, arInDTO)
```

#### 接口所属类

Authentication

#### 参数说明

参数	必选/可选	类型	位置	描述
arInDTO	必选	AuthRefreshInDTO	body	见下表AuthRefreshInDTO的描述。

AuthRefreshInDTO

参数	必选/可选	类型	位置	描述
appId	必选	String(256)	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。 <code>appId</code> 在物联网平台的SP Portal上创建应用时获得。
secret	必选	String(256)	body	第三方应用的密码，与 <code>appId</code> 对应，用于登录访问物联网平台。 <code>secret</code> 在物联网平台的SP Portal上创建应用时获得。
refreshToken	必选	String(256)	body	刷新token，用来获取一个新的accessToken。 <code>refreshToken</code> 在调用 <a href="#">鉴权</a> 接口时获得。

## 返回值

AuthRefreshOutDTO

参数	类型	描述
scope	String(256)	申请的权限范围，参数值固定为default。
tokenType	String(256)	鉴权token的类型，参数值固定为bearer。
expiresIn	Number(256)	平台生成并返回accessToken的有效时间，单位为秒。
accessToken	String(256)	鉴权参数，访问物联网平台API接口的凭证。
refreshToken	String(256)	鉴权参数，用来刷新accessToken， <code>refreshToken</code> 的有效时间为“1”个月。

## 错误码

Http状态码	错误码	错误描述	说明
400	100449	The device is freezed cant operate.	用户无操作权限。 处理建议：请检查 <code>appId</code> 所对应用户是否有接口调用权限。
400	102202	Required Parameter is null or empty.	必选参数为空。 处理建议：请检查请求参数中的必选参数是否已填写。

Http状态码	错误码	错误描述	说明
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100208	AppId or secret is not right.	appId或secret错误。 处理建议： <ul style="list-style-type: none"><li>● 请检查appId和secret是否有误，是否新增字符或缺少字符。</li><li>● 请检查请求路径中的IP地址是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 4.3.2 设备管理

第三方应用向物联网平台添加设备，获取设备的ID和验证码，待设备完成接入物联网平台流程后，设备与第三方应用建立从属关系。

### 4.3.2.1 注册直连设备

#### 典型场景

在直连设备接入物联网平台前，第三方应用需要调用此接口在物联网平台注册设备，并把设备的唯一标识码（如IMEI）设置为设备接入平台的验证码。在设备接入物联网平台时携带设备唯一标识，完成设备的接入认证。

#### 接口功能

支持第三方应用在物联网平台注册直连设备，完成注册后设备方可接入物联网平台。

#### 接口描述

```
def regDirectDevice(self, rddInDto, appId, accessToken)
```

#### 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
rddInDto	必选	RegDirectDeviceInDTO	body	见下表RegDirectDeviceInDTO的描述。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### RegDirectDeviceInDTO

参数	必选/可选	类型	位置	描述
customFields	可选	List< <a href="#">CustomField</a> >	Body	自定义字段列表，用户可设置自定义字段。
deviceInfo	可选	<a href="#">DeviceInfoDTO</a>	Body	设备信息。
deviceInfo2	可选	<a href="#">DeviceInfo2</a>	Body	设备信息2。
deviceName	可选	String(256)	Body	设备名称。
endUserId	可选	String(256)	Body	终端用户ID。 在NB-IoT方案中，endUserId设置为设备的IMSI号。在Smarthome解决方案中，endUserId设置为App账号。
groupId	可选	String(256)	Body	设备所属的设备组ID。
imsi	可选	String(1-64)	Body	NB-IoT终端的IMSI。
isSecure	可选	Boolean	Body	指定设备是否为安全设备，默认值为“false”。 ● “true”：安全设备 ● “false”：非安全设备 <b>说明</b> 如果用户需要注册安全设备，isSecure参数必须填写。
location	可选	<a href="#">Location</a>	Body	设备位置信息。

参数	必选/可选	类型	位置	描述
nodeId	必选	String(256)	Body	<p>设备的唯一标识，必须与设备上报的设备标识一致。通常使用MAC地址，Serial No或IMEI作为nodeId。</p> <p><b>说明</b></p> <p>使用IMEI作为nodeId时，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li><li>海思芯片设备的唯一标识为IMEI号</li><li>其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
psk	可选	String(8-32)	Body	请求中指定psk，则平台使用指定的psk；请求中不指定psk，则由平台生成psk。取值范围是“a-f、A-F、0-9”组成的字符串。
tags	可选	List< <b>TagDT O2</b> >	Body	设备的标签信息。
timeout	可选	Integer(>=0)	Body	<p>超时时间。当调用北向接口对设备开户，在超时时间内可绑定设备，若超过timeout时间且未绑定设备，则会删除超时的开户信息。</p> <p>取值范围：0~2147483647。若填写为“0”，则表示设备验证码不会失效（建议填写为“0”）。</p> <p>默认值：“180”（默认值可配置，具体配置值请咨询物联网平台运维人员。）</p> <p>单位：秒。</p>
verifyCode	条件可选 必选	String(256)	body	<p>设备验证码，若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。</p> <p>在NB-IoT方案中，verifyCode为必填参数，且必须与nodeId设置成相同值。</p>
productId	可选	String(256)	Body	设备所属的产品ID。
account	可选	String	Body	IPC设备账户。

CustomField:

参数	必选/可选	类型	位置	描述
fieldName	可选	String(256)	Body	字段名字。
fieldType	可选	String(256)	Body	字段类型。
fieldValue	可选	String(256)	Body	字段值。

DeviceInfoDTO:

参数	必选/可选	类型	位置	描述
nodeId	可选	String(256)	Body	设备的唯一标识。
name	可选	String(256)	Body	设备名称。
description	可选	String(2048)	Body	设备的描述信息。
manufacturerId	可选	String(256)	Body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	Body	厂商名称。
mac	可选	String(256)	Body	设备的MAC地址。
location	可选	String(2048)	Body	设备的位置信息。
deviceType	可选	String(256)	Body	设备类型，大驼峰命名方式，如 MultiSensor、ContactSensor、CameraGateway。
model	必选	String(256)	Body	设备的型号。 Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
swVersion	可选	String(256)	Body	设备的软件版本。 Z-Wave: 主版本号.次版本号，如：1.1。
fwVersion	可选	String(256)	Body	设备的固件版本。
hwVersion	可选	String(256)	Body	设备的硬件版本。
protocolType	可选	String(256)	Body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

参数	必选/可选	类型	位置	描述
bridgeId	可选	String(256)	Body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
status	可选	String	Body	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
statusDetail	可选	String(256)	Body	<p>设备的状态详情，跟status取值对应。</p> <ul style="list-style-type: none"><li>● status为“ONLINE”时，取值“NONE”、“COMMUNICATION_ERROR”，“CONFIGURATION_ERROR”、“BRIDGE_OFFLINE”或“FIRMWARE_UPDATING”、“DUTY_CYCLE”、“NOT_ACTIVE”</li><li>● status为“OFFLINE”时，取值“NONE”、“CONFIGURATION_PENDING”</li></ul>
mute	可选	String	Body	<p>表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。</p> <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
supportedSecurity	可选	String	Body	<p>表示设备是否支持安全模式。</p> <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
isSecurity	可选	String	Body	<p>表示设备当前是否启用安全模式。</p> <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
signalStrength	可选	String(256)	Body	设备的信号强度。
sigVersion	可选	String(256)	Body	设备的sig版本。
serialNumber	可选	String(256)	Body	设备的序列号。
batteryLevel	可选	String(256)	Body	设备的电池电量。

 **说明**

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceInfo2:

名称	必选/可选	类型	位置	说明
region	可选	String(256)	Body	设备所在区域。
timezone	可选	String(256)	Body	设备所在时区。
activeServiceCount	可选	Integer	Body	设备的活跃服务数量。

Location:

名称	必选/可选	类型	位置	说明
accuracy	可选	Double	Body	精确度，最小值为“0”。
crashInformation	可选	String	Body	紧急信息。
description	可选	String	Body	位置描述信息。
heading	可选	String	Body	设备的方向信息。
language	可选	String	Body	语言信息。
latitude	可选	Double	Body	设备所在的纬度，取值“-90~90”。
longitude	可选	Double	Body	设备所在的经度，取值“-180~180”。
numberOfPassengers	可选	String	Body	乘客数量。
region	可选	String	Body	设备所在区域。
time	可选	DateTime	Body	时间信息。
vehicleSpeed	可选	String	Body	车辆速度。

TagDTO2:

参数	必选/可选	类型	位置	描述
tagName	必选	String(1-128)	body	标签名称。
tagValue	必选	String(1-1024)	body	标签值。

## 返回值

RegDirectDeviceOutDTO

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
verifyCode	String(256)	验证码，设备可以通过验证码获取设备ID和密码。若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。
timeout	Number	验证码有效时间，单位秒，设备需要在有效时间内接入物联网平台。
psk	String(32)	随机psk参数，若请求中携带了psk，则使用请求中的psk，否则由平台生成随机psk参数。

## 错误码

Http状态码	错误码	错误描述	说明
200	103028	The license pool resources.	License资源用尽。
400	100003	Invalid verify code.	验证码无效。 处理建议：请检查接口请求中的verifyCode是否有误。若请求参数中未带verifyCode，请联系物联网平台维护人员处理。
400	100007	Bad request message.	参数不合法。 处理建议：deviceId未赋值，请参考请求参数说明填写请求。
400	100416	The device has already been binded.	设备已经绑定。 处理建议：请检查设备是否已经注册。

Http状态码	错误码	错误描述	说明
400	100426	The nodeId is duplicated.	nodeId重复。 处理建议：请检查接口请求中nodeId是否有误。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100025	AppId for auth not exist.	获取不到appId鉴权信息。 处理建议： <ul style="list-style-type: none"><li>● 请检查请求结构体的Header头域中是否给字段app_key赋值。</li><li>● 若通过HTTP调用，请联系物联网平台维护人员确认Header头域中放置appId的字段名是app_key还是x-app-key。</li></ul>
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
403	600002	The product not existed.	产品不存在。 处理建议：物联网平台未找到productId对应的产品，请检查productId是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	授权应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId 是否有误。</li><li>● 请检查请求路径（url）中传入的 appId 是否有误。</li></ul>
500	100412	The amount of device has reached the limit.	当前应用下设备数量达到上限。 处理建议：请检查当前应用下的设备数量是否已达到所申请资源的上限。
500	100441	The amount of nonSecure device has reached the limit.	非安全设备数量已达到上限。
500	103026	The license is not exist.	License不存在。 处理建议：物联网平台内部License问题，请联系物联网平台维护人员处理。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.2.2 刷新设备密钥

#### 典型场景

已在物联网平台注册的设备，若设备的唯一标识码发生变更（如更换设备等场景），第三方应用需要调用此接口在物联网平台更新设备的唯一标识，并重新绑定设备。

##### 说明

仅当设备处于离线状态时，才能进行刷新设备秘钥操作。

#### 接口功能

支持第三方应用在物联网平台更新已注册设备的nodeId，在设备deviceId不变的情况下，重新绑新设备。

#### 接口描述

```
def refreshDeviceKey(self, rdkInDTO, appId, accessToken)
```

#### 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
rdkInDTO	必选	RefreshDeviceKeyInDTO	body	见下表RefreshDeviceKeyInDTO的描述。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### RefreshDeviceKeyInDTO

参数	必选/可选	类型	位置	描述
deviceId	必选	String(256)	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
verifyCode	可选	String(256)	body	备验证码，若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。建议和nodeId设置成一样。
nodeId	可选	String(256)	body	<p>设备的唯一标识，通常使用MAC，MAC地址，Serial No或IMEI作为nodeId。</p> <ul style="list-style-type: none"> <li>● 值空时，nodeId不变。</li> <li>● 值不为空时，更新nodeId。</li> </ul> <p><b>说明</b></p> <p>若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"> <li>● 高通芯片设备的唯一标识为urn:imei:xxxx，xxxx为IMEI号</li> <li>● 海思芯片设备的唯一标识为IMEI号</li> <li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li> </ul>
timeout	可选	Number	body	<p>验证码超时时间，单位秒，取值大于等于0的整数。</p> <ul style="list-style-type: none"> <li>● 值为空时，使用默认值（180s）。</li> <li>● 值为0时，永不过期。</li> <li>● 值为非0，指定时间。</li> </ul>

## 返回值

### RefreshDeviceKeyOutDTO

参数	类型	描述
verifyCode	String(256)	验证码，设备可以通过验证码获取设备ID和密码。若在请求中指定verifyCode，则响应中返回请求中指定的verifyCode；若请求中不指定verifyCode，则由物联网平台自动生成。
timeout	Number	验证码有效时间，单位秒，设备需要在有效时间内接入物联网平台。

## 错误码

Http状态码	错误码	错误描述	说明
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
400	100003	Invalid verify code.	验证码无效。 处理建议：请检查接口请求中的verifyCode是否有误。若请求参数中未带verifyCode，请联系物联网平台维护人员处理。
400	100007	Bad request message.	参数不合法。 处理建议：deviceId未赋值，请参考请求参数说明填写请求。
400	100426	The nodeId is duplicated.	nodeId重复。 处理建议：请检查接口请求中nodeId是否有误。
400	100610	Device is not active.	设备未激活。 处理建议：请检查该设备是否已接入物联网平台并激活。
400	100611	Device is online.	设备在线。 处理建议：请下线该设备或者断开设备与物联网平台的连接。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
401	100025	AppId for auth not exist.	获取不到appId鉴权信息。 处理建议： <ul style="list-style-type: none"><li>● 请检查请求结构体的Header头域中是否给字段app_key赋值。</li><li>● 若通过HTTP调用，请联系物联网平台维护人员确认Header头域中放置appId的字段名是app_key还是x-app-key。</li></ul>
403	100203	The application is not existed.	授权应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.2.3 修改设备信息

#### 典型场景

第三方应用在物联网平台注册设备后，若设备的基本信息发生变更，也可调用此接口在物联网平台修改设备的信息。

#### 接口功能

支持第三方应用修改设备的基本信息，包括设备类型、设备模型、厂商信息、接入协议等。

#### 接口描述

```
def modifyDeviceInfo(self, mdiInDto, deviceId, appId, accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	类型	位置	描述
mdiInDto	必选	ModifyDeviceInfoInDTO	body	见下表ModifyDeviceInfoInDTO的描述。
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。

参数	必选/可选	类型	位置	描述
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

## ModifyDeviceInfoInDTO

参数	必选/可选	类型	位置	描述
customFields	可选	List< <a href="#">CustomField</a> >	Body	自定义字段列表，用户可设置自定义字段。
deviceConfig	可选	<a href="#">DeviceConfigDTO</a>	body	设备配置信息。
deviceType	可选	String(1~256)	body	设备类型，大驼峰命名方式，例如：MultiSensor、ContactSensor、CameraGateway。 注册设备后必须修改设备类型，且要与profile中定义的保持一致。
endUser	可选	String(1~256)	body	终端用户，若为直连设备，则endUser可选；若为非直连设备，则endUser可以为null。
location	可选	String(1~1024)	body	设备位置。
manufacturerId	可选	String(1~256)	body	厂商ID，唯一标识一个厂商。 注册设备后必须修改厂商ID，且要与profile中定义的保持一致。
manufacturerName	可选	String(1~256)	body	厂商名称。
model	可选	String(1~256)	body	设备型号，由厂商定义。 注册设备后必须修改设备型号，且要与profile中定义的保持一致。
mute	可选	Enum	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
name	可选	String(1~256)	body	设备名称。
organization	可选	String(1~256)	body	设备所属的组织信息。

参数	必选/可选	类型	位置	描述
protocolType	可选	String(1~256)	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。 注册设备后必须修改协议类型，且要与profile中定义的保持一致。
region	可选	String(1~256)	body	设备区域信息。
timezone	可选	String(1~256)	body	设备所在时区信息，使用时区编码，如上海时区对应的时区编码为Asia/Shanghai。
imsi	可选	String(1-64)	Body	NB-IoT终端的IMSI。
ip	可选	String(128)	Body	设备的IP地址。
isSecure	可选	Boolean	body	设备的安全状态，默认值为“false”。 ● “true”：安全 ● “false”：非安全
psk	可选	String(8~32)	body	psk参数，取值范围是“a-f、A-F、0-9”组成的字符串。
tags	可选	List< <a href="#">Tag 2</a> >	Body	设备的标签信息。

CustomField:

参数	必选/可选	类型	位置	描述
fieldName	可选	String(256)	Body	字段名字。
fieldType	可选	String(256)	Body	字段类型。
fieldValue	可选	String(256)	Body	字段值。

DeviceConfigDTO:

参数	必选/可选	字段类型	描述
dataConfig	可选	DataConfigDTO	数据配置信息。

DataConfigDTO:

参数	必选/可选	字段类型	描述
dataAgingTime	可选	Integer	数据老化时长，取值范围：0-90，单位：天。

Tag2:

参数	必选/可选	类型	位置	描述
tagName	必选	String(1-128)	body	标签名称。
tagValue	必选	String(1-1024)	body	标签值。
tagType	可选	Integer	body	标签类型。

## 返回值

void

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	100440	The isSecure is invalid.	isSecure参数值有误。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议:</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	<p>应用未被授权。</p> <p>处理建议: 若非应用授权场景, 请确认请求参数中的appId为空。</p>
403	1010004	app throttle exceed.	<p>应用调用接口过于频繁, 超过流控值（默认值是100次/60s）。</p> <p>处理建议: 联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。</p>
403	1010005	App_key or access_token is invalid.	<p>错误的token信息。</p> <p>处理建议: 请检查接口请求中所携带的accessToken是否有误。</p>
403	500004	The amount of frozen devices has reached the limit.	被冻结设备已达到上限。
404	100403	The device is not existed.	<p>设备不存在。</p> <p>处理建议: 物联网平台未找到deviceId对应的设备, 请检查deviceId是否有误。</p>
404	100418	The deviceData is not existed.	<p>设备信息不存在。</p> <p>处理建议:</p> <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>● Header中携带的appId有误, 请确认该 appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId, 请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议:</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	100441	The amount of nonSecure device has reached the limit.	非安全设备数量已达到上限。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.2.4 删除直连设备

#### 典型场景

已在物联网平台注册的直连设备，若不再需要接入平台时，第三方应用可调用此接口在物联网平台删除设备。后续若设备要再次接入平台，第三方应用需要在物联网平台重新注册设备。

#### 接口功能

支持第三方应用在物联网平台删除已注册的直连设备，使设备无法再接入平台。

#### 接口描述

```
def deleteDirectDevice(self, deviceId, cascade, appId, accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
cascade	可选	String	query	仅当设备下连接了非直连设备时生效，不设置时可填写None。 <ul style="list-style-type: none"><li>“true”，级联删除，即删除直连设备和其下的非直连设备。</li><li>“false”，删除直连设备，但是不删其下的非直连设备，并将非直连设备的属性变为直连设备属性。</li></ul>

参数	必选/可选	类型	位置	描述
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 返回值

void

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。

Http状态码	错误码	错误描述	说明
404	100418	The deviceData is not existed.	<p>设备信息不存在。 处理建议：</p> <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	<p>应用不存在。 处理建议：</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	<p>服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。</p>

#### 4.3.2.5 查询设备激活状态

##### 典型场景

第三方应用通过“注册直连设备”接口在物联网平台注册设备后，在设备首次接入平台之前，设备的激活状态为false；当设备首次接入平台后，设备的激活状态为true（无论设备接入后是处于在线、离线或异常状态）。第三方应用可调用此接口查询设备在物联网平台的激活状态，确定设备是否已接入平台。

##### 接口功能

支持第三方应用通过设备ID查询设备在物联网平台的激活状态，确认设备是否已接入平台。

##### 接口描述

```
def queryDeviceStatus(self, deviceId, appId, accessToken)
```

##### 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 返回值

QueryDeviceStatusOutDTO

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
activated	Boolean	激活状态，设备是否通过验证码获取密码的状态标识。 <ul style="list-style-type: none"> <li>● “true”：已激活</li> <li>● “false”：未激活</li> </ul>
name	String(256)	设备名称。

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径（url）中传入的appId是否有误。</li> </ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.2.6 查询设备实时位置信息

#### 典型场景

第三方应用若需要获取设备当前最新的位置信息时，可调用此接口查询设备的实时位置信息，当前仅支持查询NB-IoT设备的实时位置信息。

#### 接口功能

物联网平台通过和位置服务器对接，向第三方应用提供查询NB-IoT设备实时位置信息的能力。

##### 注意

根据位置服务器的能力和配置情况，查询设备实时位置信息接口会存在较大的时延，而且并发能力有限。具体情况建议咨询物联网平台维护人员。

## 接口描述

```
def queryDeviceRealtimeLocation(self, qdrlnDTO, appId, accessToken)
```

## 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
qdrlnDTO	必选	QueryDeviceRealtimeLocationInDTO	body	见下表 QueryDeviceRealtimeLocationInDTO 的描述。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

QueryDeviceRealtimeLocationInDTO

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
horAcc	必选	Integer	body	水平误差，单位：米，如果不携带，则默认“1000”米。
geoInfo	可选	CoordinateReferenceSystem	body	地理坐标信息要求，默认“WGS84”，其他坐标暂不支持。具体参见 <a href="#">CoordinateReferenceSystem</a> 结构体。

CoordinateReferenceSystem结构体说明：

参数	必选/可选	类型	位置	描述
identifier	必选	Identifier	body	坐标参考系统标识。具体参见 <a href="#">Identifier</a> 结构体。

Identifier结构体说明：

参数	必选/可选	类型	位置	描述
code	必选	String	body	坐标参考系统编号， 默认“4326”，当前只支持这个WGS84数字坐标。
codeSpace	必选	String	body	编码空间， 当前只支持EPSG。
edition	必选	String	body	编辑版本， 当前只支持6.1。

## 返回值

QueryDeviceRealtimeLocationOutDTO

参数	类型	描述
deviceId	String	设备ID， 用于唯一标识一个设备。
pd	PdIoMDTO	定位结果， 定位成功时携带， 具体参见 <a href="#">PdIoMDTO</a> 结构体。
poserr	PoserrIoMDTO	定位错误信息， 定位失败时携， 具体参见 <a href="#">PoserrIoMDTO</a> 结构体。

PdIoMDTO结构体说明：

参数	类型	描述
time	String	定位时间， 格式“yyyyMMddhhmmss”。
utcOff	String	时间时区， 格式“hhmm”。
srsName	String	坐标参考， 固定取值：“www.epsg.org#4326”。
X	String	X轴， 纬度， 格式：“DD MM SS.hhhN”或“DD MM SS.hhhS”， 最后的N指北纬， S指南纬。
Y	String	Y轴， 经度， 格式：“DD MM SS.hhhE”或“DD MM SS.hhhW”， 最后的E指东经， W指西经。
radius	String	误差半径， 单位：米。

PoserrIoMDTO结构体说明：

参数	类型	描述
time	String	定位时间， 格式“yyyyMMddhhmmss”。
resid	String	定位结果值。

参数	类型	描述
add_info	String	定位详细结果。

## 错误码

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
500	103801	Exceed the system concurrency restriction.	超过系统并发限制。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.2.7 查询设备影子

#### 典型场景

若设备处于离线或异常状态时，第三方应用无法通过命令下发的方式向设备下发配置。此时可把要下发的配置设置到设备影子，当设备重新上线时，设备影子会把配置下发给设备。第三方应用可调用此接口查看设备影子上的设备配置信息及最新的设备上报数据信息。

#### 接口功能

支持第三方应用查询单个设备的设备影子信息，包括对设备的配置信息（desired区）和设备最新上报的数据信息（reported区）。

#### 接口描述

```
def queryDeviceShadow(self, deviceId, appId, accessToken)
```

#### 接口所属类

DeviceManagement

#### 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。

参数	必选/可选	类型	位置	描述
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 返回值

QueryDeviceShadowOutDTO

参数	类型	描述
deviceId	String(36)	设备ID，用于唯一标识一个设备。
gatewayId	String(36)	网关ID，用于标识一个网关。
nodeType	Enum	设备的类型。
createTime	String(256)	设备创建的时间。
lastModifiedTime	String(256)	设备最后一次修改信息的时间。
deviceInfo	DeviceInfo	设备详细信息，具体参见 <a href="#">DeviceInfo</a> 结构体。
services	List<DeviceServiceB>	设备的服务能力，具体参见 <a href="#">DeviceServiceB</a> 结构体。

DeviceInfo结构体说明：

参数	类型	描述
nodeId	String(256)	设备的唯一标识。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacturerId	String(256)	厂商ID，唯一标识一个厂商，与设备Profile文件中的信息保持一致。
manufacturerName	String(256)	厂商名称，与设备Profile文件中的信息保持一致。
mac	String(256)	设备的MAC地址。
location	String(2048)	设备的位置信息。

参数	类型	描述
deviceType	String(256)	设备类型，大驼峰命名方式，与设备Profile文件中的信息保持一致，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号，与设备Profile文件中的信息保持一致。 Z-Wave: ProductType + ProductId, 16 进制格式 XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
swVersion	String(256)	设备的软件版本，。 Z-Wave: 主版本号.次版本号，如：1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
protocolType	String(256)	设备使用的协议类型，与设备Profile文件中的信息保持一致，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	String(256)	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
status	String	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
statusDetail	String(256)	设备的状态详情，status条件取值，取值查看下表status和statusDetail。
mute	String	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
supportedSecurity	String	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
isSecurity	String	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。
batteryLevel	String(256)	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceServiceB说明：

参数	类型	描述
serviceId	String(256)	服务ID，用于标识一个服务。
reportedProps	ObjectNode	设备上报的信息。
desiredProps	ObjectNode	设备下发的信息。
eventTime	String(256)	事件发生的时间。
serviceType	String(256)	服务类型。

## 错误码

Http状态码	错误码	错误描述	说明
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	pp_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.2.8 修改设备影子

#### 典型场景

物联网平台支持创建设备的“影子”。设备影子用于存储设备最新上报的服务属性数据，以及第三方应用对服务属性的配置（服务属性为设备Profile文件中定义的service属性）。若设备处于离线或异常状态时，第三方应用无法通过命令下发的方式向设备下发配置，此时可把要下发的配置设置到设备影子，当设备重新上线时，设备影子会把配置下发给设备。第三方应用可调用此接口在设备影子上配置要下发给设备的配置信息。

每个设备有且只有一个设备影子，设备影子上有desired和report区。

- desired区用于存储对设备服务属性的配置。若设备当前在线，desired区的配置会即时下发给设备；若设备当前不在线，desired区的配置会在设备上线时再下发给设备。

- report区用于存储设备最新上报的服务属性数据。当设备上报数据时，平台会把数据同步到设备影子的report区。

## 接口功能

支持第三方应用配置设备影子desired区的信息，用于设备上线时把配置下发给设备。

## 接口描述

```
def modifyDeviceShadow(self, mdsInDTO, deviceId, appId, accessToken)
```

## 接口所属类

DeviceManagement

## 参数说明

参数	必选/可选	类型	位置	描述
mdsInDTO	必选	ModifyDeviceShadowInDTO	body	见下表 ModifyDeviceShadowInDTO的描述。
deviceId	必选	String(256)	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

ModifyDeviceShadowInDTO

参数	必选/可选	类型	位置	描述
serviceDesireds	必选	List<ServiceDesiredDTO>	body	需要修改的设备配置或状态信息。ServiceDesiredDTO见下表。

ServiceDesiredDTO结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	可选	String(1-256 )	body	服务ID，用于标识一个服务。
desired	可选	Object	body	设备的状态。

## 返回值

void

## 错误码

Http状态码	错误码	错误描述	说明
200	100425	The special deviceCapability is not exist.	设备模板不存在。 处理建议：请检查设备模板是否已在物联网平台上传。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>请检查该设备的Profile文件是否已上传到物联网平台。</li><li>请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	107002	The properties is empty in database.	设备属性不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107003	The request properties is unknown.	设备状态为未知。 处理建议：请检查设备与平台的连接是否正常。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	10044 3	The property is forbidden to write.	设备属性不可写。
403	10100 04	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	10100 05	pp_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken是否有误。
404	10040 3	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	10041 8	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	10002 3	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	10020 3	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.3 批量处理

第三方应用通过批量处理接口对已接入物联网平台的设备进行批量操作。

### 4.3.3.1 创建批量任务

#### 典型场景

第三方应用若需要对多个设备进行批量操作时，可调用此接口创建设备的批量操作任务，当前仅支持给批量设备下发缓存命令。

#### 接口功能

支持第三方应用在物联网平台创建设备的批量操作任务。

#### 接口描述

```
def createBatchTask(self, btcInDTO, accessToken)
```

#### 接口所属类

BatchProcess

#### 参数说明

参数	必选/可选	类型	位置	描述
btcInDTO	必选	BatchTaskCreateInDTO	body	见下表BatchTaskCreateInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

BatchTaskCreateInDTO

参数	必选/可选	类型	位置	描述
appId	必选	String(64)	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。appid在物联网平台的SP Portal上创建应用时获得。
param	必选	Object Node	body	任务详细参数，根据taskType任务类型的不同对应不同类型参数。
taskName	必选	String	body	任务名称，最大长度“256”字符。
taskType	必选	String	body	任务类型，取值范围：“DeviceReg/DeviceCmd/DeviceLocation/DeviceMod/DeviceDel/DeviceLicense”。
timeout	必选	Integer	body	任务超时时长，单位分钟，范围为10-2880。

参数	必选/ 可选	类型	位置	描述
tags	可选	List<Ta gDTO2 >	body	标签列表。

ObjectNode:

参数	必选/ 可选	类型	位置	描述
type	必选	String	body	批量命令类型, 取值范围: DeviceList/DeviceType/DeviceArea/GroupList/Broadcast/GroupIdList。
device List	条件必选	List<Str ing>	body	设备ID列表, type为DeviceList时需要填写。
device Type	条件必选	String	body	设备类型, type为DeviceType时需要填写, 其值应当与profile中定义的一致。
manufa cturerI d	条件可选	String	body	厂商ID, type为DeviceType时可填写, 其值应当与profile中定义的一致。
model	条件可选	String	body	设备型号, type为DeviceType时可填写, 其值应当与profile中定义的一致。
device Locatio n	条件必选	String	body	设备位置, type为DeviceArea时需要填写。
groupL ist	条件必选	List<Str ing>	body	群组Id列表或设备组名称列表, type为 GroupIdList时需要填写群组Id, type为 GroupList时需要填写设备组名称。
comm and	必选	Comma ndDTO	body	命令信息。
callbac kUrl	可选	String	body	命令执行结果的推送地址。
maxRe transmi t	可选	Integer(0~3)	body	命令下发最大重传次数, 取值范围: 0~3。
groupT ag	可选	String	body	群组标签。

CommandDTO:

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-4)	body	命令对应的服务ID，要与profile中定义的serviceId保持一致。
method	必选	String(1-28)	body	服务下具体的命令名称，要与profile中定义的命令名保持一致。
paras	可选	ObjectNode	body	命令参数， jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。

TagDTO2:

参数	必选/可选	类型	位置	描述
tagName	必选	String(1-128)	body	标签名称。
tagValue	必选	String(1-1024)	body	标签值。

## 返回值

BatchTaskCreateOutDTO

参数	类型	描述
taskID	String	批量任务的ID。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	101001	Resource doesn't exist.	资源不存在。

Http状态码	错误码	错误描述	说明
200	105001	The batchTask count has reached the limit.	未完成的任务数大于等于10个时，返回任务达到数量限制。
200	105002	The batchTask name has exist.	任务名字已存在。 处理建议：修改任务名称。
400	105201	The tagName and tagValue has been used on the platform.	平台已使用tagName和tagValue。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
401	100028	The user has no right.	用户没有操作权限。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105202	The tag is not existed.	标签不存在。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.3.2 查询指定批量任务信息

#### 典型场景

第三方应用通过“创建批量任务”接口创建了设备的批量操作任务后，可调用此接口查询该批量任务的信息，包括任务的当前状态、子任务完成情况等。

#### 接口功能

支持第三方应用通过任务ID查询单个批量任务的信息。

#### 接口描述

```
def queryOneTask(self, taskId, select, appId, accessToken)
```

#### 接口所属类

BatchProcess

#### 参数说明

参数	必选/可选	类型	位置	描述
taskId	必选	String	path	批量任务ID，创建批量任务后获得。
select	必选	String	query	指定可选的返回值，可取值：tag。不指定时可填写None。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 返回值

QueryOneTaskOutDTO

参数	类型	描述
appId	String	批量任务所归属的appId。
taskId	String	批量任务ID。
taskName	String	批量任务名称。
operator	String	下发该批量任务的操作员。
taskFrom	String	批量任务的来源。 <ul style="list-style-type: none"><li>● “Portal”：通过SP Portal创建的。</li><li>● “Northbound”：调用北向API接口创建的。</li></ul>
taskType	String	批量任务的类型，取值范围：“DeviceReg/DeviceCmd/DeviceLocation/DeviceMod/DeviceDel/DeviceLicense”。
status	String	批量任务的状态，取值范围：“Pending/Running/Complete/Timeout”。
startTime	String	批量任务的创建时间。
timeout	Integer	批量任务的超时时间，单位秒。
progress	Integer	批量任务的进度，单位：千分比，范围：“0-1000”，向下取整。
totalCnt	Integer	任务的总数。
successCnt	Integer	成功的任务数。
failCnt	Integer	失败的任务数。
timeoutCnt	Integer	超时的任务数。
expiredCnt	Integer	未执行的失效任务数。
completeCnt	Integer	完成的任务数，完成的任务数=成功的任务数+失败的任务数+超时的任务数。
successRate	Integer	任务成功率，单位：千分比，范围：“0-1000”，向下取整。
param	ObjectNode	不同任务类型的具体参数。
tags	List<TagDT O2>	批量任务的标签列表。

ObjectNode:

参数	类型	描述
type	String	批量命令类型，取值范围：DeviceList/DeviceType/DeviceArea/GroupList/Broadcast/GroupIdList。

参数	类型	描述
deviceList	List<String>	设备ID列表, type为DeviceList时返回值。
deviceType	String	设备类型, type为DeviceType时返回值, 其值应当与profile中定义的一致。
manufacturerId	String	厂商ID, type为DeviceType时返回值, 其值应当与profile中定义的一致。
model	String	设备型号, type为DeviceType时返回值, 其值应当与profile中定义的一致。
deviceLocation	String	设备位置, type为DeviceArea时返回值。
groupList	List<String>	群组名称列表, type为GroupList时返回值。
command	CommandDTO	命令信息。
callbackUrl	String	命令执行结果的推送地址。
maxRetransmit	Integer(0~3)	命令下发最大重传次数, 取值范围: 0-3。
groupTag	String	群组标签。

CommandDTO:

参数	类型	描述
serviceId	String(1-64)	命令对应的服务ID, 要与profile中定义的serviceId保持一致。
method	String(1-128)	服务下具体的命令名称, 要与profile中定义的命令名保持一致。
paras	ObjectNode	命令参数, jsonString格式, 里面是一个个键值对, 每个键都是profile中命令的参数名 (paraName), 具体格式需要应用和设备约定。

TagDTO2:

参数	类型	描述
tagName	String(1-128)	标签名称。
tagValue	String(1-1024)	标签值。

## 错误码

Http状态码	错误码	错误描述	说明
400	100019	Illegal request.	非法请求。 处理建议：请检查接口请求中的必须参数是否为空。
400	100022	The input is invalid	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105005	The batchTask is not existed.	批量任务不存在。 处理建议：请检查接口请求中的taskId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

### 4.3.3.3 查询批量任务的子任务信息

#### 典型场景

第三方应用通过“创建批量任务”接口创建了设备的批量操作任务后，可调用此接口查询该批量任务中子任务的详情信息，如任务的执行状态、任务内容等。

#### 接口功能

支持第三方应用根据条件查询批量任务中子任务的详情信息，支持查询本应用创建的批量任务信息。

#### 接口描述

```
def queryTaskDetails(self, qtdInDTO, accessToken)
```

#### 接口所属类

BatchProcess

#### 参数说明

参数	必选/可选	类型	位置	描述
qtdInDTO	必选	QueryTaskD etailsInDTO	query	见下表QueryTaskDetailsInDTO的 描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

#### QueryTaskDetailsInDTO

参数	必选/可选	类型	位置	描述
appId	可选	String	query	如果是本应用的任务，此参数值可以填写None，否则填写授权应用的appId。
taskId	必选	String	query	批量任务的ID。
status	可选	String	query	任务的详情状态，“Pending/Success/Fail/Timeout”。
index	可选	Integer	query	批量任务文件里第几行的任务，查询批量注册任务时使用。
nodeId	可选	String	query	设备nodeId，查询批量注册任务时使用。
deviceId	可选	String	query	设备Id，查询批量命令任务时使用。

参数	必选/可选	类型	位置	描述
comm andId	可选	String	query	命令Id, 查询批量命令任务时使用。
pageNo	可选	Integer	query	分页查询参数。 ● 值为空时, 查询内容不分页。 ● 值为大于等于0的整数时, 分页查询。 ● 值为0时查询第一页。
pageSize	可选	Integer	query	分页查询参数, 取值大于等于1的整数, 缺省值: “1”。

## 返回值

QueryTaskDetailsOutDTO

参数	类型	描述
pageNo	Long	分页查询参数。 ● 值为空时, 查询内容不分页。 ● 值为大于等于0的整数时, 分页查询。 ● 值为0时查询第一页。
pageSize	Long	分页查询参数, 取值大于等于1的整数, 缺省值: “1”。
totalCount	Long	查询到的任务总数。
taskDetails	List< <a href="#">QueryTaskDetailDTOCloud2NA</a> >	任务详情信息列表。

QueryTaskDetailDTOCloud2NA:

参数	类型	描述
status	String	任务执行状态, 取值范围: Pending/WaitResult/Success/Fail/Timeout。
output	String	批量下发命令的输出信息。
error	String	任务的错误原因, 格式为: {"error_code": "****", "error_desc": "****"}。
param	<a href="#">ObjectNode</a>	不同任务类型的具体参数。

ObjectNode:

参数	类型	描述
deviceId	String	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
commandId	String	下发的命令ID，用于唯一标识一条命令，在下发设备命令时由物联网平台分配获得。

## 错误码

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>
403	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	105005	The batchTask is not existed.	批量任务不存在。 处理建议：请检查接口请求中的 taskId是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的 appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 4.3.4 订阅管理

物联网平台支持第三方应用对设备数据的订阅，若订阅的设备信息有变更，平台会推送给第三方应用。消息订阅接口需要与[消息推送](#)接口配合使用。

### 4.3.4.1 订阅平台业务数据

#### 典型场景

第三方应用可在物联网平台订阅设备相关的业务数据，当物联网平台中设备的业务信息发生变化时（如设备注册、设备数据上报、设备状态变更等），平台会向第三方应用发送通知消息，通知其具体的变化信息。第三方应用可以调用此接口向物联网平台订阅不同类型的设备业务变更通知。

#### 接口功能

支持第三方应用向物联网平台订阅设备的业务变更信息，当设备的状态、数据等信息发生变化时，平台会向第三方应用发送通知消息。

#### 接口描述

```
def subDeviceBusinessData(self, sdbdInDTO, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
sdbdInDTO	必选	SubDeviceBusinessDataInDTO	body	见下表 SubDeviceBusinessDataInDTO 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

SubDeviceBusinessDataInDTO

参数	必选/ 可选	类型	位置	描述
ownerFlag	可选	String(256)	query	<p>callbackUrl的所有者标识。</p> <ul style="list-style-type: none"><li>● ownerflag为“false”时，表示callbackUrl的owner是授权应用。</li><li>● ownerflag为“true”时，表示callbackUrl的owner为被授权应用。</li></ul>
appId	可选	String(256)	body	设备所属的appId。
notifyType	必选	String	body	<p>通知类型，第三方应用可以根据通知类型接收物联网平台推送的对应通知消息。</p> <ul style="list-style-type: none"><li>● bindDevice（绑定设备，订阅后推送绑定设备通知）</li><li>● deviceAdded（添加新设备，订阅后推送注册设备通知）</li><li>● deviceInfoChanged（设备信息变化，订阅后推送设备信息变化通知）</li><li>● deviceDataChanged（设备数据变化，订阅后推送设备数据变化通知）</li><li>● deviceDatasChanged（设备数据批量变化，订阅后推送批量设备数据变化通知）</li><li>● deviceDeleted（删除设备，订阅后推送删除设备通知）</li><li>● messageConfirm（消息确认，订阅后推送设备消息确认通知）</li><li>● commandRsp（命令响应，订阅后推送设备命令响应通知）</li><li>● deviceEvent（设备事件，订阅后推送设备事件通知）</li><li>● serviceInfoChanged（服务信息变化，订阅后推送设备服务信息变化通知）</li><li>● deviceModelAdded（添加设备模型，订阅后推送增加设备模型通知）</li><li>● deviceModelDeleted（删除设备模型，订阅后推送删除设备模型通知）</li><li>● deviceDesiredPropertiesModifyStatusChanged（设备影子状态变更，订阅后推送设备影子状态变更通知）</li></ul>

参数	必选/ 可选	类型	位置	描述
callbackUrl	必选	String(1024)	body	<p>订阅的回调地址，用于接收对应类型的通知消息。</p> <p>必须使用HTTPS信道回调地址，同时回调地址中必须指定回调地址的端口。例如：https://XXX.XXX.XXX.XXX:443/callbackurltest</p> <p><b>说明</b> HTTP信道只可用于调测。</p>
channel	可选	String(32)	Body	传输通道，若是MQTT客户端订阅，则取值为MQTT，其他情况为HTTP。

## 响应参数

SubscriptionDTO

参数	类型	描述
subscriptionId	String	订阅ID号，用于标识一个订阅。
notifyType	String	通知的类型。
callbackUrl	String	订阅的回调地址。

## 错误码

Http状态码	错误码	错误描述	说明
400	100222	The request callbackurl is illegal.	<p>回调地址非法。</p> <p>处理建议：请检查请求体中的回调地址是否有误。</p>
403	100217	The application hasn't been authorized.	<p>应用未被授权。</p> <p>处理建议：若非应用授权场景，请确认请求参数中的appId为空。</p>
403	1010004	app throttle exceed.	<p>应用调用接口过于频繁，超过流控值（默认值是100次/60s）。</p> <p>处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。</p>
403	1010005	App_key or access_token is invalid.	<p>错误的token信息。</p> <p>处理建议：请检查接口请求中所携带的accessToken是否有误。</p>

Http状态码	错误码	错误描述	说明
409	100227	The resource is conflicted.	资源冲突，通知类型已经被订阅。 处理建议：请检查要订阅的通知类型是否已被订阅。

#### 4.3.4.2 订阅平台管理数据

##### 典型场景

第三方应用可向物联网平台订阅设备相关的管理数据，当物联网平台中设备进行版本升级等相关操作时，平台会向第三方应用发送通知消息，知会其具体的操作状态或操作结果信息。第三方应用可以调用此接口向物联网平台订阅不同类型的设备升级操作通知。

##### 接口功能

支持第三方应用向物联网平台订阅设备的升级操作信息，当设备进行版本升级等相关操作时，平台会向第三方应用发送通知消息。

##### 接口描述

```
def subDeviceManagementData(self, smdInDTO, accessToken)
```

##### 参数说明

参数	必选/可选	类型	位置	描述
smdInDTO	必选	SubDeviceManagementDataInDTO	body	见下表 SubDeviceManagementDataInDTO 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

SubDeviceManagementDataInDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	<p>通知类型。</p> <ul style="list-style-type: none"><li>● swUpgradeStateChangeNotify（软件升级状态变更通知，订阅后推送软件升级状态变更通知）</li><li>● swUpgradeResultNotify（软件升级结果通知，订阅后推送软件升级结果通知）</li><li>● fwUpgradeStateChangeNotify（固件升级状态变更通知，订阅后推送固件升级状态变更通知）</li><li>● fwUpgradeResultNotify（固件升级结果通知，订阅后推送固件升级结果通知）</li></ul>
callbackurl	必选	String	body	<p>订阅的回调地址，用于接收对应类型的的通知消息。</p> <p>必须使用HTTPS信道回调地址，同时回调地址中必须指定回调地址的端口。例如： <a href="https://XXX.XXX.XXX.XXX:443/callbackurltest">https://XXX.XXX.XXX.XXX:443/callbackurltest</a></p> <p><b>说明</b> HTTP信道只可用于调测。</p>

## 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
400	100222	Internal server error.	<p>回调地址非法。</p> <p>处理建议：请检查请求体中的回调地址是否有误。</p>
400	100228	The application input is invalid.	<p>应用输入无效。</p> <p>处理建议：请参考请求参数说明检查接口请求中的参数是否有误。</p>
403	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100229	Get AppKey from header failed.	从消息头获取AppKey失败。
500	100244	register out route fail.	注册路由失败。 处理建议：联系物联网平台维护人员。

#### 4.3.4.3 查询单个订阅

##### 典型场景

第三方应用可在物联网平台上创建多个订阅，以订阅不同类型的设备变更通知。第三方应用若需要查看某个订阅的配置信息，可调用此接口进行查询。

##### 接口功能

支持第三方应用在物联网平台根据订阅ID查询指定订阅的配置信息。

##### 接口描述

```
def querySingleSubscription(self, subscriptionId, appId, accessToken)
```

##### 参数说明

参数	必选/可选	类型	位置	描述
subscriptionId	必选	String	path	订阅ID号，通过调用订阅接口返回或者查询订阅接口获得。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

SubscriptionDTO

参数	类型	描述
subscriptionId	String	订阅ID号，用于标识一个订阅。
notifyType	String	通知的类型。
callbackUrl	String	订阅的回调地址。

## 错误码

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。

### 4.3.4.4 批量查询订阅

#### 典型场景

第三方应用可在物联网平台上创建多个订阅，以订阅不同类型的设备变更通知。第三方应用若需要查看本应用下所有订阅配置，或者查看某种订阅类型下的所有订阅配置，可调用此接口进行查询。

#### 接口功能

支持第三方应用在物联网平台查询本应用下的所有订阅信息，或者某种订阅类型下的所有订阅信息。

#### 接口描述

```
def queryBatchSubscriptions(self, qbsInDTO, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
qbsInDTO	必选	QueryBatchSubInDTO	query	见下表QueryBatchSubInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

### QueryBatchSubInDTO

参数	必选/可选	类型	位置	描述
appId	可选	String(256)	query	设备或者规则等订阅内容主体所属的 appId。
notifyType	可选	String(256)	query	<p>通知类型，第三方应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice (绑定设备)</li><li>● deviceAdded (添加新设备)</li><li>● deviceInfoChanged (设备信息变化)</li><li>● deviceDataChanged (设备数据变化)</li><li>● deviceDatasChanged (设备数据批量变化)</li><li>● deviceDeleted (删除设备)</li><li>● messageConfirm (消息确认)</li><li>● commandRsp (命令响应)</li><li>● deviceEvent (设备事件)</li><li>● serviceInfoChanged (服务信息变化)</li><li>● deviceModelAdded (添加设备模型)</li><li>● deviceModelDeleted (删除设备模型)</li><li>● deviceDesiredPropertiesModifyStatusChanged (修改设备影子状态变更)</li><li>● swUpgradeStateChangeNotify (软件升级状态变更通知)</li><li>● swUpgradeResultNotify (软件升级结果通知)</li><li>● fwUpgradeStateChangeNotify (固件升级状态变更通知)</li><li>● fwUpgradeResultNotify (固件升级结果通知)</li></ul>

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer	query	分页查询参数。 ● 值为空时，查询内容不分页 ● 值大于等于0的时分页查询 ● 值等于0时查询第一页
pageSize	可选	Integer	query	分页查询参数，取值大于等于1的整数，缺省值：“10”。

## 响应参数

QueryBatchSubOutDTO

参数	类型	描述
totalCount	long	查询到的订阅记录总数。
pageNo	long	查询的页码。
pageSize	long	查询每页信息的数量。
subscriptions	List<SubscriptionDTO>	订阅信息列表，具体参见 <a href="#">SubscriptionDTO</a> 结构体。

SubscriptionDTO结构体说明：

参数	类型	描述
subscriptionId	String	订阅ID号。
notifyType	String	通知类型。
callbackUrl	String	订阅的回调地址。

## 错误码

Http状态码	错误码	错误描述	说明
400	100224	The resource exceeds 1000, please refinement query conditions.	查到端资源超出1000条，请重新设定查询条件。 处理建议：请缩小查询条件范围。
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

#### 4.3.4.5 删除单个订阅

##### 典型场景

第三方应用若不再需要接收物联网平台推送的某个订阅通知消息时，可调用此接口删除指定的订阅配置，取消订阅。

##### 接口功能

支持第三方应用在物联网平台根据订阅ID删除指定的订阅配置。

##### 接口描述

```
def deleteSingleSubscription(self, subscriptionId, appId, accessToken)
```

##### 参数说明

参数	必选/可选	类型	位置	描述
subscriptionId	必选	String(256)	path	订阅ID号。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

##### 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100225	The resource is not found	找不到资源。 处理建议：请检查订阅ID是否有误。

### 4.3.4.6 批量删除订阅

#### 典型场景

第三方应用若不再需要接收物联网平台推送的订阅通知消息，或者不再需要接收某类订阅通知消息，可调用此接口批量删除订阅配置，取消订阅。

#### 接口功能

支持第三方应用在物联网平台批量删除订阅配置，可删除全部订阅，或删除指定订阅类型的所有订阅，或删除指定回调URL地址的所有订阅。

#### 接口描述

```
def deleteBatchSubscriptions(self, dbsInDTO, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
dbsInDTO	必选	DeleteBatchSubInDTO	body	见下表DeleteBatchSubInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

DeleteBatchSubInDTO

参数	必选/可选	类型	位置	描述
appId	可选	String(256 )	query	设备或者规则等订阅内容主体所属的 appId。
notifyType	可选	String(256 )	query	<p>通知类型，第三方应用可以根据通知类型对消息分别进行处理。</p> <ul style="list-style-type: none"><li>● bindDevice (绑定设备)</li><li>● deviceAdded (添加新设备)</li><li>● deviceInfoChanged (设备信息变化)</li><li>● deviceDataChanged (设备数据变化)</li><li>● deviceDatasChanged (设备数据批量变化)</li><li>● deviceDeleted (删除设备)</li><li>● messageConfirm (消息确认)</li><li>● commandRsp (命令响应)</li><li>● deviceEvent (设备事件)</li><li>● serviceInfoChanged (服务信息变化)</li><li>● deviceModelAdded (添加设备模型)</li><li>● deviceModelDeleted (删除设备模型)</li><li>● deviceDesiredPropertiesModifyStatus Changed (修改设备影子状态变更)</li><li>● swUpgradeStateChangeNotify (软件升级状态变更通知)</li><li>● swUpgradeResultNotify (软件升级结果通知)</li><li>● fwUpgradeStateChangeNotify (固件升级状态变更通知)</li><li>● fwUpgradeResultNotify (固件升级结果通知)</li></ul>
callbackUrl	可选	String(256 )	query	订阅回调的URL地址。

## 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100225	The resource is not found	找不到资源。 处理建议：请检查订阅类型是否有误。

## 4.3.5 消息推送

第三方应用向物联网平台订阅设备信息，当设备信息发生变更时，物联网平台向第三方应用推送此消息，第三方应用根据通知类型对消息分派处理。需要配合[订阅管理](#)接口使用。

### 4.3.5.1 注册设备通知

#### 典型场景

第三方应用在物联网平台订阅了注册设备通知后（订阅的通知类型为deviceAdded），当第三方应用通过“注册直连设备”接口在平台注册设备时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了注册设备通知的第三方应用推送通知消息。

#### 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<code>def handleDeviceAdded(self):</code>
接口所属类	PushMessageReceiver

## 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
request.json	必选	NotifyDeviceAddedDTO	body	见下表NotifyDeviceAddedDTO的描述。

### NotifyDeviceAddedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“deviceAdded”。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	可选	String	body	网关ID，用于唯一标识一个网关设备。
nodeType	必选	String	body	设备类型。 <ul style="list-style-type: none"><li>● “ENDPOINT”</li><li>● “GATEWAY”</li><li>● “UNKNOWN”</li></ul>
deviceInfo	必选	DeviceInfo	body	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。

DeviceInfo结构体说明：

参数	必选/ 可选	类型	位置	描述
nodeId	必选	String(256)	body	<p>设备的唯一标识，通常使用MAC， MAC地址， Serial No或IMEI作为nodeId。</p> <p><b>说明</b> 若使用IMEI作为nodeId，根据不同厂家的芯片有不同填写要求。</p> <ul style="list-style-type: none"><li>● 高通芯片设备的唯一标识为urn:imei:xxxx， xxxx为IMEI号</li><li>● 海思芯片设备的唯一标识为IMEI号</li><li>● 其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
name	可选	String(256)	body	设备名称。
description	可选	String(2048 )	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048 )	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	可选	String(256)	body	设备的型号。
swVersion	可选	String(256)	body	设备的软件版本。 Z-Wave：主版本号.次版本号，如：1.1。
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型，当前支持的协议类型：CoAP，huaweiM2M，Z-Wave，ONVIF，WPS，Hue，WiFi，J808，Gateway，ZigBee，LWM2M。
bridgeId	可选	String(256)	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。

参数	必选/ 可选	类型	位置	描述
status	可选	String	body	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
statusDetail	可选	String(256)	body	设备的状态详情，status条件取值，取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	可选	String	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
supportedSecurity	可选	String	body	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
isSecurity	可选	String	body	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
signalStrength	可选	String(256)	body	设备的信号强度。
sigVersion	可选	String(256)	body	设备的sig版本。
serialNumber	可选	String(256)	body	设备的序列号。
batteryLevel	可选	String(256)	body	设备的电池电量。

[status](#)和[statusDetail](#):

<b>status</b>	<b>statusDetail</b>
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceAdded",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "nodeType": "GATEWAY",  
    "deviceInfo": {  
        "nodeId": "*****",  
        "name": None,  
        "description": None,  
        "manufacturerId": None,  
        "manufacturerName": None,  
        "mac": None,  
        "location": None,  
        "deviceType": None,  
        "model": None,  
        "swVersion": None,  
        "fwVersion": None,  
        "hwVersion": None,  
        "protocolType": None,  
        "bridgeId": None,  
        "status": "OFFLINE",  
        "statusDetail": "NOT_ACTIVE",  
        "mute": None,  
        "supportedSecurity": None,  
    },  
}
```

```
"isSecurity": None,  
"signalStrength": None,  
"sigVersion": None,  
"serialNumber": None,  
"batteryLevel": None  
}  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 4.3.5.2 绑定设备通知

#### 典型场景

第三方应用在物联网平台订阅了绑定设备通知后（订阅的通知类型为bindDevice），当直连设备接入并绑定物联网平台时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了绑定设备通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleBindDevice(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nbdDTO	必选	NotifyBindDeviceDTO	body	见下表NotifyBindDeviceDTO的描述。

## NotifyBindDeviceDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“bindDevice”。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
resultCode	必选	String	body	绑定结果，取值：“expired/succeeded”。
deviceInfo	可选	DeviceInfo	body	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。

DeviceInfo结构体说明：

参数	必选/可选	类型	位置	描述
nodeId	必选	String(256)	body	标识设备的唯一ID。
name	可选	String(256)	body	设备名称。
description	可选	String(2048)	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID，唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048)	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	可选	String(256)	body	设备的型号。

参数	必选/ 可选	类型	位置	描述
swVersion	可选	String(256)	body	设备的软件版本。 Z-Wave: 主版本号.次版本号, 如: 1.1。
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型: Z-Wave, ZigBee, WPS。
bridgeId	可选	String(256)	body	Bridge标识, 表示设备通过哪个Bridge接入物联网平台。
status	可选	String	body	设备的状态, 表示设备是否在线, 取值范围: “ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
statusDetail	可选	String(256)	body	设备的状态详情, status条件取值, 取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	可选	String	body	表示设备是否处于冻结状态, 即设备上报数据时, 平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE” : 冻结状态</li><li>● “FALSE” : 非冻结状态</li></ul>
supportedSecurity	可选	String	body	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE” : 支持安全模式</li><li>● “FALSE” : 不支持安全模式</li></ul>
isSecurity	可选	String	body	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE” : 启用</li><li>● “FALSE” : 未启用</li></ul>
signalStrength	可选	String(256)	body	设备的信号强度。
sigVersion	可选	String(256)	body	设备的sig版本。
serialNumber	可选	String(256)	body	设备的序列号。
batteryLevel	可选	String(256)	body	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "bindDevice",  
    "deviceId": "*****",  
    "resultCode": "succeeded",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturer": "wulian",  
        "deviceType": 90,  
        "model": "90",  
        "mac": "*****",  
        "swVersion": "...",  
        "fwVersion": "...",  
        "hwVersion": "...",  
        "protocolType": "zigbee",  
        "description": "smockdetector",  
        "nodeType": "GATEWAY"
```

```
    }  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 4.3.5.3 设备信息变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备信息变化通知后（订阅的通知类型为 deviceInfoChanged），当设备的配置或状态信息（如设备厂商信息、位置信息、版本信息、在线状态信息等）发生变化时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备信息变化通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleDeviceInfoChanged(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
ndicDTO	必选	NotifyDeviceInfoChangedDTO	body	见下表 NotifyDeviceInfoChangedDTO 的描述。

NotifyDeviceInfoChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型, 取值: “deviceInfoChanged”。
deviceId	必选	String	body	设备ID, 用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID, 用于唯一标识一个网关设备。
deviceInfo	必选	DeviceInfo	body	设备信息, 具体参见 <a href="#">DeviceInfo</a> 结构体。

DeviceInfo结构体说明:

参数	必选/可选	类型	位置	描述
nodeId	必选	String(256)	body	设备的唯一标识, 通常使用MAC, MAC地址, Serial No或IMEI作为nodeId。 <b>说明</b> 若使用IMEI作为nodeId, 根据不同厂家的芯片有不同填写要求。 <ul style="list-style-type: none"><li>高通芯片设备的唯一标识为urn:imei:xxxx, xxxx为IMEI号</li><li>海思芯片设备的唯一标识为IMEI号</li><li>其他厂家芯片的设备唯一标识请联系模组厂家确认。</li></ul>
name	可选	String(256)	body	设备名称。
description	可选	String(2048)	body	设备的描述信息。
manufacturerId	可选	String(256)	body	厂商ID, 唯一标识一个厂商。
manufacturerName	可选	String(256)	body	厂商名称。
mac	可选	String(256)	body	设备的MAC地址。
location	可选	String(2048)	body	设备的位置信息。
deviceType	可选	String(256)	body	设备类型, 大驼峰命名方式, 如MultiSensor、ContactSensor、CameraGateway。
model	可选	String(256)	body	设备的型号。
swVersion	可选	String(256)	body	设备的软件版本。 Z-Wave: 主版本号.次版本号, 如: 1.1。

参数	必选/ 可选	类型	位置	描述
fwVersion	可选	String(256)	body	设备的固件版本。
hwVersion	可选	String(256)	body	设备的硬件版本。
protocolType	可选	String(256)	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	可选	String(256)	body	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
status	可选	String	body	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
statusDetail	可选	String(256)	body	设备的状态详情，status条件取值，取值查看 <a href="#">status和statusDetail</a> 。
mute	可选	String	body	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
supportedSecurity	可选	String	body	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
isSecurity	可选	String	body	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
signalStrength	可选	String(256)	body	设备的信号强度。
sigVersion	可选	String(256)	body	设备的sig版本。
serialNumber	可选	String(256)	body	设备的序列号。
batteryLevel	可选	String(256)	body	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceInfoChanged",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "deviceInfo": {  
        "name": "Sensor_12",  
        "manufacturer": "wulian",  
        "type": 90,  
        "model": "90",  
        "mac": "*****",  
        "swVersion": "...",  
        "fwVersion": "...",  
        "hwVersion": "...",  
        "protocolType": "zigbee",  
        "description": "smock detector"  
    }  
}
```

{}

## 响应示例

```
response:  
Status Code: 200 OK
```

### 4.3.5.4 设备数据变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备数据变化通知后（订阅的通知类型为 deviceDataChanged），当设备上报单个服务属性数据时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备数据变化通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server 与 port 为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过 **PushMessageReceiver** 类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考 IoT 平台北向 API 参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleDeviceDataChanged(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是 request.json

参数	必选/可选	类型	位置	描述
nddcDTO	必选	NotifyDeviceDataChangedDTO	body	见下表 NotifyDeviceDataChangedDTO 的描述。

NotifyDeviceDataChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“deviceDataChanged”。
requestId	可选	String(1-128)	body	消息的序列号，唯一标识该消息。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID，用于唯一标识一个网关设备。
service	必选	DeviceService	body	设备的服务数据，具体参见下表 <a href="#">DeviceService</a> 结构体说明。

DeviceService结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String	body	服务ID。
serviceType	必选	String	body	服务的类型。
data	必选	ObjectNode	body	服务数据信息。
eventTime	必选	String	body	时间发生时间，时间格式yyyymmddThhmissZ，例如“20151212T121212Z”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

{

    "notifyType": "deviceDataChanged",

```
"requestId": "*****",
"deviceId": "*****",
"gatewayId": "*****",
"service": {
    "serviceId": "Brightness",
    "serviceType": "Brightness",
    "data": {
        "brightness": 80
    },
    "eventTime": "20170311T163657Z"
}
```

## 响应示例

```
response:
Status Code: 200 OK
```

### 4.3.5.5 批量设备数据变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备数据批量变化通知后（订阅的通知类型为 deviceDatasChanged），当设备同时上报多个服务属性数据时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备数据批量变化通知的第三方应用推送通知消息。

#### 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	<a href="https://server:port/v1.0.0/messageReceiver">https://server:port/v1.0.0/messageReceiver</a>
回调接口	<b>def handleDeviceDatasChanged(self)</b>

接口所属类	PushMessageReceiver
-------	---------------------

## 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nddscDTO	必选	NotifyDeviceDatasChange dDTO	body	见下表 NotifyDeviceDatasChangedDTO 的描述。

NotifyDeviceDatasChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“deviceDatasChanged”。
requestId	必选	String	body	消息的序列号，唯一标识该消息。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID，用于唯一标识一个网关设备。
services	必选	List<DeviceService>	body	服务列表信息，具体参见 <a href="#">DeviceService</a> 结构体说明。

DeviceService结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String	body	服务ID。
serviceType	必选	String	body	服务类型。
data	必选	ObjectNode	body	服务数据信息。
eventTime	必选	String	body	事件上报时间，时间格式：yyyymmddThhmissZ，例如“20151212T121212Z”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDatasChanged",  
    "requestId": "*****",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
    "service": [  
        {"  
            "serviceId": "Brightness",  
            "serviceType": "Brightness",  
            "data": {  
                "brightness": 80  
            },  
            "eventTime": "20170311T163657Z"  
        },  
        {"  
            "serviceId": "Color",  
            "serviceType": "Color",  
            "data": {  
                "value": "red"  
            },  
            "eventTime": "20170311T163657Z"  
        }]  
}
```

## 响应示例

response:

Status Code: 200 OK

### 4.3.5.6 设备服务信息变化通知

#### 典型场景

第三方应用在物联网平台订阅了设备服务信息变化通知后（订阅的通知类型为 serviceInfoChanged），当平台向设备下发命令修改设备服务信息时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备服务信息变化通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleServiceInfoChanged(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nsicDTO	必选	NotifyServiceInfoChange dDTO	body	见下表 NotifyServiceInfoChangedDTO的 描述。

NotifyServiceInfoChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	Enum	body	通知类型，取值： “serviceInfoChanged”。

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID，用于唯一标识一个网关设备。
serviceId	必选	String	body	设备服务标识。
serviceType	必选	String	body	设备服务类型。
serviceInfo	必选	ServiceInfo	body	设备服务信息，增量上报，具体参见 <a href="#">ServiceInfo</a> 结构体。

ServiceInfo结构体说明：

参数	必选/可选	类型	位置	描述
muteCmds	可选	List<String>	body	屏蔽的设备控制命令列表。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "serviceInfoChanged",  
    "deviceId": "*****",  
    "serviceId": "*****",  
    "serviceType": "*****",  
    "gatewayId": "*****",  
    "serviceInfo":  
    {  
        "muteCmds": "VIDEO_RECORD"
```

```
    }  
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 4.3.5.7 删除设备通知

#### 典型场景

第三方应用在物联网平台订阅了删除设备通知后（订阅的通知类型为deviceDeleted），当设备在物联网平台中被删除后，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了删除设备通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleDeviceDeleted(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nddDTO	必选	NotifyDeviceDeletedDTO	body	见下表NotifyDeviceDeletedDTO的描述。

NotifyDeviceDeletedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“deviceDeleted”。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	body	网关ID，用于唯一标识一个网关设备。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDeleted",  
    "deviceId": "*****",  
    "gatewayId": "*****",  
}
```

## 响应示例

response:

Status Code: 200 OK

### 4.3.5.8 设备消息确认通知

## 典型场景

第三方应用在物联网平台订阅了设备消息确认通知后（订阅的通知类型为 messageConfirm），当平台向设备下发命令，并收到设备返回的命令确认消息（如命令已送达、已执行等）时，平台会向第三方应用推送通知消息。

## 接口功能

支持物联网平台向订阅了设备消息确认通知的第三方应用推送通知消息。

## 注意事项

- 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleMessageConfirm(self)</b>
接口所属类	PushMessageReceiver

## 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nmcDTO	必选	NotifyMessageConfirmDTO	body	见下表 NotifyMessageConfirmDTO的描述。

NotifyMessageConfirmDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“messageConfirm”。
header	必选	MessageConfirmHeader	body	具体参见下表 <b>MessageConfirmHeader</b> 结构说明。
body	必选	ObjectNode	body	根据业务具体定义，确认消息可以携带的状态变化等消息。

MessageConfirmHeader结构说明：

参数	必选/ 可选	类型	位置	描述
requestId	必选	String(1-128)	body	消息的序列号，唯一标识该消息。
from	必选	String(1-128)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>设备发起的请求：“/devices/{deviceId}”</li><li>设备服务发起的请求：“/devices/{deviceId}/services/{serviceId}”</li></ul>
to	必选	String(1-128)	body	表示消息接收者的地址，To就是订阅请求中的From，如第三方应用的userId。
status	必选	String(1-32)	body	命令状态。 <ul style="list-style-type: none"><li>“sent”：已发送</li><li>“delivered”：已送达</li><li>“executed”：已执行</li></ul>
timestamp	必选	String(1-32)	body	时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "messageConfirm",  
    "header": {  
        "requestId": "*****",  
        "from": "*****",  
        "to": "*****",  
        "status": "delivered",  
        "timestamp": "20151212T121212Z"  
    }  
}
```

```
        },  
        "body": {  
        }  
    }
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 4.3.5.9 设备命令响应通知

#### 典型场景

第三方应用在物联网平台订阅了设备命令响应通知后（订阅的通知类型为 commandRsp），当平台向设备下发命令，并收到设备返回的命令响应消息（如命令执行成功或者执行失败，设备要向平台返回指定内容响应）时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备命令响应通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleCommandRsp(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
ncrDTO	必选	NotifyComm andRspDTO	body	见下表NotifyCommandRspDTO 的描述。

## NotifyCommandRspDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型, 取值: “commandRsp”。
header	必选	Command RspHeader	body	具体参见下表 <a href="#">CommandRspHeader</a> 结构说明。
body	必选	ObjectNode	body	响应命令的消息内容。

CommandRspHeader结构说明:

参数	必选/可选	类型	位置	描述
requestId	必选	String(1-128)	body	消息的序列号, 唯一标识该消息。
from	必选	String(1-128)	body	表示消息发布者的地址。 ● 设备发起的请求: “/devices/{deviceId}” ● 设备服务发起的请求: “/devices/{deviceId}/services/{serviceId}”
to	必选	String(1-128)	body	表示消息接收者的地址, To就是订阅请求中的From, 如第三方应用的userId。
deviceId	必选	String	body	设备ID, 用于唯一标识一个设备。
serviceType	必选	String	body	命令所属的服务类型。
method	必选	String(1-128)	body	存放的响应命令, 如: “INVITE-RSP”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "commandRsp",  
    "header": {  
        "requestId": "*****",  
        "from": "*****",  
        "to": "*****",  
        "deviceId": "*****",  
        "serviceType": "Camera",  
        "method": "MUTE_COMMANDS"  
    },  
    "body": {}  
}
```

## 响应示例

response:  
Status Code: 200 OK

### 4.3.5.10 设备事件通知

#### 典型场景

第三方应用在物联网平台订阅了设备事件通知后（订阅的通知类型为deviceEvent），当平台收到设备上报的事件消息时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备事件通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。

3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考 IoT 平台北向 API 参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<code>def handleDeviceEvent(self)</code>
接口所属类	PushMessageReceiver

## 参数说明

入参是 request.json

参数	必选/可选	类型	位置	描述
ndeDTO	必选	NotifyDeviceEventDTO	body	见下表 NotifyDeviceEventDTO 的描述。

### NotifyDeviceEventDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“deviceEvent”。
header	必选	DeviceEventHeader	body	具体参见下表 <b>DeviceEventHeader</b> 结构说明。
body	必选	ObjectNode	body	响应命令的消息内容。

DeviceEventHeader 结构说明：

参数	必选/可选	类型	位置	描述
eventType	必选	String(1-2)	body	事件类型。
from	必选	String(1-28)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● 设备发起的请求：“/devices/{deviceId}”</li><li>● 设备服务发起的请求：“/devices/{deviceId}/services/{serviceId}”</li></ul>

参数	必选/可选	类型	位置	描述
timestamp	必选	String (1-32)	body	时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。
eventTime	必选	String (1-32)	body	事件上报时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
serviceType	必选	String	body	命令所属的服务类型。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceEvent",  
    "header": {  
        "eventType": "*****",  
        "from": "/devices/{deviceId}/services/{serviceId}",  
        "deviceId": "*****",  
        "serviceType": "*****",  
        "timestamp": "20151212T121212Z",  
        "eventTime": "20151212T121212Z",  
    },  
    "body": {  
        "usedPercent": 80  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

### 4.3.5.11 增加设备模型通知

#### 典型场景

第三方应用在物联网平台订阅了设备模型增加通知后（订阅的通知类型为 deviceModelAdded），在物联网平台上新增设备Profile文件时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备模型增加通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleDeviceModelAdded(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
ndmaDTO	必选	NotifyDeviceModelAddedDTO	body	见下表 NotifyDeviceModelAddedDTO 的描述。

NotifyDeviceModelAddedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“deviceModelAdded”。
appId	必选	String	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceType	必选	String	body	设备的类型。
manufacturerName	必选	String	body	增加设备模型的操作者名称。
manufacturerId	必选	String	body	增加设备模型的操作者ID。
model	必选	String	body	设备型号。
protocolType	必选	String	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceModelAdded",  
    "appId": "*****",  
    "deviceType": "*****",  
    "manufacturerName": "wulian",  
    "manufacturerId": "*****",  
    "model": "*****",  
    "protocolType": "zigbee"
```

}

## 响应示例

```
response:  
Status Code: 200 OK
```

### 4.3.5.12 删除设备模型通知

#### 典型场景

第三方应用在物联网平台订阅了设备模型删除通知后（订阅的通知类型为 deviceModelDeleted），在物联网平台上删除设备Profile文件时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备模型删除通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleDeviceModelDeleted(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
ndmdDTO	必选	NotifyDeviceModelDelete dDTO	body	见下表 NotifyDeviceModelDeletedDTO 的描述。

NotifyDeviceModelDeletedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“deviceModelDeleted”。
appId	必选	String	body	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceType	必选	String	body	设备的类型。
manufacturerName	必选	String	body	增加设备模型的操作者名称。
manufacturerId	必选	String	body	增加设备模型的操作者ID。
model	必选	String	body	设备型号。
protocolType	必选	String	body	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceModelAdded",  
    "appId": "*****",  
    "deviceTyp": "*****",  
    "manufacturerName": "*****",  
    "manufacturerId": "*****",  
    "model": "*****",  
    "protocolType": "*****"
```

{}

## 响应示例

```
response:  
Status Code: 200 OK
```

### 4.3.5.13 设备影子状态变更通知

#### 典型场景

第三方应用在物联网平台订阅了设备影子状态变更通知后（订阅的通知类型为 deviceDesiredPropertiesModifyStatusChanged），当平台中的设备影子向设备同步数据成功或失败时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了设备影子状态变更通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台业务数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleDeviceDesiredStatusChanged(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nddscDTO	必选	NotifyDeviceDesiredStatusChangedDTO	body	见下表 NotifyDeviceDesiredStatusChangedDTO 的描述。

NotifyDeviceDesiredStatusChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“deviceDesiredPropertiesModifyStatusChanged”。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
serviceId	必选	String	body	服务ID，用于标识一个服务。
properties	必选	ObjectNode	body	设备影子数据属性
status	必选	String	body	状态，取值“DELIVERED/FAILED”。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "deviceDesiredPropertiesModifyStatusChanged",  
    "deviceId": "*****",  
    "serviceId": "Device",  
    "properties": {  
        "Model Number" : 1,  
        "Serial Number" : 2,  
        "Firmware Version" : "v1.1.0"  
    },  
    "status": "DELIVERED"  
}
```

## 响应示例

response:

Status Code: 200 OK

#### 4.3.5.14 软件升级状态变更通知

##### 典型场景

第三方应用在物联网平台订阅了软件升级状态变更通知后（订阅的通知类型为 swUpgradeStateChangeNotify），当设备进行软件升级，升级任务的状态发生变化时，平台会向第三方应用推送通知消息。

##### 接口功能

支持物联网平台向订阅了软件升级状态变更通知的第三方应用推送通知消息。

##### 注意事项

- 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

##### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleSwUpgradeStateChanged(self)</b>
接口所属类	PushMessageReceiver

##### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nsuscDTO	必选	NotifySwUpgradeStateChangedDTO	body	见下表 NotifySwUpgradeStateChangedDTO 的描述。

NotifySwUpgradeStateChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“swUpgradeStateChangeNotify”。

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	软件升级任务ID。
subOperationId	必选	String	body	软件升级子任务ID。
swUpgradeState	必选	String	body	软件升级状态。 <ul style="list-style-type: none"><li>● “downloading”：设备正在下载软件包</li><li>● “downloaded”：设备下载软件包完成</li><li>● “updating”：设备正在进行升级</li><li>● “idle”：设备处于空闲状态</li></ul>

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "swUpgradeStateChangeNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "swUpgradeState": "downloading"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 4.3.5.15 软件升级结果变更通知

#### 典型场景

第三方应用在物联网平台订阅了软件升级结果通知后（订阅的通知类型为 swUpgradeResultNotify），当设备的软件升级任务执行完成时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了软件升级结果通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<b>def handleSwUpgradeResult(self)</b>
接口所属类	PushMessageReceiver

#### 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nsurDTO	必选	NotifySwUp gradeResult DTO	body	见下表 NotifySwUpgradeResultDTO的描述。

NotifySwUpgradeResultDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“swUpgradeResultNotify”。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	软件升级任务ID。
subOperationId	必选	String	body	软件升级子任务ID。
curVersion	必选	String	body	设备当前的软件版本。
targetVersion	必选	String	body	设备要升级的目标软件版本。
sourceVersion	必选	String	body	设备的源软件版本。
swUpgradeResult	必选	String	body	软件升级结果。 <ul style="list-style-type: none"><li>● “SUCCESS”：设备升级成功</li><li>● “FAIL”：设备升级失败</li></ul>
upgradeTime	必选	String	body	升级时长。
resultDesc	必选	String	body	升级结果描述。
errorCode	必选	String	body	设备上报的状态错误码。
description	必选	String	body	错误原因描述。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "swUpgradeResultNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "curVersion": "*****",  
    "targetVersion": "*****",  
    "sourceVersion": "*****",  
    "swUpgradeResult": "FAIL",  
    "upgradeTime": "2018-05-28T10:00:00Z",  
    "resultDesc": "设备升级失败",  
    "errorCode": "E-SWUPGRADE-001",  
    "description": "设备升级失败，原因未知。"}
```

```
"appId":"*****",
"operationId":"*****",
"subOperationId":"*****",
"curVersion":"1.3",
"targetVersion":"1.5",
"sourceVersion":"1.0",
"swUpgradeResult":"SUCCESS",
"upgradeTime":"***",
"resultDesc":"***",
"errorCode":"***",
"description":"***",
}
```

## 响应示例

```
response:  
Status Code: 200 OK
```

### 4.3.5.16 固件升级状态变更通知

#### 典型场景

第三方应用在物联网平台订阅了固件升级状态变更通知后（订阅的通知类型为 fwUpgradeStateChangeNotify），当设备进行固件升级，升级任务的状态发生变化时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了固件升级状态变更通知的第三方应用推送通知消息。

#### 注意事项

- 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的 server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过**PushMessageReceiver**类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

#### 接口描述

回调地址	<a href="https://server:port/v1.0.0/messageReceiver">https://server:port/v1.0.0/messageReceiver</a>
回调接口	<b>def handleFwUpgradeStateChanged(self)</b>

接口所属类	PushMessageReceiver
-------	---------------------

## 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nfuscDTO	必选	NotifyFwUp gradeStateCh angedDTO	body	见下表 NotifyFwUpgradeStateChangedDT O的描述。

NotifyFwUpgradeStateChangedDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“fwUpgradeStateChangeNotify”。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	固件升级任务ID。
subOperation Id	必选	String	body	固件升级子任务ID。
step	必选	String	body	固件升级状态，可取值为“0”、“1”、“2”、“3”。
stepDesc	必选	String	body	升级状态描述。 <ul style="list-style-type: none"><li>● “downloading”：设备正在下载软件包</li><li>● “downloaded”：设备下载软件包完成</li><li>● “updating”：设备正在进行升级</li><li>● “idle”：设备处于空闲状态</li></ul>

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

{

"notifyType":"fwUpgradeStateChangeNotify",

"deviceId":"\*\*\*\*\*",

"appId":"\*\*\*\*\*",

"operationId":"\*\*\*\*\*",

"subOperationId":"\*\*\*\*\*",

"step":"1",

"stepDesc":"downloading"

}

## 响应示例

response:

Status Code: 200 OK

### 4.3.5.17 固件升级结果变更通知

#### 典型场景

第三方应用在物联网平台订阅了固件升级结果通知后（订阅的通知类型为 fwUpgradeResultNotify），当设备的固件升级任务执行完成时，平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台向订阅了固件升级结果通知的第三方应用推送通知消息。

#### 注意事项

1. 应用在[订阅平台管理数据](#)时需要订阅接口描述中的指定回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
2. 应用通过[PushMessageReceiver](#)类并重写回调接口接收推送消息的内容。
3. 如果回调地址不是应用服务器本身的地址，则第三方应用需要自己实现原始回调接口，接口内容请参考IoT平台北向API参考中的消息推送章节。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver
回调接口	<code>def handleFwUpgradeResult(self)</code>
接口所属类	PushMessageReceiver

## 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nfurDTO	必选	NotifyFwUpgradeResultDTO	body	见下表 NotifyFwUpgradeResultDTO的描述。

NotifyFwUpgradeResultDTO

参数	必选/可选	类型	位置	描述
notifyType	必选	String	body	通知类型，取值：“fwUpgradeResultNotify”。
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
appId	必选	String	body	设备所属应用的应用ID。
operationId	必选	String	body	固件升级任务ID。
subOperationId	必选	String	body	固件升级子任务ID。
curVersion	必选	String	body	设备当前的固件版本。
targetVersion	必选	String	body	设备要升级的目标固件版本。
sourceVersion	必选	String	body	设备的源固件版本。
Status	必选	String	body	升级结果。 <ul style="list-style-type: none"><li>● “SUCCESS”</li><li>● “FAIL”</li></ul>
statusDesc	必选	String	body	升级结果描述。 <ul style="list-style-type: none"><li>● “SUCCESS”：设备升级成功</li><li>● “FAIL”：设备升级失败</li></ul>
upgradeTime	必选	String	body	固件升级时长。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "notifyType": "fwUpgradeResultNotify",  
    "deviceId": "*****",  
    "appId": "*****",  
    "operationId": "*****",  
    "subOperationId": "*****",  
    "curVersion": "1.6",  
    "targetVersion": "1.6",  
    "sourceVersion": "1.3",  
    "status": "SUCCESS",  
    "statusDesc": "****",  
    "upgradeTime": "****"  
}
```

## 响应示例

response:

Status Code: 200 OK

### 4.3.5.18 NB 设备命令状态变化通知

#### 典型场景

第三方应用通过物联网平台创建设备命令时，如果设置了callbackurl回调地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达），平台会向第三方应用推送通知消息。

#### 接口功能

支持物联网平台在命令状态发生变化时向第三方应用推送通知消息。

## 注意事项

- 应用在[创建设备命令](#)需要设置回调地址为接口描述中的回调地址，回调地址中的server与port为应用服务器公网地址与服务器配置中指定的端口。
- 应用通过PushMessageReceiver类并重写回调接口接收推送消息的内容。
- 如果回调地址不是应用服务器本身地址，则第三方应用需要自己实现原始回调接口。

## 接口描述

回调地址	https://server:port/v1.0.0/messageReceiver/cmd
回调接口	<b>def handleNBCommandStateChanged(self)</b>
接口所属类	PushMessageReceiver

## 参数说明

入参是request.json

参数	必选/可选	类型	位置	描述
nNBcscDTO	必选	NotifyNBCommandStatusChangedDTO	body	见下表NotifyNBCommandStatusChangedDTO的描述。

NotifyNBCommandStatusChangedDTO

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备ID，用于唯一标识一个设备。
commandId	必选	String	body	命令ID，在创建设备命令时由平台生成的。
result	必选	NBCommandResult	body	详见下表 <b>NBCommandResult</b> 结构说明

NBCommandResult结构说明

参数	必选/可选	类型	位置	描述
resultCode	必选	String	body	命令状态结果。 <ul style="list-style-type: none"><li>● “SENT”， 平台已下发给设备，但还未收到设备响应。</li><li>● “DELIVERED”， 平台收到设备响应， 命令已到达设备。</li><li>● “SUCCESS”， 平台收到命令结果上报， 且上报的结果是成功。</li><li>● “FAIL”， 平台收到命令结果上报， 且上报的结果是失败。</li></ul>
resultDetail	必选	ObjectNode	body	设备上报命令结果上报时， 携带的自定义字段。

## 响应参数

Status Code: 200 OK

## 请求示例

Method: POST

request: {callbackUrl}

Header:

Content-Type:application/json

Body:

```
{  
    "deviceId": "*****",  
    "commandId": "*****",  
    "result": {  
        "resultCode": "DELIVERED",  
        "resultDetail": None  
    }  
}
```

## 响应示例

response:

Status Code: 200 OK

## 4.3.6 命令下发（NB 命令）

### 4.3.6.1 创建设备命令

#### 典型场景

设备Profile文件中定义了物联网平台可向设备下发的命令，第三方应用可调用此接口向设备下发命令，配置或修改设备的服务属性，以实现对设备的控制。

物联网平台有两种命令下发机制：

- 立即下发：平台收到命令后立即下发给设备，保证实时性，不保证串行性。
- 缓存下发：平台收到命令后缓存起来，在设备可达时排队串行下发，平台收到前一个命令的响应（模组自动回复的ACK）才会下发后一个命令，保证串行性，不保证实时性。

#### 接口功能

支持第三方应用向设备下发命令，实现对设备的控制。支持物联网平台立即下发命令和缓存下发命令。

#### 接口描述

```
def postDeviceCommand(self, pdcInDTO, appId, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
pdcInDTO	必选	PostDeviceCommandInDTO	body	见下表 PostDeviceCommandInDTO 的描述。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

#### PostDeviceCommandInDTO

参数	必选/可选	类型	位置	描述
deviceId	必选	String(64)	body	下发命令的设备ID，用于唯一标识一个设备。
command	必选	CommandDTOV4	body	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。

参数	必选/可选	类型	位置	描述
callbackUrl	可选	String(1024)	body	命令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	可选	Integer(>=0)	body	下发命令的有效时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。 如果expireTime设置为0，则无论物联网平台上设置的命令下发模式是什么，该命令都会立即下发给设备（如果设备处于眠状态或者链路已老化，则设备收不到命令，平台没收到设备的响应，该命令最终会超时）。
maxRetransmit	可选	Integer(0~3)	body	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-64)	body	命令对应的服务ID，用于标识一个服务。要与profile中定义的serviceId保持一致。
method	必选	String(1-128)	body	命令服务下具体的命令名称，服务属性等。要与profile中定义的命令名保持一致。
paras	必选	ObjectNode	body	命令参数，jsonString格式，里面是一个个键值对，每个键都是profile中命令的参数名（paraName），具体格式需要应用和设备约定。 若profile中命令未定义参数，则paras值中的参数可不填写，即设置为"paras": {}。

## 响应参数

PostDeviceCommandOutDTO

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	100223	Command counts has reached the upLimit.	缓存的命令数已到达限制。处于PENDING状态的命令条数不超过限定期。默认限定期为20。 处理建议：如当前平台缓存的命令还需执行，请让设备上报数据触发缓存命令的下发；如当前平台缓存的命令有不需要执行的命令，请调用“修改设备命令V4”接口，将命令的状态从PENDING修改为CANCELED。

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	100612	Device is zombie.	设备为僵尸设备。（距离设备上次上线时间超过阈值，默认7天） 处理建议：请把设备上线后再重新下发命令。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 4.3.6.2 查询设备命令

#### 典型场景

第三方应用向设备下发命令后，可调用此接口在物联网平台查询下发命令的状态及内容信息，以了解命令的执行情况。

#### 接口功能

支持第三方应用在物联网平台查询下发命令的状态和内容信息，可查询指定时间段内本应用下的所有下发命令，或者指定设备的所有下发命令。

#### 接口描述

```
def queryDeviceCommand(self, qdcInDTO, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qdcInDTO	必选	QueryDeviceCommandInDTO	query	见下表 QueryDeviceCommandInDTO 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

#### QueryDeviceCommandInDTO

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer(>=0)	query	查询的页码，大于等于0，默认值：“0”。
pageSize	可选	Integer(>=1&&≤1000)	query	查询每页信息的数量，大于等于1，最大值1000，默认值：“1000”。
deviceId	可选	String(64)	query	指定查询命令的设备ID。
startTime	可选	String	query	查询开始时间，查询下发命令时间在 startTime 之后的记录。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
endTime	可选	String	query	查询结束时间，查询下发命令时间在 endTime 之前的记录。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

参数	必选/可选	类型	位置	描述
appId	可选	String	query	如果是本应用的设备命令，此参数值可以填写None，否则填写授权应用的appId。

## 响应参数

QueryDeviceCommandOutDTO

参数	类型	描述
pagination	Pagination	页面信息，具体参见 <a href="#">Pagination</a> 结构体。
data	List<DeviceCommandRespV4>	设备命令列表，具体参见 <a href="#">DeviceCommandRespV4</a> 结构体。

Pagination结构体说明：

参数	类型	描述
pageNo	long	页码。
pageSize	long	每页信息数量。
totalSize	long	记录总数。

DeviceCommandRespV4结构体说明：

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。

参数	类型	描述
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	类型	描述
serviceId	必选	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	必选	String(1-128)	命令服务下具体的命令名称，服务属性等。
paras	可选	Object	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的 appId 是否有误。</li><li>请检查请求路径（url）中传入的 appId 是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的 appId 为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的 deviceId 错误。请确认该 deviceId 是否属于该 appId 或者 deviceId 是否写错。</li><li>Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。</li><li>如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。</li></ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与物联网平台的连接是否正常。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： <ul style="list-style-type: none"><li>请检查该设备的 Profile 文件是否已上传到物联网平台。</li><li>请检查请求参数是否正确，或者 serviceId 在 Profile 文件中是否存在。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议： <ul style="list-style-type: none"><li>请确认 startTime 和 endTime 都不能为空，且 endTime 比 startTime 晚。</li><li>请确认 pageNo 不为空，且 pageNo 大于 0。</li><li>请检查 pageSize 不为空，且 pageSize 大于 1。</li></ul>

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 4.3.6.3 修改设备命令

#### 典型场景

第三方应用向设备下发命令后，如果由于命令排队、设备不在线等原因，物联网平台还未把命令下发给设备执行，此时第三方应用可调用此接口修改该命令的状态信息。当前仅支持把命令状态修改为CANCELED状态，即撤销命令的执行。

## 接口功能

支持第三方应用修改指定命令的状态信息，仅能修改还处于PENDING状态的命令，且当前仅支持修改的命令状态为CANCELED，即撤销命令。

## 接口描述

```
def updateDeviceCommand(self, udcInDTO, deviceCommandId, appId, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
udcInDTO	必选	UpdateDeviceCommandInDTO	body	见下表 UpdateDeviceCommandInDTO的 描述。
deviceCommandId	必选	String	path	要修改的命令ID，在调用创建命令接口后获得。
appId	可选	String	query	如果是本应用的设备，此参数值 可以填写None，否则填写授权应 用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

### UpdateDeviceCommandInDTO

参数	必选/可选	类型	位置	描述
status	必选	String	body	命令执行结果，可选值：“CANCELED”，撤销 命令。

## 响应参数

### UpdateDeviceCommandOutDTO

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发 者可通过该标识来指定哪个应用来调用物联网平台的开 放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。

参数	类型	描述
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
result	ObjectNode	下发命令执行的详细结果。
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTOV4结构体说明：

参数	必选/可选	类型	位置	描述
serviceId	必选	String(1-64)	body	命令对应的服务ID，用于标识一个服务。
method	必选	String(1-128)	body	命令服务下具体的命令名称，服务属性等。
paras	可选	Object	body	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	<p>应用不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	<p>应用未被授权。</p> <p>处理建议：若非应用授权场景，请确认请求参数中的appId为空。</p>
200	100418	The deviceData is not existed.	<p>设备信息不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"><li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
200	100428	The device is not online.	<p>设备不在线。</p> <p>处理建议：请检查设备与物联网平台的连接是否正常。</p>
200	100431	The serviceType is not exist.	<p>服务类型不存在。</p> <p>处理建议：</p> <ul style="list-style-type: none"><li>● 请检查该设备的Profile文件是否已上传到物联网平台。</li><li>● 请检查请求参数是否正确，或者serviceId在Profile文件中是否存在。</li></ul>
200	100434	The device command is not existed.	<p>设备命令不存在。</p> <p>处理建议：请检查接口请求中的设备命令ID是否正确。</p>
200	100435	The device command already canceled, expired or executed, Cannot cancel.	设备命令已经取消，过期或执行，不能取消。

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
500	101016	Get iotws address failed.	获取iotws地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	101017	Get newCallbackUrl from oss failed.	从oss获取新的回调地址失败。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

#### 4.3.6.4 创建设备命令撤销任务

##### 典型场景

第三方应用向设备下发命令后，如果由于命令排队、设备不在线等原因，物联网平台还未把命令下发给设备执行（DEFAULT状态），此时第三方应用可调用此接口撤销指定设备的所有未下发的命令，对于已下发成功的命令不可撤销。

## 接口功能

支持第三方应用创建设备命令撤销任务，该任务用于撤销物联网平台中指定设备ID下所有未下发的命令（命令处于DEFAULT状态）。

## 接口描述

```
def createDeviceCmdCancelTask(self, cdctInDTO, appId, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
cdctInDTO	必选	CreateDeviceCmdCancelTaskInDTO	body	见下表 CreateDeviceCmdCancelTaskInDTO 的描述。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

CreateDeviceCmdCancelTaskInDTO

参数	必选/可选	类型	位置	描述
deviceId	必选	String(1-64)	body	待撤销设备命令的设备ID，撤销任务将会撤销所有向该设备下发的命令。

## 响应参数

CreateDeviceCmdCancelTaskOutDTO

参数	类型	描述
taskId	String(1-64)	设备命令撤销任务的任务ID。
appId	String(1-64)	设备命令撤销任务所属的应用ID。
deviceId	String(1-64)	执行设备命令撤销任务的设备ID。

参数	类型	描述
status	String	撤销任务的任务状态。 <ul style="list-style-type: none"><li>● “WAITTING” 表示等待执行中</li><li>● “RUNNING” 表示撤销任务正在执行</li><li>● “SUCCESS” 表示撤销任务执行成功</li><li>● “FAILED” 表示撤销任务执行失败</li><li>● “PART_SUCCESS” 表示撤销任务部分执行成功。</li></ul>
totalCount	Integer	撤销的设备命令总数。
deviceCommands	List<DeviceCommandRespV4>	撤销的设备命令列表，具体参见 <a href="#">DeviceCommandRespV4</a> 结构体。

DeviceCommandRespV4结构体说明：

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
result	ObjectNode	下发命令执行的详细结果。

参数	类型	描述
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTOV4结构体说明:

参数	必选/可选	类型	描述
serviceId	必选	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	必选	String(1-128)	命令服务下具体的命令名称，服务属性等。
paras	可选	Object	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者 deviceId是否写错。</li><li>Header中携带的appId有误，请确认该 appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

### 4.3.6.5 查询设备命令撤销任务

#### 典型场景

第三方应用创建了设备命令撤销任务后，可调用此接口查询设备命令撤销任务的详细信息和执行状态，以了解撤销任务的执行情况。

#### 接口功能

支持第三方应用在物联网平台查询设备命令撤销任务信息和状态，可指定条件查询单个或多个撤销任务。

#### 接口描述

```
def queryDeviceCmdCancelTask(self, qdcctInDTO, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qdcctInDTO	必选	QueryDeviceCmdCancelTaskInDTO2	query	见下表 QueryDeviceCmdCancelTaskInDTO 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

#### QueryDeviceCmdCancelTaskInDTO

参数	必选/可选	类型	位置	描述
pageNo	可选	Integer(>=0)	query	查询的页码，大于等于0，默认值：“0”。
pageSize	可选	Integer(>=1 && <=1000)	query	查询每页信息的数量，大于等于1，最大值1000，默认值：“1000”。
taskId	可选	String	query	撤销任务的任务ID。
deviceId	可选	String	query	执行设备命令撤销任务的设备ID。
status	可选	String	query	设备命令撤销任务的状态。
startTime	可选	String	query	查询开始时间，查询创建撤销设备命令任务时间在 startTime 之后的记录。 时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

参数	必选/可选	类型	位置	描述
endTime	可选	String	query	查询结束时间，查询创建撤销设备命令任务时间在endTime之前的记录。 时间格式： yyyyMMdd'T'HHmmss'Z'，如 “20151212T121212Z”。
appId	可选	String	query	如果是本应用的设备命令，此参数值可以填写None，否则填写授权应用的appId。

## 响应参数

QueryDeviceCmdCancelTaskOutDTO

参数	类型	描述
pagination	Pagination	页码信息，具体参见 <a href="#">Pagination</a> 结构体。
data	List<DeviceCommandCancelTaskRespV4>	设备命令列表，具体参见 <a href="#">DeviceCommandCancelTaskRespV4</a> 结构体。

Pagination结构体说明：

参数	类型	描述
pageNo	long	查询结果页码。
pageSize	long	每页信息数量。
totalSize	long	记录总数，即查询到的撤销任务中的命令总数。

DeviceCommandCancelTaskRespV4结构体说明：

参数	类型	描述
taskId	String(1-64)	设备命令撤销任务的任务ID。
appId	String(1-64)	设备命令撤销任务所属的应用ID。
deviceId	String(1-64)	设备命令撤销任务指定撤销命令的设备ID。

参数	类型	描述
status	String	撤销任务的任务状态。 <ul style="list-style-type: none"><li>● “WAITTING” 表示等待执行中</li><li>● “RUNNING” 表示撤销任务正在执行</li><li>● “SUCCESS” 表示撤销任务执行成功</li><li>● “FAILED” 表示撤销任务执行失败</li><li>● “PART_SUCCESS” 表示撤销任务部分执行成功。</li></ul>
totalCount	Integer	撤销的设备命令总数。
deviceCommands	List<DeviceCommandRespV4>	设备命令撤销任务信息列表，具体参见 <a href="#">DeviceCommandRespV4</a> 结构体。

DeviceCommandRespV4结构体说明：

参数	类型	描述
commandId	String(1-64)	设备命令ID。
appId	String(1-64)	第三方应用的身份标识，用于唯一标识一个应用。开发者可通过该标识来指定哪个应用来调用物联网平台的开放API。
deviceId	String(1-64)	下发命令的设备ID，用于唯一标识一个设备。
command	CommandDTOV4	下发命令的信息，具体参见 <a href="#">CommandDTOV4</a> 结构体。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时（执行失败，执行成功，超时，发送，已送达）会通知第三方应用。
expireTime	Integer(>=0)	下发命令的超时时间，单位为秒，表示设备命令在创建后expireTime秒内有效，超过这个时间范围后命令将不再下发，如果未设置则默认为48小时（86400s*2）。
status	String	下发命令的状态。 <ul style="list-style-type: none"><li>● “DEFAULT” 表示未下发</li><li>● “EXPIRED” 表示命令已经过期</li><li>● “SUCCESSFUL” 表示命令已经成功执行</li><li>● “FAILED” 表示命令执行失败</li><li>● “TIMEOUT” 表示命令下发执行超时</li><li>● “CANCELED” 表示命令已经被撤销执行</li></ul>
result	ObjectNode	下发命令执行的详细结果。

参数	类型	描述
creationTime	String(20)	命令的创建时间。
executeTime	String(20)	命令执行的时间。
platformIssuedTime	String(20)	平台发送命令的时间。
deliveredTime	String(20)	平台将命令送达到设备的时间。
issuedTimes	Integer(>=0)	平台发送命令的次数。
maxRetransmit	Integer(0~3)	命令下发最大重传次数。

CommandDTOV4结构体说明:

参数	必选/可选	类型	描述
serviceId	必选	String(1-64)	命令对应的服务ID，用于标识一个服务。
method	必选	String(1-128)	命令服务下具体的命令名称，服务属性等。
paras	可选	Object	命令参数的jsonString，具体格式需要应用和设备约定。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径(url)中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100220	Get AppKey from header failed.	获取appKey失败。 处理建议：请检查接口请求的header中是否携带appId。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

## 4.3.7 命令下发（非 NB 命令）

### 4.3.7.1 设备服务调用

#### 典型场景

设备Profile文件中定义了物联网平台可向设备下发的命令，第三方应用可调用此接口向设备下发命令，配置或修改设备的服务属性，以实现对设备的控制。

物联网平台不对设备命令进行缓存而是直接下发命令，设备不在线时命令下发失败。下发命令的具体格式需要第三方应用与设备自定义，物联网平台在接口中直接进行封装，透传。

#### 说明

目前该接口仅用于安装了IoT Agent/AgentLite的网关设备，通过向网关设备下发命令，可对网关下的非直连设备进行控制。

#### 接口功能

支持第三方应用向设备立即下发命令（当前仅支持安装了IoT Agent/AgentLite的网关设备），实现对设备的控制，支持给本应用的设备下发命令。

#### 接口描述

```
def invokeDeviceService(self, commandDTO, deviceId, serviceId, appId, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
commandDTO	必选	CommandDTO	body	见下表 <b>CommandDTO</b> 的描述。
deviceId	必选	String(1-64)	path	设备ID，用于唯一标识一个设备。
serviceId	必选	String(1-64)	path	服务ID，唯一标识一个服务。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <b>鉴权</b> 接口获取的accessToken。

#### CommandDTO

参数	必选/可选	类型	位置	描述
header	必选	CommandNA2CloudHeader	body	具体参见 <a href="#">CommandNA2CloudHeader</a> 结构体。
body	可选	Object	body	消息的消息体。JsonObject 里面是一个个键值对。每个键都是 profile 中命令的参数名 (paraName)。

CommandNA2CloudHeader 结构体说明：

参数	必选/可选	类型	位置	描述
requestId	可选	String(0 -128)	body	用于标识一个命令，不能重复。
mode	必选	Enum	body	是否要确认消息。 <ul style="list-style-type: none"><li>● “NOACK”：不需要确认消息</li><li>● “ACK”：需要确认消息</li><li>● 其它值无效</li></ul>
from	可选	String(1 -28)	body	表示消息发布者的地址。 <ul style="list-style-type: none"><li>● App 发起的请求：“/users/{userId}”</li><li>● 第三方应用发起的请求：“/{serviceName}”</li><li>● 物联网平台发起的请求：“/cloud/{serviceName}”</li></ul>
toType	可选	Enum	body	消息接受者的类型，“CLOUD/GATEWAY”。
to	可选	String(1 -28)	body	消息接收者的地址。
method	必选	String(1 -32)	body	命令名称，如：“DISCOVERY”为发现非直连设备，“REMOVE”为删除非直连设备。
callbackURL	可选	String(1 -024)	body	命令的回调路径。

## 响应参数

InvokeDeviceServiceOutDTO

参数	类型	描述
status	String(128)	命令状态。 <ul style="list-style-type: none"> <li>● “sent”：已发送</li> <li>● “delivered”：已投递，toType为“CLOUD”的请求才有该返回值</li> <li>● “failed”：失败，toType为“CLOUD”的请求才有该返回值</li> </ul>
timestamp	String(128)	发送命令的时间戳，时间格式： <code>yyyyMMdd'T'HHmmss'Z'</code> , 如 <code>“20151212T121212Z”</code> 。
requestId	String(128)	设备命令ID。 <ul style="list-style-type: none"> <li>● toType为“GATEWAY”时，如果请求有requestId则同请求的requestId一致，如果请求无requestId则平台分配一个序列号。</li> <li>● toType为“CLOUD”时，则为空。</li> </ul>

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径（url）中传入的appId是否有误。</li> </ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"> <li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li> <li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li> <li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li> </ul>
200	100428	The device is not online.	设备不在线。 处理建议：请检查设备与网关的连接是否正常。

Http状态码	错误码	错误描述	说明
200	100432	The device command is muted.	设备命令已被禁用。 处理建议：请检查接口请求参数method中的命令是否有误。
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
400	102203	CommandName is invalid.	命令名称无效。 处理建议：请检查接口请求参数method中携带的命令是否有误。
403	100450	The gateway is not online.	网关不在线。 处理建议：请检查网关与物联网平台的连接是否正常。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	100444	The serviceType is not exist.	服务类型不存在。 处理建议：请检查接口请求参数toType中的服务类型是否有误。
500	100001	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	100023	The data in DataBase is abnormal.	数据库异常。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
500	50252	Internal server error.	服务内部处理错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。
503	100501	Congestion occurs, and the current network has been flow-controlled	发生拥塞，当前的网络已经被流控。

## 4.3.8 数据采集

物联网平台支持第三方应用查询设备的基本信息，同时还支持查看设备上报的历史数据，能按时、天，月等维度查看设备上报的历史数据。

### 4.3.8.1 查询单个设备信息

#### 典型场景

已在物联网平台注册的设备，第三方应用若需要查看某个设备的详细信息（如厂商、型号、版本、状态和服务属性等信息），可调用此接口在物联网平台查询获取。

#### 接口功能

支持第三方应用在物联网平台根据设备ID查询指定设备的详细信息，包括配置信息、状态信息、服务属性信息等。

#### 接口描述

```
def querySingleDeviceInfo(self, deviceId, select, appId, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
deviceId	必选	String	path	设备ID，用于唯一标识一个设备，在注册设备时由物联网平台分配获得。
select	可选	String	query	指定查询条件，可选值：“imsi”。
appId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

QuerySingleDeviceInfoOutDTO

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
gatewayId	String(256)	网关ID，用于标识一个网关设备。
nodeType	Enum	节点类型，取值：“ENDPOINT/GATEWAY/UNKNOW”。

参数	类型	描述
createTime	String(256)	创建设备的时间，时间格式：yyyyMMdd'THmmss'Z'，如“20151212T121212Z”。
lastModifiedTime	String(256)	最后修改设备的时间。
deviceInfo	DeviceInfo	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。
services	List<DeviceService>	设备服务列表，具体参见 <a href="#">DeviceService</a> 结构体。

DeviceInfo结构体说明：

参数	类型	描述
nodeId	String(256)	设备的唯一标识。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacturerId	String(256)	厂商ID，唯一标识一个厂商。
manufacturerName	String(256)	厂商名称。
mac	String(256)	设备的MAC地址。
location	String(2048)	设备的位置信息。
deviceType	String(256)	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号。 Z-Wave: ProductType + ProductId, 16 进制格式XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
swVersion	String(256)	设备的软件版本，。 Z-Wave: 主版本号.次版本号，如：1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
protocolType	String(256)	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	String(256)	Bridge标识，表示设备通过哪个Bridge接入物联网平台。

参数	类型	描述
status	String	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
statusDetail	String(256)	设备的状态详情，status条件取值，取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	String	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>
supportedSecurity	String	表示设备是否支持安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：支持安全模式</li><li>● “FALSE”：不支持安全模式</li></ul>
isSecurity	String	表示设备当前是否启用安全模式。 <ul style="list-style-type: none"><li>● “TRUE”：启用</li><li>● “FALSE”：未启用</li></ul>
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。
batteryLevel	String(256)	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceService结构体说明:

参数	类型	描述
serviceId	String(256)	设备的服务标识。
serviceType	String(256)	设备的服务类型。
serviceInfo	ServiceInfo	设备的服务信息，具体参见 <a href="#">ServiceInfo</a> 结构体。
data	ObjectNode(2097152 )	属性值对。
eventTime	String(256)	时间格式: yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

ServiceInfo结构体说明：

参数	类型	描述
muteCmds	List<String>	屏蔽的设备控制命令列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100405	The request parameter is invalid.	请求参数不合法。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
404	100403	The device is not existed.	设备不存在。 处理建议：物联网平台未找到deviceId对应的设备，请检查deviceId是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： <ul style="list-style-type: none"><li>请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li><li>Header中携带的appId有误，请确认该appId下是否有该deviceId。</li><li>如果URL中携带了可选参数appId，请检查该appId是否有误。</li></ul>
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>请检查HTTP请求头域中携带的appId是否有误。</li><li>请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.8.2 批量查询设备信息列表

#### 典型场景

已在物联网平台注册的设备，第三方应用若需要根据条件查看多个设备的详细信息（如厂商、型号、版本、状态和服务属性等信息），可调用此接口在物联网平台查询获取。

#### 接口功能

支持第三方应用在物联网平台根据条件查询多个设备的详细信息，包括配置信息、状态信息、服务属性信息等。

#### 接口描述

```
def queryBatchDevicesInfo(self, qbdiInDTO, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
qbdiInDTO	必选	QueryBatchDevicesInfoInDTO	query	见下表 QueryBatchDevicesInfoInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### QueryBatchDevicesInfoInDTO

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
gatewayId	可选	String	query	网关ID，用于标识一个网关设备。
nodeType	可选	String	query	节点类型，取值：“ENDPOINT/GATEWAY/UNKNOW”。
deviceType	可选	String	query	设备类型。
pageNo	可选	Integer	query	查询的页码。 ● 值为空时查询内容不分页 ● 值大于等于0的整数时分页查询 ● 值等于0时查询第一页
pageSize	可选	Integer	query	查询每页信息的数量，缺省值：“1”。
status	可选	String	query	查询设备的状态。 ● “ONLINE”：在线 ● “OFFLINE”：不在线 ● “ABNORMAL”：异常状态
startTim e	可选	String	query	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”。
endTim e	可选	String	query	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”。

参数	必选/可选	类型	位置	描述
sort	可选	String	query	<p>指定返回记录的排序。</p> <ul style="list-style-type: none"> <li>● “ASC”：按注册设备的时间升序排列</li> <li>● “DESC”：按注册设备的时间降序排列</li> </ul> <p>缺省值：“DESC”。</p>
select	可选	String	query	指定返回记录，可取值：“imsi”。

## 响应参数

QueryBatchDevicesInfoOutDTO

参数	类型	描述
totalCount	long	查询的记录数量。
pageNo	long	查询的页码。
pageSize	long	查询每页信息的数量。
devices	List<QuerySingleDeviceInfoOutDTO>	设备分页列表信息，具体参见 <a href="#">QuerySingleDeviceInfoOutDTO</a> 结构体。

QuerySingleDeviceInfoOutDTO结构体说明：

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。
gatewayId	String(256)	网关ID，用于标识一个网关设备。
nodeType	Enum	节点类型，取值：“ENDPOINT/GATEWAY/UNKNOW”。
createTime	String(256)	创建设备的时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
lastModifie dTime	String(256)	最后修改设备的时间。
deviceInfo	DeviceInfoQueryDT O	设备信息，具体参见 <a href="#">DeviceInfo</a> 结构体。
services	List<DeviceService >	设备服务列表，具体参见 <a href="#">DeviceService</a> 结构体。

DeviceInfo结构体说明：

参数	类型	描述
nodeId	String(256)	标识设备的唯一ID。
name	String(256)	设备名称。
description	String(2048)	设备的描述信息。
manufacturerId	String(256)	厂商ID，唯一标识一个厂商。
manufacturerName	String(256)	厂商名称。
mac	String(256)	设备的MAC地址。
location	String(2048)	设备的位置信息。
deviceType	String(256)	设备类型，大驼峰命名方式，如MultiSensor、ContactSensor、CameraGateway。
model	String(256)	设备的型号。 Z-Wave：ProductType + ProductId，16进制格式XXXX-XXXX 补0对齐，如：001A-0A12，其他协议的格式待定。
swVersion	String(256)	设备的软件版本。 Z-Wave：主版本号.次版本号，如：1.1。
fwVersion	String(256)	设备的固件版本。
hwVersion	String(256)	设备的硬件版本。
protocolType	String(256)	设备使用的协议类型，当前支持的协议类型：CoAP, huaweiM2M, Z-Wave, ONVIF, WPS, Hue, WiFi, J808, Gateway, ZigBee, LWM2M。
bridgeId	String(256)	Bridge标识，表示设备通过哪个Bridge接入物联网平台。
status	String	设备的状态，表示设备是否在线，取值范围：“ONLINE”、“OFFLINE”、“INBOX”、“ABNORMAL”。
statusDetail	String(256)	设备的状态详情，status条件取值，取值查看 <a href="#">status</a> 和 <a href="#">statusDetail</a> 。
mute	String	表示设备是否处于冻结状态，即设备上报数据时，平台是否会管理和保存。 <ul style="list-style-type: none"><li>● “TRUE”：冻结状态</li><li>● “FALSE”：非冻结状态</li></ul>

参数	类型	描述
supportedSecurity	String	表示设备是否支持安全模式。 ● “TRUE”：支持安全模式 ● “FALSE”：不支持安全模式
isSecurity	String	表示设备当前是否启用安全模式。 ● “TRUE”：启用 ● “FALSE”：未启用
signalStrength	String(256)	设备的信号强度。
sigVersion	String(256)	设备的sig版本。
serialNumber	String(256)	设备的序列号。
batteryLevel	String(256)	设备的电池电量。

status和statusDetail:

status	statusDetail
OFFLINE	NONE CONFIGURATION_PENDING
ONLINE	NONE COMMUNICATION_ERROR CONFIGURATION_ERROR BRIDGE_OFFLINE FIRMWARE_UPDATING DUTY_CYCLE NOT_ACTIVE

#### 说明

设备向物联网平台上报设备状态时需要status、statusDetail成对上报。statusDetail建议仅用作显示，不建议用于逻辑判断。

DeviceService结构体说明:

参数	类型	描述
serviceId	String(256)	设备的服务标识。
serviceType	String(256)	设备的服务类型。
serviceInfo	ServiceInfo	设备的服务信息，具体参见 <a href="#">ServiceInfo</a> 结构体。
data	ObjectNode(2097152)	属性值对（Attribute-value pair）。
eventTime	String(256)	时间格式: yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

ServiceInfo结构体说明：

参数	类型	描述
muteCmds	List<String>	屏蔽的设备控制命令列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。
400	100218	The gatewayId and pageNo can't be both null.	网关ID和pageNo不能同时为空。 处理建议：请检查请求参数中gatewayId或pageNo是否填写。
400	100405	The request parameter is invalid.	请求参数不合法。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.8.3 查询设备历史数据

#### 典型场景

在日常运行中，物联网平台会收到并保存设备上报业务数据（设备数据的保存时间可通过“修改设备信息”接口配置，最长保存90天），第三方应用若需要查看某个设备上报到平台的历史数据，可调用此接口查询获取。

#### 接口功能

支持第三方应用根据设备ID，查询指定设备上报到物联网平台的历史数据。

#### 接口描述

```
def queryDeviceDataHistory(self, qddhInDTO, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qddhInDTO	必选	QueryDeviceDataHistoryInDTO	query	见下表QueryDeviceDataHistoryInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

QueryDeviceDataHistoryInDTO

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
deviceId	必选	String	query	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	query	网关ID，用于标识一个网关设备。
serviceId	可选	String	query	设备的服务标识。
property	可选	String	query	服务属性数据。
pageNo	可选	Integer	query	查询的页码。 ● 值为空时查询内容不分页 ● 值大于等于0的整数时分页查询 ● 值等于0时查询第一页
pageSize	可选	Integer	query	查询每页信息的数量，缺省值：“1”。
startTime	可选	String	query	查询产生时间在 startTime 之后的历史数据。 时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
endTime	可选	String	query	查询产生时间在 endTime 之前的历史数据。 时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

## 响应参数

QueryDeviceDataHistoryOutDTO

参数	类型	描述
totalCount	Long	查询的记录数量。
pageNo	Long	查询的页码。
pageSize	Long	查询每页信息的数量。
deviceDataHistoryDTOs	List<DeviceDataHistoryDTO>	设备历史数据列表，具体参见 <a href="#">DeviceDataHistoryDTO</a> 结构体。

DeviceDataHistoryDTO结构体说明：

参数	类型	描述
serviceId	String(256)	设备的服务标识。
deviceId	String(256)	设备ID，用于唯一标识一个设备。
gatewayId	String(256)	网关ID，用于标识一个网关设备。
appId	String(256)	第三方应用的身份标识，用于唯一标识一个应用。
data	JsonObject	设备上报的数据。
timestamp	String(256)	上报数据的时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。

## 错误码

Http状态码	错误码	错误描述	说明
200	100418	The deviceData is not existed.	<p>设备信息不存在。 处理建议：</p> <ul style="list-style-type: none"> <li>● 请求参数中携带的deviceId错误。请确认该deviceId是否属于该appId或者deviceId是否写错。</li> <li>● Header中携带的appId有误，请确认该appId下是否有该deviceId。</li> <li>● 如果URL中携带了可选参数appId，请检查该appId是否有误。</li> </ul>
400	100216	The application input is invalid.	<p>应用输入无效。 处理建议：请参考请求参数说明检查接口请求中的参数是否有误。例如，pageSize是否超过2000。</p>
400	100419	The deviceId and gatewayId can't be both null.	<p>deviceId和gatewayId不能同时为空。 处理建议：请检查请求参数中deviceId和gatewayId是否填写。</p>
403	100203	The application is not existed.	<p>应用不存在。 处理建议：</p> <ul style="list-style-type: none"> <li>● 请检查HTTP请求头域中携带的appId是否有误。</li> <li>● 请检查请求路径(url)中传入的appId是否有误。</li> </ul>

Http状态码	错误码	错误描述	说明
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

#### 4.3.8.4 查询设备影子历史数据

##### 典型场景

第三方应用通过“修改设备影子”接口修改设备影子的配置时，物联网平台会保存修改记录。当第三方应用需要查看设备影子的历史配置记录时，可调用此接口查询获取。

##### 接口功能

支持第三方应用根据设备ID，在物联网平台查询指定设备影子的历史配置数据。

##### 接口描述

```
def queryDeviceDesiredHistory(self, qddhInDTO, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
qddhInDTO	必选	QueryDeviceDesiredHistoryInDTO	query	见下表QueryDeviceDesiredHistoryInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### QueryDeviceDesiredHistoryInDTO

参数	必选/可选	类型	位置	描述
appId	必选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
deviceId	必选	String	query	设备ID，用于唯一标识一个设备。
gatewayId	必选	String	query	网关ID，用于标识一个网关设备。
serviceId	可选	String	query	设备的服务标识。
property	可选	String	query	服务属性数据。
pageNo	可选	Integer	query	查询的页码。 <ul style="list-style-type: none"> <li>● 值为空时查询内容不分页</li> <li>● 值大于等于0的整数时分页查询</li> <li>● 值等于0时查询第一页</li> </ul>
pageSize	可选	Integer	query	查询每页信息的数量，缺省值：“1”。
startTime	可选	String	query	查询产生时间在startTime之后的历史数据。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。
endTime	可选	String	query	查询产生时间在endTime之前的历史数据。时间格式：yyyyMMdd'T'HHmmss'Z'，如“20151212T121212Z”。

## 响应参数

### QueryDeviceDesiredHistoryOutDTO

参数	类型	描述
totalCount	Long	查询的记录数量。
pageNo	Long	查询的页码。
pageSize	Long	查询每页信息的数量。
DeviceDesiredHistoryDTO	List<DeviceDesiredHistoryDTO>	设备历史数据列表，具体参见 <a href="#">DeviceDesiredHistoryDTO</a> 结构体。

DeviceDesiredHistoryDTO结构体说明：

参数	类型	描述
serviceId	String(256)	设备的服务标识。
deviceId	String(256)	设备ID，用于唯一标识一个设备。
gatewayId	String(256)	网关ID，用于标识一个网关设备。
appId	String(256)	第三方应用的身份标识，用于唯一标识一个应用。
desired	JsonObject	设备上报的数据。
timestamp	String(256)	上报数据的时间戳，时间格式： yyyyMMdd'T'HHmmss'Z'，如 “20151212T121212Z”。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
400	100216	The application input is invalid.	应用输入无效。 处理建议：请参考请求参数说明 检查接口请求中的参数是否有误。

Http状态码	错误码	错误描述	说明
400	100419	The deviceId and gatewayId can't be both null.	deviceId和gatewayId不能同时为空。 处理建议：请检查请求参数中deviceId和gatewayId是否填写。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

#### 4.3.8.5 查询设备服务能力

##### 典型场景

第三方应用若需要了解设备可上报哪些服务属性数据，以及设备支持下发哪些命令，可调用此接口在物联网平台查询设备的Profile文件中定义的设备服务能力信息。

## 接口功能

支持第三方应用在物联网平台查询设备的服务属性、设备命令等服务能力信息。

## 接口描述

```
def queryDeviceCapabilities(self, qdcInDTO, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
qdcInDTO	必选	QueryDevice CapabilitiesInDTO	query	见下表 QueryDeviceCapabilitiesInDTO 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

### QueryDeviceCapabilitiesInDTO

参数	是否必须	类型	位置	描述
gatewayId	可选	String	query	网关ID，用于标识一个网关设备。
appId	必选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的 appId。
deviceId	可选	String	query	设备ID，用于唯一标识一个设备。

## 响应参数

### QueryDeviceCapabilitiesOutDTO

参数	类型	描述
deviceCapabilities	List<DeviceCapabilityDTO>	查询结果列表，具体参见 <a href="#">DeviceCapabilityDTO</a> 结构体。

DeviceCapabilityDTO结构体说明：

参数	类型	描述
deviceId	String(256)	设备ID，用于唯一标识一个设备。

参数	类型	描述
serviceCap abilities	List<ServiceCapabilityDTO>	设备的服务能力列表，具体参见 <a href="#">ServiceCapabilityDTO</a> 结构体。

ServiceCapabilityDTO结构体说明：

参数	类型	描述
serviceId	String(256)	设备的服务标识。
serviceType	String(256)	设备的服务类型。
option	String(256)	服务选项。
description	String(10240)	设备服务描述信息。
commands	List<ServiceCommand>	支持的命令名称列表，具体参见 <a href="#">ServiceCommand</a> 结构体。
properties	List<ServiceProperty>	支持的属性名称列表，具体参见 <a href="#">ServiceProperty</a> 结构体。

ServiceCommand结构体说明：

参数	类型	描述
commandName	String(256)	命令名称。
paras	List<ServiceCommandPara>	属性列表，具体参见 <a href="#">ServiceCommandPara</a> 结构体。
responses	List<ServiceCommandResponse>	响应列表，具体参见 <a href="#">ServiceCommandResponse</a> 结构体。

ServiceCommandPara结构体说明：

参数	类型	描述
paraName	String(256)	参数名称。
dataType	String(256)	数据类型。
required	Boolean	是否必选。
min	String	属性最小值。
max	String	属性最大值。

参数	类型	描述
step	Double	步长。
maxLength	Integer	最大长度。
unit	String	单位（符号）。
enumList	List<String>	枚举类型列表。

ServiceCommandResponse结构体说明：

参数	类型	描述
responseName	String(256)	响应名称。
paras	List<ServiceCommandPara>	属性列表，具体参见 <a href="#">ServiceCommandPara</a> 结构体。

ServiceProperty结构体说明：

参数	类型	描述
propertyName	String(256)	属性名称。
dataType	String(256)	数据类型。
required	Boolean	是否必选。
min	String	属性最小值。
max	String	属性最大值。
step	Double	步长。
maxLength	Integer	最大长度。
method	String(256)	访问方法。 <ul style="list-style-type: none"><li>● “R”：可读</li><li>● “W”：可写</li><li>● “E”：可观察</li></ul>
unit	String	单位（符号）。
enumList	List<String>	枚举类型列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	100022	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 4.3.9 设备组管理

### 4.3.9.1 创建设备组

#### 典型场景

第三方应用可调用此接口在物联网平台上创建设备组，并把设备归类到不同的设备组内，进行分组管理。一个设备可以归属到多个设备组内。

在对设备进行某些操作时（如升级设备软固件、批量下发命令等），可通过设备组来指定要进行操作的设备。

## 接口功能

支持第三方应用在物联网平台创建设备组，用于对设备进行分组管理。

## 接口描述

```
def createDeviceGroup(self, cdgInDTO, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
cdgInDTO	必选	CreateDeviceGroupInDTO	body	见下表 <a href="#">CreateDeviceGroupInDTO</a> 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

### CreateDeviceGroupInDTO

参数	必选/可选	类型	位置	描述
name	必选	String(1-50)	body	设备组名称，仅限大小写字母和数字。
description	可选	String(1024)	body	设备组的描述信息。
appId	可选	String(50)	body	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的 appId。
maxDevNum	可选	Integer(>=0)	body	设备组设备最大数量，默认最小值为“0”。当值为0时，表示对设备数量不做限制。
deviceIds	可选	List<String>	body	添加到设备组的设备ID列表。
id	可选	String(1-50)	Body	设备组ID。

## 响应参数

### CreateDeviceGroupOutDTO

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID，由平台自动生成。
appId	String(50)	第三方应用的身份标识，用于唯一标识一个应用。
maxDevNum	Integer (>=0)	设备组设备最大数量限制，当值为“0”时，表示对设备数量不做限制。
curDevNum	Integer	当前设备组内的设备数量。
deviceIds	List<String>	添加到设备组的设备ID列表。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100602	The device group name has been used.	设备组名字已经存在。 处理建议：请修改接口请求中的设备组名称。
200	100607	The devGroup has reached the limit.	设备组数目达到限制。 处理建议：请检查已创建的设备组数量是否已经达到License限定的数量上限。
400	100609	Too much devices to add.	添加太多设备至设备组。 处理建议：请确认deviceIds中的设备ID数量在maxDevNum设置值的范围内。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.9.2 删除设备组

#### 典型场景

若因分组变更，第三方应用不再需要使用某个设备组，且不想在物联网平台上继续保存该设备组信息时，可调用此接口在物联网平台删除指定设备组。

#### 接口功能

支持第三方应用在物联网平台根据设备组ID删除指定设备组。

#### 接口描述

```
def deleteDeviceGroup(self, devGroupId, accessAppId, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
devGroupId	必选	String	path	设备组ID，在增加设备组后由物联网平台返回获得。
accessAppId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId 是否有误。</li><li>● 请检查请求路径（url）中传入的 appId 是否有误。</li></ul>
200	100217	The application hasn't been authorized	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100603	The device group is not existed	设备组不存在。 处理建议：请检查设备组ID是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.9.3 修改设备组

#### 典型场景

若因业务变更需要修改设备组的信息，如设备组名称、设备组的设备数量限制等，第三方应用可调用此接口修改指定设备组的信息。

#### 接口功能

支持第三方应用在物联网平台修改指定设备组的信息。

#### 接口描述

```
def modifyDeviceGroup(self, mdgInDTO, devGroupId, accessAppId, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
mdgInDTO	必选	ModifyDeviceGroupInDTO	body	见下表 <a href="#">ModifyDeviceGroupInDTO</a> 的描述。
devGroupId	必选	String(50)	path	设备组ID，在增加设备组后由物联网平台返回获得。
accessAppId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

### ModifyDeviceGroupInDTO

参数	必选/可选	类型	位置	描述
name	必选	String(1-50)	body	设备组名称，仅限大小写字母和数字。
description	可选	String(1024)	body	设备组的描述信息。
maxDevNum	可选	Integer(>=0)	body	设备组设备最大数量，默认值为“0”。当值为0时，表示对设备数量不做限制。

## 响应参数

### ModifyDeviceGroupOutDTO

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String(50)	设备组所属的应用ID。
maxDevNum	Integer(>=0)	设备组设备最大数量限制，当值为0时，表示对设备数量不做限制。
curDevNum	Integer	当前设备组内设备数量。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100602	The device group name has been used.	设备组的名字已经存在。 处理建议：请修改接口请求中的设备组名称。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.9.4 查询设备组详情

#### 典型场景

第三方应用可调用此接口查询当前已创建的所有设备组列表信息，以了解当前设备组的分组和使用情况。

#### 接口功能

支持第三方应用在物联网平台查询已创建的所有设备组信息。

#### 接口描述

```
def queryDeviceGroups(self, qdgInDTO, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qdgInDTO	必选	QueryDeviceGroupsInDTO	query	见下表 <b>QueryDeviceGroupsInDTO</b> 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

QueryDeviceGroupsInDTO

参数	必选/可选	类型	位置	描述
accessAppId	可选	String	query	如果是本应用的设备组，此参数值可以填写 None，否则填写授权应用的appId。
pageNo	可选	Integer	query	分页查询参数。 <ul style="list-style-type: none"><li>● 值为空时，查询内容不分页。</li><li>● 值为大于等于0的整数时，分页查询。</li><li>● 值为0时查询第一页。</li></ul>
pageSize	可选	Integer	query	每页设备组记录数量，默认值为“1”。
name	可选	String	query	设备组名称。

#### 响应参数

QueryDeviceGroupsOutDTO

参数	类型	描述
totalCount	long	设备组总数。
pageNo	long	查询结果页码。
pageSize	long	每页设备组记录数量。
list	List<object>	设备组信息详情。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的 appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.9.5 查询指定设备组

#### 典型场景

第三方应用若需要查看某个设备组的信息，以了解该设备组的使用情况，可调用此接口查询指定设备组的信息。

## 接口功能

支持第三方应用在物联网平台根据设备组ID查询指定设备组的信息。

## 接口描述

```
def querySingleDeviceGroup(self, devGroupId, accessAppId, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
devGroupId	必选	String	path	设备组ID，在增加设备组后由物联网平台返回获得。
accessAppId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

## 响应参数

QuerySingleDeviceGroupOutDTO

参数	类型	描述
name	String(50)	设备组名称，仅限大小写字母和数字。
description	String(1024)	设备组的描述信息。
id	String(50)	设备组ID。
appId	String (50)	设备组所属的应用ID。
maxDevNum	Integer (>=0)	设备组设备最大数量限制。
curDevNum	Integer	当前设备组内的设备数量。
creator	String(1-50)	创建设备组的用户名称。

## 错误码

Http状态码	错误码	错误描述	说明
200	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
200	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.9.6 查询指定设备组成员

#### 典型场景

第三方应用若需要查看某个设备组中的设备成员分布情况，可调用此接口查询指定设备组的设备成员列表信息。

#### 接口功能

支持第三方应用在物联网平台根据设备组ID查询指定设备组内设备列表信息。

#### 接口描述

```
def queryDeviceGroupMembers(self, qdgmInDTO, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
qdgmInDTO	必选	QueryDeviceGroupMembersInDTO	query	见下表 <a href="#">QueryDeviceGroupMembersInDTO</a> 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

### QueryDeviceGroupMembersInDTO

参数	必选/可选	类型	位置	描述
devGroup Id	必选	String	query	设备组ID，在增加设备组后由物联网平台返回获得。
accessAppId	可选	String	query	如果是本应用的设备组，此参数值可以填写None，否则填写授权应用的appId。
pageNo	可选	Integer	query	分页查询参数，默认值为“0”。 <ul style="list-style-type: none"> <li>● 值为空时，查询内容不分页。</li> <li>● 值为大于等于0的整数时，分页查询。</li> <li>● 值为0时查询第一页。</li> </ul>
pageSize	可选	Integer(1000)	query	每页设备记录数量，默认值为“10”。

## 响应参数

### QueryDeviceGroupMembersOutDTO

参数	类型	描述
totalCount	long	设备组内设备总数。
pageNo	long	查询结果页码。
pageSize	long	每页设备记录数量。
deviceIds	List<String>	设备组内设备ID列表。

## 错误码

Http状态码	错误码	错误描述	说明
400	107001	The serviceId is not exist.	服务ID不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107002	The properties is empty in database.	设备属性不存在。 处理建议：请检查接口请求中的serviceId是否有误。
400	107003	The request properties is unknown.	设备状态为未知。 处理建议：请检查设备与平台的连接是否正常。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

### 4.3.9.7 增加设备组成员

#### 典型场景

若需要把新增的设备或已有设备增加到某个设备组中，第三方应用可调用此接口向指定设备组添加设备成员。在向设备组添加设备前，建议通过“查询指定设备组”接口查询该设备组的当前设备数量及最大设备数量限制，确保设备组中还有足够的可添加成员数量。

#### 接口功能

支持第三方应用在物联网平台向指定设备组添加设备。

#### 接口描述

```
def addDevicesToGroup(self, dgwdlDTO, accessAppId, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
dgwdlDTO	必选	DeviceGroupWithDeviceListDTO	body	见下表 DeviceGroupWithDeviceListDTO 的描述。
accessAppId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

DeviceGroupWithDeviceListDTO

参数	必选/可选	类型	位置	描述
devGroupId	必选	String(1-50)	body	设备组ID，在增加设备组后由物联网平台返回获得。
deviceIds	必选	List<String>(1000)	body	要添加到设备组的设备ID列表。
accessAppId	可选	String	query	如果是本应用的设备组，此参数值可以填写None，否则填写授权应用的appId。

#### 返回参数

DeviceGroupWithDeviceListDTO

参数	类型	描述
devGroupId	String(1-50)	设备组ID。
deviceIds	List<String>	添加到设备组的设备ID列表。

## 错误码

Http状态码	错误码	错误描述	说明
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议： <ul style="list-style-type: none"><li>● 请检查接口请求中的设备ID是否有误。</li><li>● 请检查设备组是否达到最大设备数量限制。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的Token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

#### 4.3.9.8 删除设备组成员

##### 典型场景

若设备组中的一个或多个设备不再归属于该设备组，第三方应用可调用此接口从设备组删除设备成员。

##### 接口功能

支持第三方应用在物联网平台从指定设备组删除设备。

##### 接口描述

```
def deleteDevicesFromGroup(self, dgwdlDTO, accessAppId, accessToken)
```

##### 参数说明

参数	必选/可选	类型	位置	描述
dgwdlDTO	必选	DeviceGroupWithDeviceListDTO	body	见下表 DeviceGroupWithDeviceListDTO 的描述。
accessAppId	可选	String	query	如果是本应用的设备，此参数值可以填写None，否则填写授权应用的appId。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

## DeviceGroupWithDeviceListDTO

参数	必选/可选	类型	位置	描述
devGroupId	必选	String(1-50)	body	设备组ID，在增加设备组后由物联网平台返回获得。
deviceIds	必选	List<String>(1000)	body	要从设备组删除的设备ID列表。

## 响应参数

void

## 错误码

Http状态码	错误码	错误描述	说明
200	100601	The number of device in the group has reach the max.	设备组内设备达到最大值。 处理建议：请确认当前设备组内的设备数量在maxDevNum设置值的范围内。
200	100603	The device group is not existed.	设备组不存在。 处理建议：请检查设备组ID是否正确。
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议： <ul style="list-style-type: none"><li>● 请检查接口请求中的设备ID是否有误。</li><li>● 请检查设备组是否达到最大设备数量限制。</li></ul>
400	50400	The input is invalid.	输入参数无效。 处理建议：请检查接口调用请求中携带参数的合法性。
403	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径(url)中传入的appId是否有误。</li></ul>
403	100217	The application hasn't been authorized.	应用未被授权。 处理建议：若非应用授权场景，请确认请求参数中的appId为空。

Http状态码	错误码	错误描述	说明
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： <ul style="list-style-type: none"><li>● 请检查HTTP请求头域中携带的appId是否有误。</li><li>● 请检查请求路径（url）中传入的appId是否有误。</li></ul>
500	50252	Internal server error.	服务器运行内部错误。 处理建议：物联网平台内部错误，请联系物联网平台维护人员处理。

## 4.3.10 设备升级

### 4.3.10.1 查询版本包列表

#### 典型场景

在进行设备版本升级前，第三方应用可调用此接口查询已经上传到物联网平台的版本升级包列表信息，确保设备要升级的版本包已上传到平台。

#### 接口功能

支持第三方应用在物联网平台查询已上传的版本包列表信息，可根据条件查询满足要求的版本包。

#### 接口描述

```
def queryUpgradePackageList(self, quplInDTO, accessToken)
```

## 参数说明

参数	必选/可选	类型	位置	描述
quplInDTO	必选	QueryUpgradePackageListInDTO	query	见下表 QueryUpgradePackageListInDTO 的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

### QueryUpgradePackageListInDTO

参数	必选/可选	类型	位置	描述
fileType	可选	String(256)	query	版本包类型。 <ul style="list-style-type: none"> <li>● “firmwarePackage”： 固件包</li> <li>● “softwarePackage”： 软件包</li> </ul>
deviceType	可选	String(256)	query	版本包适用的设备类型。
model	可选	String(256)	query	版本包适用的设备型号。
manufacturerName	可选	String(256)	query	版本包适用的设备厂商名称。
version	可选	String(256)	query	版本包的版本号。
pageNo	可选	Integer	query	分页查询参数， 默认值“0”。 <ul style="list-style-type: none"> <li>● 值为空时， 查询内容不分页。</li> <li>● 值为大于等于0的整数时， 分页查询。</li> <li>● 值为0时查询第一页。</li> </ul>
pageSize	可选	Integer	query	查询结果分页时的每页结果数量， 取值范围1-100， 默认值“10”。

## 响应参数

### QueryUpgradePackageListOutDTO

参数	类型	描述
data	List<QueryUpgradePackageOutDTO>	版本包列表信息，具体参见 <a href="#">QueryUpgradePackageOutDTO</a> 结构体。
pageNo	Integer	查询结果的页码。
pageSize	Integer	查询结果每页的记录数量。
totalCount	Integer	查询结果的总记录数。

QueryUpgradePackageOutDTO结构体说明：

参数	类型	描述
fileId	String	版本包ID。
name	String	版本包名称。
version	String	版本包版本号。
fileType	String	版本包文件类型。 <ul style="list-style-type: none"><li>● “firmwarePackage”： 固件包</li><li>● “softwarePackage”： 软件包</li></ul>
deviceType	String	版本包适用的设备类型。
model	String	版本包适用的设备型号。
manufacturerName	String	版本包适用的设备厂商名称。
protocolType	String	版本包适用的设备协议类型
description	String	版本包的描述信息。
date	String	版本包的生成时间。
uploadTime	String	版本包的上传日期。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中pageNo和pageSize的值是否在正常范围内。

Http状态码	错误码	错误描述	说明
400	123029	pageNo or pageSize beyond the limit.	pageNo或pageSize超出上限。 处理建议：修改pageNo或pageSize为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

### 4.3.10.2 查询指定版本包

#### 典型场景

在进行设备版本升级前，第三方应用可调用此接口在物联网平台查询某个版本升级包的信息，确保设备要升级的版本包已上传到平台。

#### 接口功能

支持第三方应用在物联网平台根据版本包ID查询指定版本包信息，在调用该接口前需要通过“查询版本包列表”接口查询获得版本包ID。

#### 接口描述

```
def queryUpgradePackage(self, fileId, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
fileId	必选	String	path	版本包ID，在上传版本包后获得。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### 响应参数

QueryUpgradePackageOutDTO

参数	类型	描述
fileId	String	版本包ID。
name	String	版本包名称。
version	String	版本包的版本号。
fileType	String	版本包类型。 <ul style="list-style-type: none"><li>● “firmwarePackage”： 固件包</li><li>● “softwarePackage”： 软件包</li></ul>
deviceType	String	版本包适用的设备类型。
model	String	版本包适用的设备型号。
manufacturerName	String	版本包适用的设备厂商名称。
protocolType	String	版本包适用的设备协议类型。
description	String	版本包的描述信息。
date	String	版本包的生成时间。
uploadTime	String	版本包的上传日期。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	123002	Device or package file not found.	设备或包不存在。 处理建议：请检查接口请求中的 fileId 是否正确。

### 4.3.10.3 删除指定版本包

#### 典型场景

对于不需要继续使用和保留的设备版本包，第三方应用可调用此接口在物联网平台上删除指定的版本包。

#### 接口功能

支持第三方应用在物联网平台根据版本包ID删除指定的版本包文件，在调用该接口前需要通过“查询版本包列表”接口查询获得要删除

#### 接口描述

```
def deleteUpgradePackage(self, fileId, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
fileId	必选	String	path	版本包ID，在上传版本包后获得。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

#### 响应参数

void

#### 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。

Http状态码	错误码	错误描述	说明
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

#### 4.3.10.4 创建软件升级任务

##### 典型场景

若需要对设备进行软件版本升级，第三方应用可调用此接口为多个设备创建软件升级任务，升级前请确保目标版本包已上传到物联网平台。当前仅支持对NB-IoT设备进行软件版本升级。

##### 接口功能

支持第三方应用在物联网平台为多个设备进行软件版本升级，当前仅支持对NB-IoT设备进行软件版本升级。

##### 接口描述

```
def createSoftwareUpgradeTask(self, cutInDTO, accessToken)
```

##### 参数说明

参数	必选/可选	类型	位置	描述
cutInDTO	必选	CreateUpgradeTaskInDTO	body	见下表CreateUpgradeTaskInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

##### CreateUpgradeTaskInDTO

参数	必选/可选	类型	位置	描述
fileId	必选	String	body	要升级的目标版本包ID。
targets	必选	OperateDevices	body	要进行升级的目标，具体参见 <a href="#">OperateDevices</a> 结构体。
policy	可选	OperatePolicy	body	升级任务的执行策略，具体参见 <a href="#">OperatePolicy</a> 结构体。

OperateDevices结构体说明:

参数	必选/可选	类型	位置	描述
deviceGroups	可选	List<String>	body	设备组名称列表，最多支持“256”个设备组。 与指定设备列表方式二选一。
deviceType	可选	String	body	设备类型。 指定设备组时，此字段必选。
model	可选	String	body	设备型号。 指定设备组时，此字段必选。
manufacturerName	可选	String	body	设备厂商名称。 指定设备组时，此字段必选。
devices	可选	List<String>	body	设备ID列表，最多支持256个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明:

参数	必选/可选	类型	位置	描述
executeType	必选	String	body	执行类型，默认值为“now”。 ● “now”：现在执行 ● “device_online”：设备上线时执行 ● “custom”：自定义
startTime	可选	String	body	任务执行时间，executeType=custom时必选，时间格式： yyyyMMdd'T'HHmmss'Z'，如： “20151212T121212Z”。
endTime	可选	String	body	任务停止时间，executeType=custom时必选，时间格式： yyyyMMdd'T'HHmmss'Z'，如： “20151212T121212Z”。
retryType	可选	Boolean	body	执行失败是否进行重试，默认不重试。 ● “true”：重试 ● “false”：不重试
retryTimes	可选	Integer	body	重试次数，取值范围1-5，retryType=true时必选。

## 响应参数

CreateUpgradeTaskOutDTO

参数	类型	描述
operationId	String	操作任务ID。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查请求参数的是否有误。
400	123016	The parameter is error, targetversion not match with device.	参数不正确，目标版本与指定设备不匹配。 处理建议：请检查接口请求中deviceType、manufacturerName、model，与参数fileId指定的目标版本包信息是否一致。
400	123019	manufacturerName is null.	厂商名称内容为空。 处理建议：请检查接口请求中的manufacturerName是否为空。
400	123020	deviceType is null	设备类型内容为空。 处理建议：请检查接口请求中的deviceType是否为空。
400	123021	model is null.	设备型号内容为空。 处理建议：请检查接口请求中的model是否为空。
400	123022	deviceGroups and devices cannot be null together	deviceGroups和devices不能同时为空。 处理建议：deviceGroups与devices必须要选择一个进行赋值。
400	123023	deviceGroups and devices cannot be exist together	deviceGroups和devices不能同时存在。 处理建议：deviceGroups与devices只能选择一个进行赋值。

Http状态码	错误码	错误描述	说明
400	123024	The number of deviceGroups or devices reached upper limit	deviceGroups或devices内容数量达到上限。 处理建议：请检查deviceGroups或devices的内容，数量不能超过256个。
400	123025	executeType is error or can not to be null.	executeType有误或不能为空。 处理建议：请检查接口请求中的executeType是否为空或者是是否有误。
400	123026	startTime or endTime is null or error.	startTime或endTime为空或有误。 处理建议：请检查接口请求中的startTime和endTime是否为空或者格式有误。
400	123028	retryTimes is null or beyond the limit.	retryTimes为空或超出上限。 处理建议：请检查接口请求中的retryTimes是否为空，或者不能小于1或者大于5。
400	123032	startTime can not be later than the endTime.	startTime的时间不能比endTime晚。 处理建议：请检查接口请求中的startTime是否晚于endTime。
400	123033	startTime can not be earlier than the now.	startTime的时间不能比当前时间早。 处理建议：请检查接口请求中的startTime是否早于当前时间。
400	123034	endtime must be greater than 5 minutes.	endtime要比startTime大5分钟。 处理建议：请检查接口请求中的startTime和endTime间隔时间必须大于5分钟。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。

Http状态码	错误码	错误描述	说明
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

### 4.3.10.5 创建固件升级任务

#### 典型场景

若需要对设备进行固件版本升级，第三方应用可调用此接口为多个设备创建固件升级任务，升级前请确保目标版本包已上传到物联网平台。当前仅支持对NB-IoT设备进行固件版本升级。

#### 接口功能

支持第三方应用在物联网平台为多个设备进行固件版本升级，当前仅支持对NB-IoT设备进行固件版本升级。

#### 接口描述

```
def createFirmwareUpgradeTask(self, cutInDTO, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
cutInDTO	必选	CreateUpgradeTaskInDTO	body	见下表CreateUpgradeTaskInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### CreateUpgradeTaskInDTO

参数	必选/可选	类型	位置	描述
fileId	必选	String	body	要升级的目标版本包ID。
targets	必选	OperateDevices	body	要进行升级的目标，具体参见 <a href="#">OperateDevices</a> 结构体。
policy	可选	OperatePolicy	body	升级任务的执行策略，具体参见 <a href="#">OperatePolicy</a> 结构体。

OperateDevices结构体说明:

参数	必选/可选	类型	位置	描述
deviceGroups	可选	List<String>	body	设备组名称列表，最多支持“256”个设备组。 与指定设备列表方式二选一。
deviceType	可选	String	body	设备类型。 指定设备组时，此字段必选。
model	可选	String	body	设备型号。 指定设备组时，此字段必选。
manufacturerName	可选	String	body	设备厂商名称。 指定设备组时，此字段必选。
devices	可选	List<String>	body	设备ID列表，最多支持“256”个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明:

参数	必选/可选	类型	位置	描述
executeType	必选	String	body	执行类型，默认值为“now”。 ● “now”：现在执行 ● “device_online”：设备上线时执行 ● “custom”：自定义
startTime	可选	String	body	任务执行时间， executeType=custom时必选，时间格式：yyyyMMdd'T'HHmmss'Z'， 如：“20151212T121212Z”。
endTime	可选	String	body	任务停止时间， executeType=custom时必选，时间格式：yyyyMMdd'T'HHmmss'Z'， 如：“20151212T121212Z”。
retryType	可选	Boolean	body	执行失败是否进行重试，默认不重试。 ● “true”：重试 ● “false”：不重试
retryTimes	可选	Integer	body	重试次数，取值范围1-5， retryType=true时必选。

## 响应参数

CreateUpgradeTaskOutDTO

参数	类型	描述
operationId	String	操作任务ID。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查请求参数的是否有误。
400	123016	The parameter is error, targetversion not match with device.	参数不正确，目标版本与指定设备不匹配。 处理建议：请检查接口请求中deviceType、manufacturerName、model，与参数fileId指定的目标版本包信息是否一致。
400	123019	manufacturerName is null.	厂商名称内容为空。 处理建议：请检查接口请求中的manufacturerName是否为空。
400	123020	deviceType is null	设备类型内容为空。 处理建议：请检查接口请求中的deviceType是否为空。
400	123021	model is null.	设备型号内容为空。 处理建议：请检查接口请求中的model是否为空。
400	123022	deviceGroups and devices cannot be null together	deviceGroups和devices不能同时为空。 处理建议：deviceGroups与devices必须要选择一个进行赋值。
400	123023	deviceGroups and devices cannot be exist together	deviceGroups和devices不能同时存在。 处理建议：deviceGroups与devices只能选择一个进行赋值。

Http状态码	错误码	错误描述	说明
400	123024	The number of deviceGroups or devices reached upper limit	deviceGroups或devices内容数量达到上限。 处理建议：请检查deviceGroups或devices的内容，数量不能超过256个。
400	123025	executeType is error or can not to be null.	executeType有误或不能为空。 处理建议：请检查接口请求中的executeType是否为空或者是是否有误。
400	123026	startTime or endTime is null or error.	startTime或endTime为空或有误。 处理建议：请检查接口请求中的startTime和endTime是否为空或者格式有误。
400	123028	retryTimes is null or beyond the limit.	retryTimes为空或超出上限。 处理建议：请检查接口请求中的retryTimes是否为空，或者不能小于1或者大于5。
400	123032	startTime can not be later than the endTime.	startTime的时间不能比endTime晚。 处理建议：请检查接口请求中的startTime是否晚于endTime。
400	123033	startTime can not be earlier than the now.	startTime的时间不能比当前时间早。 处理建议：请检查接口请求中的startTime是否早于当前时间。
400	123034	endtime must be greater than 5 minutes.	endtime要比startTime大5分钟。 处理建议：请检查接口请求中的startTime和endTime间隔时间必须大于5分钟。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123002	Device or package file not found.	设备或包不存在 处理建议：请检查接口请求中的fileId是否正确。

#### 4.3.10.6 查询指定升级任务结果

##### 典型场景

在创建了设备软件升级任务或固件升级任务后，第三方应用可调用此接口查询某个升级任务详细信息，以查看升级任务的配置信息和执行情况等。

##### 接口功能

支持第三方应用在物联网平台查询单个固件或软件升级任务的详细信息，包括升级任务的配置信息和执行情况等。

##### 接口描述

```
def queryUpgradeTask(self, operationId, accessToken)
```

##### 参数说明

参数	必选/可选	类型	位置	描述
operationId	必选	String	path	操作任务ID，在创建操作任务后由物联网平台返回获得。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

##### 响应参数

QueryUpgradeTaskOutDTO

参数	类型	描述
operationId	String	操作任务ID。
createTime	String	操作任务的创建时间。

参数	类型	描述
startTime	String	操作任务的启动时间。
stopTime	String	操作任务的停止时间。
operateType	String	操作类型。 <ul style="list-style-type: none"><li>● firmware_upgrade</li><li>● software_upgrade</li></ul>
targets	OperateDevices	执行操作的目标设备，具体参见 <a href="#">OperateDevices</a> 结构体。
policy	OperatePolicy	操作执行策略，具体参见 <a href="#">OperatePolicy</a> 结构体。
status	String	操作任务的状态。 <ul style="list-style-type: none"><li>● “wait”：等待</li><li>● “processing”：正在执行</li><li>● “failed”：失败</li><li>● “success”：成功</li><li>● “stop”：停止</li></ul>
staResult	OperationStaResult	操作结果统计，具体参见 <a href="#">OperationStaResult</a> 结构体。
extendPara	JsonString	操作扩展参数，视不同类型的操作不同，具体参见 <a href="#">extendPara</a> 请求参数。

OperateDevices结构体说明：

参数	类型	描述
deviceGroups	List<String>	设备组名称列表，最多支持“256”个设备组。与指定设备列表方式二选一。
deviceType	String	设备类型。 指定设备组时，此字段必选。
model	String	设备型号。 指定设备组时，此字段必选。
manufacturerName	String	设备厂商名称。 指定设备组时，此字段必选。
devices	List<String>	设备ID列表，最多支持“256”个设备。与指定设备组方式二选一。

OperatePolicy结构体说明：

参数	类型	描述
executeType	String	执行类型， 默认值为“now”。 ● “now”：现在执行 ● “device_online”：设备上线时执行 ● “custom”：自定义
startTime	String	任务执行时间， executeType=custom时必选， 时间格式：yyyyMMdd'T'HHmmss'Z'，如：“20151212T121212Z”。
endTime	String	任务停止时间， executeType=custom时必选， 时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
retryType	Boolean	执行失败是否进行重试， 默认不重试。 ● true：重试 ● false：不重试
retryTimes	Integer	重试次数，取值范围1-5， retryType=true时必选。

OperationStaResult结构体说明：

参数	类型	描述
total	Integer(64)	操作设备总数。
wait	Integer(64)	等待操作的设备个数。
processing	Integer(64)	正在操作的设备个数。
success	Integer(64)	操作设成功的设备个数。
fail	Integer(64)	操作失败的设备个数。
stop	Integer(64)	停止操作的设备个数。
timeout	Integer(64)	操作超时失败的设备个数。

对于操作类型为softwareUpgrade和firmwareUpgrade， extendPara请求参数如下：

参数	类型	描述
fileVersion	String	升级的目标版本号。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken 是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的 operationId 内容是否正确。

### 4.3.10.7 查询指定升级任务子任务详情

#### 典型场景

在创建了设备软件升级任务或固件升级任务后，任务中涉及的每个设备的升级是一个子任务（即升级任务中涉及多少个设备，就有多少个子任务）。第三方应用可调用此接口查询某个升级任务中各个子任务的详细信息，以查看子任务的具体执行情况。

#### 接口功能

支持第三方应用在物联网平台查询单个固件或软件升级任务中每个设备的升级执行情况。

#### 接口描述

```
def queryUpgradeSubTask(self, qustInDTO, operationId, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qustInDTO	必选	QueryUpgradeSubTaskInDTO	query	见下表 QueryUpgradeSubTaskInDTO 的描述。

参数	必选/可选	类型	位置	描述
operationId	必选	String	path	操作任务ID，在创建操作任务后由物联网平台返回获得。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的 accessToken。

## QueryUpgradeSubTaskInDTO

参数	必选/可选	类型	位置	描述
subOperationStatus	可选	String	query	子任务状态，不指定，则查询该任务下所有子任务执行详情。 <ul style="list-style-type: none"><li>● “wait”：等待</li><li>● “processing”：正在执行</li><li>● “fail”：失败</li><li>● “success”：成功</li><li>● “stop”：停止</li></ul>
pageNo	可选	Integer	query	分页查询参数，默认值“0”。 <ul style="list-style-type: none"><li>● 值为空时，查询内容不分页。</li><li>● 值为大于等于0的整数时，分页查询。</li><li>● 值为0时查询第一页。</li></ul>
pageSize	可选	Integer	query	查询结果分页时的每页结果数量，取值范围1-100，默认值“10”。

## 响应参数

## QueryUpgradeSubTaskOutDTO

参数	类型	描述
data	list<SubOperationInfo>	子任务列表信息，具体参见。
pageNo	long	查询结果的页码。
pageSize	long	查询结果每页的记录数量。
totalCount	long	查询结果的记录总数。

SubOperationInfo结构体说明：

参数	类型	描述
subOperationId	String	子任务ID。
createTime	String	子任务创建时间。
startTime	String	子任务启动时间。
stopTime	String	子任务停止时间。
operateType	String	操作类型。 <ul style="list-style-type: none"> <li>● “firmware_upgrade”</li> <li>● “software_upgrade”</li> </ul>
deviceId	String	操作设备的设备ID。
status	String	子任务状态。 <ul style="list-style-type: none"> <li>● “wait”：等待</li> <li>● “processing”：正在执行</li> <li>● “fail”：失败</li> <li>● “success”：成功</li> <li>● “stop”：停止</li> </ul>
detailInfo	String	任务状态的详细描述，对于失败场景下为失败原因。
extendInfo	JsonString	任务扩展信息，视不同类型的操作不同。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
400	123029	pageNo or pageSize beyond the limit.	pageNo 或 pageSize 超出上限。 处理建议：修改 pageNo 或 pageSize 为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。

Http状态码	错误码	错误描述	说明
403	1010005	App_key or access_token is invalid.	错误的token信息。 处理建议：请检查接口请求中所携带的accessToken是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的operationId内容是否正确。

### 4.3.10.8 查询升级任务列表

#### 典型场景

第三方应用可调用此接口查询已创建的升级任务列表信息，以了解当前已有的升级任务信息及各个任务的执行情况。

#### 接口功能

支持第三方应用在物联网平台查询升级任务的列表信息，可根据条件查询满足要求的升级任务信息。

#### 接口描述

```
def queryUpgradeTaskList(self, qutlInDTO, accessToken)
```

#### 参数说明

参数	必选/可选	类型	位置	描述
qutlInDTO	必选	QueryUpgradeTaskListInDTO	query	见下表QueryUpgradeTaskListInDTO的描述。
accessToken	必选	String	header	填写 <a href="#">鉴权</a> 接口获取的accessToken。

#### QueryUpgradeTaskListInDTO

参数	必选/可选	类型	位置	描述
operationType	可选	String(256)	query	操作类型。 <ul style="list-style-type: none"><li>“firmware_upgrade”</li><li>“software_upgrade”</li></ul>

参数	必选/可选	类型	位置	描述
operationStatus	可选	String(256)	query	操作任务的状态。 <ul style="list-style-type: none"> <li>● “wait”：等待</li> <li>● “processing”：正在执行</li> <li>● “failed”：失败</li> <li>● “success”：成功</li> <li>● “stop”：停止</li> </ul>
deviceType	可选	String(256)	query	操作任务针对的设备类型。
model	可选	String(256)	query	操作任务针对的设备型号。
manufacturerName	可选	String(256)	query	操作任务针对的设备厂家名称。
deviceId	可选	String(256)	query	操作任务针对的设备ID。
pageNo	可选	Integer	query	分页查询参数，默认值“0”。 <ul style="list-style-type: none"> <li>● 值为空时，查询内容不分页。</li> <li>● 值为大于等于0的整数时，分页查询。</li> <li>● 值为0时查询第一页。</li> </ul>
pageSize	可选	Integer	query	查询结果分页时的每页结果数量，取值范围1-100，默认值“10”。

## 响应参数

QueryUpgradeTaskListOutDTO

参数	类型	描述
data	List<OperationInfo>	任务列表信息，具体参见 <a href="#">OperationInfo</a> 结构体。
pageNo	Integer	查询结果的页码。
pageSize	Integer	查询结果每页的记录数量。
totalCount	Integer	查询结果的总记录数。

OperationInfo结构体说明：

参数	类型	描述
operationId	String	操作任务ID。
createTime	String	操作任务的创建时间。
startTime	String	操作任务的启动时间。
stopTime	String	操作任务的停止时间。
operateType	String	操作类型。 <ul style="list-style-type: none"> <li>● “firmware_upgrade”</li> <li>● “software_upgrade”</li> </ul>
targets	OperateDevices	执行操作的目标设备，具体参见 <a href="#">OperateDevices</a> 结构体。
policy	OperatePolicy	操作执行策略，具体参见 <a href="#">OperatePolicy</a> 结构体。
status	String	操作任务的状态。 <ul style="list-style-type: none"> <li>● “wait”：等待</li> <li>● “processing”：正在执行</li> <li>● “failed”：失败</li> <li>● “success”：成功</li> <li>● “stop”：停止</li> </ul>
staResult	OperationStaResult	操作结果统计，具体参见 <a href="#">OperationStaResult</a> 结构体。
extendPara	JsonString	操作扩展参数，视不同类型的操作不同。

OperateDevices结构体说明：

参数	类型	描述
deviceGroups	List<String>	设备组名称列表，最多支持“256”个设备组。 与指定设备列表方式二选一。
deviceType	String	设备类型。 指定设备组时，此字段必选。
model	String	设备型号。 指定设备组时，此字段必选。
manufacturerName	String	设备厂商名称。 指定设备组时，此字段必选。

参数	类型	描述
devices	List<String>	设备ID列表，最多支持“256”个设备。 与指定设备组方式二选一。

OperatePolicy结构体说明：

参数	类型	描述
executeType	String	执行类型，默认值为“now”。 ● “now”：现在执行 ● “device_online”：设备上线时执行 ● “custom”：自定义
startTime	String	任务执行时间，executeType=custom时必选， 时间格式：yyyyMMdd'T'HHmmss'Z'，如： “20151212T121212Z”。
endTime	String	任务停止时间，executeType=custom时必选， 时间格式：yyyyMMdd'T'HHmmss'Z'，如： “20151212T121212Z”。
retryType	Boolean	执行失败是否进行重试，默认不重试。 ● “true”：重试 ● “false”：不重试
retryTimes	Integer	重试次数，取值范围1-5，retryType=true时必选。

OperationStaResult结构体说明：

参数	类型	描述
total	Integer(64)	操作设备总数。
wait	Integer(64)	等待操作的设备个数。
processing	Integer(64)	正在操作的设备个数。
success	Integer(64)	操作成功的设备个数。
fail	Integer(64)	操作失败的设备个数。
stop	Integer(64)	停止操作的设备个数。
timeout	Integer(64)	操作超时失败的设备个数。

## 错误码

Http状态码	错误码	错误描述	说明
400	120015	Bad request error.	请求错误。 处理建议：请检查接口请求中 fileId 的格式是否正确。
400	123029	pageNo or pageSize beyond the limit.	pageNo 或 pageSize 超出上限。 处理建议：修改 pageNo 或 pageSize 为合法取值。
403	1010004	app throttle exceed.	应用调用接口过于频繁，超过流控值（默认值是100次/60s）。 处理建议：联系物联网平台维护人员调大流控阈值或者控制接口的调用频率。
403	1010005	App_key or access_token is invalid.	错误的 token 信息。 处理建议：请检查接口请求中所携带的 accessToken 是否有误。
404	123009	The requested task does not exist.	查询的任务不存在。 处理建议：请检查接口请求中的 operationId 内容是否正确。

# 5 Agent Lite API 参考(Android)

[开发者必读](#)

[接口列表](#)

[常用数据结构定义](#)

## 5.1 开发者必读

### 1. 概述

IoT Agent Lite在智慧家庭、工业物联网、车联网等领域为智能设备提供了标准接入物联网平台的能力。主要面向IPC、轻量级网关、工业网关、车机等计算能力较强的终端/网关设备。

### 2. 接口全景图

集成开发者通过全景图了解到：Agent Lite提供了哪些功能？每个功能包含哪些接口？接口之间的逻辑关系如何？从而更快速的找到正确的接口来实现具体业务。

功能	接口	说明
直连设备接入	BaseService.init	初始化Agent Lite资源。
	BaseService.destroy	释放Agent Lite资源。
	BindConfig.setConfig	绑定配置。
	BindService.bind	设备绑定。
	HubService.TOPIC_UNBIN DDEVICE	设备解绑定命令接收。
	LoginConfig.setConfig	登录配置。
	LoginService.login	设备登录。
	LoginService.logout	设备登出。

功能	接口	说明
网关管理非直连设备	HubService.addDevice	设备添加。
	HubService.deviceStatusUpdate	设备状态更新。
	HubService.rmvDevice	设备删除。
设备服务数据上报	DataTransService.reportData	设备服务数据上报。
设备命令接收	DataTransService.TOPIC_COMMAND_RECEIVE	设备服务命令接收。

## 5.2 接口列表

Agent Lite对外提供的接口主要包括广播机制、直连设备的接入、非直连设备的添加和删除、设备数据的上报以及设备命令接收接口。

### 5.2.1 广播机制

Agent Lite广播采用android系统的本地广播，用来接收Agent Lite上报的消息。

由于Agent Lite广播采用本地广播，所以注册广播时只能采用动态注册方式。

### 5.2.2 直连设备接入

第三方开发者获得Agent Lite后首先需要将设备接入到物联网平台。

- 直连设备：通过设备绑定、设备登录流程直接接入IoT平台的设备。
- 非直连设备：通过网关设备接入IoT平台的设备。

#### 5.2.2.1 初始化 Agent Lite 资源

##### 接口功能

初始化Agent Lite资源。

##### 接口描述

```
public static boolean init(String workPath, String logPath, Context context);
```

##### 接口所属类

BaseService

## 参数说明

字段	必选/可选	类型	描述
workPath	必选	String	Agent Lite工作路径，用于存放Agent Lite的配置文件与生产的临时文件，工作路径必须有效。
logPath	可选	String	日志路径（若日志路径为空则日志写在工作路径中）。
context	必选	Context	Android应用程序上下文。

## 接口返回值

返回值	描述
true	成功。
false	失败。

## 示例

```
// 开发者调用该接口初始化Agent Lite资源  
BaseService.init("/sdcard/helloWorld", null, context);
```

### 5.2.2.2 释放 Agent Lite 资源

#### 接口功能

调用此接口，Agent Lite会释放申请的所有动态资源（内存、线程等等）。

#### 接口描述

```
public static void destroy();
```

#### 接口所属类

BaseService

## 接口返回值

返回值	描述
true	成功。
false	失败。

## 示例

```
// 开发者调用该接口销毁Agent Lite资源  
BaseService.destroy();
```

### 5.2.2.3 绑定配置

#### 接口功能

在设备绑定前配置平台的地址和端口。

#### 接口描述

```
public static boolean setConfig(int key, String value);
```

#### 接口所属类

BindConfig

#### 参数说明

字段	必选/可选	类型	描述
key	必选	int	设备绑定的配置项。 <ul style="list-style-type: none"><li>● 平台IP: BindConfig.BIND_CONFIG_ADDR。</li><li>● 平台端口: BindConfig.BIND_CONFIG_PORT。</li></ul>
value	必选	String	设置的值。 <ul style="list-style-type: none"><li>● 平台IP: Agent Lite对接平台地址。</li><li>● 平台端口: 8943。</li></ul>

#### 接口返回值

返回值	描述
true	成功。
false	失败。

#### 返回结果

无

## 示例

```
BindConfig.setConfig(BindConfig.BIND_CONFIG_ADDR, "127.0.0.1");  
BindConfig.setConfig(BindConfig.BIND_CONFIG_PORT, "8943");
```

### 5.2.2.4 设备绑定

#### 接口功能

设备第一次接入物联网平台时需要进行绑定操作，上层应用通过调用该接口传入设备序列号或者MAC地址以及设备信息来绑定到物联网平台。

在绑定前需要调用[绑定配置](#)接口设置绑定服务器IP与端口（IoCM服务器地址与端口，Agent Lite会配置默认端口8943）。

##### 说明

设备绑定是指设备第一次接入IoT平台的过程，需要开发者先在IoT平台注册直连设备，之后在设备上发起绑定操作，将设备绑定到IoT平台上。如果未在IoT平台注册该设备，则绑定操作会失败，Agent Lite将会等待一段时间继续尝试。

#### 接口描述

```
public static boolean bind(String verifyCode, IotaDeviceInfo deviceInfo);
```

#### 接口所属类

BindService

#### 参数说明

字段	必选/可选	类型	描述
verifyCode	必选	String	设备绑定验证码。
deviceInfo	必选	IotaDeviceInfo 类说明	设备信息。

#### 接口返回值

返回值	描述
true	成功。
false	失败。

##### 说明

- 此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明绑定成功，绑定成功需要收到BindService.TOPIC\_BINDEVICE\_RSP广播。
- 当前绑定流程的重试策略为：如果绑定失败，则30秒后继续进行重试，如果重试超过5次（总计尝试超过6次），则返回失败，不再进行重试。如果想要重新发起绑定，需要重启设备。

## 返回结果

广播名称	广播参数	成员	描述
TOPIC_BINDDEVICE_RSP	IotaMessage对象 (使用 intent.getSerializableExtra(BindService.BIND_BROADCAST_MSG_IOTAMSG)方法获 取)	BIND_IE_RESULT	绑定结果。
		BIND_IE_DEVICEID	平台分配的逻辑设 备ID。
		BIND_IE_DEVICESECRET	设备接入的鉴权秘 钥。
		BIND_IE_APPID	开发者应用ID。
		BIND_IE_HA_ADDR	HA服务器地址。
		BIND_IE_LVS_ADDR	LVS服务器地址。

## 示例

调用设备绑定接口。

```
BindService.bind(new IotaDeviceInfo("nodeId", "manufacturerId", "Gateway", "model",  
"protocolType"));
```

接收设备绑定响应消息。

```
//当设备成功绑定之后，Agent Lite会返回给UI如下几个参数，需要UI进行持久化存储，设备登录前需要提前进  
行配置  
BroadcastReceiver mBindRsp;  
mBindRsp = new BroadcastReceiver() {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        //Do Something  
        IotaMessage iotaMsg =  
(IotaMessage) intent.getSerializableExtra(BindService.BIND_BROADCAST_MSG_IOTAMSG);  
        int result = iotaMsg.getUInt(BindService.BIND_IE_RESULT, 0);  
        String deviceId = iotaMsg.getString(BindService.BIND_IE_DEVICEID);  
        String Secret = iotaMsg.getString(BindService.BIND_IE_DEVICESECRET);  
        String Appid = iotaMsg.getString(BindService.BIND_IE_APPID);  
        String haAddr = iotaMsg.getString(BindService.BIND_IE_HA_ADDR);  
        String lvsAddr = iotaMsg.getString(BindService.BIND_IE_LVS_ADDR);  
        return;  
    }  
};  
mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);  
IntentFilter filterBind = new IntentFilter(BindService.TOPIC_BINDDEVICE_RSP);  
mLocalBroadcastManager.registerReceiver(mBindRsp, filterBind);
```

### 5.2.2.5 设备解绑定命令接收

#### 接口功能

注册设备解绑定接收广播来接收处理平台下发的直连设备解绑定命令，开发者收到该广播后需要删除直连设备的配置信息并且释放所有资源，下一次重启后需要重新进行绑定。

## 接口描述

```
HubService.TOPIC_UNBINDDEVICE;
```

## 接口所属类

HubService。

## 示例

```
BroadcastReceiver mUnbindRsp;  
mUnbindRsp = new BroadcastReceiver() {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        //Delete config file, free resource  
        return;  
    }  
};  
mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);  
IntentFilter filterUnbind = new IntentFilter(HubService.TOPIC_UNBINDDEVICE);  
mLocalBroadcastManager.registerReceiver(mUnbindRsp, filterUnbind);
```

## 5.2.2.6 参数配置

### 5.2.2.6.1 配置业务参数

## 接口功能

在登录前配置登录所需要的参数。

## 接口描述

```
public static boolean setConfig(int key, String value);
```

## 接口所属类

LoginConfig

## 参数说明

字段	必选/可选	类型	描述
key	必选	int	<p>设备登录的配置项。</p> <ul style="list-style-type: none"><li>● 设备ID: LoginConfig.LOGIN_CONFIG_DEVICEID。</li><li>● AppId: LoginConfig.LOGIN_CONFIG_APPID。</li><li>● 密码: LoginConfig.LOGIN_CONFIG_SECRET。</li><li>● HTTP地址: LoginConfig.LOGIN_CONFIG_IOCM_ADDRESS。</li><li>● HTTP端口: LoginConfig.LOGIN_CONFIG_IOCM_PORT。</li><li>● MQTT地址: LoginConfig.LOGIN_CONFIG_MQTT_ADDR。</li><li>● MQTT端口: LoginConfig.LOGIN_CONFIG_MQTT_PORT。</li></ul>
value	必选	String	<p>设置的值。</p> <ul style="list-style-type: none"><li>● 设备ID: 从绑定成功的广播中获取。</li><li>● AppId: 从绑定成功的广播中获取。</li><li>● 密码: 从绑定成功的广播中获取。</li><li>● HTTP地址: Agent Lite对接平台地址。</li><li>● HTTP端口: 8943。</li><li>● MQTT地址: Agent Lite对接平台地址。</li><li>● MQTT端口: 8883。</li></ul>

## 接口返回值

返回值	描述
true	成功。
false	失败。

## 返回结果

无

## 示例

保存绑定响应消息携带的参数。

```
private void saveBindPara(IotaMessage iotaMsg) {  
    LogUtil.i(this, TAG, "saveBindParaAndGotoLogin");  
    String appId = iotaMsg.getString(BindService.BIND_IE_APPID);  
    String deviceId = iotaMsg.getString(BindService.BIND_IE_DEVICEID);  
    String secret = iotaMsg.getString(BindService.BIND_IE_DEVICESECRET);  
    String haAddress = AgentLiteUtil.get(ConfigName.platformIP);  
  
    saveGatewayInfo(appId, deviceId, secret, haAddress, null);  
}
```

设置登录配置。

```
private void configLoginPara() {  
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_DEVICEID, GatewayInfo.getDeviceID());  
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_APPID, GatewayInfo.getAppID());  
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_SECRET, GatewayInfo.getSecret());  
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_IOCM_ADDR, GatewayInfo.getHaAddress());  
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_IOCM_PORT, "8943");  
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_MQTT_ADDR, GatewayInfo.getHaAddress());  
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_MQTT_PORT, "8883");  
}
```

### 5.2.2.6.2 配置加密算法类型（可选）

## 接口功能

在登录前配置敏感信息加密算法类型，业务可以根据安全级别选择合适的加密算法类型。

## 接口描述

```
public static boolean setAlgType (int type);
```

## 接口所属类

BaseService

## 参数说明

字段	必选/可选	类型	描述
type	必选	int	加密算法类型。 <ul style="list-style-type: none"><li>● 0: AES 256 CBC。</li><li>● 1: AES 128 GCM。</li><li>● 2: AES 256 GCM。</li></ul>

## 接口返回值

返回值	描述
true	成功。

返回值	描述
false	失败。

## 返回结果

无

## 示例

设置加密算法配置。

```
private void configAlgPara() {  
    BaseService.setAlgType(2);  
}
```

## 5.2.2.7 设备登录

### 接口功能

设备在第一次绑定后，或者在设备重启后需要进行登录的流程。

### 接口描述

```
public static boolean login();
```

### 接口所属类

LoginService

### 接口返回值

返回值	描述
true	成功。
false	失败。



#### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明登录成功，登录成功需要收到**LoginService.TOPIC\_LOGIN\_CONNECTED**广播。登录前通过参数配置接口（[配置业务参数](#)）传入所需的登录信息。

## 返回结果

广播名称	广播参数	成员	描述
TOPIC_LOGIN_CONNECTED	IotaMessage对象 (使用 intent.getSerializableExtra(LoginService. LOGIN_BROADCAST_MSG_IE_IOTAMSG)方 法获取)	无	登录成功或重连 成功。
TOPIC_LOGIN_DISCONNECT	IotaMessage对象 (使用 intent.getSerializableExtra(LoginService. LOGIN_BROADCAST_MSG_IE_IOTAMSG)方 法获取)	LOGIN_IE_REASON	登录或重连失败 原因。

## 示例

```
//配置登录参数
LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_DEVICEID, "deviceId");
LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_APPID, "appId");
LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_SECRET, "passWord");
LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_IOMC_ADDR, "haAddr");
LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_IOMC_PORT, "8943");
LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_MQTT_ADDR, "haAddr");
LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_MQTT_PORT, "8883");

//调用登录接口
LoginService.login();
```

然后等待Agent Lite的连接状态广播。

### 建议:

- 在连接成功的处理函数中进行非直连设备状态上报的处理，并且将缓存的上报数据进行上报。
- 在连接断开的处理函数中记录设备断开状态，之后如果有非直连设备上报数据，需要进行缓存，等到连接成功后再进行上报。

```
//接收登录成功响应
BroadcastReceiver mReceiverConnect;
mReceiverConnect = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        //获取IotaMessage
        IotaMessage iotaMsg = (IotaMessage) intent.getSerializableExtra(LoginService.  
LOGIN_BROADCAST_MSG_IE_IOTAMSG);
        //从回调返回的消息句柄中获取当前系统状态
        int status = uspMsg.getInt(LoginService.LOGIN_IE_STATUS, 0);
        //update device states
        ...
        return true;
    }
}
mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);
IntentFilter filterCon= new IntentFilter(LoginService.TOPIC_LOGIN_CONNECTED);
```

```
mLocalBroadcastManager.registerReceiver(mReceiverConnect, filterCon);
//接收登录失败响应
BroadcastReceiver mReceiverDisconnect;
mReceiverDisconnect = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        //获取IotaMessage
        IotaMessage iotaMsg = (IotaMessage) intent.getSerializableExtra(LoginService.LOGIN_BROADCAST_MSG_IE_IOTAMSG);
        //获取响应的错误码
        int reason = iotaMsg.getInt(LoginService.LOGIN_IE_REASON, 0);
        //stop reporting data
        ...
        return true;
    }
}
mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);
IntentFilter filterDiscon= new IntentFilter(LoginService.TOPIC_LOGIN_DISCONNECTED);

mLocalBroadcastManager.registerReceiver(mReceiverDisconnect, filterDiscon);
```

设备登录后，表示该设备已经成功的连接到物联网平台。

连接成功后，如果因为网络或服务器原因导致连接断开，Agent Lite会自动尝试重新连接，并将实时状态通过这两个广播上报给第三方应用。

### 5.2.2.8 设备登出

#### 接口功能

断开与IoT平台的连接。

#### 接口描述

```
public static boolean logout();
```

#### 接口所属类

LoginService

#### 参数说明

无

#### 接口返回值

返回值	描述
true	成功。
false	失败。

#### 返回结果

无

## 示例

```
LoginService.logout();
```

### 5.2.3 网关管理非直连设备

当开发设备为网关设备时，设备需要管理所有非直连设备（传感器设备）的接入与删除，并且记录这些设备ID与对应设备的映射关系。

#### 5.2.3.1 设备添加

##### 接口功能

当有新设备接入网关后，通过调用设备添加接口将非直连设备接入物联网平台，并且获得平台分配的唯一设备逻辑ID。

##### 接口描述

```
public static boolean addDevice(int cookie, IotaDeviceInfo deviceInfo);
```

##### 接口所属类

HubService

##### 参数说明

字段	必选/可选	类型	描述
cookie	可选	int	Cookie有效值为1-65535。
deviceInfo	必选	IotaDeviceInfo类说明	设备信息。

##### 接口返回值

返回值	描述
true	成功。
false	失败。



##### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明添加成功，设备添加成功需要收到TOPIC\_ADDDEV\_RSP广播。

## 返回结果

广播名称	广播参数	成员	描述
TOPIC_ADDDEV_RSP	IotaMessage (使用 intent.getSerializableExtra(HubService.HUB_BROADCAST_I E_IOTAMSG)方法 获取)	HUB_IE_RESULT	添加设备结果。
		HUB_IE_DEVICEID	设备ID, 如果添加成功则 返回设备ID。
		HUB_IE_COOKIE	Cookie有效值为 1-65535。

## 示例

```
//开发者调用该接口进行设备添加
HubService.addDevice(29011, new IotaDeviceInfo("nodeId", "manufacturerId", "deviceType", "model",
"protocolType"));
```

### 结果处理:

```
// java code
// 开发者注册广播接收器对设备添加结果进行相应的处理
BroadcastReceiver mAdddeviceRsp;
mAdddeviceRsp = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        //Do Something
        IotaMessage iotaMsg = (IotaMessage) intent.getSerializableExtra(HubService.HUB_BROADCAST_I  
E_IOTAMSG);
        int result = iotaMsg.getInt(HubService.HUB_IE_RESULT, 0);
        String deviceId = iotaMsg.getString(HubService.HUB_IE_DEVICEID);
        int cookie = iotaMsg.getInt(HubService.HUB_IE_COOKIE, 0);
        return;
    }
};
mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);
IntentFilter filterAddDev = new IntentFilter(HubService.TOPIC_ADDDEV_RSP);
mLocalBroadcastManager.registerReceiver(mAdddeviceRsp, filterAddDev);
```

### 5.2.3.2 设备状态更新

#### 接口功能

通过该接口更新设备的状态信息，包括直连设备与所管理的非直连设备。设备离线、上线均可通过该接口刷新设备状态信息。

直连设备状态通过设备的登录状态进行管理，当直连设备连接断开则表示设备离线，当直连设备连接或重连成功，则表示设备上线，无需通过该接口进行刷新。故建议开发者使用该接口刷新非直连设备的状态。

#### 接口描述

```
public static boolean updateDeviceStatus(int cookie, String deviceId, String status, String
statusDetail);
```

#### 接口所属类

HubService

## 参数说明

字段	必选/可选	类型	描述
cookie	可选	int	Cookie有效值为1-65535。
deviceId	必选	String	设备Id。
status	必选	String	设备状态。 <ul style="list-style-type: none"><li>● 在线: ONLINE。</li><li>● 离线: OFFLINE。</li></ul>

字段	必选/可选	类型	描述
			<p>状态详细信息</p> <p>t</p> <ul style="list-style-type: none"><li>● 无 : N O N E</li><li>● 配置等待 : C O N F I G U R A T I O N _ P E N D I N G</li><li>● 通信错误 : C O</li></ul>

字段	必选/可选	类型	描述
			M M U N I C A T I O N — E R R O R 。 配 置 错 误 ： C O N F I G U R A T I O N — E R R O R 。 桥 接 器 离 线 ：

字段	必选/可选	类型	描述

B R I D G E - O F F L I N E 。 固件升级 : F I R M W A R E - U P D A T I N G 。 循环任务 : D U T Y - C

字段	必选/可选	类型	描述

● Y C L E 。 未 激 活 : N O T - A C T I V E 。

## 接口返回值

返回值	描述
true	成功。
false	失败。

## 返回结果

广播名称	广播参数	成员	描述
TOPIC_DEVSTATUS_UPDATA_RSP	IotaMessage对象 (使用 intent.getSerializableExtra(HubService. HUB_BROADCASTIE_IOTAMSG)方 法获取)	HUB_IE_RESULT	设备状态更新结 果。
		HUB_IE_DEVICEID	设备ID。

## 示例

```
HubService.deviceStatusUpdate(0, deviceId, "ONLINE", "NONE");
```

然后等待命令执行结果:

```
// 开发者注册广播接收器对设备状态更新结果进行相应的处理
BroadcastReceiver mReceiverDevStatus;
mReceiverDevStatus = new BroadcastReceiver() {
@Override
public void onReceive(Context context, Intent intent) {
    //获取IotaMessage
    IotaMessage iotaMsg =
(IotaMessage)intent.getSerializableExtra(HubService.HUB_BROADCAST_IE_IOTAMSG);
    //获取响应的错误码
    String result = iotaMsg.getString(HubService.HUB_IE_RESULT);
    String deviceId = iotaMsg.getString(HubService.HUB_IE_DEVICEID);
    ...
    return true;
}
}
mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);
IntentFilter filterDiscon= new IntentFilter(HubService. TOPIC_DEVSTATUS_UPDATA_RSP);
mLocalBroadcastManager.registerReceiver(mReceiverDevStatus, filterDiscon);
```

### 5.2.3.3 设备删除

#### 接口功能

当有新设备需要从网关移除时，通过调用设备删除接口将非直连设备从物联网平台删除。

#### 接口描述

```
public static boolean rmvDevice(int cookie, String deviceId);
```

#### 接口所属类

HubService

#### 参数说明

字段	必选/可选	类型	描述
cookie	可选	int	Cookie有效值 1-65535。
deviceId	必选	String	设备Id。

#### 接口返回值

返回值	描述
true	成功。
false	失败。



#### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明删除成功，设备删除成功需要收到**TOPIC\_RMVDEV\_RSP**广播。

## 返回结果

广播名称	广播参数	成员	描述
TOPIC_RMVDEV_RSP	IotaMessage (使用 intent.getSerializableExtra(HubService.HUB_BROADCAST_IE_IOTAMSG)方法获 取)	HUB_IE_RESULT	删除设备结果。
		HUB_IE_COOKIE	Cookie有效值为 1-65535。

## 示例

开发者调用该接口进行设备删除。

```
HubService.rmvDevice(122, deviceId);
```

结果处理：

```
// java code
BroadcastReceiver mRmvdeviceRsp;
mRmvdeviceRsp = new BroadcastReceiver() {
@Override
public void onReceive(Context context, Intent intent) {
    //Do Something
    IotaMessage iotaMsg = (IotaMessage) intent.getSerializableExtra(HubService.HUB_BROADCAST_IE_IOTAMSG);
    int result = iotaMsg.getInt(HubService.HUB_IE_RESULT, 0);
    int cookie = iotaMsg.getInt(HubService.HUB_IE_COOKIE, 0);
    return;
}
mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);
IntentFilter filterRmvDev = new IntentFilter(HubService.TOPIC_RMVDEV_RSP);
mLocalBroadcastManager.registerReceiver(mDeldeviceRsp, filterRmvDev);
```

## 5.2.4 设备服务数据上报

### 接口功能

当直连设备（网关）有数据需要上报或者非直连设备上报数据到网关时，网关需要调用设备服务数据上报接口将数据上报到物联网平台。

### 接口描述

```
public static boolean dataReport(int cookie, String requestId, String deviceId, String serviceId,
String serviceProperties);
```

### 接口所属类

DataTransService

## 参数说明

字段	必选/可选	类型	描述
cookie	可选	int	Cookie有效值1-65535。
requestId	必选	String	请求ID，匹配之前平台下发的服务命令。可以从 <a href="#">设备命令接收</a> 的广播中获取requestId。 <ul style="list-style-type: none"><li>主动数据上报：requestId为NULL。</li><li>命令结果上报：当上报的数据匹配到某一次命令请求时，需要填写此次命令请求的请求ID。</li></ul>
deviceId	必选	String	设备ID。
serviceId	必选	String	服务ID。
serviceProperties	必选	String	服务属性。

## 接口返回值

返回值	描述
true	成功。
false	失败。



### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明服务数据上报成功，数据上报成功需要收到**TOPIC\_DATA\_REPORT\_RSP**广播。

## 返回结果

广播名称	广播参数	成员	描述
TOPIC_DATA_REPORT_RSP	IotaMessage (使用 intent.getSerializableExtra(DataTransService. .DATATRANS_BRO ADCAST_IE_IOTAM SG)方法获取)	DATATRANS_IE_ RESULT  DATATRANS_IE_ COOKIE	数据上报结果。  Cookie有效值为 1-65535。

## 示例

用户根据Profile格式使用Json组件拼装服务属性的内容(serviceProperties)。

```
DataTransService.dataReport(1211, NULL, "xxxx_xxxx_xxxx_xxxx", "DoorWindow", "{\"status\":\\\"OPEN\\\"}");
```

### 数据上报结果接收:

```
//开发者注册广播接收器对设备服务数据上报结果进行相应的的处理
BroadcastReceiver mReportDataRsp;
mReportDataRsp = new BroadcastReceiver() {
@Override
public void onReceive(Context context, Intent intent) {
//Do Something
IotaMessage iotaMsg = (IotaMessage) intent.getSerializableExtra(DataTransService.DATATRANS_BROADCAST_IE_IOTAMSG);
int cookie = iotaMsg.getInt(DataTransService.DATATRANS_IE_COOKIE, 0);
int ret = iotaMsg.getInt(DataTransService.DATATRANS_IE_RESULT, 0);
return;
}
};
mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);
IntentFilter filterReportData
= new IntentFilter(DataTransService.TOPIC_DATA_REPORT_RSP);
mLocalBroadcastManager.registerReceiver(mReportDataRsp, filterReportData);
```

## 5.2.5 设备命令接收

### 接口功能

注册设备命令接收广播来接收处理平台下发的控制命令。

### 接口描述

```
DataTransService.TOPIC_COMMAND_RECEIVE;
```

### 参数说明

无

### 返回结果

广播名称	广播参数	成员	描述
TOPIC_COMM AND_RECEIVE	IotaMessage (使用 intent.getSerializableE xtra(DataTransService. DATATRANS_BROA DCAST_IE_IOTAMS G)方法获取)	DATATRANS_IE_R EQUESTID	请求ID。
		DATATRANS_IE_D EVICEID	设备逻辑ID，在设 备添加时由平台分 配。
		DATATRANS_IE_SE RVICEID	服务ID。
		DATATRANS_IE_M ETHOD	服务方法。
		DATATRANS_IE_C MDCONTENT	命令内容。

## 示例

开发者注册广播接收器对设备命令接收结果进行相应的处理。

```
BroadcastReceiver mReceiveCmd;
mReceiveCmd = new BroadcastReceiver() {
@Override
public void onReceive(Context context, Intent intent) {
    //Do Something
    IotaMessage iotaMsg = (IotaMessage) intent.getSerializableExtra(DataTransService.
DATATRANS_BROADCAST_IE_IOTAMSG);
    String requestId = iotaMsg.getString(DataTransService.DATATRANS_IE_REQUESTID);
    String deviceId = iotaMsg.getString(DataTransService.DATATRANS_IE_DEVICEID);
    String serviceId = iotaMsg.getString(DataTransService.DATATRANS_IE_SERVICEID);
    String method = iotaMsg.getString(DataTransService.DATATRANS_IE_METHOD);
    String cmdContent = iotaMsg.getString(DataTransService.DATATRANS_IE_CMDCONTENT);
    if (serviceId.equals("switch"))
    {
        //根据Proflie定义的命令参数，使用Json组件解析cmdContent
        //Send command to Switch
    }

    return;
}
};

mLocalBroadcastManager = LocalBroadcastManager.getInstance(this);
IntentFilter filterReceiveCmd
= new IntentFilter(DataTransService.TOPIC_COMMAND_RECEIVE);
mLocalBroadcastManager.registerReceiver(mReceiveCmd, filterReceiveCmd);
```

## 5.2.6 数据发布

### 接口功能

当直连设备（网关）有数据需要发布或者非直连设备发布数据到网关时，网关需要调用设备服务数据发布接口将数据发布到物联网平台。其与数据上报功能上的区别在于，发布的数据可以自己规定Topic，同时JSON字符串需要自己组装传入接口，SDK只做透传。

### 接口描述

```
mqttDataPub(int cookie, String topic, int qos, byte[] serviceData);
```

### 接口所属类

DataTransService。

### 参数说明

字段	必选/可选	类型	描述
uiCookie	可选	int	Cookie有效值1-65535。
pucTopic	必选	String	发布数据的Topic。
uiQos	必选	int	Mqtt协议的相关参数，一般为“1”。
pbstrServiceData	必选	byte[]	发布数据的包体。

## 接口返回值

返回值	描述
true	成功。
false	失败。



### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明服务数据发布成功，数据发布成功需要收到**DataTransService**发出的通知。

## 实例

开发者调用数据发布接口。

```
DataTransService mqttDataPub (1211, "/huawei/v1/devices/336d9bac-9ebf-44e9-95cf-efac5f05da3a/services/Storage", 1, bstrBody);
```

开发者调用数据发布接口前需要实现AgentLite提供的观察者接口。

```
public class Subscribe implements MyObserver {  
    public Subscribe (Observable dataTransService) {  
        dataTransService.registerObserver (this);  
    }  
    @Override  
    public void update(IotaMessage arg0) {  
        // TODO Auto-generated method stub  
        System.out.println("AgentLiteDataTrans收到通知:" + arg0);  
        int mMsgType = arg0.getMsgType();  
        switch(mMsgType) {  
            //数据上报应答  
            case IodevService.IODEV_MSG_DATA_REPORT_RSP:  
                getDataReportAnswer(arg0);  
                break;  
            //被动接收命令  
            case IodevService.IODEV_MSG_RECEIVE_CMD:  
                getCmdReceive(arg0);  
                break;  
            //MQTT消息推送  
            case IodevService.IODEV_MSG_MQTT_PUB_RSP:  
                //logoutResultAction(iotaMsg);  
                break;  
            case IodevService.IODEV_MSG_MQTT_SUB_RSP:  
                //TopicSubscribeResultAction(iotaMsg);  
                break;  
            case IodevService.IODEV_MSG_MQTT_DATA_RECV_RSP:  
                //DataRecvAction(iotaMsg);  
                break;  
            default:  
                break;  
        }  
    }  
}
```

## 5.3 常用数据结构定义

### 5.3.1 IotaDeviceInfo 类说明

字段	必选/可选	类型	描述
nodeId	必选	String	关键参数，对接平台的网关下设备唯一标识，设备填写，平台用于判重。
name	可选	String	设备名称。
description	可选	String	设备描述。
manufacturerId	必选	String	厂商ID。
manufacturerName	可选	String	厂商名。
mac	可选	String	设备MAC地址。
location	可选	String	设备的位置。
deviceType	必选	String	设备类型。 <b>说明</b> 如果接入设备是网关设备，则“DeviceType”填写为“Gateway”，如果接入传感器是直连设备，则填写为对应的设备类型。
model	必选	String	型号。 <ul style="list-style-type: none"><li>● 直连设备：与Profile中定义的model保持一致即可。</li><li>● Z-Wave设备： ProductType + ProductId，格式（使用16进制）：XXXX-XXXX 补0对齐，如 001A-0A12，其他协议再定。</li></ul>
swVersion	可选	String	软件版本。 Z-Wave：主版本号.次版本号，如：1.1。
fwVersion	可选	String	固件版本。
hwVersion	可选	String	硬件版本。
protocolType	必选	String	协议类型：Z-Wave。
bridgeId	可选	String	表示设备通过哪个Bridge接入平台。
status	可选	String	表示设备是否在线。 <ul style="list-style-type: none"><li>● ONLINE：在线。</li><li>● OFFLINE：离线。</li></ul>

字段	必选/可选	类型	描述
statusDetail	可选	String	<p>状态详情，如果pcStatus不为空，则该参数必选。</p> <p>参数值：</p> <ul style="list-style-type: none"> <li>● 无: NONE。</li> <li>● 配置等待: CONFIGURATION_PENDING。</li> <li>● 通信错误: COMMUNICATION_ERROR。</li> <li>● 配置错误: CONFIGURATION_ERROR。</li> <li>● 桥接器离线: BRIDGE_OFFLINE。</li> <li>● 固件升级: FIRMWARE_UPDATING。</li> <li>● 循环任务: DUTY_CYCLE。</li> <li>● 未激活: NOT_ACTIVE。</li> </ul>
mute	可选	String	<p>表示设备是否被屏蔽。</p> <ul style="list-style-type: none"> <li>● TRUE。</li> <li>● FALSE。</li> </ul>

### 5.3.2 BIND\_IE\_RESULT 信元参数说明

枚举项	枚举值	类型	描述
BIND_RESULT_SUCCESS	0	NA	绑定成功。
BIND_RESULT_DEV_NOT_BIND	1	NA	未扫码。
BIND_RESULT_VERIFYCODE_EXPIRED	2	NA	验证码过期。
BIND_RESULT_FAILED	255	NA	其余失败。

### 5.3.3 LOGIN\_IE\_REASON 信元参数说明

枚举项	枚举值	类型	描述
LOGIN_REASON_NULL	0	NA	无原因。

枚举项	枚举值	类型	描述
LGN_REASON_CONN_CET_ERR	1	NA	连接失败。
LOGIN_REASON_SERVER_BUSY	2	NA	服务器忙。
LOGIN_REASON_AUTH_FAILED	3	NA	鉴权失败、开发者需要停止重新尝试登录。
LOGIN_REASON_NET_UNAVAILABLE	4	NA	网络不可用。
LOGIN_REASON_DEV_ICE_NOEXIST	5	NA	设备不存在、开发者需要停止重新尝试登录。
LOGIN_REASON_DEV_ICE_RMVED	6	NA	设备已删除、开发者需要停止重新尝试登录。
LOGIN_REASON_UNKNOWNN	7	NA	未知原因。

### 5.3.4 HUB\_IE\_RESULT 参数枚举说明

枚举项	枚举值	类型	描述
HUB_RESULT_SUCCESS	0	NA	添加/删除成功。
HUB_RESULT_DEV_ICE_EXIST	1	NA	设备已存在。
HUB_RESULT_DEV_ICE_NOTEXIST	2	NA	设备不存在。
HUB_RESULT_DEV_ICE_FAILED	255	NA	执行失败。

# 6 Agent Lite API 参考(C)

[开发者必读](#)

[接口列表](#)

[常用数据结构定义](#)

[数据类型定义](#)

## 6.1 开发者必读

### 1. 概述

IoT Agent Lite在智慧家庭、工业物联网、车联网等领域为智能设备提供了标准接入物联网平台的能力。主要面向IPC、轻量级网关、工业网关、车机等计算能力较强的终端/网关设备。

### 2. 接口全景图

集成开发者通过全景图了解到：Agent Lite提供了哪些功能？每个功能包含哪些接口？接口之间的逻辑关系如何？从而更快速的找到正确的接口来实现具体业务。

功能	接口	说明
直连设备接入	IOTA_Init	初始化Agent Lite资源。
	IOTA_Destroy	释放Agent Lite资源。
	IOTA_ConfigSetXXX	绑定配置。
	IOTA_Bind	设备绑定。
	IOTA_TOPIC_CMD_UNBIND_RECEIVE	设备解绑定命令接收。
	IOTA_SetConfig	参数配置。
	IOTA_Login	设备登录。

功能	接口	说明
	IOTA_Logout	设备登出。
网关管理非直连设备	IOTA_HubDeviceAdd	设备添加。
	IOTA_DeviceStatusUpdate	设备状态更新。
	IOTA_HubDeviceRemove	设备删除。
设备服务数据上报	IOTA_ServiceDataReport	设备服务数据上报。
设备命令接收	IOTA_TOPIC_SERVICE_COMMAND_RECEIVE/{deviceId}	设备服务命令接收。

## 6.2 接口列表

Agent Lite对外提供的接口主要包括广播机制、直连设备的接入、非直连设备的添加和删除、设备数据的上报、设备命令接收以及Json组件使用说明。

### 6.2.1 广播机制介绍

Agent Lite提供了一套广播机制给第三方开发者，用来接收Agent Lite上报的消息。

#### 订阅广播

```
HW_BroadCastReg(HW_CHAR *pcTopic , PFN_HW_BROADCAST_RECV pfnReceiver);
```

**广播接收处理函数原型：**

```
(*PFN_HW_BROADCAST_RECV)(HW_UINT uiCookie, HW_MSG *pstMsg);
```

此处uiCookie对应于接口中传入的uiCookie，用来匹配业务的请求与响应；如接口中无uiCookie参数，或传入的是无效值，则广播中该参数无意义。

#### 取消订阅广播

```
HW_BroadCastUnreg(HW_CHAR *pcTopic, PFN_HW_BROADCAST_RECV pfnReceiver);
```

#### 从 pstMsg 获取数据的函数

**获取字符串数据：**

```
HW_MsgGetStr(HW_MSG pstMsg, HW_UINT uiTag);
```

**获取无符号整形数据：**

```
HW_MsgGetUInt(HW_MSG pstMsg, HW_UINT uiTag, HW_UINT uiDefault);
```

**获取字节数组数据：**

```
HW_MsgGetByteArray(HW_MSG pstMsg, HW_UINT uiTag);
```

### 6.2.2 直连设备接入

### 6.2.2.1 初始化 Agent Lite 资源

#### 接口功能

初始化Agent Lite资源。

#### 接口描述

```
HW_INT IOTA_Init(const HW_CHAR *pcWorkPath, const HW_CHAR *pcLogPath);
```

#### 参数说明

字段	必选/可选	类型	描述
pcWorkPath	必选	String	Agent Lite工作路径，用于存放Agent Lite的配置文件与生产的临时文件，工作路径必须有效，该参数必须带结束符‘\0’。
pcLogPath	可选	String	日志路径（若日志路径为空则日志写在工作路径中），该参数必须带结束符‘\0’。

#### 接口返回值

参见[2. 函数标准返回值](#)

#### 示例

```
// 开发者调用该接口初始化Agent Lite资源
IOTA_Init(“/usr/data”, HW_NULL);
```

### 6.2.2.2 释放 Agent Lite 资源

#### 接口功能

调用此函数，Agent Lite会释放申请的所有动态资源（内存、线程等等）。

#### 接口描述

```
IOTA_VOID IOTA_Destroy();
```

#### 接口返回值

参见[2. 函数标准返回值](#)

#### 示例

```
// 开发者调用该接口销毁Agent Lite资源
IOTA_Destroy();
```

### 6.2.2.3 绑定配置

#### 接口功能

配置Agent Lite相关参数。

#### 接口描述

```
HW_INT IOTA_ConfigSetStr(HW_INT iItem, HW_CHAR *pValue)  
HW_INT IOTA_ConfigSetUint(HW_INT iItem, HW_UINT uiValue)
```

#### 参数说明

字段	必选/可选	类型	描述
key	必选	HW_UINT	设备绑定的配置项。 <ul style="list-style-type: none"><li>● 平台IP: EN_IOTA_CFG_IOCM_ADDR。</li><li>● 平台端口: EN_IOTA_CFG_IOCM_PORT。</li></ul>
value	必选	HW_CHAR *	设置的值。 <ul style="list-style-type: none"><li>● 平台IP: Agent Lite对接平台地址。</li><li>● 平台端口: 8943。</li></ul>

#### 接口返回值

参见[2. 函数标准返回值](#)

#### 示例

```
// 开发者调用该接口设置参数  
IOTA_ConfigSetStr(EN_IOTA_CONFIG_IOCM_ADDR, "10.0.0.1");  
IOTA_ConfigSetUint(EN_IOTA_CFG_IOCM_PORT, 8943);
```

### 6.2.2.4 设备绑定

#### 接口功能

设备第一次接入物联网平台时需要进行绑定操作，上层应用通过调用该接口传入设备序列号或者MAC地址以及设备信息来绑定到物联网平台。

在绑定前需要调用[设置业务参数](#)接口设置绑定服务器IP与端口（IoCM服务器地址与端口，Agent Lite会配置默认端口8943），对应参数配置为  
**EN\_IOTA\_CFG\_IOCM\_ADDR**和**EN\_IOTA\_CFG\_IOCM\_PORT**。



#### 说明

设备绑定是指设备第一次接入IoT平台的过程，需要开发者先在IoT平台注册直连设备，之后在设备上发起绑定操作，将设备绑定到IoT平台上。如果未在IoT平台注册该设备，则绑定操作会失败，Agent Lite将会等待一段时间继续尝试。

## 接口描述

```
HW_INT IOTA_Bind(const HW_CHAR *pcVerifyCode, const ST_IOTA_DEVICE_INFO *pstInfo);
```

## 参数说明

字段	必选/可选	类型	描述
pcVerifyCode	必选	String	设备绑定验证码，pcVerifyCode必须带结束符 ‘\0’。
pstInfo	必选	ST_IOTA_DEVICE_INFO 结构体	设备信息，pstInfo结构体中的成员必须带结束符 ‘\0’。

## 接口返回值

参见[2. 函数标准返回值](#)



### 说明

- 此返回值是调用接口的同步返回结果，返回0只是说明接口调用成功，并不说明绑定成功，绑定成功需要收到IOTA\_TOPIC\_BIND\_RSP广播。
- 当前绑定流程的重试策略为：如果绑定失败，则30秒后继续进行重试，如果重试超过5次（总计尝试超过6次），则返回失败，不再进行重试。如果想要重新发起绑定，建议让用户重启设备。

## 返回结果

广播名称	广播参数	成员	描述
IOTA_TOPIC_BIN_D_RSP	HW_MSG对象	EN_IOTA_BIND_I_E_TYPE	绑定返回结果（详见枚举EN_IOTA_BIND_I_E_TYPE）。

## 示例

调用IOTA\_Bind()接口进行设备绑定。

```
//开发者调用该接口进行设备绑定
ST_HW_DEVICE_INFO stDeviceInfo
stDeviceInfo.pcNodeId = "SN Number";
stDeviceInfo.pcManufacturerId = "Huawei";
stDeviceInfo.pcDeviceType = "Gateway";
stDeviceInfo.pcModel = "HW_GW101";
stDeviceInfo.pcProtocolType = "HuaweiM2M";

IOTA_Bind("SN Number", &stDeviceInfo);
```

当设备成功绑定之后，Agent Lite会返回给UI如下几个参数，需要UI进行持久化存储，设备登录前需要提前进行配置。

```
//注册广播接收处理函数
HW_BroadCastReg( "IOTA_TOPIC_BIND_RSP" , Device_RegResultHandler);
// 开发者注册该函数处理绑定结果
HW_INT Device_RegResultHandler(HW_UINT uiCookie, HW_MSG pstMsg)
{
    HW_CHAR *pcDeviceId;
    HW_CHAR *pcDeviceSecret;
    HW_CHAR *pcAppId;
    HW_CHAR *pcIoCMServerAddr;
    HW_UINT uiIoCMServerPort;
    HW_CHAR *pcMqttServerAddr;
    HW_UINT uiMqttServerPort;

    If (HW_SUCCESS != HW_MsgGetUint(pstMsg, EN_IOTA_BIND_IE_RESULT, 0))
    {
        Return 0;
    }

    pcDeviceId = HW_MsgGetStr(pstMsg, EN_IOTA_BIND_IE_DEVICEID);
    pcDeviceSecret = HW_MsgGetStr(pstMsg, EN_IOTA_BIND_IE_DEVICESECRET);
    pcAppId = HW_MsgGetStr(pstMsg, EN_IOTA_BIND_IE_APPID);
    pcIoCMServerAddr = HW_MsgGetStr(pstMsg, EN_IOTA_BIND_IE_IOCMAUTH );
    uiIoCMServerPort = HW_MsgGetUint(pstMsg, EN_IOTA_BIND_IE_IOCMPORT, 0);
    pcMqttServerAddr = HW_MsgGetStr(pstMsg, EN_IOTA_BIND_IE_MQTTADDR );
    uiMqttServerPort = HW_MsgGetUint(pstMsg, EN_IOTA_BIND_IE_MQTPORT, 0);

    Config_save( "DeviceId" ,pcDeviceId);
    Config_save( "DeviceSecret" ,pcDeviceSecret);
    Config_save( "AppId" ,pcAppId);
    Config_save( "IoCMAddr" ,pcIoCMServerAddr);
    Config_save( "IoCPort" ,pcIoCMServerPort);
    Config_save( "MqttAddr" ,pcMqttServerAddr);
    Config_save( "MqttPort" ,pcMqttServerPort);

    return 0;
}
```

### 6.2.2.5 设备解绑定命令接收

#### 接口功能

注册设备解绑定接收广播来接收处理平台下发的直连设备解绑定命令，开发者收到该广播后需要删除直连设备的配置信息并且释放所有资源，下一次重启后需要重新进行绑定。

#### 接口描述

```
IOTA_TOPIC_CMD_UNBIND_RECEIVE;
```

#### 参数说明

无。

#### 返回结果

无。

#### 示例

```
// 开发者注册该函数进行解绑定直连设备的处理
HW_INT Gateway_UnbindRecvHandler(HW_UINT uiCookie, HW_MSG pstMsg)
{
    // Delete Config file, free resources.
```

```
    return 0;
}
//初始化时进行注册该函数
HW_BroadCastReg( "IOTA_TOPIC_CMD_UNBIND_RECEIVE" , Gateway_UnbindRecvHandler);
```

## 6.2.2.6 参数配置

### 6.2.2.6.1 设置业务参数

#### 接口功能

在登录前配置登录所需要的参数。

#### 接口描述

```
HW_INT IOTA_ConfigSetStr(HW_INT iItem, HW_CHAR *pValue)
HW_INT IOTA_ConfigSetUint(HW_INT iItem, HW_UINT uiValue)
```

#### 参数说明

字段	必选/ 可选	类型	描述
key	必选	int	<p>设备登录的配置项。</p> <ul style="list-style-type: none"><li>● 设备ID: EN_IOTA_CFG_DEVICEID。</li><li>● AppId: EN_IOTA_CFG_APPID。</li><li>● 密码: EN_IOTA_CFG_DEVICESECRET。</li><li>● HTTP地址: LEN_IOTA_CFG_IOCM_ADDR。</li><li>● HTTP端口: EN_IOTA_CFG_IOCM_PORT。</li><li>● MQTT地址: EN_IOTA_CFG_MQTT_ADDR。</li><li>● MQTT端口: EN_IOTA_CFG_MQTT_PORT。</li></ul>
value	必选	String	<p>设置的值。</p> <ul style="list-style-type: none"><li>● 设备ID: 从绑定成功的回调中获取。</li><li>● AppId: 从绑定成功的回调中获取。</li><li>● 密码: 从绑定成功的回调中获取。</li><li>● HTTP地址: Agent Lite对接平台南向接入地址。</li><li>● HTTP端口: 8943。</li><li>● MQTT地址: Agent Lite对接平台南向接入地址。</li><li>● MQTT端口: 8883。</li></ul>

#### 接口返回值

参见[2. 函数标准返回值](#)

## 示例

```
// 开发者调用该接口设置参数  
IOTA_ConfigSetStr (EN_IOTA_CONFIG_IOCM_ADDR, "10.0.0.1");  
IOTA_ConfigSetUint (EN_IOTA_CFG_IOCM_PORT, 8943);
```

### 6.2.2.6.2 设置加密算法类型参数（可选）

#### 接口功能

在登录前配置敏感信息加密算法类型，业务可以根据安全级别选择合适的加密算法类型。

#### 接口描述

```
HW_UINT HW_SetAlgType(HW_UINT uiAlgType);
```

#### 参数说明

字段	必选/可选	类型	描述
uiAlgType	必选	Int	加密算法类型。 <ul style="list-style-type: none"><li>● 0: AES 256 CBC。</li><li>● 1: AES 128 GCM。</li><li>● 2: AES 256 GCM。</li></ul>

#### 接口返回值

参见[2. 函数标准返回值](#)

#### 示例

```
// 开发者调用该接口设置参数  
HW_SetAlgType(2);
```

### 6.2.2.7 设备登录

#### 接口功能

设备在第一次绑定后，或者在设备重启后需要进行登录的流程。

#### 接口描述

```
HW_INT IOTA_Login();
```

#### 参数说明

无

## 接口返回值

参见[2. 函数标准返回值](#)



### 说明

此返回值是调用接口的同步返回结果，返回0只是说明接口调用成功，并不说明登录成功，登录成功需要收到**IOTA\_TOPIC\_CONNECTED\_NTY**广播。登录前通过参数配置接口(**IOTA\_SetConfig**)传入所需的登录信息。

## 返回结果

广播名称	广播参数	成员	描述
IOTA_TOPIC_CONNECTED_NTY	HW_MSG对象	无	登录成功或重连成功。
IOTA_TOPIC_DISCONNECT_NTY	HW_MSG对象	EN_ULGN_IE_ER_R_REASON	登录失败或连接断开。

## 示例

```
Config_Get( "DeviceId" , pcDeviceId);
Config_Get( "DeviceSecret" , pcDeviceSecret);
Config_Get( "AppId" , pcAppId);
Config_Get( "HAAAddr" , pcHAServerAddr);
Config_Get( "LVSAddr" , pcLVSServerAddr);

IOTA_SetConfig(EN_IOTA_CFG_DEVICEID, pcDeviceId);
IOTA_SetConfig(EN_IOTA_CFG_DEVICESECRET, pcDeviceSecret);
IOTA_SetConfig(EN_IOTA_CFG_APPID, pcAppId);
IOTA_SetConfig(EN_IOTA_CFG_HA_ADDR, pcHAServerAddr);
IOTA_SetConfig(EN_IOTA_CFG_LVS_ADDR, pcLVSServerAddr);

IOTA_Login();
```

然后等待Agent Lite的连接状态广播。

需要提前实现连接状态通知广播接收处理函数，建议：

- 对于网关设备，在连接成功的处理函数中需要进行非直连设备状态上报的处理，并且将缓存的所有上报数据进行上报。
- 在连接断开的处理函数中记录设备断开状态，之后如果有数据上报，需要进行缓存，等到连接成功后再进行上报。

```
// 开发者注册该函数进行连接成功后的处理
HW_INT Device_ConnectedHandler(HW_UINT uiCookie, HW_MSG pstMsg)
{
//update device states
//send buffer data

return 0;
}

// 开发者注册该函数进行连接失败后的处理
HW_INT Device_DisconnectHandler(HW_UINT uiCookie, HW_MSG pstMsg)
{
//stop reporting data
return 0;
}

//绑定广播接收处理函数
HW_BroadCastReg( "IOTA_TOPIC_CONNECTED_NTY" , Device_ConnectedHandler);
HW_BroadCastReg( "IOTA_TOPIC_DISCONNECT_NTY" , Device_DisconnectHandler);
```

设备登录后，表示该设备已经成功的连接到物联网平台。

连接成功后，如果因为网络或服务器原因导致连接断开，Agent Lite会自动尝试重新连接，并将实时状态通过这两个广播上报给第三方应用。

### 6.2.2.8 设备登出

#### 接口功能

断开与IoT平台的连接。

#### 接口描述

```
HW_INT IOTA_Logout();
```

#### 参数说明

无

#### 接口返回值

参见[2. 函数标准返回值](#)



说明

此返回值是调用接口的同步返回结果，返回0只是说明接口调用成功。

#### 示例

```
// 开发者调用设备登出接口:  
IOTA_Logout();
```

### 6.2.3 网关管理非直连设备

当开发设备为网关设备时，设备需要管理所有非直连设备（传感器设备）的接入与删除，并且记录这些设备ID与对应设备的映射关系。

#### 6.2.3.1 设备添加

#### 接口功能

当有新设备接入网关后，通过调用设备添加接口将非直连设备接入物联网平台，并且获得平台分配的唯一设备逻辑ID。

#### 接口描述

```
HW_INT IOTA_HubDeviceAdd(HW_UINT uiCookie, const ST_IOTA_DEVICE_INFO *pstDeviceInfo);
```

#### 参数说明

字段	必选/可选	类型	描述
uiCookie	可选	HW_UINT	Cookie有效值为1-65535。

字段	必选/可选	类型	描述
pstDeviceInfo	必选	<b>ST_IOTA_DEVICE_INFO</b> 结构体	设备信息， pstDeviceInfo结构 体中的成员变量必 须带结束符 '\0'。

## 接口返回值

参见[2. 函数标准返回值](#)

## 返回结果

广播名称	广播参数	成员	描述
IOTA_TOPIC_HUB_ADDDEV_RSP	HW_MSG对象	<b>EN_IOTA_HUB_ID_TYPE</b>	返回设备添加结 果，如果添加成功 则返回设备ID。

## 示例

```
// 开发者调用该接口进行设备添加
ST_IOTA_DEVICE_INFO stDeviceInfo
stDeviceInfo.pcNodeId = "SN Number";
stDeviceInfo.pcManufacturerId = "Huawei";
stDeviceInfo.pcDeviceType = "Camera";
stDeviceInfo.pcModel = "HW_CAM101";
stDeviceInfo.pcProtocolType = "ONVIF";

IOTA_HubDeviceAdd(29011, &stDeviceInfo)
```

### 结果处理：

```
// 开发者注册该函数进行设备添加后的处理
HW_INT Device_AddResultHandler(HW_UINT uiCookie, HW_MSG pstMsg)
{
    uiResult = HW_MsgGetUInt(pstMsg, EN_IOTA_HUB_IE_RESULT);
    if (EN_IOTA_HUB_RESULT_SUCCESS != uiResult)
    {
        // retry with uiCookie
        return 0;
    }

    return 0;
}
//绑定广播接收处理函数
HW_BroadCastReg("IOTA_TOPIC_HUB_ADDDEV_RSP", Device_AddResultHandler);
```

## 6.2.3.2 设备状态更新

### 接口功能

通过该接口更新设备的状态信息，包括直连设备与所管理的非直连设备。设备离线、上线均可通过该接口刷新设备状态信息。

直连设备状态通过设备的登录状态进行管理，当直连设备连接断开则表示设备离线，当直连设备连接或重连成功，则表示设备上线，无需通过该接口进行刷新。故建议开发者使用该接口刷新非直连设备的状态。

## 接口描述

```
HW_INT IOTA_DeviceStatusUpdate(HW_UINT uiCookie, const HW_CHAR *pcDeviceId, const HW_CHAR  
*pcStatus, const HW_CHAR *pcStatusDetail);
```

## 参数说明

字段	必选/可选	类型	描述
uiCookie	可选	HW_UINT	Cookie有效值为1-65535。
pcDeviceId	必选	HW_CHAR	设备Id，该参数必须带结束符‘\0’。
pcStatus	必选	HW_CHAR	设备状态，该参数必须带结束符‘\0’。 <ul style="list-style-type: none"><li>● 在线: ONLINE。</li><li>● 离线: OFFLINE。</li></ul>

字段	必选/可选	类型	描述
pcStatusDetail	必选	HW_CHAR	<p>设备状态详细信息，该参数必须带结束符 ‘\0’。</p> <ul style="list-style-type: none"><li>● 无: NONE。</li><li>● 配置等待: CONFIGURATION_PENDING。</li><li>● 通信错误: COMMUNICATION_ERROR。</li><li>● 配置错误: CONFIGURATION_ERROR。</li><li>● 桥接器离线: BRIDGE_OFFLINE。</li><li>● 固件升级: FIRMWARE_UPDATING。</li><li>● 循环任务: DUTY_CYCLE。</li><li>● 未激活: NOT_ACTIVE。</li></ul>

## 接口返回值

参见[2. 函数标准返回值](#)

## 返回结果

广播名称	广播参数	成员	描述
IOTA_TOPIC_DEV_UPDATE_RSP/ {deviceId}	HW_MSG对象	无	设备状态更新结果。

## 示例

```
HW_CHAR *pcDeviceId = stDevice.pcDeviceId;  
IOTA_DeviceStatusUpdate(0, pcDeviceId, "ONLINE", "NONE");
```

然后等待命令执行结果。

```
// 开发者注册该函数进行状态更新后的处理
HW_INT Device_StatusUpdateHandler(HW_UINT uiCookie, HW_MSG pstMsg)
{
    HW_CHAR pcCmdContent;
    pcCmdContent = HW_MsgGetStr(pstMsg, EN_IOTA_DEVUPDATE_IE_RESULT);
    pcCmdContent = HW_MsgGetStr(pstMsg, EN_IOTA_DEVUPDATE_IE_DEVICEID);

    return 0;
}
//绑定广播接收处理函数
HW_BroadCastReg("IOTA_TOPIC_DEVUPDATE_RSP", Device_StatusUpdateHandler);
```

### 6.2.3.3 设备删除

#### 接口功能

当有新设备需要从网关移除时，通过调用设备删除接口将非直连设备从物联网平台删除。

#### 接口描述

```
HW_INT IOTA_HubDeviceRemove(HW_UINT uiCookie, const HW_CHAR *pcDeviceId);
```

#### 参数说明

字段	必选/可选	类型	描述
uiCookie	可选	HW_UINT	Cookie有效值1-65535。
pcDeviceId	必选	String	设备Id，该参数必须带结束符 '\0'。

#### 接口返回值

参见[2. 函数标准返回值](#)

#### 返回结果

广播名称	广播参数	成员	描述
IOTA_TOPIC_HUB_RMVDEV_RSP	HW_MSG对象	<a href="#">EN_IOTA_HUB_I E_TYPE</a>	删除结果。

#### 示例

```
// 开发者调用该接口进行设备删除
HW_CHAR *pcDeviceId = stDevice.pcDeviceId;

IOTA_HubDeviceRemove(HW_NULL, pcDeviceId);
结果处理:
HW_INT Device_RemoveResultHandler(HW_UINT uiCookie, HW_MSG pstMsg)
{
    uiResult = HW_MsgGetUInt(pstMsg, EN_IOTA_HUB_IE_RESULT);
    if (EN_IOTA_HUB_RESULT_SUCCESS != uiResult)
```

```
{  
    // retry with uiCookie  
    return 0;  
}  
  
return 0;  
}  
  
HW_BroadCastReg( "IOTA_TOPIC_HUB_RMVDEV_RSP" , Device_RemovResultHandler);
```

## 6.2.4 设备服务数据上报

### 接口功能

当直连设备（网关）有数据需要上报或者非直连设备上报数据到网关时，网关需要调用设备服务数据上报接口将数据上报到物联网平台。

### 接口描述

```
HW_INT IOTA_ServiceDataReport(HW_UINT uiCookie, const HW_CHAR *pcRequestId,  
const HW_CHAR *pcDeviceId, const HW_CHAR *pcServiceId, const HW_CHAR *pcServiceProperties);
```

### 参数说明

字段	必选/可选	类型	描述
uiCookie	可选	unsigned int	Cookie有效值1-65535。
pcRequestId	必选	String	请求ID，用来匹配之前平台下发的服务命令。当该次数据上报为此前某一次命令请求的响应时，需要填写此次命令请求的请求ID，该参数必须带结束符‘\0’。
pcDeviceId	必选	String	设备ID，该参数必须带结束符‘\0’。
pcServiceId	必选	String	服务ID，该参数必须带结束符‘\0’。
pcServiceProperties	必选	String	服务属性，该参数必须带结束符‘\0’。

### 接口返回值

参见[2. 函数标准返回值](#)

### 返回结果

广播名称	广播参数	成员	描述
IOTA_TOPIC_DAT ATRANS_REPORT _RSP/{deviceId}	HW_MSG对象	EN_IOTA_DATATTR ANS_IE_RESULT	数据上报结果。 ● 成功：0。 ● 失败：1。

## 示例

用户根据Profile格式使用Json组件拼装服务属性的内容(pcServiceProperties)。

```
HW_UINT *uiLen;  
  
IOTA_ServiceDataReport(1211, NULL, "xxxx_xxxx_xxxx_xxxx", "DoorWindow", "{\"status\": \"OPEN\"}");
```

数据上报结果接收。

```
//开发者注册该函数进行设备服务数据上报后的处理  
HW_INT Device_DataReportResultHandler(HW_UINT uiCookie, HW_MSG pstMsg)  
{  
    uiResult = HW_MsgGetUInt(pstMsg, EN_IOTA_DATATRANS_IE_RESULT);  
    if (HW_SUCCESS != uiResult)  
    {  
        // retry with uiCookie  
        return 0;  
    }  
  
    return 0;  
}  
//在设备添加成功后立即注册服务数据上报结果接收广播  
HW_BroadCastReg("IOTA_TOPIC_DATATRANS_REPORT_RSP/XXXX_XXXX_XXXX_XXXX", Device_AddResultHandler);
```

## 6.2.5 设备命令接收

### 接口功能

注册设备命令接收广播来接收处理平台下发的控制命令。

### 接口描述

```
IOTA_TOPIC_SERVICE_COMMAND_RECEIVE/{deviceId};
```

### 参数说明

请参考[EN\\_IOTA\\_DATATRANS\\_IE\\_TYPE消息信元枚举说明](#)

### 返回结果

无

## 示例

```
// 开发者注册该函数进行命令接收的处理  
HW_INT Switch_CommandRecvHandler(HW_UINT uiCookie, HW_MSG pstMsg)  
{  
    HW_CHAR *pcMethod, *pcServiceId, *pcCmdContent, *pcDeviceId;  
  
    pcDeviceId = HW_MsgGetStr(pstMsg, EN_IOTA_DATATRANS_IE_DEVICEID);  
    pcServiceId = HW_MsgGetStr(pstMsg, EN_IOTA_DATATRANS_IE_SERVICEID);  
    pcMethod = HW_MsgGetStr(pstMsg, EN_IOTA_DATATRANS_IE_METHOD);  
    pcCmdContent = HW_MsgGetStr(pstMsg, EN_IOTA_DATATRANS_IE_CMDCONTENT);  
  
    if (strcmp(pcServiceId, "switch"))  
    {  
        //根据Proflie定义的命令参数，使用Json组件解析pcCmdContent  
        //Send command to Switch
```

```
}

return 0;
}

//在设备添加成功后立即注册设备命令接收广播
HW_BroadCastReg( "IOTA_TOPIC_SERVICE_CMD_RECEIVE/XXXX_XXXX_XXXX_XXXX" ,
Device_AddResultHandler);
```

开发者需要在设备添加成功后注册该设备的命令接收广播，广播主题为“`IOTA_TOPIC_SERVICE_CMD_RECEIVE/设备ID`”，Agent Lite收到平台发往给设备的命令后会直接广播给该设备注册的广播处理函数。如果开发者不需要按设备进行分发，直接使用主题名即可，即“`IOTA_TOPIC_SERVICE_CMD_RECEIVE`”。

## 6.2.6 数据发布

### 接口功能

当直连设备（网关）有数据需要发布或者非直连设备发布数据到网关时，网关需要调用设备服务数据发布接口将数据发布到物联网平台。其与数据上报功能上的区别在于，发布的数据可以自己规定Topic，同时包体需要自己组装给入参，内部只是透传，不对json体进行拼装。

### 接口描述

```
HW_INT IOTA_MqttDataPub (HW_UINT uiCookie, const HW_UCHAR *pucTopic, HW_UINT uiQos, const HW_BYTES
*pbstrServiceData);
```

### 参数说明

字段	必选/可选	类型	描述
uiCookie	可选	unsign int	Cookie有效值1-65535。
pucTopic	必选	String	发布数据的Topic，该参数必须带结束符‘\0’。
uiQos	必选	unsign int	Mqtt协议的相关参数，一般为“1”。
pbstrServiceData	必选	HW_BYTES	发布数据的包体。

### 接口返回值

参见[2. 函数标准返回值](#)

## 返回结果

广播名称	广播参数	成员	描述
IOTA_TOPIC_MQTT_DATA_PUB_RSP	HW_MSG对象	EN_IOTA_DATATTR ANS_IE_RESULT	数据发布结果。 ● 成功：0。 ● 失败：1。

## 示例

```
HW_BYTES bstrBody;
.....
IOTA_MqttDataPub(1211, "/huawei/v1/devices/336d9bac-9ebf-44e9-95cf-efac5f05da3a/services/Storage", 1, bstrBody);
```

数据上报结果接收。

```
//开发者注册该函数进行设备服务数据上报后的处理
HW_INT Device_DataPubResultHandler(HW_UINT uiCookie, HW_MSG pstMsg)
{
    uiResult = HW_MsgGetUInt(pstMsg, EN_IOTA_DATATRANS_IE_RESULT);
    if (HW_SUCCESS != uiResult)
    {
        // retry with uiCookie
        return 0;
    }

    return 0;
}
//在设备添加成功后立即注册服务数据上报结果接收广播
HW_BroadCastReg("IOTA_TOPIC_MQTT_DATA_PUB_RSP", Device_AddResultHandler);
```

## 6.2.7 Json 组件使用说明

该组件为Agent Lite提供给开发者的工具组件，如果开发者无法进行Json格式的编码和解码，则可以使用该组件进行编码和解码。主要用于上报数据组装与下发命令解析。

### 1. Json 编码

使用Json组件进行编码的流程。

创建Json编码对象。

```
HW_JSONOBJ HW_JsonObjCreate()
```

获取Json对象根节点。

```
HW_JSON HW_JsonGetJson(HW_JSONOBJ hjson)
```

往Json对象中添加键值对：

添加pcVal为字符串的Json键值对。

```
HW_INT HW_JsonAddStr(HW_JSON pstJson, HW_CHAR *pcKey, HW_CHAR *pcVal)
```

添加uiVal为整数的Json键值对。

```
HW_INT HW_JsonAddUint(HW_JSON pstJson, HW_CHAR *pcKey, HW_UINT uiVal)
```

添加bVal为bool的Json键值对。

```
HW_INT HW_JsonAddBool(HW_JSON pstJson, HW_CHAR *pcKey, HW_BOOL bVal)
```

添加值为Json的Json键值对，获取到的为子Json对象。

```
HW_JSON HW_JsonAddJson(HW_JSON pstJson, HW_CHAR *pcKey)
```

添加值为Json数组Json键值对，获取到的为子Json数组对象。

```
HW_JSON_ARRAY HW_JsonAddArray(HW_JSON pstJson, HW_CHAR *pcKey)
```

往Json数组中添加键值对：

添加pcVal为字符串的Json键值对。

```
HW_INT HW_JsonArrayAddStr(HW_JSON_ARRAY *pstArray, HW_CHAR *pcKey, HW_CHAR *pcVal)
```

添加uiVal为整数的Json键值对。

```
HW_INT HW_JsonArrayAddUint(HW_JSON_ARRAY *pstArray, HW_CHAR *pcKey, HW_UINT uiVal)
```

添加bVal为bool的Json键值对。

```
HW_INT HW_JsonArrayAddBool(HW_JSON_ARRAY *pstArray, HW_CHAR *pcKey, HW_BOOL bVal)
```

添加pucValue为Json的Json值，获取到的为子Json对象。

```
HW_JSON HW_JsonArrayAddJson(HW_JSON_ARRAY *pstArray)
```

添加pucValue为Json数组Json键值对，获取到的为子Json数组对象。

```
HW_JSON_ARRAY *HW_JsonArrayAddArray(HW_JSON_ARRAY *pstArray) *HW_JsonArrayAddArray(HW_JSON_ARRAY *pstArray)
```

获取Json字符串。

```
HW_CHAR *HW_JsonEncodeStr(HW_JSONOBJ hJson);
```

删除Json对象。

```
HW_VOID HW_JsonObjDelete(HW_JSONOBJ *phJson);
```

### Json编码示例：

待解析Json格式：

```
{
    "temperature":22
    "otherInfo":
    {
        "batteryLevel" :" low"
    }
}
/*变量定义*/
HW_JSONOBJ jsonObj;
HW_JSON rootJson;
HW_JSON json;
HW_CHAR *pcJsonStr;

/*创建Json编码对象*/
hJsonObj = HW_JsonObjCreate();

/*获取跟节点Json对象*/
rootJson = HW_JsonGetJson(hJsonObj);

/*往根节点中添加键值对*/
HW_JsonAddUint(rootJson, "temperature", 22);

/*从根节点中获取子Json对象*/
json = HW_JsonAddJson(rootJson, "otherInfo");

/*在子Json中添加键值对*/
HW_JsonAddStr(json, "batteryLevel", "low");
```

```
/*获取Json字符串*/  
pcJsonStr = HW_JsonEncodeStr(hjsonObj);  
  
/*删除之前创建的Json编码对象，释放资源*/  
HW_JsonObjDelete(&hJsonObj);
```

## 2. Json 解码

使用Json组件进行解码的流程。

创建Json解析对象。

```
HW_JSONOBJ HW_JsonDecodeCreate(HW_CHAR *pucStr, HW_BOOL bStrCpy)
```

获取Json解析对象中的Json数据部分。

```
HW_JSON HW_JsonGetJson(HW_JSONOBJ hJson)
```

获取Json数据中与pucKey对应的字符串。

```
HW_CHAR *HW_JsonGetStr(HW_JSON pstJson, HW_CHAR *pucKey)
```

获取Json数据中与pucKey对应的无符号整型。

```
HW_UINT HW_JsonGetUint(HW_JSON pstJson, HW_CHAR *pucKey, HW_UINT uiDft)
```

获取Json数据中与pucKey对应的Boolean值。

```
HW_BOOL HW_JsonGetBool(HW_JSON pstJson, HW_CHAR *pucKey, HW_BOOL bDft)
```

获取Json数据中与pucKey对应的数组。

```
HW_UJSON_ARRAY HW_JsonGetArray(HW_JSON pstJson, HW_CHAR *pucKey)
```

获取Json数组的长度。

```
HW_UINT HW_JsonArrayGetCount(HW_UJSON_ARRAY pstArray)
```

获取Json数组中序号为uiIndex项的Json数据。

```
HW_JSON HW_JsonArrayGetJson(HW_UJSON_ARRAY pstArray, HW_UINT uiIndex)
```

获取Json数组中序号为uiIndex项的无符号整形。

```
HW_UINT HW_JsonArrayGetUint(HW_UJSON_ARRAY pstArray, HW_UINT uiIndex, HW_UINT uiDft)
```

获取Json数组中序号为uiIndex项的Boolean值。

```
HW_UINT HW_JsonArrayGetBool(HW_UJSON_ARRAY pstArray, HW_UINT uiIndex, HW_BOOL bDft)
```

获取Json数组中序号为uiIndex项的字符串。

```
HW_CHAR *HW_JsonArrayGetStr(HW_UJSON_ARRAY pstArray, HW_UINT uiIndex)
```

获取Json数组中序号为uiIndex项的子数组。

```
HW_UJSON_ARRAY HW_JsonArrayGetArray(HW_UJSON_ARRAY pstArray, HW_UINT uiIndex)
```

删除之前创建的Json解析对象。

```
HW_VOID HW_JsonObjDelete(HW_JSONOBJ *phJson)
```

### Json解析示例：

待解析Json格式：

```
{  
    "action": "notify",
```

```
"type": "userstate",
"userstateinfo":
[
{ "num": "11111 ", "state": "idle"},  

{ "num": "11111", "state": "ringing"}  

]  

}
/*变量定义*/
HW_JSONOBJ jsonObj;
HW_JSON json;
HW_UJSON_ARRAY jsonArray;
HW_CHAR *action;
HW_CHAR *type;
HW_UINT count;
HW_UINT index;

/*创建Json解析对象*/
jsonObj = HW_JsonDecodeCreate(jsonStr, HW_TRUE);

/*获取Json解析对象中的Json数据部分*/
json = HW_JsonGetJson(jsonObj);

/*获取Json数据中与"action"对应的字符串*/
action = HW_JsonGetStr(json, "action");

/*获取Json数据中与"type"对应的字符串*/
type = HW_JsonGetStr(json, "type");

/*获取Json数据中与"userstateinfo"对应的Json数组*/
jsonArray = HW_JsonGetArray(json, "userstateinfo");

/*获取数组 jsonArray 的长度*/
count = HW_ArrayGetCount(jsonArray);
for (index = 0; index < count; index++)
{
/*获取数组 jsonArray 中序号为index项的Json数据*/
HW_JSON jsonItem = HW_ArrayGetJson(jsonArray, index);

/*获取jsonItem中与" num "对应的字符串*/
HW_CHAR *num = HW_JsonGetStr(jsonItem, "num");

/*获取jsonItem中与" state "对应的字符串*/
HW_CHAR *state = HW_JsonGetStr(jsonItem, "state");
.....
}

/*删除之前创建的Json解析对象，释放资源*/
HW_JsonDelete(jsonObj);
```

## 6.3 常用数据结构定义

### 6.3.1 ST\_IOTA\_DEVICE\_INFO 结构体说明

字段	必选/可选	类型	描述
pcNodeId	必选	String	关键参数，对接平台的网关下设备唯一标识，设备填写，平台用于判重。
pcName	可选	String	设备名称。
pcDescription	可选	String	设备描述。
pcManufacturerId	必选	String	厂商ID。

字段	必选/可选	类型	描述
pcManufacturerName	可选	String	厂商名。
pcMac	可选	String	设备MAC地址。
pcLocation	可选	String	设备的位置。
pcDeviceType	必选	String	设备类型。 <b>说明</b> 如果接入设备是网关设备，则“pcDeviceType”填写为“Gateway”，如果为传感器设备直连，则填写为对应的设备类型。
pcModel	必选	String	型号。 <ul style="list-style-type: none"><li>● 直连设备：与Profile中定义的model保持一致即可。</li><li>● Z-Wave设备： ProductType + ProductId，格式（使用16进制）： XXXX-XXXX 补0对齐，如001A-0A12，其他协议再定。</li></ul>
pcSwVersion	可选	String	软件版本。 Z-Wave：主版本号.次版本号，如：1.1
pcFwVersion	可选	String	固件版本。
pcHwVersion	可选	String	硬件版本。
pcProtocolType	必选	String	协议类型：Z-Wave。
pcBridgeId	可选	String	表示设备通过哪个Bridge接入平台。
pcStatus	可选	String	表示设备是否在线。 <ul style="list-style-type: none"><li>● ONLINE 在线。</li><li>● OFFLINE 离线。</li></ul>

字段	必选/可选	类型	描述
statusDetail	可选	String	<p>状态详情，如果pcStatus不为空，则该参数必选。</p> <p>参数值：</p> <ul style="list-style-type: none"><li>● 无: NONE。</li><li>● 配置等待: CONFIGURATION_PENDING。</li><li>● 通信错误: COMMUNICATION_ERROR。</li><li>● 配置错误: CONFIGURATION_ERROR。</li><li>● 桥接器离线: BRIDGE_OFFLINE。</li><li>● 固件升级: FIRMWARE_UPDATING。</li><li>● 循环任务: DUTY_CYCLE。</li><li>● 未激活: NOT_ACTIVE。</li></ul>
pcMute	可选	String	<p>表示设备是否被屏蔽。</p> <ul style="list-style-type: none"><li>● TRUE</li><li>● FALSE</li></ul>

### 6.3.2 EN\_IOTA\_BIND\_IE\_TYPE 消息信元枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_BIND_I_E_RESULT	0	<a href="#">EN_IOTA_BIND_RESULT_TYPE</a>	绑定结果。
EN_IOTA_BIND_I_E_DEVICEID	1	String	平台分配的逻辑设备ID。
EN_IOTA_BIND_I_E_DEVICESCRET	2	String	设备接入的鉴权秘钥。
EN_IOTA_BIND_I_E_APPID	3	String	开发者应用ID。
EN_IOTA_BIND_I_E_IOCM_ADDR	4	String	IoCM服务器地址。
EN_IOTA_BIND_I_E_IOCM_PORT	5	unsigned int	IoCM服务器端口。
EN_IOTA_BIND_I_E_MQTT_ADDR	6	String	MQTT服务器地址。

枚举项	枚举值	类型	描述
EN_IOTA_BIND_I E_MQTT_PORT	7	unsigned int	MQTT服务器端口。

### 6.3.3 EN\_IOTA\_BIND\_RESULT\_TYPE 参数枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_BIND_R ESULT_SUCCESS	0	NA	绑定成功。
EN_IOTA_BIND_R ESULT_DEV_NOT _BIND	1	NA	未扫码。
EN_IOTA_BIND_R ESULT_VERIFYC ODE_EXPIRED	2	NA	验证码过期。
EN_IOTA_BIND_R ESULT_FAILED	255	NA	其余失败。

### 6.3.4 EN\_IOTA\_LGN\_IE\_TYPE 消息信元枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_LGNIE _REASON	0	EN_IOTA_LGN_R EASON_TYPE	登录失败原因。

### 6.3.5 EN\_IOTA\_LGN\_REASON\_TYPE 参数枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_LGN_R EASON_NULL	0	NA	无原因。
EN_IOTA_LGN_R EASON_CONNCE T_ERR	1	NA	连接失败。
EN_IOTA_LGN_R EASON_SERVER_ BUSY	2	NA	服务器忙。

枚举项	枚举值	类型	描述
EN_IOTA_LGN_R_EASON_AUTH_FAILED	3	NA	鉴权失败、开发者需要停止重新尝试登录。
EN_IOTA_LGN_R_EASON_NET_UNAVAILABLE	5	NA	网络不可用。
EN_IOTA_LGN_R_EASON_DEVICE_NOEXIST	12	NA	设备不存在、开发者需要停止重新尝试登录。
EN_IOTA_LGN_R_EASON_DEVICE_RMVED	13	NA	设备已删除、开发者需要停止重新尝试登录。
EN_IOTA_LGN_R_EASON_UNKNOW_N	255	NA	未知原因。

### 6.3.6 EN\_IOTA\_CFG\_TYPE 参数枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_CFG_DEVICEID	0	String	平台分配的逻辑设备ID。
EN_IOTA_CFG_DEVICESECRET	1	String	设备接入的鉴权秘钥。
EN_IOTA_CFG_APPID	2	String	开发者应用ID。
EN_IOTA_CFG_IOCM_ADDR	3	String	IoCM服务器地址。
EN_IOTA_CFG_IOCM_PORT	4	unsigned int	IoCM服务器端口。
EN_IOTA_CFG_MQTT_ADDR	5	String	MQTT服务器地址。
EN_IOTA_CFG_MQTT_PORT	6	unsigned int	MQTT服务器端口。

### 6.3.7 EN\_IOTA\_HUB\_IE\_TYPE 消息信元枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_HUB_IE_RESULT	0	EN_IOTA_HUB_R ESULT_TYPE	添加/删除执行结果。
EN_IOTA_HUB_IE_DEVICEID	1	String	添加成功后分配的设备ID。

### 6.3.8 EN\_IOTA\_HUB\_RESULT\_TYPE 参数枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_HUB_R ESULT_SUCCESS	0	NA	添加/删除执行成功。
EN_IOTA_HUB_R ESULT_DEVICE_E XIST	1	NA	设备已存在。
EN_IOTA_HUB_R ESULT_DEVICE_N OTEXIST	2	NA	设备不存在。
EN_IOTA_HUB_R ESULT_DEVICE_F AILED	255	NA	执行失败。

### 6.3.9 EN\_IOTA\_DATATRANS\_IE\_TYPE 消息信元枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_DATATTR ANS_IE_RESULT	0	unsigned int	命令执行返回结果。 <ul style="list-style-type: none"><li>● 成功: 0。</li><li>● 失败: 1。</li></ul>
EN_IOTA_DATATTR ANS_IE_DEVICEI D	1	String	设备ID。
EN_IOTA_DATATTR ANS_IE_REQUEST TID	2	String	请求ID。
EN_IOTA_DATATTR ANS_IE_SERVICEI D	3	String	服务ID。

枚举项	枚举值	类型	描述
EN_IOTA_DATATTR ANS_IE_METHOD	4	String	服务方法。
EN_IOTA_DATATTR ANS_IE_CMDCONTENT	5	String	命令内容，以Json格式进行拼装的服务命令参数，开发者根据对应服务命令的定义进行Json解析，从而获取命令参数值。

### 6.3.10 EN\_IOTA\_DEVUPDATE\_IE\_TYPE 消息信元枚举说明

枚举项	枚举值	类型	描述
EN_IOTA_DEVUPDATE_IE_RESULT	0	unsigned int	命令执行返回结果。 <ul style="list-style-type: none"><li>● 成功：0。</li><li>● 失败：1。</li></ul>
EN_IOTA_DEVUPDATE_IE_DEVICE_ID	1	String	设备ID。

## 6.4 数据类型定义

### 1. 常用数据类型

类型名称	类型原型
HW_INT	int
HW_UINT	unsigned int
HW_CHAR	char
HW_UCHAR	unsigned char
HW_BOOL	int
HW ULONG	unsigned long
HW USHORT	unsigned short
HW_MSG	void*
HW_VOID	void
HW_NULL	0

## 2. 函数标准返回值

返回值名称	值	类型原型
HW_OK	0	执行成功。
HW_ERR	1	执行错误。
HW_ERR_PTR	2	错误的指针。
HW_ERR_ID	3	错误的ID。
HW_ERR_PARA	4	错误的参数。
HW_ERR_KEY	5	错误的KEY。
HW_ERR_NOMEM	6	内存不足。
HW_ERR_MAGIC	7	保留。
HW_ERR_OVERFLOW	8	存在溢出。
HW_ERR_GVAR	9	保留。
HW_ERR_POOL	10	保留。
HW_ERR_NO_MUTEX	11	未加锁。
HW_ERR_PID	12	保留。
HW_ERR_FILEOPEN	13	文件打开失败。
HW_ERR_FD	14	错误的文件描述符。
HW_ERR_SOCKET	15	SOCKET异常。
HW_ERR_NOTSUPPORT	16	不支持。
HW_ERR_NOTLOAD	17	未加载。
HW_ERR_ENCODE	18	编码错误。
HW_ERR_DECODE	19	解码错误。
HW_ERR_CALLBACK	22	错误的回调函数。
HW_ERR_STATE	23	错误的状态。
HW_ERR_OVERTIMES	24	重试超过次数。
HW_ERR_ENDOVER	25	保留。
HW_ERR_ENDLINE	26	保留。
HW_ERR_NUMBER	27	错误的数字。
HW_ERR_NOMATCH	28	不匹配。
HW_ERR_NOSTART	29	未开始。
HW_ERR_NOEND	30	未结束。

返回值名称	值	类型原型
HW_ERR_OVERLAP	31	保留。
HW_ERR_DROP	32	丢弃。
HW_ERR_NODATA	33	无数据。
HW_ERR_CRC_CHK	34	CRC校验失败。
HW_ERR_AUTH	35	鉴权失败。
HW_ERR_LENGTH	36	长度错误。
HW_ERR_NOTALLOW	37	不被允许的操作。
HW_ERR_TOKEN	38	凭据错误。
HW_ERR_NOTIPV4	39	不支持IPV4。
HW_ERR_NOTIPV6	40	不支持IPV6。
HW_ERR_IESTOL	41	保留。
HW_ERR_IESTOL1	42	保留。
HW_ERR_IESTOL2	43	保留。
HW_ERR_AUDIO	44	保留。
HW_ERR_VIDEO	45	保留。
HW_ERR_MD5	46	保留。
HW_ERR_MD5_HA1	47	保留。
HW_ERR_MD5_RES	48	保留。
HW_ERR_DIALOG	49	错误的对话。
HW_ERR_OBJ	50	错误的对象。

# 7 Agent Lite API 参考(Java)

[开发者必读](#)

[接口列表](#)

[常用数据结构定义](#)

## 7.1 开发者必读

### 1. 概述

IoT Agent Lite在智慧家庭、工业物联网、车联网等领域为智能设备提供了标准接入物联网平台的能力。主要面向IPC、轻量级网关、工业网关、车机等计算能力较强的终端/网关设备。

### 2. 接口全景图

集成开发者通过全景图了解到：Agent Lite提供了哪些功能？每个功能包含哪些接口？接口之间的逻辑关系如何？从而更快速的找到正确的接口来实现具体业务。

功能	接口	说明
直连设备接入	BaseService.init	初始化Agent Lite资源。
	BaseService.destroy	释放Agent Lite资源。
	BindConfig.setConfig	绑定配置。
	BindService.bind	设备绑定。
	HubService.TOPIC_UNBIN DDEVICE	设备解绑定命令接收。
	LoginConfig.setConfig	登录配置。
	LoginService.login	设备登录。
	LoginService.logout	设备登出。

功能	接口	说明
网关管理非直连设备	HubService.addDevice	设备添加。
	HubService.updateDeviceStatus	设备状态更新。
	HubService.rmvDevice	设备删除。
设备服务数据上报	DataTransService.dataReport	设备服务数据上报。
设备命令接收	DataTransService.TOPIC_COMMAND_RECEIVE	设备服务命令接收。

## 7.2 接口列表

Agent Lite对外提供的接口主要包括观察者模式、直连设备的接入、非直连设备的添加和删除、设备数据的上报以及设备命令接收接口。

### 7.2.1 观察者模式

Java版本 AgentLite采用了观察者模式，用来实现AgentLite上报的消息。

第三方通过实现AgentLite提供的**public void update (IotaMessage arg0)**接口来接收数据。

通过**registerObserver(MyObserver o)**来注册观察者，**removeObserver(MyObserver o)**来移除观察者。

### 7.2.2 直连设备接入

#### 7.2.2.1 初始化 AgentLite 资源

##### 接口功能

初始化AgentLite资源。

##### 接口描述

```
public static boolean init(String workPath, String logPath);
```

##### 接口所属类

BaseService

## 参数说明

字段	必选/可选	类型	描述
workPath	必选	String	AgentLite工作路径，用于存放Agent Lite的配置文件与生产的临时文件，工作路径必须有效。
logPath	可选	String	日志路径（若日志路径为空则日志写在工作路径中）。

## 接口返回值

返回值	描述
true	成功。
false	失败。

## 示例

```
// 开发者调用该接口初始化Agent Lite资源  
BaseService.init("/sdcard/helloWorld", null);
```

### 7.2.2.2 释放 AgentLite 资源

## 接口功能

释放Agent Lite申请的所有资源（内存，线程等等）。

## 接口描述

```
public static boolean destroy();
```

## 接口所属类

BaseService

## 参数说明

无

## 接口返回值

返回值	描述
true	成功。
false	失败。

## 示例

```
// 开发者调用该接口释放Agent Lite资源  
BaseService.destroy();
```

### 7.2.2.3 绑定配置

#### 接口功能

在设备绑定前配置平台的地址和端口。

#### 接口描述

```
public static boolean setConfig(int key, String value);
```

#### 接口所属类

BindConfig

#### 参数说明

字段	必选/可选	类型	描述
key	必选	int	设备绑定的配置项。 <ul style="list-style-type: none"><li>● 平台IP: BindConfig.BIND_CONFIG_ADDR。</li><li>● 平台端口: BindConfig.BIND_CONFIG_PORT。</li></ul>
value	必选	String	设置的值。 <ul style="list-style-type: none"><li>● 平台IP: Agent Lite对接平台地址。</li><li>● 平台端口: 8943。</li></ul>

#### 接口返回值

返回值	描述
true	成功。
false	失败。

## 示例

```
// 开发者调用该接口配置设备绑定前平台的地址和端口  
BindConfig.setConfig(BindConfig.BIND_CONFIG_ADDR, "127.0.0.1");  
BindConfig.setConfig(BindConfig.BIND_CONFIG_PORT, "8943");
```

## 7.2.2.4 设备绑定

### 接口功能

设备第一次接入物联网平台时需要进行绑定操作，上层应用通过调用该接口传入设备序列号或者MAC地址以及设备信息来绑定到物联网平台。

在绑定前需要调用**BindConfig.setConfig**接口设置绑定服务器IP与端口（IoCM服务器地址与端口，Agent Lite会配置默认端口8943）。

### 接口描述

```
public static boolean bind(String verifyCode, IotaDeviceInfo deviceInfo);
```

### 接口所属类

BindService

### 参数说明

字段	必选/可选	类型	描述
verifyCode	必选	String	设备绑定验证码。
deviceInfo	必选	IotaDeviceInfo	设备信息。

### 接口返回值

返回值	描述
true	成功。
false	失败。

#### 说明

- 此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明绑定成功，绑定成功需要收到**BindService**发出的通知。
- 当前绑定流程的重试策略为：如果绑定失败，则30秒后继续进行重试，如果重试超过5次（总计尝试超过6次），则返回失败，不再进行重试。如果想要重新发起绑定，建议让用户重启设备。

### 示例

开发者调用设备绑定接口：

```
String verifyCode = "123456";
deviceInfo = new IotaDeviceInfo(nodeId, manufactureId, deviceType, model, protocolType);
BindService.bind(verifyCode, deviceInfo);
```

开发者调用绑定接口前需要实现**AgentLite**提供的观察者接口。

```
// 调用bindService.registerObserver(this)进行注册，以便接收绑定结果回调，获取回调参数，在登录配置时使用
```

```
public class AgentliteBind implements MyObserver {  
    public Subscribe (Observable bindService) {  
        bindService. registerObserver (this);  
    }  
    }  
    //绑定结果回调，在AgentliteBind中重写update方法  
    @Override  
    public void update(IotaMessage arg0) {  
        System.out.println("BindManager收到绑定通知：" + arg0);  
        int status = arg0.getUint(BindService.BIND_IE_RESULT, -1);  
        System.out.println("status is :" + status);  
        switch (status) {  
            case 0:  
                saveBindParaAndGotoLogin(arg0);  
                break;  
            default:  
                System.out.println("===== 绑定失败 === ");  
                bindAction();  
                break;  
        }  
    }  
}
```

接收设备绑定响应消息。

```
// 当设备绑定成功后，Agent Lite会返回给UI如下几个参数：appId, deviceId, secret需要UI进行持久化存储，设备登录前需要提前进行配置。  
//保存绑定响应消息携带的参数  
private void saveBindParaAndGotoLogin(IotaMessage iotaMsg) {  
    String appId = iotaMsg.getString(BindService.BIND_IE_APPID);  
    String deviceId = iotaMsg.getString(BindService.BIND_IE_DEVICEID);  
    String secret = iotaMsg.getString(BindService.BIND_IE_DEVICESECRET);  
    String haAddress = null, lvsAddress = null;  
    saveGatewayInfo(appId, deviceId, secret, haAddress, lvsAddress);  
}
```

## 7.2.2.5 设备解绑定命令接收

### 接口功能

被动接收平台下发的解绑定命令，删除直连设备的配置信息并且释放所有资源。(接到此命令说明设备已经从平台侧删除。)

### 接口描述

```
HubService.TOPIC_UNBINDDEVICE;
```

### 接口所属类

HubService。

### 示例

调用该接口需要实现AgentLite提供的观察者接口。

```
public class AgentLiteHub implements MyObserver {  
    public Subscribe (Observable hubService) {  
        hubService. registerObserver (this);  
    }  
}
```

接收设备解绑定响应消息。

```
// 当设备解绑定成功后，开发者收到该回调后需要删除直连设备的配置信息并且释放所有资源。  
public void update(IotaMessage arg0) {  
    // TODO Auto-generated method stub
```

```
System.out.println("收到hubservice通知:" + arg0);
int mMsgType = arg0.getMsgType();
switch(mMsgType) {
//收到添加设备的应答
case IodevService.IODEV_MSG_ADD_DEVICE_RSP:
getAddDeviceAnswer(arg0);
break;
//收到删除设备的应答
case IodevService.IODEV_MSG_RMV_DEVICE_RSP:
getRmvDeviceAnswer(arg0);
break;
//收到更新设备信息的应答
case IodevService.IODEV_MSG_UPDATE_DEVSTATUS_RSP:
getUpdateStatusAnswer(arg0);
break;
case IodevService.IODEV_MSG_RECEIVE_CMD:
getUnbindAnswer(arg0);
break;

default:
break;
}
}
```

## 7.2.2.6 参数配置

### 7.2.2.6.1 配置业务参数

#### 接口功能

在登录前配置登录所需要的参数。

#### 接口描述

```
public static boolean setConfig(int key, String value);
```

#### 接口所属类

LoginConfig

## 参数说明

字段	必选/可选	类型	描述
key	必选	int	<p>设备登录的配置项。</p> <ul style="list-style-type: none"><li>● 设备ID: LoginConfig.LOGIN_CONFIG_DEVICEID。</li><li>● AppId: LoginConfig.LOGIN_CONFIG_APPID。</li><li>● 密码: LoginConfig.LOGIN_CONFIG_SECRET。</li><li>● HTTP地址: LoginConfig.LOGIN_CONFIG_IOCM_ADDR。</li><li>● HTTP端口: LoginConfig.LOGIN_CONFIG_IOCM_PORT。</li><li>● MQTT地址: LoginConfig.LOGIN_CONFIG_MQTT_ADDR。</li><li>● MQTT端口: LoginConfig.LOGIN_CONFIG_MQTT_PORT。</li></ul>
value	必选	String	<p>设置的值。</p> <ul style="list-style-type: none"><li>● 设备ID: 从绑定成功的回调中获取。</li><li>● AppId: 从绑定成功的回调中获取。</li><li>● 密码: 从绑定成功的回调中获取。</li><li>● HTTP地址: Agent Lite对接平台南向接入地址。</li><li>● HTTP端口: 8943。</li><li>● MQTT地址: Agent Lite对接平台南向接入地址。</li><li>● MQTT端口: 8883。</li></ul>

## 接口返回值

返回值	描述
true	成功。
false	失败。

## 示例

```
//设置登录配置(deviceId, secret, appid是bind返回的参数)
private void configLoginPara() {
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_DEVICEID, GatewayInfo.getDeviceID());
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_APPID, GatewayInfo.getAppID());
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_SECRET, GatewayInfo.getSecret());
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_IOCM_ADDR, GatewayInfo.getHaAddress());
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_IOCM_PORT, "8943");
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_MQTT_ADDR, GatewayInfo.getHaAddress());
    LoginConfig.setConfig(LoginConfig.LOGIN_CONFIG_MQTT_PORT, "8883");
}
```

### 7.2.2.6.2 配置加密算法类型（可选）

#### 接口功能

在登录前配置敏感信息加密算法类型，业务可以根据安全级别选择合适的加密算法类型。

#### 接口描述

```
public static boolean setAlgType (int type);
```

#### 接口所属类

BaseService

#### 参数说明

字段	必选/可选	类型	描述
type	必选	int	加密算法类型。 ● 0: AES 256 CBC。 ● 1: AES 128 GCM。 ● 2: AES 256 GCM。

#### 接口返回值

返回值	描述
true	成功。
false	失败。

#### 实例

```
//设置加密算法配置
private void configAlgPara() {
    BaseService. setAlgType (2);
}
```

### 7.2.2.7 设备登录

#### 接口功能

设备在第一次绑定后，或者在设备重启后需要进行登录的流程。

#### 接口描述

```
public static boolean login();
```

## 接口所属类

LoginService

## 参数说明

无

## 接口返回值

返回值	描述
true	成功。
false	失败。



### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明登录成功，登录成功需要收到**LoginService**发出的通知。登录前通过[配置业务参数](#)接口（`LoginConfig.setConfig`）传入所需的登录信息。

## 示例

设备登录。

```
LoginService.login();
```

开发者调用登录接口前需要实现AgentLite提供的观察者接口。

```
public class AgentliteLogin implements MyObserver {  
    public AgentliteLogin (Observable loginService) {  
        loginService. registerObserver (this);  
    }  
    //登录结果回调，在AgentliteLogin中重写update方法  
    @Override  
    public void update(IotaMessage arg0) {  
        // TODO Auto-generated method stub  
        System.out.println("LoginManager收到通知：" + arg0);  
        int mMsgType = arg0.getMsgType();  
        switch(mMsgType) {  
            case 1:  
                loginResultAction(arg0);  
                break;  
            case 2:  
                logoutResultAction(arg0);  
                break;  
            default:  
                break;  
        }  
    }  
}
```

## 7.2.2.8 设备登出

### 接口功能

断开与IoT平台的连接。

## 接口描述

```
public static boolean logout();
```

## 接口所属类

LoginService

## 参数说明

无

## 接口返回值

返回值	描述
true	成功。
false	失败。



### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明登出成功，登出成功需要收到**LoginService**发出的通知。

## 示例

开发者调用设备登出接口。

```
LoginService.logout();
```

开发者调用登出接口前需要实现AgentLite提供的观察者接口。

```
public class AgentliteLogin implements MyObserver {  
    public AgentliteLogin (Observable loginService) {  
        loginService.registerObserver (this);  
    }  
    //登出结果回调，在AgentliteLogin中重写update方法  
  
    @Override  
    public void update(IotaMessage arg0) {  
        // TODO Auto-generated method stub  
        System.out.println("LoginManager收到通知：" + arg0);  
        int mMsgType = arg0.getMsgType();  
        switch(mMsgType) {  
            case 1:  
                loginResultAction(arg0);  
                break;  
            case 2:  
                logoutResultAction(arg0);  
                break;  
            default:  
                break;  
        }  
    }  
}
```

## 7.2.3 网关管理非直连设备

### 7.2.3.1 设备添加

#### 接口功能

当有新设备接入网关后，通过调用设备添加接口将非直连设备接入物联网平台，并且获得平台分配的唯一设备逻辑ID。

#### 接口描述

```
public static boolean addDevice(int cookie, IotaDeviceInfo deviceInfo);
```

#### 接口所属类

HubService

#### 参数说明

字段	必选/可选	类型	描述
cookie	可选	int	Cookie有效值为1-65535。
deviceInfo	必选	IotaDeviceInfo类说明	设备信息。

#### 接口返回值

返回值	描述
true	成功。
false	失败。



#### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明添加成功，添加成功需要收到HubService发出的通知。

#### 示例

开发者调用设备添加接口。

```
HubService.addDevice(29011, new IotaDeviceInfo("nodeId", "manufacturerId", "deviceType", "model", "protocolType"));
```

开发者调用添加设备接口前需要实现AgentLite提供的观察者接口。

```
public class AgentliteHub implements MyObserver {  
    public AgentliteHub (Observable hubService) {  
        hubService.registerObserver (this);  
    }  
    // 接收设备添加响应消息  
    @Override  
    public void update(IotaMessage arg0) {  
        // TODO Auto-generated method stub  
        System.out.println("收到hubservice通知:" + arg0);  
    }  
}
```

```
int mMMsgType = arg0.getMsgType();
switch(mMMsgType) {
//收到添加设备的应答
case IodevService.IODEV_MSG_ADD_DEVICE_RSP:
getAddDeviceAnswer(arg0);
break;
//收到删除设备的应答
case IodevService.IODEV_MSG_RMV_DEVICE_RSP:
getRmvDeviceAnswer(arg0);
break;
//收到更新设备信息的应答
case IodevService.IODEV_MSG_UPDATE_DEVSTATUS_RSP:
getUpdateStatusAnswer(arg0);
break;
case IodevService.IODEV_MSG_RECEIVE_CMD:
getUnbindAnswer(arg0);
break;

default:
break;
}
}
```

### 7.2.3.2 设备状态更新

#### 接口功能

通过该接口更新设备的状态信息，包括直连设备与所管理的非直连设备。设备离线、上线均可通过该接口刷新设备状态信息。

直连设备状态通过设备的登录状态进行管理，当直连设备连接断开则表示设备离线，当直连设备连接或重连成功，则表示设备上线，无需通过该接口进行刷新。故建议开发者使用该接口刷新非直连设备的状态。

#### 接口描述

```
public static boolean updateDeviceStatus(int cookie, String deviceId, String status, String statusDetail);
```

#### 接口所属类

HubService

#### 参数说明

字段	必选/可选	类型	描述
cookie	可选	int	Cookie有效值为1-65535。
deviceId	必选	String	设备Id。
status	必选	String	设备状态。 <ul style="list-style-type: none"><li>● 在线: ONLINE。</li><li>● 离线: OFFLINE。</li></ul>

statusDetail	必选	String	设备状态详细信息。 <ul style="list-style-type: none"><li>● 无: NONE。</li><li>● 配置等待: CONFIGURATION_PENDING。</li><li>● 通信错误: COMMUNICATION_ERROR。</li><li>● 配置错误: CONFIGURATION_ERROR。</li><li>● 桥接器离线: BRIDGE_OFFLINE。</li><li>● 固件升级: FIRMWARE_UPDATING。</li><li>● 循环任务: DUTY_CYCLE。</li><li>● 未激活: NOT_ACTIVE。</li></ul>
--------------	----	--------	--

## 接口返回值

返回值	描述
true	成功。
false	失败。

### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明设备状态更新成功，设备状态更新成功需要收到**HubService**发出的通知。

## 示例

开发者调用设备状态更新接口。

```
HubService.deviceStatusUpdate(0, deviceId, "ONLINE", "NONE");
```

开发者调用设备状态更新接口需前要实现**AgentLite**提供的观察者接口。

```
public class Subscribe implements MyObserver {  
    public Subscribe (Observable hubService) {  
        hubService.registerObserver (this);  
    }  
  
    @Override  
    同设备添加
```

### 7.2.3.3 设备删除

## 接口功能

当有新设备需要从网关移除时，通过调用设备删除接口将非直连设备从物联网平台删除。

## 接口描述

```
public static boolean rmvDevice(int cookie, String deviceId);
```

## 接口所属类

HubService

## 参数说明

字段	必选/可选	类型	描述
cookie	可选	int	Cookie有效值为1-65535。
deviceId	必选	String	设备Id。

## 接口返回值

返回值	描述
true	成功。
false	失败。



### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明删除成功，设备删除成功需要收到**HubService**发出的通知。

## 示例

开发者调用设备删除接口。

```
HubService.rmvDevice(122, deviceId);
```

开发者调用设备删除接口前需要实现AgentLite提供的观察者接口。

```
public class Subscribe implements MyObserver {  
    public Subscribe (Observable hubService) {  
        hubService. registerObserver (this);  
    }  
    @Override  
    同设备添加
```

## 7.2.4 设备服务数据上报

## 接口功能

当直连设备（网关）有数据需要上报或者非直连设备上报数据到网关时，网关需要调用设备服务数据上报接口将数据上报到物联网平台。

## 接口描述

```
public static boolean dataReport(int cookie, String requestId, String deviceId, String serviceId,  
String serviceProperties);
```

## 接口所属类

DataTransService

## 参数说明

字段	必选/可选	类型	描述
cookie	可选	int	Cookie有效值1-65535。
requestId	必选	String	请求ID，匹配之前平台下发的服务命令。可以从设备命令接收的广播中获取requestId。 <ul style="list-style-type: none"><li>● 主动数据上报：requestId为NULL。</li><li>● 命令结果上报：当上报的数据匹配到某一次命令请求时，需要填写此次命令请求的请求ID。</li></ul>
deviceId	必选	String	设备ID。
serviceId	必选	String	服务ID，在设备profile中定义的serviceId参数值。
serviceProperties	必选	String	服务属性，在设备profile中定义的属性参数名与属性值的json字符串，属性值为具体设备要上报的数据。

## 接口返回值

返回值	描述
true	成功。
false	失败。

### 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明服务数据上报成功，数据上报成功需要收到DataTransService发出的通知。

## 示例

开发者调用数据上报接口。

```
DataTransService.dataReport(1211, NULL, "xxxx_xxxx_xxxx_xxxx", "DoorWindow", "{\"status\": \"OPEN\"}");
```

开发者调用数据上报接口前需要实现AgentLite提供的观察者接口。

```
public class Subscribe implements MyObserver {  
    public Subscribe (Observable dataTransService) {  
        dataTransService.registerObserver (this);  
    }  
    @Override
```

```
public void update(IotaMessage arg0) {  
    // TODO Auto-generated method stub  
    System.out.println("AgentLiteDataTrans收到通知:" + arg0);  
    int mMsgType = arg0.getMsgType();  
    switch(mMsgType) {  
        //数据上报应答  
        case IodevService.IODEV_MSG_DATA_REPORT_RSP:  
            getDataReportAnswer(arg0);  
            break;  
        //被动接收命令  
        case IodevService.IODEV_MSG_RECEIVE_CMD:  
            getCmdReceive(arg0);  
            break;  
        //MQTT消息推送  
        case IodevService.IODEV_MSG_MQTT_PUB_RSP:  
            //logoutResultAction(iotaMsg);  
            break;  
        default:  
            break;  
    }  
}
```

## 7.2.5 设备命令接收

### 接口功能

调用设备命令接收接口来处理平台下发的控制命令。

### 接口描述

```
DataTransService.TOPIC_COMMAND_RECEIVE;
```

### 接口所属类

DataTransService

### 参数说明

无

### 接口返回值

无

### 示例

开发者需要实现AgentLite提供的观察者接口。

```
public class AgentLiteDataTrans implements MyObserver {  
    public AgentLiteDataTrans (Observable dataTransService) {  
        dataTransService.registerObserver (this);  
    }  
    @Override  
    public void update(IotaMessage arg0) {  
        // TODO Auto-generated method stub  
        System.out.println("AgentLiteDataTrans收到通知:" + arg0);  
        int mMsgType = arg0.getMsgType();  
        switch(mMsgType) {  
            //数据上报应答  
            case IodevService.IODEV_MSG_DATA_REPORT_RSP:  
                getDataReportAnswer(arg0);  
                break;
```

```
//被动接收命令
case IodevService.IODEV_MSG_RECEIVE_CMD:
getCmdReceive(arg0);
break;
//MQTT消息推送
case IodevService.IODEV_MSG_MQTT_PUB_RSP:
//logoutResultAction(iotaMsg);
break;
default:
break;
}
```

## 7.2.6 数据发布

### 接口功能

当直连设备（网关）有数据需要发布或者非直连设备发布数据到网关时，网关需要调用设备服务数据发布接口将数据发布到物联网平台。其与数据上报功能上的区别在于，发布的数据可以自己规定Topic，同时JSON字符串需要自己组装传入接口，SDK只做透传。

### 接口描述

```
public static boolean mqttDataPub(int cookie, String topic, int qos, byte[] serviceData);
```

### 接口所属类

DataTransService

### 参数说明

字段	必选/可选	类型	描述
uiCookie	可选	int	Cookie有效值1-65535。
pucTopic	必选	String	发布数据的Topic。
uiQos	必选	int	Mqtt协议的相关参数，一般为“1”。
pbstrServiceData	必选	byte[]	发布数据的包体。

### 接口返回值

返回值	描述
true	成功。
false	失败。



## 说明

此返回值是调用接口的同步返回结果，返回true只是说明接口调用成功，并不说明服务数据发布成功，数据发布成功需要收到**DataTransService**发出的通知。

## 示例

开发者调用数据发布接口。

```
DataTransService.mqttDataPub(1211, "/huawei/v1/devices/336d9bac-9ebf-44e9-95cf-efac5f05da3a/services/Storage", 1, bstrBody);
```

开发者调用数据发布接口前需要实现**AgentLite**提供的观察者接口。

```
public class Subscribe implements MyObserver {  
    public Subscribe(Observerable dataTransService) {  
        dataTransService.registerObserver(this);  
    }  
    @Override  
    public void update(IotaMessage arg0) {  
        // TODO Auto-generated method stub  
        System.out.println("AgentLiteDataTrans收到通知：" + arg0);  
        int mMsgType = arg0.getMsgType();  
        switch(mMsgType) {  
            //数据上报应答  
            case IodevService.IODEV_MSG_DATA_REPORT_RSP:  
                getDataReportAnswer(arg0);  
                break;  
            //被动接收命令  
            case IodevService.IODEV_MSG_RECEIVE_CMD:  
                getCmdReceive(arg0);  
                break;  
            //MQTT消息推送  
            case IodevService.IODEV_MSG_MQTT_PUB_RSP:  
                //logoutResultAction(iotaMsg);  
                break;  
            case IodevService.IODEV_MSG_MQTT_SUB_RSP:  
                //TopicSubscribeResultAction(iotaMsg);  
                break;  
            case IodevService.IODEV_MSG_MQTT_DATA_RECV_RSP:  
                //DataRecvAction(iotaMsg);  
                break;  
            default:  
                break;  
        }  
    }  
}
```

## 7.3 常用数据结构定义

### 7.3.1 IotaDeviceInfo 类说明

字段	必选/可选	类型	描述
nodeId	必选	String	关键参数，对接平台的网关下设备唯一标识，设备填写，平台用于判重。
name	可选	String	设备名称。
description	可选	String	设备描述。

字段	必选/可选	类型	描述
manufacturerId	必选	String	厂商ID。
manufacturerName	可选	String	厂商名。
mac	可选	String	设备MAC地址。
location	可选	String	设备的位置。
deviceType	必选	String	设备类型。 <b>说明</b> 如果接入设备是网关设备，则“DeviceType”填写为“Gateway”，如果接入传感器是直连设备，则填写为对应的设备类型。
model	必选	String	型号。 <ul style="list-style-type: none"><li>● 直连设备：与Profile中定义的model保持一致即可。</li><li>● Z-Wave设备： ProductType + ProductId，格式（使用16进制）：XXXX-XXXX 补0对齐，如：001A-0A12，其他协议再定。</li></ul>
swVersion	可选	String	软件版本。 Z-Wave：主版本号.次版本号，如：1.1。
fwVersion	可选	String	固件版本。
hwVersion	可选	String	硬件版本。
protocolType	必选	String	协议类型：Z-Wave。
bridgeId	可选	String	表示设备通过哪个Bridge接入平台。
status	可选	String	表示设备是否在线。 <ul style="list-style-type: none"><li>● ONLINE 在线。</li><li>● OFFLINE 离线。</li></ul>

字段	必选/可选	类型	描述
statusDetail	可选	String	<p>状态详情，如果pcStatus不为空，则该参数必选。</p> <p>参数值：</p> <ul style="list-style-type: none"> <li>● 无: NONE。</li> <li>● 配置等待: CONFIGURATION_PENDING。</li> <li>● 通信错误: COMMUNICATION_ERROR。</li> <li>● 配置错误: CONFIGURATION_ERROR。</li> <li>● 桥接器离线: BRIDGE_OFFLINE。</li> <li>● 固件升级: FIRMWARE_UPDATING。</li> <li>● 循环任务: DUTY_CYCLE。</li> <li>● 未激活: NOT_ACTIVE。</li> </ul>
mute	可选	String	<p>表示设备是否被屏蔽。</p> <ul style="list-style-type: none"> <li>● TRUE。</li> <li>● FALSE。</li> </ul>

### 7.3.2 IotaMessage 类说明

```
bind后接收响应获取appid, deviceid, secret:  
String appId = iotaMsg.getString(BindService.BIND_IE_APPID);  
String deviceId = iotaMsg.getString(BindService.BIND_IE_DEVICEID);  
String secret = iotaMsg.getString(BindService.BIND_IE_DEVICESECRET);
```

**上报数据收到响应:**

```
String deviceld = iotaMsg.getString(DataTransService.DATATRANS_IE_DEVICEID);  
int retcode = iotaMsg.getUint(DataTransService.DATATRANS_IE_RESULT, 0);  
int cookie = iotaMsg.getUint(DataTransService.DATATRANS_IE_COOKIE, 0);
```

**命令接收:**

```
String deviceId = iotaMsg.getString(DataTransService.DATATRANS_IE_DEVICEID);  
String requestId = iotaMsg.getString(DataTransService.DATATRANS_IE_REQUSTID);  
String serviceId = iotaMsg.getString(DataTransService.DATATRANS_IE_SERVICEID);  
String method = iotaMsg.getString(DataTransService.DATATRANS_IE_METHOD);  
String cmd = iotaMsg.getString(DataTransService.DATATRANS_IE_CMDCONTENT);
```

**设备添加收到响应 (ret = 0表示成功) :**

```
String deviceId = iotaMsg.getString(HubService.HUB_IE_DEVICEID);  
int ret = iotaMsg.getUint(HubService.HUB_IE_RESULT, HubService.HUB_RESULT_FAILED);
```

**设备删除收到响应 (result = 0表示成功) :**

```
int result = iotaMsg.getUint(HubService.HUB_IE_RESULT, 0);  
int cookie = iotaMsg.getUint(HubService.HUB_IE_COOKIE, 0);
```

```
更新设备状态收到响应 (result = 0表示成功) :  
int result = iotaMsg.getUint(HubService.HUB_IE_RESULT, 0);  
int cookie = iotaMsg.getUint(HubService.HUB_IE_COOKIE, 0);
```

```
收到设备登出 (错误码信息对照loginService定义):  
int reason = iotaMsg.getUint(LoginService.LOGIN_IE_RESULT, -1);
```

# 8 IoT 平台 MQTT 接口参考

概述

接口列表

## 8.1 概述

### 8.1.1 MQTT 介绍

MQTT标准规范参见《[mqtt-v3.1.1-os.pdf](#)》。



说明

MQTT的语法和接口细节，请以此标准为准。平台目前只支持MQTTTS接入。

### 8.1.2 消息格式

MQTT消息分为固定报头（Fixed header）、可变报头（Variable header）和有效载荷（Payload）部分。

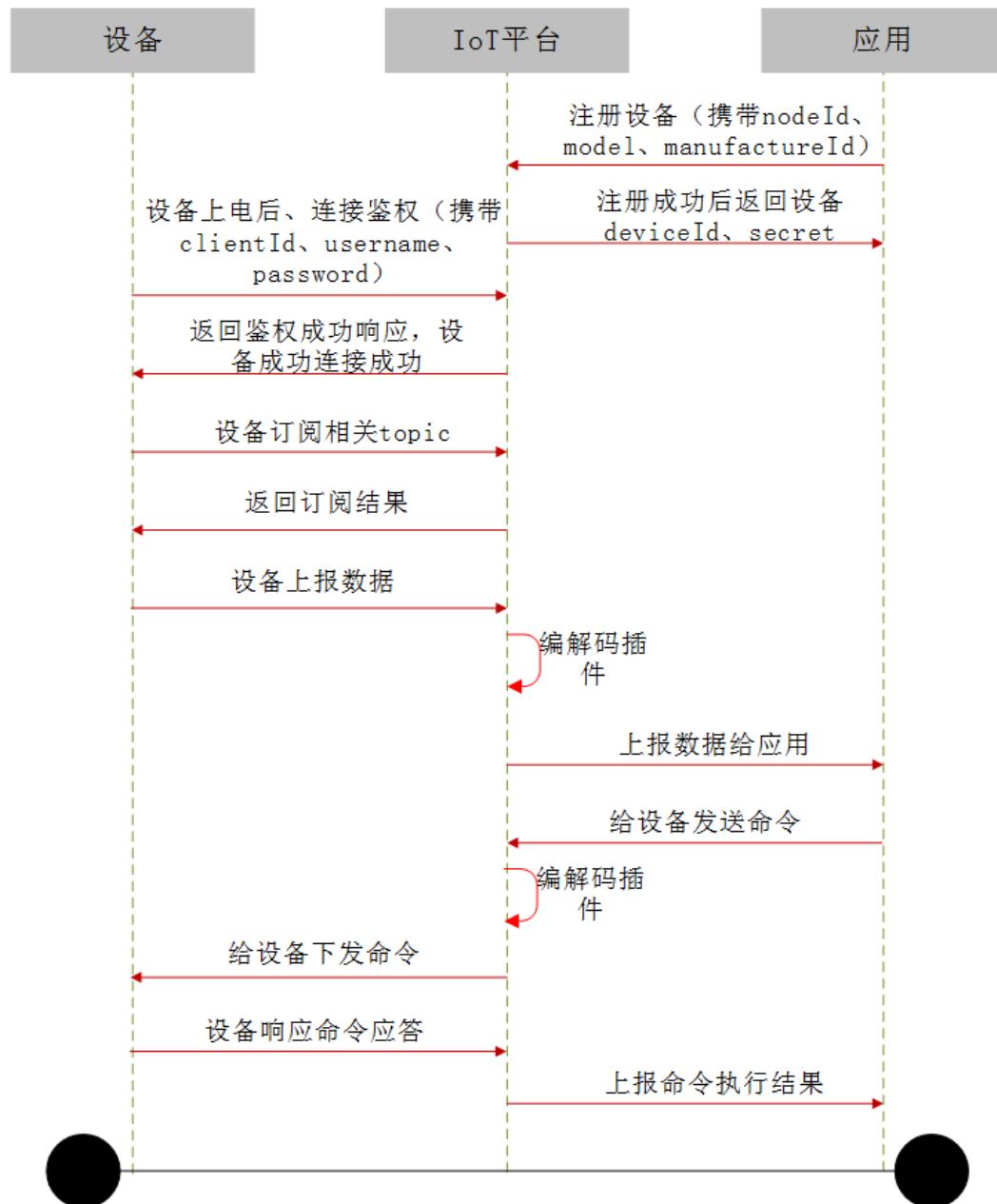
固定报头（Fixed header）和可变报头（Variable header）格式的填写直接参考[MQTT标准规范](#)。有效载荷（Payload）部分在PUB消息中可以由应用定义，即设备和IoT平台之间自己定义。

下面主要介绍CONNECT、SUB和PUB消息格式的填写。

- CONNECT - Client requests a connection to a server  
有效载荷（Payload）中的主要参数填写，具体参见[MQTT CONNECT连接鉴权](#)。
- SUBSCRIBE - Subscribe to named topics  
有效载荷（Payload）中的主要参数填写：Topic name，填写为设备想要订阅的主题消息，目前填写为设备自己的topic，具体参见[Topic定义](#)。
- PUBLISH - Publish message
  - 可变报头（Variable header）：Topic name，设备发往IoT平台时，为平台的Topic name，设备接收消息时，为设备的Topic name，具体参见[Topic定义](#)。
  - 有效载荷（Payload）中的主要参数填写：为完整的数据上报和命令下发的消息内容，目前是一个json对象。

## 8.1.3 流程说明

### 8.1.3.1 业务流程



1. 应用向IoT平台发送设备注册信息。
2. 设备注册成功则返回deviceId和secret，注册失败则返回失败原因。
3. 设备上电后，向IoT平台发起连接请求，携带clientId、username、password，具体设置参见[MQTT CONNECT连接鉴权](#)。
4. IoT平台鉴权成功，返回成功响应，设备连接IoT平台成功。
5. 设备携带包含自身deviceId的topic向IoT平台发起订阅请求。
6. IoT平台返回订阅结果。

7. 设备上报数据给IoT平台，携带相应的topic和数据进行publish。
8. IoT平台通过相应的编解码插件对数据进行解码。
9. IoT平台通过topic将成功解码后的数据发送给对应的应用。
10. 应用下发命令给IoT平台。
11. IoT平台通过相应的编解码插件对数据进行编码。
12. IoT平台将编码后的消息发送给设备。
13. 设备接收到命令，并对命令做出响应应答回复给IoT平台。
14. IoT平台将命令执行结果发送给应用。



#### 说明

应用与IoT平台交互接口文档参考IoT平台API接口参考。

### 8.1.3.2 设备注册（一机一密）

一机一密支持通过以下两种方式来进行：

- 调用北向API注册设备

通过注册设备接口（“/iocm/app/reg/v2.0.0/deviceCredentials”）在IoT平台进行一机一密设备注册，一机一密与原有注册方式主要差别是注册成功会返回设备secret，该接口新增mqttConnect字段区分一机一密的设备。

参数	必选/可选	类型	描述
mqttConnect	可选	Boolean	<ul style="list-style-type: none"><li>● “True”：表示一机一密类型，开户成功后返回设备id和密码。</li><li>● “False”：表示原有的动态密码场景，开户成功过后返回设备verifyCode，设备首次接入平台时通过验证码再获取设备id和密码。</li></ul>

- 在管理门户注册设备

选择“设备管理 > 设备 > 设备注册 > 单个注册”，点击“创建”，根据页面内容填写设备信息，册成功后返回设备“deviceId”和“secret”。

## 8.2 接口列表

### 8.2.1 Topic 定义

#### 设备订阅消息 topic

设备如果要从平台获取消息和数据，需要在登录成功后发起mqtt消息订阅topic。

订阅消息topic格式为：“/huawei/v1/devices/{deviceId}/command/{codecMode}”。

- “codecMode”为订阅的编解码类型，当用户使用自身开发的编解码插件时“codecMode”取值为“binary”，没有开发编解码插件的“codecMode”取值为“json”。

- “deviceId” 填写为设备注册成功后返回的 “deviceId” 值。

## 设备上报消息 topic

设备如果要通过MQTT通道上报数据，需要发给指定的topic。

上报消息的topic格式为：“/huawei/v1/devices/{deviceId}/data/{codecMode}”。

- “codecMode” 为上报的编解码类型，当用户使用自身开发的编解码插件时“codecMode”取值为“binary”，没有开发编解码插件的“codecMode”取值为“json”。
- “deviceId” 填写为设备注册成功后返回的 “deviceId” 值。

## 8.2.2 MQTT CONNECT 连接鉴权

### 接口功能

IoT平台设备侧支持MQTT协议的connect消息接口，接口规范参考[MQTT协议V3.1.1版本](#)，鉴权通过后建立设备与平台间的MQTT连接。

### 参数说明

参数	必选/可选	类型	参数描述
client Id	必选	String(256)	<p>一机一密的设备clientId由4个部分组成： deviceId、鉴权类型、密码签名类型、时间戳。通过下划线“_”分隔，其中鉴权类型的值为“0”。</p> <ul style="list-style-type: none"><li>鉴权类型：长度1字节，当前支持1个类型：0，表示一机一密。</li><li>密码签名类型：长度1字节，当前支持2种类型：<ul style="list-style-type: none"><li>“0”代表HMACSHA256不校验时间戳。</li><li>“1”代表HMACSHA256校验时间戳。</li></ul></li><li>时间戳：为设备连接平台时的UTC时间，格式为YYYYMMDDHH，如UTC时间2018/7/24 17:56:20则应表示为2018072417。</li></ul>
User name	必选	String(256)	一机一密的设备“Username”为“deviceId”。
Pass word	必选	String(256)	Password的值为使用“HMACSHA256”算法以时间戳为秘钥，对secret进行加密后的值。 secret为注册设备时平台返回的secret。

设备通过MQTT协议的connect消息进行鉴权，对于构造clientId的各个部分信息都必须包括进去，平台收到connect消息时，会判断设备的鉴权类型（0，代表一机一密）和密码摘要算法。

- 当采用“HMACSHA256”校验时间戳方式时，会先校验消息时间戳与平台时间是否一致，再判断密码是否正确。

- 当采用“HMACSHA256”不校验时间戳方式时，鉴权消息也必须带时间戳，但不检验时间是否准确，仅判断密码是否正确。

connect消息鉴权失败时，平台会返回错误，并自动断开MQTT链路。

## 8.2.3 设备数据上报

### 接口功能

用于设备上报数据给平台。

### 参数说明

字段名	必选/可选	类型	参数描述
msgType	必选	String	表示设备上报数据，固定值“deviceReq”。
data	必选	ServiceData[]	一组服务的数据（具体结构参考下表ServiceData定义表），当需要上传批量数据时，可在该字段中添加数据。

ServiceData定义表：

字段名	必选/可选	类型	参数描述
serviceId	必选	String	设备服务的ID。
serviceData	必选	ObjectNode	一个服务的数据，具体字段在profile里定义。
eventTime	可选	String	设备采集数据UTC时间（格式：yyyyMMddTHHmmssZ），如：20161219T114920Z。

### 示例

```
MQTT
Topic: /huawei/v1/devices/{deviceId}/data/{codecMode}
MQTT Client发送的payload:
{
  "msgType": "deviceReq",
  "data": [
    {
      "serviceId": "{serviceId}",
      "serviceData": {
        "meterId": "xxxx",
        "dailyActivityTime": 120,
        "flow": "565656",
        "cellId": "5656",
        "signalStrength": "99",
      }
    }
  ]
}
```

```
"batteryVoltage": "3.5"
}
"eventTime": "20160503T121540Z"
},
{
"serviceId": "Battery",
"serviceData": {"batteryLevel": 75},
"eventTime": "20160503T121540Z"
}
]
```

## 8.2.4 平台命令下发

### 接口功能

用于平台向设备下发设备控制命令。

### 参数说明

字段名	必选/可选	类型	参数描述
msgType	必选	String	表示平台下发的请求，固定值“cloudReq”。
serviceId	必选	String	设备服务的ID。
cmd	必选	String	服务的命令名，参见profile的服务命令定义。
paras	必选	ObjectNode	命令的参数，具体字段由profile定义。
mid	必选	Int	2字节无符号的命令id，平台内部分配（范围1-65535），设备命令响应平台时，需要返回该值。

### 示例

MQTT  
设备先订阅Topic: /huawei/v1/devices/{deviceId}/command/{codecMode} 才能收到命令推送  
北向命令下发使用以下接口:  
<https://server:port/iocom/app/signaltrans/v1.1.0/devices/{deviceId}/services/{serviceId}/sendCommand?appId={appId}>  
MQTT Client接收到的payload:  
{  
"msgType": "{msgType}",  
"serviceId": "{serviceId}",  
"mid": 2016,  
"cmd": "{cmd}",  
"paras": {  
"value": 4  
}  
}

## 8.2.5 设备响应命令

### 接口功能

设备对平台控制命令的应答。

### 参数说明

字段名	必选/ 可选	类型	参数描述
msgType	必选	String	固定值“deviceRsp”，表示设备的应答消息。
mid	必选	Int	2字节无符号的命令ID，根据平台下发命令时的mid返回给平台。建议在消息中携带此参数。
errcode	必选	Int	请求处理的结果码。 <ul style="list-style-type: none"><li>● “0”表示成功。</li><li>● “1”表示失败。</li></ul>
body	可选	ObjectNode	命令的应答，具体字段由profile定义。 <b>说明</b> body体不是数组。

### 示例

```
MQTT
Topic: /huawei/v1/devices/{deviceId}/data/{codecMode}
MQTT Client发送的payload:
{
  "msgType": "{msgType}",
  "mid": {mid},
  "errcode": {errcode},
  "body": {
    "result": 0
  }
}
```