

中国电信物联网开放平台 API 参考

1.5.0

（V1）

中国电信股份有限公司物联网分公司

二〇一八年五月

编制单位：

编制单位	中国电信股份有限公司物联网分公司平台运营部

修订记录：

版本号	日期	描述
01	2018/04/28	中国电信物联网开放平台 API 参考 1.5.0

目 录

1 总体说明.....1

1.1 网络结构.....1

1.2 修订记录.....2

1.3 接口兼容性说明.....3

2 接口列表.....4

2.1 应用安全接入.....4

2.1.1 鉴权.....4

2.1.2 刷新 token.....7

2.2 设备管理.....10

2.2.1 注册直连设备（验证码方式）.....10

2.2.2 注册直连设备（密码方式）.....16

2.2.3 刷新设备密钥（验证码方式）.....23

2.2.4 刷新设备密钥（密码方式）.....27

2.2.5 修改设备信息.....31

2.2.6 删除直连设备.....36

2.2.7 查询设备激活状态.....39

2.3 批量处理.....42

2.3.1 创建批量任务.....42

2.3.2 查询单个任务信息.....46

2.3.3 查询任务详情信息.....51

2.4 规则管理.....56

2.4.1 创建规则.....56

2.4.2 更新规则.....79

2.4.3 删除规则.....85

2.4.4 查找规则.....87

2.4.5 修改规则状态.....91

2.4.6 批量修改规则状态.....93

2.5 订阅管理.....98

2.5.1 订阅平台业务数据.....98

2.5.2 订阅平台管理数据.....101

2.5.3 查询单个订阅 103
2.5.4 批量查询订阅 106
2.5.5 删除单个订阅 109
2.5.6 批量删除订阅 111
2.6 消息推送 114
2.6.1 注册设备通知 114
2.6.2 设备信息变化通知 118
2.6.3 设备数据变化通知 122
2.6.4 批量设备数据变化通知 124
2.6.5 设备服务信息变化通知 127
2.6.6 删除设备通知 129
2.6.7 设备消息确认通知 130
2.6.8 设备命令响应通知 132
2.6.9 设备事件通知 135
2.6.10 增加设备模型通知 137
2.6.11 删除设备模型通知 139
2.6.12 规则事件通知 141
2.6.13 软件升级状态变更通知 144
2.6.14 软件升级结果变更通知 146
2.6.15 固件升级状态变更通知 148
2.6.16 固件升级结果变更通知 150
2.7 命令下发 152
2.7.1 创建设备命令 152
2.7.2 查询设备命令 158
2.7.3 修改设备命令 165
2.7.4 创建设备命令撤销任务 169
2.7.5 查询设备命令撤销任务 175
2.8 数据采集 182
2.8.1 查询单个设备信息 182
2.8.2 批量查询设备信息列表 189
2.8.3 查询设备历史数据 197
2.8.4 查询设备服务能力 201
2.9 设备组管理 208
2.9.1 增加设备组 208
2.9.2 删除设备组 211
2.9.3 修改设备组信息 214
2.9.4 查询设备组详情 217
2.9.5 查询指定设备组 221
2.9.6 查询指定设备组成员 224

2.9.7 向设备组添加设备 227
2.9.8 从设备组删除设备 229
2.10 设备升级 232
2.10.1 查询版本包列表 232
2.10.2 查询指定版本包 236
2.10.3 删除指定版本包 239
2.10.4 创建软件升级任务 240
2.10.5 创建固件升级任务 245
2.10.6 查询指定升级任务结果 249
2.10.7 查询指定升级任务子任务详情 254
2.10.8 查询升级任务列表 258
3 术语表 265

1 总体说明

1.1 网络结构
1.2 修订记录
1.3 接口兼容性说明

1.1 网络结构



中国电信物联网开放平台（下文简称 IoT 平台）提供了海量 API 接口给第三方应用开发者。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。

1.2 修订记录

文档版本	配套 IoT 平台版本	发布日期	修改说明
1.5.0	V100R001C52	2018/04	1. 公共变更： • 每个接口新增典型场景、接口变更的描述。 • 新增术语表。 2. 新增接口： • 设备管理：注册直连设备（密码方式），刷新设备秘钥（密码方式） • 订阅管理：查询单个订阅，批量查询订阅，删除单个订阅，批量删除订阅，订阅平台管理数据 • 消息推送：软件升级状态变更通知，软件升级结果变更通知，固件升级状态变更通知，固件升级结果变更通知 • 设备组管理：增加设备组，删除设备组，修改设备组信息，查询设备组详情，查询指定设备组，查询指定设备组成员，向设备组添加设备，从设备组删除设备 • 设备升级：查询版本包列表，查询指定版本包，删除指定版本包，创建软件升级任务，创建固件升级任务 • 操作任务管理：查询指定操作任务结果，查询指定操作任务子任务详情，查询操作任务列表 3. 修改接口： • 应用安全接入：鉴权，刷新 token • 设备管理：注册直连设备（验证码方式），刷新设备秘钥（验证码方式），修改设备信息，删除设备，查询设备激活状态 • 批量处理：创建批量任务，查询单个任务信息，查询任务详情信息 • 规则：创建规则，更新规则，删除规则，查找规则，批量修改规则状态 • 订阅管理：订阅平台业务数据 • 命令下发：设备服务调用，创建设备命令 • 数据采集：查询单个设备信息，批量查询设备信息列表，查询设备历史数据，查询设备服务能力

1.3 接口兼容性说明

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。

2 接口列表

2.1 应用安全接入
2.2 设备管理
2.3 批量处理
2.4 规则管理
2.5 订阅管理
2.6 消息推送
2.7 命令下发
2.8 数据采集
2.9 设备组管理
2.10 设备升级

2.1 应用安全接入

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

2.1.1 鉴权

典型场景

第三方应用首次访问 IoT 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 IoT 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 IoT 平台的开放 API。

接口功能

实现第三方应用在访问开放 API 之前的认证。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	新增错误码 100008
1.1.0	V100R001C30	响应参数增加 refreshToken
1.1.0	V100R001C10	新增接口

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:Bearer。app_key 为参数中的 appId，Authorization 的值即为调用 Auth 接口获取到的 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	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 IoT 平台的开放 API。appid 在 IoT 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 IoT 平台。secret 在 IoT 平台的 SP Portal 上创建应用时获得。

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 IoT 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 IoT 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。

请求示例

Method: POST

request:

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

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

Body:

{

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
401	100208	AppId or secret is not right.	appId 或 secret 错误。 处理建议： 1. 请检查 appId 和 secret 是否有误，是否新增字符或缺少字符。 2. 请检查请求路径中的 IP 地址是否有误。
500	100008	The haproxy address is null.	空的 haproxy 地址。 处理建议 ： IoT 平台内部错误，请联系 IoT 平台维护人员处理。

2.1.2 刷新 token

典型场景

accessToken 快过期时，第三方应用通过调用此接口，重新获取可用 accessToken.

接口功能

支持第三方应用在 accessToken 到期前，获取新的 accessToken。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	新增错误码 100008
1.1.0	V100R001C30	• 请求参数新增 appId, secret • 返回参数增加 scope, expiresIn • 返回参数删除 timeout
1.1.0	V100R001C10	新增接口

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	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 IoT 平台的开放 API。appid 在 IoT 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 IoT 平台。secret 在 IoT 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。

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)	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 IoT 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。

请求示例

Method: POST

request:

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

Content-Type:application/json

Body:

{

"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	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。
401	100006	Refresh access token failed.	accessToken 刷新失败。 处理建议 ： 1. 请检查 refreshToken 是否已过期，若过期则重新调用平台鉴权接口。 2. 请检查 refreshToken 是否有误，是否缺少或误增字符。 3. 请检查请求路径中的 IP 是否有误，确认该 refreshToken 是否属于当前 appId。
401	100208	AppId or secret is not right.	appId 或 secret 错误。 处理建议 ： 1. 请检查 appId 和 secret 是否有误，

Http 状态码	错误码	错误描述	说明
			是否新增字符或缺少字符。 2. 请检查请求路径中的 IP 地址是否有误。
500	100008	The haproxy address is null.	空的 haproxy 地址。 处理建议： IoT 平台内部错误，请联系 IoT 平台维护人员处理。

2.2 设备管理

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

2.2.1 注册直连设备（验证码方式）

典型场景

第三方应用调用此接口注册直连设备，并获取设备的验证码，在直连设备接入 IoT 平台时携带验证码，获取设备 ID 和密码。支持注册本应用和授权应用的设备。

接口功能

实现第三方应用注册直连设备，获取验设备证码。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	• 请求地址变更 • 整体参数优化 • 错误码 100412Http 状态码由 200 更改为 500 • 错误码 100416Http 状态码由 200 更改为 400 • 错误码 103026Http 状态码由 200 更改为 500 • 错误码 100203Http 状态码由 200 拆分为 403 与 500 • 错误码 100217Http 状态码由 200 更改为 403 • 删除错误码 100022，103027，103028
1.1.0	V100R001C30	• 新增请求参数 appId, psk, isSecure 字段 • 新增错误码 100002，100022 和 100217

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header：

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
verifyCode	可选	String(256)	body	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 IoT 平台自动生成。建议和 nodeId 设置成一样。
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。 说明

参数	必选/可选	类型	位置	描述
				若使用 IMEI 作为 nodeId，根据不同厂家的芯片有不同填写要求。 • 高通芯片设备的唯一标识为 urn:imei:xxxx，xxxx 为 IMEI 号 • 海思芯片设备的唯一标识为 IMEI 号 • 其他厂家芯片的设备唯一标识请联系模组厂家确认。
endUserId	可选	String(256)	body	终端用户 ID，如手机号码，email 地址。
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	body	设备验证码有效期，单位秒，默认值 180s。 • 值为 0 时，表示验证码则永不过期。 • 值为大于等于 0 的整数时，表示在指定时间内设备进行绑定，超过时间验证码无效 ，同时注册的设备会被删除。
deviceInfo	可选	DeviceInfoDTO	body	设备绑定信息，需要与设备 Profile 文件中的信息保持一致，具体参见 DeviceInfoDTO 结构体。
customFields	可选	List<CustomField>	body	自定义字段列表，用户可设置自定义字段，具体参见 CustomField 结构体。
imsi	可选	String(64)	body	NB-IoT 设备的 IMSI，用于消息计费。
isSecure	可选	Boolean	body	指定设备的安全状态，默认值为 false。 • true：安全 • false：非安全

DeviceInfoDTO 结构体定义：

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，与设备 Profile 文件中的信息保持一致。
manufacturerName	可选	String(256)	body	厂商名称，与设备 Profile 文件中的信息保持一致。
deviceType	可选	String(256)	body	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，例如：MultiSensor、ContactSensor、Camera、WaterMeter。
protocolType	可选	String(256)	body	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。
model	可选	String(256)	body	设备型号，由厂商定义，与设备 Profile 文件中的信息保持一致。

CustomField 结构体定义：

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。

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。
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 IoT 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

参数	类型	描述
		间内接入 IoT 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。

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": "**********",

"psk": "**********",

"timeout": 300,

"deviceInfo": {

"manufacturerId": "****",

"manufacturerName": "****",

"model": "****",

"protocolType": "CoAP",

"deviceType": "****"

}

"customFields": [

{

"fieldName": "****",

"fieldType": "****",

"fieldValue": "****"

}

]

}

响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

{

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

"verifyCode": "*******",

"psk": "*********",

"timeout": 360

}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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。
401	100025	AppId for auth not exist.	获取不到 appId 鉴权信息。 处理建议： 1. 请检查请求结构体的 Header 头域中是否给字段 app_key 赋值。 2. 若通过 HTTP 调用，请联系 IoT 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。
403	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	授权应用不存在。

Http 状态码	错误码	错误描述	说明
			处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。
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。
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。
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 IoT 平台维护人员处理。
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。
500	100001	Internal server error.	服务内部处理错误。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。
500	103026	The license is not exist.	License 不存在。 处理建议：IoT 平台内部 License 问题，请联系 IoT 平台维护人员处理。

2.2.2 注册直连设备（密码方式）

典型场景

第三方应用调用此接口在 IoT 平台注册设备，获取设备 ID 和密码，设备通过设备 ID 和密码接入 IoT 平台。

接口功能

支持第三方应用在 IoT 平台注册设备，获取设备 ID 和密码。支持注册本应用和授权应用的设备。

变更记录

接口版本	IoT 平台版本	变更说明
2.0.0	V100R001C52	• 请求地址变更 • 错误码 100426Http 状态码由 200 更改为 400 • 错误码 103026Http 状态码由 200 更改为 500 • 错误码 100412Http 状态码由 200 更改为 500 • 错误码 100203Http 状态码由 200 拆分为 403 和 500 • 新增错误码 100217 • 删除错误码 100022，100403，100418，103027，103028，100610 和 100611
2.0.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/reg/v2.0.0/deviceCredentials?appId={appId}
传输协议	HTTPS

请求参数

参数	必选/可选	类型	位置	描述

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IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
endUserId	可选	String(256)	body	终端用户 ID，如手机号码，email 地址。
region	可选	String(256)	body	设备所在的区域信息。
organization	可选	String(256)	body	设备所属的组织信息。
timezone	可选	String(256)	body	设备所在时区信息。
subsystem	可选	boolean	body	用户系统标识。用于区分这个设备是否是第三方子系统
deviceInfo	可选	DeviceInfo	body	设备的详细信息，具体参见 DeviceInfo 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。 **说明** 若使用 IMEI 作为 nodeId，根据不同厂家的芯片有不同填写要求。 ● 高通芯片设备的唯一标识为 urn:imei:xxxx，xxxx 为 IMEI 号 ● 海思芯片设备的唯一标识为 IMEI 号 ● 其他厂家芯片的设备唯一标识请联系模组厂家确认。
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，与设备 Profile 文件中的信息保持一致。

参数	必选/可选	类型	位置	描述
manufacturerName	可选	String(256)	body	厂商名称，与设备 Profile 文件中的信息保持一致。
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，与设备 Profile 文件中的信息保持一致，如 MultiSensor、ContactSensor、CameraGateway。
model	可选	String(256)	body	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	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，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	可选	String(256)	body	Bridge 标识，表示设备通过哪个 Bridge 接入 IoT 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 status 和 statusDetail。
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

参数	必选/可选	类型	位置	描述
supportedSecurity	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
isSecurity	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
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 IoT 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

StatusCode: 200 OK

字段	类型	描述
deviceId	String(256)	设备 ID，用于唯一标识一个设备。
secret	String	设备密码，小于 256 个字节。
psk	String(32)	随机 psk 参数，由 IoT 平台生成。

请求示例

Method: POST

request:

https://server:port/iocm/app/reg/v2.0.0/deviceCredentials?appId=******

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

Body:

{

"deviceInfo": {

"nodeId": "********"

},

"endUserId": "************"

}

响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

{

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

"secret": "**********",

"psk": "**********"

}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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	100203	The application is not existed.	授权应用不存在。

Http 状态码	错误码	错误描述	说明
			处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。
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。
500	100001	Internal server error.	服务内部处理错误。 处理建议：IoT 平台内部错误，请联系 IoT 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。
500	103026	The license is not exist.	License 不存在。 处理建议：IoT 平台内部 License 问题，请联系 IoT 平台维护人员处理。
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。

2.2.3 刷新设备密钥（验证码方式）

典型场景

第三方应用需要重新绑定原设备或者新设备的时候，可以调用此接口刷新 nodeId 和验证码。

接口功能

第三方应用刷新设备 nodeId 和验证码，支持刷新注册到本应用的设备和授予权限的其它应用的设备。

1. 支持重新绑定原设备：deviceId 不变，设备重新绑定。
2. 支持重新绑定新设备：deviceId 不变，device 表中对应的 nodeId 值发生变化。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	• 请求地址变更 • 错误码 100203Http 状态码由 200 拆分为 403 与 500 • 错误码 100403Http 状态码由 200 更改为 404 • 错误码 100610Http 状态码由 200 更改为 400 • 错误码 100610Http 状态码由 200 更改为 400 • 错误码 100611Http 状态码由 200 更改为 400 • 错误码 100418Http 状态码由 200 更改为 404 • 错误码 100426Http 状态码由 200 更改为 400 • 删除错误码 100022
1.1.0	V100R001C30	• 新增请求参数 appId • 新增错误码新增 100002，100022 和 100217
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header：

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
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 IoT 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 IoT 平台自动生成。建议和 nodeId 设置成一样。
nodeId	可选	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。 • 值空时，nodeId 不变。 • 值不为空时，更新 nodeId。 **说明** 若使用 IMEI 作为 nodeId，根据不同厂家的芯片有不同填写要求。 • 高通芯片设备的唯一标识为 urn:imei:xxxx，xxxx 为 IMEI 号 • 海思芯片设备的唯一标识为 IMEI 号 • 其他厂家芯片的设备唯一标识请联系模组厂家确认。
timeout	可选	Integer	body	验证码超时时间，单位秒，取值大于等于 0 的整数。 • 值为空时，使用默认值（180s）。 • 值为 0 时，永不过期。 • 值为非 0，指定时间。

响应参数

Status Code: 200 OK

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 IoT 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 IoT 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
401	100002	Invalid access token.	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。
401	100025	AppId for auth not exist.	获取不到 appId 鉴权信息。 处理建议： 1. 请检查请求结构体的 Header 头域中是否给字段 app_key 赋值。 2. 若通过 HTTP 调用，请联系 IoT 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。
500	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
403	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
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	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。
404	100403	The device is not existed.	设备不存在。 处理建议：IoT 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
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 IoT 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 IoT 平台的连接。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： 1. 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。 2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。 3. 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。
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	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 IoT 平台维护人员处理。

2.2.4 刷新设备密钥（密码方式）

典型场景

第三方应用需要重新绑定原设备或者新设备的时候，可以调用此接口刷新设备的接入密码。

接口功能

第三方应用刷新设备接入密码，支持刷新注册到本应用的设备和授予权限的其它应用的设备。

1. 支持重新绑定原设备：deviceId 不变，设备重新绑定。

2. 支持重新绑定新设备：deviceId 不变，device 表中对应的 nodeId 值发生变化。

变更记录

接口版本	IoT 平台版本	变更说明
2.0.0	V100R001C52	• 请求地址变更 • 错误码 100403Http 状态码由 200 更改为 404 • 错误码 100610Http 状态码由 200 更改为 400 • 错误码 100611Http 状态码由 200 更改为 400 • 错误码 100418Http 状态码由 200 更改为 404 • 错误码 100426Http 状态码由 200 更改为 400 • 错误码 100203Http 状态码由 200 拆分为 403 和 500 • 错误码 100217Http 状态码由 200 更改为 403 • 删除错误码 100003，100007，100022 和 100025
2.0.0	V100R001C30	新增接口

接口原型

请求方法	PUT
请求地址	https://server:port/iocm/app/reg/v2.0.0/deviceCredentials/{deviceId}?appId={appId}
传输协议	HTTPS

注意事项

携带头域信息

Header：

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

请求参数

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

参数	必选/可选	类型	位置	描述
				应用。开发者可通过该标识来指定哪个应用来调用 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
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 IoT 平台分配获得。
nodeId	可选	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。 说明 若使用 IMEI 作为 nodeId，根据不同厂家的芯片有不同填写要求。 ● 高通芯片设备的唯一标识为 urn:imei:xxxx，xxxx 为 IMEI 号 ● 海思芯片设备的唯一标识为 IMEI 号 ● 其他厂家芯片的设备唯一标识请联系模组厂家确认。
endUserId	可选	String(256)	body	终端用户 ID，如手机号码，email 地址。

响应参数

Status Code: 200 OK

字段	类型	描述
secret	String	设备密码，小于 256 个字节。
psk	String(32)	随机 psk 参数，由 IoT 平台生成。

请求示例

Method:PUT

request:

https://server:port/iocm/app/reg/v2.0.0/deviceCredentials/fdsafdsasafdsaf

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

Body:

{

"endUserId":"**********",

"nodeId":"**********"

}

响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

{

"secret":"**********",

"psk":"**********"

}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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

Http 状态码	错误码	错误描述	说明
			为空。
404	100403	The device is not existed.	设备不存在。 处理建议：IoT 平台未找到 deviceId 对应的设备，请检查 deviceId 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 IoT 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 IoT 平台的连接。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： 1. 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。 2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。 3. 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。
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。
500	100001	Internal server error.	服务内部处理错误。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。

2.2.5 修改设备信息

典型场景

第三方应用调用此接口修改设备的基本信息，如设备类型、厂商信息、位置信息等。

接口功能

支持第三方应用修改设备的基本信息，支持修改本应用的设备和授予权限的其它应用的设备信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C52	• 请求地址变更 • 增加请求参数 imsi • 新增错误码 100217 • 错误码 100203Http 状态码由 200 拆分为 403 和 500 • 错误码 100403Http 状态码由 200 更改为 404 • 错误码 100418Http 状态码由 200 更改为 404
1.2.0	V100R001C30	整体参数优化
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	PUT
请求地址	https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?appId={appId}
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	path	设备 ID，用于唯一标识一个设备，在

参数	必选/可选	类型	位置	描述
				注册设备时由 IoT 平台分配获得。
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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
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。 • TRUE：冻结状态 • FALSE：非冻结状态
manufacturerId	可选	String(1~256)	body	厂商 ID，唯一标识一个厂商。
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、Camera 和 WaterMeter。 在 NB-IoT 方案中，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。
model	可选	String(1~256)	body	设备型号，由厂商定义。
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。
deviceConfig	可选	DeviceConfigDTO	body	设备配置信息，具体参见 DeviceConfigDTO 结构体。
region	可选	String(1~256)	body	设备区域信息。

参数	必选/可选	类型	位置	描述
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，如北京时区对应的时区编码为Asia/Beijing。
imsi	可选	String(1-64)	body	NB-IoT 终端的 IMSI。

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 DataConfigDTO 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。

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
401	100002	Invalid access token.	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。
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.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。
404	100403	The device is not existed.	设备不存在。 处理建议：IoT 平台未找到 deviceId 对应的设备，请检查 deviceId 是否有误。
404	100418	The deviceData is	设备信息不存在。

Http 状态码	错误码	错误描述	说明
		not existed.	处理建议： 1. 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。 2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。 3. 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。

2.2.6 删除直连设备

典型场景

第三方应用调用此接口删除直连设备和其下的非直连设备。

接口功能

支持第三方应用删除直连设备和其下的非直连设备，支持删除本应用的设备和授予权限的其它应用的设备。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C52	• 请求地址变更 • 错误码 100203 的 Http 状态码由 200 拆分为 403 和 500 • 错误码 100217 的 Http 状态码由 200 更改为 403 • 错误码 100403 的 Http 状态码由 200 更改为 404 • 错误码 100418 的 Http 状态码由 200 更改为 404 • 删除错误码 100023，100428，100431，100432 和 102203
1.1.0	V100R001C30	• 删除错误码 101001 • 新增错误码 100002，100217 和 100431
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

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
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 IoT 平台分配获得。
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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。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。 • true，级联删除，即删除直连设备和其下的非直连设备。 • 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。

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
401	100002	Invalid access token.	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	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。
404	100403	The device is not existed.	设备不存在。 处理建议：IoT 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： 1. 请求参数中携带的 deviceId 错误。请

Http 状态码	错误码	错误描述	说明
			确认该 deviceId 是否属于该 appId 或者 deviceId 是否写错。 2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。 3. 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。

2.2.7 查询设备激活状态

典型场景

第三方应用调用此接口查询指定设备是否已接入 IoT 平台并激活。

接口功能

支持第三方应用通过设备 ID 查询设备激活状态，支持查询本应用的设备和授予权限的其它应用的设备。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	• 请求地址变更 • 错误码 100203Http 状态码由 200 拆分为 403 与 500 • 错误码 100217Http 状态码由 200 更改为 403 • 错误码 100403Http 状态码由 200 更改为 404
1.1.0	V100R001C30	• 新增请求参数 appId • 新增返回参数 deviceId • 新增错误码 100002 和 100217
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
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 IoT 平台分配获得。

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。
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。

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
401	100002	Invalid access token.	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	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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

Http 状态码	错误码	错误描述	说明
			appId 为空。
404	100403	The device is not existed.	设备不存在。 处理建议：IoT 平台未找到 deviceId 对应的设备，请检查 deviceId 是否有误。

2.3 批量处理

第三方应用通过批量处理接口对已接入 IoT 平台的设备进行批量操作。

2.3.1 创建批量任务

典型场景

第三方应用可以调用此接口使用 IoT 平台的批量创建设备，给批量设备下发缓存命令功能.

接口功能

支持第三方应用对设备创建批量任务。支持给本应用和授予权限的其它应用创建批量任务。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	• 增加请求参数 tags • taskType 参数增加取值 DeviceCmd（批量下发命令任务）， • param 结构提增加参数 callbackUrl 和 maxRetransmit • 新增错误码 100203，101016 和 101017
1.1.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/batchtask/v1.1.0/tasks
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(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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。
timeout	必选	Integer	body	任务超时时长，单位秒，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，DeviceReg/DeviceCmd。
param	必选	ObjectNode	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 param 参数。
tags	可选	List<TagDTO2>	body	标签列表，请参见 TagDTO2 结构体说明。

param 参数结构：

- taskType 参数取值 DeviceReg（批量注册设备任务），param 参数：

参数	必选/可选	类型	位置	描述
fileId	必选	String	body	文件上传时返回的文件 ID。

- taskType 参数取值 DeviceCmd（批量下发命令任务），param 参数：

参数	必选/可选	类型	位置	描述

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。
deviceList	可选	List<String>	body	deviceId 列表，当命令类型为 DeviceList 时，需要填写。
deviceType	可选	String	body	设备类型，当命令类型为 DeviceType 时需要填写，其值应当与 profile 中定义的一致。
manufacturerId	可选	String	body	厂商 ID，当命令类型为 DeviceType 时可填写，其值应当与 profile 中定义的一致。
model	可选	String	body	设备型号，当命令类型为 DeviceType 时可填写，其值应当与 profile 中定义的一致。
deviceLocation	可选	String	body	设备位置，当命令类型为 DeviceArea 时需要填写。
groupList	可选	List<String>	body	设备组名字列表，当命令类型为 GroupList 时需要填写。
command	必选	CommandDTOV2	body	命令，查看 CommandDTOV2 结构体说明结构定义
callbackUrl	可选	String	body	命令执行结果的回调地址。
maxRetransmit	可选	Integer(0~3)	body	命令下发最大重传次数，取值范围：0-3。

CommandDTOV2 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，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，json 体中的键（key）为服务 profile 里定义命令的参数名。

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。

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" : "DeviceReg",
"param" : {
"fileId" : "******"
}
}
```

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
401	100002	Invalid access token.	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。
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	100007	Bad request message.	错误的请求消息。 处理建议：请参考请求参数说明检查请求数据是否有误。
200	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	101016	Get iotws address failed.	获取 iotws 地址失败 处理建议：IoT 平台内部错误，请联系 IoT 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：IoT 平台内部错误，请联系 IoT 平台维护人员处理。

2.3.2 查询单个任务信息

典型场景

第三方应用可以调用此接口查询某个批量任务的信息，如任务的当前状态和进度。

接口功能

支持第三方应用查询单个批量任务的信息，支持查询本应用和授予权限的其它应用创建的批量任务信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	• 增加请求参数 select • 增加响应参数 tags • 错误码 100203Http 状态码由 200 更改为 400 • 删除错误码 100217 • 新增错误码 100002
1.1.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

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
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 IoT 平台的开放 API。appid 在 IoT 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	类型	位置	描述
select	可选	String	query	指定可选的返回值，可取值：tag。
taskId	必选	String	query	批量任务 ID，创建批量任务后获得。

响应参数

Status Code: 200 OK

参数	类型	描述
taskId	String	批量任务 ID。
taskName	String	批量任务名称。
appId	String	批量任务所归属的 appId。
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：DeviceReg/DeviceCmd/DeviceLocation。
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	Int	批量任务的超时时间，单位秒。
progress	Int	批量任务的进度，单位：千分比，范围：0-1000，向下取整。
totalCnt	Int	任务的详情总数。
successCnt	Int	成功的任务详情数。
failCnt	Int	失败的任务详情数
timeoutCnt	Int	超时的任务详情数。
expiredCnt	Int	未执行的失效任务详情数。
completeCnt	Int	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。

参数	类型	描述
successRate	Int	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，具体参见 param 参数结构。
tags	List<TagDTO2>	批量任务的标签列表，具体参见 TagDTO2 结构体。

param 参数结构：

- 批量注册设备任务，param 结构：

参数	类型	描述
fileId	String	批量注册设备时上传的批量注册文件 ID。

- 批量下发命令任务，param 结构：

参数	类型	描述
commandType	String	批量命令的类型：DeviceList / DeviceType / DeviceArea / GroupList / Broadcast。
deviceType	String	设备的类型。
manufacturerId	String	厂商 ID。
model	String	设备型号。
serviceId	String	设备中的服务标识。
method	String	命令名称。
maxRetransmit	Integer(0~3)	命令下发的最大重传次数，范围：0-3。

- 批量修改设备位置任务，param 结构：

参数	类型	描述
fileId	String	批量修改设备位置时上传的文件 ID。

TagDTO2 结构体说明：

参数	类型	描述
tagName	String(1-	标签名称。

参数	类型	描述
	128)	
tagValue	String(1-1024)	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
404	105005	The batchTask is not existed.	任务不存在。 处理建议：请检查接口请求中的 taskId 是否有误。
401	100002	Invalid access token.	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。
400	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。

2.3.3 查询任务详情信息

典型场景

第三方应用可以调用此接口查询批量任务中某个任务的详情信息，如任务详情的状态和结果。

接口功能

支持第三方应用查询批量任务中某个任务的详情信息，支持查询本应用和授予权限的其它应用创建的批量任务信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	• 响应参数 output 增加批量修改设备位置值 • 错误码 100203Http 状态码由 200 拆分为 403 与 500 • 错误码 100217Http 状态码由 200 更改为 403 • 新增错误码 100002
1.1.0	V100R001C30	新增接口

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}&deviceId={deviceId}&commandId={commandId}&pageNo={pageNo}&pageSize={pageSize}
传输协议	HTTP/HTTPS

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

请求参数

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 IoT 平台的开放 API。appid 在 IoT 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。

参数	必选/可选	类型	位置	描述
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。
commandId	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。

响应参数

Status Code: 200 OK

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 QueryTaskDetailDTOCloud2NA 结构体。

QueryTaskDetailDTOCloud2NA 结构体说明：

参数	类型	描述

参数	类型	描述
status	String	任务详情状态，取值范围：Pending/WaitResult/Success/Fail/Timeout。
output	String	批量注册设备、批量下发命令或批量修改设备位置的输出信息。
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，具体参见 param 参数结构。

param 参数结构说明：

- 批量注册设备任务，param 为：

参数	类型	描述
index	String	批量任务文件里第几行的任务。
nodeId	String	设备的唯一标识。

- 批量下发命令任务，param 为：

参数	类型	描述
deviceId	String	设备 Id。
commandId	String	下发的命令 Id。

- 批量修改设备位置任务，param 为：

参数	类型	描述
deviceId	String	设备 Id。
index	String	批量任务文件里第几行的任务。

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": 2,
"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
404	105005	The batchTask is not existed.	任务不存在。 处理建议：请检查接口请求中的 taskId 是否有误。
401	100002	Invalid access token.	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.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
403	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。

2.4 规则管理

第三方应用根据具体的业务需要，制定规则的条件和动作，当设备的状态满足设定条件后，触发相应的操作。

2.4.1 创建规则

典型场景

第三方应用可以调用此接口创建规则，当满足规则设定的触发条件后，IoT 平台会触发相应的操作。

接口功能

支持第三方应用创建规则，使用 IoT 平台的规则引擎能力。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C52	整体参数优化
1.1.0	V100R001C30	整体参数优化
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/rule/v1.2.0/rules?appId={appId}
传输协议	HTTPS

请求参数

RuleDTO1.2 结构体说明：

参数	必选/可选	类型	位置	描述
ruleId	可选	String(128)	body	规则实例的 ID，仅在规则更新时有效，创建规则时不填写。
appKey	可选	String	body	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 IoT 平台的开放 API。 appid 在 IoT 平台的 SP Portal 上创建应用时获得。

参数	必选/可选	类型	位置	描述
name	必选	String	body	规则名称。
description	可选	String(512)	body	规则描述。
author	必选	String	body	创建此规则的用户的 ID。
conditions	可选	List<Json>	body	条件列表，与 groupExpress 二选一必填，具体参见 Condition-DeviceData 结构体，Condition-DeviceGroupData 结构，Condition-DeviceTypeData 结构体，Condition-DailyTimer 结构体，Condition-CycleTimer 结构体和 Condition- NoDetected 结构体。
logic	可选	String	body	多条件之间的逻辑关系，支持 and 和 or，默认为 and。
timeRange	可选	TimeRange	body	条件场景的时间段，具体参见 TimeRange 结构体。
actions	定时场景必选	List<Json>	body	满足规则所执行的动作，具体参见 Action-DeviceCMD 结构体，Action-SMS 结构体，Action-email 结构体，Action-Delay 结构体，Action-Rule 结构体，Action-DeviceAlarm 结构体和 Action-EI_CMD 结构体。
matchNow	可选	String	body	表示是否立即触发，即是否立即进行规则条件判断，条件符合的话执行动作。 • yes：表示立即触发 • no：表示不触发
status	可选	String	body	规则的状态，默认为“active”状态。 • active 代表激活状态 • inactive 代表未激活
groupExpress	可选	GroupExpress	body	复杂多条件表达式，具体参见 GroupExpress 结构体，与 conidtions 二选一必填。
timezoneID	可选	String	body	时区 ID。 • 若为空，使用 UTC 时间 • 若不为空，使用本地时间
triggerSources	可选	List<Json>	body	触发源列表，只与 GroupExpress 联用构造复杂多条件规则，目前有 DEVICE 类型和 TIMER 类型，具体参见 TriggerSource-Device 结构体和 TriggerSource-Timer 结构

参数	必选/可选	类型	位置	描述
				体。
executor	可选	String	body	规则执行主体，当 action 和 condition 里面的设备在同一个网关下面时，取值网关 ID（代表网关执行）；否则取定值“cloud”（代表云端执行）。
transData	可选	ObjectNode	body	平台不需要识别的数据，只做保存。
refreshId	可选	Boolean	body	创建规则时是否检查 ruleId 不能为空标识，默认为 true。
checkNullAction	可选	Boolean	body	创建规则时是否检查 Action 不能为空标识，默认为 true。
priority	可选	String	body	规则优先级，保留字段。
tags	可选	List<TagDTO2>	body	标签列表，具体参见 TagDTO2 结构体。
rulePreProcessors	可选	List<RulePreProcessorDTO>	body	规则的预处理器，不符合预处理条件的规则不执行，具体参见 RulePreProcessorDTO 结构体。

TimeRange 结构体说明：

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

GroupExpress 结构体说明：

参数	必选/可选	类型	位置	描述
operator	可选	Enum	body	操作符 and/or，默认为 and。
groupEl	必选	List<G	body	最外层操作符对应的元素，具体参见

参数	必选/可选	类型	位置	描述
ements		roupElement>		GroupElement 结构体。

GroupElement 结构体说明：

参数	必选/可选	类型	位置	描述
operator	可选	Enum	body	操作符 and/or，默认为 and。
elements	必选	List<Json>	body	内层表达式运算的元素,一个 element 表示一个 condition,自定义结构 conditions。

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。
tagValue	必选	String(1-1024)	body	标签值。

RulePreProcessorDTO 结构体说明：

参数	必选/可选	类型	位置	描述
processor	必选	String	body	规则预处理类型，取值为：“rulePreProcessorForDeviceTag”。
param	必选	ObjectNode	body	根据预处理类型有对应的结构。

当预处理类型为“rulePreProcessorForDeviceTag”时，ObjectNode 结构体为：

参数	必选/可选	类型	位置	描述
tag	可选	TagDTO3	body	标签结构。
nullTags	可选	List<Str	body	标签列表。

参数	必选/可选	类型	位置	描述
		ing>		

TagDTO3 的结构体说明：

参数	必选/可选	类型	位置	描述
tagName	必选	String	body	标签名。
tagValue	必选	String	body	标签值。
matchType	必选	String	body	tagValue 的匹配方式，取值为 Any|Right。

Condition-DeviceData 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	条件触发 condition 类型，取值为“DEVICE_DATA”。
id	可选	String	body	对 condition 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
deviceInfo	必选	DeviceInfo	body	设备数据的定位信息，根据定位信息读取的数据作为比较表达式的左值，具体参见 DeviceInfo 结构体。
operator	必选	String	body	数据比较的操作符（目前支持的操作符有：>，<，>=，<=，=，between，geo.circle.in， geo.circle.out，具体使用见举例说明）。
value	可选	String	body	比较表达式的右值（与 between 操作符联用时，右值表示最小值和最大值用逗号隔开如“20,30”表示大于等于 20 小于 30，应用场景:温度在 20 到 30 之间触发规则）。
transInfo	可选	Json	body	不需要平台理解的信息。
duration	可选	Integer	body	规则执行延时时间，单位为分钟，默认为 0 不延时。
strategy	必选	Strategy	body	配置条件处理策略，具体参见 Strategy 结构体。

DeviceInfo 结构体说明：

参数	必选/可选	类型	位置	描述
deviceId	必选	String	body	设备 ID。
path	必选	String	body	以“/”分割的信息路径，按照当前的数据模型，路径为 ServiceId/DataProperty，例如门磁状态为“DoorWindow/status”。

Strategy 结构体说明：

参数	必选/可选	类型	位置	描述
trigger	可选	String	body	触发模式，默认为 pulse。 • pulse：上报的数据满足条件则触发，不判断上一次的数据。 • reverse：上一次数据不满足条件，本次数据满足条件则触发 默认为 pulse。
eValidTime	可选	Integer (>=-1)	body	事件时效性，单位为秒，表示收到的传感器数据多久有效，以 eventTime 为基准。

Condition-DeviceGroupData 结构说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	条件触发 condition 类型，取值为“DEVICE_DATA_GROUP”。
id	可选	String	body	对 condition 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
deviceGroupInfo	必选	DeviceGroupInfo	body	组设备数据的定位信息，根据定位信息读取的数据作为比较表达式的左值，具体参见 DeviceGroupInfo 结构体。
operator	必选	String	body	数据比较的操作符数据比较的操作符（目前支持的操作符有：>，<，>=，<=，=，between，geo.circle.in，geo.circle.out，具体使用见举例说明）。
value	可选	String	body	比较表达式的右值（与 between 操作符联用时，右值表示最小值和最大值用逗号隔

参数	必选/可选	类型	位置	描述
				开如“20,30”表示大于等于 20 小于 30，应用场景:温度在 20 到 30 之间触发规则）。
transInfo	可选	Json	body	不需要平台理解的信息。
duration	可选	Integer	body	规则执行延时时间，单位为分钟，默认为 0 不延时。
strategy	必选	Strategy	body	配置条件处理策略，具体参见 Strategy 结构体。

DeviceGroupInfo 结构体说明：

参数	必选/可选	类型	位置	描述
groupId	必选	String	body	设备组 ID。
path	必选	String	body	以“/”分割的信息路径，按照当前的数据模型，路径为 ServiceId/DataProperty，例如门磁状态为“DoorWindow/status”，属性支持多个，比如 OBD 需要两个属性：经度和纬度，多个属性中间以逗号分隔。

Condition-DeviceTypeData 结构说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	条件触发 condition 类型，取值为“DEVICE_DATA_TYPE”。
id	可选	String	body	对 condition 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
deviceTypeInfo	必选	DeviceTypeInfo	body	组设备数据的定位信息，根据定位信息读取的数据作为比较表达式的左值，具体参见 DeviceTypeInfo 结构体。
operator	必选	String	body	数据比较的操作符数据比较的操作符（目前支持的操作符有：>，<，>=，<=，=，between，geo.circle.in，geo.circle.out，具体使用见举例说明）。
value	可选	String	body	比较表达式的右值（与 between 操作符联用时，右值表示最小值和最大值用逗号隔

参数	必选/可选	类型	位置	描述
				开如“20,30”表示大于等于 20 小于 30，应用场景:温度在 20 到 30 之间触发规则）。
transInfo	可选	Json	body	不需要平台理解的信息。
duration	可选	Integer	body	规则执行延时时间，单位为分钟，默认为 0 不延时。
strategy	必选	Strategy	body	配置条件处理策略，具体参见 Strategy 结构体。

DeviceTypeInfo 结构体说明：

参数	必选/可选	类型	位置	描述
manufacturerId	必选	String	body	厂商 ID。
model	必选	String	body	设备型号。
deviceType	必选	String	body	设备类型。
protocolType	必选	String	body	协议类型。
path	必选	String	body	以“/”分割的信息路径，按照当前的数据模型，路径为 ServiceId/DataProperty，例如门磁状态为“DoorWindow/status”，属性支持多个,比如 OBD 需要两个属性：经度和纬度，多个属性中间以逗号分隔。

Condition-DailyTimer 结构体说明：

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

参数	必选/可选	类型	位置	描述
Week				

Condition-CycleTimer 结构体说明：

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

Condition- NoDetected 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	不检测类型，说明这个条件在平台不进行检测，只存储条件的部分信息，用于终端手机 app 直接触发，比如检测到回家后直接触发 action。取值为“NO_DETECTED”。
id	可选	String	body	对 condition 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
description	可选	String	body	条件描述。

Action-DeviceCMD 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	设备命令 action 类型，取值为“DEVICE_CMD”。
id	可选	String	body	对 action 的编号，在一条规则范围内唯一，不指定时由系统自动编号。

参数	必选/可选	类型	位置	描述
appKey	可选	String(64)	body	应用实例的 key 值。
deviceId	可选	String(64)	body	设备 ID，对于不同的 condition 类型，该字段有不同约束： DEVICE_DATA/DEVICE_DATA_GROUP/DEVICE_DATA_TYPE，允许为空，命令下发给上报数据的设备，如果希望命令下发给其他设备，请务必携带； DAILY_TIMER/CYCLE_TIMER，不允许为空。
cmd	必选	CMD	body	命令内容，具体参见 CMD 结构体。
cmdVersion	可选	String	body	下发命令的版本。
cmdMetaData	可选	ObjectNode	body	下发命令的参数。
transInfo	可选	Json	body	不需要平台理解的信息。
deviceCommandActionPreProcessors	可选	List<DeviceCommandActionPreProcessor>	body	设备命令下发预处理器列表，符合预处理器条件的才允许给设备下发命令，具体参见 DeviceCommandActionPreProcessor 结构体。

CMD 结构体说明：

参数	必选/可选	类型	位置	描述
messageType	必选	String(128)	body	命令名字。
messageBody	可选	Json	body	下发命令的消息内容。
serviceId	必选	String(64)	body	设备服务的 ID。

当 cmdVersion="V1.3.0"时，OjbectNode(cmdMetaData)结构体说明：

参数	必选/可选	类型	位置	描述

参数	必选/可选	类型	位置	描述
requestId	可选	String(64)	body	命令请求 ID。
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	body	已废弃参数，未使用。

DeviceCommandActionPreProcessor 结构体说明：

参数	必选/可选	类型	位置	描述
processor	必选	String	body	设备命令下发预处理类型，取值为：“deviceCommandActionPreProcessorForDeviceTag”。
param	必选	ObjectNode	body	根据预处理类型有对应的结构。

当预处理类型为“deviceCommandActionPreProcessorForDeviceTag”时，ObjectNode 结构体为：

参数	必选/可选	类型	位置	描述
tag	可选	TagDTO3	body	标签结构。
nullTags	可选	List<String>	body	标签列表。

TagDTO3 结构体说明：

参数	必选/可选	类型	位置	描述
tagName	必选	String	body	标签名。
tagValue	必选	String	body	标签值。
matchType	必选	String	body	tagValue 的匹配方式，取值为 Any|Right。

Action-SMS 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	发送短信 action 类型，取值为“SMS”。
id	可选	String	body	对 action 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
msisdn	必选	String	body	短信通知地址。
content	可选	String	body	短信的内容（短信 content 为空时，短信邮件由北向应用发送）。
subject	可选	String	body	短信的主题
title	可选	String	body	短信的别名。
accountId	可选	String	body	账户 ID。

Action-email 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	发送邮件 action 类型，取值为“EMAIL”。
id	可选	String	body	对 action 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
email	必选	String	body	邮箱地址。
content	可选	String	body	邮件的内容（邮件 content 或者 subject 为空时，短信邮件由北向应用发送）。
subject	可选	String	body	邮件的主题。
title	可选	String	body	邮件的别名。
accountId	可选	String	body	账户 ID。

Action-Delay 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	延时命令 action 类型，取值为

参数	必选/可选	类型	位置	描述
				“DELAY”。
id	可选	String	body	对 action 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
time	必选	Long	body	延迟时间，单位时间秒。
transInfo	可选	JsonNode	body	透传数据，平台不需识别，只做保存。

Action-Rule 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	触发规则执行 action 类型，取值为“RULE”,用于可以执行终端的手工场景。
id	可选	String	body	对 action 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
ruleId	必选	String	body	动作执行规则的规则 ID。
cmd	必选	String	body	对规则执行的动作。 • EXECUTE：执行动作 • ACTIVE：激活规则 • DEACTIVE：去激活规则

Action-DeviceAlarm 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	设备告警 action 类型，取值为“DEVICE_ALARM”。
id	可选	String	body	对 action 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
name	必选	String	body	告警名。
status	必选	String	body	告警状态 fault/recovery。
severity	必选	String	body	告警级别 warning/minor/major/critical。
descriptio	可选	String	body	告警描述。

参数	必选/可选	类型	位置	描述
n				

Action-EI_CMD 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	设备 EIG 命令 action 类型，取值为“EI_CMD”。
id	可选	String	body	对 action 的编号，在一条规则范围内唯一，不指定时由系统自动编号。
subSystem	必选	String	body	要发送的子系统名称，取值为：IVS/SNS/ICP 其中之一。
deviceId	可选	String	body	触发源设备 DI。
messageType	必选	String	body	消息类型，需要与 EIG 通知对端子系统 URL 对齐。
messageBody	必选	Json	body	需要推送的消息体。

TriggerSource-Device 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	Enum	body	设备触发源类型，取值为 DEVICE。
deviceId	必选	String	body	设备 ID。
serviceId	必选	String	body	服务 ID。

TriggerSource-Timer 结构体说明：

参数	必选/可选	类型	位置	描述
type	必选	Enum	body	设备触发源类型，取值为 TIMER。

响应参数

状态码 StatusCode: 200 OK

内容类型 Content-Type: application/json 返回结构：

参数	类型	描述
ruleId	String	规则实例 ID。

请求示例

Method: POST

Request:

https://server:port/iocm/app/rule/v1.2.0/rules

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

Body:

{

"name": "rule1",

"author": "wudashan",

"description": "this is a test rule 2015.12.23",

"status": "active",

"logic": "and",

"conditions": [

{

"type": "DEVICE_DATA",

"deviceInfo": {

"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",

"path": "DoorWindow/status"

},

"operator": "=",

"value": "OPEN"，

"strategy" :{

"trigger": "pulse",

"eValidTime": "-1"
}
}
],
"actions": [
{
"type": "DEVICE_CMD",
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"cmd": {
"serviceId": "DoorWindow",
"messageType": "MUTE",
"messageBody": {
"force": "true"
}
}
}
]
}

异步命令 1.3 版本创建规则 Demo

Method: POST

Request:

https://server:port/iocm/app/rule/v1.2.0/rules

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

Body:

{
"name": "rule1",
"author": "wudashan",
"description": "this is a test rule 2015.12.23",
"status": "active",
"logic": "and",

"conditions": [
{
"type": "DEVICE_DATA",
"deviceInfo": {
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"path": "DoorWindow/status"
},
"operator": "=",
"value": "OPEN",
"strategy" :{
"trigger": "pulse",
"eValidTime": "-1"
}
}
],
"actions": [
{
"type": "DEVICE_CMD",
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"cmd": {
"serviceId": "DoorWindow",
"messageType": "MUTE",
"messageBody": {
"force": "true"
}
},
"cmdVersion": "V1.3.0"
}
]
}

异步命令 1.3 版本创建 between 表达式规则 Demo

Method: POST

Request:

https://server:port/iocm/app/rule/v1.2.0/rules

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

Body:


```
{
"name": "rule1",
"author": "wudashan",
"description": "this is a test rule 2015.12.23",
"status": "active",
"logic": "and",
"conditions": [
{
"type": "DEVICE_DATA",
"deviceInfo": {
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"path": "Temperature/temperature"
},
"operator": "between",
"value": "20,30",
"strategy":{
"trigger":"pulse",
"eValidTime":"-1"
}
}
],
"actions": [
{
"type": "DEVICE_CMD",
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"cmd": {
"serviceId": "DoorWindow",
```

"messageType": "MUTE",

"messageBody": {

"force": "true"

}

},

"cmdVersion": "V1.3.0"

}

]

}

异步命令 1.3 版本创建 geo.circle.in/out 表达式规则 Demo

Method: POST

Request:

https://server:port/iocm/app/rule/v1.2.0/rules

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

Body:

{

"name": "rule1",

"author": "wudashan",

"description": "this is a test rule 2015.12.23",

"status": "active",

"logic": "and",

"conditions": [

{

"type": "DEVICE_DATA",

"deviceInfo": {

"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",

"path": "Location/longitude,latitude"

},

"operator": "geo.circle.in",

"value": "114074792,22641851,100",

```
"strategy":{
"trigger":"pulse",
"eValidTime":"-1"
}
}
],
"actions": [
{
"type": "DEVICE_CMD",
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"cmd": {
"serviceId": "DoorWindow",
"messageType": "MUTE",
"messageBody": {
"force": "true"
}
},
"cmdVersion": "V1.3.0"
}
]
}
```

响应示例

```
response:
Status Code: 200 OK
Content-Type: application/json
response:
Body:
{
"ruleId": “*******”
}
```

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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。
401	101409	Authentication failure.	鉴权失败。 处理建议：请检查接口请求中是否携带 appid。
200	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	101403	The rule is not existed.	规则不存在。 处理建议：在创建规则时，请确认请求参数中 ruleId 为空。
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	101402	The rule name already existed.	规则名称已存在。 处理建议：请修改接口请求中的规则名称。
500	100023	The data in dataBase is abnormal.	数据库异常。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。
200	101404	The rule count of each app has reached the upper limit, the current up limit is 30 entries.	应用下的规则数量超出上限，当前上限为 30。 处理建议：如果需要继续添加规则，请删除其他无用规则后再重新添加。
400	100216	The application input 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。
500	101424	Internal server error happened when create rule.	创建规则异常。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。

Http 状态码	错误码	错误描述	说明
200	101401	The rule input is invalid.	接口参数输入不合法。 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。
400	101416	The action of rule is invalid.	Action 参数不合法。 处理建议：请参考 Action 结构体参数说明检查接口请求中的 Action 参数是否有误。
400	101418	Action id duplicated in same rule.	Action id 重复。 处理建议：请检查接口请求中每个 Action 的 ID 唯一。
400	101425	Rule api parameter is invalid.	接口参数输入不合法。 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。
500	50204	Convert json to class failed.	数据格式转换失败。 处理建议：请检查接口请求中的 json 类型参数是否有误。
400	101406	The condition of rule is invalid.	condition 参数不合法。 处理建议：请参考 condition 结构体参数说明检查接口请求中的 condition 参数是否有误。
400	101417	Condition id duplicated in same rule.	condition id 重复。 处理建议：请确认接口请求中每个 condition 的 ID 唯一。
400	101420	The device of condition is not exist.	condition 中携带的 deviceId 所对应的设备不存在。 处理建议：请检查 condition 中的 deviceId 是否有误。
400	101421	The device of action is not exist.	acition 中携带的 deviceId 所对应的设备不存在。 处理建议：请检查 action 中的 deviceId 是否有误。
400	101422	The deviceData of condition is not exist, not bind.	condition 中携带的 deviceId 所对应的设备数据不存在。 处理建议：请检查 condition 中的 deviceId 是否有误。
400	101423	The deviceData of action is not exist,	action 中携带的 deviceId 所对应的

Http 状态码	错误码	错误描述	说明
		not bind.	设备数据不存在。 处理建议：请检查 action 中的 deviceId 是否有误。

2.4.2 更新规则

典型场景

第三方应用可以调用此接口更新已创建规则的内容。

接口功能

支持第三方应用更新规则内容。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C52	整体参数优化
1.1.0	V100R001C30	整体参数优化
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	PUT
请求地址	https://server:port/iocm/app/rule/v1.2.0/rules?appId={appId}
传输协议	HTTPS

请求参数

查看创建规则中的 RuleDTO1.2 结构体说明。

响应参数

StatusCode：200 OK

参数	类型	描述
ruleId	String	规则实例 id。

请求示例

Method: PUT

Request:

https://server:port/iocm/app/rule/v1.2.0/rules

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

Body:

{

"ruleId":"rule1"

"name": "rule1",

"author": "wudashan",

"description": "this is a test rule 2015.12.23",

"status": "active",

"logic": "and",

"conditions": [

{

"type": "DEVICE_DATA",

"deviceInfo": {

"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",

"path": "DoorWindow/status"

},

"operator": "=",

"value": "OPEN"

}

],

"actions": [

{

"type": "DEVICE_CMD",

"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",

"cmd": {

"serviceId": "DoorWindow",

"messageType": "MUTE",

"messageBody": {

"force": "true"

}

}

}

]

}

异步命令 1.3 版本更新规则 Demo

Method: PUT

Request:

https://server:port/iocm/app/rule/v1.2.0/rules

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

Body:

{

"ruleId":"rule1"

"name": "rule1",

"author": "wudashan",

"description": "this is a test rule 2015.12.23",

"status": "active",

"logic": "and",

```
"conditions": [
{
"type": "DEVICE_DATA",
"deviceInfo": {
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"path": "DoorWindow/status"
},
"operator": "=",
"value": "OPEN"
}
],
"actions": [
{
"type": "DEVICE_CMD",
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"cmd": {
"serviceId": "DoorWindow",
"messageType": "MUTE",
"messageBody": {
"force": "true"
}
},
"cmdVersion": "V1.3.0"
}
]
}
```


响应示例


```
response:
Status Code: 200 OK
Content-Type: application/json
Body:
{
"ruleId": "********"
```

}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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。
400	101419	RuleId can not be null or empty when update rule.	ruleId 不能为空。 处理建议：请确认接口请求中的 ruleId 不为空。
401	101409	Authentication failure.	鉴权失败。 处理建议：请检查接口请求中是否携带 appid。
200	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	101403	The rule is not existed.	规则不存在。 处理建议：请确认接口请求中的 ruleId 是否有误。
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	101402	The rule name already existed.	规则名称已存在。 处理建议：请修改接口请求中的规则名称。
500	100023	The data in dataBase is abnormal.	数据库异常。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。
200	101404	The rule count of each app has reached the upper limit, the current up limit is 30 entries.	应用下的规则数量超出上限，当前上限为 30。 处理建议：如果需要继续添加规则，请删除其他无用规则后再重新添加。
400	100216	The application input	请求参数不合法。

Http 状态码	错误码	错误描述	说明
		is invalid.	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。
500	101424	Internal server error happened when create rule.	创建规则异常。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。
200	101401	The rule input is invalid.	接口参数输入不合法。 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。
400	101416	The action of rule is invalid.	Action 参数不合法。 处理建议：请参考 Action 结构体参数说明检查接口请求中的 Action 参数是否有误。
400	101418	Action id duplicated in same rule.	Action id 重复。 处理建议：请检查接口请求中每个 Action 的 ID 唯一。
400	101425	Rule api parameter is invalid.	接口参数输入不合法。 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。
500	50204	Convert json to class failed.	数据格式转换失败。 处理建议：请检查接口请求中的 json 类型参数是否有误。
400	101406	The condition of rule is invalid.	condition 参数不合法。 处理建议：请参考 condition 结构体参数说明检查接口请求中的 condition 参数是否有误。
400	101417	Condition id duplicated in same rule.	condition id 重复。 处理建议：请确认接口请求中每个 condition 的 ID 唯一。
400	101420	The device of condition is not exist.	condition 中携带的 deviceId 所对应的设备不存在。 处理建议：请检查 condition 中的 deviceId 是否有误。
400	101421	The device of action is not exist.	acition 中携带的 deviceId 所对应的设备不存在。 处理建议：请检查 action 中的 deviceId 是否有误。

Http 状态码	错误码	错误描述	说明
400	101422	The deviceData of condition is not exist, not bind.	condition 中携带的 deviceId 所对应的设备数据不存在。 处理建议：请检查 condition 中的 deviceId 是否有误。
400	101423	The deviceData of action is not exist, not bind.	action 中携带的 deviceId 所对应的设备数据不存在。 处理建议：请检查 action 中的 deviceId 是否有误。

2.4.3 删除规则

典型场景

第三方应用可以调用此接口删除已创建的规则。

接口功能

支持第三方应用删除指定 ruleId 的规则。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C52	• 增加请求参数 appId • 删除错误码 101408 • 新增错误码 100203，100217
1.1.0	V100R001C30	• 新增请求头参数 app_key，Authorization • 新增错误码 100002
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	DELETE
请求地址	https://server:port/iocm/app/rule/v1.2.0/rules/{ruleId}?appId={appId}
传输协议	HTTPS

请求参数

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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
app_key	必选	String	header	鉴权用户名，填写创建应用时返回的应用 id。
Authorization	必选	String	header	鉴权信息，填写调用鉴权接口后返回的 accessToken。
ruleId	必选	String	path	规则的 ID。

响应参数

StatusCode: 204 No Content

请求示例

Method: DELETE

Request:

http://server:port/iocm/app/rule/v1.1.0/rules/{ruleId}

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
401	100002	Invalid access token.	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。
401	101409	Authentication failure.	鉴权失败。 处理建议：请检查接口请求中是否携带 appid。
200	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	101403	The rule is not existed.	规则不存在。 处理建议：请确认接口请求中的 ruleId 是否有误。
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。

2.4.4 查找规则

典型场景

第三方应用可以调用此接口查询规则的配置信息。

接口功能

支持第三方应用查询规则的配置信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C52	• 增加请求参数 appId，select • 新增错误码 100203，100217
1.1.0	V100R001C30	新增请求头参数 app_key，Authorization

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	GET
请求地址	https://server:port/iocm/app/rule/v1.2.0/rules?name={name}&author={author}
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	鉴权用户名，填写创建应用时返回的应用 id。
Authorization	必选	String	header	鉴权信息，填写调用鉴权接口后返回的 accessToken。
author	必选	String	query	创建规则的用户。
name	可选	String	query	规则名称。
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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。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

StatusCode：200 OK

Rule 列表，消息体参数定义查看创建规则中的 RuleDTO1.2 结构体说明。

请求示例

Method: GET

Request:

http://server:port/iocm/app/rule/v1.2.0/rules?name={name}&author={author}

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

{

"name": "****",

"author": "*****",

"description": "this is a test rule 2015.12.23",

"type": "noType",

"status": "active",

"logic": "and",

"conditions": [

{

"type": "DeviceData",

"id": 0,

"deviceInfo": {

"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",

"path": "DoorWindow/status"

},

"operator": "=",

"value": "OPEN"
}
],
"actions": [
{
"type": "DeviceCMD",
"id": 0,
"appId": "pnS6yhwLszJ8kZF7HhuH7cOeoGQa",
"deviceId": "0285f8e6-f0f1-441a-bc27-30f02545658c",
"cmd": {
"serviceId": "DoorWindow",
"messageType": "MUTE",
"messageBody": {
"force": "true"
}
}
}
]
}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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。
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	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。

2.4.5 修改规则状态

典型场景

第三方应用可以调用此接口修改规则的状态，如激活、去激活状态。

接口功能

支持第三方应用修改规则状态。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C30	• 新增请求头参数 appId，app_key、Authorization。 • 删除错误码 101406 • 新增错误码 100002，100203，100217，101401
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	PUT
请求地址	https://server:port/iocm/app/rule/v1.2.0/rules/{ruleId}/status/{status}?appId={appId}
传输协议	HTTPS

请求参数

参数	必选/可选	类型	位置	描述

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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
app_key	必选	String	header	鉴权用户名，填写创建应用时返回的应用 id。
Authorization	必选	String	header	鉴权信息，填写调用鉴权接口后返回的 accessToken。
ruleId	必选	String	path	规则的 ID。
status	必选	String	path	需要修改的规则状态。 • active 表示激活 • inactive 表示未激活

响应参数

StatusCode：200 OK

请求示例

Method: PUT

request:
https://server:port/iocm/app/rule/v1.2.0/rules/5a586c92050ec6cfc8972a19/status/inactive

Header:

"app_key:******"

"Authorization:Bearer ******"

"Content-Type:application/json"

Body:

{

"status": "******"

}

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
401	100002	Invalid access token.	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。
401	101409	Authentication failure.	鉴权失败。 处理建议：请检查接口请求中是否携带 appid。
200	100203	The application is not existed.	授权的应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	101403	The rule is not existed.	规则不存在。 处理建议：请确认接口请求中的 ruleId 是否有误。
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	101401	The rule input is invalid.	接口参数输入不合法。 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。

2.4.6 批量修改规则状态

典型场景

第三方应用可以调用此接口批量修改规则的状态，如激活、去激活状态。

接口功能

支持第三方应用批量修改规则的状态。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C52	增加请求参数 appId

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	PUT
请求地址	https://server:port/iocm/app/rule/v1.2.0/rules/status?appId={appId}
传输协议	HTTPS

请求参数

参数	必选	类型	位置	描述
app_key	true	String	header	鉴权用户名，填写创建应用时返回的应用 id。
Authorization	true	String	header	鉴权信息，填写调用鉴权接口后返回的 accessToken。
appId	false	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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
requests	true	List<RuleStatusUpdateReqDTO>	body	请求结构体列表，具体参见 RuleStatusUpdateReqDTO 结构体。

RuleStatusUpdateReqDTO 结构体说明：

参数	必选	类型	位置	描述
ruleId	true	String	body	规则 Id。
status	true	String	body	需要修改的规则状态。 • active 表示激活 • inactive 表示未激活

响应参数

StatusCode：200 OK

参数	类型	描述
result	RuleResultBodyDTO	总体执行结果，具体参见 RuleResultBodyDTO 结构体。
responses	List<RuleResponseBodyDTO>	每个条目的执行结果，具体参见 RuleResponseBodyDTO 结构体。

RuleResultBodyDTO 结构体说明：

参数	类型	描述
result_code	String	总体执行结果代码，若执行成功则为“OK”，否则为“Partial OK”。
result_desc	String	总体执行结果描述，若全部执行成功则为“Operation successful.”，否则为“Operation partial ok.”。

RuleResponseBodyDTO 结构体说明：

参数	类型	描述
ruleId	String	规则 Id。
response	ExceptionMsg	执行结果，如果成功则为空，否则包含异常信息。

请求示例

Method: PUT

request: https://server:port/iocm/app/rule/v1.2.0/rules/status

Header:

"app_key: ******"

"Authorization: Bearer ******"

"Content-Type: application/json"

Body:

{

"requests": [

{

"ruleId": "5a586c92050ec6cfc8972a19",

"status": "inactive"

},

{

"ruleId": "5a583812050ec69b8f9b69b5",

"status": "inactive"

},

{

"ruleId": "5a530e01050ec64a65d27b95",

"status": "inactive"

}

]

}

响应示例

response:

Status Code: 200 OK

Content-Type: application/json

Body:

{

"result": {

"result_code": "Partial OK",

"result_desc": "Operation partial ok."

},

"responses": [

{

"ruleId": "******",
"response": null
},
{
"ruleId": "******",
"response": null
},
{
"ruleId": "******",
"response": {
"error_code": "101409",
"error_desc": "Authentication failure."
}
}
]
}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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。
401	101409	Authentication failure.	鉴权失败。 处理建议：请检查接口请求中是否携带 appid。
200	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	101403	The rule is not existed.	规则不存在。 处理建议：请确认接口请求中的 ruleId 是否有误。

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	101401	The rule input is invalid.	接口参数输入不合法。 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。

2.5 订阅管理

IoT 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 2.6 消息推送接口配合使用。

2.5.1 订阅平台业务数据

典型场景

第三方应用可以调用此接口订阅设备变更通知，当设备的业务信息发生变更（如设备上报数据）时可以将变更信息推送给第三方应用。

接口功能

第三方应用订阅设备变更通知，当设备的业务信息发生变更时平台会推送给第三方应用。回调地址有效时间 30 分钟，第三方应用需要在 30 分钟内来平台刷新订阅信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/sub/v1.2.0/subscribe
传输协议	HTTPS

请求参数

字段	必选/可选	类型	位置	描述
notifyType	必选	String	body	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。 1. deviceAdded（添加新设备） 2. deviceInfoChanged（设备信息变化） 3. deviceDataChanged（设备数据变化） 4. deviceDatasChanged（设备数据批量变化） 5. deviceDeleted（删除设备） 6. messageConfirm（消息确认） 7. commandRsp（命令响应） 8. deviceEvent（设备事件） 9. serviceInfoChanged（服务信息变化） 10. ruleEvent（规则事件） 11. deviceModelAdded（添加设备模型） 12. deviceModelDeleted（删除设备模型）
callbackurl	必选	String(1024)	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：https://XXX.XXX.XXX.XXX:443/callbackurltest **说明** HTTP 信道只可用于调测。

响应参数

Status Code: 201 Created

请求示例

Method: POST

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

Header:

"app_key : ******"

"Authorization: Bearer ******"

Content-Type:application/json

Body:

{

"notifyType": "deviceInfoChanged",

"callbackurl": "http://**********"

}

响应示例

response:

Status Code: 201 Created

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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。
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。
200	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	101017	Get newCallbackUrl	从 oss 获取新的回调地址失败。

Http 状态码	错误码	错误描述	说明
		from oss failed.	处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。
500	101016	Get iotws address failed.	获取 iotws 地址失败。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。

2.5.2 订阅平台管理数据

典型场景

第三方应用可以调用此接口订阅设备变更通知，当设备的管理信息发生变更（如设备版本升级）时可以将变更信息推送给第三方应用。

接口功能

第三方应用订阅设备变更通知，当设备的管理信息发生变更时平台会推送给第三方应用。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	参数 notifyType 新增通知类型 swUpgradeStateChangeNotify，swUpgradeResultNotify，fwUpgradeStateChangeNotify 和 fwUpgradeResultNotify
1.1.0	V100R001C30	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

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
notifyType	必选	String	body	通知类型。 1. swUpgradeStateChangeNotify 2. swUpgradeResultNotify 3. fwUpgradeStateChangeNotify 4. fwUpgradeResultNotify
callbackurl	必选	String	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：https://XXX.XXX.XXX.XXX:443/callbackurltest 说明 HTTP 信道只可用于调测。

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": " sensorVersionUpgradeNotify",

"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
401	100002	Invalid access token.	错误的 token 信息。 处理建议：请检查接口请求中的 accessToken 是否正确。
500	120001	Internal server error.	服务器运行内部错误。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。
200	120203	The application is not existed.	应用不存在。 处理建议: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	120007	Bad request message.	请求错误。 处理建议：请检查接口请求中的回调地址参数 callbackurl 是否正确。

2.5.3 查询单个订阅

典型场景

第三方应用可以调用此接口查询某个订阅的信息。

接口功能

支持第三方应用根据 subscriptionId 查询单个订阅信息。

变更记录

接口版本	IoT 平台版本	变更说明

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

接口原型

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

请求参数

参数	必选/可选	类型	位置	描述
subscriptionId	必选	String(256)	path	订阅 ID 号，通过调用查询订阅接口获得。
appId	可选	String(256)	query	资源所属的 appId

响应参数

Status Code: 200 OK

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。

请求示例

Method: GET

request:https://server:port/iocm/app/sub/v1.2.0/subscriptions/*********?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
401	100002	Invalid access token.	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	200001	The resource is not found	找不到资源。 处理建议：请检查订阅 ID 是否有误。
200	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
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。

2.5.4 批量查询订阅

典型场景

第三方应用可以调用此接口查询多个订阅的信息。

接口功能

支持第三方应用按照条件查询订阅信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

接口原型

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

请求参数

参数	必选/可选	类型	位置	描述
appId	可选	String(256)	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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。

参数	必选/可选	类型	位置	描述
notifyType	可选	String(256)	query	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。 1. deviceAdded（添加新设备） 2. deviceInfoChanged（设备信息变化） 3. deviceDataChanged（设备数据变化） 4. deviceDatasChanged（设备数据批量变化） 5. deviceDeleted（删除设备） 6. messageConfirm（消息确认） 7. commandRsp（命令响应） 8. deviceEvent（设备事件） 9. serviceInfoChanged（服务信息变化） 10. ruleEvent（规则事件） 11. deviceModelAdded（添加设备型号） 12. deviceModelDeleted（删除设备型号）
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

Status Code: 200 OK

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 SubscriptionDTO 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。

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}

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
401	100002	Invalid access token.	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。
200	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
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。

2.5.5 删除单个订阅

典型场景

第三方应用可以调用此接口删除某个订阅记录。

接口功能

支持第三方应用根据 subscriptionId 删除单个订阅记录。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

接口原型

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

请求参数

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(256)	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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。

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=*********

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
401	100002	Invalid access token.	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。
200	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
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。
404	200001	The resource is not found	找不到资源。 处理建议：请检查订阅 ID 是否有误。

2.5.6 批量删除订阅

典型场景

第三方应用可以调用此接口删除多个订阅记录。

接口功能

支持第三方应用删除多个订阅记录。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C52	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

接口原型

请求方法	DELETE
请求地址	https://server:port/iocm/app/sub/v1.2.0/subscriptions?appId={appId}&notifyType={notifyType}&callbackUrl={callbackUrl}
传输协议	HTTPS

请求参数

参数	必选/可选	类型	位置	描述
appId	可选	String(256)	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 IoT 平台的开放 API。appid 在 IoT 平台的 SP Portal 上创建应用时获得。此处填写授权应用的 appid。
notifyType	可选	String(256)	query	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。 1. deviceAdded（添加新设备） 2. deviceInfoChanged（设备信息变化） 3. deviceDataChanged（设备数据变化） 4. deviceDatasChanged（设备数据批量变化） 5. deviceDeleted（删除设备） 6. messageConfirm（消息确认） 7. commandRsp（命令响应） 8. deviceEvent（设备事件） 9. serviceInfoChanged（服务信息变化） 10. ruleEvent（规则事件） 11. deviceModelAdded（添加设备型号） 12. deviceModelDeleted（删除设备型号）
callbackUrl	可选	String(256)	query	订阅回调的 URL 地址。

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=******&notifyType=******&callbackUrl=******

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
401	100002	Invalid access token.	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。
200	100203	The application is not existed.	授权应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
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。
404	200001	The resource is not found	找不到资源。 处理建议：请检查订阅 ID 是否有误。

2.6 消息推送

第三方应用向 IoT 平台订阅设备信息，当设备信息发生变更时，IoT 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 2.5 订阅管理接口使用。

2.6.1 注册设备通知

典型场景

订阅注册设备通知后（订阅的通知类型为 deviceAdded），在 IoT 平台注册设备时，平台会给第三方应用推送注册设备通知。

接口功能

支持 IoT 平台向第三方应用推送注册设备通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C10	新增接口

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	由第三方应用订阅请求消息中的 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。
gatewayId	可选	String	body	网关 ID，用于唯一标识一个网关设备。

参数	必选/可选	类型	位置	描述
nodeType	必选	String	body	设备类型。 • ENDPOINT • GATEWAY • UNKNOWN
deviceInfo	必选	DeviceInfo	body	设备信息，具体参见 DeviceInfo 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。 说明 若使用 IMEI 作为 nodeId，根据不同厂家的芯片有不同填写要求。 • 高通芯片设备的唯一标识为 urn:imei:xxxx，xxxx 为 IMEI 号 • 海思芯片设备的唯一标识为 IMEI 号 • 其他厂家芯片的设备唯一标识请联系模组厂家确认。
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。

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，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 IoT 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 status 和 statusDetail。
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
supportedSecurity	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
isSecurity	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
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 IoT 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

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

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
}
}

2.6.2 设备信息变化通知

典型场景

订阅设备信息变化通知后（订阅的通知类型为 deviceInfoChanged），设备信息发生变化时平台会给第三方应用推送设备信息变化通知。

接口功能

支持 IoT 平台向第三方应用推送设备信息变化通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C10	新增接口

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	由第三方应用订阅请求消息中的 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。
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 DeviceInfo 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。 说明 若使用 IMEI 作为 nodeId，根据不同厂家的芯片有不同填写要求。 • 高通芯片设备的唯一标识为 urn:imei:xxxx，xxxx 为 IMEI 号 • 海思芯片设备的唯一标识为 IMEI 号 • 其他厂家芯片的设备唯一标识请联系模组厂家确认。
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 IoT 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 status 和 statusDetail。
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
supportedSecurity	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
isSecurity	可选	String	body	表示设备当前是否启用安全模式。 • TRUE：启用

参数	必选/可选	类型	位置	描述
				• FALSE：未启用
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 IoT 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

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

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"
}
}

2.6.3 设备数据变化通知

典型场景

订阅设备数据变化通知后（订阅的通知类型为 deviceDataChanged），设备数据发生变化时平台会给第三方应用推送设备数据变化通知。

接口功能

支持 IoT 平台向第三方应用推送设备数据变化通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C30	删除请求参数 timestamp 和 eventTime。
1.1.0	V100R001C10	新增接口

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	由第三方应用订阅请求消息中的 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。
gatewayId	必选	String	body	网关 ID，用于唯一标识一个网关设备。
service	必选	DeviceServiceData	body	设备的服务数据，具体参见 DeviceServiceData 结构体。

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	时间发生时间，时间格式 yyyymmddThhmissZ，例如 20151212T121212Z。

响应参数

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "deviceDataChanged",

"requestId": "**********",

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

"gatewayId": "**********",

"service": {

"serviceId": "Brightness",

"serviceType": "Brightness",

"data": {

"brightness": 80

},

"eventTime": "20170311T163657Z"

}

}

2.6.4 批量设备数据变化通知

典型场景

订阅设备数据批量变化通知后（订阅的通知类型为 deviceDatasChanged），发生批量设备数据变化时，平台会给第三方应用推送设备数据批量变化通知。

接口功能

支持 IoT 平台向第三方应用推送设备数据批量变化通知。

变更记录

接口版本	IoT 平台版本	变更说明

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C30	新增接口

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	由第三方应用订阅请求消息中的 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。
gatewayId	必选	String	body	网关 ID，用于唯一标识一个网关设备。
services	必选	List<DeviceServiceData>	body	服务列表信息，具体参见 DeviceServiceData 结构体。

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。

参数	必选/可选	类型	位置	描述
eventTime	必选	String	body	事件上报时间，时间格式：yyyymmddThhmissZ，例如20151212T121212Z。

响应参数

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

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"

}]

}

2.6.5 设备服务信息变化通知

典型场景

订阅设备服务信息变化通知后（订阅的通知类型为 serviceInfoChanged），设备服务信息发生变化时平台会给第三方应用推送设备服务信息变化通知。

接口功能

支持 IoT 平台向第三方应用推送设备服务信息变化通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C30	新增接口

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	由第三方应用订阅请求消息中的 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。

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 ServiceInfo 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	设备控制命令列表。

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "serviceInfoChanged",

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

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

"serviceType": "*******",

"gatewayId": "*******",

"serviceInfo":

{

"muteCmds":"VIDEO_RECORD"

}

}

2.6.6 删除设备通知

典型场景

订阅删除设备通知后（订阅的通知类型为 deviceDeleted），设备被删除时平台会给第三方应用推送删除设备通知。

接口功能

支持 IoT 平台向第三方应用推送删除设备通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C10	新增接口

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	由第三方应用订阅请求消息中的 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。
gatewayId	必选	String	body	网关 ID，用于唯一标识一个网关设备。

响应参数

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "deviceDeleted",

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

"gatewayId": "**********",

}

2.6.7 设备消息确认通知

典型场景

订阅设备消息确认通知后（订阅的通知类型为 messageConfirm），设备向平台确认收到消息时，平台会给第三方应用推送设备消息确认通知。

接口功能

支持 IoT 平台向第三方应用推送设备消息确认通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C10	新增接口

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	由第三方应用订阅请求消息中的 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	header	通知类型，取值：messageConfirm。
header	必选	Header	header	具体参见 Header 字段。
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)	header	消息的序列号，唯一标识该消息。
from	必选	String(1-128)	header	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)	header	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)	header	命令状态。 • sent：已发送 • delivered：已送达 • executed：已执行
timestamp	必选	String(1-32)	header	时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。

响应参数

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "messageConfirm",

"header": {

"requestId": "**********",

"from": "**********",

"to": "**********",

"status": "delivered",

"timestamp": "20151212T121212Z"

},

"body": {

}

}

2.6.8 设备命令响应通知

典型场景

订阅设备命令响应通知后（订阅的通知类型为 commandRsp），设备发送命令响应消息给平台时，平台会给第三方应用推送设备命令响应通知。

接口功能

支持 IoT 平台向第三方应用推送设备命令响应通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C30	删除请求参数 timestamp。
1.1.0	V100R001C10	新增接口。

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	由第三方应用订阅请求消息中的 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	header	通知类型，取值：commandRsp。
header	必选	Header	header	具体参见 Header 字段。
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)	header	消息的序列号，唯一标识该消息。
from	必选	String(1-128)	header	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}

参数	必选/可选	类型	位置	描述
to	必选	String(1-128)	header	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	header	设备 ID，用于唯一标识一个设备。
serviceType	必选	String	header	命令所属的服务类型。
method	必选	String(1-128)	header	存放的响应命令，如：INVITE-RSP。

响应参数

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "commandRsp",

"header": {

"requestId": "**********",

"from": "**********",

"to": "**********",

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

"serviceType": "Camera",

"method": "MUTE_COMMANDS"

},

"body": {}

}

2.6.9 设备事件通知

典型场景

订阅设备事件通知后（订阅的通知类型为 deviceEvent），设备上报事件给平台时，平台会给第三方应用推送设备事件通知。

接口功能

支持 IoT 平台向第三方应用推送设备事件通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C30	删除请求参数 to
1.1.0	V100R001C10	新增接口

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	由第三方应用订阅请求消息中的 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	header	通知类型，取值：deviceEvent。
header	必选	Header	header	具体参见 Header 字段。
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)	header	事件类型。
from	必选	String(1-128)	header	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}
timestamp	必选	String (1-32)	header	时间戳，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
eventTime	必选	String (1-32)	header	事件上报时间，时间格式：yyyyMMdd'T'HHmmss'Z'，如：20151212T121212Z。
deviceId	必选	String	header	设备 ID，用于唯一标识一个设备。
serviceType	必选	String	header	命令所属的服务类型。

响应参数

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "deviceEvent",

"header": {

"eventType": "*******",

"from": "/devices/{deviceId}/services/{serviceId}",

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

"serviceType": "*******",

"timestamp": "20151212T121212Z",

"eventTime": "20151212T121212Z",

},

"body": {

"usedPercent": 80

}

}

2.6.10 增加设备模型通知

典型场景

订阅设备模型增加通知后（订阅的通知类型为 deviceModelAdded），在 IoT 平台上增加设备模型时，平台会给第三方应用推送设备模型增加通知。

接口功能

支持 IoT 平台向第三方应用推送设备模型增加通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C30	新增接口

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	由第三方应用订阅请求消息中的 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	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 IoT 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

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "deviceModelAdded",

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

"deviceType": "**********",

"manufacturerName": "wulian",

" manufacturerId ": "**********",

"model": "**********",

"protocolType": "zigbee"

}

2.6.11 删除设备模型通知

典型场景

订阅设备模型删除通知后（订阅的通知类型为 deviceModelDeleted），设备模型被删除时平台会给第三方应用推送设备模型删除通知。

接口功能

支持 IoT 平台向第三方应用推送设备模型删除通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C30	新增接口

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	由第三方应用订阅请求消息中的 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	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

参数	必选/可选	类型	位置	描述
				识来指定哪个应用来调用 IoT 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

无

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "deviceModelAdded",

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

"deviceTyp ": "**********",

" manufacturerName": "**********",

"manufacturerId ": "**********",

"model": "**********",

"protocolType": "**********"

}

2.6.12 规则事件通知

典型场景

订阅规则事件通知后（订阅的通知类型为 ruleEvent），规则事件上报 IoT 平台时，平台会给第三方应用推送规则事件通知。

接口功能

支持 IoT 平台向第三方应用推送规则事件通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C30	新增接口

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	由第三方应用订阅请求消息中的 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：ruleEvent。
author	必选	String	body	创建此规则的用户的 ID（最大长度 256 个字符）。
ruleId	必选	String	body	规则实例的 ID
ruleName	必选	String	body	规则实例的名称。
logic	可选	String	body	多条件逻辑关系。
reasons	必选	List<Reaso	body	触发原因，对应 conditions，具体

参数	必选/可选	类型	位置	描述
		n>		参考 Reason 结构体。
triggerTime	必选	String	body	规则触发的时间。
actionsResults	必选	List<ActionResult>	body	规则动作执行的结果。

Reason 结构体说明：

参数	必选/可选	类型	位置	描述
satisfactionTime	必选	String	body	满足条件的时间。
type	必选	String	body	规则条件的类型。
id	可选	String	body	condition 的 ID。
info	可选	Json	body	不同条件类型携带不同信息。
transInfo	可选	Json	body	事件推送时的回填信息，对应规则的 condition 中的 transInfo。

DEVICE_DATA 类型的条件携带的 info：

参数	必选/可选	类型	位置	描述
realValue	可选	String	body	设备数据的实际值。

ActionResult 字段说明：

参数	必选/可选	类型	位置	描述
type	必选	String	body	规则动作的类型。
id	可选	String	body	action 的 ID。
exception	与 result 二选一	String	body	规则引擎构造 action 对应的请求过程中遇到异常时携带的异常信息。
result	与 exception 二选一	Json	body	动作的执行结果，对于 DEVICE_CMD/SMS/EMAIL 类型的动作，内容为

参数	必选/可选	类型	位置	描述
				statusCode+body。
info	可选	Json	body	不同动作类型携带不同信息。
transInfo	可选	Json	body	事件推送时的回填信息，对应规则的 action 中的 transInfo。

DEVICE_CMD 类型的动作携带的 info

参数	必选/可选	类型	位置	描述
requestId	必选	String	body	下发异步命令时，为请求消息分配的 UUID。

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType": "ruleEvent",

"author": "*****",

"ruleId": "**********",

"ruleName": "name",

"reasons": [{

"satisfactionTime": "yyyyMMddTHHmmssZ",

"type": "******"

}

],

"triggerTime": "yyyyMMddTHHmmssZ",

"actionsResults": [{

"type": "******",

"exception": "******"

}

]

}

2.6.13 软件升级状态变更通知

典型场景

第三方应用订阅了设备软件升级状态变化后（订阅的通知类型为swUpgradeStateChangeNotify），当设备进行软件升级时，软件升级状态发生变化，IoT平台通过此接口向第三方应用推送软件升级状态通知。

接口功能

支持 IoT 平台向第三方应用推送软件升级状态通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

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	由第三方应用订阅请求消息中的 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。
appId	必选	String	body	设备所属应用的应用 ID。

参数	必选/可选	类型	位置	描述
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。 • downloading：设备正在下载软件包 • downloaded：设备下载软件包完成 • updating：设备正在进行升级 • idle：设备处于空闲状态

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType":"swUpgradeStateChangeNotify",

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

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

"operationId":"**********",

"subOperationId":"**********",

"swUpgradeState":"downloading"

}

2.6.14 软件升级结果变更通知

典型场景

第三方应用订阅了设备软件升级结果后（订阅的通知类型为 swUpgradeResultNotify），当设备软件升级有了升级结果时，IoT 平台通过此接口向第三方应用推送软件升级结果通知。

接口功能

支持 IoT 平台向第三方应用推送软件升级结果通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

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	由第三方应用订阅请求消息中的 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。
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
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。

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

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":"***",

}

2.6.15 固件升级状态变更通知

典型场景

第三方应用订阅了设备固件升级状态变化后（订阅的通知类型为fwUpgradeStateChangeNotify），当设备进行固件升级时，固件升级状态发生变化，IoT平台通过此接口向第三方应用推送固件升级状态通知。

接口功能

支持 IoT 平台向第三方应用推送固件升级状态通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

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	由第三方应用订阅请求消息中的 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。

参数	必选/可选	类型	位置	描述
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。 • 1：downloading：设备正在下载软件包 • 2：downloaded：设备下载软件包完成 • 3：updating：设备正在进行升级 • 0：idle：设备处于空闲状态

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType":"fwUpgradeStateChangeNotify",

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

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

"operationId":"**********",

"subOperationId":"**********",

"step":"1",

"stepDesc":"downloading"

}

2.6.16 固件升级结果变更通知

典型场景

第三方应用订阅了设备固件升级结果后（订阅的通知类型为 fwUpgradeResultNotify），当设备固件升级有了升级结果时，IoT 平台通过此接口向第三方应用推送固件升级结果通知。

接口功能

支持 IoT 平台向第三方应用推送固件升级结果通知。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

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	由第三方应用订阅请求消息中的 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。
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。

参数	必选/可选	类型	位置	描述
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。 • SUCCESS • FAIL
statusDesc	必选	String	body	升级结果描述。 • SUCCESS：设备升级成功 • FAIL：设备升级失败
upgradeTime	必选	String	body	固件升级时长。

请求示例

Method: POST

request: {callbackUrl}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type:application/json

Body:

{

"notifyType":"fwUpgradeResultNotify",

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

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

"operationId":"**********",

"subOperationId":"**********",

"curVersion":"1.6",

"targetVersion":"1.6",

"sourceVersion":"1.3",

"status":"SUCCESS",

"statusDesc":"****",

"upgradeTime":"****"

}

2.7 命令下发

2.7.1 创建设备命令

典型场景

第三方应用可调用此接口向设备下发命令，有三种下发命令模式（在 SP Portal 上设置）：PSM，DRX，EDRX，模式不同使用此接口下发命令的策略不同。

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。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C52	• 请求结构体 PostDeviceCommandReq 增加参数 maxRetransmit • 增加响应参数 maxRetransmit
1.4.0	V100R001C30	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	POST

请求地址	https://server:port/iocm/app/cmd/v1.4.0/deviceCommands
传输协议	HTTPS

请求参数

参数	必选/可选	类型	位置	描述
body	必选	PostDeviceCommandReq	body	设备命令的内容，具体参见 PostDeviceCommandReq 结构体。
appId	可选	String(1-64)	query	第三方应用的身份标识，用于唯一标识一个应用。appid 在 IoT 平台的 SP Portal 上创建应用时获得。

PostDeviceCommandReq 结构体说明：

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 CommandDTOV4 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 IoT 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	可选	ObjectNode	body	命令参数的 jsonString，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。
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 IoT 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 CommandDTOV4 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。 • DEFAULT 表示未下发 • EXPIRED 表示命令已经过期 • SUCCESSFUL 表示命令已经成功执行 • FAILED 表示命令执行失败 • TIMEOUT 表示命令下发执行超时 • CANCELED 表示命令已经被撤销执行

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。

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/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/cmd/callbackUr",
"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

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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。
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。
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	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
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。
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值。默认限定值为 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 修改未 CANCELED。
200	100428	The device is not online.	设备不在线 错误解释：立即下发命令（expireTime 取值为 0）时，若设备 OFFLINE 会报此错。 处理建议：请检查设备的状态是否在线。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： 1. 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。

Http 状态码	错误码	错误描述	说明
			2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。 3. 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。
200	100431	The serviceType is not exist.	服务类型不存在。 处理建议： 1. 请检查该设备的 Profile 文件是否已上传到 IoT 平台。 2. 请确认是否已调用“查询单个设备信息”接口修改设备的 model 和 manufacturerId 等参数。 3. 请检查请求参数是否正确，或者 serviceId 在 Profile 文件中是否存在。
500	100023	The data in dataBase is abnomal.	数据库异常。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。
500	100001	Internal server error.	服务内部处理错误。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。
500	101016	Get iotws address failed.	获取 iotws 地址失败。 处理建议：IoT 平台内部错误，请联系 IoT 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：IoT 平台内部错误，请联系 IoT 平台维护人员处理。

2.7.2 查询设备命令

典型场景

第三方应用可调用此接口查询下发命令的状态信息。

接口功能

支持第三方应用查询下发的命令信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C30	新增接口

注意事项

http 携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	GET
请求地址	https://server:port/iocm/app/cmd/v1.4.0/deviceCommands?pageNo={pageNo}&pageSize={pageSize}&deviceId={deviceId}&startTime={startTime}&endTime={endTime}&appId={appId}
传输协议	HTTPS

请求参数

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：

参数	必选/ 可选	类型	位置	描述
				yyyyMMdd'T'HHmmss'Z'，如20151212T121212Z。
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。appid 在 IoT 平台的 SP Portal 上创建应用时获得。

响应参数

Status Code: 200 OK

参数	类型	描述
pagination	Pagination	页面信息，具体参见 Pagination 结构体。
data	List<DeviceCommandRespV4>	设备命令列表，具体参见 DeviceCommandRespV4 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 IoT 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 CommandDTOV4 结构体。
callbackUrl	String(1024)	令状态变化通知地址，当命令状态变化时

参数	类型	描述
		（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。 • DEFAULT 表示未下发 • EXPIRED 表示命令已经过期 • SUCCESSFUL 表示命令已经成功执行 • FAILED 表示命令执行失败 • TIMEOUT 表示命令下发执行超时 • CANCELED 表示命令已经被撤销执行
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	ObjectNode	命令参数的 jsonString，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=20151212T121212Z

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/cmd/callbackUr",

"expireTime": null,

```
"status": "PENDDING",
"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/cmd/callbackUr",
"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
401	100002	Invalid access token.	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。
400	100022	The input is invalid.	输入参数无效。 处理建议： 1. 请确认 startTime 和 endTime 都不为空，且 endTime 比 startTime 晚。 2. 请确认 pageNo 不为空，且 pageNo 大于 0。 3. 请检查 pageSize 不为空，且 pageSize 大于 1。
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。
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	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： 1. 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。 2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。 3. 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。

2.7.3 修改设备命令

典型场景

第三方应用可调用此接口对还未撤销，过期或者执行的命令状态进行修改，当前仅能修改为撤销状态。

接口功能

支持第三方应用修改命令的状态信息，当前仅支持修改的命令状态为撤销，即将命令撤销。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C30	新增接口

注意事项

http 携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

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
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。

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。appid 在 IoT 平台的 SP Portal 上创建应用时获得。
body	必选	UpdateDeviceCommandReq	body	修改设备命令的内容，具体参见 UpdateDeviceCommandReq 结构体。

UpdateDeviceCommandReq 结构体说明：

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

Status Code: 200 OK

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 IoT 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 CommandDTOV4 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。 • DEFAULT 表示未下发 • EXPIRED 表示命令已经过期 • SUCCESSFUL 表示命令已经成功执行 • FAILED 表示命令执行失败

参数	类型	描述
		• TIMEOUT 表示命令下发执行超时 • CANCELED 表示命令已经被撤销执行
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	ObjectNode	body	命令参数的 jsonString，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:

StatusCode: 200 OK

Content-Type: application/json

Body:

{

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

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

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

"command": {

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

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

"paras": {

"paraName1": "paraValue1",

"paraName2": "paraValue2"

}

},

"callbackUrl": "http://127.0.0.1/cmd/callbackUr",

"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
401	100002	Invalid access token.	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。
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。
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。
401	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	100435	The device command already canceled, expired or executed, Cannot cancel.	命令已撤销、过期或执行，无法撤销。 处理建议：请检查要撤销的命令是否为 PENDING 状态。
200	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	100434	The device command is not existed.	命令不存在。 处理建议: 1. 请检查接口请求中的 commandId 是否有误。 2. 请检查 commandId 是否有误，是否属于当前登录的 appId。

2.7.4 创建设备命令撤销任务

典型场景

第三方应用调用此接口对某设备下发的命令进行撤销，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，该任务用于撤销指定设备 ID 下的所有设备命令。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C30	新增接口

注意事项

http 携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

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。appid 在 IoT 平台的 SP Portal 上创建应用时获得。
body	必选	CreateDeviceCommandCancelTaskReq	body	设备命令撤销任务参数，具体参见 CreateDeviceCommandCancelTaskReq 结构体。

CreateDeviceCommandCancelTaskReq 结构体说明：

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

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。
status	String	撤销任务的任务状态。 • WAITTING 表示等待执行中 • RUNNING 表示撤销任务正在执行 • SUCCESS 表示撤销任务执行成功 • FAILED 表示撤销任务执行失败 • PART_SUCCESS 表示撤销任务部分执行成功。
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 DeviceCommandRespV4 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 IoT 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 CommandDTOV4 结构体。

参数	类型	描述
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。 • DEFAULT 表示未下发 • EXPIRED 表示命令已经过期 • SUCCESSFUL 表示命令已经成功执行 • FAILED 表示命令执行失败 • TIMEOUT 表示命令下发执行超时 • CANCELED 表示命令已经被撤销执行
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	ObjectNode	命令参数的 jsonString，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/cmd/callbackUr",

"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
401	100002	Invalid access token.	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。
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	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。
401	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。
404	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。

Http 状态码	错误码	错误描述	说明
			2. 请检查请求路径（url）中传入的 appId 是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： 1. 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。 2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。 3. 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。

2.7.5 查询设备命令撤销任务

典型场景

第三方应用对某设备下发撤销命令操作后，可调用此接口查询撤销操作任务的执行情况。

接口功能

支持第三方应用查询设备命令撤销任务的执行情况。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C30	新增接口

注意事项

http 携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	GET
请求地址	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}
传输协议	HTTPS

请求参数

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。
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	第三方应用的身份标识，用于唯一标识一个应用。appid 在 IoT 平台的 SP Portal 上创建应用时获得。

响应参数

Status Code: 200 OK

参数	类型	描述

参数	类型	描述
pagination	Pagination	页码信息，具体参见 Pagination 结构体。
data	List<DeviceCommandCancelTaskRespV4>	设备命令列表，具体参见 DeviceCommandCancelTaskRespV4 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。
status	String	撤销任务的任务状态。 • WAITTING 表示等待执行中 • RUNNING 表示撤销任务正在执行 • SUCCESS 表示撤销任务执行成功 • FAILED 表示撤销任务执行失败 • PART_SUCCESS 表示撤销任务部分执行成功。
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 DeviceCommandRespV4 结构体。

DeviceCommandRespV4 结构体说明：

参数	类型	描述
commandId	String(1-64)	设备命令 ID。

参数	类型	描述
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 IoT 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 CommandDTOV4 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。 • DEFAULT 表示未下发 • EXPIRED 表示命令已经过期 • SUCCESSFUL 表示命令已经成功执行 • FAILED 表示命令执行失败 • TIMEOUT 表示命令下发执行超时 • CANCELED 表示命令已经被撤销执行
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。

参数	必选/可选	类型	描述
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	ObjectNode	命令参数的 jsonString，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/cmd/callbackUr",
"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/cmd/callbackUr",
"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
401	100002	Invalid access token.	错误的 token 信息。 处理建议：请检查接口请求中所携带的

Http 状态码	错误码	错误描述	说明
			accessToken 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	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。
401	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。
404	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
404	100418	The deviceData is not existed.	设备信息不存在。 处理建议： 1. 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。 2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。 3. 如果 URL 中携带了可选参数 appId，请检查该 appId 是否有误。

2.8 数据采集

IoT 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。

2.8.1 查询单个设备信息

典型场景

第三方应用调用此接口查询某个设备的详细信息。

接口功能

支持第三方应用查询指定条件的设备信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C52	• 请求地址变更 • 新增请求参数 select • 删除响应参数 connectionInfo，location • 返回参数 deviceInfo 类型更改为 DeviceInfoQueryDTO • 错误码 100203Http 状态码 200 拆分为 403 和 500 • 删除错误码 100022
1.3.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	GET
请求地址	https://server:port/iocm/app/dm/v1.4.0/devices/{deviceId}?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
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 IoT 平台分配获得。

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。appid 在 IoT 平台的 SP Portal 上创建应用时获得。
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。
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。
lastModifiedTime	String(256)	最后修改设备的时间。
deviceInfo	DeviceInfo	设备信息，具体参见 DeviceInfo 结构体。
services	List<DeviceService>	设备服务列表，具体参见 DeviceService 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。

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 IoT 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 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。 • TRUE：冻结状态 • FALSE：非冻结状态
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。

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 IoT 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 ServiceInfo 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>	设备控制命令列表。

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
401	100002	Invalid access token.	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	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
200	100403	The device is not existed.	设备不存在。 处理建议：IoT 平台未找到 deviceId 对应的设备，请检查 deviceId 是否有误。
200	100418	The deviceData is not existed.	设备信息不存在。 处理建议： 1. 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。 2. Header 中携带的 appId 有误，请确认该 appId 下是否有该 deviceId。

Http 状态码	错误码	错误描述	说明
			3. 如果 URL 中携带了可选参数 appId，请检查该 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。

2.8.2 批量查询设备信息列表

典型场景

第三方应用可调用此接口按条件查询设备信息列表。

接口功能

支持第三方应用按条件批量查询设备信息列表。

变更记录

接口版本	IoT 平台版本	变更说明
1.4.0	V100R001C52	• 请求地址变更 • 增加请求参数 select • 错误码 100218Http 状态码由 200 更改为 400 • 错误码 100203Http 状态码由 200 拆分为 403 和 500 • 错误码 100217Http 状态码由 200 更改为 403 • 返回参数整体优化
1.3.0	V100R001C30	• 增加请求参数 deviceType • 新增错误码 100002，100217
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	GET
请求地址	https://server:port/iocm/app/dm/v1.4.0/devices?appId={appId}&gatewayId={gatewayId}&nodeType={nodeType}&deviceType={deviceType}&pageNo={pageNo}&pageSize={pageSize}&startTime={startTime}&endTime={endTime}&status={status}&sort={sort}&selec={select}
传输协议	HTTPS

请求参数

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。appid 在 IoT 平台的 SP Portal 上创建应用时获得。
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
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

参数	必选/可选	类型	位置	描述
				20151212T121212Z。
sort	可选	String	query	指定返回记录的排序。 • ASC：按注册设备的时间升序排列 • DESC：按注册设备的时间降序排列 缺省值：DESC。
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<QueryDeviceDTO4Cloud2NA>	设备分页列表信息，具体参见 QueryDeviceDTO4Cloud2NA 结构体。

QueryDeviceDTO4Cloud2NA 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。
lastModifiedTime	String(256)	最后修改设备的时间。
deviceInfo	DeviceInfoQueryDTO	设备信息，具体参见 DeviceInfo 结构体。
services	List<DeviceService>	设备服务列表，具体参见 DeviceService 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 IoT 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 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。 • TRUE：冻结状态 • FALSE：非冻结状态

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 IoT 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 ServiceInfo 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>	设备控制命令列表。

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

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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。
400	100216	The application input 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。
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。
403	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。

2.8.3 查询设备历史数据

典型场景

第三方应用可调用此接口查询设备的历史数据。

接口功能

支持第三方应用查询设备历史数据。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C52	• 错误码 100203Http 状态码 200 拆分为 403 和 500 • 错误码 100217Http 状态码由 200 更改为 403
1.1.0	V100R001C30	• 新增请求参数 appId，property • 返回参数整体优化 • 新增错误码 100002，100217
1.1.0	V100R001C10	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

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}&serviceId={serviceId}&property={property}&pageNo={pageNo}&pageSize={pageSize}&startTime={startTime}&endTime={endTime}
传输协议	HTTPS

请求参数

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。appid 在 IoT 平台的 SP Portal 上创建应用时获得。
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。

参数	必选/可选	类型	位置	描述
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

StatusCode: 200 OK

参数	类型	描述
deviceDataHistoryDTOs	List<DeviceDataHistoryDTO>	设备历史数据列表，具体参见 DeviceDataHistoryDTO 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：

参数	类型	描述
		20151212T121212Z。

请求示例

Method: GET

request:

https://server:port/iocm/app/data/v1.2.0/deviceDataHistory?deviceId={deviceId}&gatewayId={gatewayId}&appId={appId}&serviceId={serviceId}&property={property}&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:

{

"deviceDataHistoryDTOs": [

{

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

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

"gatewayId": "*****",

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

"data": "*****",

"timestamp": "*****"

}

]

}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
500	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 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	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。

2.8.4 查询设备服务能力

典型场景

第三方应用可以调用此接口查询设备的服务能力（查询到的设备能力是 profile 中定义的服务），然后根据获取的服务能力进行命令下发等操作。

接口功能

支持第三方应用查询设备的服务能力。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	错误码 100203Http 状态码由 200 拆分为 403 和

接口版本	IoT 平台版本	变更说明
		500
1.1.0	V100R001C30	新增接口

注意事项 ：

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

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	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	可选	String	query	第三方应用的身份标识，用于唯一标识一个应用。appid 在 IoT 平台的 SP Portal 上创建应用时获得。
deviceId	可选	String	query	设备 ID，用于唯一标识一个设备。

响应参数

参数	类型	描述
deviceCapabilities	List<DeviceCapabilityDTO>	查询结果列表，具体参见 DeviceCapabilityDTO 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。
serviceCapabilities	List<ServiceCapabilityDTO>	设备的服务能力列表，具体参见 ServiceCapabilityDTO 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)	设备服务描述细心。
commands	List<ServiceCommand>	支持的命令名称列表，具体参见 ServiceCommand 结构体。
properties	List<ServiceProperty>	支持的属性名称列表，具体参见 ServiceProperty 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 ServiceCommandPara 结构体。
responses	List<ServiceCommandResponse>	响应列表，具体参见 ServiceCommandResponse 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。

参数	类型	描述
min	String	属性最小值。
max	String	属性最大值。
step	Double	步长。
maxLength	Int	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 ServiceCommandPara 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	int	最大长度。
method	String(256)	访问方法。 • R：可读 • W：可写 • E：可观察
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": "Ture",

"min": "******",

"max": "*****",
"step": "*****",
"maxLength": 1111111,
"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
401	100002	Invalid access token.	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。
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。
500	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议： 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。

2.9 设备组管理

2.9.1 增加设备组

典型场景

第三方应用可调用此接口添加设备组，用于支持设备升级、建立设备规则等，一个设备可以打上多个设备组的标签。

接口功能

支持第三方应用增加设备组，支持在本应用或授权应用下增加设备组。

变更记录

接口版本	IoT 平台版本	变更说明
1.3.0	V100R001C52	错误码 100203 的 Http 状态码由 200 拆分为 403 和 500
1.3.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

请求方法	POST
请求地址	https://server:port/iocm/app/devgroup/v1.3.0/devGroups
传输协议	HTTPS

请求参数

参数	必选/可选	类型	位置	描述
name	必选	String(1-	body	设备组名称，仅限大小写字母和数

参数	必选/可选	类型	位置	描述
		50)		字。
description	可选	String(1024)	body	设备组的描述信息。
appId	可选	String（50）	body	第三方应用的身份标识，在 IoT 平台的 SP Portal 上创建应用时获得。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

StatusCode: 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，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。

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": "*********"

}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 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: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。
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。
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。

2.9.2 删除设备组

典型场景

第三方应用可调用此接口删除已创建的设备组。

接口功能

支持第三方应用删除指定设备组，支持删除本应用或授权应用下的设备组。

变更记录

接口版本	IoT 平台版本	变更说明
1.3.0	V100R001C52	错误码 100203 的 Http 状态码由 200 拆分为 403 和 500
1.3.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

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 IoT 平台返回获得。
accessAppId	可选	String	param	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

StatusCode: 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

Body:

{

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

"accessAppId": "******",

}

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
401	100002	Invalid access token.	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。
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。
500	100203	The application is not existed.	应用不存在。 处理建议: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。

2.9.3 修改设备组信息

典型场景

第三方应用可调用此接口修改设备组信息。

接口功能

支持第三方应用修改指定设备组信息，支持修改本应用或授权应用下的设备组。

变更记录

接口版本	IoT 平台版本	变更说明
1.3.0	V100R001C52	错误码 100203 的 Http 状态码由 200 拆分为 403 和 500
1.3.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

参数	必选/可选	类型	位置	描述
accessAppId	可选	String	param	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。
devGroup	必选	String	path	设备组 ID，在增加设备组后由 IoT 平台

参数	必选/可选	类型	位置	描述
Id		（50）		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

StatusCode: 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": "******",

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

"maxDevNum": "********",

"deviceIds": "********"

}

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	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。
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。
500	100203	The application is not existed.	应用不存在。 处理建议: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 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: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。
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。

2.9.4 查询设备组详情

典型场景

第三方应用可调用此接口查询已创建设备组的详细信息。

接口功能

支持第三方应用查询满足条件的设备组信息，支持查询本应用或授权应用下的设备组。

变更记录

接口版本	IoT 平台版本	变更说明
1.3.0	V100R001C52	错误码 100203 的 Http 状态码由 200 拆分为 403 和 500
1.3.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

参数	必选/可选	类型	位置	描述
accessAppId	可选	String	param	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	param	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	param	每页设备组记录数量，默认值为 1。
name	可选	String	param	设备组名称。

响应参数

StatusCode: 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。
queryDevGroups	List<QueryDevGroupDTOCloud2NA>	设备组信息详情，具体参见 QueryDevGroupDTOCloud2NA 结构体。

QueryDevGroupDTOCloud2NA 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称。

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}

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type: application/json

Body:

{

"accessAppId": "********",

"pageNo": "******",

"pageSize": "********",

"name": "******"

}

响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

{

"totalCount": "*********",

"pageNo": "*******",

"pageSize": "*********",

"queryDevGroups": {

"name": "*******",

"description": "*******",

"id": "*******",

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

"maxDevNum": "*******",

"curDevNum": "*******",

"creator": "*******"

}

}

错误码

Http 状态码	错误码	错误描述	说明
401	100002	Invalid access token.	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: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。

2.9.5 查询指定设备组

典型场景

第三方应用可调用此接口查询指定设备组的信息。

接口功能

支持第三方应用很据设备组 ID 查询指定设备组信息，支持查询本应用或授权应用下的设备组。

变更记录

接口版本	IoT 平台版本	变更说明
1.3.0	V100R001C52	错误码 100203 的 Http 状态码由 200 拆分为 403 和 500
1.3.0	V100R001C30	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

参数	必选/可选	类型	位置	描述
devGrou	必选	String	path	设备组 ID，在增加设备组后由 IoT 平台返

参数	必选/可选	类型	位置	描述
pId				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

StatusCode: 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

Body:

{

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

"accessAppId": "******",

}

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
401	100002	Invalid access token.	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: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 请检查请求路径（url）中传入的 appId 是否有误。
403	100203	The application is not existed.	应用不存在。 处理建议: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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

Http 状态码	错误码	错误描述	说明
			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。

2.9.6 查询指定设备组成员

典型场景

第三方应用可调用此接口查询指定设备组的设备列表信息。

接口功能

支持第三方应用根据设备组 ID 查询指定设备组内设备列表，支持查询本应用或授权应用下的设备组。

变更记录

接口版本	IoT 平台版本	变更说明
1.2.0	V100R001C52	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

接口原型

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

请求参数

参数	必选/可选	类型	位置	描述
devGroupId	必选	String	query	设备组 ID，在增加设备组后由 IoT 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。
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(1000)	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

Body:

{

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

"accessAppId": "*******",

"pageNo": "*******",

"deviceIds": "*******"

}

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
401	100002	Invalid access token.	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: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。
400	100022	The input is invalid.	输入参数无效。

Http 状态码	错误码	错误描述	说明
			处理建议：请检查接口调用请求中携带参数的合法性。

2.9.7 向设备组添加设备

典型场景

第三方应用可调用此接口向指定设备组添加设备。

接口功能

支持第三方应用向指定设备组添加设备，并将设备服务能力统计到该设备组的服务能力表中，支持向本应用或授权应用下的设备组添加设备。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

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
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 IoT 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

StatusCode: 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:

{

"accessAppId": "********",

"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	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议: 1. 请检查接口请求中的设备 ID 是否有误。 2. 请检查设备组是否达到最大设备数量限制。
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	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。

2.9.8 从设备组删除设备

典型场景

第三方应用可调用此接口删除指定设备组内的设备。

接口功能

支持第三方应用从设备组删除设备，支持从本应用或授权应用下的设备组删除设备。

变更记录

接口版本	IoT 平台版本	变更说明
1.1.0	V100R001C52	新增接口

注意事项

携带头域信息

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json;

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
accessAppId	可选	String	param	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 IoT 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

StatusCode: 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:

{

"accessAppId": "********",

"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
401	100002	Invalid access token.	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: 1. 请检查 HTTP 请求头域中携带的 appId 是否有误。 2. 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。
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。

Http 状态码	错误码	错误描述	说明
400	100604	The device group request parameter is invalid.	请求参数不合法。 处理建议: 1. 请检查接口请求中的设备 ID 是否有误。 2. 请检查设备组是否达到最大设备数量限制。
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。

2.10 设备升级

2.10.1 查询版本包列表

典型场景

第三方应用可调用此接口查询已经上传到 IoT 平台的版本升级包信息。

接口功能

支持第三方应用查询满足指定条件的版本包列表信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

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}&manufactureName={manufactureName}&version={version}
传输协议	HTTPS

请求参数

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

Status Code: 200 OK

参数	类型	描述
data	array<CategoryInfo>	版本包列表信息，具体参见 CategoryInfo 结构体。

参数	类型	描述
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 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。

请求示例

Method: GET

request:

https://server:port/iodm/northbound/v1.5.0/category

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type: application/json

Body:

```
{
" deviceType ": "WaterMeter",
" fileType ": " firmwarePackage ",
" model ": "17",
" manufacturerName ": "***"
}
```

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。

2.10.2 查询指定版本包

典型场景

第三方应用可调用此接口查询已经上传到 IoT 平台的指定升级包信息。

接口功能

支持第三方应用查询指定版本包信息。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

接口原型

请求方法	GET
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
fileId	必选	string	path	版本包 ID，在上传版本包后获得。

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。 • 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。

参数	类型	描述
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
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。

2.10.3 删除指定版本包

典型场景

第三方应用可调用此接口删除已经上传到 IoT 平台的指定版本包。

接口功能

支持第三方应用删除指定的版本包文件。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

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
fileId	必选	string	path	版本包 ID，在上传版本包后获得。

响应参数

Status Code: 200ok

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
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。

2.10.4 创建软件升级任务

典型场景

第三方应用可调用此接口对 NB-IoT 设备进行软件版本升级，可同时升级多个设备。

接口功能

支持第三方应用对一个或多个 NB-IoT 设备进行软件升级，或者指定设备组的方式实现对批量设备进行软件升级。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

接口原型

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

请求参数

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 OperateDevices 结构体。
policy	可选	OperatePolicy	body	升级任务的执行策略，具体参见 OperatePolicy 结构体。

OperateDevices 结构体说明：

参数	必选/可选	类型	位置	描述

参数	必选/可选	类型	位置	描述
deviceGroups	可选	array<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	array<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

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
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。
400	123001	Bad request error.	请求错误。 处理建议：请检查 x-app-key 内容是否正确。
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	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	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	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

Http 状态码	错误码	错误描述	说明
			deviceGroups 或 devices 的内容，数量不能超过 256 个。
400	123027	some deviceGroups not found	部分设备组不存在。 处理建议：请检查参数接口请求中 deviceGroups 内容是否正确。
500	120001	Internal server error.	内部错误。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。

2.10.5 创建固件升级任务

典型场景

第三方应用可调用此接口对 NB-IoT 设备进行固件版本升级，可同时升级多个设备。

接口功能

支持第三方应用对一个或多个 NB-IoT 设备进行固件升级，或者指定设备组的方式实现对批量设备进行固件升级。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

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
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 OperateDevices 结构体。
policy	可选	OperatePolicy	body	升级任务的执行策略，具体参见 OperatePolicy 结构体。

OperateDevices 结构体说明：

参数	必选/可选	类型	位置	描述
deviceGroups	可选	array<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	array<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
executeTyp	必选	String	body	执行类型，默认值为 now。

参数	必选/可选	类型	位置	描述
e				• 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

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
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。
400	123001	Bad request error.	请求错误。 处理建议：请检查 x-app-key 内容是否正确。
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

Http 状态码	错误码	错误描述	说明
			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	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	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	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	123027	some deviceGroups not found	部分设备组不存在。 处理建议：请检查参数接口请求中 deviceGroups 内容是否正确。
500	120001	Internal server error.	内部错误。 处理建议：IoT 平台内部错误，请联系 IoT 平台维护人员处理。

2.10.6 查询指定升级任务结果

典型场景

第三方应用可调用此接口查询固件升级任务或软件升级任务的升级结果。

接口功能

支持第三方应用查询固件或软件升级任务的操作结果。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

接口原型

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

请求参数

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 IoT 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。

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。 • firmware_upgrade • software_upgrade
targets	OperateDevices	执行操作的目标设备，具体参见 OperateDevices 结构体。
policy	OperatePolicy	操作执行策略，具体参见 OperatePolicy 结构体。
status	string	操作任务的状态。 • wait：等待 • processing：正在执行 • failed：失败 • success：成功 • stop：停止
staResult	OperationStaResult	操作结果统计，具体参见 OperationStaResult 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 extendPara 请求参数。

OperateDevices 结构体说明：

参数	类型	描述
deviceGroups	array<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	array<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。

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
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。

2.10.7 查询指定升级任务子任务详情

典型场景

第三方应用可调用此接口查看创建的固件升级任务或软件升级任务的内容详情。

接口功能

支持第三方应用查看固件或软件升级任务的内容详情。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

接口原型

请求方法	GET
请求地址	https://server:port/iodm/northbound/v1.5.0/operations/{operationId}/subOperations?subOperationStatus={subOperationStatus}
传输协议	HTTPS

请求参数

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 IoT 平台返回获得。
subOperationStatus	可选	enum<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。 • wait：等待 • processing：正在执行 • fail：失败 • success：成功 • stop：停止
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

Status Code: 200 OK

参数	类型	描述
data	list<SubOperationInfo>	子任务列表信息，具体参见。

参数	类型	描述
pageNo	long	查询结果的页码。
pageSize	long	查询结果每页的记录数量。
totalSize	long	查询结果的总页数。

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	enum<string>	操作类型。 • firmware_upgrade • software_upgrade
deviceId	string	操作设备的设备 ID。
status	enum<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。

请求示例

Method: GET

request:

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

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type: application/json

Body:

{

" subOperationStatus ": " ****",

" pageNo ": 0,

" pageSize ": 1

}

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
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。

2.10.8 查询升级任务列表

典型场景

第三方应用可调用此接口查看固件升级任务或软件升级任务的任务列表。

接口功能

支持第三方应用查看固件或软件升级任务的任务列表。

变更记录

接口版本	IoT 平台版本	变更说明
1.5.0	V100R001C52	新增接口

注意事项

携带头域信息：

Header:

"app_key: ******"

"Authorization:Bearer ******"

Content-Type:application/json

接口原型

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

请求参数

参数	必选/可选	类型	位置	描述
operationType	可选	string(256)	query	操作类型。 • firmware_upgrade • software_upgrade
operationStatus	可选	string(256)	query	操作任务的状态。 • wait：等待 • processing：正在执行 • failed：失败 • success：成功 • stop：停止
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。 • 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

Status Code: 200 OK

参数	类型	描述
data	array<OperationInfo>	任务列表信息，具体参见 OperationInfo 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	enum<string>	操作类型。 • firmware_upgrade • software_upgrade
targets	OperateDevices	执行操作的目标设备，具体参见 OperateDevices 结构体。
policy	OperatePolicy	操作执行策略，具体参见 OperatePolicy 结构体。
status	enum<string>	操作任务的状态。 • wait：等待 • processing：正在执行 • failed：失败 • success：成功 • stop：停止
staResult	OperationStaResult	操作结果统计，具体参见 OperationStaResult 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	array<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	array<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。

参数	类型	描述
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。

请求示例

Method: GET

request:

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

Header:

"app_key: ******"

"Authorization:Bearer******"

Content-Type: application/json

Body:

{

"operationType":"*******",

"operationStatus":"*****",

"deviceType":"WaterMeter",

"model":"***",

"manufacturerName":"***",

"deviceId":"***"

}

响应示例

response:

Status Code: 200 OK

Content-Type:application/json

Body:

{

```
"totalCount": 1,
"pageNo": 0,
"pageSize": 1,
"date": [
{
"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
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。
400	123029	pageNo or pageSize beyond the limit.	参数 pageNo 或 pageSize 超出了限制范围。 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。

3 术语表

术语	全称	说明
Access Token	接入令牌	接入令牌包含了第三方应用接入 IoT 平台的安全信息，控制第三方应用可以访问 IoT 平台的哪些对象资源。
API	Application Programming Interface 应用程序编程接口	API 是指一些预先定义的函数，目的是提供应用程序与开发人员基于某软件或硬件得以访问一组例程的能力，而又无需访问源码或理解内部工作机制的细节。
IMEI	International Mobile Equipment Identity 国际移动设备标识	标识每一台 GSM/UMTS/LTE 手机终端的数字，具有唯一性。
IMSI	International Mobile Subscriber Identity 国际移动用户识别码	在 2/3/4G 网络中用于唯一识别移动用户的一个号码。这个号码通常被存放在 SIM 卡中，由手机发送给网络。它也可用于获取存储在本地位置寄存器（HLR）中的移动用户的其他信息，或复制在当地拜访位置寄存器中。
IoT	Internet Of Things 物联网	物联网是指将 IT 基础设施融入到物理基础设施中，具体地说，就是把感应器嵌入和装备到汽车、家庭、医院、电网、楼宇、铁路、桥梁、隧道、公路、建筑、供水系统、大坝、油气管道等各种物体中，并且被普遍连接，形成物联网。
NA	Network Application 网络应用、应用	NA 是一种在互联网或企业内部网上操作的应用软件，在 IoT 领域主要指智慧家庭、车联网、智慧医疗等行业应用。
PSK	Pre-Shared Key	由第三方应用设置或者 IoT 平台随机分配，是设备使用

术语	全称	说明
	预共享密钥	加密通道接入 IoT 的密钥。
SP Portal	Service Provider Portal	IoT 平台面向开发者用户的 Portal。通过 SP portal 可以对当前 SP 用户进行如下操作：应用管理（注册 NA）、资产（网关和传感器）管理、报表（ SP 用户下资产相关报表）管理、规则引擎（SP 编辑应用规则）、软件管理（资产的升级）。