**智能家居详细交互场景图**

## 整体流程

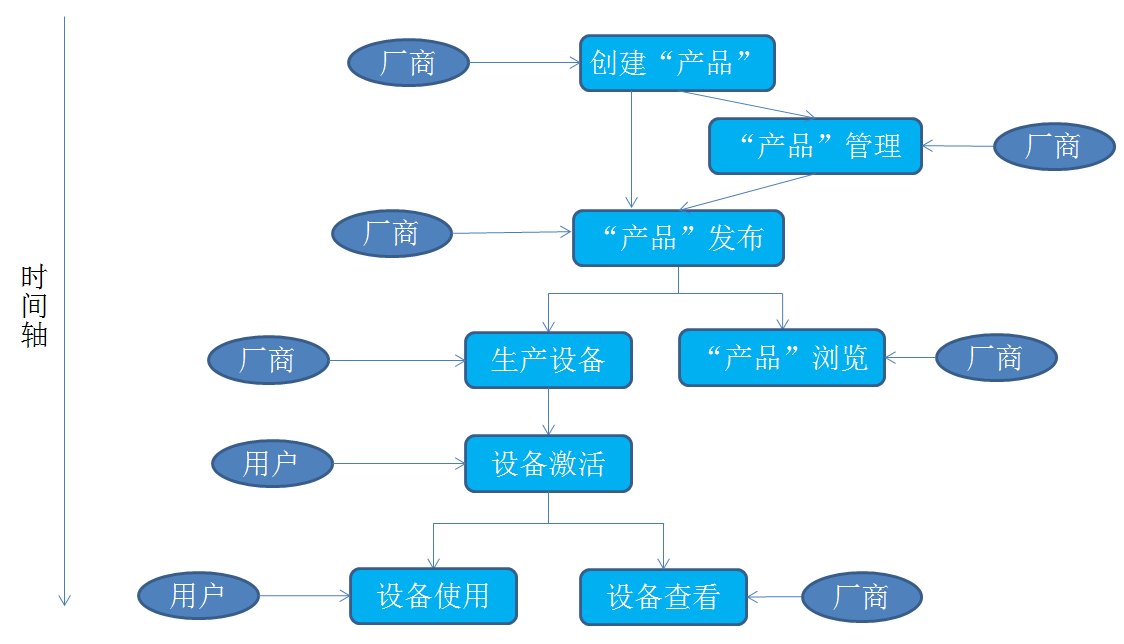


图1

图1所示为一批设备从生产到最终使用的流程场景图。

## 主要术语

1. **product\_id:**厂商在创建一个“产品”时，由京东平台分配，唯一标识一个“产品”。
2. **product\_secret:**与product\_id一起分配给厂商，作为以后设备认证秘钥。
3. **feed\_id:**京东平台对设备的唯一标识，与access\_key一起控制app与设备之间的数据交换。（由京东平台分配）
4. **access\_key:**用户/设备进行数据访问的权限，与feed\_id一起控制app与设备之间的数据交换。（由京东平台分配）
5. **stream\_id:**“产品”的属性识别id，由厂商自己定义。
6. **device\_id:**厂商对设备的唯一识别，由厂商自己确定，同一个“产品”中的device\_id不能有重复。

## 详细场景

### 一、创建“产品”（操作者：厂商）

设备厂商要将生产的设备接入京东平台，在生产设备前首先要在京东平台上创建一个“产品”；

|  |  |
| --- | --- |
| 请求方法 | POST |
| 请求URL | <https://api.smart.jcloud.com/v1/product> |
| 请求头 | JD-Key=USER\_KEY |
| 请求参数  (JSON) | {  "description" : "产品描述", //产品描述(可选)  "name" : "产品名称", //产品名称(必填)  "private" : "true", //不公开,true不公开，false公开(可选)，默认true  "streams" : [//产品属性(可选)  {  "stream\_id" : "xxx1", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx", //属性单位(可选)，如：Watts  "tags" : ""//产品tags,多个以半角逗号分隔(可选)  "symbol" : "xxx",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }, {  "stream\_id" : "xxx2", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "xxx",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }]  } |
| 返回状态码 | 200 CREATED |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息",  "data" : {  "product\_id" : "wr34dndafhllsssfd" //产品ID  “product\_secret”:”xxxxx”//产品秘钥  }  } |

### 二、“产品”管理（操作者：厂商）

#### 1)更新产品信息

不能更新product\_id和product\_secret ，会用本次信息覆盖以前信息

|  |  |
| --- | --- |
| 请求方法 | PUT |
| 请求URL | <https://api.smart.jcloud.com/v1/product/PRODUCT_ID> |
| 请求头 | JD-Key=USER\_KEY |
| 请求实体  (JSON) | { "description" : "产品描述", //产品描述(可选)  "name" : "产品名称", //产品名称(可选)  "private" : "true", //不公开,true不公开，false公开(可选)，默认true  "streams" : [//产品属性(可选)  {  "stream\_id" : "xxx", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx1", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }, {  "stream\_id" : "xxx2", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }]  } |
| 返回状态码 | 200 UPDADTED |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息",  } |

#### 2）删除产品

如果该“产品”中已经有了设备就不能删除

|  |  |
| --- | --- |
| 请求方法 | DELETE |
| 请求URL | <https://api.smart.jcloud.com/v1/product/PRODUCT_ID> |
| 请求头 | JD-Key= USER\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "错误信息",//code=200为OK,code!=200时为错误信息  } |

#### 3)查看产品

|  |  |
| --- | --- |
| 请求方法 | GET |
| 请求URL | <https://api.smart.jcloud.com/v1/product/PRODUCT_ID> |
| 请求头 | JD-Key= USER\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息",  "data" : {  “product\_id” : “xxx”,产品ID(必填)  "description" : "产品描述", //产品描述(可选)  "name" : "产品名称", //产品名称(可选)  "private" : "true", //不公开,true不公开，false公开(可选)，默认true  "streams" : [//产品属性(可选)  {  "stream\_id" : "xxx", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx1", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }, {  "stream\_id" : "xxx2", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }]  }  } |

#### 4）查看产品列表

|  |  |
| --- | --- |
| 请求方法 | GET |
| 请求URL | <https://api.smart.jcloud.com/v1/product> |
| 请求头 | JD-Key=USER\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "OK或错误信息",  "data" : [{//产品列表  “product\_id” : “xxx”,产品ID(必填)  "description" : "产品描述", //产品描述(可选)  "name" : "产品名称", //产品名称(可选)  "private" : "true", //不公开,true不公开，false公开(可选)，默认true  "streams" : [//产品属性(可选)  {  "stream\_id" : "xxx", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx1", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }, {  "stream\_id" : "xxx2", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }]  }, {  “product\_id” : “xxx”,产品ID(必填)  "description" : "产品描述", //产品描述(可选)  "name" : "产品名称", //产品名称(可选)  "private" : "true", //不公开,true不公开，false公开(可选)，默认true  "streams" : [//产品属性(可选)  {  "stream\_id" : "xxx", //属性ID(必填)  “stream\_type”:”0/1”,//stream的类型，0代表“控制型”流，1代表“感知型”流  "unit" : "xxx1", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "",//属性单位符号表示(可选),如：W  “value\_type”:”string/float/int”,//值类型  “max\_value”:”xx”,//可选，value最大值的说明  “min\_value”:”xx”,//可选，value最小值的说明  “value\_des”:{[{“val1”:”description1”},{“val2”:”description2”}]}//可选，当stream\_type=0时对value各种值的说明，例如，开关中“0”代表“关”，“1”代表“开”({[{“1”:”开”},{“0”:”关”}]})  }  ]  }]  } |

### 三、“产品”发布

厂家发布“产品”后，对产品不能进行任何修改。

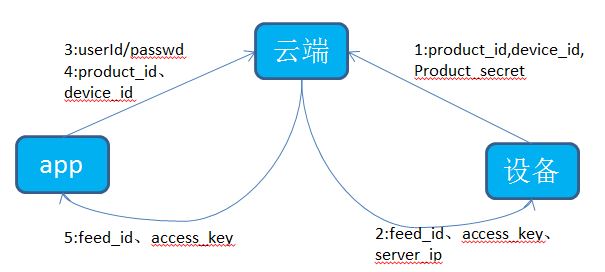
|  |  |
| --- | --- |
| 请求方法 | POST |
| 请求URL | <https://api.smart.jcloud.com/v1/product/PRODUCT_ID/release> |
| 请求头 | JD-Key= USER\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "错误信息",//code=200为OK,code!=200时为错误信息  } |

### 三、生产设备（操作者：厂商）

厂商将创建“产品”后得到的“product\_id”和“product\_secret”以及云端域名、端口烧录到硬件设备中。

### 四、设备使用（操作者：用户）

#### 1）设备激活



如上图所示：

1:设备在联网情况下，连接云端服务器，发送product\_id、product\_secret以及device\_id（厂商自己定义，属于同一个“产品”中的device\_id需要不同）到云端；

2：云端校验product\_id、product\_secret，如果正确，将device\_id存储到product\_id下，创建feed\_id（设备在云端的id）与access\_key（设备操作权限）并返回给设备。

|  |  |
| --- | --- |
| 请求方法 | POST |
| 请求URL | <https://api.smart.jcloud.com/v1/device/activate> |
| 请求头 | JD-Key=PRODUCT\_SECRET |
| 请求参数  (JSON) | {  "device\_id" : "设备ID", //设备ID，同一产品中必须唯一(必填)  "product\_id" : "产品ID", //产品ID，设备所属产品ID(必填)  } |
| 返回状态码 | 200 CREATED |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息"  “data”:{  “feed\_id”:”xxxx”，  “access\_key”:”xxx”,  “server\_ip”:[”ip:port”,”ip:port”,...],  “server\_time”:” 2013-04-22T01:35:43+08:00”//服务器时间  }  } |

3：用户使用自己的京东账号登录京东硬件平台；

4：使用已经激活的设备的product\_id、device\_id与自己的京东账号绑定（只有第一个绑定的账户能够成功，第一个绑定者可以分享给其他账户）

5：绑定成功后，京东硬件平台返回app feed\_id、access\_key（数据交互的标识与权限）

#### 3 ) 上传数据（使用者：app/device）

“上传数据”即可以是用户向设备发送的控制命令上传云端，也可以是设备将自己采集/状态数据上传云端。这两种模式的接口api一样。

注：“请求方法”=“PUT”发送控制命令；“请求方法”=“POST”创建新数据。

1. **创建一条或多条数据到每个stream**

|  |  |
| --- | --- |
| 请求方法 | POST |
| 请求URL | <https://api.smart.jcloud.com/v1/feeds/FEED_ID> |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | {  "streams" : [ {  "stream\_id" : "example",  "datapoints":[  {"at":"2013-04-22T00:35:43+08:00","value":"xx"},  {"at":"2013-04-22T00:55:43+08:00","value":"xx"},  {"at":"2013-04-22T01:15:43+08:00","value":"xx"},  {"at":"2013-04-22T01:35:43+08:00","value":"xx"}  ],  },  {  "stream\_id" : "key",  "datapoints":[  {"at":"2013-04-22T00:35:43+08:00","value":"xx"},  {"at":"2013-04-22T00:55:43+08:00","value":"xx"},  {"at":"2013-04-22T01:15:43+08:00","value":"xx"},  {"at":"2013-04-22T01:35:43+08:00","value":"xx"}  ],  },  {  "stream\_id" : "datastream",  "datapoints":[  {"at":"2013-04-22T00:35:43+08:00","value":"xx"},  {"at":"2013-04-22T00:55:43+08:00","value":"xx"},  {"at":"2013-04-22T01:15:43+08:00","value":"xx"},  {"at":"2013-04-22T01:35:43+08:00","value":"xx"}  ],  }  ] } |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息"  } |

1. **更新当前值到一个或多个stream**

|  |  |
| --- | --- |
| 请求方法 | PUT |
| 请求URL | <https://api.smart.jcloud.com/v1/feeds/FEED_ID> |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | [{  "stream\_id" : "id1",  "current\_value" : "333",  “at”:” 2013-04-22T01:35:43+08:00”  },{  "stream\_id" : "id2",  "current\_value" : "333",  “at”:” 2013-04-22T01:35:43+08:00”  }] |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息"  “data”:{JSONObject}  } |

#### 4）读取数据

可以是app读取设备当前数据，也可以是设备读取app命令。

**a)获取一个设备的某个stream当前值**

|  |  |
| --- | --- |
| 请求方法 | GET |
| 请求URL | https://api.smart.jcloud.com/v1/feeds/FEED\_ID/streams/STREAM\_ID |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息"  data:{  "stream\_id":"example",  "current\_value":"500",  "at":"2013-05-06T00:30:45+08:00"  }  } |

**b)获取一个设备的某个stream指定时间段的历史数据**

|  |  |
| --- | --- |
| 请求方法 | GET |
| 请求URL | <https://api.smart.jcloud.com/v1/feeds/FEED_ID/streams/STREAM_ID?start=yy-mm-dd> hh:mm:ss&end=yy-mm-dd hh:mm:ss |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息"  data:{  "stream\_id":"example",  "datapoints":[  {"at":"2013-04-22T00:35:43+08:00","value":"xx"},  {"at":"2013-04-22T00:55:43+08:00","value":"xx"},  {"at":"2013-04-22T01:15:43+08:00","value":"xx"},  {"at":"2013-04-22T01:35:43+08:00","value":"xx"}  ]  }  } |

**c)获取某设备所有stream的当前值**

|  |  |
| --- | --- |
| 请求方法 | GET |
| 请求URL | <https://api.smart.jcloud.com/v1/feeds/FEED_ID/streams> |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",//状态码  "error\_msg" : "OK或错误信息"  data:[{  "stream\_id":"id1",  "current\_value":"500",  "at":"2013-05-06T00:30:45+08:00"  },{  "stream\_id":"id2",  "current\_value":"500",  "at":"2013-05-06T00:30:45+08:00"  }]  } |

#### 5）设备信息查看

|  |  |
| --- | --- |
| 请求方法 | GET |
| 请求URL | <https://api.smart.jcloud.com/v1/device/FEED_ID> |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "OK或错误信息",  "data" : {  "device\_id" : "设备ID", //设备ID  "product\_id" : "产品ID", //产品ID  "name" : "设备名称", //设备名称  "private" : "true", //不公开,true不公开，false公开  "activate\_time": “yyyy-MM-dd HH:mm:ss”,//设备激活时间  “status”:”0/1”//1在线，0离线  "streams" : [//产品属性(可选)  {  "stream\_id" : "xxx1", //属性ID(必填)  "unit" : "xxx", //属性单位(可选)，如：Watts  "tags" : ""//产品tags,多个以半角逗号分隔(可选)  "symbol" : "xxx",//属性单位符号表示(可选),如：W  “value\_type”:”string/float”,//值类型  “max\_value”:”xx”,//可选，当”value\_type”=”float”时  “min\_value”:”xx”,//可选，当”value\_type”=”float”时  “description”:”xxx”//对value取值的描述，例如，开关中“0”代表“关”，“1”代表“开”  }, {  "stream\_id" : "xxx2", //属性ID(必填)  "unit" : "xxx", //属性单位(可选)，如：Watts  "tags" : "",//产品tags,多个以半角逗号分隔(可选)  "symbol" : "xxx",//属性单位符号表示(可选),如：W  “value\_type”:”string/float”,//值类型  “max\_value”:”xx”,//可选，当”value\_type”=”float”时  “min\_value”:”xx”,//可选，当”value\_type”=”float”时  “description”:”xxx”//对value取值的描述，例如，开关中“0”代表“关”，“1”代表“开”  }]  }  } |

#### 6）设备状态查看

|  |  |
| --- | --- |
| 请求方法 | GET |
| 请求URL | https://api.smart.jcloud.com/v1/device/FEED\_ID/status |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "OK或错误信息"  “data”:{“status”:”0/1”//1在线，0离线}  } |

### 五、 设备查看（操作者：厂商）

#### 1）获取已激活设备

|  |  |
| --- | --- |
| 请求方法 | GET |
| 请求URL | <https://api.smart.jcloud.com/v1/device/PRODUCT_ID?curr_page=xxx&page_size=xxx> |
| 请求头 | JD-Key=USER\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "OK或错误信息",  "data" : {  “total\_results”:xxx,//总条数  “page\_size”：xx,//每页条数  “curr\_page”:xxx//当前页号  “devices”:[{  "device\_id" : “xx1”, //设备ID  “activate\_time”: “yyyy-MM-dd HH:mm:ss”,//设备激活时间  },{ "device\_Id" : “xx2”, //设备ID  “activate\_time”: “yyyy-MM-dd HH:mm:ss”,//设备激活时间  } ]  } |

### 删除数据（操作者：未定）

#### 1）删除一条数据

|  |  |
| --- | --- |
| 请求方法 | DELETE |
| 请求URL | <https://api.smart.jcloud.com/v1/feeds/FEED_ID/streams/STREAM_ID/datapoints/TIMESTAMP> |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "OK或错误信息"  } |

#### 2）删除一个数据流

|  |  |
| --- | --- |
| 请求方法 | DELETE |
| 请求URL | <https://api.smart.jcloud.com/v1/feeds/FEED_ID/streams/STREAM_ID> |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "OK或错误信息"  } |

#### 3 ) 删除多条数据

|  |  |
| --- | --- |
| 请求方法 | DELETE |
| 请求URL | https://api.smart.jcloud.com/v1/feeds/FEED\_ID/streams/STREAM\_ID/datapoints?start=yy-mm-dd hh:mm:ss&end=yy-mm-dd hh:mm:ss |
| 请求头 | JD-Key=ACCESS\_KEY |
| 请求参数  (JSON) | 无 |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "OK或错误信息"  } |

### 刷新ACCESS\_KEY(操作者：设备)

使用场景：用户可以定期让设备刷新其access\_key，刷新完毕后，需要将新access\_key同步给app，防止设备被其他人一直控制

|  |  |
| --- | --- |
| 请求方法 | PUT |
| 请求URL | <https://api.smart.jcloud.com/v1/device/refresh> |
| 请求头 | JD-Key=PRODUCT\_SECRET |
| 请求参数  (JSON) | {  "device\_id" : "设备ID", //设备ID，同一产品中必须唯一(必填)  "product\_id" : "产品ID", //产品ID，设备所属产品ID(必填)  } |
| 返回状态码 | 200 OK |
| 返回JSON | {  "code" : "200",  "error\_msg" : "OK或错误信息"  “data”:{“access\_key”:xxx}  } |