

鑫方盛与各电商平台 对接 api 规范

文档修改记录

版本号	状态	修订说明	修订人	修订日期
1.0.0	创建	创建	星魂	2019-9-18
1.0.1	修改	获取商品详情接口新增字段 getSkuInfo: buy_limit_type revenue_code arrival_cycle parts_number length width height	刘入韶	2019-12-13
1.0.2	修改	5.2.3 接口新增 字段 retail_price 5.2.9 接口新增 字段 retail_price		
1.0.3	新增	新增消息类型 type=1 拆单		2020-04-04
1.0.4	新增	5.3.8 退货单查 询接口 5.2.3 新增 spu 名称字段		2020-04-27
1.0.5	新增	5.3.9 新增物流 信息查询接口 5.4.1 type=8、9 时，新增 orderId、 refundRole 的返 回字段		2020-05-06
1.0.6	新增	消息新增 type=18,物流信 息接口新增自有 物流下凭证信息		2020-05-14
1.0.7	新增	保存第三方订单 状态		2020-06-05
1.0.8	新增	5.2.3 接口新增 返回字段 tax_rate		2020-10-28

图例:

以下颜色主要用来区分新增/修改/着重/删除功能

文字描述	此颜色表示新增功能	
文字描述	此颜色表示修改功能	
文字描述	此颜色表示着重功能	
文字描述	删除线表示已经废弃	

1、前言

商城开放式接口 OpenAPI 主要供第三方平台系统调用，第三方平台对接云采系统使用，使用 OpenAPI 可以直接通过 HTTP 向其他业务系统提供业务接口调用；所有 API 均基于 Post 方式请求调用。

2、缩略语和术语定义

2.1 缩略语

本文中涉及到的缩略语解释如下表：

缩略语	英文全称	中文全称
SKU	Stock Keeping Unit	库存量单位
SPU	Standard Product Unit	标准产品单位

2.2 术语

本文档涉及到的术语解释如下表：

名词	解释
API 分类	是指提供的 API 按功能进行分类，主要分为：商品类、订单类等
API	是指系统间对接的文件和实时接口
API 参数	用于标识 API 的系统级和应用级的输入、输出参数定义

3、对接指南

对接之前需要做如下工作：

- ✓ 开发自身的业务系统。

- ✓ 获取 xfs 提供的外部业务系统的标识、key、密钥、授权等信息。
- ✓ 对接测试：外部业务系统与 xfs 系统进行接口联通性测试；包括接口报文格式、系统参数校验，但不进行业务数据格式校验及业务逻辑测试，所有的调用通过模拟成功返回。
- ✓ 业务测试：外部业务系统与网上商城系统进行对接测试，包括接口报文格式、系统参数校验、业务逻辑校验，进行全流程的业务贯通测试，包括网络对接。

4、对接技术

实时接口：

4.1 传输协议

- ✓ 采用 HTTP 传输协议，并采用 RESTFUL 风格的服务访问方式。
- ✓ 为了提供简单、规范化的技术实现，使用 POST 方式发起 API 请求（具体使用说明详见 5 章节接口服务）
- ✓ POST 把提交的数据则放置在 HTTP 包的包体中。
- ✓ 所有参数值需要转换为 UTF-8 编码。
- ✓ 支持 POST 请求方式的 API 接口，所需要拼接的 URL 格式示例如下：

```
https://{url}/{接口服务名称}?xxxxx
```

4.2 报文协议

4.2.1 post 请求方式

请求和响应报文消息均采用 JSON 表达。

POST 请求需要拼装 JSON 报文，报文消息使用 UTF-8 编码。

请求报文格式示例：

```
POST http://{url}/{接口服务名称} HTTP/1.1
```

```
Accept: application/json
```

```
Content-Type: application/json;charset=UTF-8
```

Content-Length: #JSON 消息长度

```
{
  "token": "x9lelkX",
  "sku": "cOKD31n20140101101937816x93z92"
}
```

响应报文格式示例:

HTTP/1.1 200 OK

Content-Type: application/json;charset=UTF-8

Content-Length: #JSON 字符串长度

```
{
  "XX": {
    ...
  }
}
```

4.2.1.1 参数说明

- 名称: 参数英文名称, 用于 HTTP-POST 方式 JSON 报文中节点的名称和 HTTP-GET 方式链接中的节点名称, 所有节点名称都小写。
- 类型: 标识此节点参数类型。

类型名称	格式	样例
boolean	true、false	true
datetime	YYYY-MM-DD HH:MM:SS 或其他时间格式	2014-01-01 16:17:24、20140101161724
timestamp	YYYY-MM-DD HH:MM:SS	2014-01-01 16:17:24
string	字符串	"手机"
int	整型数字	123123, 322342, 24234
Enum	枚举类型	"basic_service", "system_service"

*_entity	json 实体	{"phone_number":17092748163}
*_entity[]	json 实体集合	[{"phone_number":17092748163}, {"phone_number":17092748164}]

- 必填?: Y 为必填, N 为选填
- 备注: 为具体描述此参数的业务和取值范围。

4.2.1.2 公共参数

除获取 token 接口之外, 其他 API 接口都需要传递公共参数, 以保证接口数据可查

- 请求公共参数

名称	类型	必填?	说明	备注
appkey	String	Y	发起方 id	各电商为电子超市分配; 同 client_id
token	String	Y	鉴权信息	按照获取 token 接口获取
timeStamp	String	N	时间戳	格式为 yyyy-MM-dd HH:mm:ss
serialNumber	String	N	交易流水号	一次调用的唯一流水号, 20 位定长

- 返回公共参数

名称	类型	必填?	说明	备注
success	Boolean	Y	执行结果成功, 还是失败	true: 成功 false: 失败
resultCode	String	Y	错误码	成功: 0000 失败: 详见错误码 (可以使用自己的)
resultMessage	String	Y	错误描述	描述清楚, 什么问题
result	result_entity	Y	具体结果	每个接口都会单独进行说明

4.3 鉴权方式

第三方平台在调用 xfs 系统时, 需要先向 xfs 系统发起请求认证, 调用获取 token 信息, 之后才能访问 API; 在后续 API 调用时需要鉴权信息; 当鉴权信息失效或需要更换时, 必须重新进行认证。

5、接口定义

测试环境 url	https://opentest.fsyuncai.cn/yspostinterface
正式环境 url	https://openapi.fsyuncai.com/yspostinterface

5.0 鉴权类接口

5.0.1 获取 token 信息

5.0.1.1 应用场景

获取 token 鉴权值。

5.0.1.2 请求说明

API 名称	accessToken
调用入口	https://{url}/ accessToken
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.0.1.3 应用级输入参数

名称	类型	必填?	说明	备注
grant_type	String	Y	该值固定为 access_token	
client_id	String	Y	对接账号	由 xfs 事先分配
client_secret	String	Y	对接账号的密码	由 xfs 事先分配

timestamp	String	Y	当前调用时间	格式为 “2014-01-01 01:01:01”
username	String	Y	用户名	Username 如是中文, 在 url 参数里面, 需要将 username 使用 urf-8 格式进行 UriEncode 编码; username= client_id
password	String	Y	用户密码	md5 加密后的字符串 (32 位小写) password=md5(client_secret)

5.0.1.4 应用级返回结果

名称	类型	必填?	说明	备注
access_token	String	Y	Access_token 值	
refresh_token	String	N	Refresh_token 值	
time	String	Y	当前时间	
expires_in	Int	Y	Access_token 的过期时间, 秒级别, 有效期 24 小时	
refresh_token_expires	Int	N	refresh_token 的过期时间, 毫秒级别	

5.0.1.5 请求示例

入参:

```
{
  "grant_type":"access_token",
  "client_id":"x9lelkX",
  "client_secret":"sjd79and83cgem",
  "timestamp":"2017-08-10 01:01:01",
  "username":"guodianwssc",
```

```
"password":"cOKD31n20140101101937816x93z92 "  
  
}
```

出参:

```
{  
  
  "success":true,  
  
  "resultMessage":"true",  
  
  "resultCode":"0000",  
  
  "result":  
  
  {  
  
    "access_token ":"access_token",  
  
    "refresh_token ":"refresh_token",  
  
    "time":"2017-08-10 01:01:01",  
  
    "expires_in":86400,  
  
    "refresh_token_expires": 1397205488147  
  
  }  
  
}
```

5.1 基础数据接口

5.1.1 获取省市区县地址列表

5.1.1.1 应用场景

获取省市区县地址列表数据

5.1.1.2 请求说明

API 名称	getAddressListByLevel
调用入口	https://{url}/ getAddressListByLevel
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.1.1.3 应用级输入参数

名称	类型	必填?	说明	备注
page_size	Int	Y	分页大小。默认 10，最大 2000	
page_index	Int	Y	页号。默认为 1，即从第 1 页开始	
level	String	N	级别: 101-省(直辖市); 102-市(直辖市); 103-区县; 104-街道村镇',不传查询全部	
begin_time	String	N	请求时间戳: 格式为	2018-11-19 09:16:53

			yyyy-MM-dd HH:mm:ss, 数据的开始时间, 用于读取指定时间范围内的更新数据及增量数据	
end_time	String	N	请求时间戳: 格式为 yyyy-MM-dd HH:mm:ss, 数据的结束时间	2018-11-25 09:16:53

5.1.1.4 应用级返回结果

result{}

名称	类型	必填?	说明	备注
total_count	Int	Y	总页数	
page_size	Int	Y	返回的最大数据量	
page_index	Int	Y	返回的第几页的数据	
data	AddressInfo[]	Y	地址信息集合, 详见地址信息参数表	

地址信息:

名称	类型	必填?	说明	备注
code	String(24)	Y	区域编码	1
name	String(24)	Y	区域名称	北京
level	String(8)	Y	级别: 101-省(直辖市); 102-市(直辖市); 103-区县; 104-街道村镇'	101
parent_code	String(24)	Y	父级编码	1

5.1.1.5 请求示例

输入参数:

```
{  
  
  "appkey":"2019081312321231",  
  
  "timestamp":"2017-08-19 10:13:01",  
  
  "token":"7886f145d87cb5d3219e8291c225f9c8"  
  
  "page_size":10,  
  
  "page_index":1,  
  
  "level":101,  
  
  "begin_time":"2018-11-19 09:16:53",  
  
  "end_time":"2018-11-25 09:16:53"  
  
}
```

响应结果:

```
{  
  
  "success":true,  
  
  "resultMessage":"true",  
  
  "resultCode":"0000",  
  
  "result":  
  
  {  
  
    "data": [  
  
    {  
  
      "code": "7",
```

```
        "level": 101,

        "name": "河南",

        "parent_code": ""

    },

    {

        "code": "8",

        "level": 101,

        "name": "辽宁",

        "parent_code": ""

    }

]

}

}
```

5.1.2 获取下级地区列表数据

5.1.2.1 应用场景

获取省市区地址列表数据

5.1.2.2 请求说明

API 名称	getAddressDetail
调用入口	https://{url}/ getAddressDetail
HTTP 请求方式	HTTP-POST
格式	JSON

API 授权类型	需要授权
类型	基础服务

5.1.2.3 应用级输入参数

名称	类型	必填?	说明	备注
parent_code	String(24)	Y	父级编码	
begin_time	String	N	请求时间戳:格式为 yyyy-MM-dd HH:mm:ss, 数据的开始时间, 用于读取指定时间范围内的更新数据及增量数据	2018-11-19 09:16:53
end_time	String	N	请求时间戳:格式为 yyyy-MM-dd HH:mm:ss, 数据的结束时间	2018-11-25 09:16:53

5.1.2.4 应用级返回结果

result[]

名称	类型	必填?	说明	备注
code	String(24)	Y	区域编码	2801
name	String(24)	Y	区域名称	西城区
level	String(8)	Y	级别: 101-省(直辖市); 102-市(直辖市); 103-区县; 104-街道村镇'	
parent_code	String(24)	Y	父级编码	1

5.1.2.5 请求示例

输入参数:

```
{  
  
  "appkey":"2019081312321231",  
  
  "timestamp":"2017-08-19 10:13:01",  
  
  "token":"7886f145d87cb5d3219e8291c225f9c8"  
  
  "parent_code":10,  
  
  "begin_time":"2018-11-19 09:16:53",  
  
  "end_time":"2018-11-25 09:16:53"  
  
}
```

响应结果:

```
{  
  
  "success":true,  
  
  "resultMessage":"true",  
  
  "resultCode":"0000",  
  
  "result":  
  
  [  
  
    {  
  
      "code":"2801",  
  
      "name":"西城区",  
  
      "level":"102",  
  
      "parent_code":"1"  
  
    },  
  
    {
```

```
        "code": "2802",  
  
        "name": "东城区",  
  
        "level": "102",  
  
        "parent_code": "1"  
    }  
  
    ]  
  
}
```

5.2 商品相关接口

5.2.1 获取商品池编号

5.2.1.1 应用场景

获取 xfs 为第三方电商**指定**的商品池列表。

5.2.1.2 请求说明

API 名称	getCategoryList
调用入口	https://{url}/ getCategoryList
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.2.1.3 应用级输入参数

名称	类型	必填?	说明	备注
----	----	-----	----	----

appkey	String	Y		
timestamp	String	Y	当前调用时间	格式为 “2014-01-01 01:01:01”
token	String	Y		

5.2.1.4 应用级返回结果

result_entity[]

名称	类型	说明	备注
category_name	String(100)	商品池名称	Xfs 三级类
category_code	String(20)	商品池编码	

5.2.1.5 请求示例

输入参数:

```
{
  "appkey":"2019081312321231",
  "timestamp":"2017-08-19 10:13:01",
  "token":"7886f145d87cb5d3219e8291c225f9c8"
}
```

响应结果:

```
{
  "success":true,
  "resultMessage":"true",
  "resultCode":"0000",
}
```

```
"result":  
  
  {  
  
    " category_name": "塑铜线",  
  
    " category_code": "010101"  
  
  },  
  
  {  
  
    " category_name": "阻燃塑铜线",  
  
    " category_code": "010102"  
  
  },  
  
  {  
  
    " category_name": "耐火塑铜线",  
  
    " category_code": "010103"  
  
  }  
  
}
```

5.2.2 获取池内商品编号

5.2.2.1 应用场景

通过池编码获取商品池内的 SKU 编码。

5.2.2.2 请求说明

API 名称	getSkuByCategoryCode
调用入口	https://{url}/ getSkuByCategoryCode
HTTP 请求方式	HTTP-POST

格式	JSON
API 授权类型	需要授权
类型	基础服务

5.2.2.3 应用级输入参数

名称	类型	必填?	说明	备注
category_code	String	Y	池编码	
page_number	Int	Y	页码	默认取第一页；每页最多 1000 条数据，品类商品池可能存在多页数据，具体根据返回的页总数判断是否有下一页数据

5.2.2.4 应用级返回结果

名称	类型	必填?	说明	备注
page_count	Int	Y	总页数	
sku_codes	String[]	Y	Sku 编码列表	

5.2.2.5 请求示例

输入参数:

```
{
  "appkey":"2019081312321231",
  "timestamp":"2017-08-19 10:13:01",
  "token":"7886f145d87cb5d3219e8291c225f9c8"
  "category_code":" 010101",
  "page_number":1
}
```

响应结果:

```
{
  "success":true,
  "resultMessage":"true",
  "resultCode":"0000",
  "result":
  {
    "page_count":1,
    "sku_codes":["1216536","1331125","1378528","1456835","1593512","1593516"]
  }
}
```

5.2.3 获取商品详细信息

5.2.3.1 应用场景

通过 sku 编码查询 sku 的详细信息，包括商品名称、品牌、上下架状态、商品介绍等信息。

5.2.3.2 请求说明

API 名称	getSkuInfo
调用入口	http://{url}/ getSkuInfo
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.2.3.3 应用级输入参数

名称	类型	必填?	说明	备注
sku_codes	String[]	Y	商品编号列表	最多可查 100 个

5.2.3.4 应用级返回结果

result_entity[]

名称	类型	说明	备注
sku_code	String(20)	商品编号	
spu_code	String(20)	Spu 编码	
spu_name	String(20)	Spu 名称	
subtitle	String(150)	商品副标题	
detail	longtext	商品详情	Html 页面可以直接展示
specifications	String(100)	规格	
color	String(12)	颜色	没有则为 null
weight	BigDecimal(16,6)	重量	单位: kg
volume	BigDecimal(16,6)	体积	
image_path	String(100)	主图地址	
sales_status	Int(2)	上下架状态	1: 上架 0: 下架
brand_name	String(40)	品牌名称	
brand_logo	String(1000)	品牌图片	品牌图片地址
sku_name	String(250)	商品名称	Xfs 名称+颜色+规格
origin	String(20)	产地	
unit_name	String(8)	销售单位	
category_code	String(20)	类目编码	
mini_order	Double(12,2)	最小起订量	

restricted	Int(2)	是否限制类型	0 非限制类, 1 危化品, 2 管制刀具 (1/2 是限制类)
calculation_price	Double(12,2)	结算价	保留 2 位小数
buy_limit_type	int(2)	购买限制类型	(0 起订量的正整数倍, 1 不低于起订量的正整数)
revenue_code	String(50)	税类编码	
arrival_cycle	int(11)	到货周期	
parts_number	String(50)	件装数量	
length	double(16,6)	长	单位: cm
width	double(16,6)	宽	单位: cm
height	double(16,6)	高	单位: cm
retail_price	Double(12,2)	售价	保留 2 位小数
tax_rate	Double(12,2)	税率	百分比对应数字, 示例: 13 (代表税率 13%)

5.2.3.5 请求示例

输入参数:

```
{
  "appkey":"201912090841588",
  "timestamp":"2019-10-23 13:20:01",
  "token":"4ef08c3a502cebc6f11bf7b5c8a1faf",
  "sku_codes":["016351","072307"],
  "page_number":1
}
```

响应结果:

```
{
  "success": true,
  "resultCode": "0000",
  "resultMessage": "true",
  "result": [
    {
      "color": "红",
      "origin": "大兴",
      "brand_logo":
"https://fsyuncai.oss-cn-beijing.aliyuncs.com/product/brandlogo/tiantan_138-46.jpg?x-oss-proc
ess=style/marking_text_20",
      "specifications": "10 平方",
      "order_limit_type": 1,
      "parts_number": "1",
      "sales_status": 1,
      "mini_order": 1.0,
      "subtitle": "装修用电线，国标塑铜线。",
      "sku_name": "天坛 BV 塑铜线 95m",
      "revenue_code": "10904090200000000000",
      "arrival_cycle": 7,
```



```
=style/marking_detail\"/><br/><img  
src=\"https://fsyuncai.oss-cn-beijing.aliyuncs.com/product/detail/016895_02.jpg?x-oss-process  
=style/marking_detail\"/><br/></p>\",  
  
    \"sku_code\": \"016351\"  
  },  
  
  {  
  
    \"color\": null,  
  
    \"origin\": \"衢州\",  
  
    \"brand_logo\":  
  
\"https://fsyuncai.oss-cn-beijing.aliyuncs.com/product/brandlogo/hongwuhuan.jpg?x-oss-proces  
s=style/marking_text_20\",  
  
    \"specifications\": \"W-1.65/14kg/380V/15KW\",  
  
    \"order_limit_type\": 1,  
  
    \"parts_number\": \"1 台\",  
  
    \"sales_status\": 0,  
  
    \"mini_order\": 1.0,  
  
    \"subtitle\": \"\",  
  
    \"sku_name\": \"红五环 空压机 HW20012A\",  
  
    \"revenue_code\": \"10901180000000000000\",  
  
    \"arrival_cycle\": 0,  
  
    \"height\": 115.0,  
  
    \"spu_code\": \"103058\",
```

```
        "category_code": "200201",

        "length": 130.0,

        "weight": 590.000000,

        "brand_name": "红五环",

        "volume": 1.719250,

        "unit_name": "台",

        "image_path":

"https://fsyuncai.oss-cn-beijing.aliyuncs.com/product/images/003770d.jpg?x-oss-process=style

/marking_text",

        "restricted": 0,

        "calculation_price": 9900.0,

        "width": 115.0,

        "detail": "",

        "sku_code": "072307"

    }

]

}
```

5.2.4 获取商品上下架状态

5.2.4.1 应用场景

通过 sku 编码该 SKU 当前的上下架状态。

5.2.4.2 请求说明

API 名称	skuSaleStatus
调用入口	https://{url}/ skuSaleStatus
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.2.4.3 应用级输入参数

名称	类型	必填?	说明	备注
sku_codes	String []	Y	商品编号	支持批量; 最大值 1000
city	String	Y	二级地址编码	

5.2.4.4 应用级返回结果

result[]

名称	类型	说明	备注
sku_code	String(20)	商品编号	
state	Int(2)	商品状态	1: 上架 0: 下架

5.2.4.5 请求示例

请求参数:



```
{  
  
  "appkey":"2019061998302935",  
  
  "timestamp":"2019-10-23 13:20:01",  
  
  "token":"goh1vaepecuwae7Oiquoo2Weuwai",  
  
  "sku_codes":["017396","017394","017395","017392","017393"],  
  
  "city":1  
  
}
```

响应结果:

```
{  
  
  "success":true,  
  
  "resultMessage":"true",  
  
  "resultCode":"0000",  
  
  "result":  
  
  [  
  
    {  
  
      "sku_code":"4255661",  
  
      "state":1  
  
    }  
  
  ]  
  
}
```

5.2.5 获取商品图片信息

5.2.5.1 应用场景

通过 sku 编码批量查询该 sku 的所有图片信息。

5.2.5.2 请求说明

API 名称	skuImageByCode
调用入口	https://{url}/ skuImageByCode
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.2.5.3 应用级输入参数

名称	类型	必填?	说明	备注
sku_codes	String[]	Y	商品编号	支持批量，以，分隔；

5.2.5.4 应用级返回结果

result[]

名称	类型	必填?	说明	备注
sku_code	String(20)	Y	商品编号	
skuPic	skuPic_entity[]	Y	商品图片	

skuPic_entity[]

名称	类型	说明	备注
img_url	String(100)	图片地址	图片地址
order_sort	Int	排序	排序数字 (1,2,3,4,5)
isPrimary	Int	是否主图	isPrimary =1 代表主图

5.4.5.5 请求示例

请求参数:

```
{  
  
  "appkey":"CGDC",  
  
  "timestamp":"2017-08-19 10:13:01",  
  
  "token":"7886f145d87cb5d3219e8291c225f9c8",  
  
  "sku_codes":["017396","017394","017395","017392","017393"] }
```

响应结果:

```
{  
  
  "success":true,  
  
  "resultMessage":"true",  
  
  "resultCode":"0000",  
  
  "result":  
  
    [{  
  
      "sku_code":"4255661",  
  
      "skuPic":  
  
        [{  
  
          "path":"7393/ca49c783-e1bb-418f-8063-47a6d20158ca.jpg",  
  
          "isPrimary":1,  
  
          "orderSort":0  
  
        }],  
  
      {
```

```

        "path":"7393/ca49c783-e1bb-418f-8063-47a6d20158ca.jpg",

        "isPrimary":0,

        "orderSort":1

    ]

}

}

}

```

5.2.6 商品区域购买限制查询

5.2.6.1 应用场景

通过 sku 编码和地址查询该 sku 在该地址上是否可以购买。

5.2.6.2 请求说明

API 名称	checkSkuAreaLimit
调用入口	https://{url}/ checkSkuAreaLimit
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.2.6.3 应用级输入参数

名称	类型	必填?	说明	备注
sku_codes	String[]	Y	商品编号	支持批量, 以, 分隔
province	String	Y	一级地址编号	xfS 地址
city	String	Y	二级地址编号	xfS 地址
county	String	N	三级地址编号	如无传 0
town	String	N	四级地址编号	如无传 0

6.2.6.4 应用级返回结果

result[]

名称	类型	说明	备注
sku_code	String	商品编号	
isAreaRestrict	boolean	True>false	true 代表区域受限 false 区域不受限

6.2.6.5 请求示例

请求参数:

```
{
  "appkey": "CGDC",
  "serialnumber": "sjd79and83cgem",
  "timestamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "sku_codes": ["017396", "017394", "017395", "017392", "017393"],
  "province": 1,
  "city": 1,
  "county": 0,
  "town": 0
}
```

响应结果:

```
{
  "success": true,
  "resultCode": "0000",
}
```

```
"resultMessage": "true",

"result": [

  {

    "isAreaRestrict": false,

    "sku_code": "017392"

  },

  {

    "isAreaRestrict": false,

    "sku_code": "017393"

  },

  {

    "isAreaRestrict": false,

    "sku_code": "017394"

  },

  {

    "isAreaRestrict": false,

    "sku_code": "017395"

  },

  {

    "isAreaRestrict": false,

    "sku_code": "017396"

  }

]
```

```

    ]
}

```

5.2.7 查询分类列表信息

5.2.7.1 应用场景

查询类目列表信息。（供贵司做类目映射使用）

5.2.7.2 请求说明

API 名称	getCategoryInfoList
调用入口	https://{url}/ getCategoryInfoList
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.2.7.3 应用级输入参数

名称	类型	必填?	说明	备注
parent_category_code	String	N	父级类目编码	Xfs 三级类目结构；根据父类编码查询子集类目信息； 不传，则查询所有一级类目；

5.2.7.4 应用级返回结果

categorys_entity[]

名称	类型	说明	备注
category_code	String(20)	类目编码	
parent_category_code	String (20)	父分类编码	如无返回 0
category_name	String(20)	分类名称	

category_level	Int(2)	分类级别	1: 一级分类; 2: 二级分类; 3: 三级分类;
state	Int	类目状态	0: 有效; 1: 无效;

5.2.7.5 请求示例

请求参数:

```
{  
  
  "appkey": "CGDC",  
  
  "serialnumber": "sjd79and83cgem",  
  
  "timestamp": "2017-08-19 10:13:01",  
  
  "token": "7886f145d87cb5d3219e8291c225f9c8",  
  
  "cid": "670"  
  
}
```

响应结果:

```
{  
  
  "success": true,  
  
  "resultMessage": "true",  
  
  "resultCode": "0000",  
  
  "result":  
  
  {  
  
    "catId": "670",  
  
    "parentId": "0",  
  
    "catClass": 0,  
  
  }  
  
}
```

```

    "state":1
  }
}

```

5.2.8 查询分类信息

5.2.8.1 应用场景

通过类目 ID 查询类目的详细信息和状态。

5.2.8.2 请求说明

API 名称	getCategoryInfo
调用入口	http://{url}/ getCategoryInfo
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

1.1.1.1. 应用级输入参数

名称	类型	必填?	说明	备注
category_code	String	Y	分类编码	

1.1.1.2. 应用级返回结果

categorys_entity[]

名称	类型	说明	备注
category_code	String(20)	类目编码	
parent_category_code	String (20)	父分类编码	如无返回 0
category_name	String(20)	分类名称	
category_level	Int(2)	分类级别	1: 一级分类;

			2: 二级分类; 3: 三级分类;
state	Int	类目状态	1: 有效; 0: 无效;

5.2.9 按地区批量查询商品价格

5.2.9.1 应用场景

根据地区和 sku 编码批量查询价格。

5.2.9.2 请求说明

API 名称	getPriceByArea
调用入口	https://{url}/ getPriceByArea
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.9.2.3 应用级输入参数

名称	类型	必填?	说明	备注
city	String	Y	二级地址编号	
sku_codes	String[]	Y	商品编号, 请以, (英文逗号)分割	

5.9.2.4 应用级返回结果

result[]

名称	类型	说明	备注
sku_code	String(20)	SKU 编码	
calculation_price	Double(12,2)	结算价	保留 2 位小数

retail_price	Double(12,2)	售价	保留 2 位小数
--------------	--------------	----	----------

- 不在的商品池中的 SKU 不返回价格
- 价格为 null 或者小于 0 时，为暂无报价

5.2.9.5 请求示例

请求参数:

```
{  
  
  "appkey": "2019061998302935",  
  
  "timestamp": "2019-10-23 13:20:01",  
  
  "token": "goh1vaepucuwae7Oiquoo2Weuwai",  
  
  "city": "12",  
  
  "sku_codes": [ "017396", "017394", "017395", "017392", "017393"]  
  
}
```

响应结果:

```
{  
  
  "success": true,  
  
  "resultCode": "0000",  
  
  "resultMessage": "true",  
  
  "result": [  
  
    {
```

```
        "calculation_price": 13.2,

        "sku_code": "017392"

    },

    {

        "calculation_price": 19.1,

        "sku_code": "017393"

    },

    {

        "calculation_price": 33.2,

        "sku_code": "017394"

    },

    {

        "calculation_price": 52.2,

        "sku_code": "017395"

    },

    {

        "calculation_price": 80.9,

        "sku_code": "017396"

    }

]

}
```

5.2.10 批量查询商品库存

5.2.10.1 应用场景

根据 sku 编码和地址批量查询商品库存，用于展现商品的库存信息。库存=实际库存+虚拟库存

5.2.10.2 请求说明

API 名称	getStockBySkuCode
调用入口	https://{url}/ getStockBySkuCode
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.2.10.3 应用级输入参数

名称	类型	必填?	说明	备注
sku_nums	skuNums_entity[]	Y	商品和数量	
area	String	Y	区域信息	式: 1_0_0 (分别代表 1、2、3 级地址)

skuNums_entity[]

名称	类型	必填?	说明	备注
sku_code	String	Y	商品编码	
num	int	Y	数量	购买数量

5.2.10.4 应用级返回结果

result[]

名称	类型	必填?	说明	备注
area	String	Y	配送地址 id	式: 1_0_0 (分别代表 1、2、3 级地址)
sku_code	String	Y	商品编号	
stock_state_Id	Int	Y	库存状态编号	33 有货 34 无货
remain_num	Int	Y	剩余数量	-1 未知; 当库存小于 50 时展示真实库存数量

注意: 有货和无货需要根据用户设置的数量和地区进行判断

5.2.10.5 请求示例

请求参数:

```
{
  "success": true,
  "resultCode": "0000",
  "resultMessage": "true",
  "result": [
    {
      "area": "1_0_0",
      "stockStateld": 33,
      "remain_num": -1,
      "sku_code": "016010"
    }
  ],
}
```

```
{  
  "area": "1_0_0",  
  "stockStatId": 33,  
  "remain_num": -1,  
  "sku_code": "039203"  
}  
]  
}
```

响应结果:

```
{  
  "success": true,  
  "resultCode": "0000",  
  "resultMessage": "true",  
  "result": [  
    {  
      "area": "1_0_0",  
      "stockStatId": 33,  
      "remain_num": -1,  
      "sku_code": "016010"  
    },  
  ],  
}
```

```
{  
  
  "area": "1_0_0",  
  
  "stockStatId": 33,  
  
  "remain_num": -1,  
  
  "sku_code": "039203"  
  
}  
  
]  
  
}
```

5.3 订单相关接口

5.3.1 预订单下单接口

5.3.1.1 应用场景

预订单下单接口，下单成功后，会锁定订单价格，商品价格和运费等信息，等待确认或取消。

5.3.1.2 请求说明

API 名称	orderSubmit
调用入口	https://{url}/ orderSubmit
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.3.1.3 应用级输入参数

名称	类型	必填?	说明	备注
project_id	String(32)	Y	对接的项目编码	
out_trade_no	String(32)	Y	贵方交易单号	
channel_customer_name	String(64)	N	下单客户名称	北京 XX 公司
created_at	String(19)	Y	订单创建时间，格式"yyyy-MM-dd HH:mm:ss"	2019-06-09 01:01:20
email	String(50)	N	客户邮箱	
phone	String(20)	N	客户固话	
order_remark	String(120)	N	订单备注	
out_total_amount	double(16,2)	Y	订单总金额。订单总金额=商品总金额+运费金额。单位为元，精确到小数点后两位	21238.10
out_freight	double(16,2)	Y	订单运费，单位为元，精确到小数点后两位	168.10
province_code	String(16)	Y	省份编码	必传，传错会导致下单失败
city_code	String(16)	Y	市编码	必传，传错会导致下单失败
area_code	String(16)	Y	区编码	必传，传错会导致下单失败
town_code	String(16)	Y	县编码	必传，如无传 0
detail_address	String(100)	Y	详细地址	海鑫路 19 号
contacters	contacters[]	Y	收货人信息	
goods_detail	goodsDetail[]	Y	订单包含的	

			商品信息	
--	--	--	------	--

收货人信息

名称	类型	必填?	说明	备注
receiver_name	String(24)	Y	收货人姓名	张三
telephone	String(16)	Y	联系电话	15601234567

商品信息

名称	类型	必填?	说明	备注
goods_id	String(23)	Y	商品编码	049572
quantity	Int(10)	Y	购买数量	2
price	Double(16,2)	Y	商品结算单价，单位为元，精确到小数点后两位	12340.10
sale_price	Double(16,2)	N	贵方商品售价	12360.32
unit	String(16)	Y	计量单位	件

5.3.1.4 应用级返回结果

result{}

名称	类型	必填?	说明	备注
third_order	String(32)	Y	第三方订单号	201908230001
order_id	String(32)	Y	云采订单号	18102207769
freight	Double(16,2)	Y	我方计算出的订单总运费，单位为元，精确到小数点后两位	100.20
order_price	Double(16,2)	Y	我方订单商品总金额，单位为元，精确到小数点后两位	32300.20

sku	sku[]			
-----	-------	--	--	--

商品信息

名称	类型	必填?	说明	备注
sku_id	String(23)	Y	商品编码	02941
num	Int(10)	Y	购买数量	2
price	Double(16,2)	Y	商品结算价	123.20
name	String(32)	Y	商品名称	

5.3.1.5 请求示例

请求参数:

```
{  
  
  "appkey": "CGDC",  
  
  "serialnumber": "sjd79and83cgem",  
  
  "timestamp": "2017-08-19 10:13:01",  
  
  "token": "7886f145d87cb5d3219e8291c225f9c8",  
  
  "project_id": "1142076454747712",  
  
  "out_trade_no": "201909160023",  
  
  "created_at": "2019-09-16 16:53:53",  
  
  "out_total_amount": 4309,  
  
  "out_freight": 150,  
  
  "province_code": "1",  
  
  "city_code": "2801",
```

```
"area_code": "4205",

"town_code": "0",

"detail_address": "测试详细收货地址",

"contacters": [{

    "receiver_name": "测试",

    "telephone": "18628209573"

}],

"goods_detail": [{

    "goods_id": "034344",

    "quantity": 1,

    "price": 4159

}]

}
```

响应结果:

```
{

    "success":true,

    "resultMessage":"true",

    "resultCode":"0000",

    "result":

        {

            "third_order": "201909160023",

            "order_id": "19106714105",
```

```
    "freight": 50.0,

    "order_price": 4159.0,

    "sku": [

        {

            "sku_id": "034344",

            "num": 1,

            "price": 4159.0,

            "name": "开平板"

        }

    ]

}
```

5.3.2 确认预订单接口

Result{}

5.3.2.1 应用场景

预订单下单后，需要确认订单，才会变成正式订单。

5.3.2.2 请求说明

API 名称	orderConfirm
调用入口	https://{url}/orderConfirm
HTTP 请求方式	HTTP-POST
格式	JSON

API 授权类型	需要授权
类型	基础服务

5.3.2.3 应用级输入参数

名称	类型	必填?	说明	备注
order_id	String(32)	Y	云采订单号	18102207769

5.3.2.4 应用级返回结果

result{}

5.3.2.5 请求示例

请求参数:

```
{
  "appkey":"CGDC",
  "serialnumber":"sjd79and83cgem",
  "timestamp":"2017-08-19 10:13:01",
  "token":"7886f145d87cb5d3219e8291c225f9c8",
  "order_id":"18102207769"
}
```

响应结果:

```
{
  "success":true,
  "resultMessage":"true",
}
```

```

    "resultCode":"0000",

    "result":{}

}

```

5.3.3 取消订单接口

5.3.3.1 应用场景

~~预订单下单后，通过调用此接口进行取消。已确认订单无法通过此接口取消。~~

场景一：第三方发起取消

1> 订单状态为待确认（第三方确认）、待审核（xfs 审核）状态时，对方取消，实时返回成功/失败

场景二：xfs 发起

1> 订单状态为待确认（第三方确认）、待审核（xfs 审核）状态时，通过消息接口将订单状态交易关闭传递给第三方

5.3.3.2 请求说明

API 名称	orderCancel
调用入口	https://{url}/orderCancel
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.3.3.3 应用级输入参数

名称	类型	必填?	说明	备注
order_id	String(32)	Y	云采订单号	18102207769

5.3.3.4 应用级返回结果

```
result{}
```

5.3.3.5 请求示例

请求参数:

```
{  
  
  "appkey":"CGDC",  
  
  "serialnumber":"sjd79and83cgem",  
  
  "timestamp":"2017-08-19 10:13:01",  
  
  "token":"7886f145d87cb5d3219e8291c225f9c8",  
  
  "order_id":"18102207769"  
  
}
```

响应结果:

```
{  
  
  "success":true,  
  
  "resultMessage":"true",  
  
  "resultCode":"0000",  
  
  "result":{}  
  
}
```

5.3.4 查询运费接口

5.3.4.1 应用场景

贵方在下订单之前，可调用此接口查询我方生成的运费

5.3.4.2 请求说明

API 名称	queryFreight
调用入口	https://{url}/queryFreight
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.3.4.3 应用级输入参数

名称	类型	必填?	说明	备注
project_id	String(32)	Y	对接的项目编码	18102207769
province_code	String(24)	Y	省份编码	必传, 传错会导致查询失败
city_code	String(24)	Y	市编码	必传, 传错会导致查询失败
area_code	String(24)	Y	区编码	必传, 如无传 0
town_code	String(24)	Y	县编码	必传, 如无传 0
goods_detail	goods_detail[]	Y	商品信息。详见：“商品信息”	

商品信息

名称	类型	必填?	说明	备注
goods_id	String(23)	Y	商品编码	049572

quantity	Int(10)	Y	购买数量	2
price	Double(16,2)	Y	商品结算单价，单位为元，精确到小数点后两位	10.10

5.3.4.4 应用级返回结果

result{}

名称	类型	必填?	说明	备注
freight_type	Int(2)	Y	运费类型 10=免运费； 20=阶梯运费； 30=云采运费； 40=没有支持的配送方式	20
freight	Double(16,2)	Y	运费金额	124.20

5.3.4.5 请求示例

请求参数:

```
{
  "appkey": "CGDC",
  "serialnumber": "sjd79and83cgem",
  "timestamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "project_id": "1234",
  "province_code": "1",
  "city_code": "2801",
  "area_code": "4205",
```

```
"town_code":"0",

"goods_detail":[

  {

    "goods_id":123456,

    "quantity":2,

    "price":2.30

  },{

    "goods_id":016351,

    "quantity":1,

    "price":2.30

  },{

    "goods_id":084578,

    "quantity":2,

    "price":2.34

  }

]

}
```

响应结果:

```
{

  "success":true,

  "resultMessage":"true",

  "resultCode":"0000",
```

```
"result":  
  
  {  
  
    "freight_type":"10",  
  
    "freight":"120.00"  
  
  }  
  
}
```

5.3.5 查询订单详情接口

5.3.5.1 应用场景

根据我方订单号查询订单详情信息

5.3.5.2 请求说明

API 名称	queryOrderDetail
调用入口	https://{url}/ queryOrderDetail
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.3.5.3 应用级输入参数

名称	类型	必填?	说明	备注
order_id	String(32)	Y	云采订单号	18102207769

5.3.5.4 应用级返回结果

result{}

名称	类型	必填?	说明	备注
order_id	String(32)	Y	云采订单号	18102207769
third_order	String(32)	Y	第三方订单号	
order_state	Int	Y	订单状态: 0 已取消 (调用过取消订单接口) 1 有效	
submit_state	Int	Y	0 未确认订单 1 已确认订单	
order_status	Int	Y	云采订单状态: 20=> 待 审 核 ;30=> 待 配 货 ;40=> 待 出 库 ;50=> 待 发 货 ;60=> 待 收 货 ;70=> 待 结 算 ;80=> 订 单 成 功 ;90=> 交 易 完 成 ;100=> 订 单 取 消 ;110=> 交 易 关 闭	
order_price	Double(16,2)	Y	订单商品总金额,保留两位小数	14000.40
freight	Double(16,2)	Y	订单的运费价格,保留两位小数	50.47
sku	sku[]	Y	订单商品信息, 详见商品信息参数	

商品信息

名称	类型	必填?	说明	备注
sku_id	String(32)	Y	商品编码	029145
name	String(64)	Y	商品名称	
num	int	Y	购买数量	
price	Double(16,2)	Y	商品价格	

5.3.5.5 请求示例

请求参数:

```
{
  "appkey": "CGDC",
  "serialnumber": "sjd79and83cgem",
  "timestamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "order_id": "123412345"
}
```

响应结果:

```
{
  "success": true,
  "resultMessage": "true",
  "resultCode": "0000",
  "result": {
    "third_order": "1234123123",
    "order_id": "19990123123",
    "order_state": "1",
    "submit_state": "1",
    "order_status": "30",
    "order_price": "120.10",
  }
}
```

```
    "freight": "10.10",  
  
    "sku": [  
  
        "sku_id": "02951",  
  
        "num": "3",  
  
        "price": "23.40",  
  
        "name": "重工机电机电"  
  
    ]  
  
    }  
  
}
```

5.3.6 订单确认收货接口

5.3.6.1 应用场景

根据我方订单号确认订单已经收货

5.3.6.2 请求说明

API 名称	completion
调用入口	https://{url}/ completion
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.3.6.3 应用级输入参数

名称	类型	必填?	说明	备注
order_id	String(32)	Y	云采订单号	18102207769

5.3.6.4 应用级返回结果

```
result{}
```

5.3.6.5 请求示例

请求参数:

```
{
  "appkey":"CGDC",
  "serialnumber":"sjd79and83cgem",
  "timestamp":"2017-08-19 10:13:01",
  "token":"7886f145d87cb5d3219e8291c225f9c8",
  "order_id":"18102207769"
}
```

响应结果:

```
{
  "success":true,
  "resultMessage":"true",
  "resultCode":"0000",
  "result":{}
}
```

5.3.7 订单申请退货接口

5.3.7.1 应用场景

根据我方订单号和商品信息等，申请退货

5.3.7.2 请求说明

API 名称	orderRefund
调用入口	https://{url}/ orderRefund
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.3.7.3 应用级输入参数

名称	类型	必填?	说明	备注
order_id	String(32)	Y	云采订单号	18102207769
received_name	String(80)	Y	收货人姓名	
received_telephone	String(20)	Y	收货人手机号	
return_reason	int	Y	退货原因	10=商品质量问题； 60=少件/漏发； 70=多买/买错不想要了； 80=到货物流破损； 90=发票问题； 100=未按约定时间送货； 110=其它原因；

				120=收到商品与网站展示不符; 130=客户原因无理由退货; 140=资质问题;
return_desc	String(500)	N	退货原因描述	
province_code	String(8)	Y	xfS 省编码	
city_code	String(8)	Y	xfS 市编码	
area_code	String(8)	Y	xfS 区编码	为空传 0
town_code	String(8)	Y	xfS 县编码	为空传 0
take_goods_type	int	Y	退货方式	10=上门取货; 20=客户邮寄
detail_address	String(255)	Y	详细地址	
goods_detail	List	Y	退货商品信息集合	
proof_pic	List	N	退货证明图片集合, 传入最多 10 张图片地址, 图片支持 bmp、gif、jpg、png、jpeg 格式文件	

退货商品参数

名称	类型	必填?	说明	备注
sku_code	String(20)	Y	商品 code	020080
refund_count	int	Y	退货数量	20

退货证明图片参数

名称	类型	必填?	说明	备注
pic_url	String(255)	Y	图片地址 url	

5.3.7.4 应用级返回结果

```
result{}
```

名称	类型	必填?	说明	备注
service_id	String(32)	Y	申请单号	510022664

5.3.7.5 请求示例

请求参数:

```
{
  "appkey": "CGDC",
  "serialnumber": "sjd79and83cgem",
  "timestamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "order_id": "18102207769",
  "received_name": "测试收件人",
  "received_telephone": "18628209572",
  "return_reason": "12",
  "return_desc": "退货描述",
  "province_code": "1",
  "city_code": "2801",
  "area_code": "4205",
  "town_code": "",
  "detail_address": "海鑫北路9号",
  "take_goods_type": 10,
  "goods_detail": [
    {
      "sku_code": "061089",
      "refund_count": 10
    }
  ],
  "proof_pic": [
    {
      "proof_url": "123"
    }
  ]
}
```

响应结果:

```
{
  "success": true,
```

```
"resultMessage": "true",  
  
"resultCode": "0000",  
  
"result": {  
  
    "service_id": "510022664"  
  
}  
  
}
```

5.3.8 退货单信息查询接口

5.3.8.1 应用场景

根据我方退货单号（拒收单号）查询退货单信息

5.3.7.2 请求说明

API 名称	getRefundInfo
调用入口	https://{url}/ getRefundInfo
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.3.7.3 应用级输入参数

名称	类型	必填?	说明	备注
service_id	String(32)	Y	xfS 退货单号	510022969

5.3.8.4 应用级返回结果

result{}

名称	类型	必填?	说明	备注
service_id	String(32)	Y	Xfs 退货单号	510022969
status	Int	Y	售后单状态: 10=待审核; 30=退货处理 40=退款处理 50=已完成 60=已关闭	
goods_fee	double(16,2)	Y	订单商品退款金额。 单位为元, 精确到小数点后两位	21238.10
refund_freight	double(16,2)	Y	运费退款金额。 单位为元, 精确到小数点后两位	21.10
service_fee	double(16,2)	Y	XFS 扣除服务费。 单位为元, 精确到小数点后两位	45.65
shipping_fee	double(16,2)	Y	鑫方盛取货收取费用。单位为元, 精确到小数点后两位	12.00
return_reason	int	Y	退货原因	10=商品质量问题; 60=少件/漏发; 70=多买/买错不要了; 80=到货物流破损; 90=发票问题; 100=未按约定时间送货; 110=其它原因;

				120=收到商品与网站展示不符; 130=客户原因无理由退货; 140=资质问题;
return_desc	String(500)	N	退货原因描述	
take_goods_type	int	Y	退货方式	10=上门取货; 20=客户邮寄
detail_address	String(255)	Y	详细地址	
goods_detail	List	Y	退货商品信息集合	goods_detail
refund_time	String(19)	Y	退货申请时间, 格式 "yyyy-MM-dd HH:mm:ss"	2019-06-09 01:01:20

退货商品参数

名称	类型	必填?	说明	备注
sku_code	String(20)	Y	商品 code	020080
refund_count	int	Y	退货数量	20

5.3.9 物流信息查询接口

5.3.9.1 应用场景

根据我方订单号查询物流信息, 在待收货状态及该状态后, 返回订单物流信息。

5.3.9.2 请求说明

API 名称	getLogistics
调用入口	http://{url}/getLogistics
HTTP 请求方式	HTTP-POST
格式	JSON

API 授权类型	需要授权
类型	基础服务

5.3.9.3 应用级输入参数

名称	类型	必填?	说明	备注
order_id	String(32)	Y	xfS 订单号	88107083503

5.3.9.4 应用级返回结果

result{}

名称	类型	必填?	说明	备注
sentType	Int	Y	物流类型: 10=快递; 20=快运-送货上门 21=快运-站点自提 30=专车包车 40=鑫方盛物流 50=上门自提	
xfSDriverName	String	N	鑫方盛物流公司姓名	默认 "", 当 sentType=40 时, 返回字段值
xfSDriverPhone	String	N	鑫方盛物流公司电话	默认 "", 当 sentType=40 时, 返回字段值
xfSDeliverName	String	N	鑫方盛物流配送员姓名	默认 "", 当 sentType=40 时, 返回字段值
xfSDeliverPhone	String	N	鑫方盛物流配送员电话	默认 "", 当 sentType=40 时, 返回字段值
proofUrlList	List<String>	N	凭证信息集合	默认 null, 图片 url
expressNum	String	N	快递公司单号	默认 "", 当 sentType=

				10 时, 返回字段值
expressCompany	String	N	快递公司名称	默认 "", 当 sentType=10 时, 返回字段值
transportNum	String	N	快运公司单号	默认 "", 当 sentType=20、21 时, 返回字段值
transportCompany	String	N	快运公司名称	默认 "", 当 sentType=20、21 时, 返回字段值
senderName	String	N	专车包车 DC 仓发货人	默认 "", 当 sentType=30 时, 返回字段值
driverName	String	N	专车包车司机	默认 "", 当 sentType=30 时, 返回字段值
logisticsInfolist	List	N	物流轨迹	默认 null 当 sentType=10、20、21 时, 返回字段值
deliverystatus	String	N	物流状态	如: 正在派件 默认 null 在 sentType=10、20、21 时返回字段 0: 快递收件(揽件) 1.在途中 2.正在派件 3.已签收 4.派送失败 5.疑难件 6.退件签收
expName	String	N	配送公司	如: 德邦 默认 null 当 sentType=10、20、21 时, 返回字段值

物流轨迹参数 logisticsInfolist

名称	类型	必填?	说明	备注
----	----	-----	----	----

time	String	N	物流轨迹时间	2020-03-25 17:03:23
status	String	N	物流信息	【石家庄市】快件已在【长安三部】签收,签收人: 本人,感谢使用中通快递,期待再次为您服务!"

5.3.9.5 请求示例

请求参数:

```
{  
  
  "appkey": "201912279535819",  
  "timestamp": "2019-12-17 17:19:06",  
  "token": "1b560f08d9f3f81067c9ee8465bbab56",  
  "order_id": "88107848303"  
}
```

响应结果:

```
{  
  
  "success": true,  
  
  "resultCode": "0000",  
  
  "resultMessage": "true",  
  
  "result": {  
  
    "sentType": 10,  
  
    "xfsDriverName": "",  
  
    "xfsDriverPhone": "",  
  
    "expressNum": "YT44090637802",  
  
    "expressCompanyName": ""  
  }  
}
```

```
"transportNum": "",

"transportCompanyName": "",

"senderName": "",

"driverName": "",

"logisticsInfolist": [

    {

        "time": "2020-03-25 17:03:23",

        "status": "客户签收人: 退回件 已签收 感谢使用圆通速递, 期待再次为您服务

如有疑问请联系: 1800808080"

    },

    {

        "time": "2020-03-25 12:29:11",

        "status": "【某某省某某市某某县某某乡公司】 派件中 派件人: 张三 电话

18080808080 "

    },

    {

        "time": "2020-03-25 10:19:11",

        "status": "【某某省某某市某某县公司】 已收入"

    },

    {

        "time": "2020-03-25 00:20:57",

        "status": "【某某转运中心】 已发出 下一站 【某某省某某市某某县公司】 "
```

```
    },  
  
    {  
  
      "time": "2020-03-24 23:54:20",  
  
      "status": "【某某转运中心公司】 已收入"  
  
    },  
  
    {  
  
      "time": "2020-03-23 23:42:54",  
  
      "status": "【北京转运中心公司】 已打包"  
  
    }  
  
  ],  
  
  "deliverystatus": "3",  
  
  "expName": "圆通速递"  
  
}
```

5.4.0 保存第三方订单状态接口

5.4.0.1 应用场景

用于保存第三方传来的鑫方盛订单号和该订单对应的第三方订单状态

5.4.0.2 请求说明

API 名称	channelOrderInfo
调用入口	http://{url}/channelOrderInfo
HTTP 请求方式	HTTP-POST
格式	JSON

API 授权类型	需要授权
类型	基础服务

5.4.0.2 应用级输入参数

名称	类型	必填?	说明	备注
orderId	String(32)	Y	xfS 订单号	88107083503
channelOrder Status	String(20)	Y	第三方订单状态	如：“待结算”

5.4.0.3 应用级返回结果

result{}

名称	类型	必填?	说明	备注
result			没有业务数据返回	null

5.4.0.4 请求示例

请求参数:

```
{
  "appkey": "201912090841588",
  "serialnumber": "sjd79and83cgem",
  "timestamp": "2020-06-05 08:13:01",
  "token": "a4594800d8d3ee8675f438b543fac7ce",
  "orderId": "123446",
  "channelOrderStatus": "订单完成"
}
```

响应结果:

```
{
  "success": true,
  "resultCode": "0000",
}
```

```
"resultMessage": "true",  
  
"result": null  
  
}
```

5.4 公共接口—消息接口

5.4.1 获取消息接口

5.4.1.1 应用场景

电商的商品、价格、订单等信息有变化时，将变化的商品 skuCode、变化的订单 id 写入消息表，明确变化类型，需要第三方电商平台定时获取变化信息。

5.4.1.2 请求说明

API 名称	queryEventRecord
调用入口	https://{url}/queryEventRecord
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.4.1.3 应用级输入参数

名称	类型	必填?	说明	备注
type	Int	N	类型	每个类型获取的结果不一样,不传默认查所有类型消息

5.4.1.4 应用级返回结果

result[]

名称	类型	必填?	说明	备注
id	String	Y	推送 id	
time	String	Y	推送时间	
type	Int	Y	消息类型	
result	result_entity	Y	消息体	

推送消息分类说明:

- type=1 代表订单拆分变更

result_entity

名称	类型	必填?	说明	备注
orderId	String	Y	父订单 id	Xfs 单据号
subOrderId	String	Y	子订单 id	Xfs 单据号
orderStatus	Int	Y	父订单状态	20=待审核; 110=交易关闭
subOrderStatus	Int	Y	子订单状态	20=待审核
splitStatus	Int	Y	主订单拆分状态	0 默认未拆分 10 拆分中 20 已拆分

每拆一次单都会发送一次拆单消息，消息体中会传递父订单号、子订单号、拆分状态，需要通过查询订单信息接口分别获取到子单、父单信息，进行相关更新；

“已拆分”：当一个父订单被拆分成 1 个或者多个子订单时，即主订单的商品被拆分到子订单中，此时父订单的状态是交易关闭，获取拆单消息时，子单的状态是待审核状态。

- type=2 代表商品价格变更

result_entity

名称	类型	必填?	说明	备注
skuId	String	Y	商品编号	通过价格接口获取最新价格

- type=3 代表订单状态变更

result_entity

名称	类型	必填?	说明	备注
orderId	String	Y	我方订单号	
status	Int	Y	订单状态: 20=待审核; 30=待配货; 40=待出库; 50=待发货; 60=待收货; 70=待结算; 80=订单成功; 90=交易完成; 100=订单取消; 110=交易关闭	20

➤ type=4 代表商品上下架变更消息

result_entity

名称	类型	必填?	说明	备注
skuId	String	Y	商品编号	通过上下级状态接口获取状态

➤ type=6 代表添加、删除商品池内商品

result_entity

名称	类型	必填?	说明	备注
skuId	String	Y	商品编号	
page_num	String	Y	商品池编号	
state	Int	Y	添加/删除	1 添加, 2 删除

➤ type=7 代表发票信息推送 (暂不用)

名称	类型	必填?	说明	备注
markId	String	Y	申请开发票的唯一标识	
state	Int	Y	当前状态	1 全部开票 2 驳回 3 部分开票 4 发票寄出 (可调用查询)

				发票运单号接口，获取运单号)
--	--	--	--	----------------

➤ type=8 代表退货服务单状态

场景：第三方退货操作、xfs 拒收操作、xfs 退货操作都会生成一条该类型的消息

result_entity

名称	类型	必填?	说明	备注
serviceId	String	Y	售后单号	
orderId	String	Y	xfs 订单号	
refundRole	Int	Y	退货角色	10: 买家 20: 客服
status	Int	Y	售后单状态: 10=待审核;30=退货处理;40=退款处理;50=已完成;60=已关闭	10

➤ type=9 代表维修服务单状态 (暂时不用)

result_entity

名称	类型	必填?	说明	备注
serviceId	String	Y	售后单号	
orderId	String	Y	xfs 订单号	
refundRole	Integer	Y	退货角色	10: 买家 20: 客服
status	Int	Y	售后单状态: 10=待审核;30=维修处理;50=已完成;60=已关闭	10

➤ type=10 代表订单取消

场景：待审核状态下 xfs 取消订单

result_entity

名称	类型	必填?	说明	备注
orderId	String	Y	订单编号	
cancelType	Int	N	取消类型	1: 电商因为自身原因无法

				发货而取消 不填默认为 0
cancelReason	cancelReason	N	取消原因	简要描述取消原因，对于 cancelType=0 的可不传

➤ type=16 代表商品介绍及规格参数变更消息

result_entity

名称	类型	必填?	说明	备注
skuId	String	Y	商品编号	

➤ type=17 代表赠品促销变更消息（暂时不用）

result_entity

名称	类型	必填?	说明	备注
skuId	String	Y	商品编号	

➤ type=101 代表拆分发货单（暂时不用）

result_entity

名称	类型	必填?	说明	备注
packageId	String	Y	发货单号	
orderId	String	Y	电商订单号	拆分该发货单的订单编号

➤ type=18 代表凭证信息消息

result_entity

名称	类型	必填?	说明	备注
orderId	String	Y	订单号	
proofUrls	String	Y	凭证信息	凭证对应 url, 多个凭证时, 各个 url 以“,”分割

5.4.1.5 请求示例

请求参数:

```
{
```

```
"appkey":"CGDC",

"serialnumber":"sjd79and83cgem",

"timestamp":"2017-08-19 10:13:01",

"token":"7886f145d87cb5d3219e8291c225f9c8",

"type":"8"

}
```

响应结果:

```
{

  "success": true,

  "resultMessage": true,

  "resultCode": true,

  "result":[

    {

      "serviceId": "123043",

      "state": 20

    }

  ]

}
```

5.4.2 删除消息接口

5.4.2.1 应用场景

各第三方平台获取消息并处理**成功**后，需要主动将推送消息删除。

5.4.2.2 请求说明

API 名称	deleteEventRecord
调用入口	https://{url}/deleteEventRecord
HTTP 请求方式	HTTP-POST
格式	JSON
API 授权类型	需要授权
类型	基础服务

5.4.2.3 应用级输入参数

名称	类型	必填?	说明	备注
ids	List<Integer>	Y	消息主键 id 的集合，最大为 100 条记录；最小为 1 条记录；假如入参有 100 个消息 id，其中含有非正常数据，接口也会将匹配到的正常数据进行标记，并返回成功结果。	[10012,10013]

5.4.2.4 应用级返回结果

返回结果在公共参数中显示：

如果成功，则 result 返回 ture 即可

如果失败，则 resultCode 返回失败码，resultMessage 会返回失败原因

5.4.2.5 请求示例

请求示例：

```
{
  "appkey": "CGDC",
  "serialnumber": "sjd79and83cgem",
```

```

    "timestamp": "2017-08-19 10:13:01",

    "token": "7886f145d87cb5d3219e8291c225f9c8",

    "ids": "[10012,10013]"

}

```

响应结果:

```

{

    "success": true,

    "resultCode": "0000",

    "resultMessage": "成功",

    "result": []

}

```

6.附录

6.1 返回码 (不含发票类)

发票类的不在此返回码列表中，在接口中单独描述

错误码	描述	备注	详细描述
0*** 成功			
0000	操作成功		查询相关接口，调用成功时返回
0001	下单成功		
0002	取消订单成功		
0003	确认订单成功		
0004	申请开票成功		
0005	全部开票成功		
0006	部分开票成功		
0007	取消开票成功		
0008	重复提交		同一三方订单号已经存在有效订单。此时下单结果 result 会返回该三方订单号对应订单信息。
0010	返回数据为空		
1*** 参数问题			通用，请检查参数

1001	参数为空		
1002	参数格式不正确		
1003	参数值不正确		
1004	参数重复		
1005	人参转化错误		请检查输入参数
2*** 权限问题			通用，请联系业务人员咨询对应权限问题
2001	用户权限不足		
2002	合同权限不足		
2003	企业权限不足		
2004	商品池权限不足		
2005	金彩权限问题		
2006	无有效增票资质		
2007	token 已过期		请重新刷新或者获取 token
3*** 业务问题			
31** 下单业务问题			
3001	价格不存在		
3002	提交订单过快		1 分钟后提交订单
3003	订单类型不支持		订单类型只支持普通自营、厂家直送、实物礼品卡、延保商品
3004	商品类型受限制		SkU 不在商品池，或者存在奢侈品或高值贵品商品
3005	商品没查询到		SkU 对应商品不存在
3006	商品不能进行货到付款下单		
3007	地址不能进行货到付款下单		
3008	库存不足		
3009	区域限制校验没通过		商品在该区域受限
3010	实体礼品卡和其他实物不能混合下单		
3011	大家电暂不支持公司转账 预占下单		
3012	海尔仓大家电，不支持后款预占下单		
3013	厂家直送商品只能下先款订单		
3014	厂家直送商品不能使用普票随货下单		
3015	实物礼品卡订单只能下普票订单		
3016	配额不足或者已被锁定	月结权限用户	检查月结额度

3017	余额不足		检查对应支付方式剩余额度
3018	客户端价格快照中sku价格不存在		
3019	客户端价格快照中sku价格与京东价格不一致		
3051 开始的为下游接口异常			
3051	价格获取失败		接口调用失败，可重试
3052	主数据接口业务异常		接口调用失败，可重试
3053	商品基本信息接口调用失败		接口调用失败，可重试
3054	商品扩展接口调用失败		接口调用失败，可重试
3055	大家电接口调用失败		接口调用失败，可重试
3056	赠品附件接口调用失败		接口调用失败，可重试
3057	区分大家电和中小件商品失败		接口调用失败，可重试
3058	下单失败，请重新提交订单		接口调用失败，可重试（下游下单接口返回，不确定重试能否成功）
31** 确认订单业务问题			
3101	确认下单最终失败，请重新确认订单		可重新确认
3102	orderId不存在		
3103	该订单已确认下单		已确认订单，不需要重复确认
3104	不能单独确认子订单		
3105	订单对应子单已取消，不能确认!		
3106	查询子单异常		
3107	本地子单与ERP子单不一致		
3108	确认订单操作失败		
32** 取消订单业务问题			
3201	取消订单失败，请重新取消订单		可重新取消
3202	orderId不存在		
3203	该订单已经被取消		已取消订单，不需要重复取消
3204	不能取消已经生产订单		
3205	不能取消未确认订单	//取消已确认订单接口使用	
3206	不能取消预占并且已确认订单	//取消已确认订单接口使用	

接口文档统一规范

3207	不能取消父订单	//取消已确认订单 接口使用	
3208	不能取消已确认订单		
3209	不能取消子订单		
3210	查询子单异常		
3211	取消订单失败, 存在 已确认的子单		
3212	取消订单操作失败		
33** 发票业务问题			
3301	待审核	目前只有联通使用	
3302	驳回	目前只有联通使用	
3303	通过待开票	目前只有联通使用	
3304	申请单不存在	目前只有联通使用	
34** 其余接口问题			
3401	订单不存在		没查询到对应订单
3402	订单配送信息不存在		
3403	支付时余额不足		
3404	订单不能发起支付		
3405	没查询到对应地址		
3406	价格不存在		
3407	获取余额业务异常		
3408	该支付类型不支持余 额查询		
3409	区分大家电和中小件 商品失败		
3410	订单未被挂起	目前只有联通使用	
5*** 系统异常			
5001	服务异常, 请稍后重 试		可重试
5002	未知错误		
6*** 申请售后服务接口异常			
0			无错误信息
2000	创建失败		
6000	网络异常, 请稍后重 试		
6001	参数不正确		
6002	该订单不存在,		
6003	该订单未确认提交,		
6004	该订单不是子订单		
6005	该订单未妥投,		
6006	该订单不是正常订		

	单,		
6007	订单没有该商品编号,		
6008	该订单是未完成订单		
6009	订单中某商品不可以提交售后服务		
6010	数量超过订单商品数量,		
6011	不属于商品返回京东方式		
6012	不属于支持的服务类型		
6013	服务单未通过审核		