

超市与各电商平台 对接统一规范

修改历史

版本	时间	内容描述
V0.3	2017-08-11	创建
V0.6	2017-08-15	1. 修改 5.6.1 统一下单接口中 sku_entity[]中相关字段描述
V0.7	2017-08-18	1. 修改 5.2.5 获取所有图片信息入参中 sku 字段的类型为 string[] 2. 修改 5.2.8 查询赠品信息中 sku 字段类型为 string，只支持单个查询，返回参数也是单个返回 3. 删除 5.2.7 商品可售验证入参中的地址信息，修改出参中 skuld 描述 4. 5.2.9 延保信息中商品编码字段改为 skulds
V0.8	2017-08-23	1. 增加 5.6.8 订单完结通知接口 2. 修改 5.6.4 订单反查接口的 API 名称为 selectOrderIdByThirdOrder 3. 修改 5.2.3 商品详细信息中 category 字段的描述 4. 修改 4.1.3.3 公共参数中返回参数 success 字段类型为 Boolean 5. 5.6.1 下单接口中返回参数的 sku_entity[]中 type 增加“3 延保” 6. 统一修改地址传送规则，如果没有三级或四级地址，则传 0
V0.9	2017-09-09	1. 修改消息推送接口中，type=5 时的逻辑，增加发货单全部完成后需要穿订单状态的要求

		2. 5.9.1 获取推送消息接口中, 应用级返回结果 page num 字段类型改为 String 3. 5.6.1 统一下单接口, 应用级返回结果 oid 字段添加备注: 如无返回 0 4. 5.2.11/5.2.12 查询分类信息、查询分类列表信息接口, 应用级返回结果父类目 ID 字段添加备注: 如无返回 0 5. 5.6.1 统一下单接口, 应用级返回结果商品数量、税率、税额、裸价、类型字段改为必填 6. 5.6.5 查询订单信息接口, 应用级返回结果 type 字段, 增加备注: 如果未拆单则返回 2 7. 6.1 返回码 3102、3202 去掉 jd, 改为 orderId 不存在 8. 5.5.1 批量查询库存接口, 应用级返回结果 stockStateDesc 字段, 首字母改为小写 9. 5.6.7 查询发货单信息接口/5.8.1 申请售后服务接口, 应用级输入参数 orderId 字段, 首字母改为小写 10. 5.3.1 地址变更消息推送, 应用级输入参数, 父类目地址改为非必填。 11. 5.8.1 申请售后服务, 应用级返回接口, servield 字段改为必填 12. 5.6.1 统一下单接口, 应用级输入参数 promiseDate 和 invoiceState 类型改为 Int 13. 补充除售后类和发票类接口外其他接口的示例, 请关注订单查询接口示例 14. 重写 8.2 对账类接口
--	--	---

		15. 1.2.1 申请售后服务入参增加 packageId 字段，并修改部分字段说明
V1.0	2017-9-13	1. 申请售后服务中客户信息实体的邮箱和邮编字段改为非必填 2. 商品详情查询接口返回参数增加 isReturn 字段 3. 5.6.1 统一下单、5.6.3 取消订单、5.6.6 查询订单信息增加场景描述 4. 5.9.1 获取推送消息接口修订 type=5 的字段说明 5. 5.9.1 获取推送消息接口增加 type=10 订单取消的分类和原因说明 6. 修订完善 5.10 发票类相关接口 7. 修改 5.2.10 运费查询接口返回参数中运费金额相关字段为 double 型 8. 5.8.1 申请售后接口中入参 email 字段改为 String 9. 5.2.5 获取所有图片接口接口增加图片地址字段格式要求
V1.1	2017-9-19	1. 所有金额、价格、税额字段，增加备注保留 2 位小数 2. 5.6.1 统一下单接口，应用级返回结果 nakedPrice 字段增加备注裸价=含税单价/（1+税率%），taxPrice 字段增加备注税额=含税单价-裸价 3. 5.10.1 申请发票开具，应用级输入参数 supplierOrder 类型改为 String[]，currentBatch 类型改为 Long，invoiceOrg/bizInvoiceContent/invoiceDate/totalBatchInvoiceAmount 字段改为必填 4. 5.2.12 查询分类列表信息接口，应用级输入参数

		parentId 类型改为 String 5. 5.8.1 申请售后服务接口,应用级输入参数,联系电话、手机号、邮箱、邮编类型改为 String 6. 5.6.5 查询订单信息接口,应用级返回结果 state 字段增加备注 3 是部分妥投 7. 5.6.1 统一下单接口,应用级输入参数 companyName 类型改为 String 8. 5.9.1 获取推送消息接口中 type=5 增加 orderType 字段,区分是订单或者发货单
V1.2	2017-9-22	1. 5.8.1 申请售后服务接口,应用级输入参数客户信息实体、取件信息实体、返件信息实体改为必填 2. 5.10.2 查询发票信息接口,应用级返回结果 supplierOrder 类型改为 String[]
V1.3	2017-11-14	1. 增加 5.4.2 按地区批量查询商品价格
V1.4	2021-02-18	删除地址类、对账类、售后类、发票类接口
V1.5	2021-02-25	删除查询分类列表信息接口 删除查询分类信息 删除按地区批量查询商品价格
V1.6	2021-03-09	增加 “5.5.9 取消订单的部分明细（针对未产生发货单的明细）”接口

V1.7	2021-04-16	改动 “5.3.1 获取商品详细信息”接口 1. 去掉 请求入参 isShow 2. 增加 部分字段备注 已标红部分 3. 修改 返回结果 wareQD 字段的类型 4. 增加 返回结果 nappintroduction 字段 5. 增加 返回结果 wxintroduction 字段 改动 “5.5.6 查询订单配送信息接口” 1. 修改 返回结果 orderTrack 字段为必要字段并增加备注 2. 增加 返回结果 waybillCode 字段
V1.8		改动 “5.5.8 订完结通知” 增加参数 type

目录

1. 前言.....	1
2. 缩略语和术语定义.....	1
2.1. 缩略语.....	1
2.2. 术语.....	1
3. 接入指南.....	2
4. 接入技术 V GJ.....	2
4.1. 实时接口.....	2
4.1.1. 传输协议.....	2
4.1.2. 报文协议.....	3
4.1.3. API 调用方式.....	4
4.1.4. 鉴权方式.....	8
5. 接口服务.....	8
5.1. 鉴权类.....	8
5.1.1. 获取 token 信息.....	8
5.2. 商品类.....	8
5.2.1. 获取商品池编号.....	11
5.2.2. 获取池内商品编号.....	12
5.2.3. 获取商品详细信息.....	14
5.2.4. 获取商品上下架状态.....	19
5.2.5. 获取所有图片信息.....	20
5.2.6. 商品区域购买限制查询.....	23
5.2.7. 商品可售验证.....	25

5.2.8. 查询赠品信息.....	— 26 —
5.2.9. 查询延保信息.....	— 29 —
5.2.10. 运费查询.....	— 32 —
5.2.11. 查询分类信息.....	错误! 未定义书签。
5.2.12. 查询分类列表信息接口.....	5
5.3. 地址类.....	错误! 未定义书签。
5.3.1. 地址变更信息推送.....	错误! 未定义书签。
5.4. 价格类.....	— 35 —
5.4.1. 批量查询商品价格.....	— 35 —
5.5. 库存类.....	— 36 —
5.5.1. 批量查询商品库存.....	— 37 —
5.6. 订单类.....	— 39 —
5.6.1. 统一下单.....	— 39 —
5.6.2. 确认订单.....	— 46 —
5.6.3. 取消订单.....	— 47 —
5.6.4. 订单反查接口.....	— 49 —
5.6.5. 查询订单信息.....	— 50 —
5.6.6. 查询订单配送信息接口.....	— 60 —
5.6.7. 查询发货单信息接口（针对有发货单的电商）.....	— 63 —
5.6.8. 订单完结通知.....	— 65 —
5.7. 对账类.....	错误! 未定义书签。
5.7.1. 新建订单对账接口.....	错误! 未定义书签。
5.7.2. 妥投/部分妥投订单对账接口.....	错误! 未定义书签。

5.7.3. 拒收订单对账接口.....	错误! 未定义书签。
5.8. 售后类.....	错误! 未定义书签。
5.8.1. 申请售后服务.....	错误! 未定义书签。
5.8.2. 根据服务单号查询服务单明细信息.....	错误! 未定义书签。
5.9. 信息推送.....	— 69 —
5.9.1. 获取推送消息接口.....	— 69 —
5.9.2. 删除消息接口.....	— 74 —
5.10. 发票类	错误! 未定义书签。
5.10.1. 申请发票开具.....	错误! 未定义书签。
5.10.2. 查询发票信息.....	错误! 未定义书签。
5.10.3. 查询发票运单信息.....	错误! 未定义书签。
6. 附录.....	— 76 —
6.1. 返回码（不含发票类）	— 76 —

1. 前言

2. 缩略语和术语定义

2.1. 缩略语

本文档所涉及的缩略语解释请参见下表。

缩略语	英文全称	中文全称
SKU	Stock Keeping Unit	库存量单位

2.2. 术语

本文档所涉及的术语定义请参见下表。

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
实时接口	实时接口是通过 HTTP 进行服务请求的调用方式，包括同步调用和异步调用
同步调用	在实时接口调用中，服务端业务处理结果在请求的反馈报文直接返回的调用方式
异步调用	在实时接口调用中，服务端在请求的反馈报文中只反馈接收状态，不反馈业务处理结果，业务处理完毕后，由服务端通过实时接口，调用请求端反馈业务处理结果
文件接口	系统间通过文件方式进行信息传递的方式

3. 接入指南

在与网上商城对接前需要准备如下工作：

- 环境准备：
 - a) 网络接入：通过互联网网络与网上商城连接
 - a) 系统开发：开发自身业务系统；
 - b) 系统注册：获得网上商城或外部业务系统的标识、key、密钥、授权等信息；
- 对接测试：外部业务系统与网上商城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；
- 验证开通：系统在正式开放前，进行若干周期的试运行。

4. 接入技术

支持两种接入方式：实时接口、文件接口，并提供先关自服务功能，向外部业务系统提供信息公告、结算对账等服务。

- 实时接口是一种以实时数据处理进行交互的接入方式，包含同步、异步两种方式。
- 文件接口是一种以文件形式进行交互的接入方式，支持的文件类型包括结算清单、对账文件等。

4.1. 实时接口

4.1.1. 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）
- GET 请求的数据会附在 URL 之后，以?分割 URL 和传输数据，参数之间以&相连。

- 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：

```
http://{url}/{接口服务名称}?xxxxx
```

4.1.2. 报文协议

4.1.2.1. 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。

4.1.2.1.1. 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,cOKD31n20140101101937816x93z91"  
}
```

4.1.2.1.2. 响应报文格式示例

- 正常情况响应报文格式：

```
HTTP/1.1 200 OK
```

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

```
Content-Length: #JSON 字符串长度
```

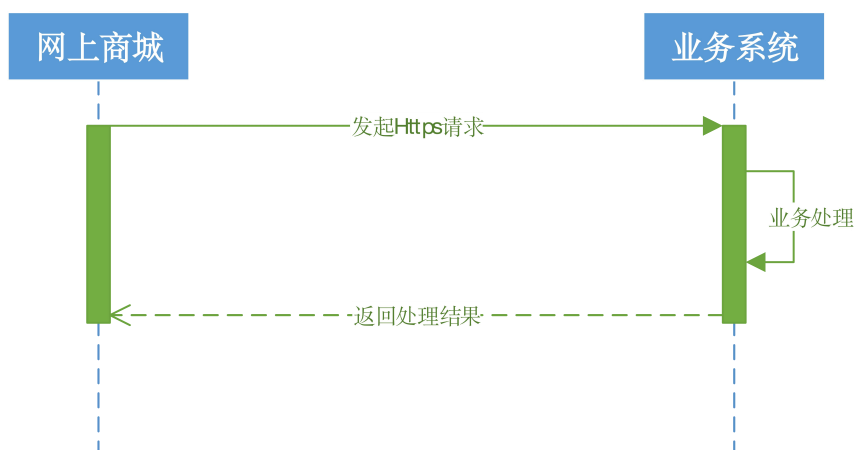
```
{  
    "XX": {  
        ...  
    }  
}
```

4.1.3. API 调用方式

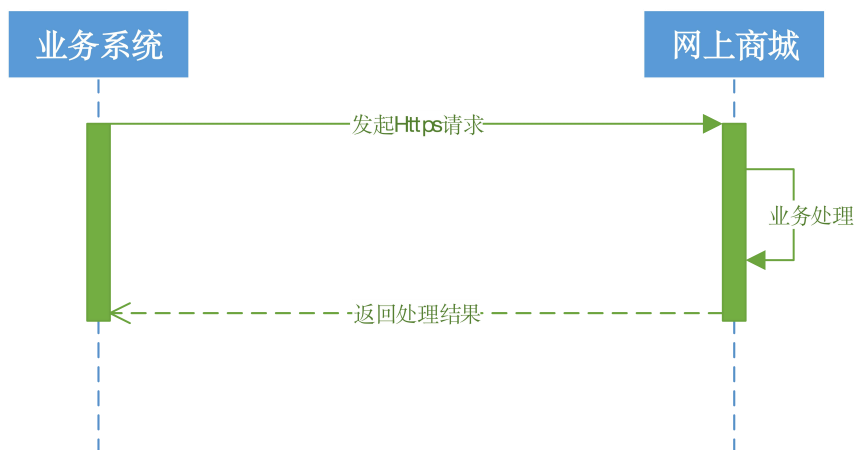
4.1.3.1. 调用机制

API 调用机制分为同步、异步两种方式，根据业务的不同，选择不同的交互机制，示例如下：

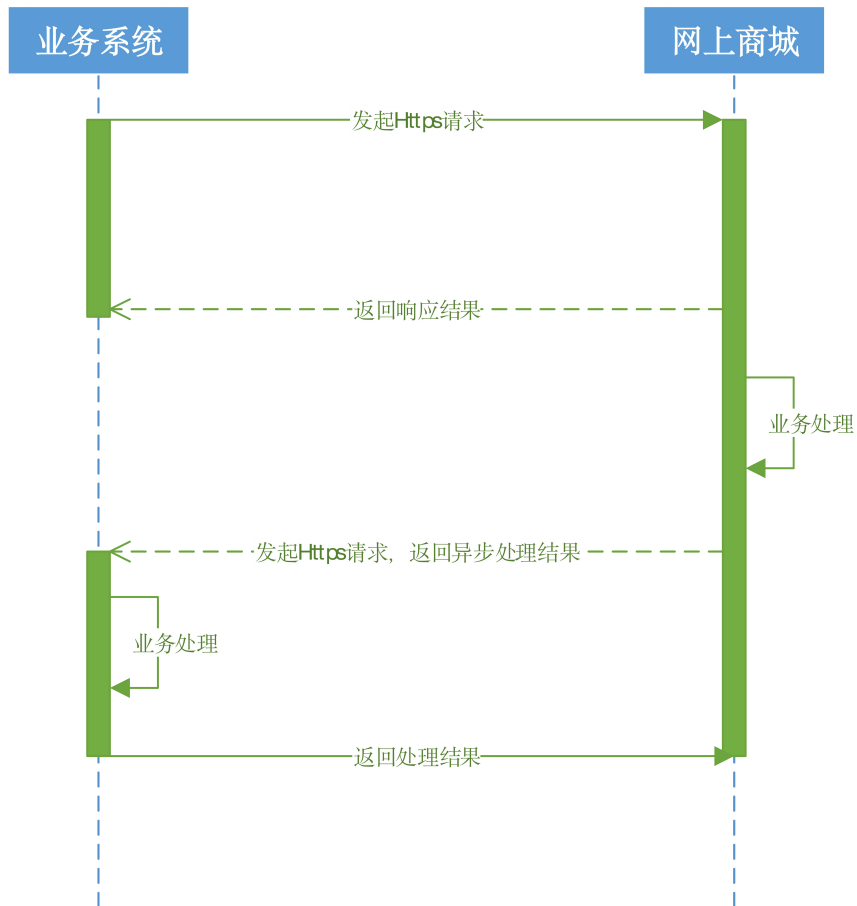
➤ 同步（网上商城 → 业务系统）：



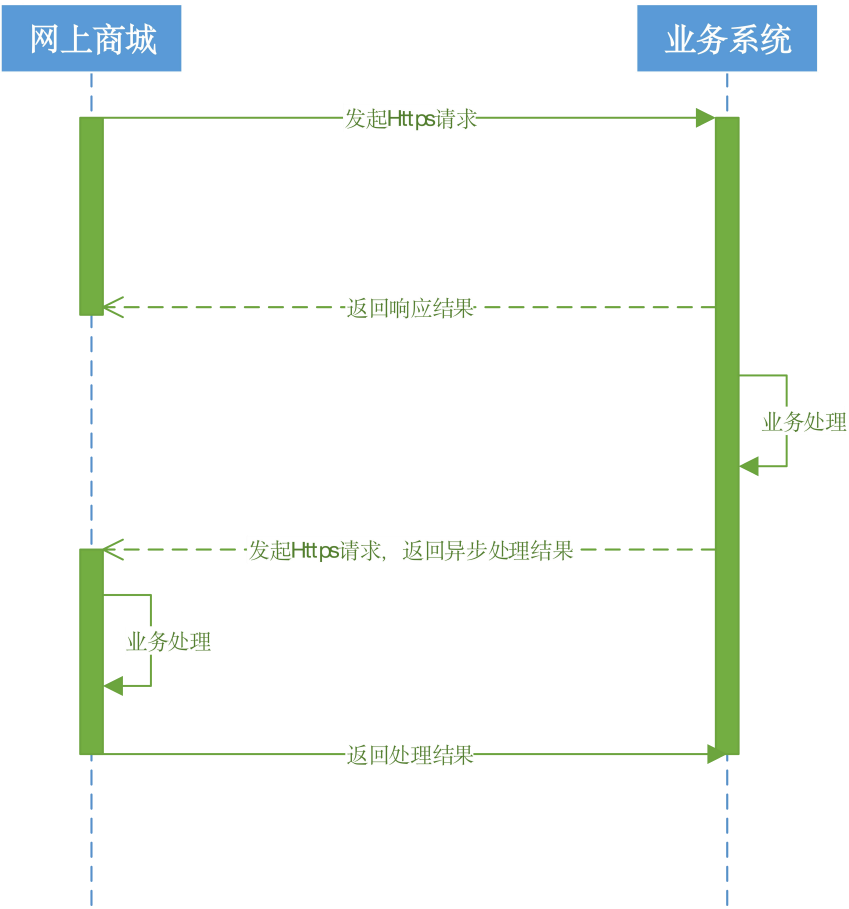
➤ 同步（业务系统 → 网上商城）：



➤ 异步（业务系统 → 网上商城）



- 异步（业务系统 → 网上商城）



4.1.3.2. 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。

4.1.3.3. 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
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

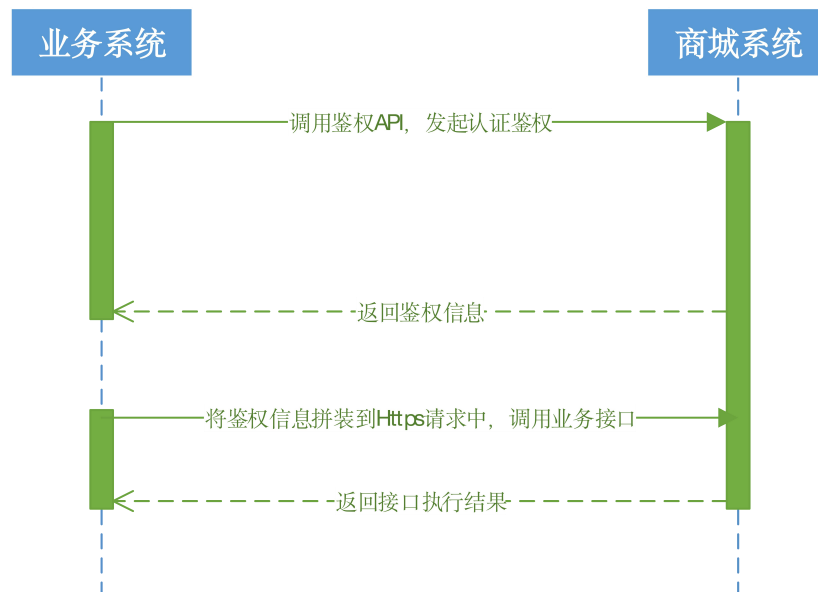
➤ 返回公共参数

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
--------	---------------	---	------	--------------

4.1.4. 鉴权方式

1. 在商城调用电商系统时，需要先向电商系统发起请求认证，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。示意图如下



2. 每一个鉴权信息都拥有失效时间，必须在鉴权信息失效前重新认证或续期；

5. 接口服务

5.1. 鉴权类

5.1.1. 获取 token 信息

5.1.1.1. 应用场景

获取电商的 token 鉴权值。如果各电商有自己 token 生成方式，可以提供自己的 token 接口。

5.1.1.2. 请求说明

API 名称	accessToken
调用入口	http://{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

5.1.1.3. 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	由电商事先分配
client_secret	String	Y	对接账号的密码	由电商事先分配
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
password	String	Y	用户密码	md5 加密后的字符串 (32 位小写)

5.1.1.4. 应用级返回结果

result _entity

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	

5.1.1.5. 示例

5.1.1.5.1. 请求示例

```
{
  "grant_type": "access_token",
  "client_id": "x9lelkX",
  "client_secret": "sjd79and83cgem",
  "timestamp": "2017-08-10 01:01:01",
  "username": "guodianwssc",
  "password": "0659c7992e268962384eb17fafa88364"
}
```

5.1.1.5.2. 返回示例

```
{
  "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
  }
}
```

```
}

```

5.1.1.6. 备注说明

Token 有效时间为 24 小时，接口每调用一次生成新 Token 并且重置有效时间。

5.2. 商品类

5.2.1. 获取商品池编号

5.2.1.1. 应用场景

获取电商为电子超市划定的商品池列表。

5.2.1.2. 请求说明

API 名称	getPageNum
调用入口	http://{url}/getPageNum
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

5.2.1.4. 应用级返回结果

result _entity[]

名称	类型	必填?	说明	备注
name	String	Y	商品池名称	
page_num	String	Y	商品池编码	

5.2.1.5. 示例

5.2.1.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8"
}
```

5.2.1.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "true",
  "resultCode": "0000",
  "result": [{
    "name": "办公用品",
    "page_num": "001"
  }]
}
```

5.2.1.6. 备注说明

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	getSkuByPage
调用入口	http://{url}/getSkuByPage

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
pageNum	String	Y	池编码	
pageNo	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

result _entity

名称	类型	必填?	说明	备注
pageCount	Int	Y	总页数	
skulds	String[]	Y	Sku 列表	

5.2.2.5. 示例

5.2.2.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8"
  "pageNum": "001",
```

```
"pageNo": 1
}
```

5.2.2.5.2. 返回示例

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

5.2.2.6. 备注说明

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、条形码、商品介绍等信息。

5.2.3.2. 请求说明

API 名称	getDetail
调用入口	http://{url}/getDetail
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	String	Y	商品编号	
isShow	Boolean	N	是否展现移动端及售后	false: 查询商品基本信息; true: 商品基本信息+商品售后信息+移动商品详情介绍信息 目前只会用到 false

5.2.3.4. 应用级返回结果

result _entity

名称	类型	必填?	说明	备注
sku	String	Y	商品编号	NULL 时不入库
name	String	Y	商品名称	NULL 时不入库
imagePath	String	Y	主图地址	全路径地址; 主图地址为直接地址, 不可做跳转
state	Int	Y	上下架状态	1: 上架 0: 下架 NULL 时不入库
category	String[]	Y	类目	按提供该 SKU 对应的所有类目信息, 并按照一二三四级顺序提供; NULL 时不入库; 映射不正确时不入库
saleUnit	String	Y	销售单位	NULL 时商品下架
taxInfo	String	Y	商品税率	乘以 100 不带百分号数字, 如税率为 3%时返回 3; 免税商品返回文字“免税”; NULL 时商品下架

电子超市对外交互规范

taxCode	String	Y	税控编码	NULL 时商品下架
introduction	String	Y	PC 商品描述	HTML 页面,可直接展现
nappintroduction	String	Y	移动端 App 商品描述	HTML 页面,可直接展现
wxintroduction	String[]	Y	微信小程序商品描述	图片地址, 全路径, 如多张图片组成许按顺序排列
moq	Int	Y	最小起订量	默认值为 1
brandName	String	Y	品牌	NULL 时商品不入库
brandPic	String	N	品牌图片	品牌图片地址, 如果没有可不传
productArea	String	N	产地	
upc	String	N	条形码	
mfgSku	String	N	制造商型号	
deliveryTime	Int	N	预计发货日	单位: 天 如果有现货则传 0
isReturn	Int	N	是否允许退换货	0: 允许退换货 1: 不允许退换货 默认为 0, 不传默认按 0 处理
weight	String	N	重量	
wareQD	wareQD _entity[]	N	包装信息	
param	param _entity[]	N	规格属性信息	

wareQD _entity[]

名称	类型	必填?	说明	备注
propName	String	Y	属性名称	

propValue	String	Y	属性值	
propGroupName	String	N	属性组名称	

param _entity[]

名称	类型	必填?	说明	备注
attrCode	String	N	属性编码	
attrName	String	Y	属性名称	
attrValue	String	Y	属性值	
attSeq	String	Y	属性排序	
attrGroupCode	String	N	属性组编码	
attrGroupName	String	N	属性组名称	
attrGroupSeq	String	N	属性组排序	

5.2.3.5. 示例

5.2.3.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8"
  "sku": "1216536",
  "isShow": "false"
}
```

5.2.3.5.2. 返回示例

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

```
"resultMessage": "true",

"resultCode": "0000",

"result": {

    "sku": "4255661",

    "weight": "40kg",

    "imagePath": "http://101.200.145.214:8081/login.jsp",

    "state": 1,

    "brandName ": "Apple",

    "name": "iphone",

    "brandPic": "http://101.200.145.214:8081/login.jsp",

    "productArea ": "广州",

    "upc": " 291387289472384729834",

    "saleUnit ": "西单商贸销售公司",

    "category ": ["01", "0101", "010101", "0101010001"],

    "taxInfo": "0.9",

    "taxCode": "12345666",

    "param ": [{

        "attrCode ": "12371283712",

        "attrName ": "颜色",

        "attrValue ": "粉色",

        "attrSeq ": "1"

    }],

    "wxintroduction "[

        "http://101.200.145.214:8081/login.jpg",

        "http://101.200.145.214:8081/login.jsp",

        "http://101.200.145.214:8081/login.jsp",

        "http://101.200.145.214:8081/login.jsp"

    ]

}
```

```
}  
  
}
```

5.2.3.6. 备注说明

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	skuState
调用入口	http://{url}/skuState
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	String []	Y	商品编号	

5.2.4.4. 应用级返回结果

result _entity[]

名称	类型	必填?	说明	备注
sku	String	Y	商品编号	
state	Int	Y	商品状态	1: 上架 0: 下架

5.2.4.5. 示例

5.2.4.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "sku": ["4255661", "4254388"]
}
```

5.2.4.5.2. 返回示例

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

5.2.4.6. 备注说明

批量查询的商品数量最低支持 100 个

5.2.5. 获取所有图片信息

5.2.5.1. 应用场景

通过 sku 编码查询该 sku 的所有图片信息。

5.2.5.2. 请求说明

API 名称	skulmage
调用入口	http://{url}/skulmage

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	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 _entity[]

名称	类型	必填?	说明	备注
sku	String	Y	商品编号	
skuPic	skuPic_entity[]	Y	商品图片	

skuPic_entity[]

名称	类型	必填?	说明	备注
path	String	Y	图片地址	图片地址
isPrimary	Int	Y	是否主图	1: 是 0: 否
orderSort	Int	Y	排序	主图可传 null 非主图传排序数字

path 字段可以由两种方式返回：

第一种方式是“固定前缀/nX/接口返回消息中图片地址”方式

其中 $n_0 > n_1 > n_2$ 为图片大小，图片大小可以不完全一致，但是不能比要求的小。

每张图传一个地址，有几张图就传几张。

第二种方式是在 path 字段中将图片地址按照从大、中、小的顺序都传过来，图片地址按照逗号分隔，其中要求最少有中图和小图。

5.2.5.5. 示例

5.2.5.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "sku": ["4255661"]
}
```

5.2.5.5.2. 返回示例

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

5.2.5.6. 备注说明

批量查询的商品数量最低支持 100 个，图片以正方形处理进行压缩

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	checkAreaLimit
调用入口	http://{url}/checkAreaLimit
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
skulds	String[]	Y	商品编号	支持批量, 以, 分隔
province	Int	Y	一级地址编号	
city	Int	Y	二级地址编号	
county	Int	Y	三级地址编号	
town	Int	N	四级地址编号	如无传 0

5.2.6.4. 应用级返回结果

result _entity[]

名称	类型	必填?	说明	备注
skuld	String	Y	商品编号	
isAreaRestrict	boolean	Y	True\false	true 代表区域受限 false 区域不受限

5.2.6.5. 示例

5.2.6.5.1. 请求示例

```
{  
  
  "appkey": "CGDC",  
  
  "serialNumber": "sjd79and83cgem",  
  
  "timeStamp": "2017-08-19 10:13:01",  
  
  "token": "7886f145d87cb5d3219e8291c225f9c8",  
  
  "skulds": ["4255661"],  
  
  "province": 1,  
  
  "city": 1,  
  
  "county": 0,  
  
  "town": 0  
  
}
```

5.2.6.5.2. 返回示例

```
{  
  
  "success": true,  
  
  "resultMessage": "true",  
  
  "resultCode": "0000",  
  
  "result": [{  
  
    "skuld": "4255661",  
  
    "isAreaRestrict": true  
  
  }]  
  
}
```

5.2.6.6. 备注说明

批量查询的商品数量最低支持 100 个

5.2.7. 商品可售验证

5.2.7.1. 应用场景

通过 sku 编码商品是否可售。电子超市根据各电商是否可售逻辑判断是否可以继续下单。

5.2.7.2. 请求说明

API 名称	check
调用入口	http://{url}/check
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
skulds	String[]	Y	商品编号	支持批量, 以, 分隔

5.2.7.4. 应用级返回结果

result _entity[]

名称	类型	必填?	说明	备注
skuld	String	Y	商品编码	
name	String	N	商品名称	
saleState	Int	Y	是否可售	1: 是 0: 否
toReturn	Int	N	支持几天无理由退货	默认为 0

5.2.7.5. 示例

5.2.7.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "skulds": ["4255661"]
}
```

5.2.7.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "true",
  "resultCode": "0000",
  "result": [{
    "skuld": "4255661",
    "name": "荣事达（Royalstar）养生壶煎药壶 1.8L 玻璃电水壶 304 不锈钢发热盘 YSH18Q",
    "saleState": 1,
    "toReturn ": 7
  }]
}
```

5.2.7.6. 备注说明

批量查询的商品数量最低支持 100 个

5.2.8. 查询赠品信息

5.2.8.1. 应用场景

通过 sku 编码和地址查询该 sku 是否由赠品。

5.2.8.2. 请求说明

API 名称	getSkuGift
调用入口	http://{url}/getSkuGift
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

5.2.8.3. 应用级输入参数

名称	类型	必填?	说明	备注
skuld	String	Y	商品编号	
province	Int	Y	一级地址编号	
city	Int	Y	二级地址编号	
county	Int	N	三级地址编号	如无传 0
town	Int	N	四级地址编号	如无传 0

5.2.8.4. 应用级返回结果

result _entity

名称	类型	必填?	说明	备注
maxNum	Int	N	赠品要求最多购买数量	为 0 表示没配置
minNum	Int	N	赠品要求最少购买数量	
promoStartTime	String	N	促销开始时间	
promoEndTime	String	N	促销结束时间	
gifts	gifts_entity[]	N	赠品附件列表	

gifts_entity[]

名称	类型	必填?	说明	备注
skuld	String	Y	商品编码	
num	Int	Y	赠品数量	
giftType	Int	Y	属性	1 附件 2 赠品

5.2.8.5. 示例

5.2.8.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "skuld": "4255661",
  "province": 1,
  "city": 1,
  "county": 0,
  "town": 0
}
```

5.2.8.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "true",
  "resultCode": "0000",
  "result": {
    "maxNum": 1,
    "minNum": 1,
  }
}
```

```
    "promoStartTime": "20170811120000",

    "promoEndTime": "20170815120000",

    "gifts": [{

        "skuld": "4255665",

        "num": 1,

        "giftType": 2

    }]

}
```

5.2.8.6. 备注说明

5.2.9. 查询延保信息

5.2.9.1. 应用场景

通过 sku 编码和地址查询该 sku 是否有延保。

5.2.9.2. 请求说明

API 名称	getYanbaoSku
调用入口	http://{url}/getYanbaoSku
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

5.2.9.3. 应用级输入参数

名称	类型	必填?	说明	备注
skulds	String[]	Y	商品编号	支持批量，以，分隔
province	Int	Y	一级地址编号	

city	Int	Y	二级地址编号	
county	Int	N	三级地址编号	如无传 0
town	Int	N	四级地址编号	如无传 0

5.2.9.4. 应用级返回结果

result _entity[]

名称	类型	必填?	说明	备注
skuld	String	N	商品编码	
yanBao	yanBao_entity[]	N	延保信息实体	

yanBao_entity[]

名称	类型	必填?	说明	备注
mainSkuld	String	Y	主商品的 sku	
imgUrl	String	Y	保障服务类别 显示图标 url	
detailUrl	String	Y	保障服务类别 静态页详情 url	
displayNo	Int	Y	保障服务类别 显示排序	
categoryCode	String	Y	保障服务分类 编码	
displayName	String	Y	保障服务类别 名称	
fuwuSkuDetailList	fuwuSkuDetailList_entity[]	Y	保障服务商品 详情列表	

fuwuSkuDetailList_entity[]

名称	类型	必填?	说明	备注
bindSkuld	String	Y	保障服务 skuld	
bindSkuName	String	Y	保障服务 sku 名称 (6 字内)	
sortIndex	Int	Y	显示排序	

price	BigDecimal	Y	保障服务 sku 价格	保留 2 位小数
tip	String	Y	保障服务说明提示语（20 字内）	
favor	Boolean	Y	是否是优惠保障服务	PC 单品页、PC 购物车会根据此标识是否展示优惠图标，优惠图标单品页提供

5.2.9.5. 示例

5.2.9.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "skulds": ["4255661"],
  "province": 1,
  "city": 1,
  "county": 0,
  "town": 0
}
```

5.2.9.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "true",
  "resultCode": "0000",
  "result": [{
    "skuld": "4255661",
    "yanBao": [{
```



```
    "mainSkuld": "4255665",

    "imgUrl": "http://sale.jd.com/act/573d95abNc672af4d.png",

    "detailUrl": "http://sale.jd.com/act/s7oXRvFNyVI.html",

    "displayNo": 1,

    "categoryCode": " VXP-YCBX",

    "displayName": "延长保修",

    "fuwuSkuDetailList": [{

        "bindSkuld": "4255667",

        "bindSkuName": "延长保修 2 年",

        "sortIndex": 1,

        "price": 198,

        "tip": "保修期延长 2 年，就近维修，不限次服务",

        "favor": "false"

    }]

  }

}
```

5.2.9.6. 备注说明

5.2.10. 运费查询

5.2.10.1. 应用场景

通过 sku 编码和地址信息，查询该 sku 配到到该地址的运费价格。

5.2.10.2. 请求说明

API 名称	getFreight
调用入口	http://{url}/getFreight
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	sku_entity[]	Y	商品信息	
province	Int	Y	一级地址	
city	Int	Y	二级地址	
county	Int	N	三级地址	如无传 0
town	Int	N	四级地址	如无传 0

sku_entity[]

名称	类型	必填?	说明	备注
skuld	String	Y	商品编码	
num	Int	Y	商品数量	

5.2.10.4. 应用级返回结果

result _entity

名称	类型	必填?	说明	备注
freight	double	Y	总运费	保留 2 位小数
baseFreight	double	Y	基础运费	保留 2 位小数
remoteRegionFreight	double	Y	偏远运费	保留 2 位小数
remoteSku	String[]	Y	需收取偏远运费的 sku	

5.2.10.5. 示例

5.2.10.5.1. 请求示例

```
{  
  
  "appkey": "CGDC",  
  
  "serialNumber": "sjd79and83cgem",  
  
  "timeStamp": "2017-08-19 10:13:01",  
  
  "token": "7886f145d87cb5d3219e8291c225f9c8",  
  
  "sku": [{  
  
    "skuld": "4255661",  
  
    "num": 10  
  
  }],  
  
  "province": 1,  
  
  "city": 1,  
  
  "county": 0,  
  
  "town": 0  
  
}
```

5.2.10.5.2. 返回示例

```
{  
  
  "success": true,  
  
  "resultMessage": "true",  
  
  "resultCode": "0000",  
  
  "result": {  
  
    "freight": 25,  
  
    "baseFreight": 5,  
  
    "remoteRegionFreight": 20,  
  
    "remoteSku": ["4255661"]  
  
  }  
  
}
```

5.2.10.6. 备注说明

5.3. 价格类

5.3.1. 批量查询商品价格

5.3.1.1. 应用场景

根据 sku 编码批量查询电商价和协议价。

5.3.1.2. 请求说明

API 名称	getSellPrice
调用入口	http://{url}/getSellPrice
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
sku	String[]	Y	商品编号, 请以, (英文逗号) 分割	

5.3.1.4. 应用级返回结果

result _entity[]

名称	类型	必填?	说明	备注
skuld	String	Y		
price	Double	Y	协议价格	保留 2 位小数
ecPrice	Double	Y	电商主站价格	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

5.3.1.5. 示例

5.3.1.5.1. 请求示例

```
{  
  
  "appkey": "CGDC",  
  
  "serialNumber": "sjd79and83cgem",  
  
  "timeStamp": "2017-08-19 10:13:01",  
  
  "token": "7886f145d87cb5d3219e8291c225f9c8",  
  
  "sku": ["4255661", "4255662"]  
  
}
```

5.3.1.5.2. 返回示例

```
{  
  
  "success": true,  
  
  "resultMessage": "true",  
  
  "resultCode": "0000",  
  
  "result": [{  
  
    "skuld": "4255661",  
  
    "price": 20,  
  
    "ecPrice": 25  
  
  ]  
  
}
```

5.3.1.6. 备注说明

批量查询的商品数量最低支持 100 个

5.4. 库存类

5.4.1. 批量查询商品库存

5.4.1.1. 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。

5.4.1.2. 请求说明

API 名称	getNewStockById
调用入口	http://{url}/getNewStockById
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
skuNums	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
skuld	String	Y	商品	
num	int	Y	数量	

5.4.1.4. 应用级返回结果

result _entity[]

名称	类型	必填?	说明	备注
areald	String	Y	配送地址 id	

desc	String	N	描述	
skuld	String	Y	商品编号	
stockStateId	Int	Y	库存状态编号	33,39,40,36,34
stockStateDesc	String	Y	库存状态描述	33 有货 现货—下单立即发货 39 有货 在途—正在内部配货, 预计 2~6 天到达本仓库 40 有货 可配货—下单后从有货仓库配货 36 预订 34 无货
remainNum	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

5.4.1.5. 示例

5.4.1.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "skuNums": [{
    "skuld": "4255661",
    "num": 10
  }],
  "area": "1_0_0"
}
```

5.4.1.5.2. 返回示例

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

```
"resultMessage": "true",

"resultCode": "0000",

"result": [{

    "areald": "1_0_0",

    "skuld": "4255661",

    "stockStateld": 33,

    "StockStateDesc": "有货",

    "remainNum": -1

}]

}
```

5.4.1.6. 备注说明

如果有商品不在商品池，不需要报错，返回在商品池中的商品库存

批量查询的商品数量最低支持 100 个

5.5. 订单类

5.5.1. 统一下单

5.5.1.1. 应用场景

电子超市将需要购买的商品传递给电商，电商进行下单校验返回下单结果。统一下单是下的预订单，电商只会锁定价格和库存，不会执行配货物流工作。

统一下单接口需要做幂等性校验，如果订单号相同，需要返回订单号重复，不能重复下单。

下单时需要按照下单价格模式要求做价格校验，价格不一致不能下单，以免后续订单确认失败。

由于目前都采用集中开票模式，统一下单接口中的发票信息只做参考使用，实际的开票信息需要按照发票申请接口确定。

下单结果需要返回订单中单个商品明细价税分离信息，包括裸价、税率、税额和总价，并满足税额=裸价*税率、总价=裸价+税额的关系。

5.5.1.2. 请求说明

API 名称	submitOrder
调用入口	http://{url}/submitOrder
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

5.5.1.3. 应用级输入参数

名称	类型	必填?	说明	备注
thirdOrder	String	Y	电子超市订单号	
sku	sku_entity[]	Y	商品信息	
name	String	Y	收货人	
province	Int	Y	一级地址	
city	Int	Y	二级地址	
county	Int	Y	三级地址	如无传 0
town	Int	Y	四级地址	如无传 0
address	String	Y	详细地址	
zip	String	N	邮编	
phone	String	N	座机号	
mobile	String	Y	手机号	
email	String	N	邮箱	
remark	String	N	备注（少于 100 字）	
submitState	Int	Y	是否预占库存	0 是预占库存（需要调

				用确认订单接口) 1 是不预占库存
doOrderPriceMode	Int	Y	下单价格模式	1:必需验证订单价格快照, 如果与电商价格不一致返回下单失败, 需要更新商品价格后, 重新下单
promiseDate	Int	N	配送预约日期	1: 工作日送货 2: 任意时间送货 不传默认工作日送货
invoiceState	Int	Y	开票方式	1 为随货开票, 0 为订单预借, 2 为集中开票 目前只支持集中开票, 选集中开票不需要传以下发票相关信息
invoiceType	Int	N	发票类型	1 普通发票 2 增值税发票
selectedInvoiceTitle	Int	N	抬头类型	4 个人, 5 单位
companyName	String	N	发票抬头	单位发票需要
invoiceContent	Int	N	发票内容	1: 明细, 3: 电脑配件, 19: 耗材, 22: 办公用品 备注: 若增值发票则只能选 1 明细
paymentType	Int	N	支付方式	101: 集中付款 目前只支持该方式
isUseBalance	Int	N	是否使用余额	默认 0
invoiceName	String	N	收票人姓名	
invoicePhone	String	N	收票人电话	
invoiceProvince	Int	N	收票人所在省	一级地址
invoiceCity	Int	N	增值票收票人所在市	二级地址
invoiceCounty	Int	N	增值票收票人所在区/县	三级地址
invoiceAddress	String	N	增值票收票人所在地址	

sku_entity[]

名称	类型	必填?	说明	备注
skuld	String	Y	商品编号	
num	Int	Y	商品数量	
bNeedAnnex	Boolean	N	是否需要附件	不传默认值为: true (需要附件)
bNeedGift	Boolean	N	否需要赠品	不传默认值为: false (不需要赠品)
price	Double	Y	商品价格	保留 2 位小数
yanbao	yanbao_entity[]	N	商品延保	

yanbao_entity[]

名称	类型	必填?	说明	备注
skuld	String	Y	商品编号	

5.5.1.4. 应用级返回结果

result_entity

名称	类型	必填?	说明	备注
orderId	String	Y	订单号	对外暴露的订单编号
freight	Double	N	总运费	总运费 = 基础运费 + 总的超重偏远附加运费, 保留 2 位小数
sku	sku_entity[]	Y	商品信息	
orderPrice	Double	Y	商品总价格	保留 2 位小数
orderNakedPrice	Double	N	订单裸价	保留 2 位小数
orderTaxPrice	Double	N	订单税额	保留 2 位小数

sku_entity[]

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

skuld	String	Y	商品编码	
num	Int	Y	商品数量	
price	Double	Y	商品价格	保留 2 位小数
name	String	Y	商品名称	
tax	Int	Y	税率	
taxPrice	Double	Y	税额	税额=含税单价-裸价, 保留 2 位小数
nakedPrice	Double	Y	裸价	裸价=含税单价/ (1+税率%), 保留 2 位小数
type	Int	Y	类型	0 普通、1 附件、2 赠品、 3
oid	String	N	主商品 skuid	如无返回 0
remoteRegionFreight	Double	N	偏远超重附加运费	保留 2 位小数

- 下单失败, "success"返回为 false。resultCode 返回失败码, resultMessage 会返回失败原因。

5.5.1.5. 示例

5.5.1.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "thirdOrder": "12345678",
  "sku": [{
    "skuld": "4255661",
    "num": 10,
    "price": 1500.00,
    "yanbao": [{
```

```
        "skuld": "4255665"

    }

}],

"name": "张三",

"province": 1,

"city": 1,

"county": 0,

"town": 0,

"address": "中关村大街 100 号",

"zip": "100025",

"phone": "010-66274625",

"mobile": "13726729464",

"email": "212381@163.com",

"remark": "尽快发货",

"submitState": 0,

"doOrderPriceMode": 1,

"invoiceState": 1,

"invoiceType": 1,

"selectedInvoiceTitle": 5,

"companyName": "北京天源迪科有限公司",

"invoiceContent": "2",

"invoiceName": "路人乙",

"invoicePhone": "13648762743",

"invoiceProvice": 1,

"invoiceCity": 1,

"invoiceCounty": 0

}
```

5.5.1.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "true",
  "resultCode": "0000",
  "result": {
    "orderId": "11223344",
    "freight": "10.00",
    "sku": [{
      "skuld": "4255661",
      "num": "10",
      "price": "117.00",
      "name": "商品名称",
      "tax": "17",
      "taxPrice": "17.00",
      "nakedPrice": "100.00",
      "type": "1"
    }],
    "orderPrice": 117,
    "orderNakedPrice": 100,
    "orderTaxPrice": 17
  }
}
```

5.5.1.6. 备注说明

如果因为某个（某几个）商品价格、库存等有问题，无法进行下预订单，请返回商品 sku 什么问题，描述清楚

5.5.2. 确认订单

5.5.2.1. 应用场景

电子超市会通过确认订单接口进行预订单确认，确认后的订单，电商可以开始执行订单配货物流操作。

5.5.2.2. 请求说明

API 名称	confirmOrder
调用入口	http://{url}/confirmOrder
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

5.5.2.3. 应用级输入参数

名称	类型	必填?	说明	备注
orderId	String	Y	电商订单号	

5.5.2.4. 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 true 即可
- 如果失败，则 result 返回 false，resultCode 返回失败码，resultMessage 会返回失败原因

5.5.2.5. 示例

5.5.2.5.1. 请求示例

```
{  
  
  "appkey": "CGDC",
```

```
{
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "orderId": "20178282739719172"
}
```

5.5.2.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "确认订单成功",
  "resultCode": "0000",
  "result": true
}
```

5.5.2.6. 备注说明

5.5.3. 取消订单

5.5.3.1. 应用场景

在订单未确认前，电子超市可以主动取消订单。

对于电商主动取消的订单需要在消息通知接口告知电商，取消订单必须按照订单维度取消。

5.5.3.2. 请求说明

API 名称	cancel
调用入口	http://{url}/cancel
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

5.5.3.3. 应用级输入参数

名称	类型	必填?	说明	备注
orderId	String	Y	电商订单号	

5.5.3.4. 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 true 即可
- 如果失败，则 result 返回 false，resultCode 返回失败码，resultMessage 会返回失败原因

5.5.3.5. 示例

5.5.3.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "orderId": "20178282739719172"
}
```

5.5.3.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "取消订单成功",
  "resultCode": "0000",
  "result": true
}
```

5.5.3.6. yeqian 备注说明

5.5.4. 订单反查接口

5.5.4.1. 应用场景

如果下单时接受返回超时，可以通过该接口查询订单是否下单成功。

5.5.4.2. 请求说明

API 名称	selectOrderIdByThirdOrder
调用入口	http://{url}/selectOrderIdByThirdOrder
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

5.5.4.3. 应用级输入参数

名称	类型	必填?	说明	备注
thirdOrder	String	Y	电子超市订单号	

5.5.4.4. 应用级返回结果

名称	类型	必填?	说明	备注
result	String	Y	电商订单号	

返回结果在公共参数中显示：

- success 为 true，则代表下单成功，result 值为电商的订单号
- success 为 false，则代表电商订单号不存在，即下单未成功，需要重新下单

5.5.4.5. 示例

5.5.4.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "thirdOrder": "20170908000000"
}
```

5.5.4.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "查询成功",
  "resultCode": true,
  "result": "jd12345656890"
}
```

5.5.4.6. 备注说明

5.5.5. 查询订单信息

5.5.5.1. 应用场景

根据电商的订单号，查询订单详细信息。如果发生拆单时，用父订单号可以查询所有拆分的子订单信息。

订单查询的结果中的金额信息也需要满足统一下单接口中的金额信息要求。

5.5.5.2. 请求说明

API 名称	qrySubOrder
调用入口	http://{url}/qrySubOrder

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

5.5.5.3. 应用级输入参数

名称	类型	必填?	说明	备注
orderId	String	Y	电商订单号	

5.5.5.4. 应用级返回结果

result _entity

名称	类型	必填?	说明	备注
pOrder	pOrder_entity	N	父订单信息	如果未拆单, 或者用子订单号查询, 则不传
cOrder	cOrder_entity[]	N	子订单信息	如果未拆单, 或者用子订单号查询, 则不传
orderId	String	N	电商订单号	如果未拆单则返回
state	Int	N	物流状态	0 是新建 1 是妥投 2 是拒收 3 是部分妥投
orderState	Int	N	订单状态	0 为取消订单 1 为有效
submitState	Int	N	确认状态	0 为未确认下单订单 1 为确认下单订单
orderPrice	Double	N	订单价格	订单含税价格, 保留 2 位小数
orderNakedPrice	Double	N	订单裸价格	不含税价格, 保留 2 位小数
orderTaxPrice	Double	N	订单税额	保留 2 位小数
type	Int	N	订单类型	1 是父订单 2 是子订单 如果未拆单则返回 2

freight	Double	N	总运费	保留 2 位小数
baseFreight	Double	N	基础运费	保留 2 位小数
sku	sku_entity[]	N	商品列表	

pOrder_entity

名称	类型	必填?	说明	备注
orderId	String	N	电商订单号	电商父订单号
orderPrice	Double	N	订单价格	订单含税价格, 保留 2 位小数
freight	Double	N	总运费	保留 2 位小数
sku	sku_entity[]	N	商品列表	

cOrder_entity[]

名称	类型	必填?	说明	备注
pOrder	String	Y	父订单号	该条子订单归属的父订单, 与 pOrder_entity 中的 orderId 一致
orderId	String	Y	子订单号	该条子订单的订单号
state	Int	N	物流状态	0 是新建 1 是妥投 2 是拒收
orderState	Int	Y	订单状态	0 为取消订单 1 为有效
submitState	Int	Y	确认状态	0 为未确认下单订单 1 为确认下单订单
freight	Double	N	总运费	保留 2 位小数
orderPrice	Double	N	订单价格	订单含税价格, 保留 2 位小数
orderNakedPrice	Double	N	订单裸价格	不含税价格, 保留 2 位小数
orderTaxPrice	Double	N	订单税额	保留 2 位小数
type	Int	N	订单类型	1 是父订单 2 是子订单
baseFreight	Double	N	基础运费	保留 2 位小数

sku	sku_entity[]	N	商品列表	
-----	--------------	---	------	--

sku_entity[]

名称	类型	必填?	说明	备注
skuld	String	Y	商品编码	
num	Int	Y	商品数量	
category	String	N	商品类目	
price	Double	Y	商品价格	
type	Int	Y	商品类型	0 普通、1 附件、2 赠品
oid	String	N	主商品编码	如果本身是主商品，则 oid 为 0
name	String	N	商品名称	
tax	Int	N	商品税率	
taxPrice	Double	N	税额	保留 2 位小数
nakedPrice	Double	N	商品裸价	保留 2 位小数

拆单说明:

- 对于未拆单的，不返回 pOrder_entity 和 cOrder_entity，订单信息直接在 result 中展现，订单中的商品列表在 sku_entity[]中展现
- 对于已拆单的，如果按照父订单查询，则返回 pOrder_entity 和 cOrder_entity，分别展现；如果按照子订单号查询，也不返回 pOrder_entity 和 cOrder_entity，result 直接返回该子订单的详细信息
- 各场景的具体返回按照示例内容展现

5.5.5.5. 示例

5.5.5.5.1. 请求示例

```
{  
  
  "appkey": "CGDC",  
  
  "serialNumber": "sjd79and83cgem",  
}
```

```
{
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "orderId": "2302507708"
}
```

5.5.5.5.2. 返回示例

1. 有拆单，用父订单查询

```
{
  "success": true,
  "resultMessage": "成功",
  "resultCode": "0000",
  "result": {
    "pOrder": {
      "orderId": 2302507708,
      "orderPrice": 18.6,
      "freight": 0,
      "sku": [{
        "skuld": 831058,
        "num": 1,
        "category": 9919,
        "price": 9.8,
        "type": 0,
        "oid": 0,
        "name": "雨花泽（YuhuaZe） YHZ-94001 黄铜锁 20 工艺精湛坚固耐用",
        "tax": 17,
        "taxPrice": 1.42,
        "nakedPrice": 8.38
      }],
    },
  }
}
```

```
        "skuld": 892726,

        "num": 1,

        "category": 9435,

        "price": 3.8,

        "type": 1,

        "oid": 831058,

        "name": "红星小二锅头 56 度 100ml",

        "tax": 17,

        "taxPrice": 0.55,

        "nakedPrice": 3.25

    }

]

},

"cOrder": [{

    "pOrder": 2302507708,

    "orderState": 0,

    "orderId": 2302541052,

    "state": 2,

    "submitState": 1,

    "orderPrice": 3.8,

    "freight": 0,

    "orderNakedPrice": 3.25,

    "type": 2,

    "sku": [{

        "skuld": 892726,

        "num": 1,

        "category": 9435,

        "price": 3.8,
```



```
        "type": 1,

        "oid": 831058,

        "name": "红星小二锅头 56 度 100ml",

        "tax": 17,

        "taxPrice": 0.55,

        "nakedPrice": 3.25

    }],

    "orderTaxPrice": 0.55

},

{

    "pOrder": 2302507708,

    "orderState": 0,

    "orderId": 2302541060,

    "state": 2,

    "submitState": 1,

    "orderPrice": 9.8,

    "freight": 0,

    "orderNakedPrice": 8.38,

    "type": 2,

    "sku": [{

        "skuld": 831058,

        "num": 1,

        "category": 9919,

        "price": 9.8,

        "type": 0,

        "oid": 0,

        "name": "雨花泽（YuhuaZe） YHZ-94001 黄铜锁 20 工艺精湛坚固耐用",

        "tax": 17,
```

```
        "taxPrice": 1.42,

        "nakedPrice": 8.38

    }],

    "orderTaxPrice": 1.42

}

],

"orderState": 1,

"submitState": 1,

"orderNakedPrice": 11.63,

"type": 1,

"orderTaxPrice": 1.97

}

}
```

2. 有拆单，用子订单号查询

```
{

    "success": true,

    "resultMessage": "成功",

    "resultCode": "0000",

    "result": {

        "pOrder": 0,

        "orderState": 1,

        "orderId": 2302507708,

        "state": 0,

        "freight": 0,

        "submitState": 0,

        "orderPrice": 124.5,

        "baseFreight ": "基础运费",

        "orderNakedPrice": 106.41,
```

```
"sku": [{
    "skuld": 852431,
    "num": 1,
    "category": 690,
    "price": 124.5,
    "name": "罗技 (Logitech) M545 无线鼠标 黑色",
    "tax": 17,
    "taxPrice": 18.09,
    "nakedPrice": 106.41,
    "type": 0,
    "oid": 0
},
{
    "skuld": 852431,
    "num": 1,
    "category": 690,
    "price": 0,
    "name": "罗技 (Logitech) M545 无线鼠标 黑色",
    "tax": 0,
    "taxPrice": 0,
    "nakedPrice": 0,
    "type": 2,
    "oid": 852431
}
],
"type": 2,
"orderTaxPrice": 18.09
}
```

```
}
```

3. 无拆单

```
{  
  
  "success": true,  
  
  "resultMessage": "成功",  
  
  "resultCode": "0000",  
  
  "result": {  
  
    "orderState": 1,  
  
    "state": 0,  
  
    "submitState": 0,  
  
    "orderPrice": 16.95,  
  
    "orderNakedPrice": 14.49,  
  
    "sku": [{  
  
      "category": 1671,  
  
      "num": 1,  
  
      "price": 16.95,  
  
      "tax": 17,  
  
      "oid": 0,  
  
      "name": "【京东超市】维达(Vinda) 抽纸 超韧 3层 130抽软抽*6包(中规格)",  
  
      "taxPrice": 2.46,  
  
      "skuld": "1131242",  
  
      "nakedPrice": 14.49,  
  
      "type": 0  
    }],  
  
    "type": 2,  
  
    "orderTaxPrice": 2.46,  
  
    "orderId": "60750428086"  
  
  }  
}
```

```
}
```

5.5.5.6. 备注说明

5.5.6. 查询订单配送信息接口

5.5.6.1. 应用场景

根据订单号查询订单的配送信息，按照时间给配送信息明细。

5.5.6.2. 请求说明

API 名称	orderTrack
调用入口	http://{url}/orderTrack
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

5.5.6.3. 应用级输入参数

名称	类型	必填?	说明	备注
orderId	String	Y	查询单号	如果发生拆单，则需要传子订单号 如果进一步拆成发货单，则传发货单号
type	Int	N	查询单号类型	1: 订单 2: 发货单

5.5.6.4. 应用级返回结果

result _entity

名称	类型	必填?	说明	备注
orderId	String	Y	查询单号	如果发生拆单，则需要传子订单号

				如果进一步拆成发货单，则传发货单号
type	Int	N	查询单号类型	1: 订单 2: 发货单
orderTrack	orderTrack_entity[]	Y	配送信息	配送信息描述: 展示的是物流配送的过程(包含: 时间和阶段) 如无法提供详细阶段, 请至少保证该接口能返回(签收时间和 配送员的姓名、联系方式), 方便客户进行电话咨询
waybillCode	waybillCode_entity	N	运单信息	

orderTrack_entity[]

名称	类型	必填?	说明	备注
msgTime	String	Y	时间	
content	String	Y	配送信息内容	
operator	String	N	操作员	

waybillCode_entity

名称	类型	必填?	说明	备注
carrier	String	Y	承运商	
deliveryOrderId	String	Y	运单号	

5.5.6.5. 示例

5.5.6.5.1. 请求示例

```
{  
  "appkey": "CGDC",
```

```
{
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "sku": "cOKD31n20140101101937816x93z92,cOKD31n20140101101937816x93z91"
}
```

5.5.6.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "成功",
  "resultCode": "0000",
  "result": {
    "orderId": "111111",
    "orderTrack": [{
      "msgTime": "2013-09-25 09:03:53",
      "content": "您提交了订单，请等待系统确认",
      "operator": "客户"
    }],
    "waybillCode": {
      "carrier": "",
      "deliveryOrderId": ""
    }
  }
}
```

5.5.6.6. 备注说明

5.5.7. 查询发货单信息接口（针对有发货单的电商）

5.5.7.1. 应用场景

根据订单号查询订单的配送信息，按照时间给配送信息明细。

5.5.7.2. 请求说明

API 名称	package
调用入口	http://{url}/package
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

5.5.7.3. 应用级输入参数

名称	类型	必填?	说明	备注
packageld	String	Y	发货单号	

5.5.7.4. 应用级返回结果

result _entity

名称	类型	必填?	说明	备注
packageld	String	Y	发货单号	
orderId	String	Y	发货单归属电商订单号	
deliveryCode	String	Y	物流单号	
deliverytime	String	Y	发货时间	
deliveryName	String	Y	物流公司名称	
deliveryItems	deliveryItems_entity[]	Y	发货单明细	

deliveryItems_entity[]

名称	类型	必填?	说明	备注
skuld	String	Y	商品编码	
num	Int	Y	商品数量	

5.5.7.5. 示例

5.5.7.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "packageld": "0987654321"
}
```

5.5.7.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "成功",
  "resultCode": "0000",
  "result": {
    "packageld": "987654321",
    "orderId": "20170915090912345678",
    "deliveryCode": "7293728024802858",
    "deliverytime": "2017-08-27 09:10:12",
    "deliveryName": "顺丰",
    "deliveryItems": cOKD31n20140101101937816x93z92,
  }
}
```

```
{
  [
    {
      "skuld": "12345678",
      "num": 5
    }
  ]
}
```

5.5.7.6. 备注说明

5.5.8. 订单完结通知

5.5.8.1. 应用场景

电子超市网上商城的订单确认收货和验收后，将订单最终完结状态通知到电商。

5.5.8.2. 请求说明

API 名称	orderCompletion
调用入口	http://{url}/orderCompletion
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

5.5.8.3. 应用级输入参数

名称	类型	必填?	说明	备注
orderId	String	Y	订单号	如果拆单则传子订单号，如果未拆单则传未拆单的订单号
type	Integer	N	查询单号类型	不传时为 1 1：订单 2：发货单

5.5.8.4. 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 true 即可
- 如果失败，则 result 返回 false，resultCode 返回失败码，resultMessage 会返回失败原因

5.5.8.5. 示例

5.5.8.5.1. 请求示例

```
{  
  
  "appkey": "CGDC",  
  
  "serialNumber": "sjd79and83cgem",  
  
  "timeStamp": "2017-08-19 10:13:01",  
  
  "token": "7886f145d87cb5d3219e8291c225f9c8",  
  
  "orderId": "20170915090912345678"  
  
}
```

5.5.8.5.2. 返回示例

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

5.5.8.6. 备注说明

5.5.9. 取消订单的部分明细（针对有发货单的电商，未产生发货单的明细进行取消）

5.5.9.1. 应用场景

目前有发货单流程的供应商会在发货的时候推送发货单消息，来表示这些明细已经发货，而云商平台所针对发货单消息做的处理就是进行了订单拆分。

针对发货单的妥投，拒收消息我们都能够正常处理，但是未产生发货消息的明细，客户会存在取消的需求，所以需要这个接口来进行取消部分明细操作。

5.5.9.2. 请求说明

API 名称	cancelPart
调用入口	http://{url}/cancelPart
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

5.5.9.3. 应用级输入参数

名称	类型	必填?	说明	备注
orderId	String	Y	电商订单号	
detail	detail_Entity[]	Y	需要取消的明细	

detail_Entity[] 需要取消的明细

名称	类型	必填?	说明	备注
skuld	String	Y	商品编码	
skuNum	Int	Y	商品数量	

5.5.9.4. 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 true 即可
- 如果失败，则 result 返回 false，resultCode 返回失败码，resultMessage 会返回失败原因

5.5.9.5. 示例

5.5.9.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "orderId": "20178282739719172",
  "detail": [{
    "skuld": "ADY811",
    "skuNum": 2
  }]
}
```

5.5.9.5.2. 返回示例

```
{
  "success": true,
  "resultMessage": "取消订单成功",
  "resultCode": "0",
  "result": true
}
```

5.5.9.6. 备注说明

resultCode

0	成功
1	失败
101	重复取消

5.6.信息推送

5.6.1. 获取推送消息接口

5.6.1.1. 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 skuid、变化的订单 id 写入消息表，明确变化类型，需要电子超市定时获取变化信息。

消息池建议最大 100 条数据，当需要推送的消息多余 100 条时，将多余的消息缓存到队列中，当消息池中消息被删除后数量小于 100 条时，再将队列中缓存的消息放到消息池中。

5.6.1.2. 请求说明

API 名称	get
调用入口	http://{url}/get
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

5.6.1.3. 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

5.6.1.4. 应用级返回结果

result _entity[]

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

id	String	Y	推送 id	
time	String	Y	推送时间	格式为: yyyy-MM-dd HH:mm:ss
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
pOrder	String	Y	父订单 id	

每拆一次单都会发送一次拆单消息, 但父订单号只会传递原始父订单, 需要通过查询订单信息接口获取到最新所有子单, 进行相关更新;

➤ type=2 代表商品价格变更

result_entity

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

➤ 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
skuld	String	Y	商品编号	通过上下级状态接口获取状态

➤ type=5 代表订单已妥投/拒收

result_entity

名称	类型	必填?	说明	备注
orderId	String	Y	订单编号	按照（子）订单发货的，传（子）订单编号；按照发货单发货的，传发货单号；如果此时传订单号，则表示该订单下所有发货单均收货；
orderType	Int	N	类型	1：订单 2：发货单 不传默认是订单
state	Int	Y	妥投状态	1 是妥投，2 是拒收，3 是部分妥投(3 针对拆分发货单)

对于拆分发货单的订单，需要在所有发货单都妥投或者拒收后，电商将该订单的状态写入消息池，该订单有妥投、拒收和部分妥投三种状态；如果发货单全部妥投则传“妥投”，如果发货单全部拒收则传“拒收”，如果部分妥投部分拒收则传“部分妥投”；

➤ 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
skuld	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

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=10 代表订单取消

订单取消包括两种情况，一是订单满 7 天未确认自动取消，另一个是订单商品电商因为自身原因无法发货而取消。

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	0: 7 天未确认自动取消 1: 电商因为自身原因无法发货而取消 不填默认为 0
cancelReason	String	N	取消原因	简要描述取消原因，对于 cancelType=0 的可不传

注：妥投之前的订单（预订单，确认订单，已发货订单）供应商可以取消订单，必须推送取消消息，强调发货中的订单，必须和用户确认后才可取消

➤ 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
skuld	String	Y	商品编号	

➤ type=17 代表赠品促销变更消息

result_entity

名称	类型	必填?	说明	备注
skuld	String	Y	商品编号	

➤ type=101 代表存在发货单

result_entity

名称	类型	必填?	说明	备注
packageld	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=102 代表售后服务单完成

result_entity

名称	类型	必填?	说明	备注
serviceld	String	Y	售后单号	
state	Int	Y	成功/失败	1 代表退货成功 2 代表退货失败，即原返

5.6.1.5. 示例

5.6.1.5.1. 请求示例

```
{
  "appkey": "CGDC",
  "serialNumber": "sjd79and83cgem",
  "timeStamp": "2017-08-19 10:13:01",
  "token": "7886f145d87cb5d3219e8291c225f9c8",
  "type": 16
}
```

5.6.1.5.2. 返回示例

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

```
"resultMessage": "查询成功",

"result": [{

    "id": "63628929",

    "time": "2019-03-25 18:27:02",

    "type": 16,

    "result": {

        "skuld": "RGN072"

    }

}]

}
```

5.6.1.6. 备注说明

5.6.2. 删除消息接口

5.6.2.1. 应用场景

电子超市获取消息并处理成功后，需要主动将推送消息删除。

5.6.2.2. 请求说明

API 名称	delete
调用入口	http://{url}/delete
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

5.6.2.3. 应用级输入参数

名称	类型	必填?	说明	备注
id	String	Y	推送消息 id	

5.6.2.4. 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 true 即可
- 如果失败，则 result 返回 false，resultCode 返回失败码，resultMessage 会返回失败原因

5.6.2.5. 示例

5.6.2.5.1. 请求示例

```
{  
  
  "appkey": "CGDC",  
  
  "serialNumber": "sjd79and83cgem",  
  
  "timeStamp": "2017-08-19 10:13:01",  
  
  "token": "7886f145d87cb5d3219e8291c225f9c8",  
  
  "id": "cOKD31n20140101101937816x93z92"  
  
}
```

5.6.2.5.2. 返回示例

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

5.6.2.6. 备注说明

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	入参转化错误		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

电子超市对外交互规范

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		接口调用失败, 可重试 (下游下单)

电子超市对外交互规范

	交订单		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	
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		

电子超市对外交互规范

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		