

# 快递接口 (API) 技术文档

## 版本

日期	版本	说明	作者
2014-12-01	3.0	重新整理，根据功能分类，部分有调整	杨文胜
2014-01-11	3.1	对接口的说明进行了注释，提高可读性	杨汉汶
2014-01-13	3.2	新增推送接口	杨汉汶
2015-03-11	3.3	新增批量订阅推送接口，删除 3.2 版本推送接口	杨文胜
2015-10-12	3.4	新增订单分发接口，整理电子面单接口	曹强
2016-01-19	3.5	重新调整在线下单接口	曹强
2016-03-01	3.6	增加智选物流接口	曹强

## 名词定义

必须要求	说明
R	必填 (Required)。
O	可选 (Optional)
C	报文中该参数在一定条件下可选 (Conditional)

## 前言

本文档目的是定义快递鸟的通用对外服务接口，以便客户接入快递鸟API，从而打通客户与快递鸟系统之间的信息流，实现整体物流信息化，进而达到快递鸟与客户的双赢。

版本.....	2
名词定义.....	2
前言.....	3
一、智选物流 .....	7
1、功能说明.....	7
2、接口规则.....	7
3、系统级输入参数.....	7
4、应用级输入参数.....	7
5、返回结果参数.....	9
二、在线下单接口 .....	9
1、功能说明.....	9
2、接口规则.....	9
3、系统级输入参数.....	10
4、应用级输入参数.....	10
5、返回结果参数.....	12
6、Json 请求示例.....	13
7、Json 请求返回示例.....	14
三、取消订单 .....	14
1、功能说明.....	14
2、接口规则.....	14
3、系统级输入参数.....	15
4、应用级输入参数.....	15
5、返回结果参数.....	15
6、XML 请求示例.....	15
7、JSON 请求示例 .....	16
8、XML 返回示例.....	16
9、JSON 返回示例 .....	16
四、物流轨迹查询 .....	17
1、功能说明.....	17
2、接口规则.....	17
3、系统级输入参数.....	17
4、应用级输入参数.....	17
5、返回结果参数.....	18
6、XML 请求示例.....	18
7、JSON 请求示例 .....	18
8、XML 返回示例.....	19
9、JSON 返回示例 .....	20
五、绑定运单 .....	22
1、功能说明.....	22
2、接口规则.....	22

3、系统级输入参数.....	22
4、应用级输入参数.....	22
5、返回结果参数.....	23
6、XML 请求示例.....	23
7、JSON 请求示例 .....	23
8、XML 响应示例.....	23
9、JSON 响应示例 .....	24
六、运单订阅 .....	24
1、功能说明.....	24
2、接口规则.....	24
3、系统级输入参数.....	24
4、应用级输入参数.....	25
5、返回结果参数.....	25
6、JSON 请求示例 .....	25
7、JSON 返回示例 .....	26
8、XML 请求示例.....	27
9、XML 返回示例.....	27
七、推送接口 .....	28
1、功能说明.....	28
2、接口规则.....	28
3、系统级输入参数.....	28
4、应用级输入参数.....	28
5、返回结果参数.....	30
6、JSON 请求示例 .....	30
7、JSON 返回示例 .....	33
八、电子面单 .....	33
1、功能说明.....	33
2、接口规则.....	33
3、系统级输入参数.....	33
4、应用级输入参数.....	34
5、返回结果参数.....	36
6、JSON 请求示例 .....	37
7、JSON 返回示例 .....	38
九、分发及订阅 .....	39
1、功能说明.....	39
2、接口规则.....	39
3、系统级输入参数.....	39
4、应用级输入参数.....	39
5、返回结果参数.....	41
6、JSON 请求示例 .....	42
7、JSON 返回示例 .....	43

十、快递公司编码 .....	43
1、支持快递下单的公司 .....	43
十一、使用说明 .....	45
1、支持 http 传输协议 .....	45
2、支持的数据传输格式 .....	45
3、接入步骤 .....	45
4、接口数据包结构 .....	46
5、XML 范例/JSON 范例 .....	46
6、流程示意图 .....	47
7、网关地址 .....	47
十二、签名说明 .....	47
1、关于签名 .....	47
2、(C#)DataSign 签名加密代码 .....	48
十三、附录 .....	49
1、增值服务列表 .....	49
2、返回编码列表 .....	50
十四、技术支持 .....	50

## 一、智选物流(即将上线)

### 1、功能说明

同一线路不同快递公司的时效对比，帮助商家选择服务能力好的快递公司，可根据快递不到区域，运费、时效、服务等综合判断最优快递。

### 2、接口规则

- (1)、接口只支持 **Json** 格式参数，接口指令 **2001**。只支持 POST 请求方式，utf-8 编码。
- (2)、支持增值服务，保价、代收货款、货款直退、货款垫付等。
- (3)、测试地址：<http://112.74.112.208:8085/api/exrecommend/>
- (3)、正式地址：<http://api.kdniao.cc/api/exrecommend/>

### 3、系统级输入参数

参数名称	类型	是否必须	描述
RequestData	String	R	请求内容，JSON 或 XML 格式, 须和 DataType 一致
EBusinessID	String	R	电商 ID
RequestType	String	R	请求指令类型： <b>2001</b>
DataSign	String	R	数据内容签名
DataType	String	0	请求、返回数据类型：1-xml, 2-json；默认为 xml 格式(本接口只支持 Json 格式)

### 4、应用级输入参数

	参数名称	类型	是否必须	描述
Receiver	ProvinceName	String	R	收件省（如广东省，不要缺少“省”）
	CityName	String	R	收件市（如深圳市，不要缺少

				“市”)
	ExpAreaName	String	R	收件区（如福田区，不要缺少“区”或“县”）
	Subdistrict	String	0	收件街道（如华强北街道）
	Address	String	0	收件人详细地址
Sender	ProvinceName	String	R	派件省（如广东省，不要缺少“省”）
	CityName	String	R	派件市（如深圳市，不要缺少“市”）
	ExpAreaName	String	R	派件区（如福田区，不要缺少“区”或“县”）
	Subdistrict	String	0	派件街道（如华强北街道）
	Address	String	0	派件人详细地址
Volume		Double	0	物品总体积 m3
Weight		Double	0	物品总重量 kg
GoodsCategory		String	0	商品类目
IsCOD		Bool	0	是否代收货款：0. 否，1. 是
ExpressList		List	0	商家合作快递公司列表
SortBy		Integer	0	排序依据：0. 综合，1. 时效优先，2. 价格优先

	参数名称	类型	是否必须	描述
ExpressList	ExpCode	String	R	快递公司编码
	First	Double	R	首重价格(元)
	Additional	Double	R	首重价格(元)



## 5、返回结果参数

参数名称	类型	是否必须	描述
ExpressList	List	R	推荐快递公司列表

	参数名称	类型	是否必须	描述
ExpressList	ExpCode	String	R	快递公司编码
	ExpName	String	R	快递公司名称
	Fee	Double	0	总运费（元）
	EstimatedDeliveryTime	Integer	0	时效（预计到达时长:小时）
	IsSupport	String	R	是否支持派件区域（不到:0. 不到, 1. 到）

## 二、在线下单接口

### 1、功能说明

在线下单接口是快递鸟提供给独立电商、仓储管理系统、物流供应链等物流系统平台使用的下单接口，为客户解决在线发货需求，客户通过网络选择快递公司发送请求通知贵公司有快递要发货，贵公司把数据通过此接口转发到快递鸟，由快递鸟为您安排快递员上门取件的服务。

### 2、接口规则

- (1)、订单编号(OrderCode)不可重复提交，重复提交系统会返回具体错误代码。
- (2)、支持主流快递公司在线下单，请联系技术人员(联系方式请看文档尾部)。
- (3)、接口只支持 **Json** 格式参数，接口指令 **1001**。只支持 POST 请求方式，utf-8 编码。
- (4)、支持增值服务，保价、代收货款、货款直退、货款垫付等。
- (5)、测试地址：<http://112.74.112.208:8085/api/oorderservice/>

### 3、系统级输入参数

参数名称	类型	是否必须	描述
RequestData	String	R	请求内容, JSON 或 XML 格式, 须和 DataType 一致
EBusinessID	String	R	电商 ID
RequestType	String	R	请求指令类型: 1001
DataSign	String	R	数据内容签名
DataType	String	0	请求、返回数据类型: 1-xml, 2-json; 默认为 xml 格式(本接口只支持 Json 格式)

### 4、应用级输入参数

参数名称	类型	是否必须	描述
WarehouseID	String	0	仓库标识(备用字段)
WarehouseAddress	String	0	仓库地址(备用字段)
CallBack	String	0	商户标识(备用字段)
MemberID	String	0	会员标识(备用字段)
ShipperCode	String	R	快递公司编码
LogisticCode	String	0	快递单号
OrderCode	String	R	订单编号
MonthCode	String	C	月结编码
PayType	Int	R	邮费支付方式: 1-现付, 2-到付, 3-月结, 4-第三方支付
ExpType	Int	R	快递类型: 1-标准快件

Cost		Double	0	寄件费（运费）
OtherCost		Double	0	其他费用
Receiver	Company	String	0	收件人公司
	Name	String	R	收件人
	Tel	String	R	电话与手机，必填一个
	Mobile	String		
	PostCode	String	0	收件人邮编
	ProvinceName	String	R	收件省（如广东省，不要缺少“省”）
	CityName	String	R	收件市（如深圳市，不要缺少“市”）
	ExpAreaName	String	0	收件区（如福田区，不要缺少“区”或“县”）
	Address	String	R	收件人详细地址
Sender	Company	String	0	发件人公司
	Name	String	R	发件人
	Tel	String	R	电话与手机，必填一个
	Mobile	String		
	PostCode	String	0	发件人邮编
	ProvinceName	String	0	发件省（如广东省，不要缺少“省”）
	CityName	String	R	发件市（如深圳市，不要缺少“市”）
	ExpAreaName	String	0	发件区（如福田区，不要缺少“区”或“县”）
	Address	String	R	发件详细地址
StartDate		String	0	上门取货时间段：
EndDate		String	0	“yyyy-MM-dd HH:mm:ss”格式

				化，本文中所有时间格式相同
Weight		Double	0	物品总重量 kg
Quantity		Int	0	件数/包裹数
Volume		Double	0	物品总体积 m3
Remark		String	0	备注
AddService	Name	String	0	<a href="#">增值服务名称</a>
	Value	String	0	增值服务值
	CustomerID	String	0	客户标识(选填)
Commodity	GoodsName	String	R	商品名称
	GoodsCode	String	0	商品编码
	Goodsquantity	Int	0	件数
	GoodsPrice	Double	0	商品价格
	GoodsWeight	Double	0	商品重量 kg
	GoodsDesc	String	0	商品描述
	GoodsVol	Double	0	商品体积 m3

## 5、返回结果参数

参数名称		类型	必须要求	说明
EBusinessID		String	R	电商用户 ID
Order	OrderCode	String	R	订单编号
	ShipperCode	String	R	快递公司编码
	LogisticCode	String	0	快递单号
Success		Bool	R	成功与否
ResultCode		String	R	<a href="#">结果编码</a>
Reason		String	0	失败原因

UniquerRequestNumber	String	R	唯一标识
----------------------	--------	---	------

## 6、Json 请求示例

```
{
  "OrderCode": "012657018199",
  "ShipperCode": "SF",
  "PayType": 1,
  "MonthCode": "7553045845",
  "ExpType": 1,
  "Cost": 1.0,
  "OtherCost": 1.0,
  "Sender": {
    "Company": "LV",
    "Name": "Taylor",
    "Mobile": "15018442396",
    "ProvinceName": "上海",
    "CityName": "上海",
    "ExpAreaName": "青浦区",
    "Address": "明珠路"
  },
  "Receiver": {
    "Company": "GCCUI",
    "Name": "Yann",
    "Mobile": "15018442396",
    "ProvinceName": "北京",
    "CityName": "北京",
    "ExpAreaName": "朝阳区",
    "Address": "三里屯街道"
  },
  "Commodity": [
    {
      "GoodsName": "鞋子",
      "Goodsquantity": 1,
      "GoodsWeight": 1.0
    }
  ],
  "AddService": [
    {
      "Name": "COD",
      "Value": "1020"
    }
  ]
}
```

```
}  
],  
"Weight": 1.0,  
"Quantity": 1,  
"Volume": 0.0,  
"Remark": "小心轻放"  
}
```

## 7、Json 请求返回示例

```
{  
  "EBusinessID": "1237100",  
  "Success": true,  
  "Order": {  
    "OrderCode": "012657018199",  
    "ShipperCode": "SF",  
    "LogisticCode": ""  
  },  
  "ResultCode": "100",  
  "Reason": ""  
}
```

## 三、取消订单

### 1、功能说明

物流公司还未指派业务员去取件的物流订单可进行取消操作，撤销时需用户填写原因，填写后快递鸟会通知物流公司取消订单。

### 2、接口规则

- (1)、接口支持 Json/xml 格式。
- (2)、接口指令 1004。

### 3、系统级输入参数

参数名称	类型	必须要求	说明
RequestData	String	R	请求内容, JSON 或 XML 格式, 须和 DataType 一致
EBusinessID	String	R	电商 ID
RequestType	String	R	请求指令类型: 1004
DataSign	String	R	数据内容签名
DataType	String	0	请求、返回数据类型: 1-xml, 2-json; 默认为 xml 格式

### 4、应用级输入参数

参数名称	类型	必须要求	说明
OrderCode	String	R	订单编号
ShipperCode	String	R	物流公司编码
LogisticCode	String	R	物流单号

### 5、返回结果参数

参数名称	类型	必须要求	说明
EBusinessID	String	R	电商用户 ID
OrderCode	String	R	订单编号
ShipperCode	String	R	快递公司编码
LogisticCode	String	R	快递单号
Success	Bool	R	成功与否: true, false
Reason	String	0	失败原因

### 6、XML 请求示例

```
<Content>
  <OrderCode>201406240005</OrderCode>
  <ShipperCode>ZJS</ShipperCode>
  <LogisticCode>118650888088</LogisticCode>
```

```
</Content>
```

## 7、JSON 请求示例

```
{
  "OrderCode": "201406240005",
  "ShipperCode": "ZJS",
  "LogisticCode": "118461988807"
}
```

## 8、XML 返回示例

```
<Response>
  <EBusinessID>1109259</EBusinessID>
  <OrderCode>201406240005</OrderCode>
  <ShipperCode>ZJS</ShipperCode>
  <LogisticCode>118650888088</LogisticCode>
  <Success>true</Success>
  <Reason></Reason>
</Response>
```

## 9、JSON 返回示例

```
{
  "EBusinessID": "1109259",
  "OrderCode": "201406240005",
  "ShipperCode": "ZJS",
  "LogisticCode": "118461988807",
  "Success": false,
  "Reason": " 订单取消失败"
}
```



## 四、物流轨迹查询

### 1、功能说明

物流轨迹查询-使用的物流单号和快递单号即可实现查询物流信息。

### 2、接口规则

- (1)、查询接口支持按照运单号查询(单个查询)。
- (2)、指定的物流运单号选择相应的快递公司编码，格式不对或则编码错误都会返失败的信息。如 EMS 物流单号应选择快递公司编码（EMS）
- (3)、返回的物流跟踪信息按照发生的时间升序排列。
- (4)、接口指令 1002。

### 3、系统级输入参数

参数名称	类型	必须要求	说明
RequestData	String	R	请求内容，JSON 或 XML 格式, 须和 DataType 一致
EBusinessID	String	R	电商 ID
RequestType	String	R	请求指令类型：1002
DataSign	String	R	数据内容签名
DataType	String	0	请求、返回数据类型：1-xml, 2-json; 默认为 xml 格式

### 4、应用级输入参数

参数名称	类型	必须要求	说明
OrderCode	String	0	订单编号
ShipperCode	String	R	快递公司编码
LogisticCode	String	R	物流单号

## 5、返回结果参数

参数名称	类型	必须要求	说明
EBusinessID	String	R	电商用户 ID
OrderCode	String	O	订单编号
ShipperCode	String	R	快递公司编码
LogisticCode	String	R	物流运单号
CallBack	String	O	用户标识
Success	Bool	R	成功与否
Reason	String	O	失败原因
State	String	R	物流状态：2-在途中, 3-签收, 4-问题件
Traces/物流轨迹详情			
AcceptTime	String	R	时间
AcceptStation	String	R	描述
Remark	String	O	备注

## 6、XML 请求示例

```
<Content>
  <OrderCode></OrderCode>
  <ShipperCode>ZT0</ShipperCode>
  <LogisticCode>851245789632</LogisticCode>
</Content>
```

## 7、JSON 请求示例

```
{
  "OrderCode": "",
  "ShipperCode": "SF",
  "LogisticCode": "118650888018"
}
```

## 8、XML 返回示例

没有轨迹

```
<Response>
  <EBusinessID>1109259</EBusinessID>
  <OrderCode />
  <ShipperCode>SF</ShipperCode>
  <LogisticCode>118650888088</LogisticCode>
  <Success>false</Success>
  <Traces></Traces>
</Response>
```

有物流轨迹

```
<Response>
  <EBusinessID>1109259</EBusinessID>
  <OrderCode>
    <OrderCode />
  <ShipperCode>SF</ShipperCode>
  <LogisticCode>118650888018</LogisticCode>
  <Success>true</Success>
  <State>3</State>
  <Traces>
    <Trace>
      <AcceptTime>2014/04/09 20:23:53</AcceptTime>
      <AcceptStation>已收件[深圳市]</AcceptStation>
    </Trace>
    <Trace>
      <AcceptTime>2014/04/09 21:08:47</AcceptTime>
      <AcceptStation>快件在 深圳 ,准备送往下一站 深圳集散中心 [深圳
市]</AcceptStation>
    </Trace>
    <Trace>
      <AcceptTime>2014/04/10 01:45:45</AcceptTime>
      <AcceptStation>快件在 深圳集散中心 [深圳市]</AcceptStation>
    </Trace>
    <Trace>
      <AcceptTime>2014/04/10 04:42:20</AcceptTime>
      <AcceptStation>快件在 深圳集散中心 ,准备送往下一站 深圳 [深圳
市]</AcceptStation>
    </Trace>
    <Trace>
      <AcceptTime>2014/04/10 08:10:31</AcceptTime>
```

```
<AcceptStation>正在派件..(派件人:邓裕富,电话:18718866310)[深圳
市]</AcceptStation>
</Trace>
<Trace>
  <AcceptTime>2014/04/10 09:49:39</AcceptTime>
  <AcceptStation>派件已签收[深圳市]</AcceptStation>
</Trace>
<Trace>
  <AcceptTime>2014/04/10 09:49:00</AcceptTime>
  <AcceptStation>签收人是: **[深圳市]</AcceptStation>
</Trace>
</Traces>
</OrderCode>
</Response>
```

## 9、JSON 返回示例

### 没有物流轨迹的

```
{
  "EBusinessID": "1109259",
  "Traces": [],
  "OrderCode": "",
  "ShipperCode": "SF",
  "LogisticCode": "118461988807",
  "Success": false,
  "Reason": null
}
```

### 有物流轨迹的

```
{
  "EBusinessID": "1109259",
  "OrderCode": "",
  "ShipperCode": "SF",
  "LogisticCode": "118461988807",
  "Success": true,
  "CallBack": "",
  "State": 3,
  "Reason": null,
  "Traces": [
    {
      "AcceptTime": "2014/06/25 08:05:37",
      "AcceptStation": "正在派件..(派件人:邓裕富,电话:18718866310)[深圳
```

```
市]",
    "Remark": null
  },
  {
    "AcceptTime": "2014/06/25 04:01:28",
    "AcceptStation": "快件在 深圳集散中心 ,准备送往下一站 深圳 [深圳
市]",
    "Remark": null
  },
  {
    "AcceptTime": "2014/06/25 01:41:06",
    "AcceptStation": "快件在 深圳集散中心 [深圳市]",
    "Remark": null
  },
  {
    "AcceptTime": "2014/06/24 20:18:58",
    "AcceptStation": "已收件[深圳市]",
    "Remark": null
  },
  {
    "AcceptTime": "2014/06/24 20:55:28",
    "AcceptStation": "快件在 深圳 ,准备送往下一站 深圳集散中心 [深圳
市]",
    "Remark": null
  },
  {
    "AcceptTime": "2014/06/25 10:23:03",
    "AcceptStation": "派件已签收[深圳市]",
    "Remark": null
  },
  {
    "AcceptTime": "2014/06/25 10:23:03",
    "AcceptStation": "签收人是: 已签收[深圳市]",
    "Remark": null
  }
]
}
```

## 五、绑定运单

### 1、功能说明

客户把订单号与物流单号通过接口推送到快递鸟，快递鸟会及时跟进这些物流单，物流轨迹有变化系统会发送数据给客户，通知轨迹有更新(本接口供使用快递鸟在线下单用户)。

### 2、接口规则

- (1)、此功能只有部分物流公司支持，后续会支持更多物流公司的绑单。
- (2)、接口支持 Json/xml 格式。
- (3)、接口指令 1003。

### 3、系统级输入参数

参数名称	类型	必须要求	说明
RequestData	String	R	请求内容，JSON 或 XML 格式, 须和 DataType 一致
EBusinessID	String	R	电商 ID
RequestType	String	R	请求指令类型：1003
DataSign	String	R	数据内容签名
DataType	String	0	请求、返回数据类型：1-xml, 2-json; 默认为 xml 格式

### 4、应用级输入参数

参数名称	类型	必须要求	说明
OrderCode	String	R	订单编号
ShipperCode	String	R	快递公司编码
LogisticCode	String	R	物流单号

## 5、返回结果参数

参数名称	类型	必须要求	说明
EBusinessID	String	R	电商用户 ID
OrderCode	String	R	订单编号
ShipperCode	String	R	快递公司编码
LogisticCode	String	R	物流单号
Success	Bool	R	成功与否
Reason	String	0	失败原因

## 6、XML 请求示例

```
<Content>
  <OrderCode>201406240002</OrderCode>
  <ShipperCode>ZJS</ShipperCode>
  <LogisticCode>109932607888</LogisticCode>
</Content>
```

## 7、JSON 请求示例

```
{
  "OrderCode": "201406240002",
  "ShipperCode": "ZJS",
  "LogisticCode": "118461988807"
}
```

## 8、XML 响应示例

```
<Response>
  <EBusinessID>1109259</EBusinessID>
  <OrderCode>201406240002</OrderCode>
  <LogisticCode>109932607888</LogisticCode>
  <Success>true</Success>
  <Reason></Reason>
</Response>
```

## 9、JSON 响应示例

```
{
  "EBusinessID": "1109259",
  "OrderCode": "201406240002",
  "ShipperCode": "ZJS",
  "LogisticCode": "118461988807",
  "Success": true,
  "Reason": ""
}
```

## 六、运单订阅

### 1、功能说明

此功能为专业定制，为特殊商户提供订阅服务，快递鸟系统会定时推送新的物流信息到商户系统。

### 2、接口规则

- (1)、支持 Json/xml 格式。
- (2)、请求指令 1005。

### 3、系统级输入参数

参数名称	类型	必须要求	说明
RequestData	String	R	请求内容，JSON 或 XML 格式，须和 DataType 一致
EBusinessID	String	R	电商 ID
RequestType	String	R	请求指令类型：1005
DataSign	String	R	数据内容签名
DataType	String	0	请求、返回数据类型：1-xml, 2-json；默认为 xml 格式



## 4、应用级输入参数

参数名称	类型	必须要求	说明
Code	String	R	快递公司编码
Item	String	R	物流单号集合

参数名称		类型	必须要求	说明
Item	No	String	R	快递物流单号
	Bk	String	0	用户标识
	MemberID	String	0	会员标识
	WarehouseID	String	0	仓库标识
	WarehouseAddress	String	0	仓库地址

## 5、返回结果参数

参数名称	类型	必须要求	说明
EBusinessID	String	R	电商用户 ID
UpdateTime	String	0	时间
Success	String	R	成功与否:true, false
Reason	String	0	备注

## 6、JSON 请求示例

```
[
  {
    "Code": "YTO",
    "Item": [
      {
        "Bk": "",
        "No": "880143779586547836",
        "MemberID": "TB92578",
        "WarehouseID": "SZ86547836",
        "WarehouseAddress": "皇岗-01"
      },
      {
        "Bk": "",
        "No": "880143779586547837"
      }
    ]
  }
]
```

```
    }  
  ]  
},  
{  
  "Code": "SF",  
  "Item": [  
    {  
      "Bk": "",  
      "No": "589871688930"  
    },  
    {  
      "Bk": "",  
      "No": "589871688931"  
    }  
  ]  
}  
]
```

## 7、JSON 返回示例

```
{  
  "EBusinessID": "1151847",  
  "UpdateTime": "2015-03-13 16:42:38",  
  "Success": true,  
  "Reason": ""  
}
```

## 8、XML 请求示例

```
<Content>
  <ExpressCodes>
    <Code>YD</Code>
    <Items>
      <Item>
        <No>1901085671075</No>
        <Bk></Bk>
      </Item>
      <Item>
        <No>1000509345244</No>
        <Bk></Bk>
      </Item>
    </Items>
  </ExpressCodes>
  <ExpressCodes>
    <Code>SF</Code>
    <Items>
      <Item>
        <No>589707398027</No>
        <Bk></Bk>
      </Item>
      <Item>
        <No>589852819866</No>
        <Bk></Bk>
      </Item>
    </Items>
  </ExpressCodes>
</Content>
```

## 9、XML 返回示例

```
<Response>
  <EBusinessID>1237100</EBusinessID>
  <UpdateTime>2015/5/28 11:28:05</UpdateTime>
  <Success>true</Success>
  <Reason />
</Response>
```

## 七、推送接口

### 1、功能说明

客户通过订阅接口，把运单号在快递鸟系统订阅。运单物流信息有更新，快递鸟通过推送接口把更新的信息推送给客户。推送接口由用户按照快递鸟规则实现。商户开放接口，快递鸟定时将最新的物流轨迹推送到商户平台。优点：客户不需要关注轨迹变化，一旦有更新，快递鸟自动推送。缺点：对客户的服务器性能有所要求，快递鸟定时请求接口，对客户的服务器有一定的影响，客户服务器异常将接受不到最新数据。

### 2、接口规则

(1)、客户服务器必须稳定，并且有一定的接受数据能力。快递鸟通过多线程推送物流信息给客户。

(2)、客户需要按快递鸟要求开发接口，保证信息的正常接收。客户接数据后，存储并马上返回接收响应。如果客户对数据进行处理，在再做出响应。这会造成网络超时，传输效率地下。

(3)、主动推送时物流信息接收接口由客户提供，接口必须按照快递鸟的要求实现。

(4)、**客户提供接口地址**，请联系快递鸟技术人员，联系方式请求查看文档尾页。

(5)、接口只支持 Json 数据格式。

(6)、POS 方式请求。

### 3、系统级输入参数

参数名称	类型	必须要求	说明
RequestData	String	R	推送数据经 url 转码 utf-8

### 4、应用级输入参数

参数名称	类型	必须要求	说明
EBusinessID	String	R	用户电商 ID
PushTime	String	R	推送时间
Count	String	R	推送物流单号轨迹个数
Data	String	R	推送物流单号轨迹集合

参数名称		类型	必须要求	说明
Response	EBusinessID	String	0	电商用户 ID
	OrderCode	String	0	订单编号
	ShipperCode	String	R	快递公司编码
	LogisticCode	String	R	物流运单号
	Success	Bool	R	成功与否
	Reason	String	0	失败原因
	State	String	R	物流状态：2-在途中，3-签收,4-问题件
	CallBack	String	0	订阅接口的 Bk 值
	Traces	String	R	物流轨迹详情
	EstimatedDeliveryTime	String	0	预计到达时间 yyyy-mm-dd
	PickerInfo	Object	0	收件员信息
	SenderInfo	Object	0	派件员信息
	Location	String	0	当前所在省市 例：广东省, 深圳市

注：通过订单分发与电子面单接口自动订阅的运单，会返回预计达到时间, 收派件人员网点信息, 当前快件所在省市等字段

参数名称		类型	必须要求	说明
Traces	AcceptTime	String	R	时间
	AcceptStation	String	R	描述
	Remark	String	0	备注

参数名称		类型	必须要求	说明
PickerInfo 与 SenderInfo	PersonName	String	0	快递员姓名
	PersonTel	String	0	快递员电话
	PersonCode	String	0	快递员工号
	StationName	String	0	网点名称
	StationAddress	String	0	网点地址
	StationTel	String	0	网点电话

## 5、返回结果参数

参数名称	类型	必须要求	说明
EBusinessID	String	R	用户 ID
UpdateTime	String	R	时间
Success	String	R	成功与否
Reason	String	0	失败原因

## 6、JSON 请求示例

```
{
  "EBusinessID": "1109259",
  "Count": "2",
  "PushTime": "2015-03-11 16:21:06",
  "Data": [
    {
      "EBusinessID": "1109259",
      "OrderCode": "",
      "ShipperCode": "EMS",
      "LogisticCode": "5042260908504",
      "Success": true,
      "Reason": "",
      "State": "2",
      "CallBack": "0",
      "Traces": [
        {
          "AcceptTime": "2015-03-06 21:16:58",
          "AcceptStation": "深圳市横岗速递营销部已收件, (揽投员姓名: 钟定基;联系电话: )",
          "Remark": ""
        },
        {
          "AcceptTime": "2015-03-07 14:25:00",
          "AcceptStation": "离开深圳市 发往广州市",
          "Remark": ""
        },
        {
          "AcceptTime": "2015-03-08 00:17:00",
          "AcceptStation": "到达广东速递物流公司广航中心处理中心 (经转)"
        }
      ]
    }
  ]
},
```

```
        "Remark": ""
    },
    {
        "AcceptTime": "2015-03-08 01:15:00",
        "AcceptStation": "离开广州市 发往北京市（经转）",
        "Remark": ""
    },
    {
        "AcceptTime": "2015-03-09 09:01:00",
        "AcceptStation": "到达北京黄村转运站处理中心（经转）",
        "Remark": ""
    },
    {
        "AcceptTime": "2015-03-09 18:39:00",
        "AcceptStation": "离开北京市 发往呼和浩特市（经转）",
        "Remark": ""
    },
    {
        "AcceptTime": "2015-03-10 18:06:00",
        "AcceptStation": "到达 呼和浩特市 处理中心",
        "Remark": ""
    },
    {
        "AcceptTime": "2015-03-11 09:53:48",
        "AcceptStation": "呼和浩特市邮政速递物流分公司金川揽投部安排  
投递（投递员姓名：安长虹;联系电话：18047140142）",
        "Remark": ""
    }
]
},
{
    "EBusinessID": "1109259",
    "OrderCode": "",
    "ShipperCode": "EMS",
    "LogisticCode": "5042260943004",
    "Success": true,
    "Reason": "",
    "State": "2",
    "CallBack": "0",
    "Traces": [
        {
            "AcceptTime": "2015-03-07 15:26:09",
```

```
"AcceptStation": "深圳市横岗速递营销部已收件，（揽投员姓名：周宏彪;联系电话：13689537568）",
  "Remark": ""
},
{
  "AcceptTime": "2015-03-08 16:32:00",
  "AcceptStation": "离开深圳市 发往广州市",
  "Remark": ""
},
{
  "AcceptTime": "2015-03-09 00:58:00",
  "AcceptStation": "到达广东速递物流公司广航中心处理中心（经转）",
  "Remark": ""
},
{
  "AcceptTime": "2015-03-09 01:15:00",
  "AcceptStation": "离开广州市 发往北京市（经转）",
  "Remark": ""
},
{
  "AcceptTime": "2015-03-10 05:20:00",
  "AcceptStation": "到达北京黄村转运站处理中心（经转）",
  "Remark": ""
},
{
  "AcceptTime": "2015-03-10 11:59:00",
  "AcceptStation": "离开北京市 发往廊坊市（经转）",
  "Remark": ""
},
{
  "AcceptTime": "2015-03-10 14:23:00",
  "AcceptStation": "到达廊坊市处理中心（经转）",
  "Remark": ""
},
{
  "AcceptTime": "2015-03-11 08:55:00",
  "AcceptStation": "离开廊坊市 发往保定市（经转）",
  "Remark": ""
}
]
```



```
}  
}
```

## 7、JSON 返回示例

```
{  
  "EBusinessID": "1109259",  
  "UpdateTime": "2015-03-11 16: 26: 11",  
  "Success": true,  
  "Reason": ""  
}
```

## 八、电子面单

### 1、功能说明

电子面单接口是快递鸟提供给独立电商、仓储管理系统、物流供应链等物流系统平台使用的下单接口，为客户解决在线发货需求，客户通过网络选择物流公司发送请求通知贵公司有快递要发货，贵公司把数据通过此接口转发到快递鸟，由快递鸟为您安排快递员上门取件的服务。

电子面单成功下单后，快递鸟系统内将自动订阅，一旦轨迹发生变化时，快递鸟将调用客户方推送接收接口

### 2、接口规则

- (1)、只支持 **Json** 格式。接口指令 1007。
- (2)、测试地址：<http://testapi.kdniao.cc:8081/api/eorderservice/>

### 3、系统级输入参数

参数名称	类型	必须要求	说明
RequestData	String	R	请求内容，JSON 或 XML 格式，须和 DataType 一致
EBusinessID	String	R	电商 ID
RequestType	String	R	请求指令类型： <b>1007</b>
DataSign	String	R	数据内容签名
DataType	String	0	请求、返回数据类型：1-xml, 2-json;

			默认为 xml 格式
--	--	--	------------

#### 4、应用级输入参数

参数名称		类型	是否必须	描述
CallBack		String	0	用户自定义回调信息
MemberID		String	0	会员标识 平台方与快递鸟统一用户标识的商家 ID
CustomerName		String	0	电子面单客户账号 (与快递网点申请)
CustomerPwd		String	0	电子面单密码
SendSite		String	0	收件网点标识
ShipperCode		String	R	快递公司编码
LogisticCode		String	0	快递单号
OrderCode		String	R	订单编号
MonthCode		String	C	月结编码
PayType		Int	R	邮费支付方式： 1-现付，2-到付，3-月结，4-第三方支付
ExpType		String	R	快递类型：1-标准快件
Cost		Double	0	寄件费（运费）
OtherCost		Double	0	其他费用
Receiver	Company	String	0	收件人公司
	Name	String	R	收件人
	Tel	String	R	电话与手机，必填一个
	Mobile	String		
	PostCode	String	0	收件人邮编

	ProvinceName	String	R	收件省（如广东省，不要缺少“省”）
	CityName	String	R	收件市（如深圳市，不要缺少“市”）
	ExpAreaName	String	0	收件区（如福田区，不要缺少“区”或“县”）
	Address	String	R	收件人详细地址
Sender	Company	String	0	发件人公司
	Name	String	R	发件人
	Tel	String	R	电话与手机，必填一个
	Mobile	String		
	PostCode	String	0	发件人邮编
	ProvinceName	String	0	发件省（如广东省，不要缺少“省”）
	CityName	String	R	发件市（如深圳市，不要缺少“市”）
	ExpAreaName	String	0	发件区（如福田区，不要缺少“区”或“县”）
	Address	String	R	发件详细地址
StartDate		String	0	上门取货时间段：
EndDate		String	0	“yyyy-MM-dd HH:mm:ss”格式化，本文中所有时间格式相同
Weight		Double	0	物品总重量 kg
Quantity		Int	0	件数/包裹数
Volume		Double	0	物品总体积 m3
Remark		String	0	备注
AddService	Name	String	0	<a href="#">增值服务名称</a>
	Value	String	0	增值服务值

	CustomerID	String	0	客户标识(选填)
Commodity	GoodsName	String	R	商品名称
	GoodsCode	String	0	商品编码
	Goodsquantity	Int	0	件数
	GoodsPrice	Double	0	商品价格
	GoodsWeight	Double	0	商品重量 kg
	GoodsDesc	String	0	商品描述
	GoodsVol	Double	0	商品体积 m3
IsReturnPrintTemplate		String	0	返回电子面单模板： 0-不需要；1-需要

## 5、返回结果参数

### ● 同步返回

参数名称		类型	必须要求	说明
EBusinessID		String	R	电商用户 ID
Order	OrderCode	String	R	订单编号
	ShipperCode	String	R	快递公司编码
	LogisticCode	String	R	快递单号
	MarkDestination	String	0	大头笔
	OriginCode	String	0	始发地区域编码
	OriginName	String	0	始发地/始发网点
	DestinationCode	String	0	目的地区域编码
	DestinationName	String	0	目的地/到达网点
	SortingCode	String	0	分拣编码
	PackageCode	String	0	集包编码
Success		Bool	R	成功与否

ResultCode	String	R	<a href="#">错误编码</a>
Reason	String	0	失败原因
UniquerRequestNumber	String	R	唯一标识
PrintTemplate	String	0	面单打印模板
EstimatedDeliveryTime	String	0	订单预计到货时间 yyyy-mm-dd
Callback	String	0	用户自定义回调信息

## 6、JSON 请求示例

```
{
  "OrderCode": "012657700387",
  "ShipperCode": "HTKY",
  "PayType": 1,
  "ExpType": 1,
  "Cost": 1.0,
  "OtherCost": 1.0,
  "Sender": {
    "Company": "LV",
    "Name": "Taylor",
    "Mobile": "15018442396",
    "ProvinceName": "上海",
    "CityName": "上海",
    "ExpAreaName": "青浦区",
    "Address": "明珠路 73 号"
  },
  "Receiver": {
    "Company": "GCCUI",
    "Name": "Yann",
    "Mobile": "15018442396",
    "ProvinceName": "北京",
    "CityName": "北京",
    "ExpAreaName": "朝阳区",
    "Address": "三里屯街道雅秀大厦"
  },
  "Commodity": [
    {
      "GoodsName": "鞋子",
```

```
        "Goodsquantity": 1,  
        "GoodsWeight": 1.0  
    },  
    ],  
    "AddService": [  
        {  
            "Name": "COD",  
            "Value": "1020"  
        }  
    ],  
    "Weight": 1.0,  
    "Quantity": 1,  
    "Volume": 0.0,  
    "Remark": "小心轻放",  
    "IsReturnPrintTemplate":1  
}
```

## 7、JSON 返回示例

```
{  
    "EBusinessID": "1237100",  
    "Order": {  
        "OrderCode": "012657700387",  
        "ShipperCode": "HTKY",  
        "LogisticCode": "50002498503427",  
        "MarkDestination": "京-朝阳(京-1)",  
        "OriginCode": "200000",  
        "OriginName": "上海分拨中心",  
        "PackageCode": "北京"  
    },  
    "PrintTemplate": "此处省略打印模板 HTML 内容",  
    "EstimatedDeliveryTime": "2016-03-06",  
    "Callback": "调用时传入的 Callback",  
    "Success": true,  
    "ResultCode": "100",  
    "Reason": "成功"  
}
```

## 九、分发及订阅

### 1、功能说明

订单分发接口是快递鸟提供给独立电商、仓储管理系统、物流供应链等物流系统平台使用的下单接口，为客户解决在线发货需求，客户通过网络选择物流公司发送请求通知贵公司有快递要发货，贵公司把数据通过此接口转发到快递鸟，由快递鸟为您安排快递员上门取件的服务。

### 2、接口规则

(1)、订单分发的信息(分给了的网点，业务员的信息)，会通过推送接口推给客户。客户需要按要求实现接口。

(2)、只是支持 **Json** 格式。

(2)、请求指令 1006。

### 3、系统级输入参数

参数名称	类型	必须要求	说明
RequestData	String	R	请求内容，JSON 或 XML 格式，须和 DataType 一致
EBusinessID	String	R	电商 ID
RequestType	String	R	请求指令类型：1006
DataSign	String	R	数据内容签名
DataType	String	0	请求、返回数据类型：1-xml, 2-json；默认为 xml 格式

### 4、应用级输入参数

参数名称	类型	是否必须	描述
WarehouseID	String	0	仓库标识
WarehouseAddress	String	0	仓库地址
MemberID	String	0	会员标识
ShipperCode	String	R	快递公司编码

LogisticCode	String	0	快递单号
CallBack	String	0	商户标识(商户自定义)
OrderCode	String	R	订单编号
OrderType	Int	R	运单类型: 1-电子运单, 2-纸质运单
PayType	Int	R	邮费支付方式: 1-现付, 2-到付
IsNeedPay	Int	R	是否代收货款: 1-是, 2-否
Payment	Decimal	C	代收货款金额
ToCompany	String	0	收件人公司
ToName	String	R	收件人
ToPostCode	String	0	收件人邮编
ToProvinceID	String	R	收件省(如广东省, 不要缺少“省”)
ToCityID	String	R	收件市(如深圳市, 不要缺少“市”)
ToExpAreaID	String	R	收件区(如福田区, 不要缺少“区”或“县”)
ToAddressArea	String	R	收件人详细地址
ToTel	String	0	电话和手机至少要一个
ToMobile	String	0	
FromCompany	String	0	发件人公司
FromName	String	R	发件人
FromPostCode	String	0	发件人邮编
FromProvinceID	String	R	发件省(如广东省, 不要缺少“省”)
FromCityID	String	R	发件市(如深圳市, 不要缺少“市”)
FromExpAreaID	String	R	发件区(如福田区, 不要缺少“区”或“县”)
FromAddressArea	String	R	发件人详细地址
FromTel	String	0	电话和手机至少要一个
FromMobile	String	0	



Cost		Decimal	0	寄件费（运费）
OtherCost		String	0	其他费用
StartDate		String	0	物流公司上门取货时间段， “yyyy-MM-dd HH:mm:ss” 格式化， 本文中所有时间格式相同
EndDate		String	0	
Weight		Decimal	0	物品总重量
Volume		Decimal	0	物品总体积
Remark		String	0	备注
AddService	Name	String	0	<a href="#">增值服务名称</a>
	Value	String	0	增值服务值
	Customer ID	String	0	客户标识(选填)
Commoditys/商品信息				
GoodsName		String	0	商品名称
GoodsCode		String	0	商品编码
Goodsquantity		Int	0	商品数量
GoodsPrice		Decimal	0	商品价格
GoodsWeight		Decimal	0	商品重量
GoodsDesc		String	0	商品描述

## 5、返回结果参数

### ● 同步返回

参数名称	类型	必须要求	说明
EBusinessID	String	R	电商用户 ID
OrderCode	String	R	订单编号
Success	Bool	R	成功与否:true, false
Reason	String	0	失败原因
EstimatedDeliveryT	String	0	订单预计到货时间 yyyy-mm-dd

ime			
Callback	String	0	用户自定义回调信息

注：订单分发会自动触发运单订阅推送

## 6、JSON 请求示例

```
{
  "WareHouseID": "TB001",
  "WareHouseAddress": "广东省深圳市福田区赛格广场 54 楼 5401",
  "MemberID": "8383383",
  "ShipperCode": "ZTO",
  "LogisticCode": "4126316756",
  "CallBack": "TB/3939839",
  "OrderCode": "zto001",
  "OrderType": 1,
  "PayType": 1,
  "IsNeedPay": 1,
  "Payment": 100,
  "ToCompany": "LV",
  "ToName": "Taylor",
  "ToPostCode": "525359",
  "ToProvinceID": "广东省",
  "ToCityID": "茂名市",
  "ToExpAreaID": "信宜市",
  "ToAddressArea": "华侨中学",
  "ToTel": "06688799562",
  "ToMobile": "15018442396",
  "FromCompany": "GUCL",
  "FromName": "Yann",
  "FromPostCode": "518000",
  "FromProvinceID": "广东省",
  "FromCityID": "深圳市",
  "FromExpAreaID": "福田区",
  "FromAddressArea": "赛格广场 54 楼 5401",
  "FromTel": "07555687945",
  "FromMobile": "18826732224",
  "Cost": 23,
  "OtherCost": 39,
  "StartDate": "2015-01-12 23:00:00",
  "EndDate": "2015-11-12 23:00:00",
}
```

```
"Weight": 2,
"Volume": 3,
"Remark": "工作日送货",
"Commodities": [
  {
    "GoodsDesc": "小心轻放",
    "GoodsName": "IPad Air2",
    "GoodsPrice": 3588
  },
  {
    "GoodsDesc": "小心轻放",
    "GoodsName": "Surface 4",
    "GoodsPrice": 6588
  }
]
```

## 7、JSON 返回示例

```
失败:
{
  "EBusinessID": "1109259",
  "OrderCode": "1155552111444211",
  "Success": "false",
  "Reason": "订单编号重复"
}
成功:
{
  "EBusinessID": "1109259",
  "OrderCode": "1155552111444211",
  "Success": "true",
  "Reason": ""
}
```

## 十、快递公司编码

### 1、支持快递下单的公司

物流公司编码	物流公司名称	支持返回快递单号
--------	--------	----------

UAPEX	全一快递	N
SF	顺丰速运	N
LB	龙邦速递	N
FAST	快捷速递	N
NEDA	港中能达	N
ZHQKD	汇强快递	N
ZTO	中通速递	N
GTO	国通快递	N
QFKD	全峰快递	N
ZJS	宅急送	N
DBL	德邦物流	Y
YTO	圆通速递	Y
HTKY	汇通快运	Y
HOAU	天地华宇	Y
UC	优速快递	Y

以上编码同样支持物流轨迹查询，支持快递查询的国内外物流公司编码详情请与技术对接人索取。

在线快递公司编码列表：<http://www.kdniao.com/file/ExpressCode.xls>

## 十一、使用说明

### 1、支持 http 传输协议

### 2、支持的数据传输格式

Json/XML(暂不支持<content/>这种形式)。

编码格式：UTF-8

交互协议上统一用 UTF-8，避免传递中文数据出现乱码。

### 3、接入步骤

**注册成为快递鸟用户**(注：注册成功选择服务立即使用，无需审批)。注册成功后，登录就可以查看 ID 和 key。

注册方式通过网络平台：

<http://www.kdniao.com/ServiceApply.aspx>（官网 API 介绍）。

按照快递鸟官网提示进行注册申请，注册成功并且登录，您将得到有快递鸟提供一组电商 ID 和 AppKey。电商 ID 是调用接口服务的身份证明，不可更改、不可转用，AppKey 是应用访问 API 的签名附加密钥，必须妥善保管。两者关系类似于登录帐号和密码，两者都会在签名和业务参数中使用。

注册成功登录之后：进入我的服务查看 ID 和 AppKey

 快递鸟

首页 解决方案 应用示例 业务资源 技术

我的服务

商户ID：API Key：

说明：API Key是平台为应用分配的密钥，用于保证应用来源的可靠性，避免应用被伪造，被不法人员使用。

(1) 开发服务

开发自身业务的服务，服务需按照后面接口的定义规则。

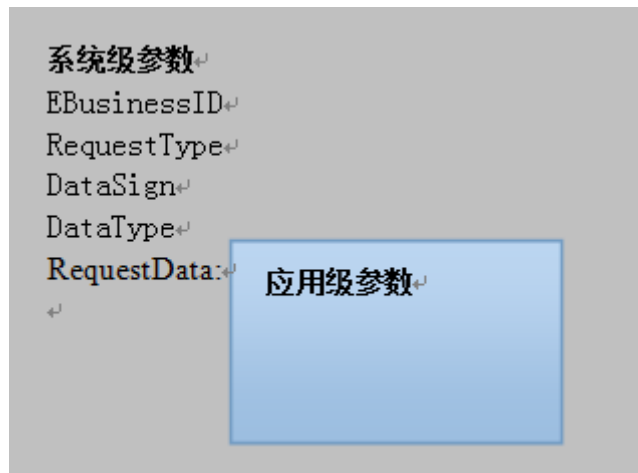
(2) 查看接口调用的 DEMO

快递鸟物流技术接口提供调用快递鸟物流接口的 DEMO(.Net 版本)，包括网上在线订单下单、物流轨迹查询等物流行业标准化的下单操作流程与数据查询格式。提交接入处理的相关代码如有不明白的地方可与快递鸟物流技术接口人协作开发。

DEMO 下载地址：<http://www.kdniao.com/ResourceDownload.aspx> (如链接无法打开，请打开浏览器，在浏览器中输入)

## 4、接口数据包结构

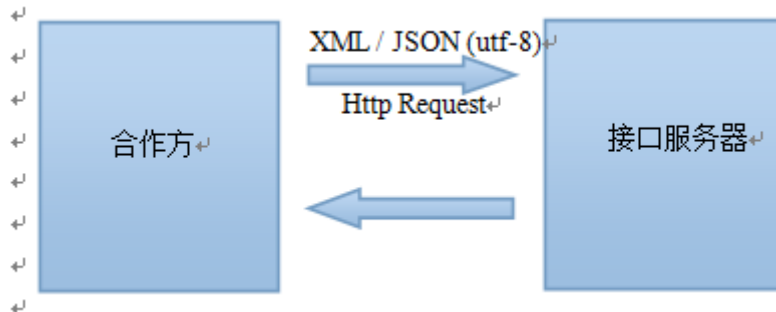
图例 - 数据包结构 (系统级{数据})



## 5、XML 范例/JSON 范例

```
string userID = "1109259";
string keyValue = "56da2cf8-c8a2-44b2-b6fa-476cd7d1ba17";//加密私钥，由发货通提供
string url = "http://api.kdniao.cc:8081/Ebusiness/EbusinessOrderHandle.aspx";
//请求地址
string DataType = "1"; //1-xml;2-json
string charset="UTF-8";//字符编码采用UTF-8
string xmlStr =
"<Content><OrderCode></OrderCode><ShipperCode>SF</ShipperCode><LogisticCode>118650888088</LogisticCode></Content>";// XML字符串
string xmlStr =
"{\"OrderCode\": \"\", \"ShipperCode\": \"SF\", \"LogisticCode\": \"118461988807\"}";
// JSON字符串
string datasign = MD5(xmlStr+keyValue, "UTF-8");//生成加密签名加密通过对参数xmlStr+ keyValue得到加密后的字符串
string PostStr = "RequestType=1002&EBusinessID= userID &RequestData=xmlStr &DataSign=datasign&DataType=DataType"; //请求报文参数
string post = this.DoPost(url, PostStr);// 通讯协议使用Http协议Post请求方式
```

## 6、流程示意图



## 7、网关地址

测试接口: <http://api.kdniao.cc:8081/Ebusiness/EbusinessOrderHandle.aspx>

测试电商 ID==1237100, AppKey==518a73d8-1f7f-441a-b644-33e77b49d846

正式接口: <http://api.kdniao.cc/Ebusiness/EbusinessOrderHandle.aspx>

注意: 正式(自行注册)环境下使用时, 需提供请求的 IP 给快递鸟登记

# 十二、签名说明

## 1、关于签名

快递鸟和第三方电子商务公司系统进行对接, 有一定的安全机制。采用 IP 认证加签名的方式对接, 具体方案如下:

### (1) 防止数据被篡改

在 POST 请求中会传递 5 个参数

4 个必须参数:

RequestType=请求指令类型

EBusinessID==电商 ID

RequestData==数据内容(URL 编码:UTF-8)

DataSign== 签名摘要(URL 编码:UTF-8(base64(MD5(数据内容 +AppKey))))

1 个可选参数:

DataType==返回数据类型(1-xml, 2-json; 默认为 2 格式)

注: DataSign 生成后, 对方接收到数据后, 以同样的算法进行签名, 生成摘要, 对比两者的摘要是否相同, 如果不同, 说明传递过程中发生数据篡改, 对接过程中如有不明白之处可双方技术协作进行

### (2) 调用接口的身份认证

注册成为快递鸟用户后, 会生成对应的电商 ID 和 AppKey, 电商 ID 相当于用户名, AppKey 相当于密码。注: 请勿遗漏或传递可能递增大快递鸟接口调用维护

举例说明：

(a) 假设 RequestData (XML)内容为：

<Content><ShipperCode>SF</ShipperCode><LogisticCode>118954907573</LogisticCode></Content>，经过 URL(UTF-8)编码的内容为%3cContent%3e%3cShipperCode%3eSF%3c%2fShipperCode%3e%3cLogisticCode%3e118954907573%3c%2fLogisticCode%3e%3c%2fContent%3e， EBusinessID 为 1109259， AppKey 为 56da2cf8-c8a2-44b2-b6fa-476cd7d1ba17

(b) 那么 DataSign 签名的内容为

<Content><ShipperCode>SF</ShipperCode><LogisticCode>118954907573</LogisticCode></Content>56da2cf8-c8a2-44b2-b6fa-476cd7d1ba17，经过 md5 和 base64 后的内容就为 YzBjODA3MjA2NmViM2ZlZGM5NWNjMDAONDRiMDdmOTE=，在经过 URL(UTF-8)编码的内容为 YzBjODA3MjA2NmViM2ZlZGM5NWNjMDAONDRiMDdmOTE%3d

(c) 最终要发送的数据为 RequestType=1002&EBusinessID=1109259&RequestData

=%3cContent%3e%3cShipperCode%3eSF%3c%2fShipperCode%3e%3cLogisticCode%3e118954907573%3c%2fLogisticCode%3e%3c%2fContent%3e &DataSign = YzBjODA3MjA2NmViM2ZlZGM5NWNjMDAONDRiMDdmOTE%3d& DataType=2

(d) 接收方收到数据后，获得 EBusinessID 和 RequestData 和 DataSign 等这几个数据。

(e) 接收方对 EBusinessID 得到 AppKey， RequestData+AppKey 的数据进行 md5 和 base64 后的内容就为 YzBjODA3MjA2NmViM2ZlZGM5NWNjMDAONDRiMDdmOTE=

(f) 接收方判断签名后的数据跟传递过来的 DataSign 是否一致，如果一致进行业务操作，如果不一直返回错误。

## 2、(C#)DataSign 签名加密代码

```
///<summary>
///电商 Sign 签名
///</summary>
///<param name="content">内容</param>
///<param name="keyValue">Appkey</param>
///<param name="charset">URL 编码 </param>
///<returns>DataSign 签名</returns>
Public String Encrypt (String content, String keyValue, String charset)
{
    if (keyValue != null)
    {
        return base64(MD5(content + keyValue, charset), charset);
    }
    return base64(MD5(content, charset), charset);
}

///<summary>
/// 字符串 MD5 加密
```



```
///</summary>
///<param name="Text">要加密的字符串</param>
///<returns>密文</returns>
Private string MD5(string Text, string charset)
{
    byte[] buffer = System.Text.Encoding.GetEncoding(charset).GetBytes(Text);
    try
    {
        System.Security.Cryptography.MD5CryptoServiceProvider check;
        check = new System.Security.Cryptography.MD5CryptoServiceProvider();
        byte[] somme = check.ComputeHash(buffer);
        string ret = "";
        foreach (byte a in somme)
        {
            if (a < 16)
                ret += "0" + a.ToString("X");
            else
                ret += a.ToString("X");
        }
        return ret.ToLower();
    }
    catch
    {
        throw;
    }
}

Private static string base64(String str, String charset)
{
    return Convert.ToBase64String(System.Text.Encoding.GetEncoding(charset).GetBytes(str));
}
```

## 十三、附录

### 1、增值服务列表

名称	代码	说明	备注
代收货款	COD	Value 为货款，保留3 位小数。 CustomerID 为代收货款卡号	

保价	INSURE	value 为声明价值，保留3 位小数	
----	--------	---------------------	--

## 2、返回编码列表

编码	信息
100	成功
101	缺少必要参数
102	校验问题
103	格式问题
104	用户问题
105	其他错误

## 十四、技术支持

联系人	负责工作	电话/手机	QQ	邮箱
任海洋	技术负责	18620300500	993200	renhaiyang@kdniao.com
曹强	技术对接	18826732223	1069712970	caoqiang@kdniao.com