

# IPOS-SA

## IOT支付版



微信扫一扫  
文档随身带

# 目 录

## 文档说明

### 1.引言

1.1.文档概述

1.2.交互模式

### 2.接入方式及报文格式规定

2.1.JSON格式

2.2.XML格式

### 3.iPos接口说明

3.3.获取设备绑定状态与POS信息

3.2.获取设备授权码交易

3.1.终端绑定交易

3.4.条码支付交易

3.5.支付查询交易

3.6.支付退款交易

3.7.退款查询交易

3.8.支付撤销交易

3.9.银行卡刷卡下单交易

3.10.银行卡刷卡结果查询交易

3.11.银行卡刷卡申请退款交易

3.12.银行卡刷卡退款查询交易

### 4.iPos接口接入报文规范

4.1.终端绑定交易

4.3.获取设备绑定状态与POS信息

4.2.获取设备授权码交易

4.4.条码支付交易

4.5.支付查询交易

4.6.支付退款交易

4.7.退款查询交易

4.8.支付撤销交易

4.9.银行卡刷卡下单交易

4.10.银行卡刷卡结果查询交易

4.11.银行卡申请退款交易

4.12.银行卡刷卡退款查询交易

### 5.错误码说明

# 文档说明

## 文档概述

本文档适用于外部平台以iPOS方式接入翼码业务支撑平台进行微信、支付宝条码支付的接口文档，适用于第三方自己有POS收银系统的接入平台。

## 版本说明

本文档《iPos-SA-IOT支付版》由 陈炯烁 编写，最后编辑于 2020-11-16 10:42:42。

## 发布日期

2020-12-11

## 在线文档

<https://open.wangcaio2o.com/docs-ipos-sa-iot.html>

## 注意事项

如果当前文档生成时间太久，请到 [翼码开放平台\(open.wangcaio2o.com\)](https://open.wangcaio2o.com) 获取最新的文档，以免信息不对称!

如果您发现文档内容有不恰当的地方，请向我们 [反馈意见](#)。

感谢您使用 [翼码开放平台\(open.wangcaio2o.com\)](https://open.wangcaio2o.com)。

# 1.引言

---

- 1.1.文档概述
- 1.2.交互模式

## 1.1.文档概述

本文档适用于外部平台以iPOS方式接入翼码业务支撑平台进行微信、支付宝条码支付的接口文档，适用于第三方自己有POS收银系统的接入平台，报文格式支持Json或Xml

本文档中所有请求节点若需传入XML解析器无法解析的特殊字符可使用CDATA标签包裹节点内容传入，例如：`<xml><![CDATA[节点内容]]></xml>`

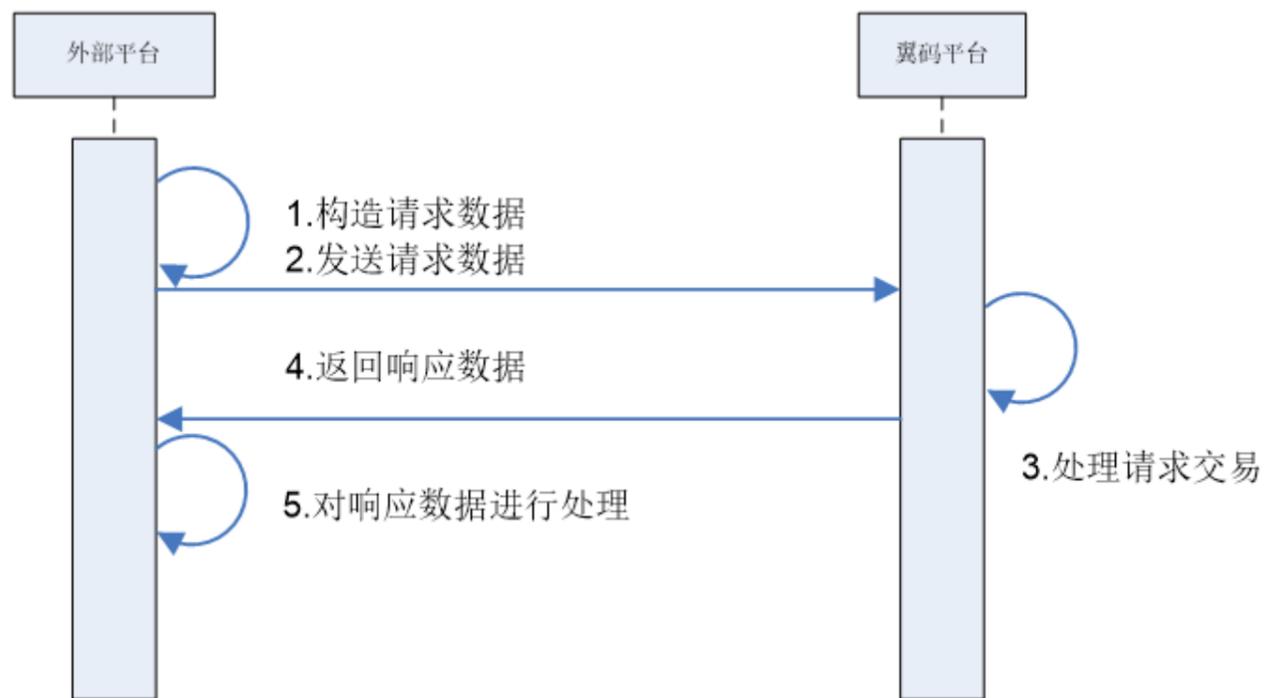
**重要：**  
标准接口会因业务升级需要而新增节点，  
请在解析返回报文时避免新增未知节点产生的程序出错！

## 1.2.交互模式

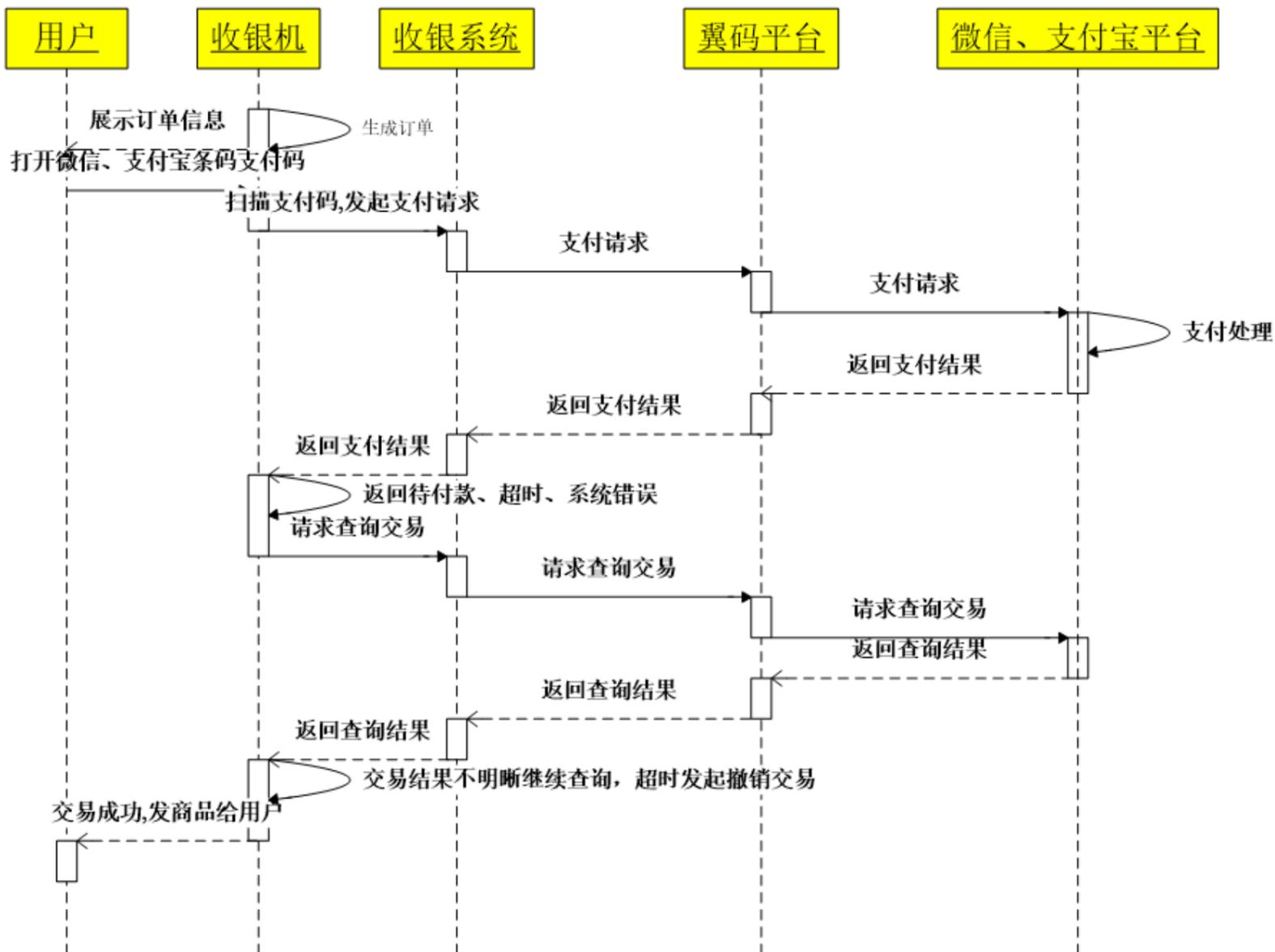
### 请求/响应交互模式

请求/响应模式是最常见的一种交互模式。在这种交互模式下，外部平台向翼码平台提供的HTTPS URL，发送请求数据，并同步等待翼码平台处理完毕之后，返回的响应数据。

### 处理流程



### 流程描述



翼码提供给商户的4个接口：绑定、支付、查询、退款/撤销。每个接口都有其特定的使用场景。绑定是后续交易的前提；支付是核心业务；而合理利用好查询和退款/撤销接口，可以有效的避免繁琐的对账工作。

按照支付宝和微信支付的建议，遇到订单状态为未支付（9998）、网络异常等情况，需要通过查询接口来确定订单的最新状态。

目前支付宝和微信支付提供了查询接口，翼码封装了这些查询接口，形成了统一的查询接口。商户只需要调用翼码的查询接口即可获得对应支付交易的最新状态。

支付交易返回成功或者失败的时候，一般可以认为支付交易的结果是明确的，返回0000为成功，9990为失败，这种情况下一般无需调用查询接口。

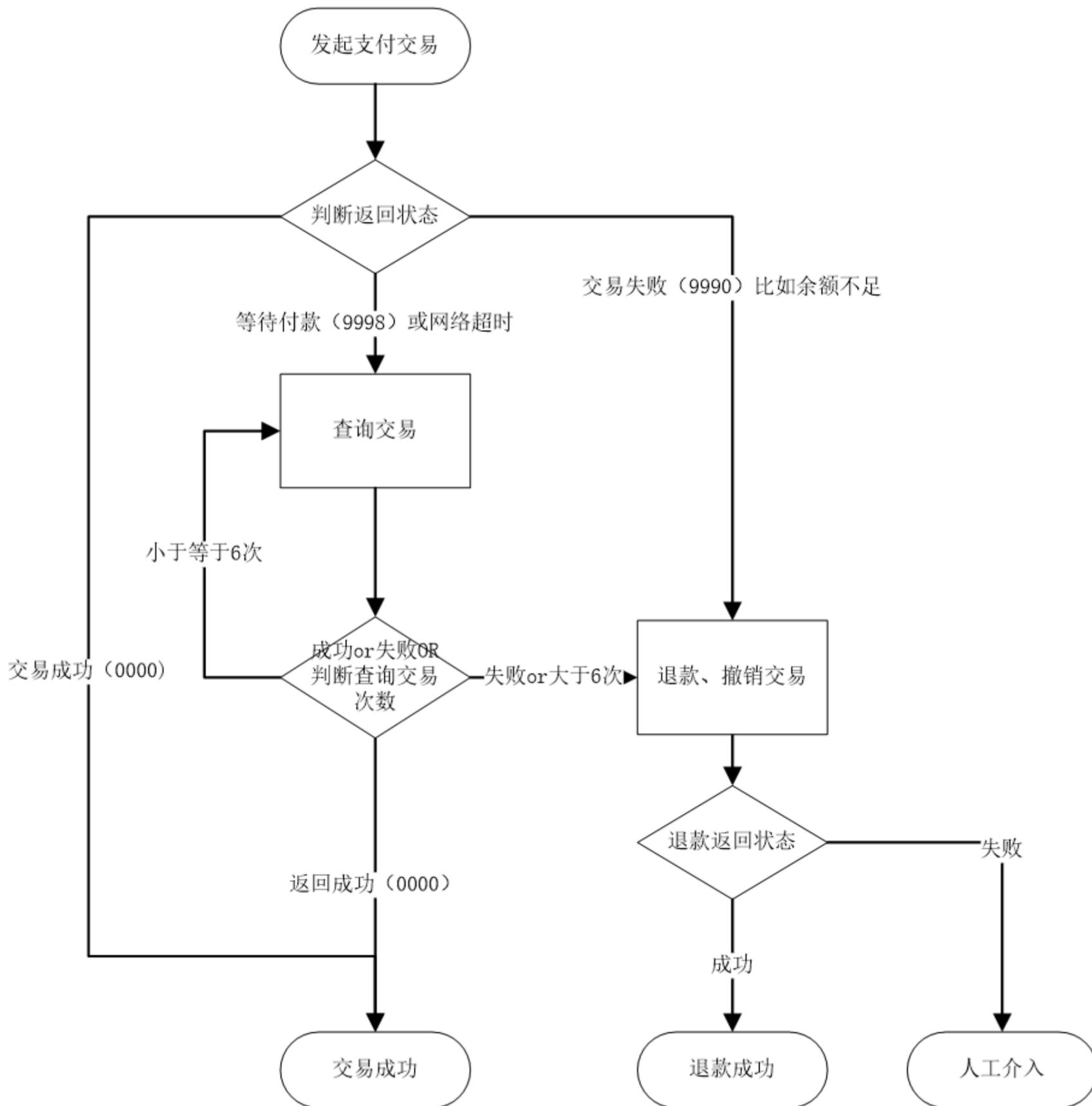
支付交易返回超时（接口返回码9998，需密支付等待付款的情况也为9998）或其他异常情况，查询接口调用间隔为5秒，调用的总次数建议为12次。

查询交易调用超过12次后依然返回9998，请务必立即调用退款/撤销接口进行支付交易订单的关闭，这点很重要，因为如果没有调用退款/撤销接口，那么消费者有可能在超过1分钟后去支付该订单，而商户在此之后没有进行查询的话，

就会导致商户的订单为“未支付”，而客户实际上已支付，给商户的对账造成很大的麻烦以及加大了客服的解释工作。

对支付宝和微信异常导致了客户实际上支付成功但是返回给翼码/商户的为支付失败、以及退款成功但是返回给翼码/商户的为退款失败，

上述情况非常罕见，但是也不排除可能性，这时依旧需要调用查询接口来确定最新的状态。



#### 终端绑定交易流程描述

1.收银系统向翼码平台发起终端绑定交易，绑定报文中一些重要参数：

商户号——由翼码平台提供。

系统号——由翼码平台提供（同一服务商下属多商户可共用一套系统号与密钥）。

终端号——收银系统上收银机或终端的编号。

门店号——收银系统上收银机或终端所在门店编号（若收银系统没有门店概念，即一个门店仅有一个终端不存在多终端的情况，门店号可与终端号相同）。

**注：商户号、终端号、门店号会在翼码平台确定一台唯一终端，绑定成功后，在支付、查询、撤销、日结等交易报文中填入相应参数即表示此终端做出的交易请求。**

2.翼码平台在接收到收银系统绑定终端请求后会进行报文参数校验、商户权限校验，校验通过后根据商户号、终端号和门店号的组合查询是否绑定过终端（即参数所对应的翼码终端是否存在），若绑定过终端则返回交易成功给收银系统。若尚未绑定过终端则进行绑定终端处理，在翼码平台上开设所传入参数对应的终端，并返回信息给收银系统。

3.收银系统根据响应码判断终端是否绑定成功：

(1) 当接收到的响应码为“0000”时表示交易成功，即商户号、终端号、门店号对应的翼码终端开设成功。

(2) 当接收到的响应码为“9990”时表示交易失败，可根据响应码解释进行相应处理，本次绑定交易失败需重新发起绑定。

### 支付交易流程描述

- 1.收银机结算后生成订单，向用户展示订单信息。
- 2.用户确认订单后，打开支付宝、微信等支付工具的条码支付功能在手机屏幕上显示出相应支付条码。
- 3.营业员使用扫码枪等工具扫描用户手机上的付款码，向收银系统发起支付请求。
- 4.收银系统收到收银机请求后，组装对应报文向翼码平台发起支付请求，等待返回报文。
- 5.翼码平台在接收到收银系统的请求后会进行报文参数校验、商户权限校验、IP安全校验等，校验通过后向支付宝、微信等支付平台发起支付请求，并在平台上记录相应流水信息。
- 6.支付宝、微信等支付平台进行支付订单的处理，将结果返回翼码平台。
- 7.翼码平台接收支付平台返回的支付结果进行解析后，将处理过的支付结果信息返回给收银系统。
- 8.收银系统接收到翼码平台的返回报文，将信息返回给收银机。
- 9.收银机接收到支付结果信息后，对信息中的响应码进行判断并做出相应处理：
  - (1) 当接收到的响应码为“0000”时表示交易成功，即可将商品给用户，完成本次交易。
  - (2) 当接收到的响应码为“9990”时表示交易失败，可根据响应码解释进行相应处理，本次交易付款失败不能把商品给用户。
  - (3) 当接收到的响应码为“9998”时表示等待用户付款（当用户支付工具设置为非免密支付或处理时间超长会出现此种情况），收银机需循环调用支付查询交易来查询此订单是否交易成功（建议循环间隔等待时间大于等于5秒）。直到接收到的查询接口返回信息中的响应码为准确结果（“0000”-交易成功或“9990”-交易失败）后终止查询进行相应的处理，否则继续循环查询该订单。循环查询订单60秒后判断为该订单支付超时，需调用支付撤销交易讲改订单撤销，以免用户重复付款，结束本次交易。

### 查询交易流程描述

- 1.收银机发起支付查询交易，将查询信息（原交易流水等）传给收银系统。
- 2.收银系统向翼码平台请求查询交易。
- 3.翼码平台在接收到收银系统的请求后会进行报文参数校验、商户权限校验、IP安全校验等，校验通过后向支付宝、微信等支付平台发起查询请求。
- 4.支付宝、微信等支付平台将订单信息返回给翼码平台。
- 5.翼码平台接收支付平台返回的查询结果进行解析后，将处理过的查询结果信息返回给收银系统。
- 6.收银系统将查询结果信息返回给收银机。
- 7.收银机在接收到查询结果后，对信息中的响应码进行判断并做出相应处理：
  - (1) 当接收到的响应码为“0000”时表示该笔订单已支付成功，即可将商品给用户，完成本次交易。
  - (2) 当接收到的响应码为“9990”时表示该笔订单支付失败或已撤销，可根据响应码解释进行相应处理，本次交易付款失败不能把商品给用户。
  - (3) 当接收到的响应码为“9998”时表示该笔订单状态为等待用户付款，用户尚未付款，进行下一次循环查询或其他操作。

### 撤销交易流程描述

- 1.收银机发起支付撤销交易，将订单信息（原交易流水等）传给收银系统。
- 2.收银系统向翼码平台请求撤销交易。
- 3.翼码平台在接收到收银系统的请求后会进行报文参数校验、商户权限校验、IP安全校验等，校验通过后向支付宝、微信等支付平台发起撤销请求，并在平台上记录相应流水信息。
- 4.支付宝、微信等支付平台对原订单进行撤销处理，并将处理结果返回给翼码平台。
- 5.翼码平台接收支付平台返回的撤销结果进行解析后，将处理过的撤销结果信息返回给收银系统。
- 6.收银系统将撤销结果信息返回给收银机。
- 7.收银机在接收到撤销结果后，对信息中的响应码进行判断并做出相应处理：
  - (1) 当接收到的响应码为“0000”时表示该笔订单撤销成功，完成本次撤销交易。
  - (2) 当接收到的响应码为“9990”时表示该笔订单撤销失败，可再次发起请求进行撤销。
  - (3) 当接收到的响应码为“9998”时表示该笔订单撤销交易超时，可再次发起请求进行撤销。
  - (4) 当接收到的响应码为“4073”时表示该笔订单已经撤销成功（可能由于超时等原因第一次撤销结果未成功，但实际已经撤销了），完成本次撤销交易。

## 安全机制

通过固定IP绑定和https来保障报文传输的安全性。

## 接入URL

接口测试地址：<https://ipos-sa-test.wangcaio2o.com>

## 2.接入方式及报文格式规定

---

- [2.1.JSON格式](#)
- [2.2.XML格式](#)

## 2.1.JSON格式

- 1、接口的请求和响应的报文：采用JSON作为具体业务报文的载体
- 2、请求方式为：POST方式
- 3、报文中中文字符编码需要设置为：UTF-8
- 4、sign计算规则为：
  - (1) 筛选参数并排序

获取所有请求参数，不包含有子节点的参数，不包括字节类型参数，如文件、字节流，剔除sign字段，剔除值为空的参数（NULL或全空格都视为空），并按照第一个字符的键值ASCII码递增排序（字母升序排序），如果遇到相同字符则按照第二个字符的键值ASCII码递增排序，以此类推。

- (2) 拼接参数

将排序后的参数与其对应值，组合成“参数参数值”的格式，并且把这些参数全部拼接起来，此时生成的字符串为待签名字符串。

- (3) 计算签名

将待签名字符串首尾拼接密钥后，进行md5运算，并转为大写

### 5.报文及sign计算示例

- 如以下请求报文

```

1. {
2.   "request_type": "scanpay_request",
3.   "isspid": "39493002",
4.   "pos_id": "1511300001",
5.   "store_id": "1511300001",
6.   "pos_seq": "1812100000002",
7.   "user_id": "imageco",
8.   "system_id": "8888",
9.   "scanpay_request": {
10.    "pay_type": "502",
11.    "tx_amt": "1",
12.    "goods_detail": [
13.      {
14.        "goods_id": "apple-01啊",
15.        "payment_goods_id": "20010001",
16.        "goods_name": "ipad",
17.        "quantity": 1,
18.        "price": 200000,
19.        "goods_category": "34543238",
20.        "body": "特价手机",
21.        "show_url": "http://www.alipay.com/xxx.jpg"
22.      },
23.      {
24.        "goods_id": "apple-02额",
25.        "payment_goods_id": "20010002",
26.        "goods_name": "ipad2",
27.        "quantity": 1,
28.        "price": 200000,
29.        "goods_category": "34543239",
30.        "body": "特价手机2",
31.        "show_url": "http://www.alipay.com/xxx2.jpg"
32.      }
33.    ]
34.  }
35. }
```

- 接入密钥为 `test`

- (1) 筛选参数并排序

```

1. "isspid":"39493002"
2. "pos_id":"1511300001"
3. "pos_seq":"1812100000002"
```

```

4. "request_type":"scanpay_request"
5. "store_id":"1511300001"
6. "system_id":"8888"
7. "user_id":"imageco"
    
```

## (2) 拼接参数生成待签名字符串

```
1. isspid39493002pos_id1511300001pos_seq1812100000002request_typescanpay_requeststore_id1511300001system_id8888user_idimageco
```

## (3) 计算sign值

```

1. md5(testisspid39493002pos_id1511300001pos_seq1812100000002request_typescanpay_requeststore_id1511300001system_id8888user_idimagecotest)
2. =b7eee879c312adb820f44588d32c8222
    
```

## (4) 将计算出的sign值转为大写放入请求报文形成最终请求JSON

```

1. {
2.   "request_type": "scanpay_request",
3.   "isspid": "39493002",
4.   "pos_id": "1511300001",
5.   "store_id": "1511300001",
6.   "pos_seq": "1812100000002",
7.   "user_id": "imageco",
8.   "system_id": "8888",
9.   "scanpay_request": {
10.     "pay_type": "502",
11.     "tx_amt": "1",
12.     "goods_detail": [
13.       {
14.         "goods_id": "apple-01啊",
15.         "payment_goods_id": "20010001",
16.         "goods_name": "ipad",
17.         "quantity": 1,
18.         "price": 200000,
19.         "goods_category": "34543238",
20.         "body": "特价手机",
21.         "show_url": "http://www.alipay.com/xxx.jpg"
22.       },
23.       {
24.         "goods_id": "apple-02额",
25.         "payment_goods_id": "20010002",
26.         "goods_name": "ipad2",
27.         "quantity": 1,
28.         "price": 200000,
29.         "goods_category": "34543239",
30.         "body": "特价手机2",
31.         "show_url": "http://www.alipay.com/xxx2.jpg"
32.       }
33.     ]
34.   },
35.   "sign": "B7EEE879C312ADB820F44588D32C8222"
36. }
    
```

## 2.2.XML格式

1、接口的请求和响应的报文：采用XML作为具体业务报文的载体

2、请求方式为：POST方式

3、传输时的charset需要设置为：GBK

4、传递数据报文格式如下：

请求：xml=请求报文的urlencode&mac=mac计算值。

应答：响应报文

5、MAC计算规则为：**密钥+请求/应答报文+密钥** 进行MD5加密。其中的密钥由翼码平台提供给各接入平台。

示例：

请求报文为：<?xml version="1.0" encoding="GBK" ?><a>123</a>

密钥为：654321

则通讯报文为：**xml=urlencode(<?xml version="1.0" encoding="GBK" ?><a>123</a>)&mac=md5(654321<?xml version="1.0" encoding="GBK" ?><a>123</a>654321)**

黄色部分为固定字符串

红色部分为函数名称

绿色部分为请求报文

紫色部分为密钥

备注：请求报文中的中文需转换为GBK字符编码

xml参数报文格式如下：

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans version="1.0">
3.     <request_type>request_type</request_type>
4.     <pos_id>12345678</pos_id>
5.     <pos_seq>000000000001</pos_seq>
6.     <user_id>000001</user_id>
7.     <request_type_name>
8.         <field_tag1>field1</field_tag1>
9.         .....
10.    </request_type_name>
11. </business_trans>
    
```

说明：

XML元素	意义	说明
encoding	报文的字符集编码	GBK
business_trans	翼码平台交易标志，交易XML包的根标记，所有交易的内容必须位于此标注内	version为本接口规范版本号
request_type	交易类型，其值可能为： query_request barcode_pay_request barcode_query_request 等，红色部分请求为request，响应为response	
pos_seq	终端交易流水号	同一终端，终端流水号唯一
user_id	终端操作员号	

注：报文中灰色节点为可选节点，其它为必填节点。

由于返回报文可能增加元素，节点位置可能会发生变化，建议接收返回报文解析时按报文节点名称获取数据，而不要按照节点顺序获取。

## 3.iPos接口说明

当外部收银系统第一次接入翼码平台,并需先调用终端绑定交易,进行门店的创建,终端的开设。门店创建、终端开设完成后,后续才可以进行微信、支付宝的条码支付交易、条码支付的撤销交易、日结交易。

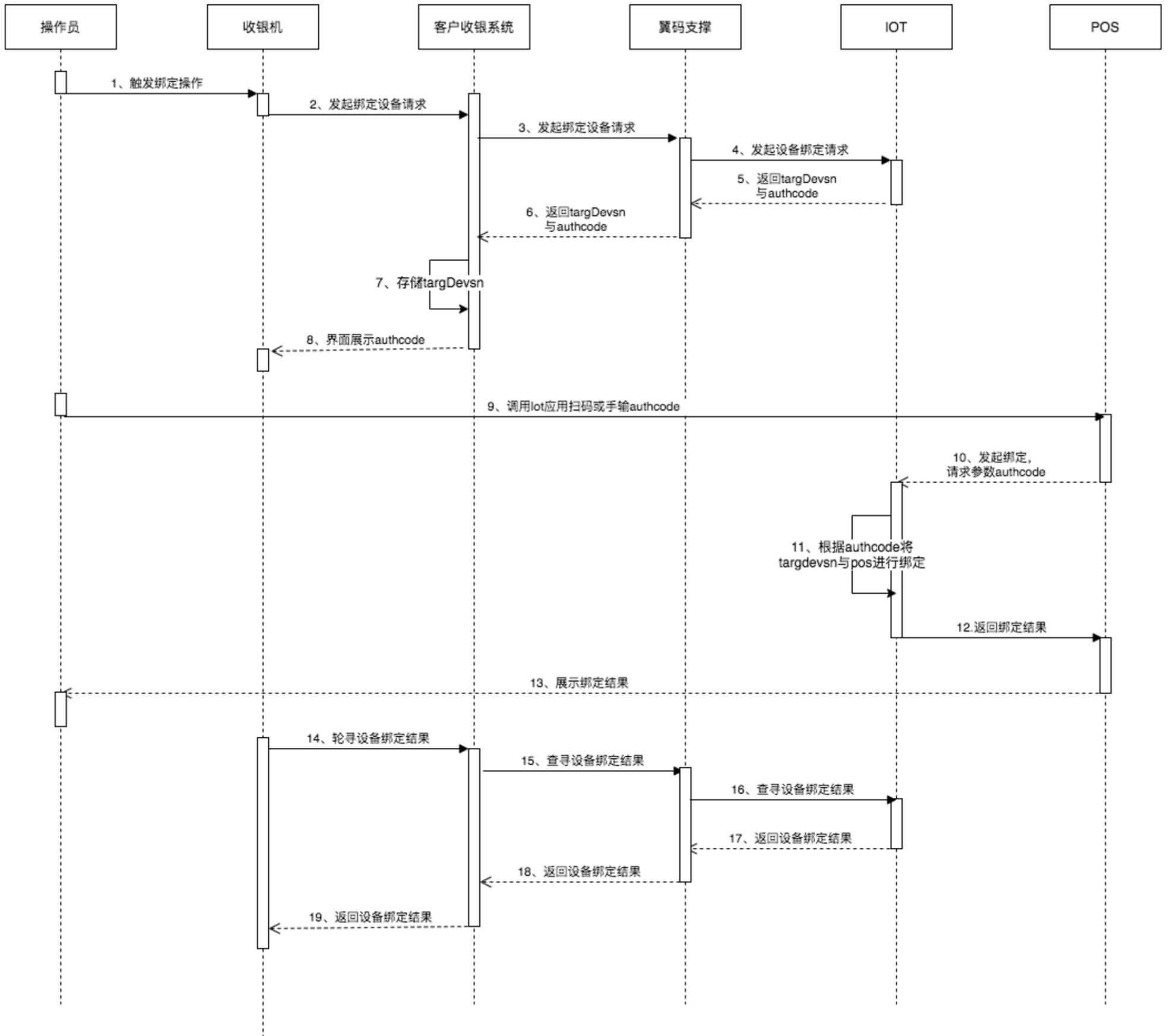
- 3.1.终端绑定交易
- 3.2.条码支付交易
- 3.3.扫码支付交易
- 3.4.支付查询交易
- 3.5.支付退款交易
- 3.6.退款查询交易
- 3.7.支付撤销交易
- 3.8.账单下载交易
- 3.9.支付日结交易
- 3.10.批量查询订单信息交易
- 3.11.查询订单信息交易
- 3.12.支付结果异步通知交易
- 3.13.JSAPI支付交易
- 3.14.统一下单交易
- 3.15.银行卡刷卡下单交易
- 3.16.银行卡刷卡结果查询交易
- 3.17.银行卡刷卡申请退款交易
- 3.18.银行卡刷卡退款查询交易
- 3.19.银行卡刷卡订单获取交易(翼码APP调用)
- 3.20.银行卡刷卡交易结果通知(翼码APP调用)
- 3.21.获取WebSocket配置交易(翼码APP调用)
- 3.22.现金退款交易
- 3.23.现金退款查询交易

### 3.3. 获取设备绑定状态与POS信息

#### 交易说明

使用授权码查询，设备是否绑定成功，并获取POS的设备SN号用于后续交易。

#### 时序图

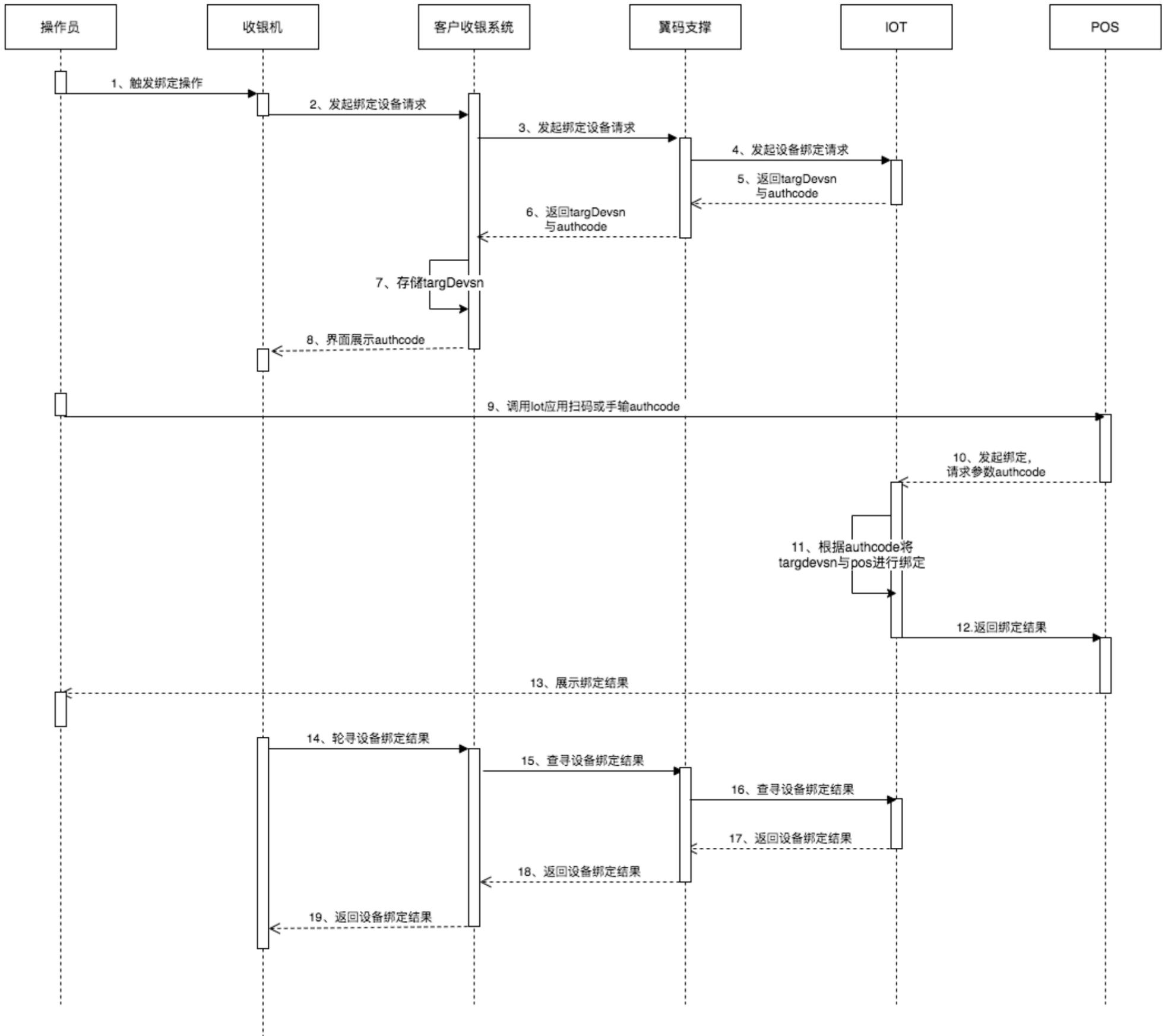


## 3.2.获取设备授权码交易

### 交易说明

获取设备绑定授权码，在设备上输入返回的授权码进行设备绑定。

### 时序图



## 3.1.终端绑定交易

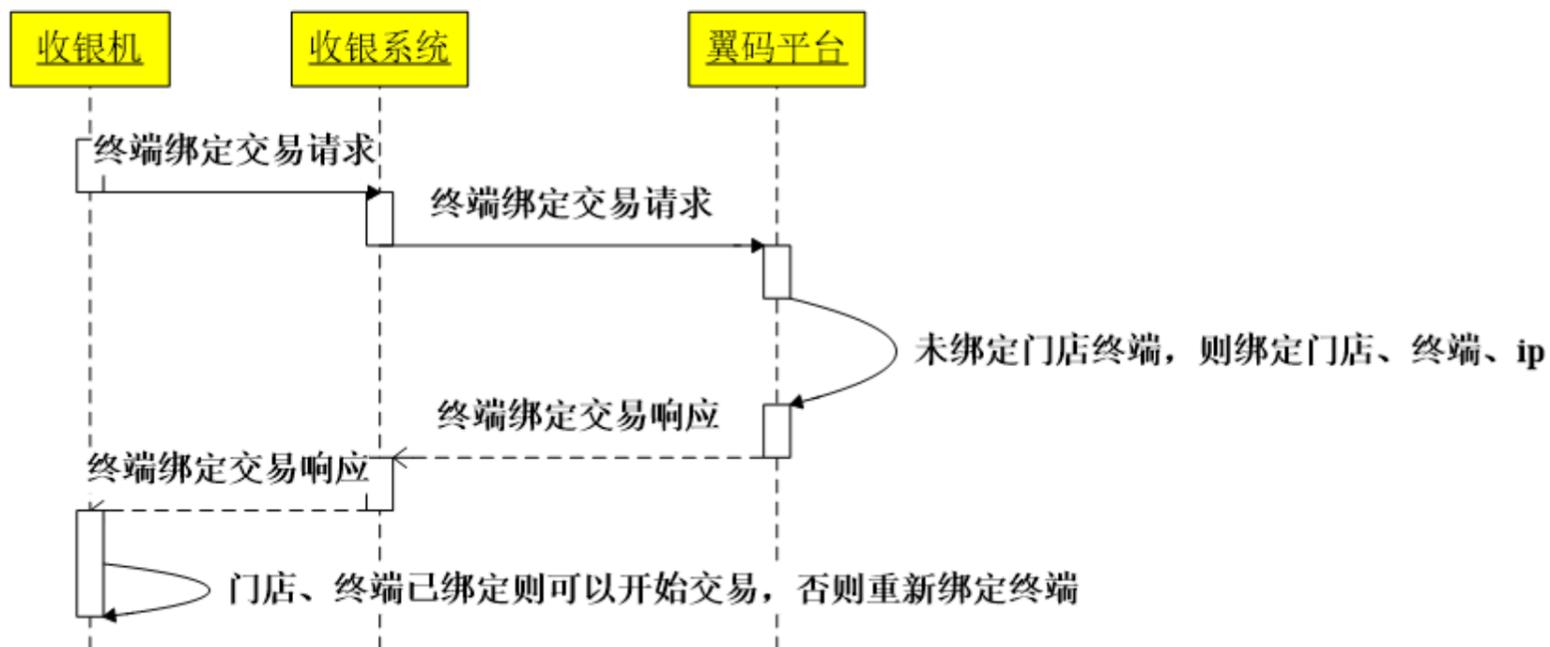
### 交易说明

外部收银系统要通过ipos方式接入翼码平台，首先要在翼码平台上创建收银机所在门店的信息，并在翼码平台上创建一台pos终端与收银机对应起来。

当外部收银系统调用翼码平台ipos的终端绑定接口时候，翼码平台收到请求报文后的处理如下：

- 1、翼码平台根据请求报文中的门店信息，查询翼码平台上是否存在该门店，存在则继续，不存在则创建门店。门店名称已存在时不会新增门店在原门店下新增终端，门店信息不会修改。
- 2、翼码平台根据请求报文中的终端信息，查询翼码支撑2.0平台上是否存在该终端，存在则继续，不存在则创建终端。

### 时序图



## 3.4. 条码支付交易

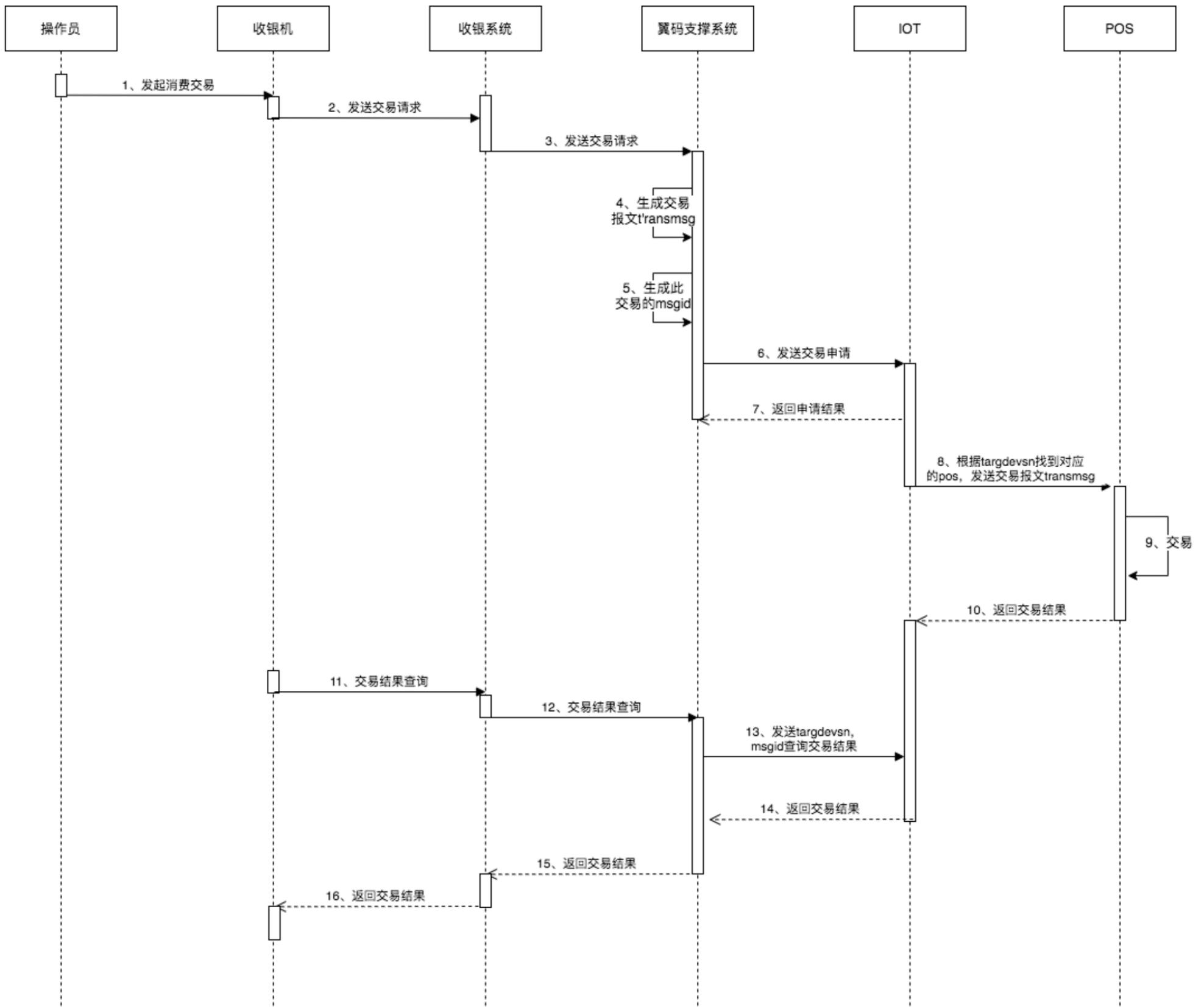
### 交易说明

支付交易的支付步骤说明如下：

- 1、收银系统先生成订单，选择支付方式，进入支付流程。
- 2、组支付交易报文，调用翼码条码支付请求接口，提交支付请求。
- 3、翼码检验请求报文的合法性，合法则组微信或支付宝的支付请求报文，调用云POS平台的消费接口，提交支付请求。
- 4、收银员在ME50、ME52、ME55等设备收到支付消息后，扫描消费者手机APP的付款码。
- 5、提交的支付交易的金额未超过免密限额，则直接扣款成功，否则用户手机上会弹出输入密码的提示界面。支付成功用户手机上会弹出交易结果。

步骤5会同步返回支付结果，如果需要密码输入，则返回等待付款状态（返回码9998）给收银系统，这时候商户收银系统需间隔6秒，然后进行查询交易，查询支付的结果，直到支付成功或超时（需查询12次，12次后还未支付成功的算超时）。当支付交易超时，支付交易还不明晰的情况下，认为该订单尚未支付成功。

### 时序图





## 3.5.支付查询交易

---

### 交易说明

查询交易，当支付交易结果不明晰（待付款、超时、系统错误等）的时候，收银系统将间隔几秒（支付宝支付间隔5s，微信支付间隔10s），发起查询交易，直到交易成功或超时。

## 3.6.支付退款交易

---

### 交易说明

当支付交易返回失败或交易超时时，需要调用退款接口，退款支付交易。支付交易为失败的交易，则会关闭订单，当支付交易为成功的，则撤销支付并退款。退款接口返回退款处理中时，需调用退款查询交易进行确认结果。

## 3.7.退款查询交易

---

### 交易说明

通过此接口查询退款订单是否成功

## 3.8.支付撤销交易

---

### 交易说明

支付的撤销接口，区别于退款接口的自动判断交易状态进行退款或撤销，本接口直接向支付平台发起撤销交易。注意：1.微信的扫码支付，若已经支付成功是无法撤销的，需调用退款接口进行退款。2.支付宝的扫码支付，生成订单码在还未被客户扫描之前支付宝侧不会生成订单，若此时调用撤销接口会返回撤销失败。

## 3.9. 银行卡刷卡下单交易

---

### 交易说明

通过此接口对银行卡刷卡进行下单记录信息

## 3.10.银行卡刷卡结果查询交易

---

### 交易说明

通过此接口查询银行卡刷卡支付结果

## 3.11.银行卡刷卡申请退款交易

---

### 交易说明

通过此接口进行银行卡刷卡的退款申请，退款结果以退款查询为准。

## 3.12. 银行卡刷卡退款查询交易

---

### 交易说明

通过此接口进行银行卡刷卡的退款结果。

## 4.iPos接口接入报文规范

- 4.1.终端绑定交易
- 4.2.条码支付交易
- 4.3.扫码支付交易
- 4.4.支付查询交易
- 4.5.支付退款交易
- 4.6.退款查询交易
- 4.7.支付撤销交易
- 4.8.账单下载交易
- 4.9.支付日结交易
- 4.10.批量查询订单信息交易
- 4.11.查询订单信息交易
- 4.12.支付结果异步通知交易
- 4.13.JSAPI支付交易
- 4.14.统一下单交易
- 4.15.银行卡刷卡下单交易
- 4.16.银行卡刷卡结果查询交易
- 4.17.银行卡申请退款交易
- 4.18.银行卡刷卡退款查询交易
- 4.19.银行卡刷卡订单获取交易(翼码APP调用)
- 4.20.银行卡刷卡交易结果通知(翼码APP调用)
- 4.21.获取WebSocket配置交易(翼码APP调用)
- 4.22.现金退款交易
- 4.23.现金退款查询交易

## 4.1.终端绑定交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "bind_pos_request",
3.   "isspid": "39493002",
4.   "pos_id": "1511300001",
5.   "pos_seq": "1511300000001",
6.   "user_id": "user01",
7.   "system_id": "8888",
8.   "store_info": {
9.     "store_id": "1511300001",
10.    "store_name": "门店名称",
11.    "store_addr": "门店地址",
12.    "contact": "门店联系人",
13.    "contact_tel": "13900000000",
14.    "contact_email": "11@qq.com",
15.    "province": "福建",
16.    "city": "福州",
17.    "town": "鼓楼"
18.  },
19.   "sign": "B973E02223C49798CDCBCEE48FC323EF"
20. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2.   <business_trans>
3.     <request_type>bind_pos_request</request_type>
4.     <isspid>39493002</isspid>
5.     <pos_id>1511300001</pos_id>
6.     <pos_seq>1511300000001</pos_seq>
7.     <user_id>user01</user_id>
8.     <system_id>8888</system_id>
9.     <store_info>
10.      <store_id>1511300001</store_id>
11.      <store_name>门店名称</store_name>
12.      <store_addr>门店地址</store_addr>
13.      <contact>门店联系人</contact>
14.      <contact_tel>13900000000</contact_tel>
15.      <contact_email>11@qq.com</contact_email>
16.      <province>福建</province>
17.      <city>福州</city>
18.      <town>鼓楼</town>
19.    </store_info>
20.  </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为bind_pos_request	必填
system_id	系统平台号	String(4)	由翼码提供	必填
isspid	商户号	String(8)	接入SP的ID号，为翼码平台的提供的业务机构号	必填
pos_id	终端号	String(20)	终端号（对方自定义的终端号）	必填
pos_seq	终端流水号	String(32)	该接口流水号可选填，便于日志定位	可选

节点名称	中文名称	类型	说明	是否必填
user_id	操作员标识	String(20)	终端操作员号, 对应于外部POS终端的操作员编号	可选
store_info->store_id	门店编号	String(20)	门店编号	必填
store_info->store_name	门店名称	String(100)	门店名称 (50个中文) 门店名称已存在时不会新增门店在原门店下新增终端, 门店信息不会修改	必填
store_info->store_addr	门店地址	String(100)	门店地址	可选
store_info->contact	门店联系人	String(32)	门店联系人	可选
store_info->contact_tel	门店联系电话	String(32)	门店联系电话	可选
store_info->contact_email	门店联系邮箱	String(64)	门店联系邮箱	可选
store_info->province	门店省份	String(32)	门店省份	可选
store_info->city	门店城市	String(32)	门店城市	可选
store_info->town	门店城镇	String(32)	门店城镇	可选

## 交易返回报文

JSON:

```
{
  "response_type": "bind_pos_response",
  "isspid": "39493002",
  "pos_id": "123456789",
  "pos_seq": "000000000001",
  "trans_time": "20181214113740",
  "result": {
    "id": "0000",
    "comment": "操作成功"
  },
  "sign": "EADFE430D0E856CF5F818690155F1258"
}
```

XML:

```
<?xml version="1.0" encoding="GBK"?>
<business_trans>
  <response_type>bind_pos_response</response_type>
  <isspid>39493002</isspid>
  <pos_id>123456789</pos_id>
  <pos_seq>000000000001</pos_seq>
  <sys_seq>000000000244</sys_seq>
  <trans_time>20090405010104</trans_time>
  <result>
    <id>0000</id>
    <comment>成功</comment>
  </result>
</business_trans>
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点, 请在解析返回报文时避免新增未知节点产生的程序出错!

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型, 固定为bind_pos_response	必填
isspid	商户号	String(8)	接入SP的ID号, 由翼码平台分配给ipos接入受理商户的机构号	必填
pos_id	终端号	String(20)	终端号 (对方自定义的终端号)	必填
pos_seq	终端流水号	String(32)	终端交易流水号和请求报文的交易流水相同	可选
trans_time	交易时间	String(14)	交易时间, 格式: yyyymmddhhnnss	必填
result->id	响应码	String(4)	翼码平台返回的处理结果,0000-成功, 其它失败, 错误信息说明见“响应码解释”字段	必填
result->comment	响应码解释	String(64)	响应码对应中文注解, 用于显示提示	必填

---

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

## 4.3. 获取设备绑定状态与POS信息

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "pos_info_request",
3.   "isspid": "39493002",
4.   "transaction_id": "20160823000001",
5.   "system_id": "8888",
6.   "auth_code": "111111",
7.   "sign": "B973E02223C49798CDCBCEE48FC323EF"
8. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>pos_info_request</request_type>
4.   <isspid>39493002</isspid>
5.   <transaction_id>20160823000001</transaction_id>
6.   <system_id>8888</system_id>
7.   <auth_code>111111</auth_code>
8. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为pos_info_request	必填
system_id	系统平台号	String(4)	由翼码提供	必填
isspid	商户号	String(8)	翼码提供的商户号	必填
transaction_id	请求流水号	String(32)	方便查询日志定位	必填
auth_code	授权码	String	需要查询绑定结果及信息的授权码	必填

### 交易返回报文

JSON:

```

1. {
2.   "response_type": "pos_info_response",
3.   "isspid": "39493002",
4.   "transaction_id": "20160823000001",
5.   "auth_code": "111111",
6.   "dev_sn": "222222",
7.   "pos_name": "被绑定的设备的备注名称",
8.   "result": {
9.     "id": "0000",
10.    "comment": "设备绑定成功"
11.  },
12.   "sign": "EADFE430D0E856CF5F818690155F1258"
13. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
```

```

2. <business_trans>
3.   <response_type>pos_info_response</response_type>
4.   <isspid>39493002</isspid>
5.   <transaction_id>20160823000001</transaction_id>
6.   <auth_code>111111</auth_code>
7.   <dev_sn>222222</dev_sn>
8.   <pos_name>被绑定的设备的备注名称</pos_name>
9.   <result>
10.     <id>0000</id>
11.     <comment>设备绑定成功</comment>
12.   </result>
13. </business_trans>
    
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型，固定为pos_info_response	必填
isspid	商户号	String(8)	翼码商户号	必填
transaction_id	请求流水号	String(32)	方便查询日志定位	必填
auth_code	授权码	String	需要查询绑定结果及信息的授权码	必填
dev_sn	设备SN号	String	绑定成功时必填 设备SN号,用于后续推送消息至该设备的唯一识别ID 该SN与绑定的POS机背后的的SN号一致	可空
pos_name	被绑定的设备的备注名称	String	绑定成功时必填	可空
result ->id	响应码	String(4)	0000-设备绑定成功 9998-设备绑定中，请稍后查询 其它表示失败，错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解，用于显示提示	必填

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

## 4.2. 获取设备授权码交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "pos_authcode_request",
3.   "isspid": "39493002",
4.   "transaction_id": "20160823000001",
5.   "system_id": "8888",
6.   "pos_name": "被绑定的设备的备注名称",
7.   "sign": "B973E02223C49798CDCBCEE48FC323EF"
8. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>pos_authcode_request</request_type>
4.   <isspid>39493002</isspid>
5.   <transaction_id>20160823000001</transaction_id>
6.   <system_id>8888</system_id>
7.   <pos_name>被绑定的设备的备注名称</pos_name>
8. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为pos_authcode_request	必填
system_id	系统平台号	String(4)	由翼码提供	必填
isspid	商户号	String(8)	翼码提供的商户号	必填
transaction_id	请求流水号	String(32)	方便查询日志定位	必填
pos_name	被绑定的设备的备注名称	String	用于绑定多个设备时人工区分(可用贴纸贴在设备上快速区分)	必填

### 交易返回报文

JSON:

```

1. {
2.   "response_type": "pos_authcode_response",
3.   "isspid": "39493002",
4.   "transaction_id": "20160823000001",
5.   "auth_code": "111111",
6.   "result": {
7.     "id": "0000",
8.     "comment": "成功"
9.   },
10.   "sign": "EADFE430D0E856CF5F818690155F1258"
11. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>pos_authcode_response</response_type>
```

```

4.   <isspid>39493002</isspid>
5.   <transaction_id>20160823000001</transaction_id>
6.   <auth_code>111111</auth_code>
7.   <result>
8.       <id>0000</id>
9.       <comment>成功</comment>
10.  </result>
11. </business_trans>
    
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型，固定为pos_authcode_response	必填
isspid	商户号	String(8)	翼码商户号	必填
transaction_id	请求流水号	String(32)	方便查询日志定位	必填
auth_code	授权码	String	交易成功时必填 生成二维码后在 POS 机上的 IOT 服务应用中扫码输入 或者在 POS 机上的 IOT 服务应用中手工输入	可空
result ->id	响应码	String(4)	0000-成功 其它表示失败，错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解，用于显示提示	必填

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

## 4.4. 条码支付交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "barcode_pay_request",
3.   "isspid": "39493002",
4.   "pos_id": "1511300001",
5.   "store_id": "1511300001",
6.   "pos_seq": "1511300000002",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "barcode_pay_request": {
10.     "barcode_info": "1234567890123456",
11.     "pay_type": "502",
12.     "tx_amt": "1",
13.     "pos_sn": "xxxxxxxxxxxxx"
14.   },
15.   "sign": "EB0F3B9404AAFC27C520C1D64A69C169"
16. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>barcode_pay_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>1511300000002</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <barcode_pay_request>
11.    <barcode_info>1234567890123456</barcode_info>
12.    <pay_type>502</pay_type>
13.    <tx_amt>10</tx_amt>
14.    <pos_sn>xxxxxxxxxxxx</pos_sn>
15.  </barcode_pay_request>
16. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为barcode_pay_request	必填
isspid	商户号	String(8)	接入SP的ID号，由翼码平台分配给ipos接入受理商户的机构号（原值返回）	必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	同一个商户必须唯一，且大于12位	必填
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填
barcode_pay_request ->barcode_info	条码支付动态码	String(64)	条码支付动态码	必填

节点名称	中文名称	类型	说明	是否必填
barcode_pay_request ->pay_type	支付类型	String(3)	502: 支付宝支付 503: 微信支付 512: 银联二维码	可空
barcode_pay_request ->tx_amt	交易金额	Number(9)	条码支付金额, 单位为分, 不支持小数点	必填
barcode_pay_request ->pos_sn	设备sn号	String	设备sn号, 设备上可查看	

## 交易返回报文

JSON:

```

1. {
2.   "response_type": "barcode_pay_response",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "pos_seq": "2018121400000001",
6.   "trans_time": "20181214135756",
7.   "pay_type": "502",
8.   "extend": "<![CDATA[]]>",
9.   "sys_seq": "5d170b6b78a0cb13",
10.  "result": {
11.    "id": "0000",
12.    "comment": "交易成功"
13.  },
14.  "pay_response": {
15.    "order_status": "0",
16.    "channel_id": "alipay",
17.    "trade_no": "xxxxxxxxxxxx",
18.    "total_fee": "10",
19.    "receipt_amount": "10",
20.    "trans_time": "20201130111111"
21.  },
22.  "sign": "C557C5D68AA5D803B2E6FF54AA28E5DF"
23. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>barcode_pay_response</response_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>123456789</pos_id>
6.   <pos_seq>000000000001</pos_seq>
7.   <sys_seq>00000000244</sys_seq>
8.   <trans_time>20090405010104</trans_time>
9.   <pay_type>502</pay_type>
10.  <extend><![CDATA[扩展字段(json格式)]]></extend>
11.  <pay_response>
12.    <order_status>0</order_status>
13.    <channel_id>alipay</channel_id>
14.    <trade_no>xxxxxxxxxxxx</trade_no>
15.    <total_fee>10</total_fee>
16.    <receipt_amount>10</receipt_amount>
17.    <trans_time>20201130111111</trans_time>
18.  </pay_response>
19.  <result>
20.    <id>0000</id>
21.    <comment>成功</comment>
22.  </result>
23. </business_trans>
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型，固定为barcode_pay_response	必填
isspid	商户号	String(8)	接入SP的ID号，为翼码平台的提供的业务机构号	必填
pos_id	终端号	String(20)	终端号	必填
pos_seq	终端流水号	String(32)	终端交易流水号，和请求报文中的交易流水号相同	必填
sys_seq	随机串号	String(16)	随机生成串号，仅用于日志跟踪用，不保存	必填
trans_time	交易时间	String(14)	交易时间，格式：yyyymmddhhnss	必填
pay_type	支付类型	String(3)	502：支付宝支付 503：微信支付 512：银联二维码	必填
extend	扩展信息	String	扩展信息字段内容为json格式 预留字段后续业务扩展使用	可空
result ->id	响应码	String(4)	翼码平台返回的处理结果 0000-成功 9998-等待付款，需要调用支付查询交易确认结果 其它失败，错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解，用于显示提示	必填
交易返回数据			仅result->id返回为0000时返回一下数据	可空
pay_response ->order_status	订单状态	Number(1)	0.正常 1.已退货 2.已撤销 3.未知/处理中	必填
pay_response ->channel_id	实际支付渠道	String(10)	支付宝支付：alipay, 微信支付：wxpay, 收单支付：acquire/boc ..., 会员卡：card	必填
pay_response ->trade_no	原系统交易流水	String	支付宝、微信等渠道返回的交易流水号	必填
pay_response ->total_fee	应付金额	Number(9)	订单应付金额,单位:分,不支持小数点	必填
pay_response ->receipt_amount	实付金额	Number(9)	用户实付金额,单位:分,不支持小数点	必填
pay_response ->buyer_logon_id	买家支付宝用户号	String	买家支付宝用户号,仅在channel_id 为alipay时生效	可空
pay_response ->trans_time	系统交易时间	String(14)	支付宝、微信等系统交易时间，格式：yyyymmddhhnss	必填

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

## 4.5.支付查询交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "barcode_query_request",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "store_id": "8888888888",
6.   "pos_seq": "2018121400000005",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "barcode_query_request": {
10.    "pay_type": "502",
11.    "org_pos_seq": "2018121400000001"
12.  },
13.   "sign": "79CB18BC49A4D60C457506EA4EF27C14"
14. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>barcode_query_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>151130000003</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <barcode_query_request>
11.    <pay_type>502</pay_type>
12.    <org_pos_seq>151130000001</org_pos_seq>
13.  </barcode_query_request>
14. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为barcode_query_request	必填
isspid	商户号	String(8)	接入SP的ID号，由翼码平台分配给ipos接入受理商户的机构号 (原值返回)	必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	若传入要求同一个商户下不重复，且大于12位 可以选填	可选
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填
barcode_query_request ->pay_type	支付类型	String(3)	502: 支付宝支付 503: 微信支付 512: 银联二维码 515: 小程序支付	可空
barcode_query_request ->org_pos_seq	需要查询的支付交易流水号	String(32)	需要查询的支付交易流水号，且大于12位	必填

节点名称	中文名称	类型	说明	是否必填
barcode_query_request ->pos_sn	设备sn号	String	设备sn号, 设备上可查看	必填

## 交易返回报文

JSON:

```

1. {
2.   "response_type": "barcode_query_response",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "pos_seq": "2018121400000005",
6.   "trans_time": "20201130111111",
7.   "pay_type": "502",
8.   "extend": "<![CDATA[]]>",
9.   "sys_seq": "0f041092d0ccc6ae",
10.  "result": {
11.    "id": "0000",
12.    "comment": "交易成功"
13.  },
14.  "pay_response": {
15.    "order_status": "0",
16.    "channel_id": "alipay",
17.    "trade_no": "xxxxxxxxxxxx",
18.    "total_fee": "10",
19.    "receipt_amount": "10",
20.    "trans_time": "20201130111111"
21.  },
22.  "sign": "79D928004531AE15DFA6AD7E9A6A2447"
23. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>barcode_query_response</response_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>123456789</pos_id>
6.   <pos_seq>000000000001</pos_seq>
7.   <sys_seq>00000000244</sys_seq>
8.   <trans_time>20201130111111</trans_time>
9.   <pay_type>502</pay_type>
10.  <extend><![CDATA[扩展字段(json格式)]]></extend>
11.  <result>
12.    <id>0000</id>
13.    <comment>成功</comment>
14.  </result>
15.  <pay_response>
16.    <order_status>0</order_status>
17.    <channel_id>alipay</channel_id>
18.    <trade_no>xxxxxxxxxxxx</trade_no>
19.    <total_fee>10</total_fee>
20.    <receipt_amount>10</receipt_amount>
21.    <trans_time>20201130111111</trans_time>
22.  </pay_response>
23. </business_trans>
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点, 请在解析返回报文时避免新增未知节点产生的程序出错!

节点名称	中文名称	类型	说明	是否必填
------	------	----	----	------

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型, 固定为barcode_query_response	必填
isspid	商户号	String(8)	接入SP的ID号, 为翼码平台提供的业务机构号	必填
pos_id	终端号	String(20)	终端号	必填
pos_seq	终端流水号	String(32)	终端交易流水号, 和请求报文中的交易流水号相同	可选
sys_seq	随机串号	String(16)	随机生成串号, 仅用于日志跟踪用, 不保存	必填
trans_time	交易时间	String(14)	交易时间, 格式: yyyymmddhhnss	必填
pay_type	支付类型	String(3)	502: 支付宝支付 503: 微信支付 512: 银联二维码 515: 小程序支付	必填
extend	扩展信息	String	扩展信息字段内容为json格式 预留字段后续业务扩展使用	可空
result ->id	响应码	String(4)	翼码平台返回的处理结果 0000-成功 9998-等待付款, 需要调用支付查询交易确认结果 其它失败, 错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解, 用于显示提示	必填
交易返回数据			仅result->id返回为0000时返回一下数据	可空
pay_response ->order_status	订单状态	Number(1)	0.正常 1.已退货 2.已撤销 3.未知/处理中	必填
pay_response ->channel_id	实际支付渠道	String(10)	支付宝支付: alipay, 微信支付: wxpay, 收单支付: acquire/boc ..., 会员卡: card	必填
pay_response ->trade_no	原系统交易流水	String	支付宝、微信等渠道返回的交易流水号	必填
pay_response ->total_fee	应付金额	Number(9)	订单应付金额,单位:分,不支持小数点	必填
pay_response ->receipt_amount	实付金额	Number(9)	用户实付金额,单位:分,不支持小数点	必填
pay_response ->buyer_logon_id	买家支付宝用户号	String	买家支付宝用户号,仅在channel_id 为alipay时生效	可空
pay_response ->trans_time	系统交易时间	String(14)	支付宝、微信等系统交易时间, 格式: yyyymmddhhnss	必填

标准接口会因业务升级需要而新增节点, 请在解析返回报文时避免新增未知节点产生的程序出错!

## 4.6.支付退款交易

接口会根据支付订单情况自动判断退款或撤销。退款交易适用于已成功的支付交易，有效期一般有几个月，不同的支付平台有所差异；撤销适用于所有状态的支付交易，但是有效期只有24小时。

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "barcode_reverse_request",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "store_id": "8888888888",
6.   "pos_seq": "2018121400000006",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "barcode_reverse_request": {
10.    "tx_amt": "1",
11.    "org_pos_seq": "2018121400000001",
12.    "pos_sn": "xxxxxxxxxxxx",
13.    "org_trade_no": "xxxxxxxxxxxx"
14.  },
15.   "sign": "E622B688C360F8444788740B111E4787"
16. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>barcode_reverse_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>151130000004</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <barcode_reverse_request>
11.    <tx_amt>10</tx_amt>
12.    <org_pos_seq>151130000001</org_pos_seq>
13.    <pos_sn>xxxxxxxxxxxx</pos_sn>
14.    <org_trade_no>xxxxxxxxxx</org_trade_no>
15.  </barcode_reverse_request>
16. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为barcode_reverse_request	必填
isspid	商户号	String(8)	接入SP的ID号，由翼码平台分配给ipos接入受理商户的机构号（原值返回）	必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	同一个商户必须唯一，且大于12位	必填
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填

节点名称	中文名称	类型	说明	是否必填
barcode_reverse_request ->tx_amt	交易金额	Number(9)	条码支付金额, 单位为分,不支持小数点	必填
barcode_reverse_request ->org_pos_seq	需要退款交易的流水号	String(32)	需要退款的原支付流水号, 且大于12位	必填
barcode_reverse_request ->pos_sn	设备sn号	String	设备sn号, 设备上可查看	必填
barcode_reverse_request ->org_trade_no	原系统交易流水	String	交易请求/查询交易返回的trade_no	必填

## 交易返回报文

JSON:

```

1. {
2.   "response_type": "barcode_reverse_response",
3.   "system_id": "8888",
4.   "isspid": "39493002",
5.   "pos_id": "8888888888",
6.   "pos_seq": "2018121400000006",
7.   "sys_seq": "b7ff9fa5bb435e2",
8.   "trans_time": "20201130111111",
9.   "pay_type": "502",
10.  "extend": "<![CDATA[]]>",
11.  "result": {
12.    "id": "0000",
13.    "comment": "交易成功"
14.  },
15.  "pay_response": {
16.    "trade_no": "xxxxxxxxxxxxxx",
17.    "total_fee": "10",
18.    "receipt_amount": "10",
19.    "trans_time": "20201130111111"
20.  },
21.  "sign": "207482F2C0A998B73F388B8AA401DB76"
22. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>barcode_reverse_response</response_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>123456789</pos_id>
6.   <pos_seq>000000000001</pos_seq>
7.   <sys_seq>000000000244</sys_seq>
8.   <trans_time>20201130111111</trans_time>
9.   <pay_type>502</pay_type>
10.  <extend><![CDATA[扩展字段(json格式)]]></extend>
11.  <result>
12.    <id>0000</id>
13.    <comment>成功</comment>
14.  </result>
15.  <pay_response>
16.    <trade_no>xxxxxxxxxx</trade_no>
17.    <total_fee>10</total_fee>
18.    <receipt_amount>10</receipt_amount>
19.    <trans_time>20201130111111</trans_time>
20.  </pay_response>
21. </business_trans>
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型，固定为barcode_reverse_response	必填
isspid	商户号	String(8)	接入SP的ID号，为翼码平台提供的业务商机构号	必填
pos_id	终端号	String(20)	终端号	必填
pos_seq	终端流水号	String(32)	终端交易流水号，和请求报文中的交易流水号相同	必填
sys_seq	随机串号	String(16)	随机生成串号，仅用于日志跟踪用，不保存	必填
trans_time	交易时间	String(14)	交易时间，格式：yyyymmddhhnss	必填
pay_type	支付类型	String(3)	502：支付宝支付 503：微信支付 512：银联二维码 515：小程序支付	必填
extend	扩展信息	String	扩展信息字段内容为json格式 预留字段后续业务扩展使用	可空
result ->id	响应码	String(4)	翼码平台返回的处理结果 0000-成功 9998-退款处理中，需要调用退款查询交易确认退款结果 其它为失败，错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解，用于显示提示	必填
交易返回数据			仅result->id返回为0000时返回一下数据	可空
pay_response ->trade_no	原系统交易流水	String	支付宝、微信等渠道返回的交易流水号	必填
pay_response ->total_fee	应付金额	Number(9)	订单应付金额,单位:分,不支持小数点	必填
pay_response ->receipt_amount	实付金额	Number(9)	用户实付金额,单位:分,不支持小数点	必填
pay_response ->trans_time	系统交易时间	String(14)	支付宝、微信等系统交易时间，格式：yyyymmddhhnss	必填

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

## 4.7.退款查询交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "barcode_refundquery_request",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "store_id": "8888888888",
6.   "pos_seq": "2018121400000008",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "barcode_refundquery_request": {
10.     "pay_type": "502",
11.     "org_pos_seq": "2018121400000006" ,
12.     "pos_sn": "xxxxxxxxxxxx"
13.   },
14.   "sign": "A95561E9CBB30BD049000FD9C530B1AF"
15. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>barcode_refundquery_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>151130000003</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <barcode_refundquery_request>
11.    <pay_type>502</pay_type>
12.    <org_pos_seq>151130000001</org_pos_seq>
13.    <pos_sn>xxxxxxxxxx</pos_sn>
14.  </barcode_refundquery_request>
15. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为barcode_refundquery_request	必填
isspid	商户号	String(8)	接入SP的ID号，由翼码平台分配给ipos接入受理商户的机构号 (原值返回)	必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	若传入要求同一个商户下不重复，且大于12位 可以选填	可选
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填
barcode_refundquery_request ->pay_type	支付类型	String(3)	502: 支付宝支付 503: 微信支付 512: 银联二维码 515: 小程序支付	可空

节点名称	中文名称	类型	说明	是否必填
barcode_refundquery_request ->org_pos_seq	需要查询的退款交易流水号	String(32)	需要查询的退款交易流水号，且大于12位	必填
barcode_refundquery_request ->pos_sn	设备sn号	String	设备sn号，设备上可查看	必填

## 交易返回报文

JSON:

```

1. {
2.   "response_type": "barcode_refundquery_response",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "pos_seq": "2018121400000008",
6.   "sys_seq": "a4507a682d26403a",
7.   "trans_time": "20201130111111",
8.   "pay_type": "502",
9.   "extend": "<![CDATA[]]>",
10.  "result": {
11.    "id": "0000",
12.    "comment": "交易成功"
13.  },
14.  "pay_response": {
15.    "trans_amount": "10",
16.    "amount": "10",
17.    "trade_no": "xxxxxxxxxx",
18.    "trans_time": "20201130111111"
19.  },
20.  "sign": "B94B9113A279C761A831C5B592E0BF09"
21. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>barcode_refundquery_response</response_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>123456789</pos_id>
6.   <pos_seq>00000000001</pos_seq>
7.   <sys_seq>00000000244</sys_seq>
8.   <trans_time>20201130111111</trans_time>
9.   <pay_type>502</pay_type>
10.  <extend><![CDATA[扩展字段(json格式)]]></extend>
11.  <result>
12.    <id>0000</id>
13.    <comment>成功</comment>
14.  </result>
15.  <pay_response>
16.    <trans_amount>10</trans_amount>
17.    <amount>10</amount>
18.    <trade_no>xxxxxxxx</trade_no>
19.    <total_fee>10</total_fee>
20.    <trans_time>20201130111111</trans_time>
21.  </pay_response>
22. </business_trans>
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选

节点名称	中文名称	类型	说明	是否必填
response_type	返回类型	String(32)	标识返回报文的类型, 固定为barcode_refundquery_response	必填
isspid	商户号	String(8)	接入SP的ID号, 为翼码平台提供的业务商机构号	必填
pos_id	终端号	String(20)	终端号	必填
pos_seq	终端流水号	String(32)	终端交易流水号, 和请求报文中的交易流水号相同	可选
sys_seq	随机串号	String(16)	随机生成串号, 仅用于日志跟踪用, 不保存	必填
trans_time	交易时间	String(14)	交易时间, 格式: yyyymmddhhnss	必填
pay_type	支付类型	String(3)	502: 支付宝支付 503: 微信支付 512: 银联二维码 515: 小程序支付	必填
extend	扩展信息	String	扩展信息字段内容为json格式 预留字段后续业务扩展使用	可空
result ->id	响应码	String(4)	翼码平台返回的处理结果 0000-成功 9998-退款处理中, 需要调用退款查询交易确认退款结果 其它为失败, 错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解, 用于显示提示	必填
交易返回数据 pay_response			仅result->id返回为0000时返回以下数据	可空
pay_response ->trans_amount	实际退款金额	Number(9)	单位:分,不支持小数	必填
pay_response ->amount	请求退款金额	Number(9)	单位:分,不支持小数	必填
pay_response ->trade_no	系统交易流水	String	支付宝、微信、银联等返回的交易流水号	必填
pay_response ->total_fee	交易订单金额	Number(9)	支付时的订单金额	必填
pay_response ->trans_time	系统交易时间	String(14)	支付宝、微信等系统交易时间, 格式: yyyymmddhhnss	必填

标准接口会因业务升级需要而新增节点, 请在解析返回报文时避免新增未知节点产生的程序出错!

## 4.8.支付撤销交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "barcode_cancel_request",
3.   "isspid": "39493002",
4.   "pos_id": "6666666666",
5.   "store_id": "6666666666",
6.   "pos_seq": "2018121400000012",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "barcode_cancel_request": {
10.     "pay_type": "",
11.     "org_pos_seq": "2018121400000010" ,
12.     "org_trade_no": "xxxxxxxxxxxx",
13.     "pos_sn": "xxxxxxxx"
14.   },
15.   "sign": "38C73CFED3B3C6E0ED18AF04E8D34D94"
16. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>barcode_cancel_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>1511300000004</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <barcode_cancel_request>
11.    <pay_type>502</pay_type>
12.    <org_pos_seq>1511300000001</org_pos_seq>
13.    <org_trade_no>xxxxxxxxxx</org_trade_no>
14.    <pos_sn>xxxxxxxxxx</pos_sn>
15.  </barcode_cancel_request>
16. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为barcode_cancel_request	必填
isspid	商户号	String(8)	接入SP的ID号，由翼码平台分配给ipos接入受理商户的机构号（原值返回）	必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	同一个商户必须唯一，且大于12位	必填
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填
barcode_cancel_request ->pay_type	支付类型	String(3)	502: 支付宝支付 503: 微信支付 512: 银联二维码 515: 小程序支付	可空

节点名称	中文名称	类型	说明	是否必填
barcode_cancel_request ->org_pos_seq	需要撤销交易的流水号	String(32)	需要撤销的原支付流水号, 且大于12位	必填
barcode_cancel_request ->pos_sn	设备sn号	String	设备sn号, 设备上可查看	必填
barcode_cancel_request ->org_trade_no	原系统交易流水	String	交易请求/查询交易返回的trade_no	必填

## 交易返回报文

JSON:

```

1. {
2.   "response_type": "barcode_cancel_response",
3.   "isspid": "00026713",
4.   "pos_id": "6666666666",
5.   "pos_seq": "2018121400000012",
6.   "sys_seq": "f135e20440ea261b",
7.   "trans_time": "20201130111111",
8.   "pay_type": "502",
9.   "extend": "<![CDATA[]]>",
10.  "result": {
11.    "id": "0000",
12.    "comment": "交易成功"
13.  },
14.  "pay_response": {
15.    "trade_no": "xxxxxxxxxxx",
16.    "receipt_amount": "10",
17.    "trans_time": "20201130111111"
18.  },
19.  "sign": "F3417D5E9CEF3AE32110C7EFBF5AB400"
20. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>barcode_cancel_response</response_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>123456789</pos_id>
6.   <pos_seq>000000000001</pos_seq>
7.   <sys_seq>00000000244</sys_seq>
8.   <trans_time>20201130111111</trans_time>
9.   <pay_type>502</pay_type>
10.  <extend><![CDATA[扩展字段(json格式)]></extend>
11.  <result>
12.    <id>0000</id>
13.    <comment>成功</comment>
14.  </result>
15.  <pay_response>
16.    <trade_no>xxxxxxxxxxxx</trade_no>
17.    <receipt_amount>10</receipt_amount>
18.    <trans_time>20090405010104</trans_time>
19.  </pay_response>
20. </business_trans>
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点, 请在解析返回报文时避免新增未知节点产生的程序出错!

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选

节点名称	中文名称	类型	说明	是否必填
response_type	返回类型	String(32)	标识返回报文的类型, 固定为barcode_cancel_response	必填
isspid	商户号	String(8)	接入SP的ID号, 为翼码平台提供的业务机构号	必填
pos_id	终端号	String(20)	终端号	必填
pos_seq	终端流水号	String(32)	终端交易流水号, 和请求报文中的交易流水号相同	必填
sys_seq	随机串号	String(16)	随机生成串号, 仅用于日志跟踪用, 不保存	必填
trans_time	交易时间	String(14)	交易时间, 格式: yyyymmddhhnss	必填
pay_type	支付类型	String(3)	502: 支付宝支付 503: 微信支付 512: 银联二维码 515: 小程序支付	必填
extend	扩展信息	String	扩展信息字段内容为json格式 预留字段后续业务扩展使用	可空
result ->id	响应码	String(4)	翼码平台返回的处理结果,0000-成功, 其它失败, 错误信息说明见“响应码解释” 字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解, 用于显示提示	必填
交易返回数据 pay_response			仅result->id返回为0000时返回以下数据	可空
pay_response ->trade_no	系统交易流水号	String	支付宝、微信等平台返回的交易流水号	必填
pay_response ->receipt_amount	实际撤销金额	Number(9)	单位:分,不支持小数	必填
pay_response ->trans_time	系统交易时间	String(14)	支付宝、微信等系统交易时间, 格式: yyyymmddhhnss	必填

标准接口会因业务升级需要而新增节点, 请在解析返回报文时避免新增未知节点产生的程序出错!

## 4.9. 银行卡刷卡下单交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "bcm_ordercreate_request",
3.   "isspid": "39493002",
4.   "pos_id": "1511300001",
5.   "store_id": "1511300001",
6.   "pos_seq": "2018121400000004",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "bcm_ordercreate_request": {
10.     "pos_sn": "N9NL0000001",
11.     "tx_amt": "1",
12.     "extend_params": null
13.   },
14.   "sign": "314DA43E7E244105ECBD45F797264854"
15. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>bcm_ordercreate_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>1511300000002</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <bcm_ordercreate_request>
11.    <pos_sn>N9NL0000001</pos_sn>
12.    <tx_amt>1</tx_amt>
13.    <extend_params></extend_params>
14.    <goods_detail><![CDATA[
15.      [
16.        {
17.          "goods_name": "啊ipad",
18.          "quantity": 1,
19.          "price": 200000
20.        },
21.        {
22.          "goods_name": "额ipad2",
23.          "quantity": 1,
24.          "price": 200000
25.        }
26.      ]
27.    ]]></goods_detail>
28.  </bcm_ordercreate_request>
29. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为bcm_ordercreate_request	必填
isspid	商户号	String(8)	接入SP的ID号，由翼码平台分配给ipos接入受理商户的机构号（原值返回）	必填

节点名称	中文名称	类型	说明	是否必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	同一个商户必须唯一，且大于12位	必填
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填
bcm_ordercreate_request ->pos_sn	POS机身号	String(20)	需要推送到的POS机身号 不传则不推送，使用二维码方式传递订单信息	必填
bcm_ordercreate_request ->tx_amt	订单金额	Number(9)	订单金额，单位为分，不支持小数点	必填
bcm_ordercreate_request ->extend_params	扩展参数	String	用于扩展字段，暂不启用	可空

## 交易返回报文

JSON:

```

1. {
2.   "response_type": "bcm_ordercreate_response",
3.   "isspid": "39493002",
4.   "pos_id": "6666666666",
5.   "pos_seq": "2018121400000004",
6.   "trans_time": "20181214143516",
7.   "sys_seq": "0741136885a748c1",
8.   "result": {
9.     "id": "9998",
10.    "comment": "创建订单成功"
11.  },
12.   "sign": "F6106656C20AB23B547B0C30DD819793"
13. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>bcm_ordercreate_response</response_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <pos_seq>151130000002</pos_seq>
7.   <sys_seq>0311095321465179</sys_seq>
8.   <trans_time>20160311095321</trans_time>
9.   <result>
10.    <id>9998</id>
11.    <comment>创建订单成功</comment>
12.  </result>
13. </business_trans>
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型，固定为bcm_ordercreate_response	必填
isspid	商户号	String(8)	接入SP的ID号，为翼码平台的提供的业务商机构号	必填
pos_id	终端号	String(20)	终端号	必填

节点名称	中文名称	类型	说明	是否必填
pos_seq	终端流水号	String(32)	终端交易流水号, 和请求报文中的交易流水号相同 创建订单成功(id=9998)之后, 将该字段作为二维码内容生成, 使用POS扫码	必填
sys_seq	随机串号	String(16)	随机生成串号, 仅用于日志跟踪用, 不保存	必填
trans_time	接口返回时间	String(14)	接口返回时间, 格式: yyyyymmddhhnss	必填
result ->id	响应码	String(4)	银行卡刷卡下单时返回 9998 表示创建订单成功, 进入轮询调用银行卡刷卡查询接口确认交易结果 其它失败, 错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解, 用于显示提示	必填

标准接口会因业务升级需要而新增节点, 请在解析返回报文时避免新增未知节点产生的程序出错!

## 4.10. 银行卡刷卡结果查询交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "bcm_resultquery_request",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "store_id": "8888888888",
6.   "pos_seq": "2018121400000005",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "bcm_resultquery_request": {
10.     "org_pos_seq": "2018121400000001",
11.     "pos_sn": "xxxxxxxxxx"
12.   },
13.   "sign": "79CB18BC49A4D60C457506EA4EF27C14"
14. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>bcm_resultquery_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>151130000003</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <bcm_resultquery_request>
11.    <org_pos_seq>151130000001</org_pos_seq>
12.    <pos_sn>xxxxxxxx</pos_sn>
13.  </bcm_resultquery_request>
14. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型, 固定为bcm_resultquery_request	必填
isspid	商户号	String(8)	接入SP的ID号, 由翼码平台分配给ipos接入受理商户的机构号 (原值返回)	必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	同一个商户必须唯一, 且大于12位	必填
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填
bcm_resultquery_request->org_pos_seq	需要查询的支付交易流水号	String(32)	需要查询的支付交易流水号, 且大于12位 创建订单时的终端流水号	必填
bcm_resultquery_request->pos_sn	POS机身号	String(20)	需要推送到的POS机身号	必填

### 交易返回报文

JSON:

```

1. {
2.   "response_type": "bcm_resultquery_response",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "pos_seq": "2018121400000005",
6.   "trans_time": "20181214143957",
7.   "sys_seq": "0f041092d0ccc6ae",
8.   "bcm_res_info": {
9.     "order_id": "1805040000021",
10.    "mid": "301310070118940",
11.    "tid": "99800010",
12.    "batch_no": "000001",
13.    "trace_no": "000001",
14.    "ref_no": "000523823884",
15.    "card_no": "622262*****9671",
16.    "trade_amount": "1",
17.    "trade_time": "20190808151515",
18.    "channel": "4"
19.  },
20.   "result": {
21.     "id": "0000",
22.     "comment": "交易成功"
23.   },
24.   "sign": "79D928004531AE15DFA6AD7E9A6A2447"
25. }
    
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>bcm_resultquery_response</response_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>123456789</pos_id>
6.   <pos_seq>000000000001</pos_seq>
7.   <sys_seq>000000000244</sys_seq>
8.   <trans_time>20090405010104</trans_time>
9.   <bcm_res_info>
10.    <order_id>1805040000021</order_id>
11.    <mid>301310070118940</mid>
12.    <tid>99800010</tid>
13.    <batch_no>000001</batch_no>
14.    <trace_no>000001</trace_no>
15.    <ref_no>000523823884</ref_no>
16.    <card_no>622262*****9671</card_no>
17.    <trade_amount>1</trade_amount>
18.    <trade_time>20190808151515</trade_time>
19.    <channel>4</channel>
20.  </bcm_res_info>
21.  <result>
22.    <id>0000</id>
23.    <comment>成功</comment>
24.  </result>
25. </business_trans>
    
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型，固定为bcm_resultquery_response	必填
isspid	商户号	String(8)	接入SP的ID号，为翼码平台的提供的业务机构号	必填

节点名称	中文名称	类型	说明	是否必填
pos_id	终端号	String(20)	终端号	必填
pos_seq	终端流水号	String(32)	终端交易流水号, 和请求报文中的交易流水号相同	必填
sys_seq	随机串号	String(16)	随机生成串号, 仅用于日志跟踪用, 不保存	必填
trans_time	接口返回时间	String(14)	接口返回时间, 格式: yyyymmddhhnss	必填
result ->id	响应码	String(4)	0000-成功 9998-等待付款 9990-交易失败 其它失败, 错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解, 用于显示提示	必填
bcm_res_info ->order_id	银行卡订单号	String(64)	刷卡通道返回的订单号	可空
bcm_res_info ->mid	银联商户号	String(64)		可空
bcm_res_info ->tid	银联设备号	String(8)		可空
bcm_res_info ->batch_no	银联批次号	String(20)		可空
bcm_res_info ->trace_no	银联流水号	String(20)		可空
bcm_res_info ->ref_no	银联交易参考号	String(20)		可空
bcm_res_info ->card_no	交易卡号	String(20)	前6后4明文, 中间隐藏	可空
bcm_res_info ->trade_amount	交易金额	String(16)	以分为单位	可空
bcm_res_info ->trade_time	系统交易时间	String(14)	支付宝、微信等平台交易时间,例如: 20190808151515	可空
bcm_res_info ->channel	交易通道	String(1)	-: POS尚未拉取订单, 通道未知 3: 交行POS 4: 星POS(翼码APP)	可空

标准接口会因业务升级需要而新增节点, 请在解析返回报文时避免新增未知节点产生的程序出错!

## 4.11. 银行卡申请退款交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "bcm_refund_request",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "store_id": "8888888888",
6.   "pos_seq": "2018121400000006",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "bcm_refund_request": {
10.    "tx_amt": "1",
11.    "org_pos_seq": "2018121400000001"
12.  },
13.   "sign": "E622B688C360F8444788740B111E4787"
14. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>bcm_refund_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>151130000004</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <bcm_refund_request>
11.    <tx_amt>10</tx_amt>
12.    <org_pos_seq>151130000001</org_pos_seq>
13.  </bcm_refund_request>
14. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为bcm_refund_request	必填
isspid	商户号	String(8)	接入SP的ID号，由翼码平台分配给ipos接入受理商户的机构号（原值返回）	必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	同一个商户必须唯一，且大于12位	必填
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填
bcm_refund_request ->org_pos_seq	需要退款的支付交易流水号	String(32)	需要退款的支付交易流水号，且大于12位 创建订单时的终端流水号	必填

### 交易返回报文

JSON:

```

1. {
2.   "response_type": "bcm_refund_response",
3.   "system_id": "8888",
4.   "isspid": "39493002",
5.   "pos_id": "8888888888",
6.   "pos_seq": "2018121400000006",
7.   "trans_time": "20181214145332",
8.   "sys_seq": "b7ff9fa5bb4355e2",
9.   "result": {
10.     "id": "9998",
11.     "comment": "申请成功"
12.   },
13.   "sign": "207482F2C0A998B73F388B8AA401DB76"
14. }
    
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <response_type>bcm_refund_response</response_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>123456789</pos_id>
6.   <pos_seq>000000000001</pos_seq>
7.   <sys_seq>00000000244</sys_seq>
8.   <trans_time>20090405010104</trans_time>
9.   <result>
10.     <id>0000</id>
11.     <comment>交易成功</comment>
12.   </result>
13. </business_trans>
    
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型，固定为bcm_refund_response	必填
isspid	商户号	String(8)	接入SP的ID号，为翼码平台的提供的业务商机构号	必填
pos_id	终端号	String(20)	终端号	必填
pos_seq	终端流水号	String(32)	终端交易流水号，和请求报文中的交易流水号相同	必填
sys_seq	随机串号	String(16)	随机生成串号，仅用于日志跟踪用，不保存	必填
trans_time	接口返回时间	String(14)	接口返回时间，格式：yyyymmddhhnss	必填
result ->id	响应码	String(4)	0000-退款成功 9998-退款结果需调用退款查询确认 9990-退款失败 其它失败，错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解，用于显示提示	必填

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

## 4.12. 银行卡刷卡退款查询交易

### 交易请求报文

JSON:

```

1. {
2.   "request_type": "bcm_refundquery_request",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "store_id": "8888888888",
6.   "pos_seq": "2018121400000005",
7.   "user_id": "user01",
8.   "system_id": "8888",
9.   "bcm_refundquery_request": {
10.     "org_pos_seq": "2018121400000001"
11.   },
12.   "sign": "79CB18BC49A4D60C457506EA4EF27C14"
13. }
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.   <request_type>bcm_refundquery_request</request_type>
4.   <isspid>39493002</isspid>
5.   <pos_id>1511300001</pos_id>
6.   <store_id>1511300001</store_id>
7.   <pos_seq>1511300000003</pos_seq>
8.   <user_id>user01</user_id>
9.   <system_id>8888</system_id>
10.  <bcm_refundquery_request>
11.    <org_pos_seq>1511300000001</org_pos_seq>
12.  </bcm_refundquery_request>
13. </business_trans>
```

### 交易请求报文说明

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	请求报文格式为JSON时必填 计算签名方式请看2.1章节	可选
request_type	请求类型	String(32)	标识请求报文的类型，固定为bcm_refundquery_request	必填
isspid	商户号	String(8)	接入SP的ID号， 由翼码平台分配给ipos接入受理商户的机构号（原值返回）	必填
pos_id	终端号	String(20)	终端号	必填
store_id	门店号	String(20)	门店号	必填
pos_seq	终端流水号	String(32)	同一个商户必须唯一，且大于12位	必填
user_id	操作员标识	String(20)	操作员号	可空
system_id	系统平台号	String(4)	由翼码提供	必填
bcm_refundquery_request->org_pos_seq	需要查询的退款交易流水号	String(32)	需要查询的退款交易流水号，且大于12位 申请退款时的终端流水号	必填

### 交易返回报文

JSON:

```

1. {
```

```

2.   "response_type": "bcm_refundquery_response",
3.   "isspid": "39493002",
4.   "pos_id": "8888888888",
5.   "pos_seq": "2018121400000005",
6.   "trans_time": "20181214143957",
7.   "sys_seq": "0f041092d0ccc6ae",
8.   "bcm_res_info": {
9.       "refund_date": "20191205103111",
10.      "refund_amt": "1"
11.  },
12.  "result": {
13.      "id": "0000",
14.      "comment": "交易成功"
15.  },
16.  "sign": "79D928004531AE15DFA6AD7E9A6A2447"
17. }
    
```

XML:

```

1. <?xml version="1.0" encoding="GBK"?>
2. <business_trans>
3.     <response_type>bcm_refundquery_response</response_type>
4.     <isspid>39493002</isspid>
5.     <pos_id>123456789</pos_id>
6.     <pos_seq>000000000001</pos_seq>
7.     <sys_seq>000000000244</sys_seq>
8.     <trans_time>20090405010104</trans_time>
9.     <bcm_res_info>
10.         <refund_date>20191205103111</refund_date>
11.         <refund_amt>1</refund_amt>
12.     </bcm_res_info>
13.     <result>
14.         <id>0000</id>
15.         <comment>成功</comment>
16.     </result>
17. </business_trans>
    
```

## 交易返回报文说明

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

节点名称	中文名称	类型	说明	是否必填
sign	报文签名	String(32)	返回报文格式为JSON时必填 计算签名方式请看2.1章节	可选
response_type	返回类型	String(32)	标识返回报文的类型，固定为bcm_refundquery_response	必填
isspid	商户号	String(8)	接入SP的ID号，为翼码平台的提供的业务机构号	必填
pos_id	终端号	String(20)	终端号	必填
pos_seq	终端流水号	String(32)	终端交易流水号，和请求报文中的交易流水号相同	必填
sys_seq	随机串号	String(16)	随机生成串号，仅用于日志跟踪用，不保存	必填
trans_time	接口返回时间	String(14)	接口返回时间，格式：yyyymmddhhnss	必填
result ->id	响应码	String(4)	0000-退款成功 9998-退款处理中 9990-退款失败 其它失败，错误信息说明见“响应码解释”字段	必填
result ->comment	响应码解释	String(64)	响应码对应中文注解，用于显示提示	必填
bcm_res_info ->refund_date	退货完成时间	String(10)	yyyymmddhhnss，只做参考，非实际退货时间	可空
bcm_res_info ->refund_amt	退货金额	String(16)	退货金额	可空

标准接口会因业务升级需要而新增节点，请在解析返回报文时避免新增未知节点产生的程序出错！

## 5.错误码说明

接入错误:

返回码	返回描述
6110	系统繁忙
6109	请求报文仅支持xml、json格式
6108	请求报文接收不正确
6107	请求方法不支持 (request_type) 节点错误
6106	XML/JSON格式错误无法解析
6105	接入平台号 (system_id) 格式错误
6104	商户号 (isspid) 格式错误
6103	发起请求的IP地址没有授权访问该接口
6102	接入平台号 (system_id) 不正确
6101	商户号 (isspid) 访问受限
6100	签名校验失败

接口传参错误:

返回码	返回描述
5200	传参错误, 对应的返回错误提示
5209	操作员编号 (user_id) 格式不正确
5210	门店号 (store_id) 格式不正确
5211	终端号 (pos_id) 格式不正确
5212	终端流水号 (pos_seq) 格式不正确或已使用
5213	统一支付数据 (barcode_info) 格式不正确
5214	支付类型/查询类型/退款类型/退款查询类型 (pay_type) 格式不正确
5215	原支付流水/原退款流水 (org_pos_seq) 格式不正确或未找到
5216	门店名称 (store_name) 格式不正确
5218	日结日期 (daily_day) 格式不正确
5221	交易金额 (tx_amt) 格式不正确
5304	支付平台商户号 (partner_id) 格式不正确
5305	账单日期 (bill_date) 格式不正确
5402	备注信息 (remark) 格式不正确
5403	商品明细 (goods_detail) 校验格式错误
5411	扩展参数 (extend_params) 格式不正确
5412	花呗分期数 (hb_fq_num) 格式不正确
5413	花呗卖家承担收费比例 (hb_fq_seller_percent) 格式不正确
5414	使用花呗分期必须上传商品明细
5415	订单类型 (type) 格式不正确
5416	订单类型 (type) 与原支付/退款订单类型不符合
5501	商户订单号 (out_trade_no) 格式不正确
5502	支付凭证图片数据 (image) 不能为空
5503	支付凭证图片数据 (image) 无法解析
5504	支付凭证图片数据 (image) 文件类型不正确

返回码	返回描述
5508	支付用户标识 (buyer_id) 格式不正确
5509	异步通知地址 (notify_url) 格式不正确
5510	支撑门店号 (iss_store_id) 格式不正确
5511	开始日期 (s_time) 格式不正确
5512	结束日期 (e_time) 格式不正确
5513	交易类型 (trade_type) 格式不正确
5514	是否使用优惠券 (use_coupon) 格式不正确
5515	POS机身号 (pos_sn) 格式不正确
5516	原交易订单号 (org_order_id) 格式不正确
5517	收款人帐号 (payee_account) 格式不正确
5518	收款人帐号类型 (payee_account_type) 取值有误
5519	收款人姓名 (payee_name) 格式不正确
5520	原现金退款流水 (org_pos_seq) 格式不正确
5521	原现金退款流水 (org_pos_seq) 未找到
5522	银联二维码银联支付标志 (pay_code) 必填

交易结果:

返回码	返回描述
0000	成功
9998	等待付款中/退款处理中
9990	交易失败
0001	交易失败, 支付参数有误。
5212	终端流水号 (pos_seq) 格式不正确或已使用 原支付流水/原退款流水 (org_pos_seq) 格式不正确或未找到
5301	终端激活失败 (翼码系统新终端激活失败)
5302	终端状态异常禁止交易
5306	平台账单文件未生成/翼码账单文件未生成
5307	支付平台商户号不属于该商户
5505	未找到该商户订单号
5506	商户订单号与交易类型不匹配
5507	商户订单号重复上传
7000	没有该终端
8000	没有该商户
7101	原流水重发请求金额与原单不一致
7102	原流水重发请求终端与原单不一致
7103	原流水重发请求收款人帐号与原单不一致
7104	原流水重发请求渠道与原单不一致
其他	交易失败, 对应错误描述

注: 根据业务发展需要或支付平台有相关调整, 返回码可能会有增删。