# *王府井支付平台接口*

修订记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **日期** | **版本号** | **修订说明** | **修订人** | **审核人** | **批准人** |
| 2015.12.15 | 1.0.0 | 初稿 | 王金刚 |  |  |

目录

*[王府井支付平台接口](#_Toc1812)* [1](#_Toc1812)

[1. 文档说明 5](#_Toc20475)

[1.1. 功能描述 5](#_Toc15606)

[1.2. 阅读对象 5](#_Toc4902)

[1.3. 业务术语 5](#_Toc7327)

[2. 功能演示 5](#_Toc29127)

[3. 数据交互 9](#_Toc7302)

[3.1. 构造请求数据 9](#_Toc24789)

[3.2. 发送请求数据 10](#_Toc15543)

[3.3. 支付平台对请求数据进行处理 10](#_Toc2899)

[3.4. 返回处理的结果数据 10](#_Toc8873)

[3.5. 对获取的返回结果数据进行处理 10](#_Toc5463)

[3.6. 对支付平台返回结果数据进行校验 10](#_Toc3246)

[4. 接口定义 10](#_Toc3126)

[访问地址和key分配 11](#_Toc8679)

[4.1. PC端创建订单请求 11](#_Toc5413)

[4.1.1.含义 11](#_Toc2794)

[4.1.2接口地址 11](#_Toc13775)

[4.1.3接口参数说明 11](#_Toc24693)

[4.2. 页面跳转同步返回数据 13](#_Toc11952)

[4.2.1.业务平台监听地址 13](#_Toc10472)

[4.2.2.含义 13](#_Toc23862)

[4.2.3.接口参数说明 13](#_Toc8794)

[4.3. 服务器异步通知数据 15](#_Toc3061)

[4.3.1.业务平台监听地址 15](#_Toc18049)

[4.3.2.含义 15](#_Toc2418)

[4.3.3.接口参数说明 15](#_Toc7888)

[4.4. 校验是否是支付平台通知 17](#_Toc25170)

[4.4.1.含义 18](#_Toc29292)

[4.5. 获取防钓鱼时间戳 18](#_Toc16877)

[4.6. PC端代理支付 18](#_Toc25837)

[4.6.1请求参数 18](#_Toc21656)

[4.6.2 返回参数 18](#_Toc5125)

[5. 同步异步说明 19](#_Toc8485)

[5.1页面跳转同步通知页面特性 19](#_Toc29570)

[5.2服务器异步通知页面特性 19](#_Toc15201)

[6. 签名机制 20](#_Toc25164)

[6.1生成签名参数说明 20](#_Toc32675)

[6.1.1.需要参与签名的参数 20](#_Toc2116)

[6.1.2.业务平台密钥参数说明 20](#_Toc18340)

[6.2.生成签名规则 20](#_Toc19505)

[6.2.1.生成待签名字符串 20](#_Toc19249)

[6.2.2.MD5 签名 21](#_Toc27177)

[6.3实例 21](#_Toc13190)

[6.3.1 Java实例（仅供参考） 22](#_Toc13839)

[6.3.2 PHP实例（仅供参考） 22](#_Toc24136)

[7. 注意事项 23](#_Toc13077)

# 文档说明

## 功能描述

通过王府井支付平台的支付渠道，付款者可以直接支付给王府井各支付渠道的账号。

## 阅读对象

本接口文档是面向接入王府井支付平台的人员，具有一定的网站开发能力，了解ASP、 PHP、 JAVA、ASP.NET 等开发语言中的一种及SQL 数据库语言的网站开发、维护和管理人员。

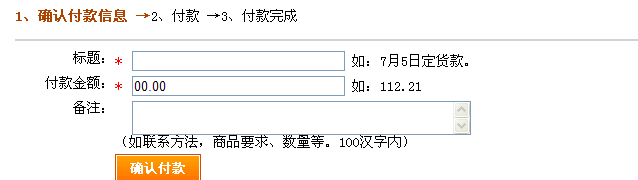
## 业务术语

|  |  |
| --- | --- |
| **名称** | **描述** |
| **业务系统** | 如OMS等接入王府井支付平台的应用 |
| **王府井支付平台** | 具有支付功能的王府井核心支付业务平台 |
| **第三方支付平台** | 如支付宝、银联、财付通等成熟的第三方支付平台 |
| **返回** | 页面跳转同步通知。王府井支付平台根据得到的数据处理完成后，当前页面从第三方支付平台的页面自动跳转回业务系统的网站，同时携带处理完成的结果信息反馈给业务系统。 |
| **请求** | 通过 HTTP、HTTPS协议把需要传输的数据发送给接收方的过程。 |
| **通知** | 服务器异步通知。王府井支付平台根据得到的数据处理完成后，王府井的服务器主动发起通知给业务系统，同时携带处理完成的结果信息反馈。 |
| **订单总金额** | 单次订单支付中的支付金额 |
| **支付账号** | 王府井的用户通行证账号 |

# 功能演示

步骤1：挑选商品并使用王府井支付平台付款。

(1) 选择商品放入购物车，生成交易参数：

(2) 跳转到王府井支付平台页面。选择付款方式：





步骤2：跳转到收银台登录，填写相应的登录信息。



步骤3：进入收银台，选择支付方式。

 

步骤4：完成付款。

付款完成后，支付宝提示成功付款

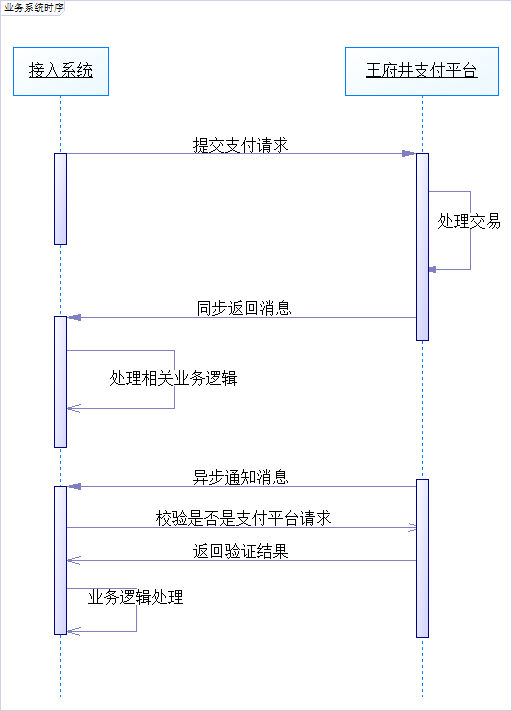


步骤5：页面自动跳转回商户网站。

若在支付时，传递给王府井支付平台的参数中包含returnUrl参数，则支付完成后，支付宝

的提示支付成功页面会停留几秒后跳转回returnUrl参数指定的商户路径页面。

# 数据交互



## 构造请求数据

业务系统根据王府井支付平台提供的接口规则，通过程序生成得到签名结果及要传输给支付平台的数据集合。

## 发送请求数据

把构造完成的数据集合，通过页面链接跳转或表单提交的方式传递给支付平台。

## 支付平台对请求数据进行处理

支付平台得到这些集合后，会先进行安全校验等验证，一系列验证通过后便会处理这

次发送过来的数据请求。

## 返回处理的结果数据

对于处理完成的交易，支付平台会以两种方式把数据反馈给业务系统。

（1）程序上自动进行重新构造URL地址链接，在用户当前页面上通过自动跳转的

方式跳回业务系统在请求时设定好的页面路径地址（参数returnUrl，如果业务系统没

有设定，则不会进行该操作）；

（2）支付平台服务器主动发起通知，调用业务系统在请求时设定好的页面路径（参数

notifyUrl，如果业务系统没有设定，则不会进行该操作）。

## 对获取的返回结果数据进行处理

业务系统在同步通知处理页面（参数returnUrl指定页面文件）或服务器异步通知页面

（参数notifyUrl指定页面文件）获取返回的结果数据后，可以结合自身的业务逻辑进行数据处理（如：订单更新等）。

## 对支付平台返回结果数据进行校验

业务系统在收到支付平台异步请求，需要对身份进行校验，需要调用王府井支付平台校验接口判断是否是支付平台发送的请求。

# 接口定义

## 访问地址和key分配

* 正式环境地址：
* 联调环境地址：
* 测试key请联系王金刚进行分配

## PC端创建订单请求

### 4.1.1.含义

请求参数是业务系统在与王府井支付平台进行数据交互时，提供给支付平台的请求数据，以便支付平台根据这些数据进一步处理。

### 4.1.2接口地址

PC端接口地址:[Http://localhost:8080/wfjpay/front/order/orderMsg.do](Https://pay.17173.com/17pay/front/order/orderMsg.do)

注意：在调用接口前，先进行相应的用户登陆，订单中的unid是否是登录状态。

### 4.1.3接口参数说明

**请求参数:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **长度** | **必填** | **说明** |
| bpId | 业务平台ID | String | 16 | 是 | 分配给业务系统的标识 |
| sign | 签名 | String | 255 | 是 | 把相关参数结合在一起通过MD5加密得出的结果，具体加密方法参见本文档附录一中的[签名机制](#_签名机制) |
| bpOrderId | Bp的ID | String | 100 | 是 | 业务系统的订单唯一标识 |
| goodsName | 商品名称 | String | 100 | 是 | 购买的商品名称 |
| totalFee | 订单总金额 | String |  | 是 | 单位为RMB元，范围[0.01，99999999.99]，精确到小数点后两位 |
| content | 订单内容 | String | 100 | 是 |  |
| remark | 订单备注 | String | 128 | 否 |  |
| bpParams | 业务平台回传参数 | String | 200 | 否 | 如果用户请求时传递了该参数，则返回给业务平台时会回传该参数 |
| notifyUrl | 异步通知地址 | String | 100 | 否 | 用于支付完毕后支付平台回调应用平台处理业务 |
| returnUrl | 页面重定向通知地址 | String | 100 | 否 | 支付完毕用，支付平台进行重定向的地址 |
| showUrl | 商品展示的链接地址 | String | 100 | 否 |  |
| orderUrl | 订单详情页面地址 | String | 100 | 否 | 业务平台的订单详情页 |
| unid | 用户的unid | String | 32 | 是 |  |
| antiPhishingKey | 防钓鱼时间戳 | String | 200 | 是 | 通过时间戳查询接口获取的加密支付平台时间戳。见4.5获取防钓鱼时间戳 |
| merCode | 签约商户编码 | String | 32 | 是 | 签约商户编码（需在支付平台申请） |
| goodsContent | 业务平台订单详情 | String | 4000 | 否 | 用来计算支付渠道和签约商户费用分摊。  格式：0000000000101^99.50^2|0000000000102^18.98^3 |

注意：所有参数使用UTF-8编码，建议使用post方式传递参数，中文无需特殊处理

如果使用GET方式，中文以及特殊符号（&，=等）需要进行URL Encoding，统一UTF-8编码

**URL 语法：**

http URL 方案最初由[RFC 1738](http://tools.ietf.org/html/rfc1738)定义（实际上，在之前的[RFC 1630](http://tools.ietf.org/html/rfc1630)也有涉及），而在http URL 方案被重新定义之前，整个 URL 语法就已经由[扩展](http://tools.ietf.org/html/rfc2396)了[几次](http://tools.ietf.org/html/rfc2732)以适应发展的[规范](http://tools.ietf.org/html/rfc1808)进化为一套统一资源标识符（Uniform Resource Identifiers 即 URIs)。

对于 URLs 如何拼装，各部分如何分隔有一套语法。例如："://"分隔方案和主机部分。主机同路径片段部分由"/"分隔，而查询部分紧跟在"?"之后。这意味着有些字符为语法保留。有些为整个URIs保留，而有些则被特定方案保留。所有出现在不应出现位置的保留符（例如路径片段——以文件名为例——可能包含"?"）必须被URL 编码。URL 编码将字符转变成对 URL 解析无意义的无害形式。它将字符转化成为一种特定[字符编码](http://en.wikipedia.org/wiki/Character_encoding)的字节序列，然后将字节转换为16进制形式，并将其前面加上"%"。问号的 URL 编码形式为"%3F"。现今多数浏览器显示 URLs 前都会对其解码（将百分号编码字节转回其原本字符），并在获取其网络资源的时候重新编码。这样一来，很多用户从未意识到编码的存在。

## 页面跳转同步返回数据

### 4.2.1.业务平台监听地址

由业务平台提供同步监听地址，用于接收王府井支付平台同步通知消息。

例如：

<http://xx.xxx.com/xxx.do>

method: post

### 4.2.2.含义

支付平台对业务系统的请求数据处理完成后，会将处理的结果数据通过系统程序控制客户

端页面自动跳转的方式返回给业务系统。

### 4.2.3.接口参数说明

**支付平台的返回参数:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | 含义 | 类型 | 长度 | 必填 | 说明 |
| tradeStatus | 交易状态 | String | 10 | 是 | 接口调用是否成功，success：成功；failure：失败；timeout:订单已超时 |
| notifyType | 通知方式 | String | 10 | 是 | 支付平台通知业务平台的类型；页面跳转的同步通知：rediectNotify，后台异步通知：asynNotify |
| sign | 签名 | String | 255 | 是 | 把相关参数结合在一起通过MD5加密得出的结果，具体加密方法参见本文档附录一中的签名机制 |
| bpOrderId | Bp平台的订单ID | String | 255 | 是 | 业务系统的订单唯一标识 |
| totalFee | 订单总金额 | Number |  | 是 | 单位RMB元，范围[0.01，100000000]，精确到小数点后两位 |
| orderTradeNo | 支付平台订单流水号 | String | 500 | 是 | 王府井基础支付平台的订单标识 |
| payTime | 付款时间 | Long |  | 是 | 用户成功付款的完成时间 |
| notifyId | 通知校验ID | String | 200 | 是 | 支付平台通知校验ID，业务平台可以用这个流水号询问支付平台该条通知的合法性。验证方法见本4.4校验是否是支付平台通知 |
| payType | 支付渠道 | String | 20 | 是 | 如通过支付宝支付:ALIPAY |
| unid | 用户登录帐号的unid | String | 128 | 是 | 对应请求时的帐号unid，原样返回 |
| bpParams | 回传参数 | String | 128 | 否 | 对应请求时的回传参数 |
| payBank | 支付银行简码 | String | 32 | 否 | 支付银行的简码 |
| payMediumCode | 支付介质编码 | String | 32 | 是 | 支付介质编码 |
| channelFeeCostDetail | 渠道佣金费用分摊 | String | 4000 | 否 | 渠道费用针对商品行项目的分摊  比如某订单走支付宝的渠道费用为1元：0000000000101^0.6|0000000000102^0.4 |
| platformFeeCostDetail | 签约商户佣金费用分摊 | String | 4000 | 否 | 签约商户费用针对商品行项目的分摊  比如应收签约商户的佣金费用为1元：0000000000101^0.6|0000000000102^0.4 |

## 服务器异步通知数据

### 4.3.1.业务平台监听地址

由业务平台提供异步监听地址，用于接收王府井支付平台异步通知消息。

例如：

<http://xx.xxx.com/xxx.do>

method : post

### 4.3.2.含义

支付平台对业务系统的请求数据处理完成后，会将处理的结果数据通过服务器主动通知的

方式通知给业务系统。

### 4.3.3.接口参数说明

**支付平台的返回参数:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | 含义 | 类型 | 长度 | 必填 | 说明 |
| tradeStatus | 交易状态 | String | 10 | 是 | 接口调用是否成功，success：成功；failure：失败；timeout:订单已超时 |
| notifyType | 通知方式 | String | 10 | 是 | 支付平台通知业务平台的类型；页面跳转的同步通知：rediectNotify，后台异步通知：asynNotify |
| sign | 签名 | String | 255 | 是 | 把相关参数结合在一起通过MD5加密得出的结果，具体加密方法参见本文档附录一中的签名机制 |
| bpOrderId | Bp平台的订单ID | String | 255 | 是 | 业务系统的订单唯一标识 |
| totalFee | 订单总金额 | Number |  | 是 | 单位RMB元，范围[0.01，100000000]，精确到小数点后两位 |
| orderTradeNo | 支付平台订单流水号 | String | 500 | 是 | 王府井支付平台的订单标识 |
| payTime | 付款时间 | Long |  | 是 | 用户成功付款的完成时间 |
| notifyId | 通知校验ID | String | 200 | 是 | 支付平台通知校验ID，业务平台可以用这个流水号询问支付平台该条通知的合法性。验证方法见本4.4校验是否是支付平台通知 |
| payType | 支付渠道 | String | 20 | 是 | 如通过支付宝支付:ALIPAY |
| unid | 用户登录帐号的unid | String | 128 | 是 | 对应请求时的帐号的unid，原样返回 |
| bpParams | 回传参数 | String | 128 | 否 | 对应请求时的回传参数 |
| payBank | 支付银行简码 | String | 32 | 否 | 支付银行的简码 |
| payMediumCode | 支付介质编码 | String | 32 | 是 | 支付介质编码 |
| channelFeeCostDetail | 渠道佣金费用分摊 | String | 4000 | 否 | 渠道费用针对商品行项目的分摊  比如某订单走支付宝的渠道费用为1元：0000000000101^0.6|0000000000102^0.4 |
| platformFeeCostDetail | 签约商户佣金费用分摊 | String | 4000 | 否 | 签约商户费用针对商品行项目的分摊  比如应收签约商户的佣金费用为1元：0000000000101^0.6|0000000000102^0.4 |

业务平台返回信息(json格式):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **长度** | **必填** | **说明** |
| status | 参数接收状态 | String | 20 | 是 | 成功:success  失败:failed |
| message | 接口反馈消息 | String | 255 | 否 | 接口错误信息提示 |

## 校验是否是支付平台通知

### 4.4.1.含义

验证此次通知信息是否是支付平台服务器发来的信息，以帮助校验反馈回来的数据的

真假性。

* 工作原理

获取支付平台返回数据之一的通知校验ID（notify\_id），按照支付平台要求的格式拼接成要请求的链接，其中orderTradeNo为支付平台返回给业务平台的支付平台订单流水号，notifyId为支付平台对于的返回参数。如：

http://localhost:8080/pp-service-impl/api/verifyNotifyId.do?orderTradeNo=20150424101004104528164&notifyId=e037c198f9ca4f9a87962f8eae776825

返回的信息即为验证是否成功，如: true : false

通过访问这个请求链接，利用编程方法来模拟http请求与支付平台服务器进行交互，获得支付平台服务器上处理的结果。如果获得的信息是true，则校验成功；如果获得的信息是其他，则校验失败。

## 获取防钓鱼时间戳

通过时间戳查询接口获取的支付平台系统时间戳地址：

http://localhost:8080/pp-service-impl/api/generateAntiPhishingKey.do

返回的信息即为支付平台时间戳.

## 4.6. PC端代理支付

业务系统直接将订单信息和[支付方式](#_4.10_获取PC端支付方式列表)传给支付平台，支付平台直接跳转到第三方支付平台进行支付（对用户来说，业务系统和支付平台的交互是透明的）。

请求地址（该地址是重定向地址）：

[http:// localhost:8080/pp-service-impl/front/order/fastPay.do](https:// pay.17173.com/pp-service-impl/api/order/getNeedPayPrice.do)

### 4.6.1请求参数

与PC端创建订单请求参数一致（[4.1.3接口参数说明](#_4.1.3接口参数说明)），在创建订单之前需要[获取防钓鱼时间戳](#_获取防钓鱼时间戳)。

### 4.6.2 返回参数

无，页面直接重定向到第三方支付平台（支付宝、财付通等）

# 5. 同步异步说明

## 5.1页面跳转同步通知页面特性

(1) 买家在支付成功后会看到一个第三方支付平台提示交易成功的页面，该页面会停留几

秒，然后会自动跳转回业务系统指定的同步通知页面（参数returnUrl）。

(2) 该页面中获得参数的方式，需要使用GET 方式获取，如

request.QueryString("out\_trade\_no")、$\_GET['out\_trade\_no']。

(3) 该方式仅仅在买家付款完成以后进行自动跳转，因此只会进行一次。

(4) 该方式不是王府井支付平台主动去调用业务系统页面，而是用页面自动跳

转的函数，使用户的当前页面自动跳转。

(5) 该方式可在本机而不是只能在服务器上进行调试。

(6) 返回URL只有一分钟的有效期，超过一分钟该链接地址会失效，验证则会失

败。

(7) 设置页面跳转同步通知页面（returnUrl）的路径时，不要在页面文件的后面

再加上自定义参数。例如：

错误的写法：http://www.wangfujing.com/pay/returnUrl.php?xx=11

正确的写法：http://www.wangfujing.com/pay/returnUrl.php

(8) 设置页面跳转同步通知页面（returnUrl）的路径时，不要设置成本机域名，也不能带有特殊字符（如“!”）

## 5.2服务器异步通知页面特性

(1) 支付平台主动发起通知，该方式才会被启用；

(2) 服务器间的交互，不像页面跳转同步通知可以在页面上显示出来，这种交互

方式是不可见的；

(3) 第一次交易状态改变时，不仅页面跳转同步通知页面会启用，而且服务器异步通知页面也会收到支付平台发来的处理结果通知；

(7) 程序执行完业务处理后。如果业务系统没有反馈处理信息给支付平台，支付平台服务器会不断重发通知，直到

超过24 小时 22 分钟。

一般情况下，25 小时以内完成8 次通知（通知的间隔频率一般是：

2m,10m,20m,1h,2h,6h,15h）；

(8) 程序执行完成后，该页面不能执行页面跳转。如果执行页面跳转，支付平台会

收不到字符，会被支付平台服务器判定为该页面程序运行出现异常，而重发处理结果通知；

(9) cookies、session等在此页面会失效，即无法获取这些数据；

(10) 该方式的调试与运行必须在服务器上，即互联网上能访问；

(11) 该方式的作用主要防止订单丢失，即页面跳转同步通知没有处理订单更新，

它则去处理；

# 6. 签名机制

## 6.1生成签名参数说明

### 6.1.1.需要参与签名的参数

在请求参数列表中，除去sign、sign\_type两个参数外，其他需要使用到的参数皆是要签名的参数。（空值参数不要参与签名，个别接口中参数sign\_type也需要参与签名。）

在通知返回参数列表中，除去sign、sign\_type两个参数外，凡是通知返回回来的

参数皆是要签名的参数。

### 6.1.2.业务平台密钥参数说明

业务平台密钥，下面简称密钥（常用代码变量名为bpKey）。支付平台会给每个业务系统分配一个（需向支付平台申请，业务平台ID即bpId也需要申请），在生成签名时，需要密钥参与签名。密钥是以英文字母和数字组成的 32位字符串。

## 6.2.生成签名规则

### 6.2.1.生成待签名字符串

对于如下的参数数组：

string[] parameters={

"service=create\_direct\_pay\_by\_user",

"partner=2088101568338364",

"returnUrl= http://www.test.com/alipay/returnUrl.asp",

"out\_trade\_no=6741334835157966",

"subject=贝尔金护腕式",

"payment\_type=1 ",

"seller\_email=alipay-test01@alipay.com",

"total\_fee=100"

};

对数组里的每一个值从 a 到 z 的顺序排序，若遇到相同首字母，则看第二个字母，

以此类推。

排序完成之后，再把所有数组值以“&”字符连接起来，如：

out\_trade\_no=6741334835157966&partner=2088101568338364

&payment\_type=1&returnUrl=http://www.test.com/alipay/returnUrl.asp&sell

er\_email=alipay-test01@alipay.com&service=create\_direct\_pay\_by\_user&subje

ct=贝尔金护腕式&total\_fee=100

这串字符串便是待签名字符串。

注意：

1. 根据HTTP 协议要求，传递参数的值中如果存在特殊字符（如：&、@等），

那么该值需要做URL Encoding，这样请求接收方才能接收到正确的参数值。这

种情况下，待签名数据应该是原生值而不是encoding 之后的值。例如：调用某

接口需要对请求参数email 进行数字签名，那么待签名数据应该是

email=test@msn.com，而不是email=test%40msn.com。

### 6.2.2.MD5 签名

*  请求时签名

当拿到请求时的待签名字符串后，需要把私钥直接拼接到待签名字符串后面，形成

新的字符串，利用MD5 的签名函数对这个新的字符串进行签名运算，从而得到32

位签名结果字符串（该字符串赋值于参数sign）。

* 通知返回时验证签名

当获得到通知返回时的待签名字符串后，同理，需要把私钥直接拼接到待签名字符

串后面，形成新的字符串，利用MD5的签名函数对这个新的字符串进行签名运算，

从而得到 32 位签名结果字符串。此时这个新的字符串需要与支付平台通知返回参数

中的参数 sign 的值进行验证是否相等，来判断签名是否验证通过。

## 6.3实例

**伪代码：**

Begin

输入：参数数组

string[] parameters=

{

'account=test@wangfujing.com',  
'antiPhishingKey=1234567891011',  
'bpId=10068',  
'content=实例',  
'goodsName=伪代码',  
'totalFee=1000000',

……(其他参数)

};

parameters=字典表排序(parameters)

while(parameter<parameters){

prestr+=parameter+'&'

}

prestr=删除末尾'&'字符(prestr);

prestr=prestr+([业务平台密钥](#_6.1.2.业务平台密钥参数说明))

sign = md5加密（sign）

End

### 6.3.1 Java实例（仅供参考）

/\*\*

\* 生成签名

\*

\* **@param**sArray

\* 要签名的数组

\* **@param** key

\* 为支付平台分配给业务平台的密钥

\* **@return**签名结果字符串

\*/

**Public static** String buildMysign(Map<String, String>sArray, String bpKey) {

List<String> keys = **new**ArrayList<String>(sArray.keySet());

// 按字典表顺序排序

Collections.sort(keys);}

// 待签名字符串

String prestr = **null**;

// 拼接参数

**for** (String key : keys) {

String value = sArray.get(key);

prestr = prestr + key + "=" + value + "&";

}

prestr.substring(0, prestr.length() - 1);

// 把拼接后的字符串再与业务平台密钥直接连接起来

prestr = prestr + bpKey;

//进行md5加密（该例子中用Alipay提供的md5加密方式）

String sign = AlipayMd5Encrypt.*md5*(prestr);

**return** sign;

}

### 6.3.2 PHP实例（仅供参考）

//设置参数  
$paramsArr= array(  
'account' =>$\_SESSION[self::QIANZHUI.'\_username'],  
'antiPhishingKey'=>$timestamp,  
'bpId' =>'10000',  
'bpOrderId' =>$insertId,  
'content' =>self::ACT\_NAME,  
'goodsName' =>self::ACT\_NAME,  
'notifyUrl' =>'http://www.wangfujing.com/callback.php',  
'returnUrl' =>'http://awww.wangfujing.com/zjdls.php',  
'totalPriceStr' =>$price,  
'unid' =>$\_SESSION[self::QIANZHUI.'\_userid'],  
);  
//参数排序  
ksort($paramsArr);  
$sign = '';  
//拼接参数  
foreach($paramsArras$k=>$v){  
$sign .= $k . '=' . $v .'&' ;  
}  
$sign = trim($sign,'&');  
//尾部拼接业务平台密钥，并进行md5码加密  
$paramsArr['sign'] = md5($sign.'密钥');

# 注意事项

当支付平台处理完成后，支付平台会把处理结果以当前页面跳转同步通知和支付平台服务器主动通知业务系统两种形式，反馈给业务系统。业务系统可以从页面跳转同步通知页面或者从服务器异步通知页面中获得这些信息，并且在验证通过的判断中增加的业务逻辑处理程序。当业务有传递参数notifyUrl（服务器异步通知页面路径）或returnUrl（页面跳转同步通知页面路径）时，业务系统必须判断商户网站中是否已经对该次的通知结果数据做过同样处理。**如果不判断，存在潜在的风险，业务系统自行承担因此而产生的所有损失**。