**爱贝微支付平台**

**服务端接入指南**

V3.4.x

爱贝公司 版权所有

2015年1月

[1 接口说明 4](#_Toc413427795)

[1.1 简介 4](#_Toc413427796)

[1.2 What’s New? 4](#_Toc413427797)

[1.3 API接口列表 4](#_Toc413427798)

[1.4 接口接入注意事项(必看) 5](#_Toc413427799)

[2 API接口定义 6](#_Toc413427800)

[2.1 登陆令牌认证(非必要) 6](#_Toc413427801)

[2.1.1功能说明 6](#_Toc413427802)

[2.1.2请求URL 6](#_Toc413427803)

[2.1.3数据格式 6](#_Toc413427804)

[2.1.4请求方式 6](#_Toc413427805)

[2.1.5请求基本业务参数 6](#_Toc413427806)

[2.1.6请求示例 6](#_Toc413427807)

[2.1.7返回结果业务参数 7](#_Toc413427808)

[2.1.8返回示例 7](#_Toc413427809)

[2.2 下单 8](#_Toc413427810)

[2.2.1 功能说明 8](#_Toc413427811)

[2.2.2 请求URL 8](#_Toc413427812)

[2.2.3 数据格式 8](#_Toc413427813)

[2.2.4 请求方式 8](#_Toc413427814)

[2.2.5 请求基本业务参数 9](#_Toc413427815)

[2.2.6 请求示例 9](#_Toc413427816)

[2.2.7 返回结果业务参数 10](#_Toc413427817)

[2.2.8 返回示例 10](#_Toc413427818)

[2.3 交易结果通知 11](#_Toc413427819)

[2.3.1 功能说明 11](#_Toc413427820)

[2.3.2 请求URL 11](#_Toc413427821)

[2.3.3 数据格式 11](#_Toc413427822)

[2.3.4 请求方式 11](#_Toc413427823)

[2.3.5 请求基本业务参数 11](#_Toc413427824)

[2.3.6 请求示例 12](#_Toc413427825)

[2.3.7 返回结果业务参数 12](#_Toc413427826)

[2.3.8返回示例 13](#_Toc413427827)

[2.4 支付结果查询（非必须实现） 13](#_Toc413427828)

[2.4.1 功能说明 13](#_Toc413427829)

[2.4.2 请求URL 13](#_Toc413427830)

[2.4.3 数据格式 13](#_Toc413427831)

[2.4.4 请求方式 13](#_Toc413427832)

[2.4.5 请求基本业务参数 14](#_Toc413427833)

[2.4.6 请求示例 14](#_Toc413427834)

[2.4.7 返回结果业务参数 14](#_Toc413427835)

[2.4.8返回示例 15](#_Toc413427836)

[2.5 游戏类一般不需用到契约型商品，可忽略以下 16](#_Toc413427837)

[2.6 契约查询 16](#_Toc413427838)

[2.6.1 功能说明 16](#_Toc413427839)

[2.6.2 请求URL 16](#_Toc413427840)

[2.6.3 数据格式 16](#_Toc413427841)

[2.6.4 请求方式 17](#_Toc413427842)

[2.6.5 请求基本业务参数 17](#_Toc413427843)

[2.6.6 请求示例 17](#_Toc413427844)

[2.6.7 返回结果业务参数 17](#_Toc413427845)

[2.5.8返回示例 18](#_Toc413427846)

[2.7 契约鉴权 19](#_Toc413427847)

[2.7.1 功能说明 19](#_Toc413427848)

[2.7.2 请求URL 19](#_Toc413427849)

[2.7.3 数据格式 19](#_Toc413427850)

[2.7.4 请求方式 19](#_Toc413427851)

[2.7.5 请求基本业务参数 20](#_Toc413427852)

[2.7.6 请求示例 20](#_Toc413427853)

[2.7.7 返回结果业务参数 20](#_Toc413427854)

[2.7.8 返回示例 21](#_Toc413427855)

[2.8 契约退订 22](#_Toc413427856)

[2.8.1 功能说明 22](#_Toc413427857)

[2.8.2 请求URL 22](#_Toc413427858)

[2.8.3 数据格式 22](#_Toc413427859)

[2.8.4 请求方式 22](#_Toc413427860)

[2.8.5 请求基本业务参数 23](#_Toc413427861)

[2.8.6 请求示例 23](#_Toc413427862)

[2.8.7 返回结果业务参数 23](#_Toc413427863)

[2.8.8 返回示例 24](#_Toc413427864)

[3 接入服务器下单SDK 24](#_Toc413427865)

[3.1服务端下单SDK构成 24](#_Toc413427866)

[3.2服务端下单SDK集成简易描述 25](#_Toc413427867)

[3.2.1复制SDK工程代码到工作工程 25](#_Toc413427868)

[3.2.2修改SDK中资源配置文件 26](#_Toc413427869)

[4 附录 26](#_Toc413427870)

[4.1feetype定义 26](#_Toc413427871)

[4.2paytype定义 26](#_Toc413427872)

[4.3code定义 27](#_Toc413427873)

# 接口说明

## 简介

本文档主要描述计费平台服务端为商户提供的API接口，用于指导商户接入并使用计费平台的相关能力

## What’s New?

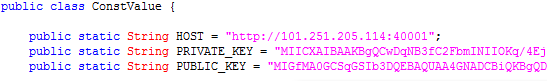
提供了服务端下单SDK，CP接入时不再需要实现数据加密、数据验签逻辑，直接调用SDK中已提供的请求对象和响应对象即可。

## API接口列表（游戏类不需要关注契约的接口）

|  |  |  |
| --- | --- | --- |
| 接口名称 | 接口说明 | 接口地址 |
| 登录令牌认证 | 商户使用计费平台账号作为用户的应用登录账号时，应用客户端调用计费平台sdk登录接口会拿到登录令牌，然后再调用本接口获取用户的账号信息。 | http://ipay.iapppay.com:9999/payapi/tokencheck |
| 下单 | 商户调用计费平台sdk支付接口之前需要先调用本接口进行下单。 | http://ipay.iapppay.com:9999/payapi/order |
| 支付结果通知 | 付完成后计费平台通过本接口向商户后台发送支付结果通知。 | Cp自定义 |
| 支付结果查询 | 商户在发起支付请求后可以调用本接口查询支付结果。 | http://ipay.iapppay.com:9999/paiapi/queryresult |
| 契约查询 | 商户可以调用本接口查询托管在计费平台的契约数据。 | http://ipay.iapppay.com:9999/payapi/subsquery |
| 契约鉴权 | 用户在使用商户的业务时，商户可以通过本接口进行使用鉴权。 | http://ipay.iapppay.com:9999/payapi/subsauth |
| 契约退订 | 商户可以调用本接口将用户的契约进行退订。 | http://ipay.iapppay.com:9999/payapi/subcancel |

## 接口接入注意事项(必看)

* 接口均采用http协议，POST方法。
* 数据传输的时候参数值需要进行urlencode，urlencode规则使用application/x-www-form-urlencoded方式，即空格” ”会被编码成加号”+”。进行urlencode编码时，中文字符采用UTF-8编码。
* 请求和应答参数均为transdata、sign、signtype。transdata为具体业务参数，数据格式为json格式；sign为transdata的签名数据；signtype为签名算法类型，目前只支持RSA算法。具体呈现方式为transdata=xxxx&sign=yyyy&signtype=RSA，其中yyyy就是对xxxx的签名数据，采用RSA MD5数字签名算法，私钥签名、公钥验签。和Java里面的MD5WithRSA算法一致。
* 商户需要以应用私钥对请求数据进行签名、使用平台公钥对应答数据进行验签。错误应答无须验签。**（应用私钥和平台公钥是两套里的，不是一对）**
* 服务端SDK中仅提供了SDK与支付服务器之间的数据加密/解密功能，用户可以根据自己业务需求决定是否实现APP和自己业务服务器之间的数据加密解密功能；
* 服务端SDK中的应用私钥和平台公钥使用商户自服务平台上提供的，即与客户端使用一套秘钥。如果使用的Jar包库，需要在系统启动前进行设置，如果是使用源代码，则可以直接修改com.ipay.util.ConstValue中的值。见下图



* + 使用Jar包，需要做如下设置：

C:\Users\visen\AppData\Roaming\Tencent\Users\6565187\QQ\WinTemp\RichOle\1_]A[EDV~0R_4R@O%2WS2_4.png

* 示例代码开发环境为JDK1.6+ Tomcat7.0.57 + eclipse for JavaEE 4.4.1

# API接口定义

## 登陆令牌认证(非必要)

### 2.1.1功能说明

商户使用计费平台账号作为用户的应用登录账号时，应用客户端调用计费平台sdk登录接口会拿到登录令牌，然后再调用本接口获取用户的账号信息。

### 2.1.2请求URL

http://ipay.iapppay.com:9999/payapi/tokencheck

### 2.1.3数据格式

json

### 2.1.4请求方式

post

### 2.1.5请求基本业务参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| appid | 应用编号 | String(20) | 必填 | 平台分配的应用编号 |
| logintoken | 登录令牌 | String(32) | 必填 | 商户调用计费平台sdk登录接口得到的登录令牌 |

请求的格式为：

transdata={"appid":"应用编号","logintoken":"数据"}&sign=xxxxxxx&signtype=RSA

sign是对transdata数据的签名。签名方式可参考SDK包内示例代码。

数据传输的时候参数值需要进行urlencode。

### 2.1.6请求示例

transdata=%7B%22**appid**%22%3A%22500000185%22%2C%22**logintoken**%22%3A%225dce3b2096ccc184f7f1d19ddde16822%22%7D&sign=OhZMFsrpVD6GkJUCZKtNQDRf0Zn9Hi0y4WMssjesW5sRoH0cmyqADo916PZbzH1OAtWbe09ap5ck4t9AMGC0D9q6YFthtYHM1BsvNKU75pRZ6Lfv%2F7%2FonhlAiRS3I%2FwbOfa8l12HKNrwjHR1ZK%2FC0BWOLziETMrj4s1eXkAO1Uo%3D&signtype=RSA

使用服务端SDK请求示例

**TokenCheckReq** tokenCheckReq = new **TokenCheckReq** ();

**tokenCheckReq.setAppId** (params.getString(ConstValue.APPID));

**tokenCheckReq.setLoginToken** (params.getString(ConstValue.LOGINTOKEN));

**TokenCheckRes** tokenCheckRes = **tokenCheckReq.send**();

### 2.1.7返回结果业务参数

**成功应答参数列表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| loginname | 账号的登录名 | String(32) | 必填 | 账号登录名 |
| userid | 账号的唯一标识 | String(20) | 必填 | 账号编号 |

**失败应答参数列表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| code | 错误代码 | String(4) | 必填 | 请求失败时，返回的错误代码。具体定义参见附录 |
| errmsg | 错误信息描述 | String(64) | 必填 | 错误原因描述 |

### 2.1.8返回示例

成功应答

transdata=%7B%22**loginname**%22%3A%2218701637882%22%2C%22**userid**%22%3A%2214382295%22%7D&sign=HU6L6dZNR0PJEgsINI5Dlt2L2WfCsN8WDAUP%2Bi%2FmLNIIwMVCHBBB6GKSrLvz10B5w5LGnX0PQf74oJx8O7JBOMJyQ7oQWoIs4NcpRi73BSxqdnt8XUTIBjfg33sfuGCCQO6GEW6gFHnocsXzNq8MIWk9mvCOFRL3pp%2FGmKdbbhQ%3D&signtype=RSA

失败应答

transdata=%7B%22**code**%22%3A1001%2C%22**errmsg**%22%3A%22%E5%BA%94%E7%94%A8%E5%8F%91%E8%B5%B7%E7%9A%84%E8%AF%B7%E6%B1%82%E7%AD%BE%E5%90%8D%E9%AA%8C%E8%AF%81%E5%A4%B1%E8%B4%A5%22%7D

使用服务端SDK解析数据实例

**TokenCheckRes** tokenCheckRes = **tokenCheckReq.send**();

JSONObject respObject = new JSONObject();

if( **tokenCheckRes.getErrorCode**() != null && **tokenCheckRes.getErrorCode**().trim().length() > 0 ){

respObject.put(ConstValue.CODE, **tokenCheckRes.getErrorCode**());

respObject.put(ConstValue.ERROR\_MSG, **tokenCheckRes.getErrMsg**() );

}else{

respObject.put(ConstValue.LOGIN\_NAME, **tokenCheckRes.getLoginName**());

respObject.put(ConstValue.USERID, **tokenCheckRes.getUserId**());

}

## 下单

### 功能说明

用户购买商户的商品时，商户先通过本接口进行下单，下单成功计费平台会返回交易流水号，商户再通过交易流水号调用计费平台sdk的支付接口。

### 请求URL

http://ipay.iapppay.com:9999/payapi/order

### 数据格式

json

### 请求方式

post

### 请求基本业务参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| appid | 应用编号 | String(20) | 必填 | 平台分配的应用编号 |
| waresid | 商品编号 | Integer | 必填 | 应用中的商品编号 |
| waresname | 商品名称 | String(32) | 可选 | 商品名称，对于消费型\_应用传入价格的计费方式有效，如果不传则展示后台设置的商品名称 |
| cporderid | 商户订单号 | String(64) | 必填 | 商户生成的订单号，需要保证系统唯一 |
| price | 支付金额 | Float(6,2) | 可选 | 支付金额，对于消费型\_应用传入价格的计费方式有效，其它计费方式不需要传入本参数 |
| currency | 货币类型 | String(32) | 必填 | 货币类型以及单位：  RMB – 人民币（单位：元） |
| appuserid | 用户在商户应用的唯一标识 | String(32) | 必填 | 用户在商户应用的唯一标识，建议为用户帐号。对于游戏，需要区分到不同区服，#号分隔；比如游戏帐号abc在01区，则传入“abc#01” |
| cpprivateinfo | 商户私有信息 | String(64) | 可选 | 商户私有信息，支付完成后发送支付结果通知时会透传给商户 |
| notifyurl | 支付结果通知地址 | String(128) | 可选 | 商户服务端接收支付结果通知的地址 |

请求的格式为：

transdata={"appid":"xxx","appuserid":"xxx","cporderid":"xxx","cpprivateinfo":"xxx","currency":"RMB","参数名":"数据","price":1,"waresid":2}&sign=xxxxxxx&signtype=RSA

sign是对transdata数据的签名。签名方式可参考SDK包内示例代码。

数据传输的时候参数值需要进行urlencode。

### 请求示例

transdata=%7B%22***appid***%22%3A%22500000185%22%2C%22***appuserid***%22%3A%22A100003A832D40%22%2C%22***cporderid***%22%3A%221421206966472%22%2C%22***cpprivateinfo***%22%3A%22cpprivateinfo123456%22%2C%22***currency***%22%3A%22RMB%22%2C%22***notifyurl***%22%3A%22http%3A%2F%2F192.168.0.140%3A8094%2Fmonizhuang%2Fapi%3Ftype%3D100%22%2C%22***price***%22%3A1%2C%22***waresid***%22%3A2%7D&sign=PNkLyWO5dxzZJrGNRJhSQGJ1oRMpvNDOHmQJntCt7OP3faT6oyL3Jc4Ne6r4IyJMxm3CAk1rxiQBoSuuAf06zsoEWbT4pNIkgqyafP4ai7zKfkJxeX7gsiG6wycT3PqRlwtmF0L7W4RDicrnAGrOQ3ynUxsrGW4oJ%2B7dKdHM4ZA%3D&signtype=RSA

使用服务端SDK请求示例

**OrderReq** orderReq = new **OrderReq**();

**orderReq.setAppId** (params.getString (ConstValue.APPID));

**orderReq.setWaresId** (params.getInt (ConstValue.WARESID));

**orderReq.setWarasName** (params.getString (ConstValue.WARESNAME));

**orderReq.setAppUserId** (params.getString (ConstValue.APPUSERID));

**orderReq.setCpOrderId** (params.getString (ConstValue.CPORDERID));

**orderReq.setCpPrivateInfo** (params.getString (ConstValue.CPPRIVATEINFO));

**orderReq.setCurrency** (params.getString (ConstValue.CURRENCY));

**orderReq.setNotifyUrl** (params.getString (ConstValue.NOTIFYURL));

**orderReq.setPrice** (Float.parseFloat (params.getString (ConstValue.PRICE)));

**OrderRes** orderRes = **orderReq.send()**;

### 返回结果业务参数

**成功应答参数列表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| transid | 交易流水号 | String(32) | 必填 | 计费支付平台的交易流水号 |

**失败应答参数列表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| code | 错误代码 | String(4) | 必填 | 请求失败时，返回的错误代码。具体定义参见附录 |
| errmsg | 错误信息描述 | String(64) | 必填 | 错误原因描述 |

### 返回示例

成功应答

transdata=%7B%22**transid**%22%3A%2232011501141440430237%22%7D&sign=NJ1qphncrBZX8nLjonKk2tDIKRKc7vHNej3e%2FjZaXV7Gn%2Fm1IfJv4lNDmDzy88Vd5Ui1PGMGvfXzbv8zpuc1m1i7lMvelWLGsaGghoXi0Rk7eqCe6tpZmciqj1dCojZoi0%2FPnuL2Cpcb%2FaMmgpt8LVIuebYcaFVEmvngLIQXwvE%3D&signtype=RSA

失败应答

transdata=%7B%22**code**%22%3A1002%2C%22**errmsg**%22%3A%22%E8%AF%B7%E6%B1%82%E5%8F%82%E6%95%B0%E9%94%99%E8%AF%AF%22%7D

使用服务端SDK解析数据实例

**OrderRes** orderRes = **orderReq.send** ();

JSONObject respObject = new JSONObject ();

if( **orderRes.getErrorCode**() != null &&**orderRes.getErrorCode**().trim().length() > 0 ){

respObject.put ("code", **orderRes.getErrorCode** ());

respObject.put ("ErrMsg", **orderRes.getErrMsg** () );

else{

respObject.put ("transid", **orderRes.getTransid**());

}

## 交易结果通知

### 功能说明

用户在计费平台完成支付或契约退订后，如果商户在下单接口有传入notifyurl，则交易平台会主动向商户发送支付结果通知。

平台在没有收到商户应答的情况下，会定时重发。重发超过一定次数后，将不再进行重发。

### 请求URL

商户在下单接口传入的notifyurl。

### 数据格式

json

### 请求方式

post

### 请求基本业务参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| transtype | 交易类型 | integer | 必填 | 交易类型：  0–支付交易；  1–支付冲正（暂未启用）；  2–契约退订  3–自动续费 |
| cporderid | 商户订单号 | String(64) | 可选 | 商户订单号 |
| transid | 交易流水号 | String(32) | 必填 | 计费支付平台的交易流水号 |
| appuserid | 用户在商户应用的唯一标识 | String(32) | 必填 | 用户在商户应用的唯一标识 |
| appid | 游戏id | String(20) | 必填 | 平台为商户应用分配的唯一代码 |
| waresid | 商品编码 | integer | 必填 | 平台为应用内需计费商品分配的编码 |
| feetype | 计费方式 | integer | 必填 | 计费方式，具体定义见附录 |
| money | 交易金额 | Float(6,2) | 必填 | 本次交易的金额 |
| currency | 货币类型 | String(32) | 必填 | 货币类型以及单位：  RMB – 人民币（单位：元） |
| result | 交易结果 | integer | 必填 | 交易结果：  0–交易成功  1–交易失败 |
| transtime | 交易完成时间 | String(20) | 必填 | 交易时间格式：  yyyy-mm-dd hh24:mi:ss |
| cpprivate | 商户私有信息 | String(64) | 可选 | 商户私有信息 |
| paytype | 支付方式 | integer | 可选 | 支付方式，具体定义见附录 |

### 请求示例

无，该接口提供给支付服务器调用

### 返回结果业务参数

**成功应答：**

商户成功接收到支付结果通知之后，返回http包体数据为SUCCESS。

**失败应答：**

商户如果接收失败，则返回http包体数据为FAILURE。

### 2.3.8返回示例

回调的格式为：

transdata={"appid":"xxx","appuserid":"xxx","cporderid":"xxx","cpprivateinfo":"xxx","currency":"RMB","参数名":"数据","price":1,"waresid":2}&sign=xxxxxxx&signtype=RSA

sign是对transdata数据的签名。验签方式可参考SDK包内示例代码。

数据传输的时候参数值会做urlencode。

transdata=%7B%22**appid**%22%3A%22500000185%22%2C%22**cporderid**%22%3A%221421220957285%22%2C%22**cpprivate**%22%3A%22cpprivateinfo123456%22%2C%22**currency**%22%3A%22RMB%22%2C%22**feetype**%22%3A6%2C%22**money**%22%3A1.0%2C%22**paytype**%22%3A5%2C%22result%22%3A0%2C%22**transid**%22%3A%2232011501141535580250%22%2C%22**transtime**%22%3A%222015-01-14+15%3A36%3A23%22%2C%22**waresid**%22%3A4%7D&sign=cbKD9RDjYpjgPMaaWJ0K4q7f6ldXOj8QA%2BIWkPdCfjmOlj2Gb9pRQg88CAONSB49wsNvHkiM%2B4OAOpqcohN%2FkdgDc92fmqzGcPb5GyUSdDacKHWFqFkyxxeboFiH0P5SrDIsklHBAfBiWP5DXTnL1Y6RiG52JyjX2sRWnGm1dAI%3D&signtype=RSA

## 支付结果查询（非必须实现）

### 功能说明

本接口适用于在支付交易完成一段时间后，如果CP没有接收到交易结果同步，则可主动向支付平台服务端发起交易结果查询。

### 请求URL

http://ipay.iapppay.com:9999/payapi/queryresult

### 数据格式

json

### 请求方式

post

### 请求基本业务参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| appid | 应用代码 | String(20) | 必填 | 商户在支付平台的应用代码 |
| cporderid | 外部订单号 | String(64) | 必填 | 应用平台的交易订单号 |

### 请求示例

transdata=%7B%22**appid**%22%3A%22500000185%22%2C%22**cporderid**%22%3A%221421220957285%22%7D&sign=oWwgYyT83JYM%2Bflmy0fC3erhJSmK5DeVcRKU5wpRiBr96TRbVnCSJ%2B50gGgMIVzYK6JaQ7VBTi7mWhQVzx%2Fm3XRqhoJ%2F5sxuEKp3Iy8w%2FDtSlyuVUKO0OmoMQ5d8t%2BoqcQUeFYFs%2FI6qeWECx998Bf0KAGbIZH63EGFSbzmBBWs%3D&signtype=RSA

使用服务端SDK请求示例

**QueryPayResultReq** request = new **QueryPayResultReq** ();

**request.setAppId** ( params.getString (ConstValue.APPID) );

**request.setCpOrderId** ( params.getString (ConstValue.CPORDERID) );

**QueryPayResultRes** response = **request.send** ();

### 返回结果业务参数

**成功应答参数列表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| cporderid | 商户订单号 | String(64) | 必填 | 商户订单号 |
| transid | 交易流水号 | String(32) | 必填 | 计费支付平台的交易流水号 |
| appuserid | 用户在商户应用的唯一标识 | String(32) | 必填 | 用户在商户应用的唯一标识 |
| appid | 游戏id | String(20) | 必填 | 平台为商户应用分配的唯一代码 |
| waresid | 商品编码 | integer | 必填 | 平台为应用内需计费商品分配的编码 |
| feetype | 计费方式 | integer | 必填 | 计费方式，具体定义见附录 |
| money | 交易金额 | Float(6,2) | 必填 | 本次交易的金额 |
| currency | 货币类型 | String(32) | 必填 | 货币类型以及单位：  RMB – 人民币（单位：元） |
| result | 交易结果 | integer | 必填 | 交易结果：  0–交易成功；  2–待支付 |
| transtime | 交易完成时间 | String(20) | 必填 | 交易时间格式：  yyyy-mm-dd hh24:mi:ss |
| cpprivate | 商户私有信息 | String(64) | 可选 | 商户私有信息 |
| paytype | 支付方式 | integer | 可选 | 支付方式，具体定义见附录 |

**失败应答参数列表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数含义 | 数据类型 | 是否可选 | 参数说明 |
| code | 错误代码 | String(4) | 必填 | 请求失败时，返回的错误代码。具体定义参见附录 |
| errmsg | 错误信息描述 | String(64) | 必填 | 错误原因描述 |

### 2.4.8返回示例

成功应答

transdata=%7B%22**appid**%22%3A%22500000185%22%2C%22**cporderid**%22%3A%221421220957285%22%2C%22**cpprivate**%22%3A%22cpprivateinfo123456%22%2C%22**currency**%22%3A%22RMB%22%2C%22**feetype**%22%3A6%2C%22**money**%22%3A1.0%2C%22**paytype**%22%3A5%2C%22**result**%22%3A0%2C%22**transid**%22%3A%2232011501141535580250%22%2C%22**transtime**%22%3A%222015-01-14+15%3A36%3A23%22%2C%22**waresid**%22%3A4%7D&sign=cbKD9RDjYpjgPMaaWJ0K4q7f6ldXOj8QA%2BIWkPdCfjmOlj2Gb9pRQg88CAONSB49wsNvHkiM%2B4OAOpqcohN%2FkdgDc92fmqzGcPb5GyUSdDacKHWFqFkyxxeboFiH0P5SrDIsklHBAfBiWP5DXTnL1Y6RiG52JyjX2sRWnGm1dAI%3D&signtype=RSA

失败应答

transdata=%7B%22**code**%22%3A1001%2C%22**errmsg**%22%3A%22%E5%BA%94%E7%94%A8%E5%8F%91%E8%B5%B7%E7%9A%84%E8%AF%B7%E6%B1%82%E7%AD%BE%E5%90%8D%E9%AA%8C%E8%AF%81%E5%A4%B1%E8%B4%A5%22%7D

使用服务端SDK解析数据实例

**QueryPayResultRes** response = **request.send** ();

JSONObject respObject = new JSONObject();

if( **queryPayRes.getErrorCode**() != null &&**queryPayRes.getErrorCode**().trim().length() > 0 ){

respObject.put(ConstValue.CODE, **queryPayRes.getErrorCode**());

respObject.put (ConstValue.ERRMSG, **queryPayRes.getErrMsg**() );

}else{

respObject.put(ConstValue.RESULT, **queryPayRes.getResult**());

respObject.put(ConstValue.CPORDERID, **queryPayRes.getCpOrderId**());

respObject.put(ConstValue.TRANSID, **queryPayRes.getTransId**());

respObject.put(ConstValue.APPUSERID, **queryPayRes.getAppUserId**());

respObject.put(ConstValue.APPID, **queryPayRes.getAppId**());

respObject.put(ConstValue.WARESID, **queryPayRes.getWaresId**());

respObject.put(ConstValue.FEETYPE, **queryPayRes.getFeeType**());

respObject.put(ConstValue.MONEY, **queryPayRes.getMoney**());

respObject.put(ConstValue.CURRENCY, **queryPayRes.getCurrency**());

respObject.put(ConstValue.TRANSTIME, **queryPayRes.getTransTime**());

respObject.put(ConstValue.CPPRIVATEINFO, **queryPayRes.getCpPrivateInfo**());

respObject.put(ConstValue.PAYTYPE, **queryPayRes.getPaytype**());

}

# 接入服务器下单SDK

## 3.1服务端下单SDK构成

当您解压完SDK（3.4.0）的压缩包后，里面包括的文件如下：

|  |  |
| --- | --- |
| 01-接入必看-爱贝服务端接入指南 | 服务端接入资料 |
| 02-接入必看-服务端接入示例 | 服务端接入示例 |

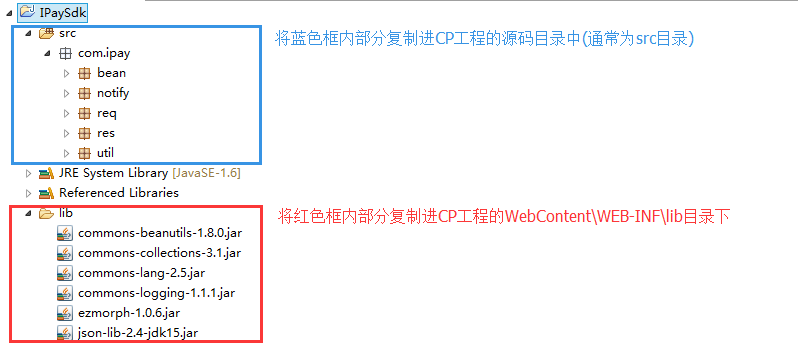
注意:

示例代码和服务端下单SDK支持JDK1.6以上版本

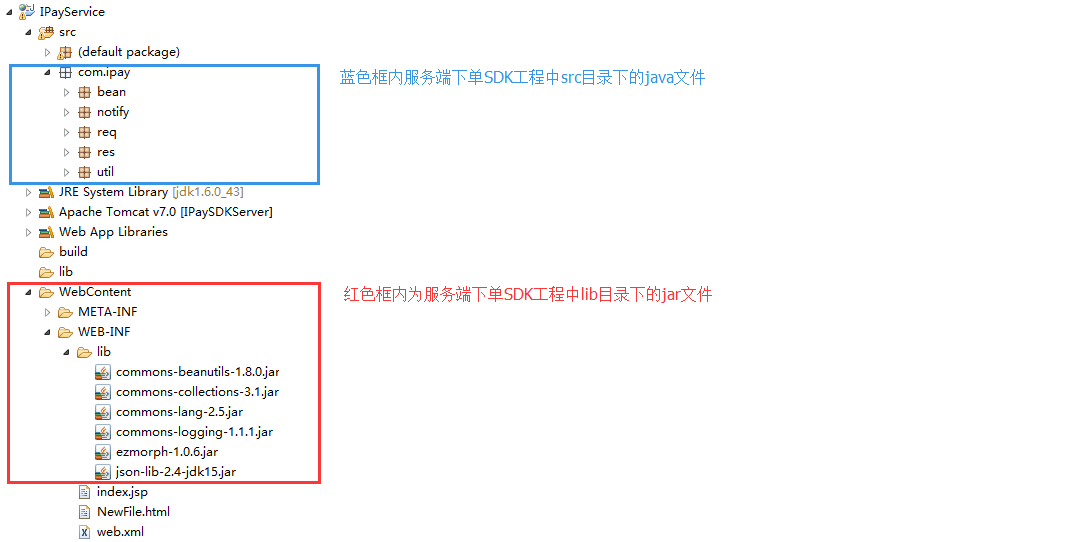
## 3.2服务端下单SDK集成简易描述

### 3.2.1复制SDK工程代码到工作工程

将SerrverSDK.zip解压缩后，将工程中src下的代码复制到工作工程，同时将lib目录下的jar文件复制到工作工程中WebContent\WEB-INF\lib目录下



服务端下单SDK代码结构



示例代码加入服务端SDK后的代码结构

### 3.2.2修改SDK中资源配置文件

复制SDK工程代码到工作工程后，注意修改*com.ipay.util.ConstValue.java*中*HOST、PRIVATE\_KEY、PUBLIC\_KEY*这三个静态变量的值。

HOST：当前支付服务器地址，通常不需要更改；

PRIVATE\_KEY：商户自服务平台获取到的应用私钥；

PUBLIC\_KEY:商户自服务平台获取到的平台公钥；

# 附录

## 4.1feetype定义

feetype表示商品采用的计费方式，目前定义如下：

0 – 消费型\_应用传入价格

2 – 消费型\_平台设置价格

3 – 订阅型\_按时长

4 – 订阅型\_自动续费

5 – 非消费型

6 – 订阅型\_按次数

因为业务拓展，计费方式定义可能会扩展，建议商户不要做严格校验

## 4.2paytype定义

paytype表示用户采用的支付方式，目前定义如下：

1   - 充值卡

2   - 游戏点卡

4   - 银行卡

5   - 爱贝币

6   - 爱贝快付

8   - 海马币（限海马版本）

16  - 百度钱包

30  - 移动话费

31  - 联通话费

32  - 电信话费

401 - 支付宝

402 - 财付通

因为业务拓展，支付方式定义可能会扩展，建议商户不要做严格校验

## 4.3code定义

code表示计费平台处理请求失败时返回的错误代码，目前定义如下：

1001– 应用发起的请求签名验证失败

1002– 请求参数错误

2001– 登录令牌已过期

2002 – 登录令牌认证失败

3001– 商户订单号已存在

3002– 商品数据异常

3003–支付金额为0，不需要支付

3004– 用户存在有效契约，不需要支付

4001– 订单正在处理

5001– 用户不存在有效契约

5002– 用户契约已过期

5003– 用户契约已过完

9999– 系统繁忙，请稍后再试