**读书计费需求及接口定义**

**iPAD Reader Payment Proposal & IF Definition**

**Author**

**Revision No – 0.1**

**Issue Date – 2011-06-01**

|  |  |
| --- | --- |
| **DOCUMENT PDS ID:** | |
| **PRODUCT: MOBEE** | **RELEASE: 0.1** |
| **FEATURE:** | **SUBSYSTEM: MOBEE** |
| **DISTRIBUTE TO:** | |

***Borqs* Proprietary**

Not for use or disclosure outside ***Borqs, Inc.*** or any of its subsidiaries

Except under prior written agreement.

Other Intended Audience: Quality Assurance Group & Internal Auditors.

This document is controlled electronically and any hard copy should be considered uncontrolled.

The latest version of this document may be obtained from the PDS ID -.

This document is owned by **<<Manager / Author>>**.

Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Revision #** | **Author** | **Reviewed By** | **A. Description Of Change, Including MR No.**  **B. Summary of Review** | **Issued by / Date (yyyy-mm-dd)** |
| 0.1 (draft) |  |  | New document | 2011-06-01 |

**Table of Contents**

[1 读书计费需求 4](#_Toc295315461)

[2 处理流程 4](#_Toc295315462)

[2.1 图书展示 4](#_Toc295315463)

[2.2 积分购买 4](#_Toc295315464)

[2.3 现金购买 5](#_Toc295315465)

[2.4 充值流程 6](#_Toc295315466)

[2.5 积分兑换 6](#_Toc295315467)

[3 接口定义 6](#_Toc295315468)

[3.1 客户端⬄服务器计费接口 6](#_Toc295315469)

[3.1.1 查询帐号 7](#_Toc295315470)

[3.1.2 查询积分兑换比例 7](#_Toc295315471)

[3.1.3 积分兑换 8](#_Toc295315472)

[3.1.4 查询充值方式 8](#_Toc295315473)

[3.1.5 现金充值 9](#_Toc295315474)

[3.1.6 现金购买 9](#_Toc295315475)

[3.1.7 积分购买 10](#_Toc295315476)

[3.1.8 消费次数 10](#_Toc295315477)

[3.1.9 充值消费明细 11](#_Toc295315478)

[3.2 书籍管理⬄商品管理接口 12](#_Toc295315479)

[3.2.1 查询包/书/章节价格 12](#_Toc295315480)

[3.2.2 批量查询包/书/章节价格 12](#_Toc295315481)

[3.2.3 查询用户是否已购买过商品 12](#_Toc295315482)

[3.2.4 批量查询用户是否已经购买过商品 12](#_Toc295315483)

[3.2.5 批量新增书籍定价 12](#_Toc295315484)

[3.2.6 批量新增章节定价 13](#_Toc295315485)

[4 后台管理页面 13](#_Toc295315486)

[4.1 定价页面 13](#_Toc295315487)

[4.1.1 书籍定价 13](#_Toc295315488)

[4.1.2 章节定价 13](#_Toc295315489)

[4.1.3 打包定价 13](#_Toc295315490)

[4.1.4 自动定价（新增书籍） 13](#_Toc295315491)

[4.2 统计数据 14](#_Toc295315492)

[4.2.1 查看第三方支付平台充值明细和统计 14](#_Toc295315493)

[4.2.2 查看书籍消费明细和统计 14](#_Toc295315494)

[5 数据定义 14](#_Toc295315495)

[5.1.1 业务类别 14](#_Toc295315496)

[5.1.2 业务计划 14](#_Toc295315497)

[5.1.3 商品定义 15](#_Toc295315498)

[5.1.4 用户帐号 15](#_Toc295315499)

[5.1.5 用户业务状态 15](#_Toc295315500)

[5.1.6 充值交易记录 16](#_Toc295315501)

[5.1.7 统一支付平台状态码 16](#_Toc295315502)

# 读书计费需求

读书计费的要求：

1. 支持两种购买方式：积分购买、现金（或虚拟币）购买；

同一本书在积分兑换专区按积分定价，在分类专区按现金定价。

1. 支持多个第三方支付平台（支付宝、移动手机支付等）；
2. 支持多种收费模式：

1）单本定价收费  ；

2）按章节收费 ，比如前10章节免费  第10章节以后每章节收费0.1元 ；

3)  10000本书中，100本书打包，按月收费；

4）用户自动购买整本书所有章节，下载时自动扣费；

5）按字数收费，所有书籍统一按1000字5分钱收费；

4、按用户级别不同给予不同的优惠，比如金卡0.95折，银卡0.98折。

# 处理流程

## 图书展示



## 积分购买

在积分兑换专区看到以积分定价的书籍，可直接用积分兑换该商品，兑换记录在积分明细和消费明细中都可查询。

商品管理统一采用现金定价，在积分兑换专区以规定比率转换为积分值并展示，购买时直接扣除用户积分。





## 现金购买



## 充值流程



说明：不涉及用户如何将现金转入第三方支付平台！充值后存入个人账户，特殊情况可由客服将余额转回客户的第三方支付平台账户。

## 积分兑换

积分以一定的兑换率兑换为现金，该现金可消费，不可提现。



# 接口定义

## 客户端⬄服务器计费接口

以下接口如需区分语言，需读取HTTP头部X-Bqs-Lang信息，并返回不同的语言信息。

HTTP头部扩展参数如下：

|  |  |
| --- | --- |
| 名称 | 描述 |
| X-Borqs-Client | 与《mSpaces\_Interface\_Control\_Document》相同 |
| X-Device-Res | 屏幕尺寸，例如 320x480 |
| X-Device-DensityFactor | 屏幕点密度，例如1.0，2.0  屏幕尺寸x屏幕点密度可以获得屏幕像素，  例如bqsScrRes=320x480, bqsScrSacle=2.0，则屏幕像素为640x960 |
| X-Bqs-Market | 下载源：apple，91，weiphone |
| X-Bqs-Lang | 客户端当前显示语言，中文=zhcn，英文=en |

### 查询帐号

HTTP GET请求URL： /bpay/myAccount.action

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <bpay>  <userId>123</userId>  <status>0</status>  <account>10.00</account>  <unit>元</unit>  </bpay> |

输出字段说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| userId | bpay | String | 用户ID |
| status | bpay | Integer | 0 正常1冻结 |
| account | bpay | Float | 账户余额 |
| unit | bpay | String | 帐号单位，中文（元）英文（$） |

### 查询积分兑换比例

URL： /bpay/scoreRule.action

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <bpay>  <rule>兑换为现金（1:1000）</rule>  </bpay> |

输出字段说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| rule | bpay | String | 用户可读的积分⬄现金兑换规则。需区分语言返回。 |

### 积分兑换

URL： /bpay/scoreChange.action?score=5000

输入参数为：

|  |  |  |
| --- | --- | --- |
| **HTTP Get 参数** | **类型** | **说明** |
| score | Integer | 需兑换的积分数 |

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <bpay>  <result>200</result>  </bpay> |

输出字段说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| result | bpay | String | 200 成功  400 兑换值超过当前积分  500 服务器原因兑换失败 |

### 查询充值方式

URL： /bpay/paymentMethods.action

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <bpay>  <type>  <id> 1</id>  < name>支付宝</name>  </type>  <type>  <id> 2</id>  < name>移动手机支付</name>  </type>  </bpay> |

输出字段说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| type | bpay | 节点 |  |
| id | type | Integer | 支付平台ID |
| name | type | String | 用户可读的支付平台名称, 需区分语言。 |

### 现金充值

URL: /bpay/paymentExec.action?typeId=1&fee=50

输入参数为：

|  |  |  |
| --- | --- | --- |
| **HTTP Get 参数** | **类型** | **说明** |
| typeId | Integer | 对应[3.1.4](#_充值方式查询)接口的id |
| fee | Float | 充入的费用 |

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <bpay>  <result>200</result>  <url>http://alipay.com/sessionkey=\*\*\*\*</url>  </bpay> |

输出字段说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| result | bpay | Integer | 服务器处理结果：  200 ok  500 服务器其他错误 |
| url | bpay | String | 需用浏览器跳转的url地址。 |

### 现金购买

URL: /bpay/cashBuy.action?goodsId=999&planId=21

输入参数为：

|  |  |  |
| --- | --- | --- |
| **HTTP Get 参数** | **类型** | **说明** |
| goodsId | Long | 商品ID |
| planId | Long | 业务计划ID，该业务计划指定了按积分、现金、包月购买等。 |

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <bpay>  <result>200</result>  </bpay> |

输出字段说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| result | bpay | Integer | 服务器处理结果：  200 购买成功  300 账户余额不足  400请求参数错误  500 服务器其他错误 |

### 积分购买

URL: /bpay/scoreBuy.action?goodsId=999&planId=21

输入参数为：

|  |  |  |
| --- | --- | --- |
| **HTTP Get 参数** | **类型** | **说明** |
| goodsId | Long | 商品ID |
| planId | Long | 业务计划ID，该业务计划指定了按积分、现金、包月购买等。 |

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <bpay>  <result>200</result>  </bpay> |

输出字段说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| result | bpay | Integer | 服务器处理结果：  200 购买成功  300 积分余额不足  400 请求参数错误  500 服务器其他错误 |

### 消费次数

查询用户在指定时间的交易记录个数，用于分页。

URL: /bpay/myTransNum.action?serviceId=5&timeStart=2011-05-01&timeEnd=2011-06-01

输入参数:

|  |  |  |
| --- | --- | --- |
| **HTTP Get 参数** | **类型** | **说明** |
| type | Integer | 类型，可选参数。  0 查询消费明细  1 查询充值明细  NULL查询所有明细。 |
| serviceId | Integer | 业务类别，  5 -读书 |
| timeStart | Date | 开始时间，null表示不设定开始时间 |
| timeEnd | Date | 结束时间, null表示不设定结束时间 |

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <bpay>  <number>10</number>  </bpay> |

输出字段说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| number | bpay | Integer | 交易记录数。 |

### 充值消费明细

URL: /bpay/myTransRecord.action?serviceId=5&timeStart=2011-05-01&timeEnd=2011-06-01&currPage=0&pageSize=20

输入参数:

|  |  |  |
| --- | --- | --- |
| **HTTP Get 参数** | **类型** | **说明** |
| type | Integer | 类型，可选参数。  0 查询消费明细  1 查询充值明细  NULL查询所有明细。 |
| serviced | Integer | 业务类别，  5 -读书 |
| timeStart | Date | 开始时间null表示不设定开始时间 |
| timeEnd | Date | 结束时间null表示不设定开始时间 |
| currPage | Integer | 当前页，从0开始 |
| pageSize | Integer | 每页显示量 |

服务端返回的xml格式数据，举例如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  < bpay>  <trade>  <tradeTime>2011-06-03</tradeTime>  <amount>1000</amount>  <unit>积分</unit>  <goodsName>斗破苍穹第一章</goodsName>  </trade>  <trade>  <tradeTime>2011-06-03</tradeTime>  <amount>0.1</amount>  <unit>元</unit>  <goodsName>斗破苍穹第二章</goodsName>  </trade>  <trade>  <tradeTime>2011-06-03</tradeTime>  <amount>50</amount>  <unit>元</unit>  <goodsName>支付宝</goodsName>  </trade>  </ bpay> |

输出字段说明:

|  |  |  |  |
| --- | --- | --- | --- |
| 标签 | 父结点 | 类型 | 描述 |
| trade | bpay | node | 消费记录 |
| tradeTime | trade | Date | 交易时间 |
| amount | trade | String | 交易金额 |
| goodsName | trade | String | 商品名称 |
| unit | trade | String | 货币单位，中文（元）英文（$） |

## 书籍管理⬄商品管理接口

### 查询包/书/章节价格

List<[Goods](#_商品定义)> queryGoodsPrice([ServiceType](#_业务类别) **serviceType**, Integer **refType**, Integer **refID**);

查询书或章节的商品及价格信息。

参数：

*serviceType*  – 业务类别 Book

*refType* – TYPE\_PACKAGE(打包)，TYPE\_BOOK(书)，TYPE\_CHAPTER（章）

*refID* – 由refType对应的ID信息

返回：

书或章节对应的商品信息和定价。

### 批量查询包/书/章节价格

List<Goods> batchQueryGoodsPrice(ServiceType serviceType, Integer refType, List<Integer> refIDList);

批量查询书或章节的商品及价格信息。

参数：

*serviceType*  – 业务类别 Book

*refType* – TYPE\_PACKAGE(打包) TYPE\_BOOK(书)，TYPE\_CHAPTER（章）

*refIDList* – 由refType对应的ID信息

返回：

书或章节对应的商品信息和定价。

### 查询用户是否已购买过商品

public Boolean findGoodInUserTradeRecord(Long userId, Long goodsId);

### 批量查询用户是否已经购买过商品

Public Map<Long, Boolean>batchFindGoodInUserTradeRecord(Long userId, List<Long> goodsList);

### 批量新增书籍定价

Public Boolean newBooksSetPrice(List<Integer> **bookID**);

参数：

*bookID* – Book ID信息

返回：

True 表示成功， False表示失败

### 批量新增章节定价

Public Boolean newChaptersSetPrice(Map<Integer, Integer>chapterMap);

参数：

chapterMap *<chapterID* *, word>*

*chapterID* – 章节ID

*word* – 该章节对应的字数

返回：

True 表示成功， False表示失败。

# 后台管理页面

## 定价页面

### 书籍定价

管理员可对某本书单独定价，也可一次性对所有书籍批量定价。

### 章节定价

管理员可对某章节单独定价，也可一次性对一本书的所有章节，书库中的所有章节批量定价。

### 打包定价

管理员可对打包书籍单独定价，也可一次性对书库中的所有打包书籍批量定价。

### 自动定价（新增书籍）

商品管理根据自动定价规则（如章节每1000字\*\*\*元，单本\*\*\*元/本），对新增书籍进行定价

后台管理可以调整自动定价规则。

## 统计数据

### 查看第三方支付平台充值明细和统计

### 查看书籍消费明细和统计

同上。

提供该页面的查看权限给盛大。

# 数据定义

### 业务类别

Enum ServiceType { Music，News，，Video，Application，Photo， Contacts， Suit，Book}

### 业务计划

class ServicePlan{

protected Long id； //计划ID

protected String name； //计划名称, 如音乐在线包月计划

protected Float price; //价格（元）

protected Integer Unit; //价格单元，虚拟币（0），积分（1）

protected Integer expireUnit; //有效期描述单位。天(0), 周(1), 月(2), 季度（3）, 年(4), 当时有效(5), 无限期(6)

protected Integer expireValue; //有效期数值。单位和数值决定业务计划的有效期。

protected Boolean repeat; //描述计划到期后是否自动续期

protected String description; //业务计划的描述信息，提供给用户直观的业务计划描述，如“5元/月畅听所有歌曲，每月自动扣费”

public static final int UNIT\_DAY=0;

public static final int UNIT\_WEEK=1;

public static final int UNIT\_MONTH=2;

public static final int UNIT\_QUARTER=3;

public static final int UNIT\_YEAR=4;

public static final int UNIT\_CURRENT=5;

public static final int UNIT\_UNLIMITED=6;

}

### 商品定义

class Goods{

protected Long id; //商品ID

protected ServiceType type; //业务类别

protected Integer refType ; //引用类别 （0-package，1-product）

protected Long refID; //引用ID

protected Integer opCode; //操作类别（online、download）

List<[ServicePlan](#_业务计划)> servicePlan;

public static final int REF\_TYPE\_ PACKAGE=0;

public static final int REF\_TYPE\_ PRODUCT=1;

public static final int OP\_ONLINE=0;

public static final int OP\_DOWNLOAD=1;

}

### 用户帐号

class UserAccount{

Protected Long userID;

Protected Integer status; //激活、未激活、冻结

Protected Float virtual\_account; //虚拟帐户

Protected Float recharege\_account; //充值帐户

public static final int STATUS\_ACTIVE=0; //帐号激活

public static final int STATUS\_DEACTIVE=1; //帐号未激活

public static final int STATUS\_FREEZE=2; //帐号冻结

public static final int ACCOUNT\_VIRTUAL=0; //虚拟帐号

public static final int ACCOUNT\_RECHARGE=1; //充值帐号

}

### 用户业务状态

Class UserService{

Protected Long userID;

Protected Long goodsID;

Protected ServicePlan servicePlan;

Protected Date expireTime; //到期时间

Protected Integer **status**; //1.正常、2.暂停、3.取消、200.扣费失败、201.账户余额不足

}

### 充值交易记录

Class PaymentRecord{

Protected Long id; //交易流水号

Protected Long userID;

Protected Long accountType; //帐号类型

Protected Date payTime; //交易时间

Protected Float amount; //交易金额

Protected Float balance; //帐号余额

Protected String remark; //购买商品信息、充值信息等

}

### 统一支付平台状态码

Class PaymentStatusCode{

public static final int PAYMENT\_OK = 200;

public static final int PAYMENT\_NO\_MONEY = 300;

public static final int PAYMENT\_ LACK\_MONEY= 301;

public static final int PAYMENT\_BAD\_REQUEST = 400;

public static final int PAYMENT\_SERVICE\_UNAVAILABLE = 500;

}