|  |  |
| --- | --- |
| 产品名称Product name | 密级Confidentiality level |
| Member | 秘密 |
| 产品版本Product version | Total 11 pages 共11页 |
| V100R001C03 |

Member V100R001C01

系统设计规格书

|  |  |  |  |
| --- | --- | --- | --- |
| Prepared by  拟制 | 郭站伟/g00198733 | Date  日期 | 2015-07-27 |
| Reviewed by  评审人 |  | Date  日期 |  |
| Approved by  批准 |  | Date  日期 |  |
| Authorized by  签发 |  | Date  日期 |  |



Huawei Technologies Co., Ltd.

华为技术有限公司

All rights reserved

版权所有 侵权必究

（PJM03T06 V2.9/ IPD-CMM V3.0 / for internal use only）

（PJM03T06 V2.9/ IPD-CMM V3.0 / 仅供内部使用）

Revision Record 修订记录

| Date  日期 | Revision Version  修订 版本 | CR ID / Defect ID CR号 | Sec No.  修改 章节 | Change Description  修改描述 | Author  作者 |
| --- | --- | --- | --- | --- | --- |
| 2015-07-27 | 1.00 |  |  | initial 初稿完成 | 郭站伟 |
|  |  |  |  |  |  |

Catalog 目 录

[1 业务概述 4](#_Toc425865857)

[1.1 业务目的与背景 4](#_Toc425865858)

[1.2 业务的名称和版本号 4](#_Toc425865859)

[1.3 整体系统架构 4](#_Toc425865860)

[2 系统设计规格 5](#_Toc425865861)

[2.1 AR.FUNC 内容商数据同步和积分订购 5](#_Toc425865862)

[2.1.1 用户需求概述 5](#_Toc425865863)

[2.1.2 设计方案概述 5](#_Toc425865864)

[2.1.3 需求分解与分配 11](#_Toc425865865)

[2.2 AR.FUNC 我的卡券 11](#_Toc425865866)

[2.2.1 用户需求概述 11](#_Toc425865867)

[2.2.2 设计方案概述 12](#_Toc425865868)

[2.2.3 需求分解与分配 12](#_Toc425865869)

[2.3 AR.FUNC 积分乐园商品分享 12](#_Toc425865870)

[2.3.1 用户需求概述 12](#_Toc425865871)

[2.3.2 设计方案概述 12](#_Toc425865872)

[2.3.3 需求分解与分配 13](#_Toc425865873)

[2.4 AR.FUNC 查询会员状态体验优化 13](#_Toc425865874)

[2.4.1 用户需求概述 13](#_Toc425865875)

[2.4.2 设计方案概述 13](#_Toc425865876)

[2.4.3 需求分解与分配 14](#_Toc425865877)

[2.5 AR.FUNC 我的成长、升级规则、帮助FAQ 14](#_Toc425865878)

[2.5.1 用户需求概述 14](#_Toc425865879)

[2.5.2 设计方案概述 14](#_Toc425865880)

[2.5.3 需求分解与分配 15](#_Toc425865881)

[2.6 AR.FUNC 灰度升级和工具验证（优先级降低） 15](#_Toc425865882)

[2.6.1 用户需求概述 15](#_Toc425865883)

[2.6.2 设计方案概述 15](#_Toc425865884)

[2.6.3 需求分解与分配 15](#_Toc425865885)

[2.7 AR.FUNC 积分乐园卡片类型可配置 16](#_Toc425865886)

[2.7.1 用户需求概述 16](#_Toc425865887)

[2.7.2 设计方案概述 16](#_Toc425865888)

[2.7.3 需求分解与分配 16](#_Toc425865889)

[3 业务整体性能目标 16](#_Toc425865890)

[4 系统接口描述 16](#_Toc425865891)

Member V100R001C01XSRV000003

系统设计规格书

# 业务概述

## 业务目的与背景

1、会员服务集成用户经营体系下的各类商品在积分乐园使用积分进行购买（后续会支持现金+积分模式）。积分乐园的定位为“商城”，各个产品将商品列表同步到积分乐园，用户购买时积分乐园与积分平台交互扣除用户积分，积分收入归商品所属的产品，然后积分乐园将用户购买信息同步到对应的产品。例如，主题产品将主题列表同步到会员服务，用户购买主题时积分乐园与积分平台交互扣除用户积分并通知主题记录用户购买订单，用户在主题产品上即可下载已订购的主题。

2、会员服务引入各个卡券提供商，用户以权益的形式领取或者产品主动赠送给用户后，用户在各个提供商的产品上输入对应的卡券后即可享受对应的优惠或者权益。例如，滴滴打车代金券、唯品会代金券等。

3、提供产品分享链接页面用于会员服务客户端的推广，其他需求都是提升用户体验功能。

## 业务的名称和版本号

增量版本开发：Member V100R001C01XSRV000003

## 整体系统架构

Member产品总体技术架构如下：



# 系统设计规格

## AR.FUNC 内容商数据同步和积分订购

### 用户需求概述

会员服务新增商品同步部件，主要用于商品列表和权益卡券列表的数据同步，而权益卡券的领取仍在会员权益部件完成，积分购买操作仍在积分乐园完成。会员服务预先为每个内容商分配栏目标识、鉴权密钥，各个内容提供商通过增量方式将新增、更新、删除的商品列表同步到会员服务。此时会员服务将同步过来的商品挂载到对应的栏目下。用户购买商品时则根据商品所属的内容商信息判断如何扣除用户积分和形成订单方式。

### 设计方案概述

商品同步部件新增对接平台信息表t\_platform(系统启动时加载至缓存，且支持自动刷新)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **字段名** | **类型** | **主键** | **非空** | **默认值** | **说明** |
| 1 | id | int | 是 | 是 |  | 自增序列，auto\_increment |
| 2 | platformid | varchar(32) |  | 是 |  | 内容提供商从开发者联盟申请的接入渠道号即应用标识appid  唯一索引。 |
| 3 | platformname | varchar(128) |  | 是 |  | 平台名称，例如Vmall商城，游戏中心 |
| 4 | secret | varchar(1024) |  | 是 |  | 会员服务分配给各个内容提供商的鉴权命令字，加密存储，用于内容同步接口的鉴权 |
| 5 | inserttime | datetime |  | 是 | now() | 数据新增时间 |

* 为不同的内容商分配不同的鉴权字，使用RSA2048签名验证内容商访问的合法性；若鉴权不通过则返回200007；
* 会员运营人员预先创建所有会使用到的栏目（电影票、彩票），然后将栏目ID和栏目名称列表暂时通过邮件的形式通知各个内容提供商，内容商将自己的产品进行分类然后将商品与对应的栏目同步到会员服务；
* 若商品对应的二级栏目不存在则返回错误码303004；
* 限制每次增量同步的增删改总的数据条数不超过500；
* 增删改数据入库时按照事物处理，要么全部成功要么全部回滚，不存在部分数据入库成功的现象；
* 接口定义时已经区分数据的增删改列表，所以在入库操作时不进行容错处理。新增的商品已存在返回303001，更新的商品不存在返回303002，删除的商品不存在返回303003，在会员服务侧入库失败时需要将失败的商品标识和失败原因通过日志打印完成，便于问题的定位和失败数据的统计；
* 由于内容商同步过来的数据不包含敏感信息，但是在传输过程中必须防篡改，所以接口通道为https。会员服务获取到请求报文时明文方式打印，便于后续的问题定位和数据统计；
* ~~同步接口支持将内容商商品列表全部删除，删除后按照增量的方式进行重新同步；~~

积分乐园部件修改对接平台信息表t\_platform(系统启动时加载至缓存，且支持自动刷新)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **字段名** | **类型** | **主键** | **非空** | **默认值** | **说明** |
| 1 | id | int | 是 | 是 |  | 自增序列，auto\_increment |
| 2 | platformid | varchar(32) |  | 是 |  | 内容提供商从开发者联盟申请的接入渠道号即应用标识appid  唯一索引。 |
| 3 | projectid | varchar(16) |  | 是 |  | 产品项目id，与积分平台交互时使用 |
| 4 | platformname | varchar(128) |  | 是 |  | 平台名称，例如Vmall商城，游戏中心 |
| 5 | ismember | tinyint |  | 是 | 0 | 是否为会员自身配置标识，非会员产品则需要调用第三方接口进行购买通知。  0：非会员  1：会员 |
| 6 | notifyurl | varchar(256) |  |  |  | 内容提供商的订购通知地址，用于用户订购时通知第三方接口 |
| 7 | inserttime | datetime |  | 是 | now() | 数据新增时间 |

platformid由int修改为varchar(32)；

新增notifyurl、ismember、projectid字段；

修改商品详情信息表t\_productinfo(系统启动时加载至内存，且支持定时刷新)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **字段名** | **类型** | **主键** | **非空** | **默认值** | **说明** |
| 1 | id | bigint | 是 | 是 |  | 自增序列，auto\_increment |
| 2 | productid | varchar(32) |  | 是 |  | 商品id，  参见唯一标识码生成规则章节，唯一索引 |
| 3 | outerid | varchar(32) |  | 是 |  | 内容商侧商品ID，若管理员通过Portal录入则默认与productid值相同 |
| 4 | productname | varchar(100) |  | 是 |  | 商品名称，索引 |
| 5 | status | tinyint |  | 是 | 1 | 商品状态  1：待上线  2：已上线  3：已下线 |
| 6 | showorder | bigint |  | 是 |  | 返回类别数据时根据该字段进行控制顺序。  使用数据库当前时间的数字做为默认值，例如20150619121212。排序时按照该字段降序。使用时间格式展示到页面上共运营人员编辑 |
| 7 | logourl | varchar(256) |  | 是 |  | 缩略图片地址 |
| 8 | bannerurl | varchar(256) |  | 是 |  | 主图片地址 |
| 9 | description | varchar(256) |  |  |  | 商品描述 |
| 10 | detailid | text |  | 是 |  | 点击商品图片跳转的商品详情地址，  参考《华为云平台会员服务接口文档》 |
| 11 | priceunit | varchar(10) |  | 是 | ‘￥’ | 价格标识符 |
| 12 | oriprice | decimal(12, 2) |  |  |  | 商品原始价格，单位为元 |
| 13 | oripoints | int |  |  |  | 商品原始的积分 |
| 14 | points | int |  | 是 |  | 购买需要支付的积分单价 |
| 15 | detailimg | varchar(2048) |  |  |  | 商品详情图，多个图片地址的拼接，例如’ <http://img1.baidu.com>,<http://img2.baidu.com>,<http://img3.baidu.com> |
| 16 | platformid | varchar(32) |  | 是 |  | 商品来源平台的标识，例如Vmall商城，游戏中心。  参照t\_platform |
| 17 | stocknum | bigint |  | 是 | 0 | 库存剩余数量，默认为0。  虚拟类型商品不需要对库存做判断 |
| 18 | inserttime | datetime |  | 是 | now() | 数据新增时间 |
| 19 | onlinetime | datetime |  |  |  | 商品上线时间 |
| 20 | offlinetime | datetime |  |  |  | 商品下架时间 |
| 21 | categid | int |  | 是 |  | 栏目ID |
| 22 | parentid | int |  | 是 |  | 父栏目ID |
| 23 | buylimit | int |  | 是 | 0 | 该商品限制购买数量  0：不限制  正数为用户可在客户端上购买的最大数量。 |
| 24 | producttype | tinyint |  | 是 | 1 | 1：实物  2：虚拟物品 |

platformid由int修改为varchar(32)；

detailimg长度调整为2048；

oriprice、oripoints字段修改为可空；

stocknum默认值修改为0，商品类型为虚拟商品时不需要对库存数量进行判断；

将subpoints、subprice、stockunit、price字段删除；

新增outerid来记录内容提供商侧商品ID；

新增bannerurl字段录入商品详情中的通栏地址，在获取商品详情时做为icon字段值返回；

增加outerid、platformid联合唯一索引；

* 表字段来源：
* id，自增长生成；
* productid，根据已有规则生成；
* outerid，内容提供商传入，若管理员通过Portal新增商品时默认与productid一致，若管理员手动录入内容商的商品时输入了outerid则使用录入数据否则默认与productid一致。
* productname，内容提供商传入或者管理员录入；
* status，若为内容提供商新增或者更新数据则默认为上线。管理员录入则保持现有能力；
* showorder，若为内容提供商数据同步则默认为数据库当前时间，管理员录入则使用录入时间，录入为空则默认为数据库当前时间；
* logourl，内容提供商传入或者管理员录入；
* bannerurl，内容提供商传入或者管理员录入；
* description，内容提供商传入或者管理员录入；
* detailid，根据已有规则拼接商品productID生成；
* priceunit, 内容提供商传入或者管理员录入,默认为￥；
* oriprice，内容提供商传入或者管理员录入,默认为0；
* oripoints，内容提供商传入或者管理员录入,默认为0；
* points，内容提供商传入或者管理员录入,默认为0；
* detailimg，内容提供商传入或者管理员录入；
* platformid，内容提供商传入的渠道编码或者管理员录入；
* stocknum，内容商数据同步时默认为0，或者管理员录入；
* categid，内容商接口传入(已校验是否存在)，或者管理员录入；
* parentid,根据categid到t\_categoryinfo中获取的（已加载至内存，可以使用内存中缓存的数据）；
* buylimit，内容提供商传入或者管理员录入，默认为0不限制；
* producttype，内容提供商同步数据时直接默认为2虚拟类型，或者管理员录入；
* 推荐内容中的商品可能被下线，由运营人员手动在管理Portal上修改暂不考虑自动替换下线的推荐内容；
* 获取商品详情接口中的图片地址为bannerurl。

修改用户订单信息表t\_userorder（按照userid进行分库）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 主键 | 非空 | 默认值 | 说明 |
| 1 | id | bigint | 是 | 是 |  | 自增序列号，auto\_increment |
| 2 | userid | varchar(32) |  | 是 |  | 华为内部用户标识，索引 |
| 3 | orderid | varhar(32) |  | 是 |  | 订单唯一标识，  参见唯一标识码生成规则章节 |
| 4 | productid | varhar(32) |  | 是 |  | 商品id，参见t\_productinfo |
| 5 | outerid | varchar(32) |  | 是 |  | 内容商侧商品ID，若管理员通过Portal录入则默认与productid值相同 |
| 6 | ordertime | datetime |  | 是 | now() | 下单时间 |
| 7 | status | tinyint |  | 是 |  | 1. 待付款 2. 待发货 3. 待收货 4. 已完成 5. 待退货 6. 已退货 |
| 10 | platformid | varchar(32) |  | 是 |  | 商品来源平台的标识，例如Vmall商城，游戏中心。  参照t\_platform |
| 11 | username | varchar(256) |  |  |  | 收货人姓名，加密  虚拟物品时可为空 |
| 12 | phonenumber | varchar(256) |  |  |  | 收货人电话号码，加密  虚拟物品时可为空 |
| 13 | province | varchar(64) |  |  |  | 所在省份  虚拟物品时可为空 |
| 14 | city | varchar(64) |  |  |  | 所在城市  虚拟物品时可为空 |
| 15 | area | varchar(64) |  |  |  | 所在区域  虚拟物品时可为空 |
| 16 | avenue | text |  |  |  | 所在街道地址，加密  虚拟物品时可为空 |
| 17 | postcode | varchar(64) |  |  |  | 邮编，不为空时加密 |
| 18 | productname | varchar(100) |  | 是 |  | 商品名称 |
| 19 | amount | bigint |  | 是 |  | 购买商品数量 |
| 20 | logourl | varchar(256) |  | 是 |  | 商品主图片 |
| 21 | description | varchar(256) |  |  |  | 商品描述 |
| 22 | priceunit | varchar(10) |  | 是 |  | 价格单位 |
| 23 | oriprice | decimal(12, 2) |  | 是 |  | 商品原始价格，单位为元 |
| 24 | oripoints | int |  | 是 |  | 商品原始的积分 |
| 25 | points | int |  | 是 |  | 购买时需要支付的积分单价 |
| 26 | updatetime | datetime |  | 是 | now() | 数据更新时间 |
| 27 | transport | text |  |  |  | 物流信息，目前有运营人员填写。后续对接物流平台后会自动回写。 |
| 28 | producttype | tinyint |  | 是 |  | 1：实物  2：虚拟物品 |

将subpoints、subprice、paidmoney、paidpoints、price、buyresult、buydesc、paidway字段删除；

新增platformid字段；

* 用户购买时内存中判断商品所属的platformid为会员服务产品还是第三方内容提供商，
* 若ismember字段配置为1则此商品按照属于会员服务自己的商品处理。会员服务直接与积分平台进行交互，并根据扣除结果记录对应订单。
* 若ismember字段配置为0则此商品按照第三方内容提供商的商品处理，积分乐园与积分平台交互扣除用户积分，然后通知内容商用户购买商品的信息userid、outerid、amount等，内容商写订单成功后会员服务本地记录订单。
* 用户购买商品时需要使用Mysql中的商品数据的商品状态、库存等条件判断是否可以购买，若商品不存在和商品已下线则返回资源不存在错误码、库存为0时返回已售罄错误码、用户购买数量满足最大量限制但是库存不足时返回库存不足错误码。
* 由于积分乐园先到积分平台扣除用户积分，然后通知内容商写订单，若积分扣除成功但是订单写失败时返回“系统忙，被扣除的积分会尽快退回到您的账户”错误码，并记录这种错误数据到t\_cancelpoints，暂时使用人工方式给用户回退积分。人工处理时数据状态有运营来维护，后期通过定时任务进行积分回退，由定时任务按照数据处理结果进行维护。
* 与对接的平台交互进行购买通知时，系统间交互的超时时间、字符集等使用统一的配置在t\_accessconfig中新增，包括签名使用的私钥（工作密钥加密存储）。
* 与积分平台交互时传入的商品ID为t\_productinfo.outerid，这样可以满足积分平台和业务之间核对账单而不需要经过会员服务。

新增积分回退表t\_cancelpoints（按照userid进行分库）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 主键 | 非空 | 默认值 | 说明 |
| 1 | id | bigint | 是 | 是 |  | 自增序列号，auto\_increment |
| 2 | userid | varchar(32) |  | 是 |  | 华为内部用户标识 |
| 3 | orderid | varhar(32) |  | 是 |  | 订单唯一标识，  参见唯一标识码生成规则章节 |
| 4 | transactionid | varhar(64) |  | 是 |  | 与UP交互扣除用户积分时的交易流水号 |
| 5 | productid | varhar(32) |  | 是 |  | 商品id，参见t\_productinfo |
|  | productname |  |  |  |  |  |
| 6 | platformid | varchar(32) |  | 是 |  | 商品来源平台的标识，例如Vmall商城，游戏中心。  参照t\_platform |
| 7 | platformname | varchar(128) |  | 是 |  | 平台名称，例如Vmall商城，游戏中心 |
| 8 | status | tinyint |  | 是 | 1 | 数据状态，  1：待处理状态  2：处理中  3：处理完成 |
| 9 | inserttime | datetime |  | 是 | now() | 数据插入时间 |
| 10 | finishedtime | datetime |  |  |  | 积分回退完成时间 |
| 11 | paidpoints | int |  | 是 |  | 支付的积分 |

* 表t\_userorder字段来源，
* id，自增长生成；
* userid,客户端参数传入；
* orderid，根据唯一标识码生成规则生成；
* productid，客户端传入参数；
* outerid，t\_productinfo.outerid字段值；
* ordertime，数据库系统当前时间；
* status，当前只支持纯积分消费，消费完成则订单完成。
* platformid，t\_productinfo.platformid字段值；
* username，客户端接口参数；
* phonenumber，客户端接口参数；
* province，客户端接口参数；
* city，客户端接口参数；
* area，客户端接口参数；
* avenue，客户端接口参数；
* postcode，客户端接口参数；
* productname，t\_productinfo.productname字段值；
* amount，客户端接口参数；
* logourl，t\_productinfo.logourl字段值；
* description，t\_productinfo.description字段值；
* priceunit，t\_productinfo.priceunit字段值；
* oriprice，t\_productinfo.oriprice字段值；
* oripoints，t\_productinfo.oripoints字段值；
* points，t\_productinfo.points字段值；
* updatetime，数据库系统当前时间；
* transport，暂时直接为空；
* producttype，t\_productinfo.producttype字段值；

用户购买内容商虚拟商品流程如下：



方案设计为先扣除用户积分然后通知内容商写订单，主要考虑到用户积分不足时客户端仍会请求服务器，所以先判断积分会减少很多无谓的订单回退。

若与积分平台和内容商全部交互成功但是本地记录订单失败时，需要打印告警日志，便于后续的原因分析定位，注意过滤敏感信息。而不需要将已经支付的积分和内容商已经记录的订单回滚掉，即使会员服务本地没有记录订单数据也不会影响用户在对应的内容商购买商品或者权益的使用。

内容商和会员商品的购买流程区别主要为是否需要与内容商交互通知订单生成。其余的业务逻辑完全一致。

注意：业务不存在免费的商品，所有商品的价格必须大于0。不管是资源同步还是管理员通过Portal维护，都必须保证积分大于0。用户在订购时不管积分是否为0（不能小于0）都按照正常流程与积分平台、内容商交互。

### 需求分解与分配

1. 新增部件支持外部资源同步，支持访问数据库、https与外部交互；
2. 新增部件增加商品列表同步接口；
3. 积分乐园新增订购通知第三方接口；
4. 积分乐园订购流程优化；
5. 积分管理Portal商品管理页面优化；

## AR.FUNC 卡券查询和领取功能

### 用户需求概述

会员权益部件增加“我的卡券”功能，用户通过会员客户端可以查询已登录华为账号下的卡券列表信息，若卡券已过期时间超过配置时间段后则不再返回客户端。产品规划有以下三种方式领取或者发放：1、用户主动领取运营活动赠送的卡券；2、按照运营过滤条件给指定的用户发放卡券；3、服务部根据需要给指定的某个用户发放卡券。

### 设计方案概述

新增卡券提供者信息表t\_couponprovider(加载至内存，且支持自动更新，数据由管理员在Portal上产生)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 主键 | 非空 | 默认值 | 说明 |
| 1 | id | int | 是 | 是 |  | 自增序列号，auto\_increment |
| 2 | providerid | varchar(32) |  | 是 |  | 卡券提供者标识，  由于需要使用该标识统计相关的卡券数据，所以即使数据重新生成也不能发生变化。  使用提供者在开发者联盟上申请的接入渠道即应用标识appid。  唯一索引 |
| 3 | providername | varchar(32) |  | 是 |  | 卡券提供者名称 |
| 4 | description | varhar(512) |  |  |  | 卡券提供者描述 |
| 5 | inserttime | datetime |  | 是 | now() | 数据插入时间 |

providerid唯一索引。

新增卡券信息表t\_couponlist(加载至内存，且支持自动更新，数据由管理员在Portal上产生)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 主键 | 非空 | 默认值 | 说明 |
| 1 | id | int | 是 | 是 |  | 自增序列号，auto\_increment |
| 2 | couponid | varchar(32) |  | 是 |  | 卡券标识，  按照唯一标识码生成规则生成。与providerid组合唯一索引 |
| 3 | providerid | varchar(32) |  | 是 |  | 卡券提供者标识，  参考t\_couponprovider |
| 4 | title | varhar(128) |  | 是 |  | 卡券标题 |
| 5 | logourl | varchar(256) |  | 是 |  | 卡券图标地址 |
| 6 | bannerurl | varchar(256) |  | 是 |  | 卡券详情的通栏图标地址 |
| 7 | starttime | datetime |  | 是 |  | 卡券生效时间 |
| 8 | endtime | datetime |  | 是 |  | 卡券失效时间 |
| 9 | eachcost | int |  |  |  | 每张卡券成本，由运营人员维护和使用，其他业务暂时未使用到 |
| 10 | totalnum | int |  | 是 |  | 卡券总数 |
| 11 | detailid | text |  | 是 |  | 点击卡券图片后的跳转样式，  参考《华为云平台会员服务接口文档》 |
| 12 | type | tinyint |  | 是 | 1 | 领取卡券方式，  1：本地卡券号  2：调用第三方接口  当前只有本地卡券号的类型 |
| 13 | description | text |  | 是 |  | 卡券描述 |
| 14 | inserttime | datetime |  | 是 | now() | 数据插入时间 |

couponid唯一索引。

新增卡券券码信息表t\_couponcode(定时任务加载至Redis，Redis中获取卡券号)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 主键 | 非空 | 默认值 | 说明 |
| 1 | id | int | 是 | 是 |  | 自增序列号，auto\_increment |
| 2 | couponid | varchar(32) |  | 是 |  | 卡券标识，  按照唯一标识码生成规则生成。与ticketcode组合唯一索引 |
| 3 | ticketcode | varchar(64) |  | 是 |  | 卡券串码 |
| 4 | status | tinyint |  | 是 | 1 | 卡券状态  1：待领取  2：已领取 |
| 5 | loadstatus | tinyint |  | 是 | 1 | 卡券码加载到Reids中的处理状态  1：待加载；  2：加载中；  3：加载成功； |
| 6 | threadflag | varchar(64) |  |  |  | 线程标识，防止重复获取数据处理 |
| 7 | inserttime | datetime |  | 是 | now() | 数据插入时间 |

couponid与ticketcode组合唯一索引。

新增活动信息列表t\_activitylist(加载至内存，且支持自动更新，数据由管理员在Portal上产生)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 主键 | 非空 | 默认值 | 说明 |
| 1 | id | int | 是 | 是 |  | 自增序列号，auto\_increment |
| 2 | activityid | varchar(32) |  | 是 |  | 活动标识，  按照唯一标识码生成规则生成。唯一索引 |
| 3 | couponid | varchar(32) |  | 是 |  | 卡券标识 |
| 4 | activityprovider | varchar(32) |  | 是 |  | 活动来源，即活动赞助者，仅是活动详情使用 |
| 5 | title | varhar(128) |  | 是 |  | 活动标题 |
| 6 | logourl | varchar(256) |  | 是 |  | 活动图标地址 |
| 7 | labelurl | varchar(256) |  |  |  | 标签图片地址 |
| 8 | bannerurl | varchar(256) |  | 是 |  | 活动详情的通栏图标地址 |
| 9 | starttime | datetime |  | 是 |  | 活动开始时间 |
| 10 | endtime | datetime |  | 是 |  | 活动失效时间 |
| 11 | detailid | text |  | 是 |  | 点击活动图片后的跳转样式，  参考《华为云平台会员服务接口文档》 |
| 12 | description | text |  | 是 |  | 活动描述 |
| 13 | inserttime | datetime |  | 是 | now() | 数据插入时间 |
| 14 | categid | int |  | 是 |  | 活动所属栏目标识，  参见t\_categoryinfo |
| 15 | showorder | bigint |  | 是 |  | 控制数据返回顺序，  使用数据库当前时间的数字做为默认值，例如20150619121212。排序时按照该字段降序。使用时间格式展示到页面上共运营人员编辑 |
| 16 | footnote | varchar(64) |  |  |  | 注脚：例如： “领取人数： 超过30w”或者“活动开始时间：2015-08-05，活动结束时间2015-08-06” |
| 17 | tips | varchar(64) |  | 是 |  | 用户无法参与该活动时的提示语，即领取额接口返回304005时对应的错误描述 |
| 18 | type | tinyint |  | 是 |  | 活动类型，  1：卡券领取活动；  2：线下活动；  3：直接跳转到外部参与活动；  需要根据不同的type拼接成不同的detailid，用户点击时访问不同的接口和传递不同的参数。  跳转到外部参与活动的detailid需要直接通过初始化数据预置或者管理员Portal录入 |
| 19 | status | tinyint |  | 是 | 0 | 活动上下线状态  0：下线；  1：上线； |

新增权益栏目列表t\_categoryinfo(加载至内存，且支持自动更新，数据由管理员在Portal上产生)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 主键 | 非空 | 默认值 | 说明 |
| 1 | id | int | 是 | 是 |  | 自增序列，auto\_increment |
| 2 | categid | int |  | 是 |  | 栏目唯一标识，  从300001开始增加，管理员操作时获取最大的id，然后加1。  唯一索引。 |
| 3 | categname | varchar(100) |  | 是 |  | 栏目名称，  例如游戏权益、生活权益 |
| 4 | status | tinyint |  | 是 | 0 | 在线状态  0：下线状态  1：在线状态 |
| 5 | showorder | bigint |  | 是 |  | 返回类别数据时根据该字段进行控制顺序。  使用数据库当前时间的数字做为默认值，例如20150619121212。排序时按照该字段降序。使用时间格式展示到页面上共运营人员编辑 |
| 6 | description | varchar(256) |  |  |  | 栏目描述 |
| 7 | detailid | text |  | 是 |  | 点击栏目图片跳转的地址，  参考《华为云平台会员服务接口文档》 |
| 8 | inserttime | datetime |  | 是 | now() | 数据新增时间 |

新增用户卡券信息表t\_usercoupon（用户查询时，实时到数据库获取，根据userid进行分库）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 主键 | 非空 | 默认值 | 说明 |
| 1 | id | bigint | 是 | 是 |  | 自增序列号，auto\_increment |
| 2 | userid | varchar(32) |  | 是 |  | 华为内部用户标识，  与activityid组合唯一索引  每个用户最多参见一次同一个活动 |
| 3 | activityid | varchar(32) |  | 是 |  | 活动标识，  与userid组合唯一索引 |
| 4 | couponid | varchar(32) |  | 是 |  | 卡券标识 |
| 5 | ticketcode | varhar(64) |  | 是 |  | 卡券码 |
| 6 | inserttime | datetime |  | 是 | now() | 数据插入时间 |

使用userid、activityid做为组合唯一索引。

管理员导入卡券，

* 卡券提供商需在开发者联盟上申请接入渠道编码即应用标识appid；
* 会员服务运营人员通过Portal上录入卡券提供商信息到t\_couponprovider；
* 会员服务运营人员使用Portal可以通过excel方式批量导入卡券码，运营人员维护卡券的基本信息然后选择卡券提供者按照新增处理，也可以先选择卡券然后追加处理，追加的卡券的所有属性必须一致。导入卡券码时首先校验文件的格式要求必须是xlsx不需要支持多种excel格式解析、文件大小（配置项page\_file\_size，默认10M）、数据条数（配置项page\_file\_rows，默认50000条），满足条件时获取第一个sheet中的第一列的值做为卡券值,只需要处理第一列数据即可。例如，
* 卡券的基本信息记录到t\_couponlist，
* id, 自增序列号，auto\_increment；
* couponid，由代码按照唯一标识码生成规则生成；
* providerid、providername由运营人员通过页面选择t\_couponprovider中的提供者信息；
* title、logourl、bannerurl、starttime、endtime、eachcost、description由运营人员手动录入；
* totalnum，导入的数据量累加追加的卡券数量，导入后需要提示运营人员本次导入的数据量；
* detailid，由代码按照规则生成,将接口的全名(例如，<http://domain/member/profile/getCouponActivityDetail>)拼接到intent中；
* inserttime，数据库系统默认时间；
* 卡券号信息记录到t\_couponinfo，
* id, 自增序列号，auto\_increment；
* couponid，与记录到t\_couponlist中的值一致；
* ticketcode，excel中的券码值；
* status，默认为待领取，用户领取成功后置为已领取；
* inserttime，数据库系统时间；

运营人员创建栏目，

* 会员权益首页布局结构为“权益的四叶草”、“banner广告列表”、“活动栏目列表”，布局排序按照已有配置项root\_card\_order控制。
* “权益的四叶草”，根据用户等级到t\_memberrights、t\_devicerights、t\_rightsinfo中获取与已有流程一致不需要适配。
* “banner广告列表”，根据用户等级到t\_memberad、t\_devicead、t\_adinfo中获取与已有流程一致不需要适配。
* “活动栏目列表”,按照生活类，游戏类，购物类，XX机型专属, 线下活动为维度进行划分，通过栏目上下线字段status控制返回数据是否返回，通过排序字段showorder控制数据返回顺序，通过跳转样式detailid字段控制用户点击后跳转到下一级的参数，数据列表获取该栏目下在线的活动，使用activity\_list\_size控制每页数据量默认15条。新增配置项default\_column\_size默认为3，控制所有活动列表栏目下推荐位内容个数。在加载、组装首页数据时，推荐位的内容采用降序排列方式获取配置项default\_column\_size限制个数。

t\_categoryinfo表字段来源，

* id, 自增序列号，auto\_increment；
* categid，初始化数据时从300001开始，管理员操作时获取表中最大的id，然后加1。
* categname，运营人员手动录入；
* status，运营人员创建的栏目默认下线，必须执行上线操作后才能是上线状态；
* showorder，默认为数据库当前时间，当时状态正常的栏目详情中，该字段使用时间格式展示到页面上供运营人员编辑，将运营人员选择的时间转换成数字记录到数据库，若运营人员将其清空则使用数据库当前时间；
* description，由运营人员手动录入；
* detailid，由代码按照规则生成，将栏目ID拼接到intent中；
* inserttime，数据库系统当前时间；

运营人员创建活动，当前活动分为直接跳转到外部参与活动或者展示活动详情两种形式。跳转到外部参与的活动，用户领取卡券的相关数据会员服务获取不到。若为获取活动详情需要判断活动的有效性若活动未开始则返回304002且返回信息中提示用户据活动开始的剩余时间，活动已结束则返回304003，t\_activitylist字段来源

* id，自增序列号，auto\_increment；
* activityid，按照唯一码生成规则生成；
* couponid，运营人员选择已创建的卡券标识；
* categid，运营人员选择已创建的栏目标识；
* activityname、title、starttime、endtime、description、footnote、tips，运营人员手动输入；
* type，根据数据表中的取值，直接使用静态资源初始化，不同的type生成detailid的参数值不一致，所以只要新增活动类型Portal一定需要升级了；
* logourl、bannerurl，运营人员上传图片，注意参照积分平台控制上传图片目录和图片的权限；
* detailid，根据代码规则生成，将接口的全名(例如，<http://domain/member/profile/getCouponActivityDetail>)和活动类型拼接到intent中；
* showorder、inserttime数据库系统当前时间；

领取卡券功能，当前用户领取的卡券都在会员服务本地的数据库中。

* 判断活动有效期；
* 判断卡券是否存在，不存在时返回卡券已领完；
* 判断卡券有效期，若卡券有效期已过则返回无剩余卡券可领304001，在管理Portal上创建活动时必须校验关联卡券的有效期保证活动期内的卡券用户领取后都是有效的；
* 判断用户是否已经成功参与过该活动，已成功参加则返回304004错误码；
* 使用t\_couponcode.couponid = t\_couponlist.couponid and t\_couponcode.status = 1 limit 1的形式获取卡券码；
* 获取到卡券码记录到t\_usercoupon中，获取不到则返回304001提示用户已领取完。
* 若成功获取到卡券号，但是写数据库失败时需打印告警日志并返回用户系统忙。为减少同一个用户多次参加同一个活动领取卡券时访问数据库的次数，用户领取卡券成功时将用户和活动关系记录到Redis，有效期建议10分钟支持可配置。记录Redis的结果不影响业务流程。Reids中获取不到领取记录时查询数据库。
* 当前t\_couponcode和t\_usercoupon中的数据没有清理机制，后续t\_couponcode中已领取的数据通过Portal由运营控制是否删除。t\_usercoupon中用户的卡券数据可以通过配置项控制过期多久的数据转移到历史表或者直接删除。

用户过期卡券清理定时任务，

* 用户领取卡券的有效期超过N天后，接口上不再返回。新增配置项my\_rights\_expire默认为3天,单位为天控制卡券超过有效期超过最大

我的卡券查询接口，

* 新增配置项my\_rights\_size控制接口每页数据返回条数，默认15条；
* 根据用户领取时间降序排列，表t\_usercoupon中存在的数据进行分页返回；
* 用户查询卡券列表时根据userid实时根据查询条件实时到Mysql数据库获取，数据根据inserttime降序返回。
* 若获取页已无数据返回则返回成功码0、hasNextPage为0、couponList为空列表；

我的权益中展示卡券数量，

* 修改t\_rightsinfo中类型为我的权益的卡券包配置数据的footNote为“您有0张卡券”；
* 仍然采用内存方式缓存“我的权益”中的配置；
* 根据userID实时访问数据库查询用户的卡券数量，表t\_usercoupon中有多少数据就展示有多少张卡券；
* 若userID下无卡券包则使用默认的缓存数据，若userID下的卡券数量超过99张时，显示为99+，否则显示实际的数字；

### 需求分解与分配

1. 新增用户卡券查询接口getCouponList
2. 我的权益的卡券包展示卡券数量
3. 新增获取活动列表接口
4. 新增获取活动详情接口
5. 新增卡券领取接口
6. 修改会员权益首页数据接口

## AR.FUNC 积分乐园商品分享(优先级为中)

### 用户需求概述

当前客户端在微信上分享积分乐园中的商品时，会员服务的下载地址会被屏蔽掉，用户无法点击分享的连接下载会员服务客户端。产品规划服务器新增WAP页面提供分享商品详情展示和提供会员服务APK下载地址的入口。

### 设计方案概述

* 建议WAP页面直接展示静态的宣传页面，而不是根据商品ID到积分乐园服务器获取商品详情展示；
* 增加会员服务APK下载地址链接，用户点击下载时将配置的APK下载到XXXX目录，系统会自动安装？
* 高保真参见配置库地址，https://szxsvn04-ex:3690/svn/TC\_EMUI\_DOC\_SVN/06.用户经营/HwPhoneService/01.Projects/会员服务/02.高保真/2.1积分和消费/商品详情页.jpg

### 需求分解与分配

1、新增WAP页面提供积分乐园宣传和下载会员服务APK入口。

## AR.FUNC 查询会员状态体验优化

### 用户需求概述

会员权益部件，体验现象：使用新的华为账号在已被其他人激活过会员的金银卡手机上登陆会员服务时，会提醒用户激活成为金银卡会员，但是用户实际激活为普通会员，提示与实际体验不一致。方案优化为：新的华为账号在已被其他人激活过会员的金银卡手机上登陆会员服务时提醒用户激活为普卡会员。

### 设计方案概述

* 若根据userID查询到用户已经和手机设备绑定过则按照用户已是会员的逻辑处理，与之前的流程一致不需要调整；
* 若根据userID查询用户未与任何设备绑定过，
* 若根据客户端参数deviceLevel判断，登录设备不是金银卡，
* 若客户端brand参数包含Huawei或者honor时则按照brand配置的等级提示用户可激活等级否则返回非华为手机错误码禁止激活会员。
* 若根据客户端参数deviceLevel判断，登录设备是金银卡，则使用客户端传入的IMEI访问UP获取用户登录的设备与用户的绑定关系，
* 若访问异常或者返回错误码则提示用户可激活为deviceLevel转换的会员等级；
* 若设备与该用户绑定，则用户已是会员，根据UP返回的rightsID返回云和售后权益列表；
* 若设备已与用户绑定，则提示用户激活为普卡会员；
* 若设备未与任何用户绑定，则提示用户可激活为UP返回的deviceLevel转换的会员等级；



### 需求分解与分配

1. 优化会员查询会员状态业务流程

## AR.FUNC 我的成长、升级规则、帮助FAQ

### 用户需求概述

产品静态介绍页面配置在服务器侧，可以动态灵活的调整内容而不需要客户端版本配合升级，用户体验较之前版本预置在客户端有很大提升。

### 设计方案概述

* UI设计人员提供“我的成长”、“升级规则”、“帮助FAQ”三个静态页面资源的高保真，服务器侧将其挂载到Nginx的对应目录，并生成对应的访问地址；
* 静态资源的地址通过配置刷新接口getConf下发给客户端，在原有响应报文data节点基础上增加"help":"www.membership.hicloud.com/static/help.html", "myGrowth":"www.membership.hicloud.com/static/myGrowth.html",两个个配置数据；

### 需求分解与分配

1、配置静态资源并将地址初始化到配置刷新报文体中；

## AR.FUNC 灰度升级和工具验证（优先级为低）

### 用户需求概述

1. Nginx白名单限制访问接口
2. Nginx脚本支持灰度升级
3. 验证Router访问不了MQ时将Router直接杀死的工具
4. 验证运维通过工具清除积压缓存工具

### 设计方案概述

Nginx上使用文件配置接口访问白名单，防止恶意用户访问业务服务器没有提供的能力，造成MQ中产生非法的队列。每次业务服务增加新的接口时需要更新配置文件，否则将导致接口调用不通。参考开放网关的配置脚本输出会员服务的白名单脚本。

参考开放网关灰度升级配置的lua脚本，输出会员服务灰度升级脚本。



受限于开放网关GAF框架机制，Router集群中的一个Tomcat只能访问一个Redis进行消息缓存，若MQ组件的Redis挂掉而负载均衡Nginx侧继续分配请求给对应的Router会导致接收请求后无法缓存到消息队列，业务不可用。所以当Tomcat无法与MQ正常交互时需要将Tomcat进程kill掉，这样Nginx就会将请求分配到其他的Tomcat机器，保证业务继续可用。

短时间内请求过多而业务服务服务器无法及时处理时，会导致资源被大量占用。若不及时清除会导致MQ内存被沾满导致新的请求无法缓存最终体现业务不可用。若出现异常情况导致请求积压过多时需要能够使用工具及时清理。验证开放网关提供的删除队列脚本工具是否可用。

### 需求分解与分配

1. Nginx白名单限制访问接口
2. Nginx脚本支持灰度升级
3. 验证Router访问不了MQ时将Router直接杀死的工具
4. 验证通过工具清除积压缓存工具

## AR.FUNC 产品体验配置优化

### 用户需求概述

新增配置项分别支持栏目类别、广告列表、推荐内容、商品列表卡片类型；

售后权益跟设备走，手机售后权益为普卡时跳转到服务部官网地址；

即将上线页面美观度不满足产品需要，将其调整为客户端本地页面；

我的权益中，将“云空间礼包”更名为“云空间”；

订单详情中增加商品ID

### 设计方案概述

* 积分乐园新增配置项分别支持栏目类别root\_categ\_layout、各个页面的广告列表ad\_list\_layout、推荐内容root\_column\_layout、商品列表product\_list\_layout卡片类型即layoutid字段可配置，每种数据集的layoutid是相同的。例如，广告列表的layoutid要么是2要么是15而不能配置为部分2部分15。
* 会员权益部件广告列表的layoutid修改为使用配置项控制，将t\_memberad、t\_devicead中的layoutid字段删除 。同样广告列表中layoutid只能支持一个值。
* 售后权益跟设备走，查询会员状态时若用户已是会员且售后等级为普卡则detailid中返回服务部官网地址。修改售后普卡权益的detailid值。
* 目前服务器有两个“即将上线”的WAP页面A和B，只保留售后权益功能使用的A页面，将B页面相关的代码、静态资源全部删除，且将数据库中预置的与B页面地址相关的数据修改为客户端本地“即将上线”的地址。已经和产品经理确认保留的A页面都要是为了容错处理，产品已经开始规划会员有效期的概念，理论上会存在查询会员状态时为金卡，用户点击售后权益时会员有效期过期变成了普卡，此时则使用A展示即将上线。
* 修改t\_rightsinfo中类型为我的权益的“云空间礼包”名称修改为“云空间”；

### 需求分解与分配

1. 积分乐园卡片类型可配置；
2. 会员权益广告列表卡片类型可配置；
3. 修改普卡售后权益detailid地址；
4. 即将上线页面删除和地址修改；
5. 修改t\_rightsinfo中类型为我的权益的“云空间礼包”名称修改为“云空间”；

## AR.FUNC Address部件返回区域ID(方案待确认)

### 用户需求概述

O2O为实现快递运费计算，希望提供收货地址区域的id；

### 设计方案概述

* 积
* ；

### 需求分解与分配

## AR.FUNC 统计打点

### 用户需求概述

Router部件，会员服务记录用户的请求数据，然后推送给BI，有BI统计用户的活跃度。

### 设计方案概述

* Router部件增加接口请求和响应日志，用于将所有进入到会员系统的用户请求访问记录下来。
* 请求内容格式为time|requestid|imei(AES/CBC/安全随机IV)| uid|interfaceName |channel|cVer|ip
* 响应内容格式为time|requestid|resultcode|reason
* 所有接口请求和响应的日志输出到单独的interface日志文件中

### 需求分解与分配

1. 完善Router部件接口访问日志记录。

## AR.FUNC pushToken信息上报接口

### 用户需求概述

会员权益部件新增pushToken消息上报接口

### 设计方案概述

* 会员权益部件（profile）新增接口支持pushToken消息上报，业务逻辑与之前营销部件一致。

### 需求分解与分配

新增pushToken上报接口；

## AR.FUNC 批量获取数据库序列号

### 用户需求概述

商品列表中展示的商品顺序可以由管理员通过管理Portal进行上移和下移调整，排序的方式是通过交换对应数据的showOrder字段值实现的。在内容批量同步商品数据到会员服务时showOrder字段取值为数据库系统当前时间，所以会有很多数据的showOrder字段值相同，管理员人员在调整上移和下移时就无法达到排序的能力了。将showOrder字段取值调整为类似Oracle数据库sequence方式来实现showOrder字段值的唯一性。

### 设计方案概述

新增序列表t\_sequence（业务使用时访问数据库进行操作）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **字段名** | **类型** | **主键** | **非空** | **默认值** | **说明** |
| 1 | id | int | 是 | 是 |  | 自增序列，auto\_increment |
| 2 | name | varchar(32) |  | 是 |  | 序列名称  唯一索引。 |
| 3 | value | bigint |  | 是 | 0 | 序列当前值 |
| 4 | increment | int |  | 是 | 1000 | 序列增量变化值 |
| 5 | inserttime | datetime |  | 是 | now() | 数据新增时间 |

新增序列名称为唯一索引。

* 业务需要使用序列值且内存中不存在时，访问数据库获取批量个（increment）序列值；
* 开启事物；
* 根据序列名称锁定数据表中对应数据，并获取序列当前值和变化值；
* 根据序列名称将序列值更新为value + increment；
* 内存中使用记录本次获取序列的区间；
* 回滚或者提交锁定数据，连接超时、异常等必须回滚正常处理逻辑必须提交防止一直占用资源；
* 业务依次使用缓存的序列区间值；
* 为保证边界值不会被重复使用，每批获取的数据不使用最小值；

### 需求分解与分配

批量获取序列公共方法；

# 业务整体性能目标

# 系统接口描述

请参见华为会员服务器接口规范.docx。