网店用户授权信息改造

项目技术方案

|  |  |  |
| --- | --- | --- |
| 日期 | 作者 | 说明 |
| 2011．12.21 | 刘凌 | 初稿 |
|  |  |  |

1. 项目背景

由于在使用TOP授权接口时，需要将各个产品分开。在技术上则需要将各个产品的appkey所对应的sessonKey进行分离。故为了满足该技术上面的要求，做此改造！为了提高代码质量和水平，请各自配置Tala，会做好codereview的工作，同时也会集成test框架。

Tala的使用方法见：<http://b2b-doc.alibaba-inc.com/pages/viewpage.action?pageId=57561365>

JTester的使用方法见：<http://b2b-doc.alibaba-inc.com/pages/viewpage.action?pageId=62500272>

1. 针对即有问题，新生成的代码简单约定
2. 每个方法体的代码不能超100行，如果超过，请修改；
3. 自己调试打印日志时，一律为debug级别，且加上isEnabledDebug判断。info级别的日志可能会在生产环境中打开，所以要对该日志的打印是否有必要，否则请设置为debug；
4. 每个任务业务逻辑描述，加在类的头部，至少说清楚这个任何类用在哪个模块，起什么作用；
5. 在spring配置任务类时，不能直接在biz层，需要单独在包里建立相关Task类。
6. 网店调用的TOP接口列表(共61个接口，43个需要授权， 17个不需要授权，1个即将下线。)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 淘宝接口 | 接口类别 | 用途 | 调用量 | 授权 |
| 用户API | | | | |
| taobao.user.get | 用户API | 获取用户信息 | 14789640 | **√** |
| taobao.user.tag.add | 用户API (隐私) | 用户打标签 | 211728 | **√** |
| 子帐号管理API | | | | |
| taobao.sellercenter.subusers.get | 子帐号管理API | 查询指定账户的 子账号列表 | 1154070 | **√** |
| 店铺API | | | | |
| taobao.sellercats.list.get | 店铺API | 卖家自定义商品类目 | 911088 | **×** |
| taobao.shopcats.list.get | 店铺API | 获取卖家的店铺类目 | 8056 | **×** |
| taobao.shop.remainshowcase.get | 店铺API | 获取剩余橱窗数量 | 36684938 | **√** |
| taobao.shop.get | 店铺API | 获取卖家店铺的基本信息 | 18180211 | **√** |
| 在线订购API | | | | |
| taobao.vas.pagesubs.get | 在线订购API (隐私) | 查询汇金订购关系 | 25800 | **×** |
| taobao.vas.service.validate | 在线订购API | 增值服务订购服务验证 | 12427420 | **×** |
| taobao.vas.subscribe.get | 在线订购API | 订购关系查询 | 86371597 | **×** |
| taobao.vas.order.search | 在线订购API | 查询在线订购订单记录 | 0 | **×** |
| ~~taobao.vas.service.getServTimes~~ |  |  | ~~1650309~~ | ~~将要下架~~ |
| 店铺会员管理API | | | | |
| taobao.crm.membergrade.set | 店铺会员管理API | 卖家设置会员的等级 | 30526 | **√** |
| taobao.crm.grade.set | 店铺会员管理API | 卖家设置等级规则 |  | **√** |
| 类目API | | | | |
| taobao.itempropvalues.get | 类目API | 获取标准类目属性值 | 11141662 | **×** |
| taobao.itemcats.get | 类目API | 标准商品类目 | 15078918 | **×** |
| 营销API | | | | |
| taobao.promotion.coupons.get | 营销API | 查询卖家优惠券 |  | **√** |
| taobao.promotion.meal.get | 营销API | 搭配套餐查询 |  | **√** |
| 主动通知业务API | | | | |
| taobao.increment.user.authorize | 主动通知业务API (隐私) | 用户授权ISV获取其 增量消息 | 5314311 | **√** |
| taobao.increment.items.get | 主动通知API (增值) | 变量通知信息 | 5647287 | **√** |
| taobao.increment.authorizemessages.get | 主动通知业务API | 用户授权信息查询 | 5273957 | **×** |
| 店铺会员管理API | | | | |
| taobao.crm.grade.get | 店铺会员管理API | 卖家查询等级规则 | 77160 | **√** |
| taobao.crm.rule.group.set | 店铺会员管理API | 设置规则适用的分组 |  | **√** |
| taobao.crm.rule.add | 店铺会员管理API | 分组规则添加 |  | **√** |
| taobao.crm.members.get | 店铺会员管理API | 获取卖家的会员 |  | **√** |
| taobao.crm.group.delete | 店铺会员管理API | 删除分组 |  | **√** |
| taobao.crm.rules.get | 店铺会员管理API | 获取规则 | 94 | **√** |
| taobao.crm.groups.get | 店铺会员管理API | 查询卖家的分组 | 213 | **√** |
| taobao.crm.members.search | 店铺会员管理API | 获取卖家会员 | 660 | **√** |
| taobao.crm.group.add | 店铺会员管理API | 卖家创建一个分组 |  | **√** |
| taobao.crm.grouptask.check | 店铺会员管理API | 查询分组任务是否完成 |  | **√** |
| taobao.crm.rule.delete | 店铺会员管理API | 分组规则删除 |  | **√** |
| taobao.crm.members.groups.batchdelete | 店铺会员管理API | 批量删除分组 |  | **√** |
| taobao.crm.members.group.batchadd | 店铺会员管理API | 给一批会员添加一个分组 |  | **√** |
| taobao.crm.group.update | 店铺会员管理API | 修改一个已经存在的分组 |  | **√** |
| 退款API | | | | |
| taobao.refund.get | 退款API | 获取单笔退款详情 | 635612 | **√** |
| 商品API | | | | |
| taobao.item.update.listing | 商品API | 一口价商品上架 | 2512442 | **√** |
| taobao.item.update | 商品API | 修改宝贝 | 12339321 | **√** |
| taobao.item.sku.update | 商品API | 修改宝贝SKU信息 | 136786 | **√** |
| taobao.item.get | 商品API | 获取宝贝信息 | 216611612 | **~~√~~** |
| taobao.items.list.get | 商品API | 批量获取宝贝信息 | 29536874 | **~~√~~** |
| taobao.items.search | 商品API | 批量查询宝贝 | 7327413 | **×** |
| taobao.items.onsale.get | 商品API | 卖家架上宝贝 | 31486374 | **√** |
| taobao.item.recommend.add | 商品API | 橱窗推荐一个商品 | 394505535 | **√** |
| taobao.item.recommend.delete | 商品API | 取消橱窗推荐的一个商品 | 471175627 | **√** |
| taobao.item.skus.get | 商品API | 根据商品ID列表 获取SKU信息 | 13774366 | **×** |
| taobao.items.get | 商品API | 搜索商品信息 | 28291 | **×** |
| taobao.items.inventory.get | 商品API | 库存中的商品列表 | 481928 | **√** |
| 交易API | | | | |
| taobao.topats.history.trades.get | 交易API (私有接口) | 历史交易接口 | 14414 | **√** |
| taobao.trades.sold.get | 交易API | 卖家已卖出的交易数据 | 5176477 | **√** |
| taobao.trade.fullinfo.get | 交易API | 获取单笔交易的详细信息 | 685939508 | **√** |
| taobao.topats.result.get | 交易API | 获取异步任务结果 | 104885 | **×** |
| taobao.trades.sold.increment.get | 交易API | 卖家已卖出的 增量交易数据 | 292468840 | **√** |
| 评价API | | | | |
| taobao.traderate.list.add | 评价API | 针对父子订单 新增批量评价 | 79548210 | **√** |
| taobao.traderates.get | 评价API | 获取评价信息 | 10728510 | **√** |
| taobao.traderates.search | 评价API | 商品评价信息 | 24897 | **×** |
| 物流API | | | | |
| taobao.logistics.orders.detail.get | 物流API | 批量查询物流订单 | 9 | **√** |
| taobao.delivery.send | 物流API (隐私) | 发货处理 | 1805026 | **√** |
| taobao.logistics.trace.search | 物流API | 物流流转信息查询 | 37336828 | **×** |
| taobao.logistics.orders.get | 物流API | 批量查询物流订单 | 31079954 | **√** |
| taobao.logistics.companies.get | 物流API | 查询物流公司信息 | 83 | **×** |

以上接口都需要做好封装，详细封装说明见后。

二、现存任务列表(共22个任务，19个类)

|  |  |  |  |
| --- | --- | --- | --- |
| Trigger名称 | 类名 | 备注 | 是否  TOP |
| smsStatTask | com.alibaba.eshop.task.SmsStatTask |  | ? |
| smsAlarmTaskTrigger | com.alibaba.eshop.task.SmsAlarmTask |  | **√** |
| studyUserInfoToNoifyTask | com.alibaba.eshop.task.StudyUserInfoToNoifyTask |  | **√** |
| delHistoryTaskTrigger | com.alisoft.c2c.delHistoryData.manager.DelHistoryData |  | **×** |
| autoSendAutoPostErrorWWMsgToUsersJobTrigger | com.alisoft.c2c.task.AutoContactTask |  | **√** |
| autoUpdateSettingTaskJobTrigger | com.alisoft.c2c.task.AutoContactTask |  | **√** |
| doMonitorJobTrigger | com.alisoft.c2c.task.MonitorTask |  | **×** |
| activeSessionKeyTaskTrigger | com.alisoft.c2c.task.activeSessionKey.ActiveSessionKeyTask | 弃用 |  |
| autoPostJobTrigger | com.alisoft.c2c.task.BatchUpShelfTask |  | **√** |
| autoShopWindowTrigger\_D\_1 | com.alisoft.c2c.task.shopwindow.AutoShopWindowTask |  | **√** |
| autoShopWindowTrigger\_D\_2 | com.alisoft.c2c.task.shopwindow.AutoShopWindowTask |  | **√** |
| autoWindowDueRemindTrigger | com.alisoft.c2c.task.shopwindow.AutoShopWindowTask |  | **√** |
| userLevelTrigger | com.alisoft.c2c.task.UserLevelTask |  | **×** |
| syncSubInfoTaskJobTrigger | com.alisoft.c2c.task.SyncSubInfoTask |  | **×** |
| supplyGoodsTaskTrigger | com.alibaba.eshop.supplygoods.service.SupplyGoodsService | 配置文件散落 | ? |
| smsSendServerTask | com.alibaba.eshop.sms.impl.DefaultSMSSendServer | 配置文件散落 | **×** |
| smsMonitorTrigger | com.alibaba.eshop.sms.SmsMonitorTask |  | **×** |
| smsLogisticsTrigger | com.alibaba.eshop.sms.SmsLogisticsReminder |  | **√** |
| smsMarketingTrigger | com.alibaba.eshop.sms.marketing.SmsMarketing |  | **√** |
| doMobileCacheJobTrigger | com.alisoft.c2c.task.mobile.task.MoblieCacheTask |  | **×** |
| doSaleStatJobTrigger | com.alibaba.eshop.stock.report.task.SaleStatTask | 包位置 不对 | **√** |
| doTaoReportTrigger | com.alibaba.eshop.task.tool.task.TaoReportTask |  | **×** |

由于任务已经达20多个，包结构不明晰，故这次改造规定:

1. 项目eshop，将所有的task类添加或移动到com.alisoft.c2c.task
2. 项目neweshop，将所有的task类添加或移动到com.alibaba.eshop.task
3. Spring配置要求:
4. task类以task后缀结尾，如doSomethingTask
5. 配置FactoryBean以detail后缀结尾，如doSomethingDetail
6. 配置trigger时以trigger后缀结尾，如doSomethingTrigger
7. 保持所有名称一致，只是后缀不同，如果有多个情况以下划线+数字来判断,如doSomethingDetail\_1
8. 因为neweshop依赖eshop，所以将task的spring配置文件统一移动到neweshop工程，如果某个任务配置过多，可以单独出来。如spring-app-task.xml, spring-app-task-autoshopwindow.xml
9. Web更改

4.1在每个服务页面，都需要检测其sessionkey是否存在且有效，否则弹出授权页面；

4.2 将用户每个应用的sessionKey都持久化；

4.3 该更改发布要提前于定时任务的改造，以充分获取用户的sessionKey；

4.4 发布后，及时出公告通知用户来系统到网店授权，从而让系统获取到用户的sessionKey；

4.5 淘宝授权页面开发文档：<http://open.taobao.com/doc/detail.htm?id=105>

4.6 淘宝授权时长说明文档：http://open.taobao.com/doc/detail.htm?id=817

4.7 sessionKey注意事项和检测点： sessionkey以4开头的都为将废弃的session，以6开头的为新授权产生的session。

1. 封装TOP
   1. 功能性
2. 封装所有可预知的错误码，以异常表现
3. sessionKey失效异常
4. sessionkey不正确的异常
5. 淘宝其他返回结果码的异常
6. 到遇到网络连接超时时或由于硬件网络引起的异常，才进行重试
7. 日志记录每个TOP调用的花费时间，业务模块，调用状态（成功或失败）
8. 传入用户唯一标识，传入appkey，从而获取到top的sessionKey
9. 以继承或其他方式将taobao的类进行替换以供业务代码使用(?)
10. Taobao中SDK的Response和Request由于太多，可以不作封装，意义也不大(?)
    1. 接口名称：IAppClient
11. 在业务代码中不能出现直接使用top的SDK方法
12. 业务代码可以不用关心sessionKey的存在，只有当有sessionKey异常时才触发，可以替业务代码选择合适的sessionKey来使用

代码示例

|  |
| --- |
| //需要sessionKey的  Response response = appClient.execute(Request request, String domainUserId)  //不需要sessionKey的  Response response = appClient.execute(Request request) |

* 1. 获取应用客户端

根据antx的配置项，在spring的配置所有应用的client，然后在FactoryClient中根据调用参数选择正确的client.

|  |  |  |
| --- | --- | --- |
| FactoryClient.getAppClient | | |
| 作用 | 获取指定应用配置的client | |
| 参数1 | String domainUserId | 用户唯一标识 |
| 参数2 | String app\_id | 业务模块编号(自定义) |
| 返回值 | IAppClient | 返回使用该app的sessionKey的client，如果失效则抛出sessionKeyExpiredException |

|  |  |  |
| --- | --- | --- |
| FactoryClient.getAppClient | | |
| 作用 | 可以根据规则，选择合适的client | |
| 参数1 | String domainUserId | 用户唯一标识 |
| 返回值 | IAppClient | 返回使用该app的sessionKey的client，如果失效则抛出sessionKeyExpiredException |

1. 数据库

用户在各个应用的sessionKey信息，存放于MySQL中。定时清除过期用户的sessionkey记录。

|  |  |  |
| --- | --- | --- |
| 字段名称 | 类型 | 使用 |
| id | Bigint(20) | 自增ID |
| domain\_user\_id | Varchar(32) | 用户唯一ID |
| app\_id | Int | 应用标识 |
| session\_key | Varchar(50) | 用户会话 |
| expiration\_date | datetime | 失效时间 |

1. 通用应用的迁移
2. 将现有的通用应用转为可订购，当未订购的用户跳转到订购页面；
3. 将现有的免费应用的订购关系转移到通用应用下；
4. 保证通用应用的sessionKey的有效时间为免费应用的订购关系有效期一致；
5. 保证通用应用的sessionKey是经过用户授权的。
6. 应用权限

共八个收费，一个免费，其每个应用需要的权限整理如下。

|  |  |  |
| --- | --- | --- |
| 模块名称 | 权限 | 备注 |
| 网店版批量打印 |  |  |
| 批量推荐 |  |  |
| 自动橱窗 |  |  |
| 淘关怀 |  |  |
| 批量修改 |  |  |
| 发货与库存 |  |  |
| 行情参谋 |  |  |
| 淘榜样 |  |  |