

运维操作手册

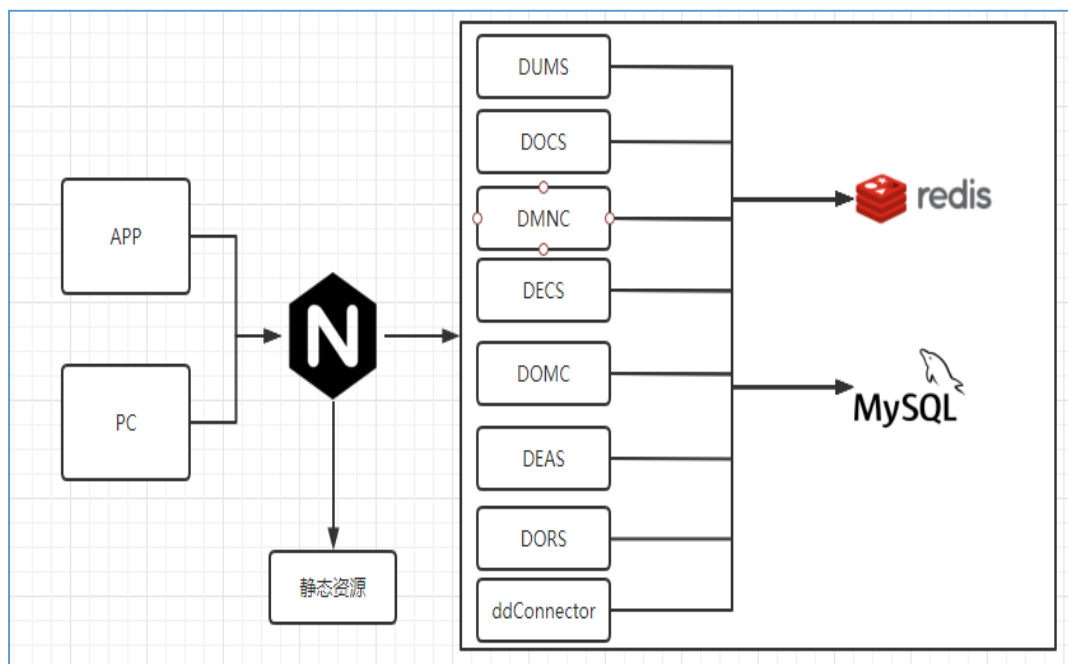
1. 概述

本文档主要用于常用运维功能操作步骤说明。

2. 快速了解

2.1. 项目简介

系统介绍：



2.2. 功能划分

平台服务于功能对应：

服务名称	功能描述
dums	用户登录、游客注册、加盟、推广权限获取、支付宝收款账号、绑定银行卡、修改服务密码、更换手机号
docs	购买设备支付(支付宝)、等级计算(修复)、提现、话分计算、有效卡计算、加盟个数限制
deas	加盟保证金、公益金、人脉团队信息、账户、首页 banner、售号订单、开卡订单

dmnc	App 消息获取、
decs	订单查询、商户管理、banner 管理、黑名单管理、推广/开卡/制卡订单查询
domc	发送系统消息、app 版本管理、app 检查更新、售卖设备记录任务
dors	短信发送
ddConnector	接收平台的回调数据

3. 常用运维操作教程

3.1.1. 配置说明

1、目采用多环境配置,在 src/main/resources/application.properties 配置当前所属的环境
spring.profiles.active=

2、服务配置说明 server-ddd.properties

```

1# 服务器名称
2 server.name = dums-server
3# 项目名称
4 server.project = dums
5# 服务器绑定端口
6 server.port = 6090
7# 服务器项目顶级包名
8 server.package = com.borya
9# HTTP请求最大字节数
10 server.maxContentLength = 134217728
11# 是否使用HTTPS
12 server.useHttps = false
13# HTTPS证书链地址
14 server.httpsCertChainFile = D:\\xxx\\https\\1538711886161.key
15# HTTPS私Key文件地址
16 server.httpsPrivateKeyFile = D:\\xxx\\https\\1538711886161.pem
17# Log4j日志文件目录
18 server.log4jdir = D:/xums/logs
19# 是否允许发送广播
20 server.so.broadcast = false
21# 是否启用心跳保活机制
22 server.so.keepalive = true
23# 发送缓冲区字节数
24 server.so.sndbuf = 1024
25# 接收缓冲区字节数
26 server.so.rcvbuf = 4096
27# 是否重用地址
28 server.so.reuseaddr = true
29# 是否启用连接延迟关闭
30 server.so.linger = 0
31# 连接队列大小
32 server.so.backlog = 16384
33# 连接超时
34 server.so.timeout = 5000
35# 是否关闭延迟发送
36 server.tcp.nodelay = true

```

注意:server.project 配置的字符为 app、web 端访问的地址中需要携带,修改配置后,需要重启服务生效

3. Redis 配置

项目缓存数据库使用的是 Redis,找到 src/main/resources/conf/redis-dev.properties

```
redis-dev.properties
1# 最大活跃对象数
2redis.pool.maxTotal = 64
3# 最大空闲对象数
4redis.pool.maxIdle = 16
5# 最小空闲对象数
6redis.pool.minIdle = 0
7# 是否在创建时测试对象
8redis.pool.testOnCreate = true
9# 是否在借出时测试对象
10redis.pool.testOnBorrow = false
11# 是否在还回时测试对象
12redis.pool.testOnReturn = false
13# 是否在空闲时测试对象
14redis.pool.testWhileIdle = true
15# 连接池耗尽时调用者最大阻塞时间, 超时抛出异常, -1表示永不超时
16redis.pool.maxWaitMillis = 1000
17
18# 内网IP地址
19redis.ip = 127.0.0.1
20redis.port = 6379
21redis.timeout = 5000
22redis.password = 123456
23
```

修改配置新需要重启后生效

3. MySQL 配置

安装完成后进行项目配置 src/main/resources/conf/sql-ddd.properteis

```
sql-dev.properties
1# 最大活跃对象数
2sql.pool.maxTotal = 64
3# 最大空闲对象数
4sql.pool.maxIdle = 16
5# 最小空闲对象数
6sql.pool.minIdle = 0
7# 是否在创建时测试对象
8sql.pool.testOnCreate = true
9# 是否在借出时测试对象
10sql.pool.testOnBorrow = false
11# 是否在还回时测试对象
12sql.pool.testOnReturn = false
13# 是否在空闲时测试对象
14sql.pool.testWhileIdle = true
15# 连接池耗尽时调用者最大阻塞时间, 超时抛出异常, -1表示永不超时
16sql.pool.maxWaitMillis = 1000
17
18# 内网数据库地址]
19sql.driverClassName = com.mysql.jdbc.Driver
20sql.username = root
21sql.password = 123456
22sql.url = jdbc:mysql://127.0.0.1:3306/xxx?characterEncoding=utf8&useSSL=false
??
```

修改配置后需要重启服务后生效

4. 日志配置

日志配置采用 log4j, 生成日志文件在 log4j.dml 配置, 修改后需要重启后生效。

3.2. 重要功能说明

3.2.1. 号卡佣金配置修改

1、找到佣金比例配置表 `sk_card_income_conf`，根据 `money_type` 区分号码占用费、预存，修改相应字段，修改后立即生效，新产生的号卡业务按照最新配置佣金比例进行返佣，不用重启服务。

3.2.2. 话分佣金配置修改

1、步骤说明

- (1)、decs 上传话分文件并将文件内数据导入数据库
- (2)、向 docs 发送请求，docs 收到请求开始计算佣金

2、话分佣金比例在 `sk_huafen_income_conf` 表，根据比例与商户等级可以计算出各个商户所得佣金，话分佣金记录表为 `sk_huafen_income`

`sk_huafen_income_conf`：表只有一条数据，配置不同等级的佣金比例



id	shanghu	shanghu_jianjie	shanghu_pingji	tuiguang	tuiguang_jianjie	tuiguang_pingji	yunying	yunying_jianjie	yunying_pingji	zhanlue	zhanlue_jianjie	zhanlue_pingji
21												

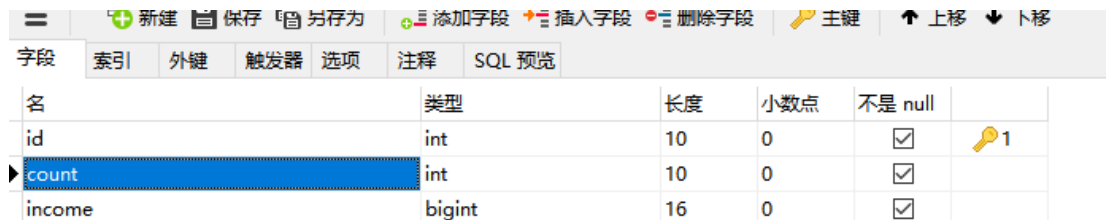
`sk_huafen_income`:

id	file_id	month_date	create_time	merchant_id	ytsk_status	money	type	discount	income
(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)

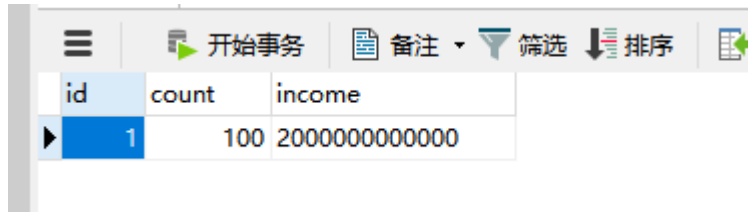
free_zen	modify_time	out_order_id	charge_type	from_merchant_id	out_month_date
(Null)	(Null)	(Null)	(Null)	(Null)	(Null)

3.2.3. 开卡达量佣金配置表

1、找到 `sk_daliang_conf` 表



字段	索引	外键	触发器	选项	注释	SQL 预览			
名					类型	长度	小数点	不是 null	
id					int	10	0	<input checked="" type="checkbox"/>	1
count					int	10	0	<input checked="" type="checkbox"/>	
income					bigint	16	0	<input checked="" type="checkbox"/>	



id	count	income
1	100	2000000000000

2、更加 count 张数修改 income， income 为达量激励,分后 8 位位单位
例如修改:开卡达到 100 张的激励 100 元

```
Update sk_daliang_conf set income=1000000000000 where count=100;
```

3、修改后立即生效

3.2.4. 加盟佣金开关

1、此操作的开关放在 redis 中存储,采用字符 String 格式:

2、键值说明

eg:

打开加盟佣金开关: `set dd:close:commission 1`

关闭加盟佣金开关: `del dd:close:commission`

3.2.5. 等级计算开关

1、此操作的开关放在 redis 中存储,采用字符 String 格式:

2、键值说明

eg:

打开等级计算开关: `set dd:calculate:buy:str 1`

关闭等级计算开关: `del dd:calculate:buy:str`

3.2.6. 加盟等级计算配置

1、佣金配置:

配置文件在 docs 网元, `conf-ddd.properties` 配置。如需修改直接修改配置文件:

```

28 #佣金:分后面8个0位单位
29 #D
30 commission.shanghu.zhijie = 400000000000
31 commission.shanghu.jianjie = 200000000000
32 commission.shanghu.pingji = 1
33 #C
34 commission.tuiguang.zhijie = 400000000000
35 commission.tuiguang.jianjie = 500000000000
36 commission.tuiguang.pingji = 2
37 #B
38 commission.yunying.zhijie = 400000000000
39 commission.yunying.jianjie = 1000000000000
40 commission.yunying.pingji = 5
41 #A
42 commission.zhanlue.zhijie = 400000000000
43 commission.zhanlue.jianjie= 1500000000000
44 commission.zhanlue.pingji= 8
45
46 #纯利润,单位分
47 commission.profit.fen = 70000

```

注意:

#纯利润,单位分

commission.profit.fen = 70000 //计算出的利润小于这个配置则佣金计算出错, 关闭佣金计算

2、等级计算配置:

配置文件也在 docs 网元, conf-ddd.properties 配置。如需修改直接修改配置文件:

```

8 #层级
9 #C
0 merchant.hierarchy.tuiguang.zhitui.num = 2
1 merchant.hierarchy.tuiguang.tuandui.num = 4
2 merchant.hierarchy.tuiguang.active.card.num=-1
3 #B
4 merchant.hierarchy.yunying.zhitui.num = 3
5 merchant.hierarchy.yunying.zhitui.tuiguang.num = 0
6 merchant.hierarchy.yunying.tuandui.num = 6
7 merchant.hierarchy.yunying.active.card.num=160
8 #A
9 merchant.hierarchy.zhanlue.zhitui.num = 5
0 merchant.hierarchy.zhanlue.zhitui.yunying.num = 0
1 merchant.hierarchy.zhanlue.tuandui.num = 8
2 merchant.hierarchy.zhanlue.active.card.num=-1

```

说明: 以上的配置以 A 举例, 要达到等级为 A 则必须直接邀请 5 个商户, 其中 5 个商户里面有 0 个达到 B, 并且有 8 个团队人数, 并且成功开卡激活张数没有限制;

以上 1 和 2 如果配置修改后需要重新打包 docs 网元上传到服务器后重启服务才能生效;

3.2.7. 加盟等级纠正

1、找到 docs 服务请求接口

浏览器执行: `http://host:port/docs/s/commission/repair`

命令行执行: `curl http://host:port/docs/s/commission/repair`

接口返回: `{"code": "200"}`就说明请求成功了

3.2.8. 加盟佣金计算

1、找到 docs 服务请求接口

浏览器执行: `http://host:port/docs/s/commission/calculate`

命令行执行: `curl http://host:port/docs/s/commission/calculate`

接口返回: `{"code": "200"}`就说明请求成功了

3.2.9. 加盟数限制配置

1、连接数据库找到 `tb_sys_conf` 表

2、打开 `tb_sys_conf` 表。修改最大加盟数量,(目前表里面只有一条数据)

把 `prop_value` 值修改为想要限制的加盟数即可, 比如限制每天加盟只允许 10 个:

```
UPDATE `tb_sys_conf` SET prop_value='10';
```

3、修改好数据库后更新 redis 键值, 或删除 redis:

`dd:merchant:sys:conf:hash` (如果 reids 图形化工具, 可选中键后直接右键删除)

4、如果想达到不允许加盟,参考 2.3 步骤, 表里面的 `prop_value` 值修改为 0 再把 redis 更新:

```
UPDATE `tb_sys_conf` SET prop_value='0';
```

或者也可以直接连接 reids 把 redis 的相关值改为 0 即可。

5、修改加盟限制数后, 立即生效。