

beanpay

提供钱包帐户的加款、扣款、退款
服务包可用数的增加、扣减、退回

- ✓ 支持钱包帐户
- ✓ 支持服务包
- ✓ 独立服务运行，使用接口集成
- ✓ 集成服务运行，通过包代码集成
- ✓ 高并发支持
- ✓ 基于 hydra 构建

一、独立服务

启动一个独立的服务器，向外提供 api , rpc 服务

1、创建数据库

编译beanpay服务

```
~/work/bin$ go install github.com/micro-plat/beanpay #mysql
```

或

```
~/work/bin$ go install -tags "oci" github.com/micro-plat/beanpay # oracle
```

创建数据表

beanpay [注册中心地址] [平台名称] 即可将 beanpay 需要的表创建到 /平台/var/db/db 配置对应的数据库中。

```
~/work/bin$ beanpay zk://192.168.0.109 mall #根据/mall/var/db/db创建数据库

或

~/work/bin$ beanpay zk://192.168.0.109 mall mdb #根据/mall/var/db/mdb创建数据库
```

2、启动 api 服务

启动后对外提供 http , rpc 服务

编译api服务

```
~/work/bin$ go install github.com/micro-plat/beanpay/apiserver
```

安装服务

```
apiserver install -r zk://192.168.106.18 -c t
```

启动服务

```
apiserver start -r zk://192.168.106.18 -c t
```

3、服务列表

分类	服务名	类型	说明
钱包帐户	/account/create	api、rpc	添加账户
	/account/balance/add		账户余额加款
	/account/balance/deduct		账户余额扣款
	/account/balance/refund		账户余额退款
	/account/balance/query		账户余额查询
	/account/record/query		账户变动查询
服务包	/package/create	api、rpc	添加服务包
	/package/capacity/add		服务包数量添加
	/package/capacity/deduct		服务包数量扣除

	/package/capacity/refund		服务包数量退回
	/package/capacity/query		服务包数量查询
	/package/record/query		服务包变动查询

二、代码集成

将数据表直接创建到业务系统数据库，或使用独立的数据库，业务系统使用本地代码，直接调用调用包代码进行帐户、服务包操作

1、创建数据表:

编译 beanpay

```
~/work/bin$ go install github.com/micro-plat/beanpay #mysql
```

或

```
~/work/bin$ go install -tags "oci" github.com/micro-plat/beanpay # oracle
```

运行命令

beanpay [注册中心地址] [平台名称] 即可将 beanpay 需要的表创建到 /平台/var/db/db 配置对应的数据库中。

```
~/work/bin$ beanpay zk://192.168.0.109 mall #根据/mall/var/db/db创建数据库
```

或

```
~/work/bin$ beanpay zk://192.168.0.109 mall mdb #根据/mall/var/db/mdb创建数据库
```

2、编码

创建帐户

```
account, err := beanpay.CreateAccount(ctx,
    ctx.Request.GetString("eid"),
    ctx.Request.GetString("name"))
if err != nil {
    return err
}

ctx.Log.Info("3. 处理返回结果")
return account
```

帐户加款

```
record, err := beanpay.AddAmount(ctx,
    ctx.Request.GetString("eid"),
    ctx.Request.GetString("trade_no"),
    ctx.Request.GetInt("amount"))
if err != nil {
    return err
}
return record
```

帐户扣款

```
record, err := beanpay.DeductAmount(ctx,
    ctx.Request.GetString("eid"),
    ctx.Request.GetString("trade_no"),
    ctx.Request.GetInt("amount"))
if err != nil {
    return err
}

ctx.Log.Info("3. 处理返回结果")
return record
```

其它操作请查看<http://github.com/micro-plat/beanpay/beanpay/service.go>