

## 5.28 限流提测文档 V1.0.2

### 1. 需求

限流服务目前仅在渠道内部使用，后台配置的限流指令默认是渠道所有。应二维码团队希望调用限流服务的诉求，对限流进行了功能扩展，即，不同团队均可以进行配置自己的限流指令，来达到实现限流的目的。

### 2. 变更内容

1. 限流请求参数`LimitReqDTO`中添加`requestSystem`字段：用以区分限流请求来自哪个系统。为兼容之前的逻辑（目前线上的`channel`在调用限流服务时未设置该参数），`requestSystem`默认值为“`channel`”（即，如果不设置该参数的值，该请求默认为来自渠道）。
2. `requestSystem`校验：设置开关`isCheck`，后台可配置参数为`true`或`false`，`true`代表需要对`requestSystem`的合法性进行校验，`false`为不校验。校验的标准为：用户传入的`requestSystem`是否已在后台配置。如果校验不通过，放行,不进行限流。
3. 缓存key更新：之前缓存的key为：`limiting{}`+限流指令，引入请求系统字段后，为保证来自一个请求系统的限流指令最终落脚到redis中的一个槽中，将前缀`limiting{}`改为`limiting{requestSystem}:limitOrder`，其中`requestSystem`传入的参数值，`limitOrder`为限流指令内容。
4. http接口：删除`LimitController`，不再提供http请求限流接口。
5. redis集群连接超时时间由300s改为50ms。
6. dubbo接口服务端连接超时时间设置为100ms。
7. `LimitConfig`接口：`addConfig`方法，对增加`requestSystem`进行适配。

### 3. 接口

#### 1. facade接口 1527562954553

result:

true: 交易通过

false: 交易不通过

（所有限流异常均返回true，不影响实时交易）

#### 2. `LimitConfigController` 1527564809196

- URL: `http://url:port/config/add`，参数格式: json 请求: post方式
  - 新增请求系统，参数如下：

```
{
  "code": "requestSystem",
  "codeValue": "channel"
}
```

- 设置校验值，参数如下：

```
{
  "code": "isCheck",
  "codeValue": "true"
}
```

### 3. LimitOrderController

- URL: http://url:port/limit/set, 参数格式: json 请求: post方式
  - 新增限流指令, 参数如下

```
{
  "keys":
  {
    "db": "11",
    "drctn": "22"
  },
  "level": "man",
  "value": 100,
  "requestSystem": "channel"
}
```

### 4. 影响分析

已适配之前的调用方式, 新增的字段和参数均已设置默认值, 不影响目前渠道线上调用限流的服务。后期渠道上线时再同步更新调用限流的请求参数即可。

### 5. 提测前提

- ☒ FindBugs扫描完成并修复所有扫描bug;
- ☒ 阿里巴巴代码规范插件扫描完成并修复所有待改善点;
- ☒ 提供提测相关的接口文档、需求文档, 如果全新项目需要提供设计文档;
- ☒ 如果提测功能存在与其他模块交互, 开发人员需要完成开发联调;
- ☒ 代码评审