FAMS开发手册（v0.0.1）

# 目录

[目录 1](#_Toc34394231)

[修订（开发）记录 3](#_Toc34394232)

[一、FAMS简介 4](#_Toc34394233)

[二、FAMS总体架构 5](#_Toc34394234)

[2.1 业务层面系统划分 5](#_Toc34394235)

[2.2 服务层面系统划分 5](#_Toc34394236)

[三、环境汇总 6](#_Toc34394237)

[3.1 外部服务依赖 6](#_Toc34394238)

[3.2 开发环境 6](#_Toc34394239)

[3.2.1 Mysql 6](#_Toc34394240)

[四、模块划分 7](#_Toc34394241)

[4.1 总体模块划分 7](#_Toc34394242)

[4.2 fams-framework模块 7](#_Toc34394243)

[4.3 fams-springboot-starter模块 8](#_Toc34394244)

[五、模块依赖 9](#_Toc34394245)

[六、服务治理微服务 10](#_Toc34394246)

[6.1 服务注册中心MonitorService 10](#_Toc34394247)

[6.2 权限管理中心AuthService 10](#_Toc34394248)

[6.3 配置管理中心ConfigService 10](#_Toc34394249)

[七、Framework层对外暴露的Service 11](#_Toc34394250)

[7.1 Core模块 11](#_Toc34394251)

[7.1.1 数据加载服务DataLoaderService 11](#_Toc34394252)

[7.1.2 全局参数服务GlobalParamService 11](#_Toc34394253)

[7.1.3 redis服务RedisService 12](#_Toc34394254)

[7.1.4 命令行服务CommondService 12](#_Toc34394255)

[7.2 Ignite模块 14](#_Toc34394256)

[7.2.1 数据加载服务 14](#_Toc34394257)

[7.2.2 数据缓存服务 14](#_Toc34394258)

# 修订（开发）记录

| 日期 | 版本 | 修订说明 | 修订者 |
| --- | --- | --- | --- |
| 2020.03.04 | v0.0.1 | 新建 截止到2020.03.04 已有功能：   1. ignite集成cache 2. 序列化模块，框架搭建，规范化编码 3. 集成redis，集成mybatis | 姜瀚 |
|  |  |  |  |

# 一、FAMS简介

金融资产管理系统FAMS（Financial Asset Manager System），旨在建立一个涵盖金融投资行为的前中台的全覆盖产品，对标系统为恒生O32系统。愿景为可以支持基金公司级的日常投资业务需求。

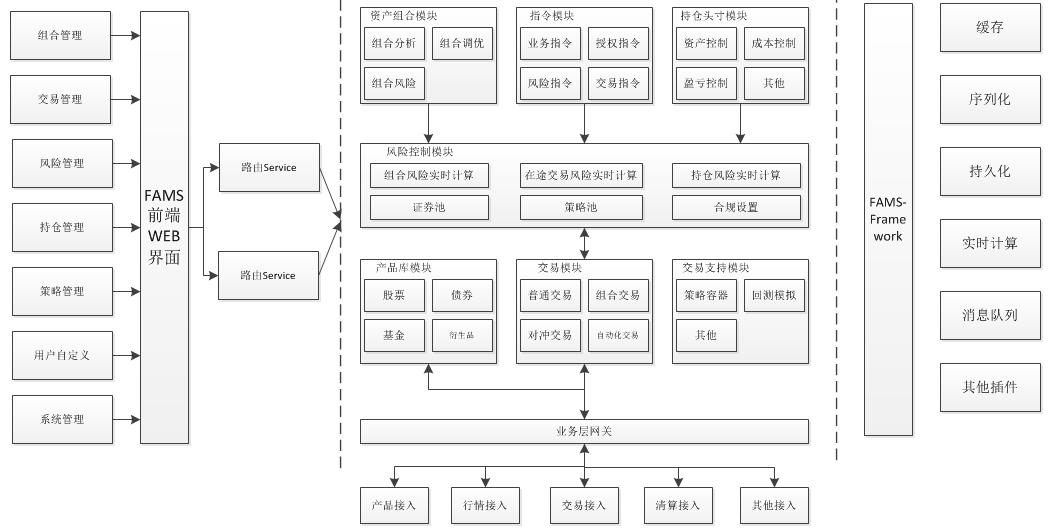
从业务角度讲，FAMS提供投前、投中、投后的管理。投前服务，将提供策略池、股票池、投前风控等功能；投中服务，将提供内置的指令流转模块支持投资指令的生命周期、投中市场/系统风险控制；投后服务，提供实时估值，持仓管理，持仓风险管理等功能。

从系统角度讲，FAMS提供包括交易、行情、风控、估值服务。交易模块包括，普通交易、策略交易、伪高频交易（内存交易）；风控风控包括事前风控、实时交易风控、事后风控等。

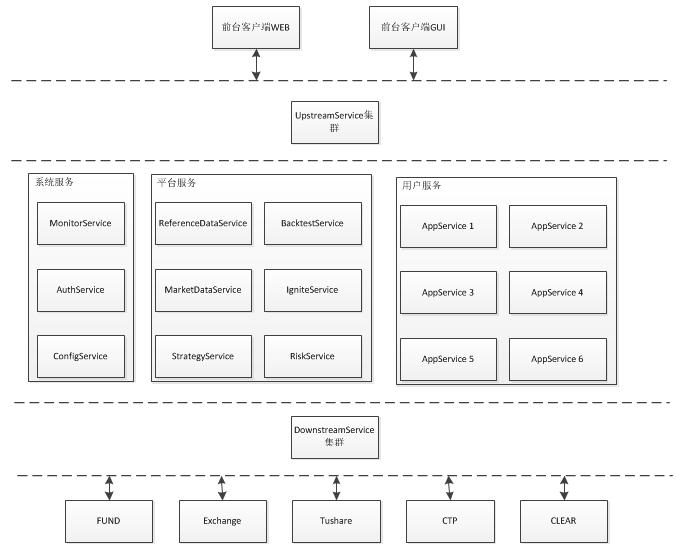
从架构角度讲，FAMS采用标准的分布式架构，支持高可用性、高可扩展。从服务粒度讲，FAMS主体架构是基于Java语言的Spring-Cloud微服务架构，同时支持dubbo类型的服务接入。可扩展性：采用松散的结构，服务之间弱依赖性，支持python版本和c/c++版本的微服务接入，以支持更多用户的习惯。

# 二、FAMS总体架构

## 2.1 业务层面系统划分



## 2.2 服务层面系统划分



# 三、环境汇总

## 3.1 外部服务依赖

|  |  |  |
| --- | --- | --- |
| 外部依赖组件名称 | 版本 | 系统中功能 |
| Zookeeper | 3.5.6 | Kafka依赖，后续作为dubbo类服务的服务治理 |
| Kafka | 2.4.0 | 消息中间件 |
| Redis | 3.2.1 | 普通数据的缓存 |
| Mysql | 5.6.46 | 普通数据的存储 |
| Ignite | 2.7.6 | 内存计算、流数据处理 |

## 3.2 开发环境

### 3.2.1 Mysql

|  |  |
| --- | --- |
| Ip | 122.51.159.32 |
| Port | 3307 |
| Username |  |
| Password |  |

# 四、模块划分

## 4.1 总体模块划分

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 模块名称 | 模块项目 | 模块说明 |
| 1 | 消息序列化 | fams-bo | 系统内消息的序列化模块 |
| 2 | 公共包模块 | fams-common | 系统的公共工具模块，封装全系统通用的工具类，如StringUtil，DateUtil等 |
| 3 | FAMS框架模块 | fams-framework | 系统框架所有插件的实现层 |
| 4 | FAMS启动模块 | fams-springboot-starter | 系统框架所有插件的对外暴露层 |

## 4.2 fams-framework模块

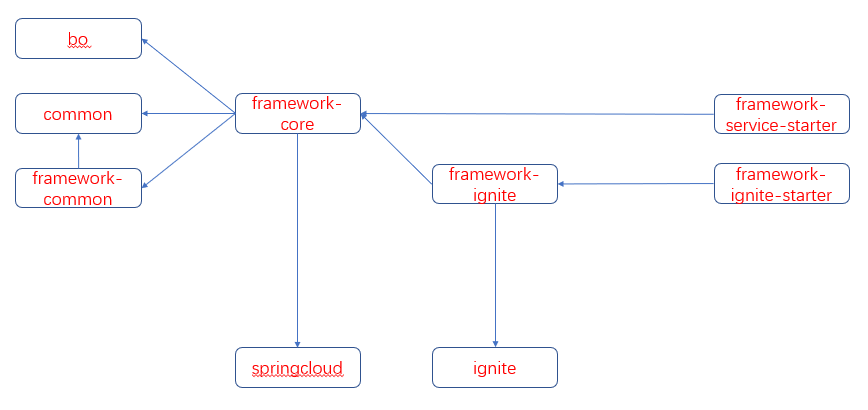
|  |  |  |
| --- | --- | --- |
| 序号 | 模块名称 | 模块说明 |
| 1 | fams-framework-common | Framework模块的公共包，会封装自己的Exception、util以及其他全模块通用的常量 |
| 2 | fams-framework-core | Framework模块的核心包，封装框架所有都需要的插件，比如数据库模块、消息队列模块、缓存队列等 |
| 3 | fams-framework-ignite | 封装ignite相关业务的实现，FAMS使用ignite作为分布式缓存、流式数据处理（交易和风控）、机器学习（策略）、大数据处理（回测框架） |
| 4 | fams-framework-service | 集成框架用于治理系统的微服务：如注册中心，权限中心等 |

## 4.3 fams-springboot-starter模块

|  |  |  |
| --- | --- | --- |
| 序号 | 模块名称 | 模块说明 |
| 1 | fams-framework-ignite-springboot-starter | Framework-ignite模块的启动包，封装  @EnableIgnite：启用ignite节点  @EnalbeIgniteCache:ignite节点启用cache |
| 2 | fams-framework-service-springboot-starter | 框架服务相关启动包  @EnableFAMSServerRegister：标识服务为注册中心  @EnableFAMS：标识服务为FAMS类服务  @FAMSEventListener：标识方法为事件接收方 |
| 3 |  |  |

# 五、模块依赖

截止到2020.02.26，service-starter不再依赖ignite-starter



# 六、服务治理微服务

本模块提供的服务，是FAMS平台的系统级微服务，用于支持整个平台的正常运行

## 6.1 服务注册中心MonitorService

用于管理当前平台中的service，提供负载均衡，服务治理，以及service状态异常的后续处理。一个FAMS系统中，必须存在至少一个active状态的MonitorService。当已存在active状态的MonitorService时，新加入的MonitorService将作为备用服务，暂时不支持同时存在多个active的MonitorService。主要是考虑到当前环境，存储的信息必须要实现多个服务的同步，才能保证MonitorService的正确性。

目前主要功能：

1、服务上线、心跳、状态变化、掉线时，会由MonitorService做服务的治理（已完成）

2、服务下线时，如果所有该服务的实例都已退出，MonitorService会做一系列的善后工作，如删除缓存数据等（开发中）

3、提供其他服务的问询和应答

## 6.2 权限管理中心AuthService

## 6.3 配置管理中心ConfigService

# 七、Framework层对外暴露的Service

## 7.1 Core模块

### 7.1.1 数据加载服务DataLoaderService

根据用户需求，支持将DB数据加载到用户指定的仓库，框架实现了redis或者ignite，用户也可以根据需要自行实现自己的DataLoaderService

|  |  |
| --- | --- |
| 服务名 | DataLoaderService |
| 服务描述 | 该接口未声明任何方法，只是作为一个数据仓库的总接口 |
| 上层服务依赖 | 无 |
| 子类实现 | CacheLoaderService：加载mysql数据到指定缓存（redis或者ignite） |
| 方法列表 |  |

根据用户需求，从db加载数据到redis

|  |  |
| --- | --- |
| 服务名 | RedisCacheLoaderServiceImpl |
| 服务描述 | 该接口未声明任何方法，只是作为一个数据仓库的总接口 |
| 上层服务依赖 | RedisService |
| 子类实现 | 无 |
| 方法列表 | loadData2Cache  loadData2CacheBySelf |

### 7.1.2 全局参数服务GlobalParamService

提供查询全局变量的服务，先查缓存，缓存如果没有的话查数据库

|  |  |
| --- | --- |
| 服务名 | GlobalParamService |
| 服务描述 | 全局变量表的查询和更新服务，全局变量表存储关键的交易日等全局信息 |
| 上层服务依赖 | RedisService |
| 子类实现 | GlobalParamServiceImpl |
| 方法列表 | getParamValueWithCache  updateParamValue |

### 7.1.3 redis服务RedisService

提供redis的各种操作

|  |  |
| --- | --- |
| 服务名 | RedisService |
| 服务描述 | 提供redis操作的各类方法 |
| 上层服务依赖 | 无 |
| 子类实现 | RedisServiceImpl |
| 方法列表 | 见代码 |

### 7.1.4 命令行服务CommondService

提供service的指令服务，用户可以在命令行界面执行内置或者自定义的指令。

|  |  |
| --- | --- |
| 服务名 | CommondService |
| 服务描述 | 不对外直接提供服务，允许二次开发适合用户自定义的指令 |
| 上层服务依赖 | Core-threadPool核心线程池 |
| 子类实现 | CommondServiceImpl |
| 方法列表 | 目前支持的命令见CommondEnum |

## 7.2 Ignite模块

### 7.2.1 数据加载服务

提供将DB数据加载到ignite缓存的服务

### 7.2.2 数据缓存服务

提供缓存服务，供后续的流处理、网格计算做准备