Bettbio系统架构文档

Bettbio目前架构存在的问题和未来架构概述

|  |  |  |
| --- | --- | --- |
| 文件状态  [ √ ] 草稿  [ ] 修改  [ ] 完成 | 文件标识 | 抱团-Project-1683 |
| 当前版本 | 1.0 |
| 作者 | 郭春波 |
| 开始日期 | 2016/8/3 |

修订历史

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 版本 | 编写/修订说明 | 修订人 | 修订日期 | 备注 |
| 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[Bettbio目前系统架构遇到的问题 4](#_Toc458091569)

[数据表结构问题 4](#_Toc458091570)

[平台代码问题 4](#_Toc458091571)

[文件存储问题 4](#_Toc458091572)

[服务器问题 4](#_Toc458091573)

[Bettbio系统架构的延伸 5](#_Toc458091574)

[业务拆分、模块部署 5](#_Toc458091575)

[业务通信、消息系统 5](#_Toc458091576)

[性能优化、缓存系统 5](#_Toc458091577)

[系统监控、安全可控 5](#_Toc458091578)

[Bettbio目前的架构以及部署 6](#_Toc458091579)

[Bettbio系统重构后的部分架构及部署 7](#_Toc458091580)

[Bettbio应用架构的推进 8](#_Toc458091581)

# Bettbio目前系统架构遇到的问题

## 数据表结构问题

1. 数据库表设计没有文档
2. 数据表结构和表字段缺少详细说明

## 平台代码问题

1. 代码逻辑不清晰、处理过于复杂，产生大量代码冗余
2. 代码过于冗长，结构不合理，存在一个文件上千行代码
3. 代码风格不统一，没有严格统一使用共用处理方法等，如有变动需要多处变动
4. 控制层与业务层没有做到明显分离，
5. 实体Bean、业务Bean、视图Bean以及其他转换Bean、视图封装Bean过多以及之间的转换方式不合理，降低系统效率
6. 前端代码没有良好的分离，页面与CSS样式，页面与JS，JS与CSS混合编程，没有良好的复用，如需修改，则多处修改，且修改难度复杂

## 文件存储问题

1. 图片文件等采用流的方式存放和读取，频繁的IO操作会造成网络性能的损耗和服务器IO资源的损耗
2. 随着业务的增长，图片以及文件数量与日剧增，采用单独的文件存储服务器以及单独的二级域名访问，而当前系统满足不了这一需求

## 服务器问题

1. 服务器单点问题，目前PC和移动端由一台Tomcat支撑，当业务过大或者网络原因或者其他原因造成宕机，会造成业务的中断与停滞
2. 数据库的单点问题，目前数据的写入和读取由一台Mysql数据库支撑，没有采用读写分离，大量的产品导入对数据库压力过大

# Bettbio系统架构的延伸

* 1. 随着业务的增加，以及原有业务的扩展，业务体系逐渐庞大，系统也日渐复杂，变得难以维护和扩展，业务的拆分、协同以及平台系统的性能优化变得迫在眉睫

## 业务拆分、模块部署

1. 平台入口 核心搜索模块单独部署 在所有业务功能发布保证平台核心搜索功能可用
2. 订单模块 不受其他模块影响保持订单模块可供用户使用，等等
3. 模块单独部署以及模块的集群有利于平台系统用户体验和平台系统健壮性

## 业务通信、消息系统

* 1. 随着业务的拆分，业务之间的协同处理是系统解决的首要问题

1. 用户、商家的注册、找回密码、以及商家产品的发布等需要发送邮件给用户，用户需要等待整个流程的完成才会得到系统的反馈，如用户注册成功需要等待发送邮件成功才会得到系统的反馈，若邮件发送失败，系统给予的错误反馈是不正确的
2. 产品的添加和产品索引的创建，可采用消息系统，实现异步处理，提高业务的处理等

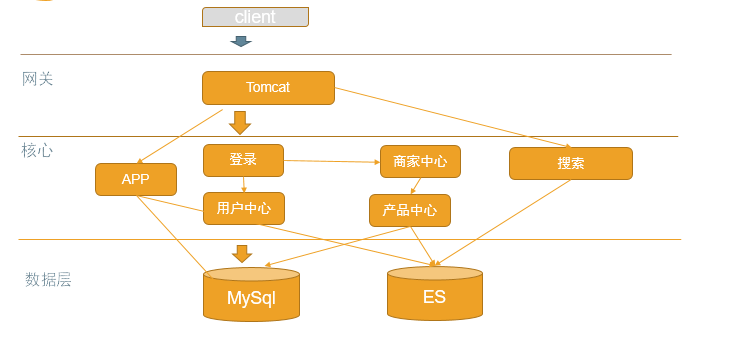
## 性能优化、缓存系统

1. 增加缓存机制，形成缓存系统，把系统常用的数据，数据查询放入如缓存，可降低数据库操作的频繁，提高数据库性能

## 系统监控、安全可控

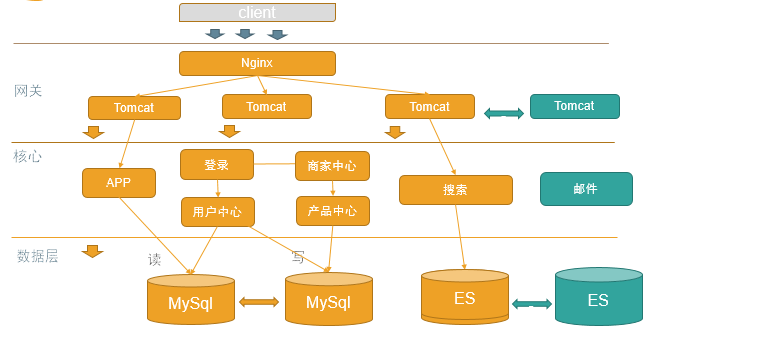
1. 建立系统监控平台，实时监控系统性能并发出预警
2. 日志采集，数据挖掘，以及日志分析，数据分析等

# Bettbio目前的架构以及部署

* 1. 

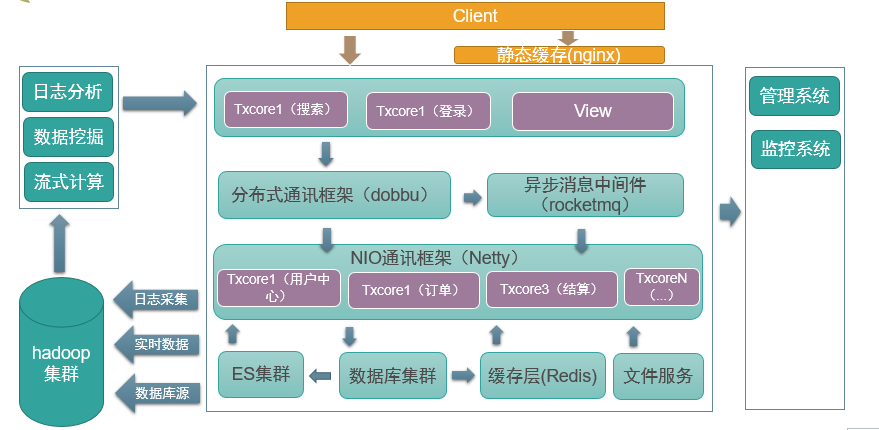
由上图可以看出，当前PC和APP共同由一台Tomcat、Mysql、Es共同支撑，当任意一个节点压力过大而产生服务宕机则导致整个业务链停滞

# Bettbio系统重构后的部分架构及部署

* 1. 

1. 根据业务拆分，单独部署，以多台tomcat分别部署不同业务，每个业务也可以根据访问量做横向集群扩展，添加消息系统，采用异步方式提高业务之间的处理效率
2. 数据库的读写分离，降低数据库压力，提高数据库性能

# Bettbio应用架构的推进

* 1. 

1. 引入动态缓存系统，将热数据和常用数据缓存起来，减少数据库的交互，提供系统的运行速度和降低数据库的负荷
2. 引入消息系统将其他无关与主体业务的操作剥离开来，提供主体业务的速度和及时性
3. 引入Nginx静态缓存，提供静态页面的读取能力
4. 引入新的数据库连接池，提高数据库的读写能力
5. 建立完善的预警，监控和管理系统
6. 围绕核心数据，搭建数据仓库，建立日志收集系统、实时流式计算、离线处理。搭建数据中心，建立大数据生态