# 重构文档

## 系统重构的原因

* 数据库缺乏文档，表结构关系复杂，数据结构不易理解，
* 代码层级太多，代码的阅读，理解以及扩展比较耗费时间
* 前端页面较复杂，修改和维护较难
* 图片存储采用infinispan，图片显示采用流输出到页面方式，耗费IO资源，不好维护（个人理解）

## 重构之后需要解决的问题

* 数据结构设计问题
* 根据新数据结构整体功能开发问题
* 数据迁移 由于表结构不同需要数据整体迁移
* 文件迁移 文件存储方式不同，老系统存取文件采用流方式，新系统采用传统文件存储方式，需要文件整体转换，图片以及文件采用二级域名访问

## 项目模块划分

* 核心公共模块 sys-core 提供公共模块方法，Mybatis映射接口，Service公共接口，异常定义，页面参数拦截器，通用文件上传，下载、富文本编辑器，邮件发送，异步任务等通用模块
* Elsticsearch搜素模块 sys-core-search 对Elsticsearch操作的封装，直接沿用老系统
* 业务逻辑模块 sys-core-model 主要包含实体，Dao层接口已经Service层逻辑处理，以及Mybatis \*mapper.xml中Sql编写
  + 具体包说明
    - com.bettbio.core.\*.search 搜索相关实体、业务逻辑处理等
    - com.bettbio.core.mongo.\* mongodb相关操作
    - com.bettbio.core.mapper 系统dao层接口定义和\*.xml中的sql映射
    - com.bettbio.core.model.\* 系统业务实体封装以及service层逻辑处理等
* Web模块 sys-shop-web 主要包含系统控制器层，拦截器处理，核心配置，页面模板文件以及静态资源文件等

## 项目技术选型

### 管理

* Maven依赖和项目管理
* Git版本控制

### 后端

* Ioc容器 Spring 4.0.0.RELEASE
* Web框架 Springmvc
* Orm Mybatis
* 安全框架 shiro 未集成 暂时使用拦截器
* 缓存 ehcache
* 数据源 druid
* K-v存储 mongodb 1.8.2.RELEASE
* 日志 slf4j+logback
* Json fastjson 后改为Gson
* 模板视图 beetl http://ibeetl.com/

### 前端

* Jquery
* Bootstrap
* Ztree
* Kindeditor
* jquery blockUI
* bootstrap-datatimepicker
* 等插件

## 开发环境

Jdk1.8

Eclipse 4.5.1

Tomcat 8.0.36

Mysql 5.5

Elasticsearch-rtf-1.6.0

## Git 库

由于搭建git在windows系统，公司内网

## Svn公共代码库 svn://123.59.49.58/

重构代码目录sys-bettbio

安装软件目录 software

文档目录 docs

测试文档目录 docs/test

数据库设计 DB

现Git 和 Svn代码未同步

## 部署问题

采用mvn package –P development/production 未成功 jdk1.8打包失败 采用jdk1.7打包成功无法运行

## 配置信息

### 测试数据库Mysql配置

Ip 123.59.49.58:3306

DbName shop\_db

Username root

Password shop

### 测试库Mongodb配置

Ip 106.75.28.20: 27017

Db shop\_db

Username shop

Password shop

### 测试库Elasticsearch配置

Ip 123.59.49.58:9300

## 测试服务器配置

Ip 106.75.28.20

部署目录

/app/tomcat-test

外包app部署目录

/app/tomcat-app

## Mongodb数据库集合信息字段说明

产品Product

**private** Integer id;//主键

**private** String code;//产品唯一码

**private** Date createDate;//创建时间

**private** Date updateDate;//更新时间

**private** Integer qualityScore;//质量分

**private** Integer isExamine;//确认审核

**private** Integer isEnable;//确认上架

**private** Date enableDate;//上架时间

**private** String storeCode;//所属店铺

**private** SProductClassification productClass;//分类编号

**private** Date closingDate;//截止时间

**private** SProductBrand productBrand;//品牌

**private** String productNameCh;//产品中文名称

**private** String productNameEn;//产品英文名称

**private** String productNo;//货号

**private** String simpleDescription;//简单描述

**private** String detailedDescription;//详细描述

**private** List<String> imgUrls=**new** ArrayList<String>();//产品图片集合

**private** List<String> files=**new** ArrayList<String>();//附件集合

**private** String batchCode;// 批次号

**private** String CasCode;// Cas号

**private** String storageCondition;// 存储条件描述

**private** List<ProductPrice> productPrices = **new** ArrayList<ProductPrice>();// 价格集

**private** List<Literature> literatures = **new** ArrayList<Literature>();// 文献引用集

**private** List<PurchaseVoucher> purchaseVouchers = **new** ArrayList<PurchaseVoucher>();// 购买凭证

**private** List<AuthorityCertification> authorityCertifications = **new** ArrayList<AuthorityCertification>();// 第三方认证

**private** List<ExperimentReport> experimentReports = **new** ArrayList<ExperimentReport>();// 实验报告

**private** String price;// 服务价格private String model;// 型号

**private** String placeOfOrigin;// 产地

**private** String model;//商品型号

购物车

ShoppingCart

**private** String productCode;// 产品编号

**private** String productNameCh;//产品中文名称

**private** String productNameEn;//产品英文名称

**private** String productSpec;// 规格

**private** **int** number;//数量

**private** Double unitPrice;// 单价

**private** String storeName;//商家名称

**private** String storeCode;//商家编号

**private** String productImg;//产品图片