

# OASIS详细设计描述文档

## 更新历史

修改人员	日期	变更内容	版本
田晨江	2020.3.2	创建文档	1.0
陈耿阳、田晨江、赵文祺、孙逸伦	2020.3.7	评审文档	1.1
田晨江	2020.3.31	根据迭代二补充文档	2.0
赵文祺	2020.4.5	加入物理部署图	2.1
陈耿阳、田晨江、赵文祺、孙逸伦	2020.4.5	评审文档	2.2
田晨江	2020.4.5	补充修改	3.0

## 目录

- OASIS详细设计描述文档
  - 更新历史
  - 目录
  - 1. 引言
    - 1.1 编制目的
    - 1.2 参考资料
    - 1.3 词汇表
  - 2. 产品概述
  - 3. 逻辑视图
  - 4. 组合视图
    - 4.1 开发包图
    - 4.2 运行时进程
    - 4.3 物理部署
  - 5. 架构设计
    - 5.1 模块职责
    - 5.2 用户界面层分解
      - 5.2.1 展示层模块的职责

- 5.2.2 展示层模块的接口规范
- 5.3 业务逻辑层分解
  - 5.3.1 职责
  - 5.3.2 接口规范
- 5.4 数据层分解
- 6. 信息视角
  - 6.1 数据持久化对象
  - 6.2 数据库表

## 1. 引言

---

### 1.1 编制目的

本报告详细描述了对OASIS系统的概要设计，达到指导详细设计和开发的目的，同时实现和测试人员及用户的沟通。

本报告面向开发人员、测试人员及最终用户而编写，是了解系统的导航。

### 1.2 参考资料

- (1)IEEE标准
- (2)《需求规格说明文档》
- (3)《软件工程与计算(卷三)》

### 1.3 词汇表

词汇名称	词汇含义	备注
OASIS	学术关系图谱系统	@Data注解表示为该类添加构造器和Set、Get方法

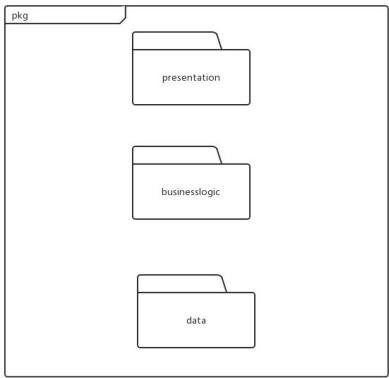
## 2. 产品概述

---

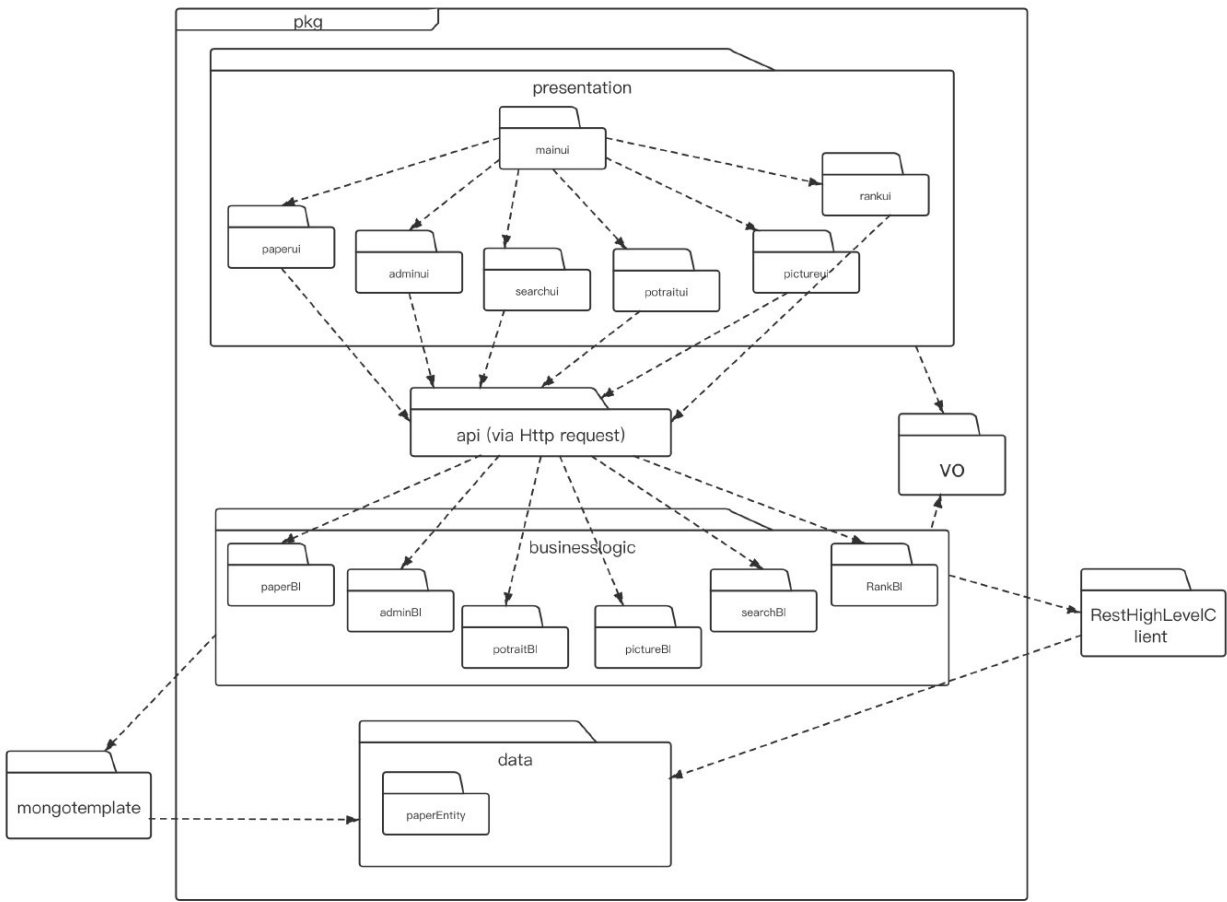
OASIS是一个学术关系图谱系统，可以提供多种应用服务，如学术关系分析、学术同行评价、专家推荐系统、学术机构评价、科研行为画像等服务。

### 3. 逻辑视图

OASIS系统中，选择了MVC体系结构风格，但因为Web开发的特殊性，以及选用的开发技术（Spring Boot 框架）在配置上的简化，使得在进行逻辑设计时，只需要将系统分为3层（展示层、业务逻辑层、数据层），即可示意整个高层抽象。展示层包含Web页面的实现和用户行为的响应，业务逻辑层包含业务逻辑处理的实现，数据层负责数据的持久化和访问。体系结构的逻辑视角和逻辑设计方案如图1和图2所示。



图一 参照体系结构风格的包图表达逻辑视角



图二 软件体系结构逻辑设计方案

## 4. 组合视图

### 4.1 开发包图

OASIS系统的最终开发包设计如表1所示。

表1 OASIS系统的最终开发包设计

开发包	依赖的其他开发包
mainui	paperui, adminui,searchui,rankui,portraitui,pictureui
paperui	api,vo
adminui	api,vo
searchui	api,vo
rankui	api,vo
portratiui	api,vo
pictureui	api,vo

### 4.2 运行时进程

OASIS系统中，会有多个客户端进程和一个服务器端进程。结合部署图，客户端进程在客户端机器上运行，服务器端进程在服务器端机器上运行。进程图如图4所示。

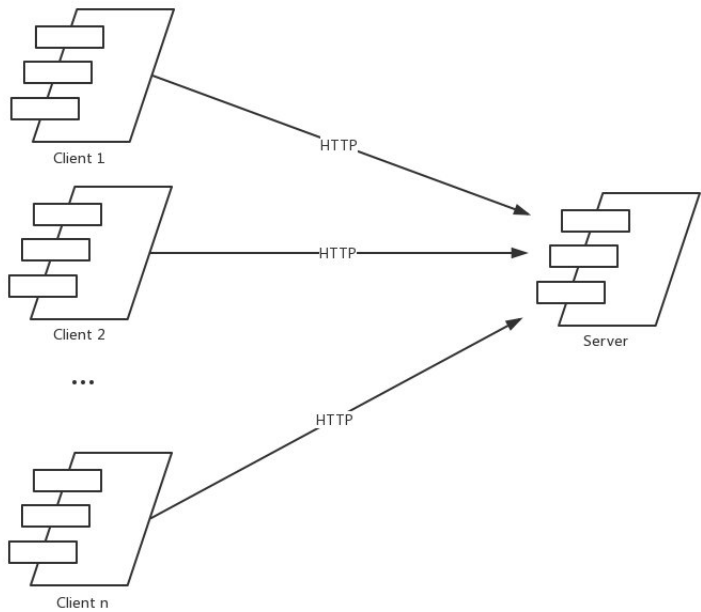


图3 进程图

4.3 物理部署

OASIS系统中，客户端构件是放在客户端机器上的，服务器端构件是放在服务器端机器上的。在客户端节点上，只需要安装现代浏览器（即支持HTML、XHTML、CSS、ECMAScript及W3C DOM标准的浏览器）即可。具体部署图如图5所示。

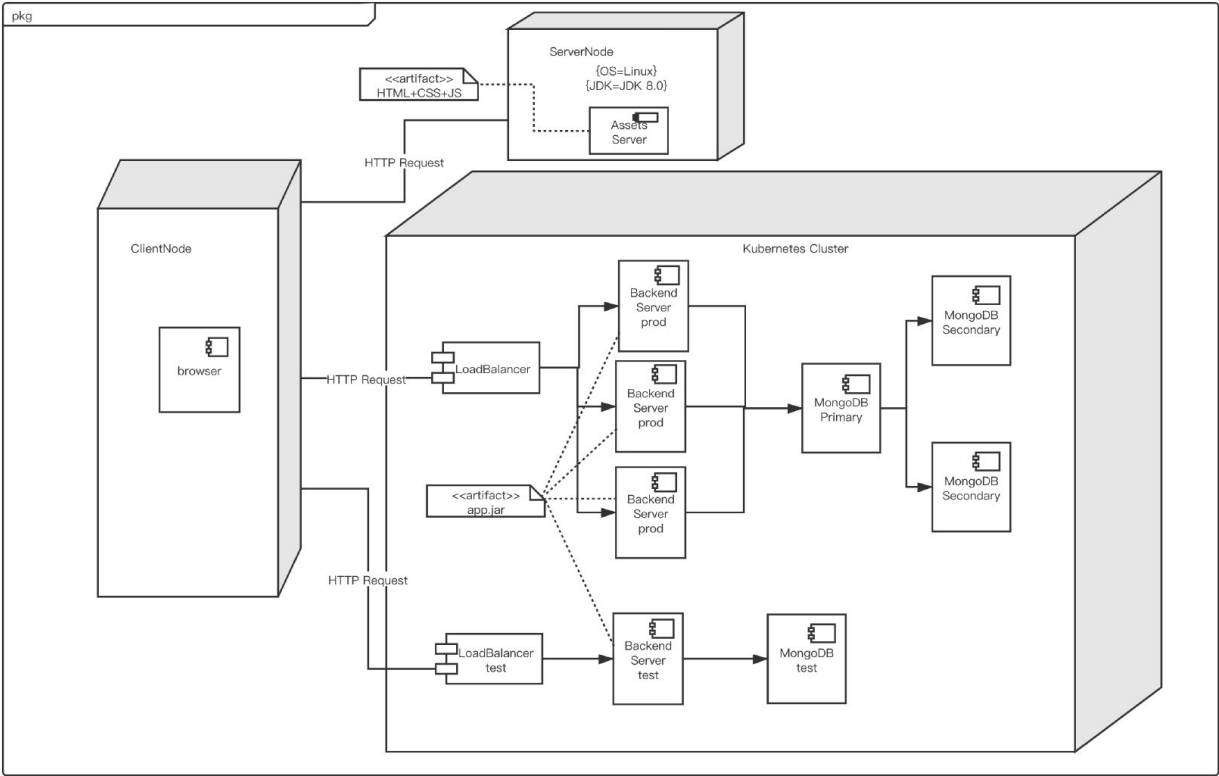


图4 部署图

5. 架构设计

5.1 模块职责

由于Web开发的特殊性，OASIS系统的用户端不需要进行开发，只需用户自行安装浏览器即可，故略过。服务器端模块视图如图6所示。服务器端各层的职责如表2所示。

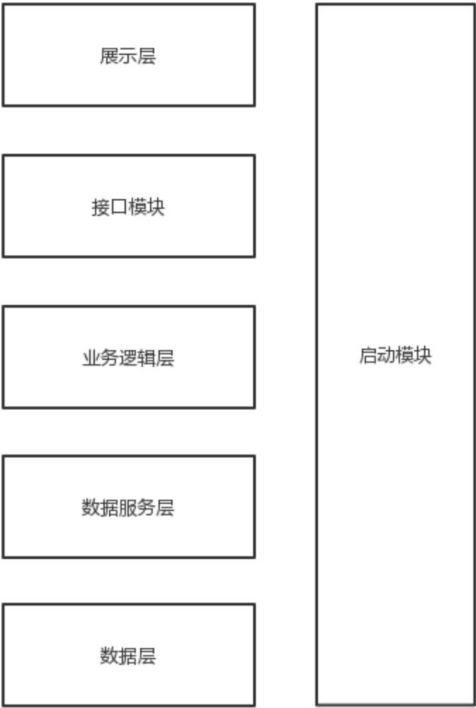


图5 服务器模块视图

表2 服务器端各层的职责

层	职责
启动模块	负责启动系统，初始化网络通信机制和数据服务的连接
展示层	基于Web的OASIS系统的客户端用户界面
接口模块	负责客户端和服务端端的通信及数据传递
业务逻辑层	对用户界面的输入进行响应并执行业务处理逻辑
数据服务层	抽象出的数据操作接口
数据层	负责数据的持久化和访问

每一层只是使用下方直接接触的层。层与层之间仅仅是通过接口的调用来完成的。层之间调用的接口如表3所示。

表3 层之间调用接口

接口	服务调用方	服务提供方
API	服务器端展示层	服务器端接口模块
PaperBLService AdminBLService SearchBLService RankBLService PotraitBLService PictureBLService	服务器端接口模块	服务器端业务逻辑层
mongoTemplate提供的接口 RestHighLevelClient提供的接口	服务器端业务逻辑层	服务器端数据服务层

5.2 用户界面层分解

根据需求，系统存在 3 个用户界面：

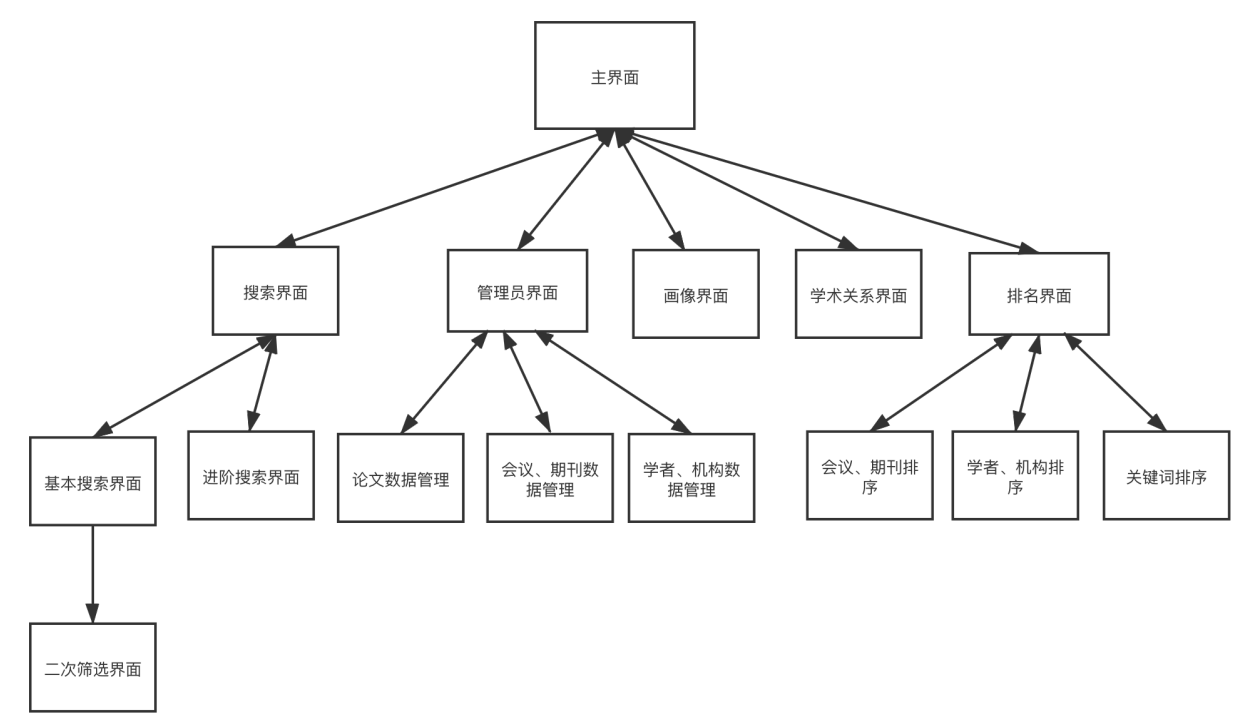


图6 用户界面跳转

5.2.1 展示层模块的职责

展示层模块的职责如表4所示

表4 展示层模块的职责

模块	职责
mainui	负责其他业务界面的显示以及跳转
paperui	负责展示论文相关内容查询界面
adminui	负责管理员对数据进行管理的界面
searchui	负责展示对论文的检索界面
rankui	负责展示对论文的排名界面
potraitui	负责展示各种用户画像的界面
pictureui	负责展示各种关系图的界面

5.2.2 展示层模块的接口规范

展示层 mainui 模块、paperui 模块、adminui 模块、searchui模块、rankui模块、potraitui模块、pictureui的接口规范分别如表5、6、7、8、9、10所示。

表5 paperui 模块的接口规范

需要的接口（需接口）	
服务名	服务
PaperBLService.getResearcherInterest(String id)	获得学者研究兴趣关键词
PaperBLService.getAffiliationInterest(String id)	获得机构研究兴趣关键词
PaperBLService.getJournalInterest(String journal)	获得期刊研究兴趣关键词
PaperBLService.getConferenceInterest(String conference)	获得会议研究兴趣关键词
PaperBLService.getReferenceById(String id)	获得某篇论文的相关引用
PaperBLService.getAuthorPapersById(String authorId, int page, String sortKey)	获得某个作者的全部文章，并按一定标准排序
PaperBLService.getAffiliationPapers(String affiliation, int page, String sortKey)	获得某个机构的全部文章，并按一定标准排序
PaperBLService.getKeywordPapers(String keyword, int page, String sortKey)	获得包含某个关键词的全部文章，并按一定标准排序



表6 adminui 模块的接口规范

需要的接口（需接口）	
服务名	服务
AdminBLService.importPaperData(MultipartFile file)	管理员导入论文数据
AdminBLService.getConferenceInfo(int page, String name)	管理员根据名字搜索相关会议信息
AdminBLService.getAffiliationInfo(int page, String name)	管理员根据名字搜索相关机构信息
AdminBLService.getJournalInfo(int page, String name)	管理员根据名字搜索相关期刊信息
AdminBLService.getAuthorInfo(int page, String name)	管理员根据名字搜索相关作者信息
AdminBLService.mergeAffiliationInfo(List src, String desc)	管理员将几个机构名进行合并
AdminBLService.updateConferenceInfo(String src, String desc)	管理员更新会议信息
AdminBLService.updateJournalInfo(String src, String desc)	管理员更新期刊信息
AdminBLService.updatePaperInfo(UpdatePaperParameter parameter)	管理员更新论文信息
AdminBLService.mergeAuthorInfo(List src, String desc)	管理员将几个作者名进行合并

表7 searchui 模块的接口规范

需要的接口（需接口）	
服务名	服务
SearchBLService.basicSearchByES(String keyword, int page, String sortKey)	对输入的关键词做基本搜索
SearchBLService.advancedSearchByES(String author, String affiliation, String publicationName, String keyword, int startYear, int endYear, int page, String sortKey)	对输入的关键词做进阶搜索

表8 rankui 模块的接口规范

需要的接口（需接口）	
服务名	服务
RankBLService.getAffiliationBasicRanking(String sortKey, String year)	对机构进行录用量/引用数的排名
RankBLService..getAuthorBasicRanking(String sortKey, String year)	对作者进行录用量/引用数的排名
RankBLService.getJournalBasicRanking(String sortKey, int year)	对作者进行录用量/引用数的排名
RankBLService.getConferenceBasicRanking(String sortKey, int year)	对作者进行录用量/引用数的排名
RankBLService.getKeywordBasicRanking(int year)	对关键词做录用量排名
RankBLService.getAuthorAdvancedRanking(String sortKey, int startYear, int endYear)	对作者进阶排名
RankBLService.getAffiliationAdvancedRanking(String sortKey, int startYear, int endYear)	对机构进阶排名
RankBLService.getAffiliationDetailRankingById(String id)	获得某个机构的详细排名信息
RankBLService.getAuthorDetailRanking(String affiliation)	获得某个机构下的作者信息
RankBLService.getAuthorDetailRankingById(String id)	获得某个作者的详细排名信息

表9 potraitui 模块的接口规范

需要的接口（需接口）	
服务名	服务
PotraitBLService.getAuthorPortraitById(String id)	获得某个作者的画像
PotraitBLService.getAffiliationPortrait(String affiliation)	获得某个机构的画像
PotraitBLService.getKeywordPortrait(String keyword)	获得某个关键词的画像
PotraitBLService.getConferencePortrait(String conference)	获得某个会议的画像
PotraitBLService.getJournalPortrait(String journal)	获得某个期刊的画像

表10 pictureui 模块的接口规范

需要的接口（需接口）	
服务名	服务
PictureBLService.getAcademicRelationByAuthorId(String id)	获得某个作者的学术关系图

5.3 业务逻辑层分解

业务逻辑层包含多个针对界面的业务逻辑处理对象

5.3.1 职责

业务逻辑层的职责如表11所示

表11 业务逻辑层职责

模块	职责
PaperBLService	负责实现论文相关的服务
AdminBLService	负责实现管理员所需要的服务
SearchBLService	负责实现关键词搜索的服务
RankBLService	负责实现排名相关的服务
PotraitBLService	负责实现画像相关的服务
PictureBLService	负责实现关系图相关的服务

### 5.3.2 接口规范

业务逻辑层的模块PaperBLService、模块AdminBLService、模块SearchBLService、模块RankBLService、模块PotraitBLService、模块PictureBLService的接口规范如表12、13、14、15、16、17所示

表12 PaperBLService模块的接口规范		
提供的服务（供接口）		
PaperBLService.getResearcherInterest	语法	public BasicResponse getResearcherInterest(String id);
前置条件	无	
后置条件	无	
PaperBLService.getAffiliationInterest	语法	public BasicResponse getAffiliationInterest(String id)
前置条件	无	
后置条件	无	
PaperBLService.getJournalInterest	语法	public BasicResponse getJournalInterest(String id)
前置条件	无	
后置条件	无	
PaperBLService.getConferenceInterest	语法	public BasicResponse getConferenceInterest(String id)
前置条件	无	
后置条件	无	
PaperBLService.getActivePaperAbstract	语法	public BasicResponse getActivePaperAbstract();
前置条件	无	
后置条件	无	
PaperBLService.getReferenceById	语法	public BasicResponse getReferenceById(String paperId);
前置条件	无	
后置条件	无	

PaperBLService.getAuthorPapersById	语法	public BasicResponse getAuthorPapersById(String authorId, int page, String sortKey);
前置条件	无	
后置条件	无	
PaperBLService.getAffiliationPapers	语法	public BasicResponse getAffiliationPapers(String affiliation, int page, String sortKey);
前置条件	无	
后置条件	无	
PaperBLService.getKeywordPapers	语法	public BasicResponse getKeywordPapers(String keyword, int page, String sortKey);
前置条件	无	
后置条件	无	
需要的接口（需接口）		
mongoTemplate.aggregate()	根据聚合条件对 数据集进行聚合 操作	
mongoTemplate.find()	查找符合条件的 结果集	
mongoTemplate.findOne()	查找符合条件的 结果集的第一条	
mongoTemplate.count()	计算符合条件的 结果集大小	

表13 AdminBLService模块的接口规范		
提供的服务（供接口）		
AdminBLService.importPaperData	语法	public BasicResponse importPaperData(MultipartFile file);
前置条件	管理员已被授权	
后置条件	导入数据	
AdminBLService.getConferenceInfo		public BasicResponse getConferenceInfo(int page, String name);

前置条件	管理员已被授权	
后置条件	无	
AdminBLService.getAffiliationInfo		public BasicResponse getAffiliationInfo(int page, String name);
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.getJournalInfo		public BasicResponse getJournalInfo(int page, String name);
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.getAuthorInfo		public BasicResponse getAuthorInfo(int page, String name);
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.mergeAffiliationInfo		public BasicResponse mergeAffiliationInfo(List src, String desc);
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.updateConferenceInfo		public BasicResponse updateConferenceInfo(String src, String desc);
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.updateJournalInfo		public BasicResponse updateJournalInfo(String src, String desc);
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.updatePaperInfo		public BasicResponse updatePaperInfo(UpdatePaperParameter parameter);
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.mergeAuthorInfo		public BasicResponse mergeAuthorInfo(List src, String desc);

前置条件	管理员已被授权	
后置条件	无	
需要的接口（需接口）		
mongoTemplate.aggregate()	根据聚合条件对数据集进行聚合操作	
mongoTemplate.find()	查找符合条件的结果集	
mongoTemplate.findOne()	查找符合条件的结果集的第一条	
mongoTemplate.count()	计算符合条件的结果集大小	

表14 SearchBLSERVICE模块的接口规范		
提供的服务（供接口）		
SearchBLSERVICE.basicSearchByES	语法	public BasicResponse basicSearchByES(String keyword, int page, String sortKey);
前置条件	无	
后置条件	无	
SearchBLSERVICE.advancedSearchByES	语法	public BasicResponse advancedSearchByES(String author, String affiliation, String publicationName, String keyword, int startYear, int endYear, int page, String sortKey);
前置条件	无	
后置条件	无	
SearchBLSERVICE.getBasicSearchFilterCondition	语法	public BasicResponse getBasicSearchFilterCondition(String keyword);
前置条件	无	
后置条件	无	
需要的接口（需接口）		
mongoTemplate.find()	查找满足查询条件的数据	
mongoTemplate.count()	统计满足条件的结果条数	

表15 RankBLSERVICE模块的接口规范		
提供的服务（供接口）		
RankBLSERVICE.getAffiliationBasicRanking	语法	public BasicResponse getAffiliationBasicRanking(String sortKey, String year);
前置条件	无	
后置条件	无	
RankBLSERVICE.getAuthorBasicRanking	语法	public BasicResponse getAuthorBasicRanking(String sortKey, String year);
前置条件	无	



后置条件	无	
RankBLSERVICE.getJournalBasicRanking		public BasicResponse getJournalBasicRanking(String sortKey, int year);
前置条件	无	
后置条件	无	
RankBLSERVICE.getConferenceBasicRanking		public BasicResponse getConferenceBasicRanking(String sortKey, int year);
前置条件	无	
后置条件	无	
RankBLSERVICE.getKeywordBasicRanking		public BasicResponse getKeywordBasicRanking(int year);
前置条件	无	
后置条件	无	
RankBLSERVICE.getAuthorAdvancedRanking		public BasicResponse getAuthorAdvancedRanking(String sortKey, int startYear, int endYear);
前置条件	无	
后置条件	无	
RankBLSERVICE.getAffiliationAdvancedRanking		public BasicResponse getAffiliationAdvancedRanking(String sortKey, int startYear, int endYear);
前置条件	无	
后置条件	无	
RankBLSERVICE.getAffiliationDetailRankingById		public BasicResponse getAffiliationDetailRankingById(String id);
前置条件	无	
后置条件	无	
RankBLSERVICE.getAuthorDetailRanking		public BasicResponse getAuthorDetailRanking(String affiliation);
前置条件	无	
后置条件	无	
RankBLSERVICE.getAuthorDetailRankingById		public BasicResponse AuthorRankDetail getAuthorDetailRankingById(String id);
前置条件	无	

后置条件	无	
需要的接口（需接口）		
mongoTemplate.aggregate()	根据聚合条件对数据集进行聚合操作	
mongoTemplate.find()	查找符合条件的结果集	
mongoTemplate.findOne()	查找符合条件的结果集的第一条	
mongoTemplate.count()	计算符合条件的结果集大小	

表16 PotraitBLService模块的接口规范		
提供的服务（供接口）		
PotraitBLService.getAuthorPortraitById	语法	public BasicResponse getAuthorPortraitById(String id);
前置条件	无	
后置条件	无	
PotraitBLService.getAffiliationPortrait	语法	public BasicResponse getAffiliationPortrait(String affiliation);
前置条件	无	
后置条件	无	
PotraitBLService.getKeywordPortrait	语法	public BasicResponse getKeywordPortrait(String keyword);
前置条件	无	
后置条件	无	
PotraitBLService.getAffiliationPortrait	语法	public BasicResponse getAffiliationPortrait(String conference);
前置条件	无	
后置条件	无	

PotraitBLService.getAffiliationPortrait	语法	public BasicResponse getAffiliationPortrait(String journal);
前置条件	无	
后置条件	无	
需要的接口（需接口）		
mongoTemplate.aggregate()	根据聚合条件对数 据集进行聚合操作	
mongoTemplate.find()	查找符合条件的结 果集	
mongoTemplate.findOne()	查找符合条件的结 果集的第一条	
mongoTemplate.count()	计算符合条件的结 果集大小	

表17 PictureBLService模块的接口规范		
提供的服务（供接口）		
PictureBLService.getAcademicRelationByAuthorId	语法	public BasicResponse getAcademicRelationByAuthorId(String id);
前置条件	无	
后置条件	无	
需要的接口（需接口）		
mongoTemplate.find()	查找符合条件的结果 集	
mongoTemplate.count()	计算符合条件的结果 集大小	

## 5.4 数据层分解

数据层主要给业务逻辑层提供数据访问服务，包括对于持久化数据的增、删、改、查。模块设计和各个模块所提供的服务。这部分Spring中的mongoTemplate已经完成了封装，由BL层直接进行调用。

## 6. 信息视角

## 6.1 数据持久化对象

在此对系统的实体类做简单的介绍：

- PaperEntity: id, 题目, 作者列表, 摘要, 出版题目, doi, 出版年份, 度量指标, 关键词, 引用, 会议名, 链接
- AuthorEntity: id, 名字, 机构名, 姓, 名
- MetricsEntity: 下载量, 被引用次数, 被引用专利次数
- ReferenceEntity: 次序, 谷歌学术链接, 文本, 引用类型, id, 上下文

```
@Data
public class PaperEntity {
    @Id
    private String id;

    private String title;

    private List<AuthorEntity> authors;

    private String _abstract;

    private String publicationTitle;

    private String doi;

    private int publicationYear;

    private MetricsEntity metrics;

    private List<String> keywords;

    private List<ReferenceEntity> references;

    private String contentType;

    private String publicationName;

    private String ieeeId;

    private String link;
}
```

## 6.2 数据库表

MongoDB的papers集合中存储着paper的文档。