

# OASIS体系结构描述文档

## 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
田晨江	2020.6.24	根据迭代三补充文档	4.0

## 目录

- OASIS体系结构描述文档
  - 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
    - 6.1.1 Paper
    - 6.1.2 Recommend
    - 6.1.3 Talents
    - 6.1.4 Task
    - 6.1.5 Trend
  - 6.2 数据库表
    - 6.2.1 MongoDB中实体与数据库表的对应
    - 6.2.2 ElasticSearch中实体与Index的对应

# 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，@Id注解表示该属性是该实体的id

## 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。

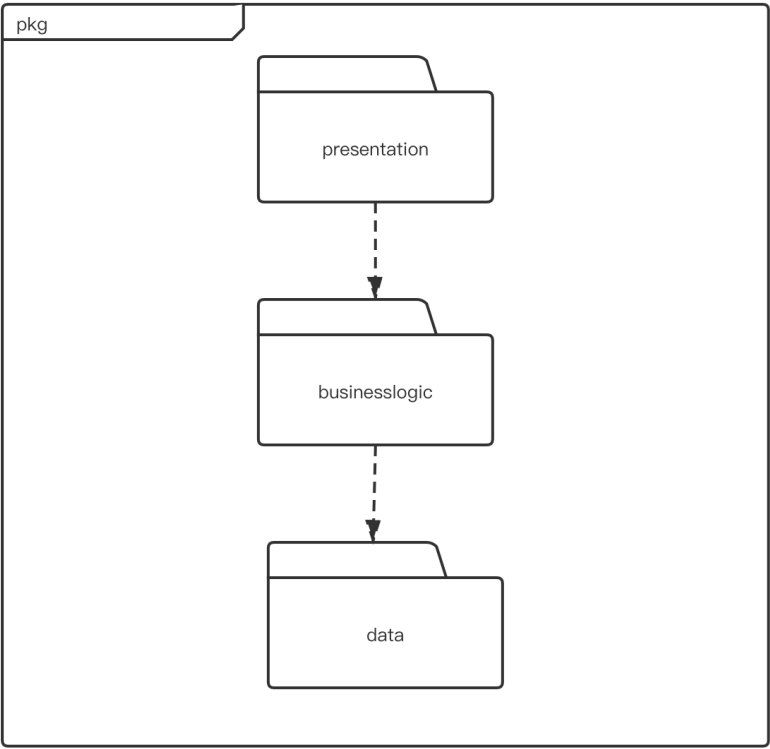


图1 参照体系结构风格的包图表达逻辑视角

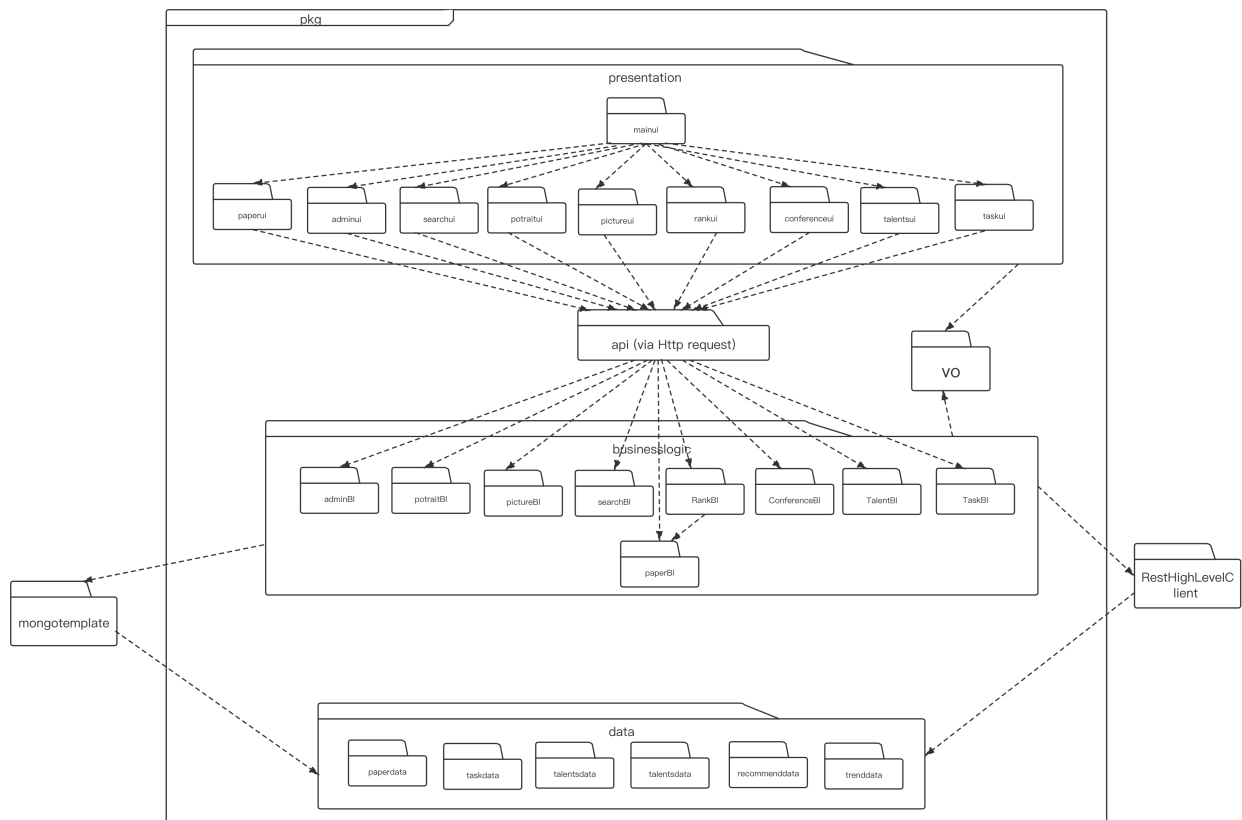


图2 软件体系结构逻辑设计方案

## 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,searchui
portratiui	api,vo
pictureui	api,vo

开发包	依赖的其他开发包
taskui	api,vo
conferenceui	api,vo
talentui	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。

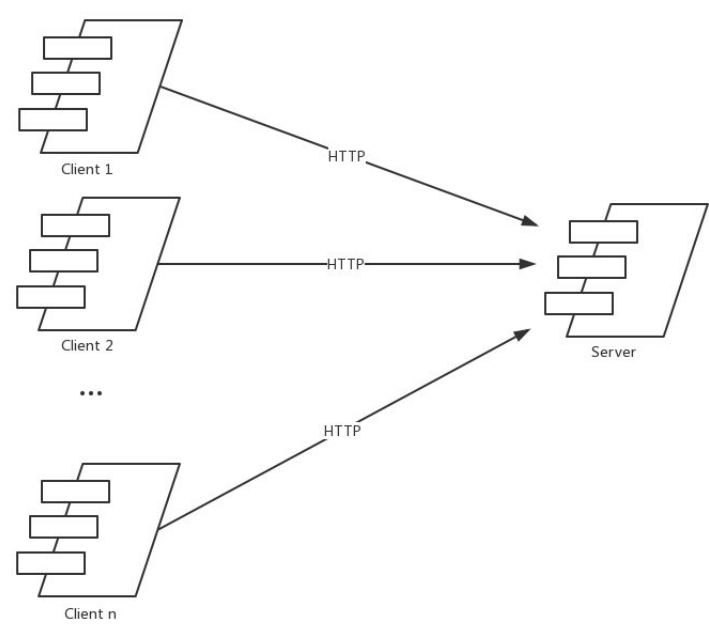


图3 进程图

## 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。

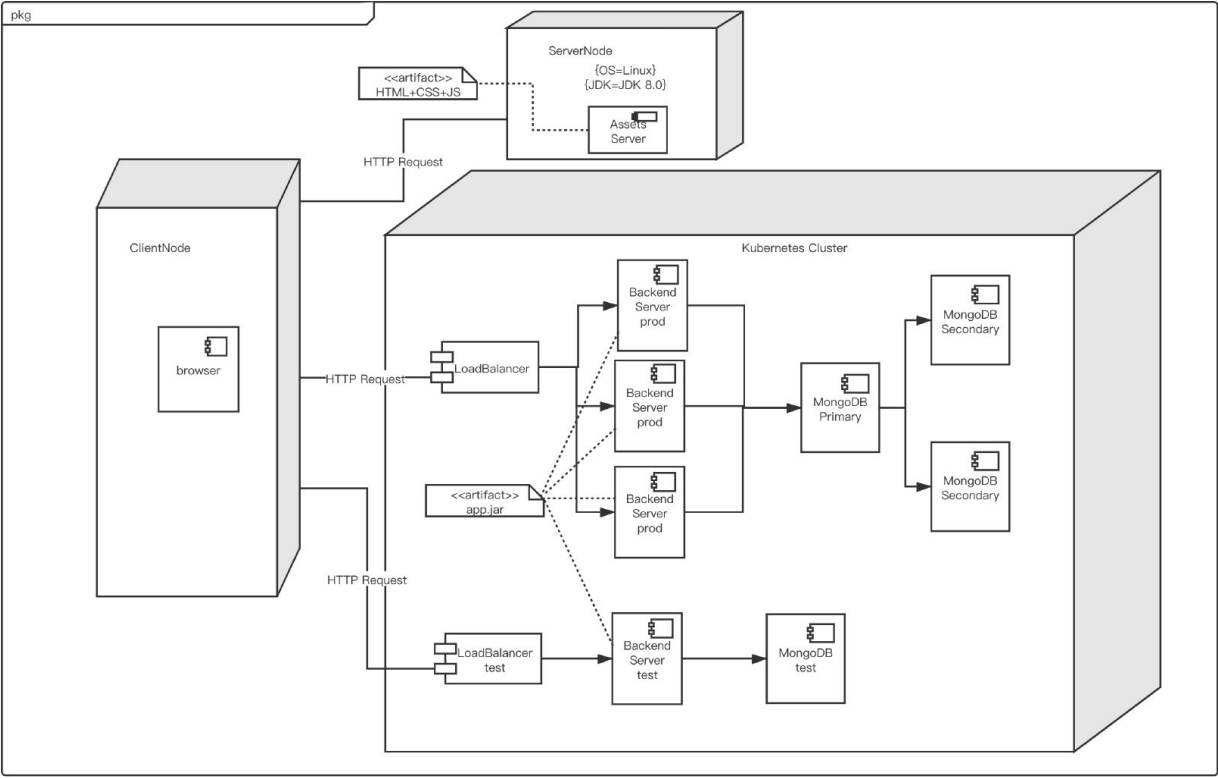


图4 部署图

## 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5所示。服务器端各层的职责如表2所示。

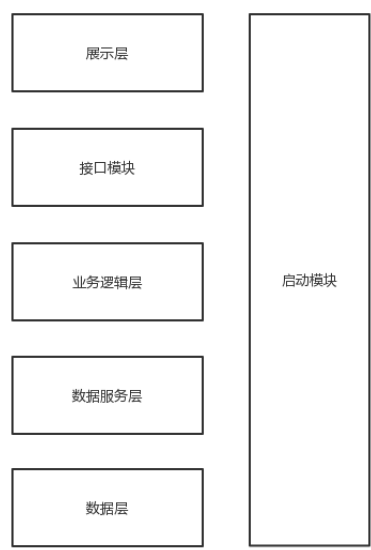


图5 服务器模块视图

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 ConferenceBLService TaskBLService TalentBLService	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 21 个用户界面：

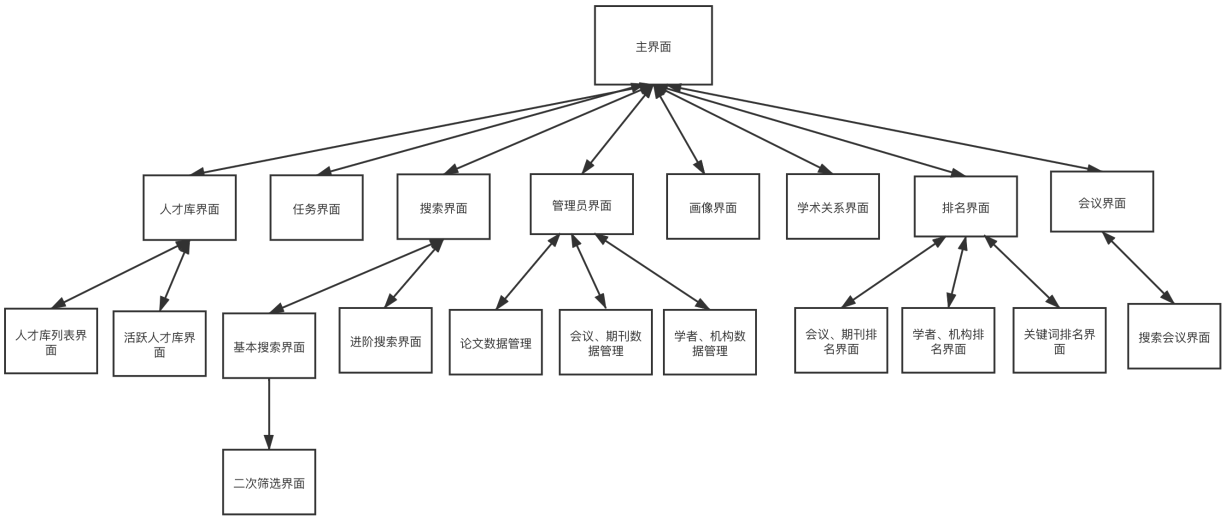


图6 用户界面跳转

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
conferenceui	负责展示会议，期刊的界面
taskui	负责展示爬虫任务的界面
talentui	负责展示人才库的界面

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、conferenceui、taskui、talentui的接口规范分别如表5、6、7、8、9、10、11、12、13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

需要的接口（需接口）	
服务名	服务
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
PaperBLService.getSimilarAuthors(String authorId)	获取某学者的相关学者推荐
PaperBLService.getKeyword3DTrend()	查看研究方向的排名3D趋势

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

需要的接口（需接口）	
服务名	服务
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
AdminBLService.getKeywordInfo(int page, String name)	管理员获取关键词的信息
AdminBLService.getRecommendedSimilarAffiliation()	为管理员推荐合并机构
AdminBLService.getRecommendedSimilarAuthor()	为管理员推荐合并作者

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
SearchBLService.getBasicSearchFilterCondition(String keyword)	获取普通搜索的二次筛选条件
SearchBLService.commandSearch(String query, int page, String sortKey)	对输入的查询语句做命令搜索

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

需要的接口（需接口）	
服务名	服务
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
RankBLService.getKeywordAdvancedRanking(String sortKey, int startYear, int endYear)	获得研究方向排名详情
RankBLService.getAuthorDetailRankingByKeyword(String keyword, String sortKey, int startYear, int endYear)	获取某研究方向下的作者排名详情
RankBLService.getAffiliationDetailRankingByKeyword(String keyword, String sortKey, int startYear, int endYear)	获取某研究方向下的机构排名详情
RankBLService.getKeywordDetailRankingById(String keyword)	根据研究方向获取排名详情

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

表11 conferenceui 模块的接口规范

需要的接口（需接口）	
服务名	服务
ConferenceBLService.getConferencesAndJournalsList(String keyword, int page)	获得包含某个关键词的会议、期刊列表

表12 taskui 模块的接口规范

需要的接口（需接口）	
服务名	服务
TaskBLService.getCrawlTask(String filterKey, Date date)	获得爬虫任务状态

表13 talentui 模块的接口规范

需要的接口（需接口）	
服务名	服务
TalentBIService.getActiveTalentsBase()	获得活跃人才库
TalentBIService.getTalentsListByTalentBase(String field, int page)	获得某人才库的人才列表

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14 业务逻辑层职责

模块	职责
PaperBIService	负责实现论文相关的服务
AdminBIService	负责实现管理员所需要的服务
SearchBIService	负责实现关键词搜索的服务
RankBIService	负责实现排名相关的服务
PotraitBIService	负责实现画像相关的服务
PictureBIService	负责实现关系图相关的服务
ConferenceBIService	负责实现查找会议相关的服务
TaskBIService	负责实现爬虫相关的服务
TalentBIService	负责实现人才库相关的服务

5.3.2 接口规范

业务逻辑层的模块PaperBIService、模块AdminBIService、模块SearchBIService、模块RankBIService、模块PotraitBIService、模块PictureBIService、ConferenceBIService、TaskBIService、TalentBIService的接口规范如表15、16、17、18、19、20、21、22、23所示。

表15 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	
		public BasicResponse

PaperBLService.getAffiliationPapers	语法	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	
PaperBLService.getSimilarAuthors	语法	public BasicResponse getSimilarAuthors(String authorId)
前置条件	无	
后置条件	无	
PaperBLService.getKeyword3DTrend()	语法	public BasicResponse getKeyword3DTrend()
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 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	语法	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	语法	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	语法	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	语法	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	语法	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	语法	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	语法	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	语法	public BasicResponse updatePaperInfo(UpdatePaperParameter parameter);
前置条件	管理员已被授权	
后置条件	无	



AdminBLService.mergeAuthorInfo	语法	public BasicResponse mergeAuthorInfo(List src, String desc);
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.getKeywordInfo(int page, String name)	语法	public BasicResponse getKeywordInfo(int page, String name)
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.getRecommendedSimilarAffiliation	语法	public BasicResponse getRecommendedSimilarAffiliation()
前置条件	管理员已被授权	
后置条件	无	
AdminBLService.getRecommendedSimilarAuthor	语法	public BasicResponse getRecommendedSimilarAuthor()
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17 SearchBLSERVICE模块的接口规范		
提供的服务（供接口）		
SearchBLSERVICE.basicSearchByES	语法	public BasicResponse 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）		
SearchBLSERVICE.commandSearch(String query, int page, String sortKey)	语法	public BasicResponse commandSearch(String query, int page, String sortKey)
前置条件	无	
后置条件	无	
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	
RestHighLevelClient.search()	通过ES进行搜索	

表18 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	语法	public BasicResponse getJournalBasicRanking(String sortKey, int year);
前置条件	无	
后置条件	无	
RankBLService.getConferenceBasicRanking	语法	public BasicResponse getConferenceBasicRanking(String sortKey, int year);
前置条件	无	
后置条件	无	
RankBLService.getKeywordBasicRanking	语法	public BasicResponse getKeywordBasicRanking(int year);
前置条件	无	
后置条件	无	
RankBLService.getAuthorAdvancedRanking	语法	public BasicResponse getAuthorAdvancedRanking(String sortKey, int startYear, int endYear);
前置条件	无	
后置条件	无	
RankBLService.getAffiliationAdvancedRanking	语法	public BasicResponse getAffiliationAdvancedRanking(String sortKey, int startYear, int endYear);
前置条件	无	
后置条件	无	
RankBLService.getAffiliationDetailRankingById	语法	public BasicResponse getAffiliationDetailRankingById(String id);
前置条件	无	
后置条件	无	
RankBLService.getAuthorDetailRanking	语法	public BasicResponse getAuthorDetailRanking(String affiliation);
前置条件	无	
后置条件	无	
		public BasicResponse AuthorRankDetail

RankBLService.getAuthorDetailRankingById	语法	getAuthorDetailRankingById(String id);
前置条件	无	
后置条件	无	
RankBLService.getKeywordAdvancedRanking	语法	public BasicResponse getKeywordAdvancedRanking(String sortKey, int startYear, int endYear)
前置条件	无	
后置条件	无	
RankBLService.getAuthorDetailRankingByKeyword		public BasicResponse getAuthorDetailRankingByKeyword(String keyword, String sortKey, int startYear, int endYear)
前置条件	无	
后置条件	无	
RankBLService.getAffiliationDetailRankingByKeyword	语法	public BasicResponse getAffiliationDetailRankingByKeyword(String keyword, String sortKey, int startYear, int endYear)
前置条件	无	
后置条件	无	
RankBLService.getKeywordDetailRankingById	语法	public BasicResponse getKeywordDetailRankingById(String keyword)
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19 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.getAffiliationPortrait	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20 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	

表21 ConferenceBLService模块的接口规范		
提供的服务（供接口）		
cgetConferencesAndJournalsList	语法	public BasicResponse getConferencesAndJournalsList(String keyword, int page)
前置条件	无	
后置条件	无	
ConferenceBLService.getConferencesAndJournalsProceedings	语法	public BasicResponse getConferencesAndJournalsProceedings(String titleId)
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	

表22 TaskBLService模块的接口规范		
提供的服务（供接口）		
TaskBLService.getCrawlTask	语法	public BasicResponse getCrawlTask(String filterKey, Date date)
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	

表23 TalentBLService模块的接口规范		
提供的服务（供接口）		
TalentBLService.getActiveTalentsBase	语法	public BasicResponse getActiveTalentsBase();
前置条件	无	
后置条件	无	
TalentBLService.getTalentsListByTalentBase	语法	public BasicResponse getTalentsListByTalentBase(String field, int page)
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：

#### 6.1.1 Paper

- 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

```
@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;
}
```

- AuthorEntity: id, 名字, 机构名, 姓, 名



```

@Data
public class AuthorEntity {
    private String id;

    private String name;

    private String affiliation;

    private String firstName;

    private String lastName;
}

```

- MetricsEntity: 下载量, 被引用次数, 被引用专利次数

```

@Data
public class MetricsEntity {
    private int citationCountPaper;

    private int citationCountPatent;

    private int totalDownloads;
}

```

- ReferenceEntity: id, 次序, 文本, 题目, 上下文, googleScholar链接, 其他链接, 引用类型

```

@Data
public class ReferenceEntity {
    private String id;

    private String order;

    private String text;

    private String title;

    private List<ReferenceContextEntity> context;

    private String googleScholarLink;

    private LinkEntity links;

    private String refType;
}

```

### 6.1.2 Recommend

- SimilarAffiliation: id, 相似机构名列表

```

@Data
public class SimilarAffiliation {
    @Id
    private String id;

    private List<String> affiliations;
}

```

- SimilarAuthor: id, 相似作者列表

```

@Data
public class SimilarAuthor implements Serializable {
    @Id
    private String id;

    private List<BriefSimilarAuthor> authors;
}

```

### 6.1.3 Talents

- ExpertsEntity: 作者, 论文列表

```

@Data
public class ExpertsEntity {
    private AuthorEntity author;

    private List<PaperEntity> papers;
}

```

- TalentsEntity: id, 领域, 发表论文数, 专家列表, 领域指数

```

@Data
public class TalentsEntity {
    @Id
    private String id;

    private String field;

    private int count;

    private List<ExpertsEntity> experts;

    private double index;
}

```

### 6.1.4 Task

- ProceedingEntity: 会议/期刊名, 年份, 会议/期刊id, 题目id, 会议/期刊url

```
@Data
public class ProceedingEntity {
    private String proceedingTitle;

    private int year;

    private String proceedingId;

    private String titleId;

    private String proceedingUrl;
}
```

- TaskEntity: 进程列表, 开始时间, 结束时间, 是否已完成, 描述, 已完成爬取的论文数量, 总共要爬取的论文数量

```
@Data
public class TaskEntity {
    private List<ProceedingEntity> proceedings;

    private Date startTime;

    private Date endTime;

    private boolean isFinished;

    private String description;

    private int paperCount;

    private int totalPaperNum;
}
```

### 6.1.5 Trend

- TrendEntity: 包含的年份, 关键词趋势值, 关键词列表

```
@Data
public class TrendEntity {
    List<Integer> years;

    List<List<Integer>> value;

    List<String> keywords;
}
```

## 6.2 数据库表

### 6.2.1 MongoDB中实体与数据库表的对应

实体名	数据库表
Paper	oasis.large
Recommend.SimilarAuthor	oasis.author
Recommend.SimilarAffiliation	oasis.affiliation
Talent	oasis.experts
Task	oasis.task
Trend	oasis.trend

6.2.2 ElasticSearch中实体与Index的对应

实体名	数据库表
PaperDocument	oasis.large