上海双地信息系统有限公司

上海图书馆  
全文OCR等检索功能项目

概要设计方案

目 录

[1. 概述 4](#_Toc24014405)

[1.1. 项目背景 4](#_Toc24014406)

[1.2. 项目目标和内容 5](#_Toc24014407)

[2. 系统架构 6](#_Toc24014408)

[2.1. 服务平台检索架构 6](#_Toc24014411)

[2.1.1. 服务平台整体架构 6](#_Toc24014412)

[2.1.2. 架构分层图 7](#_Toc24014413)

[2.1.3. 整体架构说明 7](#_Toc24014414)

[2.1.4. 未来发展需求支撑说明 10](#_Toc24014415)

[2.2. 数据框架设计 11](#_Toc24014416)

[2.2.1. 业务数据架构 11](#_Toc24014417)

[2.2.2. 数据流程 11](#_Toc24014418)

[2.2.3. 服务平台数据库集群 13](#_Toc24014419)

[2.2.4. 搜索引擎架构 14](#_Toc24014420)

[2.3. 应用框架设计 18](#_Toc24014421)

[2.3.1. 应用架构 18](#_Toc24014422)

[2.3.2. 技术详解 18](#_Toc24014423)

[2.4. 日志框架设计 22](#_Toc24014424)

[2.4.1. 日志架构 22](#_Toc24014425)

[2.4.2. 日志存储 22](#_Toc24014426)

[3. 检索功能设计 24](#_Toc24014427)

[3.1. 普通检索 24](#_Toc24014429)

[3.2. 高级检索 24](#_Toc24014430)

[3.3. 专业检索 24](#_Toc24014431)

[3.4. 检索浏览 25](#_Toc24014432)

[3.5. 简繁检索一致性 25](#_Toc24014433)

[3.6. 文献导航 25](#_Toc24014434)

[4. 整本浏览功能设计 26](#_Toc24014435)

[4.1. 整本浏览 26](#_Toc24014437)

[4.2. 整本检索 27](#_Toc24014438)

[5. 镜像站服务整体设计 28](#_Toc24014439)

[5.1. 整体架构 28](#_Toc24014441)

[5.2. 加密基础库 28](#_Toc24014442)

[5.3. 镜像站安全管理 29](#_Toc24014443)

[5.3.1. 镜像站分布点管理 29](#_Toc24014444)

[5.3.2. 镜像站更新包管理 29](#_Toc24014445)

[5.3.3. 镜像站更新包安全下发工具 29](#_Toc24014446)

[5.3.4. 关于镜像站版本更新 30](#_Toc24014447)

[5.4. 安全镜像站服务 30](#_Toc24014448)

[5.4.1. 镜像站嵌入式WEB应用服务 30](#_Toc24014449)

[5.4.2. 镜像站嵌入式全文索引 33](#_Toc24014450)

[5.4.3. 镜像站嵌入式数据库 33](#_Toc24014451)

[5.4.4. 镜像站嵌入式资源库 34](#_Toc24014452)

[5.4.5. 镜像站更新包接收工具 34](#_Toc24014453)

[5.5. 技术选型和说明 35](#_Toc24014454)

[5.5.1. 安全应用服务 （Lighttpd） 35](#_Toc24014456)

[5.5.2. 安全检索服务（CLucene） 35](#_Toc24014457)

[5.5.3. 安全数据库（Sqlite） 36](#_Toc24014458)

[5.5.4. 安全的资源存储（S-VFS） 36](#_Toc24014459)

[6. 安全设计 37](#_Toc24014460)

[6.1. 安全要素 37](#_Toc24014461)

[6.2. 信息安全 37](#_Toc24014462)

[6.3. 应用安全 38](#_Toc24014463)

[6.3.1. 安全威胁 38](#_Toc24014464)

[6.3.2. 应用安全防火墙 ModSecurity 2.9 39](#_Toc24014465)

[6.4. 系统安全 39](#_Toc24014466)

[6.5. 物理安全 39](#_Toc24014467)

[7. 性能测试设计 40](#_Toc24014468)

[7.1. 测试工具 HP LoadRunner v11.0 40](#_Toc24014469)

[7.2. 性能测试方法 40](#_Toc24014470)

[7.2.1. 性能测试 40](#_Toc24014471)

[7.2.2. 负载测试 40](#_Toc24014472)

[7.2.3. 压力测试 40](#_Toc24014473)

[7.2.4. 稳定性测试 40](#_Toc24014474)

[8. 自动化功能测试设计 41](#_Toc24014475)

[8.1. 测试工具 (HP QTP v11.0) 41](#_Toc24014476)

[8.2. 测试流程 41](#_Toc24014477)

[9. 自动化安全测试 43](#_Toc24014478)

[9.1. 测试工具 43](#_Toc24014479)

[9.2. 测试流程 43](#_Toc24014480)

# 概述

## 项目背景

《全国报刊索引》创立于1955年，是上海图书馆（上海科学技术情报研究所）主管主办并倾力打造的知名信息服务品牌。

上海图书馆的中文报刊文献馆藏在海内外具有举足轻重的地位。1955年，在国家文化部的委托下，上海图书馆创办了《全国报刊索引》杂志，旨在汇聚和揭示全国出版发行的所有报刊文献资源内容，它也成为国内最早出版发行的综合性中文报刊文献检索工具。

六十余年来，《全国报刊索引》已由最初的月刊，发展成为集印刷版与网络服务平台为一体的综合性知识服务体系，现可提供5万余种报刊、5000余万篇文献的一站式服务，年更新数据量超过500万条，相继挖掘出版了《晚清期刊全文数据库》（1833～1911）、《民国时期期刊全文数据库》（1911～1949）、《字林洋行中英文报全文纸数据库》（1850～1951）等珍稀数字资源。

为了给读者用户带来更好的使用体验，网络服务平台历经多次改版升级，逐步优化平台架构、扩充服务功能，努力实现知识服务的全面性、多样性与互动性，为读者用户提供更专业化、规范化、人性化的数字资源服务。凭借珍贵丰富的资源收藏、专业的二次文献组织和深度的知识挖掘，《全国报刊索引》以精致的服务获得了海内外用户的广泛认可，并致力于成为中国近代报刊数字文献资源的权威知识服务体系。

随着资源加工的逐步深入，以中国近代中文期刊全文数据库（简称全文OCR）为代表的精品资源不断加入到服务平台的资源序列中，而目前的服务平台在对这些精品资源的服务呈现上难以发挥精品资源的内在价值，亟需进行服务功能的升级，让这些资源得到充分揭示，满足广大用户的专业服务需求。

## 项目目标和内容

本项目的建设目标是针对全文OCR资源升级服务功能，具体如下：

1） 文献检索增加文章正文内容检索的功能，同时简繁检索结果保持一致；

2） 支持全文OCR精品资源的整本浏览，充分挖掘精品资源的文化价值；

3） 升级镜像站管理工具和前台服务功能，全面支持全文OCR精品资源；

# 系统架构



## 服务平台检索架构

### 服务平台整体架构



### 架构分层图



### 整体架构说明

* 数据层
  + Oracle数据库集群
    - Oracle集群存储了整个服务平台拥有的文献数据和业务数据，Oracle集群规模暂定为两台。
    - 文献数据的来源是基于信息中心的生产数据库导入进来的数据。
  + Solr搜索引擎
    - Solr搜索引擎是对文献数据进行反向索引，提供检索服务，采用SolrCloud集群进行部署，目前暂定采用4台主机组成集群。
  + 资源存储
    - 资源存储是文献资源的存储，同时还有部分上传资源进行存储（会议这块的内容）。
  + 缓存
    - 数据层还有一块为了访问提速而在服务器内存中设置的缓存区域，通过将常用数据进行缓存（再次访问直接从内存读取而不需要从数据库操作），提供系统的响应速度。
    - 缓存采用Memcached进行，主要由于Memcached的特性非常适合进行进一步的扩展，对分布式缓存提供非常好的支持。（暂时用虚线表示本项目内不采用分布式缓存，以后如果有需要提供可以平滑扩展）。
* 服务层
  + 统一数据访问层

统一数据访问层是将所有的对于数据层的数据访问合并为一个功能层次，进行统一的管理和维护。

* + - 关系数据库访问ORM：采用Hibernate作为ORM技术，提供关系数据库访问的功能，主要是对于一些业务数据的处理。
    - Solr Client : SolrCloud检索引擎的访问接口，提供了构建Solr查询语句调用Solr服务器查询功能获取查询结果的公用接口。
    - Storage Access： 存储访问接口，我们提供了支持多路径优化寻址的资源访问，可以通过此接口透明访问需要的文献资源。当前对上传资源也一样支持。为了安全起见，对上传资源和文献资源进行逻辑或者物理上的隔离。
    - Memcached Client：缓存访问客户端，是对缓存区进行访问的接口，可以自由的进行缓存的读取和写入，控制缓存失效机制等等。
  + 通用服务层（或成为基础服务层）

我们定义的通用通用服务层是非业务相关的服务，主要为业务提供一些基础性的服务。主要包含了：

* + - 数据框架管理，对通用数据框架进行管理，提供元数据管理支撑。
    - 规则引擎，提供对数据约束规则的引擎支撑，可以指定数据约束，数据转换中进行约束控制。数据约束就是一种数据严格。
    - 数据导入，提供数据导入功能，可以根据数据框架的定义和规则引擎的支持，进行严格的数据导入。
    - 在线支付，提供通用的银联支付的支撑。
    - 定时任务，服务平台将会由很多的定时或触发任务需要调度
    - 邮件服务，提供邮件发送支持，可以向用户发送邮件
    - 数据检索，通用的数据检索服务，减少业务层数据检索的功能耦合。
* 应用层
  + 业务逻辑层

业务逻辑层是和业务密切相关的处理层，所有视图需要的数据和业务处理都是业务逻辑层进行提供的。

* + - 前台业务

一般检索、高级检索、专业检索、原文索取、文献导航、会议中心、新闻等等

* + - 后台业务

数据管理、用户管理、产品管理、订单管理、会议管理、统计报表等等

* + 视图层
    - Web视图

目前只需要支持Web视图，采用目前流行的Bootstrap+Jquery+Html方式进行呈现，大量采用Ajax技术提升用户体验，部分采用HTML5技术，同时考虑兼容IE8以上的浏览器。

### 未来发展需求支撑说明



严格的分层机制，松散耦合的模块化业务模块、业务管理模块和表现层页面的模块化设计，可以方便的集成新的模块进入整个架构，未来需要扩展支持的积分机制、众包服务、讨论社区都可以通过模块化的开发方式集成到现有的系统中。

## 数据框架设计

### 业务数据架构



图1.1.- 数据库业务架构示意图

### 数据流程

#### 流程A



#### 流程B



### 服务平台数据库集群

#### 数据库集群架构



图1.2.- Oracel RAC架构

#### 硬件环境

|  |  |  |
| --- | --- | --- |
| 名称 | 参数 | 数量 |
| 节点数据库服务器 | CPU：2颗 8核 2.9GHZ  Memory：32G DDR3-1600  NIC：2块千兆网卡 | 2 |
| 交换机 | 8口千兆交换机1台（私有）  24口千兆交换机1台（公共） | 2 |
| 共享数据存储 | 支持共享存储的存储设备，可以使用ISCSI、NAS、FC SAN等等 | 1 |

### 搜索引擎架构

#### 服务平台Solr架构



Solr架构模块简介

* 数据转换传输服务：是之前数据业务框架中的数据中间件；支持从生产数据库提取数据进行转换和传输，同时传输到关系数据库和Solr中。
* Solr索引管理：集成了Solr配置管理、词典集中管理、数据缓冲的功能，集成了Solr DIH功能(全量索引更新、增量索引更新、重载配置等等)
* 工作队列：一个高性能磁盘队列，对发送过来的数据进行缓冲处理，防止直接提交Solr失败。
* QueueReader是一个定时任务，定时从工作队列中将数据读取出来，发送给Solr，加入一些重试处理和失败处理等等操作。

好的SolrCloud 实践

* Schema 或config 通过Zookeeper进行集中管理，能够自动进行配置修改的下发；
* 词典文件配置到Zookeeper中，通过插件支持对字典的监控和重载，实现自动化词典更新，不需要重载Core即可即时生效，带来业务上的方便。
* 数据导入，采用javabin格式提供更加高效的数据导入，放弃json这种仅可读性具备优势的数据格式。采用单线程读、多线程发的方式发挥集群优势。

#### 中文分词

主流中文分词工具如下：

* IK Analyzer 2012 FF
* mmseg4j-Solr-2.2

根据网上的评测，两种分词器的实际分词效率和效果基本接近，主要还是需要看词库的配置。需要进一步测试分词的效率和效果，以及对Solr4.10.2的支持情况，来最终选择使用的版本。

至于其他的一些分词方法，例如庖丁解牛因为早就停止更新所以基本没有使用的必要，不过部分词库可以借鉴使用。

#### 搜索引擎测试标准

* 查全率

查全率是指搜索引擎搜索出的与该查询关键字相关的信息的数量与搜索引擎索引数据库中实际与该查询关键字相关的信息总量的比值。比如该搜索引擎的索引数据库中有100条关于A的信息，而搜索引擎只搜索出其中的90条信息，那么查全率就是90%。

查全率不能以某一次查询的结果作为评价标准，而是以相当多数据的查询结果的平均值作为其查全率。这个比值越大则说明该搜索引擎查询的相关信息就越多，它的使用价值自然就越大。

* 查准率

查准率是指搜索引擎搜索出的结果中实际与该查询关键字相关的信息的数量与搜索引擎搜索出的信息总量的比值。比如该搜索引擎的索引数据库中有100条关于A的信息，而搜索引擎只搜索出90条信息，而这90条信息中只有80条与A有关，那么其查准率为80%。这个比值越大查准率就越高，则说明该搜索引擎搜索信息就越准确。

* 索引更新频率

索引库的更新频率会直接影响到搜索引擎的查全率和查准率。因为信息是变化更新的，因此搜索引擎的索引库也要适时进行更新。首先，新增的索引信息应能及时地出现在索引库中。其次，信息源文档已经不存在的索引要及时删除，否则，将会增加检索的响应时间，以致于影响整个搜索引擎的性能。一般情况下，索引的更新采用两种方法，一种是大批量的索引重建，也叫全量索引更新，另外一种是小批量的索引扩展，也叫增量式索引更新。大批量的索引更新一般是针对数据量在10G以下的信息资料库，可以定期对所有文档进行索引重建。但对于那些经常处于动态更新并且数据量庞大的信息资料库而言，增量式索引更新更适合且效率更高。

* 检索结果排序

搜索引擎对检索结果的排序的好坏也是评价一个搜索引擎优劣与否的一个重要因素。81.7%的用户不会浏览第三页之后的搜索结果，而52.2%的用户只会关注第一页的搜索结果。如何将最重要的和用户最关注的搜索结果放在最靠前的位置就成为了提高搜索引擎性能的一个重要指标。

* 响应时间

搜索引擎的响应时间是指从用户提交搜索请求到搜索引擎返回搜索结果所花的时间，它包括进入搜索引擎的等待时间和得到搜索结果的等待时间。一个搜索引擎无论它的查全率和查准率如何的高，如果其响应时间超出了用户能够忍受的程度，都会被用户弃之不用。因此，响应时间对用户是否选择使用某个搜索引擎显得至关重要。另外，响应时间不仅仅取决于搜索引擎本身的响应速度，在很大程度上还取决于用户的网络设备、网络状况等外部因素。

# 检索功能设计



## 普通检索

普通检索，为了实现对全文OCR资源的正文文本内容的检索支持，我们是通过将正文文本内容加入到检索引擎的All 字段中实现。All字段包括整个检索引擎支持的所有检索字段的内容，因此将正文文本加入到All中，即可在普通检索中检出。

## 高级检索

服务平台的高级检索的资源关联的字段是通过配置高级检索字段来实现，我们为全文OCR资源配置相应的支持高级检索的字段，分别按照正文、图片和广告进行了配置。

另外，在高级检索的产品树上，根据要求加上了全文OCR资源的节点，一旦用户选择了这个节点，那在检索下拉字段中自动添加全文OCR特定的高级检索字段，用户可以使用这些字段进行组合检索。

## 专业检索

专业检索中，为了支持全文OCR资源，我们为需要扩展的检索字段，配置特定代码，这些代码要求唯一性，和其他所有的字段编码都不一样，这样，用户在检索框中输入合法的检索语法，包含这些字段的时候，后台即可按照唯一编码，转换为实际的检索语句，能够支持全文OCR的特定字段的检索。

## 检索浏览

检索结果按照之前的需求分析要求，我们对摘要和全文都进行了高亮呈现，按照不同的呈现类型进行显示。全文高亮需要经过特殊的处理，主要是因为在加工过程中，这些数据包含了一些特殊的标记，需要对这些内容进行过滤。

## 简繁检索一致性

另外，我们将通过权限的Schema设计，从检索引擎的数据定义层面，将简繁检索结果一致性问题进行解决。具体思路是，用户输入的信息全部转换为简体内容，然后基于Schema中预先转换为简体的字段进行检索，显示的时候仍然显示原有的内容，这样可以保障简繁检索结果的一致性。

唯一需要注意的是，因为涉及简繁转换，我们检索的实际是简体字段，高亮的内容也是基于简体内容，需要将高亮信息同步到原始的字段内容，然后对用户呈现，这里基于的原则是按字转换，一个简体字和一个繁体字对应的原则。

## 文献导航

对于全文OCR资源，文献导航是归于近代期刊。为了实现这个目标，就要求将全文OCR资源在文献分类中归于近代期刊下，我们在数据框架上进行这样的定义。

另外，为了实现文献导航的OCR显示和隐藏的功能，通了将通过开关的方式是，来构造检索筛选语句，保障OCR资源按照需要进行显示。

# 整本浏览功能设计

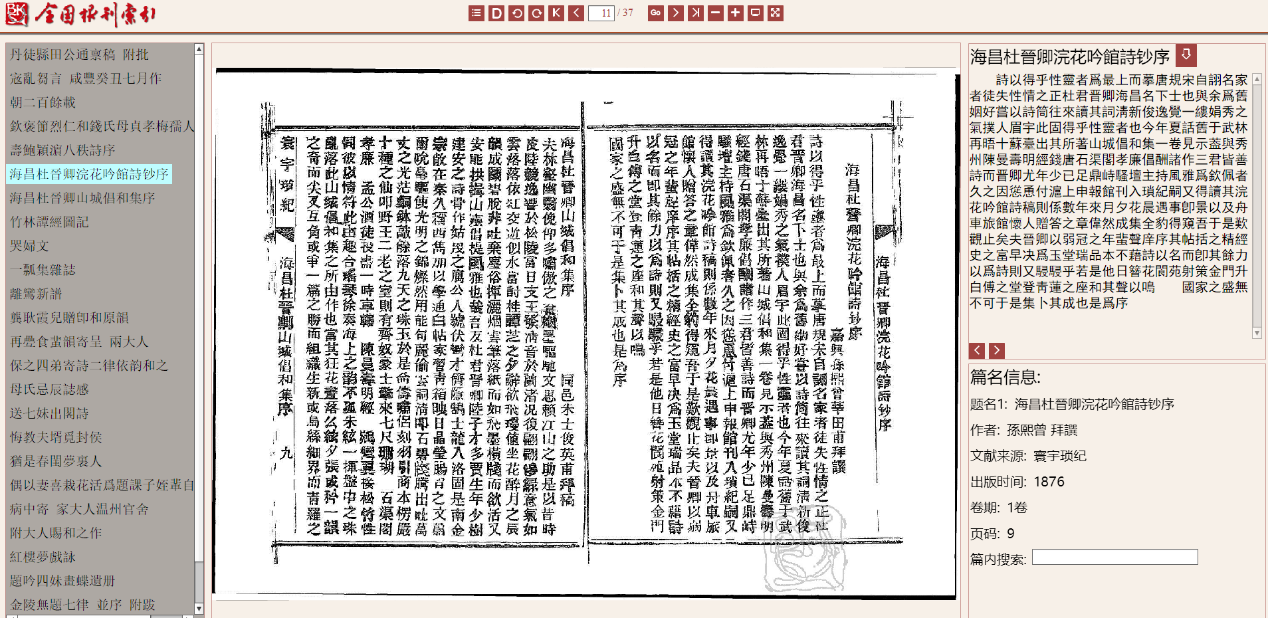


## 整本浏览

全文OCR资源的整本浏览，是比较复杂的，我们必须支持图文并排的呈现，能够按照目录结构进行组织。

目前，整个服务平台的整本浏览主要是基于Flex技术实现，我们同样继承了这个传统，仍然采用Flex技术，按照组件化的设计思想来实现这一功能。在数据的组织上，我们按照全文OCR的资源特点进行整理和整合，充分利用缓存技术，既可减少交互次数，又可减少交互内容，提高整体响应效率。

整个整本浏览的设计如下图所示：



1. 顶部是功能按钮区，提供了需求中定义的各种功能；
2. 左侧是目录去，可以呈现整本刊物的目录结构；
3. 中间是图片区，显示当前页面的图片；
4. 右侧上面是全文区，显示当前选中文章在当前页面的内容；
5. 右侧下面是篇名信息区，提供了篇名的相关信息。

## 整本检索

整本检索，是对单页文本内容提供检索功能，检索匹配的内容红色高亮显示，同时也支持简繁检索。为了实现这一目标，同时减少和服务端的交互次数，我们将文本内容的简体版本和原始版本全部传递到客户端，检索过程中，输入的内容按照简体版本进行检索，匹配项目按照位置同步到原始内容上，保障整个整本检索的正确性和高效性。

# 镜像站服务整体设计



## 镜像站管理

镜像站管理为了支持全文OCR资源，我们将进行如下改造：

1） 升级索引结构，支持全文OCR的索引数据，这个可以通过在服务平台进行数据框架扩展后，通过导出的方式进行Schema输出。

2） 升级缓存工具，全文OCR资源和原有的期刊不一样的地方在于包含了图片和广告数据，这些新的类型也需要加入到缓存中；

3） 将加密算法应用到新的索引和资源转换中，保障新资源的镜像安全。

## 镜像站前端

镜像站前端按照服务平台的功能调整进行代码同步调整，因为镜像站采用的是不同框架，因此只有部分核心逻辑可以重用，框架部分的调整仍然需要按需进行改造。

### 镜像站前端架构调整

镜像站前端管理，主要管理了镜像站的站点分布组成，我们需要为各个镜像站建立独有的安全认证机制，因为，如果采用公共安全认证机制，可能会发生镜像站用户之间的数据资源的非法共享和使用。

我们的镜像站的初始化安装包，需要在镜像站分布点配置完成安全秘钥后，才能生成打包完成，将安全认证过的镜像站安装包发送给镜像站用户，用户按照提示安装到自有的服务器上。镜像站的安装包包含了镜像站所在用户的信息，而且不允许篡改，因此，镜像站的安装包泄露，将可以追查到泄露者的责任。

镜像站分布点的密钥采用多组密钥方式，从机制方面，加大安全破解的难度。

### 镜像站更新包升级

镜像站更新包管理，是对于数据版本发生变化或产品发生修改后，需要下发资源到镜像站用户，我们需要对下发资源进行抽取，然后进行加密的。采用的安全加密密钥就是镜像站分布点管理的密钥组中按照选择规则进行选取。

镜像站更新包包含了数据框架、版本变化的全文索引数据、变化的数据资源等等，以及需要删除的数据脚本，这些内容全部进行加密存储，整个文件包再通过镜像点密钥加密存储。

### 镜像站更新包安全下发工具升级

镜像站更新包安全下发工具，将提供加密链路，提供个镜像站进行远程下载，当然，也支持手动下载，手动更新工具包。

镜像站更新包下发工具，采用的HTTPS协议，SSL加密通讯机制，保障更新包不存在被非法篡改的可能。

更新包内通过目录组织不同的更新数据，程序版本更新放到“program”目录，数据框架放到“framework”目录，索引数据放到“data”目录，资源数据放到“resource”目录，产品数据变化放到“product”目录。