**基于Lucene的分析与应用**

**项目计划**

Version 1.0

小组成员：

刘宏宇

滕延林

顾泽鹏

杨帆

**版本变更记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 变更时间 | 修改人 | 审核人 | 备注 |
| 1.0 | 20160316 | 滕延林 | 刘宏宇 顾泽鹏 杨帆 | 初稿 |
| 1.01 | 20160320 | 滕延林 | 刘宏宇 顾泽鹏 杨帆 | 针对老师提出的问题进行修改 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1. 前言 4](#_Toc445929058)

[1.1目的 4](#_Toc445929059)

[1.2术语与缩略语 4](#_Toc445929060)

[Token 5](#_Toc445929061)

[2. 项目概述 5](#_Toc445929062)

[2.1简介 5](#_Toc445929063)

[2.2索引和搜索 6](#_Toc445929064)

[2.3Lucene 软件包分析 7](#_Toc445929065)

[3. 实施计划 7](#_Toc445929066)

[3.1工作内容分解 7](#_Toc445929067)

[3.2参加人员 8](#_Toc445929068)

[3.3完成项目最后期限 8](#_Toc445929069)

[3.4本计划的批准者与批准日期 8](#_Toc445929070)

[3.5项目分工 8](#_Toc445929071)

[3.6组织形式 8](#_Toc445929072)

[3.7交付的产品 8](#_Toc445929073)

[3.8进度安排 9](#_Toc445929074)

[3.9时间预算 9](#_Toc445929075)

[3.10遵循标准 9](#_Toc445929076)

[3.11硬件环境 9](#_Toc445929077)

[3.12软件环境 10](#_Toc445929078)

[4. 参考资料 10](#_Toc445929079)

# 前言

### 1.1目的

为了便于协调组内成员进行后期的工作，对项目进行跟踪和监控，对任务的进度进行安排与调控，故对后期工作进行计划。

### 1.2术语与缩略语

|  |  |
| --- | --- |
| Analyzer | 它负责把一个字符串按某种规则划分成一个个词语，并去除其中的无效词语 |
| Document | Document类似于数据数据库存储时的Record（一行记录）的概念，它是建立索引时提供的一条记录（包含多个字段），Lucene创建索引时就是把一条记录以Document的形式存储在索引文件中的，当用户进行搜索时，结果数据也是以Document列表的形式返回的。 |
| Field | 一个Document可以包含多个信息域，例如一篇文章可以包含“标题”、“正文”、“最后修改时间”等信息域，这些信息域就是通过Field在Document中存储的。有点类似于数据库中某一个字段。 |
| Term | Term是搜索的最小单位，它表示文档的一个词语，term由两部分组成：它表示的词语和这个词语所出现的field。 |
| Token | Token是term的一次出现，它包含trem文本和相应的起止偏移，以及一个类型字符串。一句话中可以出现多次相同的词语，它们都用同一个term表示，但是用不同的tocken，每个tocken标记该词语出现的地方。 |

# 项目概述

### 2.1简介

Lucene 是一个基于 Java 的全文信息检索工具包，它不是一个完整的搜索应用程序，而是为你的应用程序提供索引和搜索功能。Lucene 目前是 Apache Jakarta 家族中的一个开源项目。也是目前最为流行的基于 Java 开源全文检索工具包。

目前已经有很多应用程序的搜索功能是基于 Lucene 的，比如 Eclipse 的帮助系统的搜索功能。Lucene 能够为文本类型的数据建立索引，所以你只要能把你要索引的数据格式转化的文本的，Lucene 就能对你的文档进行索引和搜索。比如你要对一些 HTML 文档，PDF 文档进行索引的话你就首先需要把 HTML 文档和 PDF 文档转化成文本格式的，然后将转化后的内容交给 Lucene 进行索引，然后把创建好的索引文件保存到磁盘或者内存中，最后根据用户输入的查询条件在索引文件上进行查询。不指定要索引的文档的格式也使 Lucene 能够几乎适用于所有的搜索应用程序。

图 1 表示了搜索应用程序和 Lucene 之间的关系，也反映了利用 Lucene 构建搜索应用程序的流程：

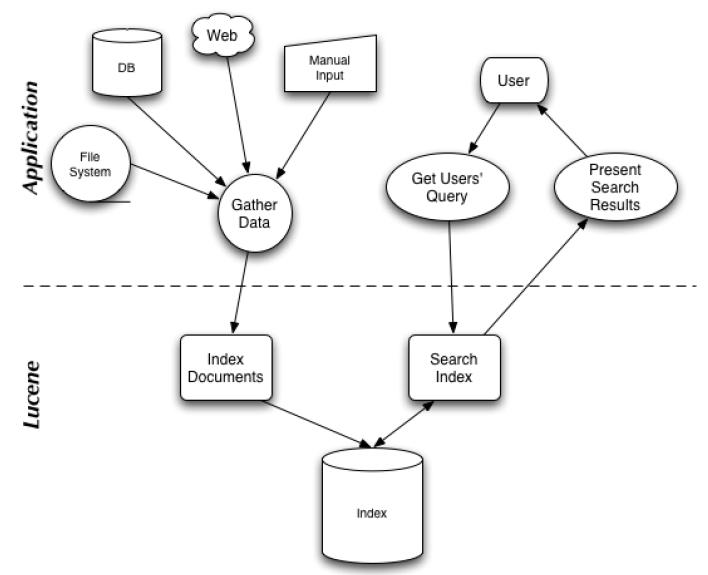


图 1搜索应用程序和 Lucene 之间的关系

### 2.2索引和搜索

索引是现代搜索引擎的核心，建立索引的过程就是把源数据处理成非常方便查询的索引文件的过程。为什么索引这么重要呢，试想你现在要在大量的文档中搜索含有某个关键词的文档，那么如果不建立索引的话你就需要把这些文档顺序的读入内存，然后检查这个文章中是不是含有要查找的关键词，这样的话就会耗费非常多的时间，想想搜索引擎可是在毫秒级的时间内查找出要搜索的结果的。这就是由于建立了索引的原因，你可以把索引想象成这样一种数据结构，他能够使你快速的随机访问存储在索引中的关键词，进而找到该关键词所关联的文档。Lucene 采用的是一种称为倒排索引（inverted index）的机制。倒排索引就是说我们维护了一个词 / 短语表，对于这个表中的每个词 / 短语，都有一个链表描述了有哪些文档包含了这个词 / 短语。这样在用户输入查询条件的时候，就能非常快的得到搜索结果。我们将在本系列文章的第二部分详细介绍 Lucene 的索引机制，由于 Lucene 提供了简单易用的 API，所以即使读者刚开始对全文本进行索引的机制并不太了解，也可以非常容易的使用 Lucene 对你的文档实现索引。

对文档建立好索引后，就可以在这些索引上面进行搜索了。搜索引擎首先会对搜索的关键词进行解析，然后再在建立好的索引上面进行查找，最终返回和用户输入的关键词相关联的文档。

### 2.3Lucene 软件包分析

Lucene 软件包的发布形式是一个 JAR 文件，下面我们分析一下这个 JAR 文件里面的主要的 JAVA 包，使读者对之有个初步的了解。

Package: org.apache.lucene.document这个包提供了一些为封装要索引的文档所需要的类，比如 Document, Field。这样，每一个文档最终被封装成了一个 Document 对象。

Package: org.apache.lucene.analysis这个包主要功能是对文档进行分词，因为文档在建立索引之前必须要进行分词，所以这个包的作用可以看成是为建立索引做准备工作。

Package: org.apache.lucene.index这个包提供了一些类来协助创建索引以及对创建好的索引进行更新。这里面有两个基础的类：IndexWriter 和 IndexReader，其中 IndexWriter 是用来创建索引并添加文档到索引中的，IndexReader 是用来删除索引中的文档的。

Package: org.apache.lucene.search这个包提供了对在建立好的索引上进行搜索所需要的类。比如 IndexSearcher 和 Hits, IndexSearcher 定义了在指定的索引上进行搜索的方法，Hits 用来保存搜索得到的结果。

# 实施计划

### 3.1工作内容分解

首先，了解lucence的工作流程。

然后，利用所需资源，如源代码、开发文档等等，了解lucence的功能需求，对需求进行分析，细致划分，并且找到源代码相应部分。

最后，针对项目探求过程中发现的问题，对lucence进行特定的测试，理想情况下对lucence进行进一步的优化，或者是针对lucence进行上层应用的开发。

### 3.2参加人员

刘宏宇

滕延林

顾泽鹏

杨帆

### 3.3完成项目最后期限

本学期第十七周

### 3.4本计划的批准者与批准日期

刘超老师与任建老师

校历第三周

### 3.5项目分工

首先进行准备工作，对lucence的开发环境进行搭建，对lucence版本进行选取，以及对lucence学习资料进行收集，这些准备工作由滕延林来完成。

项目的计划制定工作由顾泽鹏来完成，包括在Microsoft project上进行计划的安排。

答辩与记录工作由刘宏宇与杨帆进行。之后将lucence分成四部分，具体可见表1。

|  |  |
| --- | --- |
| 姓名 | 任务 |
| 刘宏宇 | Lucene 索引机制部分代码研读 |
| 滕延林 | Lucene 搜索机制部分代码研读 |
| 顾泽鹏 | Lucene 段合并(merge) 机制部分代码研读 |
| 杨帆 | Lucene 的分词器Analyzer部分代码研读 |

根据lucence源代码的阅读情况讨论学习，之后对基于lucence进行上层应用的开发。滕延林与杨帆负责后台开发，顾泽鹏进行前台的开发，刘宏宇负责测试工作。

### 3.6组织形式

a)每周一次会议，讨论当前工作进展以及各自的问题。

b)建微信群，保持实时的交流。

c)资源的共享通过github进行实现。

### 3.7交付的产品

|  |  |  |
| --- | --- | --- |
| **序号** | **产品名称** | **交付时间** |
| 1 | 软件项目计划书 | 第三周 |
| 2 | 需求规格说明书 | 第四周 |
| 3 | 软件评审报告 | 待定 |
| 4 | 软件产品改进与展示 | 待定 |
| 5 | 软件测试分析报告 | 待定 |
| 6 | 软件测试评审报告 | 待定 |
| 7 | 软件进度分析报告 | 待定 |
| 8 | 工作量分析报告 | 待定 |
| 9 | 配置管理 | 待定 |

### 3.8进度安排

### 3.9时间预算

小组成员每周开会两次约2人时，额外的用于本项目的时间约为15小时。项目从第二周开始到第十七周共有16周的时间。总工作量为960人时。

### 3.10遵循标准

本项目遵从以下标准：

GB/T 13702-1992 计算机软件分类与代码

GB/T 19003-2008 软件工程

GB/T 9386-2008 计算机测试文档编制

GB/T 9385-2008 计算机软件需求规格说明

GB/T 5532-2008 计算机软件测试规范

GB/T 8567-2006 计算机软件文档编制规范

### 3.11硬件环境

小组每名成员在实验室都有自己的工作电脑，使用实验室的硬件设备即可进行代码的阅读、开发、测试、ppt的制作演示等等。

### 3.12软件环境

|  |  |
| --- | --- |
| 操作系统 | Windows7及以上 |
| 源码分析 | eclipse |
| 项目管理 | Github |

### 3.13待定问题及解决方案

a) 本小组的工作重点是什么？

对lucence进行优化，或者是针对lucence进行上层应用的开发。首选是尝试对lucence进行优化，改进的方向可以有以下几个方面

(1)索引过程。采用合理的编码方式，节省存储空间。

(2)搜索过程。减低检索时间；改善匹配规则，增加准确性。

(3)段合并。适用于分布式系统。

(4)分词器。更加适用于中文分词。

(5)安全性。对于索引进行加密，校检等。

如果以上的尝试均不成功，将退而求其次，针对lucence进行上层应用的开发，初步想法是从互联网上下载一些文章，搭建文章库，构建索引，对其进行检索。

但是无论哪种，第一步要做的都是对源代码进行阅读。

b) 时间节点的待定问题

时间节点暂时根据上课的安排进行制定。

# 参考资料

[1] <https://lucene.apache.org/>

[2] <http://www.cnblogs.com/forfuture1978/category/300665.html>