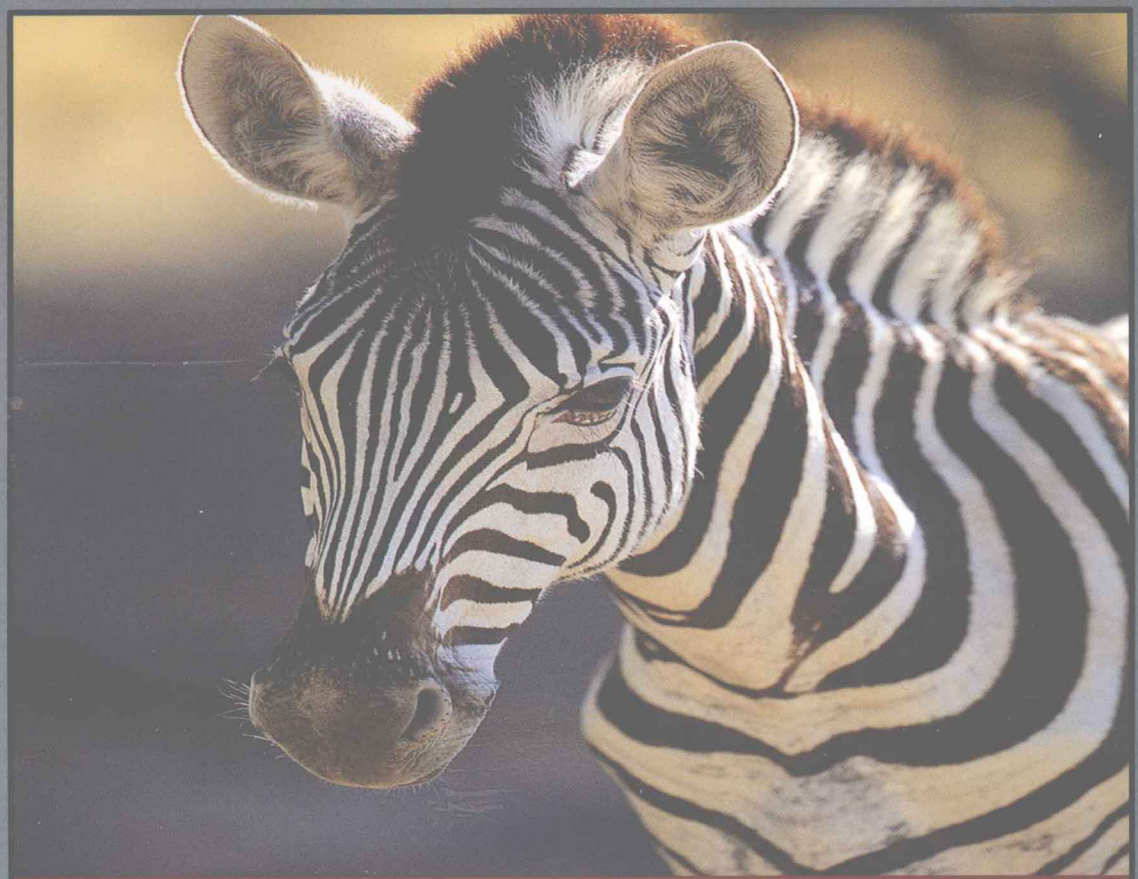
HZ BOOKS

原创
精品系列



Lucene分析与应用

吴众欣　沈家立　编著

机械工业出版社
China Machine Press

原创精品系列

Lucene分析与应用

吴众欣　沈家立　编著

机械工业出版社
China Machine Press

本书对 Lucene 搜索引擎的源代码进行分析讲解，并用一些具体实例把所有源代码进行组织与剖析，完整地展示 Lucene 从建立索引到查询的过程。本书通过介绍 Lucene 的应用，分析 Lucene 具体项目开发的应用环境。最后简单地介绍了 Nutch 和 Hadoop。

本书适用于开发搜索引擎的技术人员、Lucene 爱好者等读者。

图书在版编目（CIP）数据

Lucene 分析与应用/吴众欣，沈家立编著. —北京：机械工业出版社，2008.9

（原创精品系列）

ISBN 978-7-111-24992-4

Ⅰ. L…　Ⅱ. ①吴…　②沈…　Ⅲ. 计算机网络—程序设计　Ⅳ. TP393.09

中国版本图书馆 CIP 数据核字（2008）第 130008 号

机械工业出版社（北京市西城区百万庄大街 22 号　邮政编码　100037）

责任编辑：李东震

北京瑞德印刷有限公司印刷·新华书店北京发行所发行

2008 年 9 月第 1 版第 1 次印刷

186mm×240mm·18 印张

标准书号：ISBN 978-7-111-24992-4

定价：39.00 元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

本社购书热线：（010）68326294

前　言

Google 被人熟知，Baidu 在中国成功推广，“搜索”吸引着 IT 界的眼球，也吸引了更多开发者的好奇心。于是诞生了 Lucene，一个开源的全文检索 API（Application Program Interface，应用程序界面）。并在 Lucene 的基础上，衍生出了一个全文检索引擎（Nutch）和分布式文件系统（Hadoop）。

大家一定很好奇，Google 的搜索引擎是如何工作的？采用什么样的文件系统？提供什么样的服务？……我们无法得知。Lucene 与其相关的项目 Nutch 和 Hadoop 弥补了这个不足，让我们有机会了解到搜索引擎、分布式文件系统的内部工作原理。

如果介绍一个软件或者一套框架如何使用是比较容易的，但是要从源代码剖析内核，却不容易。老吴与家立在写作期间，辗转难眠，思索如何表述才能够准确地把 Lucene 的设计精髓展现给读者。最终确定通过对 Lucene 源代码的解说、辅以图表，并通过一些具体实例把所有源代码进行组织与剖析，完整地展示 Lucene 从建立索引到查询的完整过程。并通过介绍一些 Lucene 的应用，和读者分享 Lucene 在具体项目开发中的应用环境。同时，插入一些 Lucene 开发实例，抛砖引玉，试图让读者也能亲自体会 Lucene 本身的强大功能。最后，为了进一步说明 Lucene 的应用环境，本书简单地介绍了 Nutch 和 Hadoop。

老吴很早就开始研读 Lucene 的源代码，并阅读了 Dong Cutting 的相关论文，对 Lucene 的内核具有深刻的认识。我们很想与大家分享自己的学习体会和研究成果，于是决定把它写出来，家立负责 Lucene 多处应用部分的写作。Lucene 是一个很活跃的开源项目，因为老吴研究得比较早，版本以 1.4.3 为主。为了能够跟上 Lucene 的步伐，家立推荐采用了较新的 1.9 ~ 2.1 版本进行分析。但是该版本的内核变化比较大，因此需要重新分析、调试、总结。为了尽快完成，我们日日熬夜，真所谓痛并快乐着。在此非常感谢家人的支持，朋友的鼓励。

在此，向我的爱妻张信健对我的一贯支持表示感谢！谢谢你，我的爱人！

希望对搜索引擎内核与运行机制感兴趣的朋友阅读此书，由于时间仓促，难免有所疏漏，请读者批评指正。

吴众欣

目　录

第1章　搜索引擎与 Lucene

1.1　搜索引擎与 Lucene 简介

从最初的图书检索到链接查询，对图片、多媒体的搜索，直至现在的人肉搜索，搜索引擎作为信息融合平台将万千世界带到你的周围，让你触手可得，悄悄改变着你的生活，同时也可能将你暴露于众目睽睽之下。有心人可能会考虑它背后的机理，以体味搜索引擎给我们的生活带来的变化。

现今的商业搜索引擎还是 Google 一家独大，微软也提供了 MSN 搜索引擎，但技术与经验积累还不够。而百度则是中文搜索中的佼佼者。在商业搜索引擎中，核心技术与外部世界之间隔着一扇沉重的大门，幸好开源社区常常会将这扇门撬开些许缝隙，让我们能窥得冰山一角。Lucene 与 MG4J 正是开源搜索引擎项目，虽然“代码之前，了无秘密”，但是冰山一角也非轻易窥得。本章我们也是浮光掠影地谈一谈搜索引擎，力图能给大家一些新的信息，以便从多个角度来认识搜索引擎与 Lucene。

1.1.1　搜索引擎分类

搜索引擎可以分成三大类：网络全文搜索引擎、目录索引搜索引擎和元搜索引擎（Meta Search Engine）。下面我们大致介绍一下各类搜索引擎的含义。

1. 网络全文检索引擎

顾名思义，“全文检索”会主动对互联网上的信息进行抓取，对网页的内容进行全文扫描，并对这些内容进行评分，从而形成一个强大的信息索引库。用户输入查询内容后，根据建立的索引返回结果信息。评分高的信息会优先展现在页面中。全文搜索引擎会在一定的时间内主动对索引进行更新，即重新对已有网站的内容进行抓取。这个过程会考虑网站的评分结果，对评分较高的网站会优先抓取并且提高抓取的速度，从而保证内容的实时性。当然，不同的搜索引擎更新的周期是不同的。下面我们列举一些全文搜索引擎。

（1）Google：www. google. com

Google 是目前大家最熟悉、用得最多的搜索引擎了。Google 不仅仅提供文本内容搜索，还提供很多其他搜索服务，例如：图片、地图、资讯、视频、博客……可以说 Google 占据了我们生活中很大一部分。Google 拥有先进的搜索算法，并支持多语言搜索。据统计，Google 每天处理

的搜索请求超过2亿次，并拥有近50亿的数据存储量。Google对数据更新的时间一般是28天，会定期对现有存储的网站进行更新。而且对那些PageRank(网站排名)较高的网站会优先更新。

Google会对用户输入内容进行再次分析，并提示一些新的搜索方案，即所谓的“相关搜索”。对用户输入的一些错别字，Google会进行纠正。例如，输入“Weblogid”，其实是想搜索“Weblogic”，Google会提示“您是不是要找：Weblogic”。这点显得非常智能。

(2) AllTheWeb：http://www.alltheweb.com

由于Google的兴起，AllTheWeb才渐渐地淡出了人们的视野，其实该搜索引擎在几年前是全球最著名的英文搜索引擎之一，拥有的数据存储仅次于Google。如今AllTheWeb已收归于Yahoo!旗下。Yahoo!还发布了该搜索引擎的一个LiveSearch功能，可以通过http://search.yahoo.com/?ek=0进行体验。

(3) Lycos：http://www.lycos.com/

Lycos拥有悠久的历史，是最早提供搜索服务的网站之一。它整合了数据库搜索、在线服务以及其他互联网工具，提供网页、图片和音乐等文件的搜索。Lycos提供的搜索功能比较全面，允许针对网页标题、主题进行搜索，并提供多语言支持。

(4) 百度：http://www.baidu.com

全球最大的中文搜索引擎百度也是我们常用的搜索引擎之一。百度拥有超过10亿的中文网页数据库，每天响应的请求超过数亿次，这些搜索请求来自一百多个不同的国家。和Google一样，除了提供网页搜索以外，百度还提供了其他搜索服务，例如：地图、音乐、文档、影视等搜索服务。作为企业，百度提出了竞价排名的商业模式。

在人性化设计上，百度首先提出了“相关搜索”的概念，这一做法也被Google效仿。另外，百度又效仿了Google，对用户输入的内容进行简单的逻辑判断，并会纠正用户输入的内容，提供一些有效的提示。当用户点击搜索结果时，百度采用新窗口显示相关网站链接(这个功能也被Google采用了)。

(5) 微软Live Search(http://www.live.com)

微软在搜索领域算是新人，但是进步却很快。微软的搜索引擎原来是MSN Search，随着微软对Live服务的推广，全文搜索服务也移植到Live平台。Live Search包括了新的检索方式，可以区分用户在输入搜索条件时，评估用户需要搜索用户的目的。同时，Live Search还对图像搜索做了改进。另外，Live Search也增加了地图搜索功能。

微软在易用性方面一向表现出色。Live Search提供了高级搜索功能，并能够为用户提供订制功能。可以订制搜索结果的显示模式；设置您当前所在的城市(Live Search会根据用户的IP自动识别用户所在的城市)，并能够设定搜索结果的语言。Live Search提供的高级搜索功能和一般的搜索网站不一样。Live Search提供以下几个功能：1)针对搜索条件的设置。用户可以设置搜索结果的逻辑(例如，搜索结果是否需要排除一些内容)；2)用户可以指定在多个站点进行搜索；3)还能搜索链接到指定URL的网页(这个功能很有用。例如，可以查看用户的博客

URL 是否有人引用）。4）针对某种语言的内容进行搜索。不过，Live Search 对搜索结果的显示没有采用新窗口方式。

全文检索的搜索引擎还有很多，如：AltaVista（http://www. altavista. com）等，读者可以亲自去体验一下。

2. 目录索引

“目录索引”搜索引擎拥有一个大而全的目录体系。被搜索的网站必须符合该类搜索引擎的规则才能被这类搜索引擎所“捕获”。如果是商业用户，可以通过付费方式把网站信息加入到该类搜索引擎的索引库中，这类搜索引擎通常采用手工的方式把商业用户的信息加入到它的目录体系中以提供用户检索。下面介绍几个这类搜索引擎。

（1）Yahoo！：http://www. yahoo. com

Yahoo！可以说是最早提出目录搜索的搜索引擎了。Yahoo！的目录分类比较合理，层次比较深，网站摘要很清楚。网站收录的内容非常丰富，检索的结果精确度也很高，并提供相关网页和新闻的查询链接。因为采用人工分类的缘故，Yahoo！搜索的精确度是目前较高的一个搜索引擎。此外，Yahoo！还提供高级检索方式，支持流行的逻辑查询。目前，Yahoo！也提供了全文检索的功能，并采用开源的 hadoop 作为其分布式文件系统。

（2）Open Directory（ODP）：http://www. dmoz. org/

Open Directory 的发展很有意思。ODP 创建于 1998 年，当时加州一位叫做 Rich Skrenta 的程序员为了方便自己的使用，把原来收集的、经常使用的一些资源进行整理，以解决日益膨胀的信息，并和一些网友一起维护，渐渐形成了一个开放的网络共享分类目录。ODP 的发展很快，在 2000 年 4 月，其数据库规模超过了 Yahoo！，成为了目前最大的目录搜索引擎。目前 ODP 由全球的 6 万多名志愿者进行维护管理，并设有近 60 万类的分类目录。ODP 的特点在于，互联网的用户可以直接参与修改、维护。正因为如此，现在很多知名的搜索引擎也使用了 ODP 的数据库，例如 Google。

（3）Ask Jeeves：http://www. ask. com

Ask Jeeves 是一个比较小型的目录搜索引擎网站，而且非常有特色。Ask Jeeves 是一个回答用户提问的自然语言搜索引擎。搜索时，会首先提供 Ask Jeeves 本身数据库含有的答案，然后再提供一些相关网站的链接。

Ask Jeeves 和其他搜索引擎一样，也提供多样化的搜索：blog、地图、视频、图片、新闻等。它在易用性方面非常有特色。它采用 Ajax 技术，在 UI 上给用户全新的操作体验；搜索结果界面采用分块处理：中间部分是搜索结果；左侧是和搜索内容紧密相关的链接；右侧则是该查询条件的不同类型查找结果的头条内容，可能是图片信息，也可能是视频信息。例如，输入查询条件“Lucene”进行查询，查询结果页面的中间是和“Lucene”相关的一些网页信息；查询结果页面的右侧，则出现与“Lucene”相关的“图片”、“新闻”等查询的结果；在查询结果页面的左侧显

示针对“Lucene”更紧密(Narrow)以及相关(Expand)的内容，例如出现：“Jakarta Lucene”，“Nutch”等信息的链接。

这类搜索引擎还有：AOL(http://www.aol.com/)、HotBot(http://www.hotbot.com/)等等，读者可以亲自去体验一下。

3. 元搜索引擎

“元搜索引擎”又叫做“后搜索引擎”，也可以叫做“聚类搜索引擎”，因为这类搜索引擎是建立在前两种搜索引擎之上的，所以这类搜索引擎把用户输入的内容先转到其他搜索引擎上进行搜索，然后整合各个搜索引擎返回的信息进行筛选，把提取后或挖掘出的优质结果提供给用户。下面介绍几个这类搜索引擎。

(1) Dogpile：http://www.dogpile.com/

Dogpile是源搜索引擎的代表，提供良好的性能。用户输入的搜索请求会发布到20多个不同的搜索引擎上进行搜索，这些搜索引擎包括常见的Web页面搜索引擎，也有FTP搜索引擎等。

Dogpile的特点在于，它采用分组式查询。即首先并行地调用几个不同的搜索引擎，如果得到结果不够多，就会再次并行地调用另外几个不同地搜索引擎，如此不断下去，直到得到一定数目的结果(例如，满足需求的100条结果)。Dogpile也支持通配符、布尔逻辑查询，用户可以使用*、+、-等符号进行组合查询。从搜索的结果来看，Dogpile会优先考虑在目前流行的搜索引擎网站上查询，可以在搜索的结果中看到Dogpile采用其他搜索引擎的信息，即该结果来源于哪个搜索引擎。正因为如此，Dogpile的查询速度不是很快，这是Dogpile的一个缺点。

(2) Bbmao：http://www.bbmao.com/

Bbmao是一个优秀的国产的元搜索引擎，它同Dogpile一样，也依赖目前常用的搜索引擎进行二次整理。搜索的结果不仅仅只提供相关内容的链接、链接来自哪个搜索引擎网站，还会根据搜索结果的内容进行简单的分类(加入了“目录搜索”的概念)，进一步提供搜索结果的质量。例如，在查询结果页面的左侧，提供一些分类，每个分类提示用户有几个结果，非常的人性化。对搜索结果提供预览功能。Bbmao还提供高级搜索，用户可以订制搜索过程中，去哪些搜索引擎提取数据，并且定制信息会保存下来，保证搜索引擎的个性化。

Bbmao还提供收藏夹功能，使用收藏功能用户需要注册成Bbmao的用户。Bbmao用户的收藏内容也作为资源为普通用户提供服务。也就是说，Bbmao形成了一个社区，让用户相互推荐自己的收藏信息(也是搜索的结果)。如此一来，能够大幅度地提高搜索结果的准确性。

1.1.2 Lucene项目简介

Lucene并不是一个完整的全文检索应用，但却是一个可靠性高、易于扩展的全文检索工具包，可以嵌入到各种应用中，为应用提供全文检索功能。Lucene是Apache的一级项目，而且版

本更新很快，读者可以到 http://lucene.apache.org 上及时了解 Lucene 的进展情况。

Lucene 的开发者 Doug Cutting 是一位资深的全文索引、检索专家。他曾经是 V-Twin 搜索引擎的主要开发者，后在 Excite 担任高级系统架构设计师，目前就职于 Yahoo!，全力负责 Yahoo! 的全文检索以及 Hadoop 的应用。

Lucene 具有如下突出的优点：

- 自定义的索引文件格式，使得 Lucene 可以很容易地嵌入到应用系统中，并部署在不同的平台上。
- 使用 MDA（Model Driven Architecture，模型驱动架构，是由 OMG 提出的软件工程方法）规则设计，使得 Lucene 易于扩展。
- 提供丰富而又强大的查询实现。

如今，Lucene 已经拥有 Perl、Python、C++、.NET Ruby 等语言的移植版本，广泛使用，显示出强大的生命力。

1.1.3 其他搜索引擎开发包介绍

其实，除了 Lucene 以外，其实还有其他开源的搜索引擎开发包。这里列举一些，供读者参考。

1. MG4J

MG4J（Managing Gigabytes for Java）是一个使用 Java 技术开发的全文检索引擎。MG4J 是一个极易使用、高性能、成熟的搜索引擎，提供了最新的一些特性，使用了一些最新的搜索算法。如果感兴趣，可以进入 MG4J 的官方网站做进一步了解：http://mg4j.dsi.unimi.it/。

MG4J 具有以下一些优点。

1）强大的标引过程：支持大规模文档进行持续地分析、索引、查询，返回让用户容易理解、高亮方式显示的相关文档摘要。

2）效率：MG4J 可以轻松应对数千万文件进行索引。

3）多索引间隔语义：当提交一个查询时，MG4J 使用了一个正确率很高的评分器（Scorer），采用了新的算法，在线性时间内可以对用户的查询做出应答。

4）丰富的查询语法：MG4J 提供了丰富的查询语法：短语查询、相似查询、次序合取查询以及多索引查询语句。

5）Virtual fields：MG4J 支持虚拟 Fields，虚拟 Field 包含虚拟文档。例如由链接来指向的文档。

6）灵活的索引：通过 MG4J，可以产生既小又灵活的索引（索引中可以不包含 term 的位置信息或者 term 的简单计数即可）。为了平衡效率和索引规模之间的关系，MG4J 提供了多种压缩编码方式。

7）分布式处理：索引可以分布在不同的区域，可以灵活地合并。

8）多线程处理：MG4J 可以并发索引和查询。

2. WebLucene

WebLucene 是国内车东先生开发的一个基于 Lucene 的搜索引擎。WebLucener 提供一个通用 XML 接口，方便地在 Web 应用中嵌入全文检索功能。其特点如下：

- 提供了 XML 数据输入接口：适合将原有基于各种数据库的数据源导入到全文索引中，保证了数据源的平台无关性。
- 通过了基于 XML 的搜索结果输出：方便通过 XSLT 进行前台的结果显示。

另外，WebLucene 还为传统数据库应用增加了全文检索功能。

- XML 的数据导入：映射 Document 对象。
- XML 的结果输出：映射 Hits 对象。
- 纯 Servlet 应用：数据导入使用 PHP 脚本。

WebLucene 的工作原理大概如图 1.1 所示(该图引用自车东个人网站对 WebLucene 的介绍文章)。

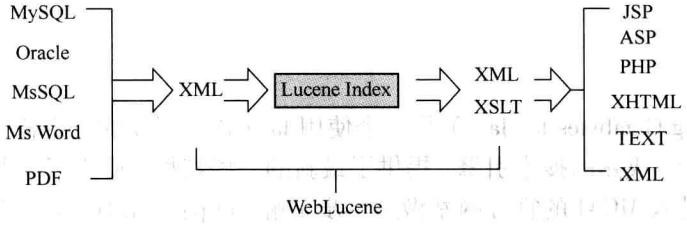


图 1.1 WebLucene 结构示意图

使用过程如下：

1）将数据用脚本导出成 XML 格式；

2）将 XML 数据源导入 Lucene 生成索引；

3）从 Web 界面得到 XML 结果输出，并通过 XSLT 生成 HTML 页面。

要详细了解 WebLucene 的内容，请访问：http://sourceforge.net/projects/weblucene。

当然还有很多开源的搜索引擎，有 Java、C++，python、Ruby、C#等不同语言平台的作品。这里不再一一列举。希望这里介绍的内容能对大家开拓视野有所帮助。

3. Woogle

Woogle 是一个比较特殊的搜索引擎。常用的搜索引擎是抓取页面信息并给用户提供内容检索，而 Woogle 的搜索对象是 Web Service。Woogle 的主要特点有：通过 UDDI 注册节点发布服

务；拥有标准的描述性文档 WSDL；每一个 Web Service 由若干的 operation（操作）组成，并且每个 operation 都拥有输入参数和输出参数（如果对输入输出参数的名字做相似度匹配，输入输出参数类型做可兼容匹配，则可返回用户所需要的服务）。因此，从 Web Service 搜索引擎搜索到的内容与通常的页面搜索引擎不同的是，可以从 Web Service 搜索引擎搜索到 UDDI 上注册的 Web Service 服务；从 WSDL 的描述信息中提取语义；从 Web Service 仓储中进行服务组合；提供查询接口，该接口可以更大限度地扩展 Web Service 的语义关系。

下面是该搜索引擎的一些特性：

- 提供 Web Service 目录浏览功能。
- 提供关键字查询。
- 提供 operation 的输入/输出参数查询。
- 提供在线方式查找。
- 提供被搜索网站的服务状态信息。
- 相似度查询，返回类似功能的 Web Service 服务
- 模板查找。针对给定的输入、输出和功能表述（由特定的输入、输出和功能形成的模板），返回相应的 Web Service 操作。
- 服务组合相似度查找。对于需要查找的功能，返回可以完成该功能的服务组合。

1.2　Lucene 的系统架构

下面先通过一个很简单的例子，来看看 Lucene 建立索引，进行查询的操作过程。

1.2.1　Lucene 最简示例

Lucene 是一个完全使用 Java SDK 开发的全文检索工具，没有使用到第三方的 Java 开发包。因此，我们使用 Lucene 时，只需要把 lucene-core-2.0.0.jar 包引入到工程中就可以了。我们给出一个较有代表性的示例，展示索引文件结构，也可作为后续章节的示例。

例 1.1 EntryLucene.java：使用 Lucene 的最简单示例应用（标引部分）

```
public void buildIndex(String indexDir)throws IOException{
        IndexWriter writer =
                new IndexWriter(indexDir,
                new StandardAnalyzer(), true); ① 创建 Lucene 索引
        writer.setUseCompoundFile(false); ② 非复合式索引模式
        writer.setMergeFactor(2); ③ 设置合并频率的大小为 2

        Document docOne = new Document( ); ④ 建立 Document 对象
        docOne.add(new Field("Title1",
```

```
            "term1 term2 term3",
            Field.Store.YES,
            Field.Index.TOKENIZED)); ⑤ 索引文件内容
writer.addDocument(docOne);
Document docTwo = new Document();
docTwo.add(new Field("Title1",
                     "term2 term3 term2",
                     Field.Store.YES,
                     Field.Index.TOKENIZED));
writer.addDocument(docTwo);
Document docThree = new Document();
docThree.add(new Field("Title1",
                     "term3 term3",
                     Field.Store.YES,
                     Field.Index.TOKENIZED));
writer.addDocument(docThree); ⑥将 Document 对象加到索引中去
Document docFour = new Document();
docFour.add(new Field("Title2",
                     "Hello Lucene!",
                     Field.Store.YES,
                     Field.Index.UN_TOKENIZED));
writer.addDocument(docFour);
Document docFive = new Document();
docFive.add(new Field("Title2",
                     "I like Lucene!",
                     Field.Store.NO,
                     Field.Index.UN_TOKENIZED,
                     Field.TermVector.WITH_POSITIONS_OFFSETS));
writer.addDocument(docFive);
Document docSix = new Document();
docSix.add(new Field("Title3",
                     "Do you want use it?",
                     Field.Store.NO,
                     Field.Index.TOKENIZED,
                     Field.TermVector.YES));
docSix.add(new Field("docName",
                     "docSix",
                     Field.Store.YES,
                     Field.Index.UN_TOKENIZED));
writer.addDocument(docSix);
```

```
        writer.close();⑦ 关闭索引,完成索引建立过程
    }
```

这个示例中，有几处需要特别指出的地方。

② Lucene 的索引形式有两种："复合式索引"和"非复合式索引"。通过 IndexWriter 类的 setUseCompoundFile（boolean）方法进行设置。本例设为"False"表示不使用复合索引（为了展示 Lucene 具体的索引文件）。建立后的索引文件请参见图 1.2；若设为"True"或者默认情况下（不调用该方法），表示开启复合索引功能，建立后的索引文件请参见图 1.3。使用复合式索引的优点是可以减少文件的 IO 操作（《Lucence In Action》一书中有介绍）。

③ 内存中的 Segments 数量达到指定的数量时（Lucene 默认值为 10），会把这些数量的 Segments 合并成一个 Segment。

⑦ 执行关闭索引的 close()方法时，真正完成了将索引文件写入磁盘的操作。

执行上面的代码，能索引目录下看到相关的字典、频率，位置等文件，如图 1.2 所示。

```
_0.tvf          _0.tvx          segments.gen    segments_2      _0.fdt
_0.fdx          _0.fnm          _0.frq          _0.nrm          _0.prx
_0.tii          _0.tis          _0.tvd
```

图 1.2　Lucene 建立索引的文件

如果采用复合索引形式的话，图 1.2 中的那些文件名中带_N 的文件就会聚合成一个 .cfs 文件。图 1.3 就是采用复合索引模式执行的结果。

```
_6.cfs          deletable       segments
```

图 1.3　Lucene 建立复合式索引后的文件

示例中采用非复合式索引形式，表 1.1 对各个文件分别进行了简单介绍。

表 1.1　索引文件含义

索 引 文 件	索引文件含义
.f（n）	规格化文件
.fdt	包含各个域数据（field 的特性）信息
.fdx	它是指向 .fdt 文件的指针。填写的是 .fdt 文件中每个文档的域数据信息的起始位置
.fnm	各个域的名字信息
.frq	词元（term）的频率信息
.prx	term 在文档中的位置信息
.tis	包含 term 数据信息，指向位置文件与频率文件的指针

（续）

索 引 文 件	索引文件含义
. tii	是 . tis 文件的快表，可以快速定位 . tis 文件中 term 数据信息
. tvd	保存有 document 信息，用词元向量（TermVector）方式保存 field 的信息，同时包含一个指针表，表内的指针指向 . tvf 文件中的 field 信息
. tvf	以 TermVector 方式保存 field 数据信息
. tvx	是 . tvd 的索引文件，保存了指向 . tvd 指针信息
deletable	包含要删除的文档信息
Segments_N 与 segments. gen	保存了相关段的信息

生成的这些索引文件常常是成组出现的，如 . fdx 和 . fdt 形成一组，. fdx 文件是 . fdt 文件的索引快表；. tis 和 . tii 又形成一组，. tii 文件是 . tis 文件的索引快表。图 1. 4 大致给出了这些具体文件之间的关系。

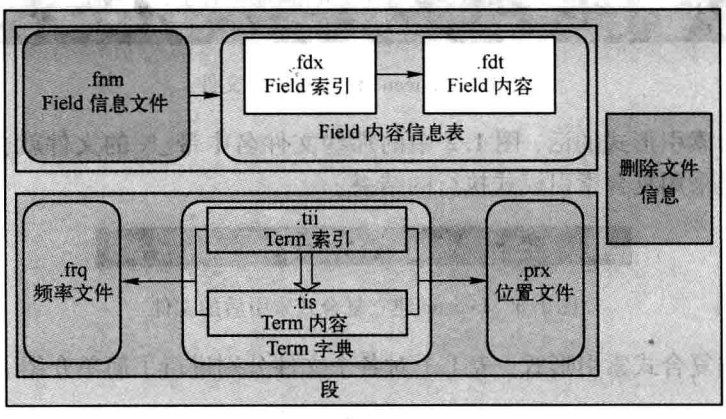


图 1. 4　各个索引文件之间的关系

下面的示例展示如何使用 Lucene 搜索。

我们先看下面的展示 Lucene 查询功能代码，详见如下代码。

例 1. 2 EntryLucene. java：使用 Lucene 的最简单示例应用（查询部分）

```
public void doSearcher(String docName,
                                String keyword, String indexDir) {

    Query query = null; QueryParser queryParser = null;
```