**用 Xapian 和 Omega 读取并索引文档**

安装并部署开源 Xapian 来索引多种信息



Martin C. Brown  
2010 年 11 月 09 日发布

[Weibo](http://service.weibo.com/share/share.php?url=http%3A%2F%2Fwww.ibm.com%2Fdeveloperworks%2Fcn%2Fopensource%2Fos-xapianomega%2Findex.html%25&title=%E7%94%A8%20Xapian%20%E5%92%8C%20Omega%20%E8%AF%BB%E5%8F%96%E5%B9%B6%E7%B4%A2%E5%BC%95%E6%96%87%E6%A1%A3&language=zh_cn)[Google+](https://plus.google.com/share?url=http%3A%2F%2Fwww.ibm.com%2Fdeveloperworks%2Fcn%2Fopensource%2Fos-xapianomega%2Findex.html&t=%E7%94%A8%20Xapian%20%E5%92%8C%20Omega%20%E8%AF%BB%E5%8F%96%E5%B9%B6%E7%B4%A2%E5%BC%95%E6%96%87%E6%A1%A3)[用电子邮件发送本页面](mailto:?subject=%E7%94%A8%20Xapian%20%E5%92%8C%20Omega%20%E8%AF%BB%E5%8F%96%E5%B9%B6%E7%B4%A2%E5%BC%95%E6%96%87%E6%A1%A3&body=http%3A%2F%2Fwww.ibm.com%2Fdeveloperworks%2Fcn%2Fopensource%2Fos-xapianomega%2Findex.html)

[Comments](https://www.ibm.com/developerworks/cn/opensource/os-xapianomega/index.html#icomments)

[0](https://www.ibm.com/developerworks/cn/opensource/os-xapianomega/index.html" \l "icomments)

**Xapian 基础知识**

Xapian 和 Omega 是独立的两个组件，旨在相互协作来提供索引和搜索功能。Xapian 组件提供了核心的数据库功能（存储信息）以及搜索和检索系统来查找单词和单词组合（参见 [参考资料](https://www.ibm.com/developerworks/cn/opensource/os-xapianomega/index.html#artrelatedtopics)）。

Omega 组件提供了一些工具来将不同格式的信息转化并解析成 Xapian 需要的原始格式以便它可以被索引。Omega 使用了多种工具，比如 pdftotext，然后再提交转化并过滤了的基于文本的信息，以便文档的索引和结构可以被标识并存储在 Xapian 数据库。Omega 是 Xapian 下载的一部分。

**安装 Xapian**

**构建环境**

这里提供的操作指南假设环境是类似 Linux® 或其他 UNIX® 的。在一个 Windows 环境中，您可以使用 GCC 与 mingw、cygwin 或 MSVC。

Xapian 组件是作为一个简单的 tar.gz 下载提供的，并可通过使用常规的 configure 和 make 步骤加以构建。首先，通过键入如下代码解压缩这个归档文件：

|  |  |
| --- | --- |
| 1 | $ tar zxf xapian-core-1.2.0.tar.gz |

通过键入 $ cd xapian-core-1.2.0. 更改到此目录。

然后再进行配置，键入：

|  |  |
| --- | --- |
| 1 | $ ./configure |

此代码还可以安装到其他位置。比如，如果想要将 Xapian 组件安装到主目录，可以使用前缀选项：

|  |  |
| --- | --- |
| 1 | $ ./configure --prefix=/home/xapian |

然后使用

|  |  |
| --- | --- |
| 1 | $ make |

进行构建。

最后，安装库和工具。记住您需要 superuser/root 才能将这些组件安装到标准位置（/usr/local/bin and /usr/local/lib）：$ make install。

安装了主要的 Xapian 工具箱后，就可以继续并安装 Omega 组件了。

**安装 Omega**

Omega 的安装步骤类似于 Xapian 核的安装。首先，通过键入 $ tar zxf xapian-omega-1.2.0.tar.gz 进行解压缩。

进入目录：

|  |  |
| --- | --- |
| 1 | $ cd xapian-omega-1.2.0 |

运行 configure：

|  |  |
| --- | --- |
| 1 | $ ./configure |

注意：如果已经将 Xapian 安装在一个特定目录，而不是默认位置，那么需要指定所提供的配置工具的位置以及 Xapian 组件的库的位置：

|  |  |
| --- | --- |
| 1 | $ ./configure --prefix=/home/xapian XAPIAN\_CONFIG=/home/xapian/bin/xapian-config |

最后，构建并安装各种工具，如下所示。

|  |  |
| --- | --- |
| 1  2 | $ ./make  $ ./make install |

这就是全部步骤。之后，您就可以开始索引信息，然后再检索结果了。

**索引数据**

使用 Xapian 的第一个步骤是通过添加某些文档来用信息填充一个数据库。Xapian 数据库使用了一个目录-/URL-风格的寻址方法来区分数据，因此可以将信息组织到不同的位置，这样一来，它就可以搜索整个被索引了的数据库和数据库适当的特定区域。

为了填充数据，可以编写您自己的索引和提交系统来将数据提供给 Xapian 索引做处理。不过，这个过程非常耗时并且在很多情况下，都可能需要索引标准的数据类型，比如 HTML、PDF 等。在使用 Omega 工具转化和索引数据时，Xapian 支持所有这些类型。

omindex 工具可以在一个文件目录内爬行，标识可被索引的文件，然后再将这些文件适当地添加到这个索引。为了创建一个新的索引，必须指定索引的名称、用来标识索引内信息的 URL 以及文件所在目录。比如，可以像这样索引一个目录结构：

|  |  |
| --- | --- |
| 1 | $ omindex --db info --url information /mnt/data0/Information |

这实际上会启动索引过程，加载文件。清单 1 所示的是结果的一个示例（某些地方做了修整），其中显示了这个索引的过程。

**清单 1. 显示了索引过程的示例输出**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | [Entering directory /]  [Entering directory /Manuals]  [Entering directory /Manuals/Amazon]  Indexing "/Manuals/Amazon/prod-adv-api-dg-20091001.pdf" as application/pdf ... added.  [Entering directory /Manuals/Apple]  Indexing "/Manuals/Apple/Leopard\_Server\_OSX.5.pdf" as application/pdf ... added.  Indexing "/Manuals/Apple/Extending\_Your\_Wiki\_Server.pdf" as application/pdf ... added.  ...  [Entering directory /Manuals/OmniGroup]  [Entering directory /Manuals/OmniGroup/OmniPlan]  Indexing "/Manuals/OmniGroup/OmniPlan/OmniPlan-1.0-mini-manual.pdf"    as application/pdf ... added.  Indexing "/Manuals/OmniGroup/OmniPlan-Manual.pdf" as application/pdf ... added.  [Entering directory /Manuals/Asus]  Indexing "/Manuals/Asus/e2968b\_p5n-e sli.pdf" as application/pdf ... added.  [Entering directory /Manuals/Asterisk]  Indexing "/Manuals/Asterisk/Asterisk Handbook.pdf" as application/pdf ... added.  [Entering directory /Manuals/VirtualBox]  Indexing "/Manuals/VirtualBox/VBoxUserManual.pdf" as application/pdf ... added.  [Entering directory /Books]  [Entering directory /Books/Apache Cookbook]  Indexing "/Books/Apache Cookbook/44386-12004-14591-0-596-00191-6-apacheckbk    -CHP-3.PDF" as application/pdf ... added.  ...  Indexing "/Books/TheArtofSEO1stEdition.pdf" as application/pdf ... added.  [Entering directory /Books/Apache Definitive Guide 3ed]  Indexing "/Books/Apache Definitive Guide   3ed/44385-12004-14591-0-596-00203-3-apache3-CHP-10.PDF" as application/pdf ... added.  [Entering directory /Books/IBM]  [Entering directory /Books/IBM/Redbooks]  Indexing "/Books/IBM/Redbooks/sg246622.pdf" as application/pdf ... added.  Indexing "/Books/IBM/Redbooks/sg247186.pdf" as application/pdf ... added. |

在本例中，此目录包含的大多是 PDF 文档，但只要使用了正确的过滤工具，对于 HTML、Microsoft Office、Abiword 等文档也可以获得相同的结果，这些工具必须被安装在您的系统上才能将原始资料转变为 Xapian 能够索引的文本格式。关于此的更多信息，可以通过查看 Omega 文档（参见 [参考资料](https://www.ibm.com/developerworks/cn/opensource/os-xapianomega/index.html#artrelatedtopics)）获得。如果想要对一个 web 站点进行爬行并索引，可以使用 htdig2omega，它能接受一个 URL 并搜索整个 web 站点。

构建了初始的数据库后，就能向该数据库添加更多的文档和目录了，不过，为了这个目的，应该使用不同 URL 目录，以便能更为显式地在索引内定位文档。您应该使用 -p 选项来确保现有的文档没有在添加过程中被删除：

|  |  |
| --- | --- |
| 1 | $ omindex -p --db info --url documents /mnt/data0/Documents |

现在，让我们来看看如何能从已创建的这个索引获得信息。

**搜索一个数据库**

为了快速并简单地测试文档的索引，可以使用 quest 命令行工具。该命令可接受数据库目录作为参数之一，然后是 Xapian 格式的一个查询字符串。例如，可以通过清单 2 内所示的命令搜索单个单词。

**清单 2. 使用 quest 工具搜索单个单词 redbook**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29 | $ quest --db=info redbook  Query: Xapian::Query(Zredbook:(pos=1))  MSet:  7218 [100%]  url=info/Books/IBM/Redbooks/sg246622.pdf  sample=Front cover AIX and Linux Interoperability Effective centralized user   management in AIX 5L and Linux environments Sharing files and printers between  AIX 5L and Linux systems Learn interoperable networking solutions Abhijit Chavan  Dejan Muhamedagic Jackson Afonso Krainer Janethe Co KyeongWon Jeong ibm.com/redbooks   International Technical Support Organization AIX and Linux Interoperability April   2003 SG24-6622-00 Note: Before using this information and the product it supports,   read the information in ...  caption=SG246622.book  type=application/pdf  modtime=1050429813  size=4466549  7219 [98%]  url=info/Books/IBM/Redbooks/sg247186.pdf  sample=Front cover Solaris to Linux Migration: A Guide for System Administrators A   comprehensive reference for a quick transition Presents a task-based grouping of   differences between the operating system environments Additional content about how to   optimize Linux on IBM Eserver platforms Mark Brown Chuck Davis William Dy Paul   Ionescu Jeff Richardson Kurt Taylor Robbie Williamson ibm.com/redbooks International   Technical Support Organization Solaris to Linux Migration: A Guide for System   Administrators February ...  caption=Solaris to Linux Migration: A Guide for System Administrators  type=application/pdf  modtime=1138980702  size=3743923 |

在 Xapian 系统内，可以更为明确地指定查询结构，比如，可以指定想要的是组合单词，还是不同的单词和其他结构。

与很多系统一样，Xapian 提供了大量的操作符供您指定想要搜索哪些信息。受支持的主要操作符有：

* AND— 匹配那些单词或表达式均匹配的文档
* OR— 匹配那些任一个表达式匹配的文档
* NOT— 匹配那些第一个子表达式匹配的文档
* XOR— 匹配那些或者第一个或者第二个表达式匹配，而不是两个都匹配的文档

对于熟悉 Google 系统的用户，还可以使用 + 和 - 来标示单词。比如：+IBM +Java -WebSphere。

对于更为细化的搜索，还可执行一个 NEAR 搜索来查找靠近和邻近（ADJ）其他单词的单词，不过它查找的是靠近其他单词的单词，但只能按指定的顺序。二者均支持单词阈值（默认为 10），比如 ADJ/6，其中 6 是单词限制。又比如，IBM NEAR Java 会查找彼此靠近的两个单词，而 IBM ADJ Java 则会查找彼此相邻的那些单词，其中 IBM 是第一个，Java 是第二个。

当然，以这种方式解析输出不是很有用。在很多情况下，需要将搜索结果集成到一个 web 站点或另外一个应用程序。Xapian/Omega 根据想要的集成水平为此提供了不同的解决方案。

**使用 Omega web 接口**

Omega 自带基于模板的 web 接口，这个接口自身非常强大，可提供对 Xapian 数据库的一个直接接口。为了使用它，需要将 omega 命令复制到配置好的 CGI-BIN 目录，或是其他配置好且能支持 CGI 脚本的目录。还可以在 CGI 目录以及在 Omega 安装时所安装的那个版本之间创建一个符号链接。二者之间的差别是在每种情况下，配置文件位于何处。

在将 omega 命令复制到 CGI 目录后，可以创建一个本地配置文件，名为 omega.conf。如果使用了这个符号链接，将需要编辑安装目录内的这个配置文件。默认地，这个配置文件是 /usr/local/etc/omega.conf。

配置文件需要三个配置设置，已创建的 Xapian 数据库的位置、用来保存在搜索和输出期间将被解析的 Omega 模板的模板目录的位置。一个示例配置文件如清单 4 所示。

**清单 3. 示例配置文件**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | # Directory containing Xapian databases:  database\_dir /var/lib/omega/data    # Directory containing OmegaScript templates:  template\_dir /var/lib/omega/templates    # Directory to write Omega logs to:  log\_dir /var/log/omega |

在本例中，之前创建的数据库已经复制到了 /var/lib/omega/data 目录以便您能搜索。

之后就可以创建包含了搜索表单的查询文档并显示结果。主模板文档应被称为 query 并放置于这个模板目录内。此文件的格式和结构是 HTML，其内嵌有 OmegaScript 术语以便执行搜索。在此模板目录的 Xapian/Omega tarball 内提供了这些模板的一个示例文件集合。虽然内容和结构均非常复杂，但可以将这个目录复制到配置好了的模板目录以便对其有些概念。

一个更好的解决方案是通过现有的应用程序和 web 部署环境（比如 Java、Perl 或 PHP）来使用针对 Xapian 索引的这些接口中的一个接口。

**与其他应用程序集成**

如果构建了 Xapian Bindings 包（使用与之前构建 Omega 工具相同的顺序），就可以安装 Java、PHP、Python 和 Ruby 的扩展，前提是配置在系统上可以找到它们。

比如，清单 4 显示了一个使用 Python 搜索数据库的例子。

**清单 4. 使用 Python 搜索数据库**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | #!/usr/bin/env python    import sys  import xapian    try:      database = xapian.Database('info')        enquire = xapian.Enquire(database)        query\_string = str.join(' ', sys.argv[1:])        qp = xapian.QueryParser()      stemmer = xapian.Stem("english")      qp.set\_stemmer(stemmer)      qp.set\_database(database)      qp.set\_stemming\_strategy(xapian.QueryParser.STEM\_SOME)      query = qp.parse\_query(query\_string)        enquire.set\_query(query)      matches = enquire.get\_mset(0, 10)        print "%i results found." % matches.get\_matches\_estimated()      print "Results 1-%i:" % matches.size()        for m in matches:          print "%i: %i%% docid=%i [%s]" % (m.rank + 1, m.percent, m.docid,   m.document.get\_data())    except Exception, e:      print >> sys.stderr, "Exception: %s" % str(e)      sys.exit(1) |

可以使用如下所示从这个索引获得结果，方式与 quest 命令相同：

|  |  |
| --- | --- |
| 1 | $ simplesearch.py IBM |

所有接口均使用同一个基于类的、对 Xapian 库的接口，所以这个过程的基本结构是相同的。您可以调整如下的示例以便适合在您的 WebSphere web 应用程序中使用。请注意 Java 结构与 Python 代码在本质上是多么一致，只有一些明显的语言上的差异（参见清单 5）。

**清单 5. 用来搜索数据库的 Java 代码**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | import org.xapian.\*;    public class SimpleSearch {        public static void main(String[] args) throws Exception {          String dbpath = 'info';            Query query = new Query(args[0]);            Database db = new Database(dbpath);            Enquire enquire = new Enquire(db);          enquire.setQuery(query);          MSet matches = enquire.getMSet(0, 2500);          MSetIterator itr = matches.iterator();            System.err.println("Found " + matches.size() + " matching documents  using " + query);          while (itr.hasNext()) {              itr = (MSetIterator) itr.next();              Document doc = itr.getDocument();              System.err.println(itr.getPercent() + "% [" + itr.getDocumentId()  + "] " + doc.getValue(0\));          }      }    } |

相同的结构可被用于其他的接口语言。

**结束语**

在本文中，您了解了 Xapian 文本索引系统，通过 Omega 扩展，这个系统允许您索引各类文档，然后进行搜索并报告内容。Xapian 的灵活性是通过索引的文本本质实现的，而前端的提交系统会将二进制文档（PDF、Microsoft Word）转化成文本格式。您还看到了从索引再次获得信息的多种方式，比如可以通过命令行，也可以通过不同的语言扩展。

**相关主题**

* 察看 [Xapian 项目](http://xapian.org/)。
* 阅读全部的 [Xapian API Referenece](http://xapian.org/docs/apidoc/html/index.html)。
* [Omega overview](http://xapian.org/docs/omega/overview.html)：整体了解 Omega，它是您 web 站点的一个打包了的搜索引擎，在 Xapian 上构建。
* 下载 [Xapian](http://xapian.org/download)。
* 下载 [Python](http://www.python.org/)。
* [IBM WebSphere Application Server](http://www-01.ibm.com/software/webservers/appserv/was/)：用 IBM WebSphere Application Server 在减少应用程序基础设施成本的同时构建、部署和管理所有类型的强壮、敏捷和可重用的 SOA 业务应用程序和服务。
* 访问 developerWorks [Open source 专区](http://www.ibm.com/developerworks/cn/opensource/)获得丰富的 how-to 信息、工具和项目更新以及[最受欢迎的文章和教程](http://www.ibm.com/developerworks/cn/opensource/best2009/index.html)，帮助您用开放源码技术进行开发，并将它们与 IBM 产品结合使用。
* 随时关注 developerWorks [技术活动](http://www.ibm.com/developerworks/cn/offers/techbriefings/)和[网络广播](http://www.ibm.com/developerworks/cn/swi/)。
* 下载 [IBM 产品评估试用版软件](http://www.ibm.com/developerworks/cn/downloads/) 或 [IBM SOA Sandbox for People](http://www.ibm.com/developerworks/cn/downloads/soasandbox/people/) ，并开始使用来自 DB2®、Lotus®、Rational®、Tivoli® 和 WebSphere® 的应用程序开发工具和中间件产品。