

# 基于 PI3000 平台的全文检索研究与实现

郭 伟

(南京南瑞集团公司信息通信技术分公司, 江苏 南京 210003)

**摘 要:** 全国网省电力系统的用户在使用 PI3000 平台时, 希望能提供快速有效的全文检索功能。本文根据搜索引擎的工作原理, 研究和分析对开源的搜索引擎工具包 Lucene 进行封装和集成, 通过 PI3000 建模工具对全文检索的建模, 快速地构建适合电力系统的全文检索应用。

**关键词:** PI3000; 模型驱动; 全文检索; Lucene

## 0 引言

传统信息系统直接面向各类复杂的操作系统、数据库、中间件构建, 实现技术难度大、成本高, 开发人员存在巨大的手工编码工作量, 无法集中精力解决业务问题, 导致项目实施成本高、周期长、成功率低。PI3000平台结合最新信息技术, 研制全新特性的电力业务基础软件平台, 它的设计参考了业界成熟的业务基础软件平台的理念, 以模型驱动和构件化设计思想为基础, 并采用了先进的面向服务体系架构。

随着PI3000平台的广泛应用于全国网省电力信息系统和生产管理系统, 它可以利用的信息量越来越多, 如何解决“信息超负荷”已成为电力系统迫切需要解决的问题。为了有效地提高海量信息的检索效率, 本文研究了基于PI3000平台, 对开源搜索引擎Lucene进行封装并应用于该平台。

## 1 PI3000 平台概述

### 1.1 技术架构

PI3000平台基于.Net和Java实现的多层分布式架构系统, 它采用了业界最佳实践的面向服务架构(SOA), 可最大程度保证整个系统的兼容性和开放性。此外, PI3000平台还采用了模型驱动思想、组件化框架以及面向对象的分析设计方法。

PI3000平台由基础框架, 模型服务、虚拟文件服务、对象运行时、工作流服务、报表服务、消息服务、任务调度服务等服务, 系统管理员客户端、以及网站系统等组成。系统结构如图1所示。

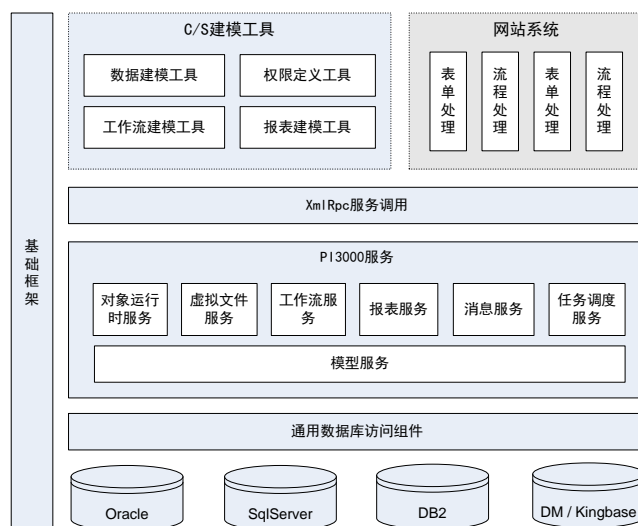


图1 PI3000系统结构

PI3000中的模型对具体电力业务应用系统进行抽象，对业务系统上的业务应用抽象出与业务无关的模型元素，例如数据库实体和实体属性在PI3000平台中用类型和属性模型元素表示，模型元素关系分为继承和关联关系，平台提供一套安全模型元素用于表示业务实体的权限和访问控制。

## 1.2 模型驱动

PI3000 平台还采用了模型驱动架构的设计思想。模型驱动的核心思想是通过深入分析特定领域的数据和应用等方面的共性特征，抽象提炼出一个领域信息系统的元模型（一般是面向对象的），并依此自动或半自动化地构建整个系统。具体而言，就是要针对不同的应用领域，通过各种建模工具将具体企业的业务模型实例化，并由配套的运行环境解释此模型实例，自动生成相应的业务应用功能，从而能大幅度提升系统对业务需求变化的响应速度。

PI3000模型驱动理论中，平台本身不直接涉及特定的业务信息或业务过程，而是通过所建模型间接达到实现具体业务功能的目的。这种理论强调平台负责抽象的信息与过程处理，而特定的业务信息或业务过程对平台而言被视为了一种“数据”。在模型驱动理论看来，平台本身是一个高度抽象的信息系统，某个具体的信息系统实施过程可被视为平台的一次“实例化”。

## 2 全文检索引擎 Lucene

### 2.1 Lucene 简介

Lucene 是一个高性能的、可伸缩的信息检索库，它可以让软件开发者为自己的应用程序添加索引和搜索能力。由信息提取IR( Information Retrieval) 领域的泰斗人物Doug Cutting创立。2001年，Lucene 成为Apache 软件基金会Jakarta 项目组的子项目，并基于Apache 软件许可协议，开放源代码。Lucene 不是一个完整的全文检索引擎，而仅仅是一个全文检索引擎的架构。它是一个软件库，一个开发工具包，而不是一个具备完整特性的应用程序<sup>[1]</sup>。所以使用Lucene 构件全文检索需要在它的基础上做二次开发。

Lucene 可以对任何的文本数据做索引和搜索。它不管数据是什么格式，只要能转化成文本，它都能处理。许多项目都使用了Lucene 作为其后台的全文检索引擎，比较著名的有Jive 的Web 论坛系统、Eyebrows 的邮件列表HTML 归档查询系统、Cocoon 的基于XML 的web 发布框架以及Eclipse 的帮助文档<sup>[2]</sup>。

### 2.2 Lucene 系统结构

Lucene作为一个优秀的全文检索引擎，其系统结构具有强烈的面向对象特征。首先是定义了一个与平台无关的索引文件格式，其次通过抽象将系统的核心组成部分设计为抽象类，具体的平台实现部分设计为抽象类的实现，此外与具体平台相关的部分比如文件存储也封装为类，经过层层的面向对象式的处理，最终达成了一个低耦合高效率，容易二次开发的检索引擎系统。

从图2中我们清楚的看到，Lucene的系统由基础结构封装、索引核心、对外接口三大部分组成。其中直接操作索引文件的索引核心又是系统的重点。Lucene的将所有源码分为了7个模块（在java语言中以包即package来表示），各个模块所属的系统部分也如上图所示。需要说明的是org.apache.lucene.queryPaser是做为org.apache.lucene.search的语法解析器存在，不被系统之外实际调用，因此这里没有当作对外接口看待，而是将之独立出来。

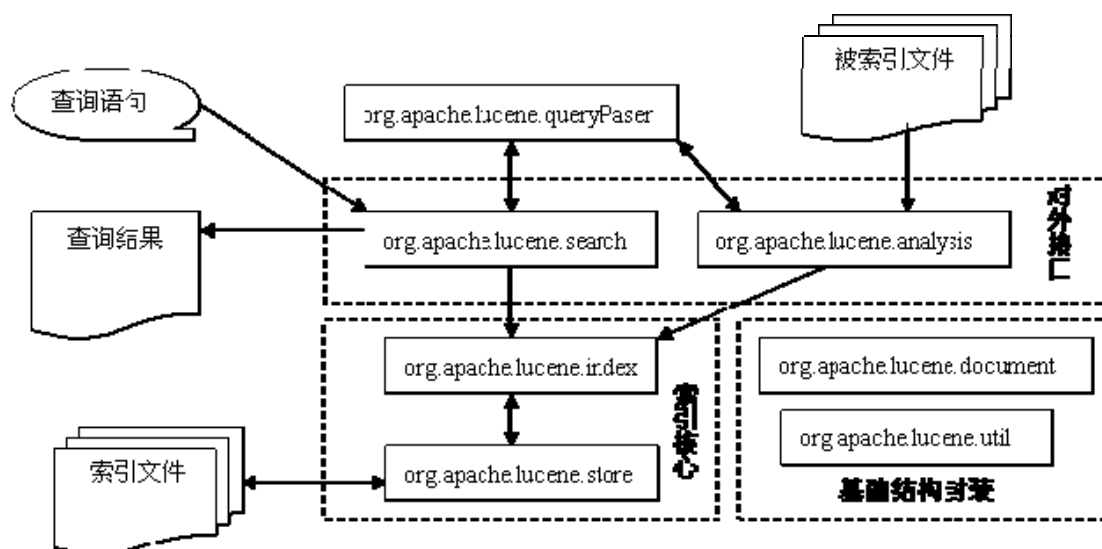


图2 Lucene系统结构

从面向对象的观点来考察，Lucene应用了最基本的一条程序设计准则：引入额外的抽象层以降低耦合性。首先，引入对索引文件的操作org.apache.lucene.store的封装，然后将索引部分的实现建立在（org.apache.lucene.index）其之上，完成对索引核心的抽象。在索引核心的基础上开始设计对外的接口org.apache.lucene.search与org.apache.lucene.analysis。在每一个局部细节上，比如某些常用的数据结构与算法上，Lucene也充分的应用了这一条准则。在高度的面向对象理论的支撑下，使得Lucene的实现容易理解，易于扩展。

Lucene在系统结构上的另一个特点表现为其引入了传统的客户端服务器结构以外的的应用结构。Lucene可以作为一个运行库被包含进入应用本身中去，而不是做为一个单独的索引服务器存在。这自然和Lucene开放源代码的特征分不开，但是也体现了Lucene在编写上的本来意图：提供一个全文索引引擎的架构，而不是实现。

### 2.3 Lucene 的索引

Lucene索引信息存储有三种可选方式：内存(RAM)，文件系统(FS)和数据库(DB)。RAM存储适用于较小的检索系统，文件系统存储可用于中型检索，数据库存储则适用于对检索性能要求更高的系统。下面对其索引存储逻辑进行分析。

在Lucene中，索引(index)由段(segment)组成，段(segment)由记录(document)组成，记录(document)由域(field)组成，域(field)由字符串(term)组成。例如，一个document可以与一个物理文件对应，将其中文件名、文件内容、文件创建时间等信息提取为数据源放入document中<sup>[3]</sup>，也可将多个物理文件的数据源同时放入一个document。

Lucene在维护和扩展索引的时候不断创建新的索引文件，最终将这些新的小索引文件并入大索引中。使程序员可以根据不同的要求自行调整批次大小、周期长短<sup>[4]</sup>。这点对于Lucene的检索效率也相当的重要。建立索引，是需要占用内存资源的，当有新的记录加入索引时，并不直接写入硬盘而是先放在内存中，所以最直接提高检索速度的方法就是提高内存存放索引的缓冲区的大小<sup>[5]</sup>。具体情况的大小设置需要根据实际情况而定。

## 3 PI3000 全文检索实现

### 3.1 PI3000 全文搜索系统结构

PI3000平台将Lucene封装成平台的SOA服务，提供全文搜索的建模型、索引文档的同步和检索服务方法接口。PI3000全文搜索系统结构如图3所示，模型服务在对象的创建、修改和删除时检查当前类型是

否是被索引类型，如果是则调用模型服务的索引事件插件异步地将当前BD的属性含义转换后并序列化成XML保存到索引文档消息队列表，文件服务类似于模型服务，在文件上传、更新和删除时调用索引事件插件将文件转换成文本格式并序列化成xml保存到索引文档消息队列表。

任务调度服务实现索引文档同步，周期扫描消息队列表，如果消息队列表中有待索引的文档时，则调用索引服务的同步索引文档方法，将索引记录写入和编制到磁盘上的索引文件中。任务调度维护索引模型同步，通过任务调度服务定期检查索引模型一致性。

网站直接调用全文搜索服务的查询方法实现全文搜索，无论是查询参数还是查询结果都以XML格式和全文搜索服务交互。

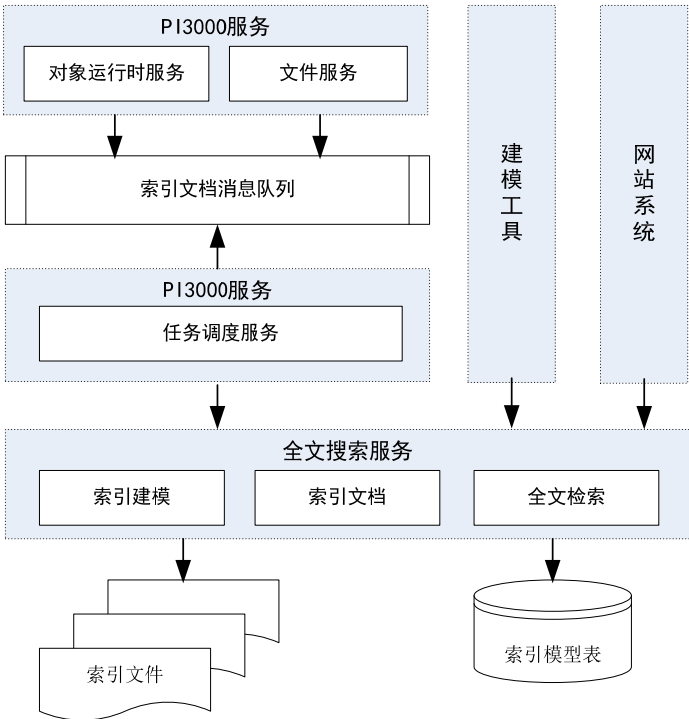


图3 PI3000全文搜索系统结构

### 3.2 全文索引建模

全文索引建模作为主控台的动态插件，管理员通过全文搜索建模工具统一管理平台中的全文索引类型。建模过程定义索引类型和索引的属性，索引类型需要绑定PI3000平台的数据类型，定义索引文件具体存放的物理磁盘位置、索引文件的切词方式等。

全文搜索的属性建模将数据类型的属性编制到索引文件中，定义索引属性是否存储、是否是摘要属性和搜索的权值等。在索引建模完成后，第一次需要重建索引文件，将指定索引类型中的数据模型实例全部编制到全文索引文件中。

### 3.3 索引文档的同步

目前Java模型服务支持BD事件处理的插件，事件处理以插件形式配置在config.xml的”/BusinessModel/BDEventProcessors”中，PI3000对象运行时服务在操作实例对象后，以异步方式调用这些插件进行实例对象的后续处理。

索引文件记录的同步分别在模型服务和文件服务端维护。以模型服务为例，在模型服务端实现IBDEventProcessor接口，此插件主要负责实例对象创建、更新和删除时，将事件中的对象实例插入到索引服务的消息队列表，任务调度定期调用索引服务同步方法，以完成索引文件的同步。文件服务类似于模型服务实现事件处理插件，在文件的上传、更新和删除时调用插件来完成文件流索引的同步。

### 3.4 全文检索

PI3000平台的服务基于SOA的面向服务的架构，提供一个通用的全文检索服务方法，检索的服务接

口便于后期的扩展，该方法的参数和返回值均使为XML格式。XML参数中的keywords元素为搜索的关键字，highlighter属性为在返回匹配的结果集中是否以高亮html标签显示关键字的摘要。sorting属性为排序字符条件，默认根据关键字的匹配度降序排序，timeRange元素为索引创建时间过滤条件，typeRange元素为搜索的类型(平台的对象实例和上传的文件流，docFormat元素为搜索文件流的类型。检索的参数如下所示：

```
<searchRequest>
  <keywords type="quick" highlighter="false">南瑞</keywords>
  <paging pageIndex="0" pageSize="20" />
  <sorting order="lastModifyTime,headLineNews" />
  <conditions>
    <timeRange begin="2009-01-01" end="2009-12-31" />
    <typeRange>OBJECT,OBJECTFILESTREAM</typeRange>
    <clsRange>6F9619FF-8B86-D011-B42D-00C04FC964FF</clsRange>
    <docFormat>pdf,doc,xlsx,wps,pptx</docFormat>
  </conditions>
</searchRequest>
```

搜索的结果集以XML格式返回，该XML的summary元素中的属性包含搜索的关键字、匹配的结果集大小和搜索的服务端耗时。Item元素代表每条匹配的结果集，title元素为检索记录的标题，abstract为检索记录的摘要，在网页上点击检索记录的标题时弹出PI3000的单对象表单，显示完整的平台对象详细信息。

```
<searchResults>
  <summary keywords="年终 总结 小结" pageIndex="1" pageSize="20" totalCount="14012" cost="0.984"/>
  <items>
    <item type="OBJECT">
      <objID>7C156268-B7F8-4BCB-8FBA-EC9014DA4BDA</objID>
      <clsID>C96E33A2-623F-48E9-B032-E62D18EFF057</clsID>
      <title>2011 年终小结</title>
      <abstract>年终工作总结范文参考以及写作指导。 ... 2011 年度年终工作总结 · 2011 年工作总结 · 2011 年个人年终工作总结范文 · 2011 年年终工作 ...</abstract>
      <lastModifyTime>2011-12-14 12:14:53</lastModifyTime>
      <subTitle>2011 年再接再厉</subTitle>
      <headLineNews>T</headLineNews>
    </item>
    <item type="OBJECTFILESTREAM">
      <vfileID>16B68593-4351-410F-8717-46EAA20E0C78</vfileID>
      <objID>7C156268-B7F8-4BCB-8FBA-EC9014DA4BDA</objID>
      <clsID>C96E33A2-623F-48E9-B032-E62D18EFF057</clsID>
      <title>年终总结.doc</title>
      <abstract>新的一年即将过去...</abstract>
      <lastModifyTime>2011-11-02 11:09:01</lastModifyTime>
    </item>
    ....
  </items>
```

</searchResults>

## 4 结论

文中给出了基于PI3000平台的全文搜索的系统架构，综合运用了PI3000平台的模型驱动搜索建模的思想。通过平台任务调度服务将索引对象增量同步到索引文件中，维护索引文件与平台对象的一致性，搜索服务提供了易扩展的检索服务方法。PI3000的全文搜索服务还存在待改进的功能点，目前将全文索引文件保存在服务器的磁盘上，不支持分布式检索和负载均衡，将来需要进一步研究PI3000全文搜索的分存式计算。

### 参考文献：

- [1] 周登明,谢康林. Lucene 搜索引擎[J].计算机工程,2007,33(18):95-96.
- [2] 张校乾,金玉玲,侯丽波. 一种基于Lucene 检索引擎的全文数据库的研究与实现[J].信息检索技术,2005(2):40-43.
- [3] 陈立.全文检索引擎的设计研究[J].现代情报,2007(10):223-225.
- [4] 车东. Lucene：基于Java 的全文检索引擎简介[EB/OL]. [2009-03-20]. <http://www.chedong.com/tech/lucene.Html>
- [5] 周锦程,王丹. 基于Lucene 的全文搜索引擎研究与应用[J]. 黔南民族师范学院学报,2009(3):7-12.

---

### 作者简介：

郭 伟, 13770754594, Email: guowei2@sgepri.sgcc.com.cn。