Lucene仅仅是一个提供信息检索(IR,Information Research)的类库，在它的核心Jar包外，有大量的扩展模块，这些模块位于contrib域，比如spellchecker、highlighter等。

Lucene处理的是文本内容。

**搜索处理流程**

Raw Content(原始数据) → Acquire Content(有用数据) → Build Document(建立文档) →

Analyse Document(分析文档) → Indexing Document(对文档建立索引) → Index(索引) →

Run Query(运行查询) → Build Query(建立查询)、Render Results(渲染结果)

原始数据到有效数据的转换，可以使用的工具很多。对于关系型数据库和XML文件，可以使用Solr(Lucene的子项目)

索引(Index)是一种特殊的数据结构，可以提供快速的文本随机访问。

Document一般包含title、body(正文)、abstract(摘要)、author和url，这些字段在建立的时候可以插入加权值(加权是为了优先匹配)。

Document分析一般是指分词。

搜索处理过程就是从索引中查找单词，从而找到包含该单词的文档。搜索质量主要由精确率(Precision)和查全率(Recall)来衡量。精确率衡量查找相关文档的能力，查全率衡量过滤不相关文档的能力。

用户输入将被QueryParser组件分析装配成Query对象。

搜索理论模型：

Pure Boolean Model，不管文档是否匹配查询，都不会被打分。因此，返回结果是无序的。

Vector Space Model，查询语句和文档都是高维空间的向量模型，每个独立的项都是一个维度。查询语句和向量之间的相关性或相似性由各向量之间的距离计算得到。

Probabilistic Model，使用全概率方法来计算查询语句和文档之间的相关性。

Lucene使用的是前两种。

后台管理：

缓存大小，多长时间合并索引，什么时候删除索引，等等。另外：用户对哪些东西经常搜索，索引命中率如何等等，都需要记录分析，可以使用Google Analytics来做。伸缩性：分布式，单点故障。

常用对象：

IndexWriter，写入、更新、删除索引；索引需要放在某个目录下，因此有Directory；原始数据不能被索引，需要整理成Document。Document包含了多个Field，其中包含了[转换后的]文本数据的那个需要使用分词器Analyzer对内容进行切割，才能被索引。