

# Apache Lucene - Lucene Contrib

## Table of contents

1 Lucene Contrib.....	2
1.1 analyzers.....	2
1.2 ant.....	2
1.3 benchmark.....	2
1.4 db.....	3
1.5 highlighter.....	3
1.6 icu.....	3
1.7 instantiated.....	3
1.8 lucli.....	3
1.9 memory.....	3
1.10 misc.....	3
1.11 queryparser.....	4
1.12 queries.....	4
1.13 remote.....	4
1.14 spatial.....	4
1.15 spellchecker.....	4
1.16 swing.....	4
1.17 wordnet.....	4
1.18 xml-query-parser.....	5

## 1. Lucene Contrib

The Lucene Java project also contains a workspace, Lucene Contrib (formerly known as the Lucene Sandbox), that is open both to all Lucene Java core committers and to developers whose commit rights are restricted to the Contrib workspace; these developers are referred to as "Contrib committers". The Lucene Contrib workspace hosts the following types of packages:

- Various third party contributions.
- Contributions with third party dependencies - the Lucene Java core distribution has no external runtime dependencies.
- New ideas that are intended for eventual inclusion into the Lucene Java core.

Users are free to experiment with the components developed in the Contrib workspace, but Contrib packages will not necessarily be maintained, particularly in their current state. The Lucene Java core backwards compatibility commitments (see <http://wiki.apache.org/lucene-java/BackwardsCompatibility>) do not necessarily extend to the packages in the Contrib workspace. See the README.txt file for each Contrib package for details. If the README.txt file does not address its backwards compatibility commitments, users should assume it does not make any compatibility commitments.

See Contrib CHANGES for changes included in the current release.

You can access the current trunk Contrib repository at <http://svn.apache.org/repos/asf/lucene/dev/trunk/lucene/contrib/>.

### 1.1. analyzers

Contributed Analyzers, Tokenizers, and Filters for various uses and languages.

See analyzers javadoc

### 1.2. ant

Ant task to create Lucene indexes.

See ant javadoc

### 1.3. benchmark

The benchmark contribution contains tools for benchmarking Lucene using standard, freely available corpora.

See benchmark javadoc

#### **1.4. db**

Provides integration with Berkley DB.

See db javadoc

#### **1.5. highlighter**

A set of classes for highlighting matching terms in search results.

See highlighter javadoc

#### **1.6. icu**

Provides integration with ICU (International Components for Unicode) for stronger Unicode and internationalization support.

See icu javadoc

#### **1.7. instantiated**

RAM-based index that enables much faster searching than RAMDirectory in certain situations.

See instantiated javadoc

#### **1.8. lucli**

An application that allows Lucene index manipulation from the command-line.

See lucli javadoc

#### **1.9. memory**

High-performance single-document main memory index.

See memory javadoc

#### **1.10. misc**

A variety of miscellaneous files, including QueryParsers, and other alternate Lucene class

implementations and tools.

See misc javadoc

### **1.11. queryparser**

A new Lucene query parser implementation, which matches the syntax of the core QueryParser but offers a more modular architecture to enable customization.

See queryparser javadoc

### **1.12. queries**

Additional queries for Lucene.

See queries javadoc

### **1.13. remote**

Classes to help use Lucene with RMI.

See remote javadoc

### **1.14. spatial**

Classes to help with efficient distance based sorting.

See spatial javadoc

### **1.15. spellchecker**

Provides tools for spellchecking and suggestions with Lucene.

See spellchecker javadoc

### **1.16. swing**

Swing components designed to integrate with Lucene.

See swing javadoc

### **1.17. wordnet**

Tools to help utilize wordnet synonyms with Lucene

See wordnet javadoc

### **1.18. xml-query-parser**

A QueryParser that can read queries written in an XML format.

See xml-query-parser javadoc