

Lucene-Search

Plugin Information

View Lucene-Search [on the plugin site](#) for more information.

Lucene-Search adds support to search for build data using the Jenkins search box (aka quick navigation). The plugin stores the data after each build in a search database.

Installation

Install it as a normal plugin and all new builds will be searchable.

Before old builds can be searched, a "Rebuild database" must be performed via the "Manage Jenkins" >>> "Lucene Search Manager". This step should never have to be done again, but may take a long time if you have many jobs and builds.

Upgrade to Lucene-Search 4.x

Version 4.0 changed Lucene search engine from version 4 to version 5 so you should recreate the Lucene database after upgrade.

Supported keywords

console: Console output
projectdisplayname: The project display name
projectname: The project name
buildnumber: Project build number*
id: The unique ID generated by Jenkins*
duration: Build duration in ms*
starttime: Start time in ms*
builton: Node name for where the build was run
startcause: Who or what started the build
color: Icon color for the build*
result: Build result, case sensitive. Example SUCCESS, FAILURE

* Not included in the default search

For more information on how to use search usnig Lucene read

http://lucene.apache.org/core/4_9_0/queryparser/org/apache/lucene/queryparser/classic/package-summary.html#package_description

Solr

The easiest way to get Solr up and running is by downloading the latest stable version (5.3.1 at the time of writing), unpack and do the following:

```
cd solr-5.3.1/example/example-schemaless
java -jar ../start.jar
```

Lucene-Search will set up the fields before it needs them. Adding fields via extension points will require restarting Jenkins or reconfiguring the plugin (known bug).

Extension point

If you are a developer and want to add your own data to the database you can implement the extension point "FreeTextSearchExtension". This will allow the Lucene plugin to add your data to the database so it can be searched. It is NOT an all or nothing deal, it's easy to support Lucene-Search without requiring it.

Maven dependency

```
<dependency>
  <groupId>org.jenkins-ci.plugins</groupId>
  <artifactId>lucene-search</artifactId>
  <version>X.X</version>
  <optional>true</optional> <!-- Only support, not require, lucene-search -->
</dependency>
```

The

```
<optional>true</optional>
```

is to only add support for Lucene-Search. This is for when you have a plugin that can use Lucene-Search, but doesn't require it for the plugin to work. If Lucene-Search is in the Jenkins instance at runtime, the Lucene-Search plugin will use the extension points defined by the plugin. If not, the ExtensionPoint will not cause any problems.

Example

```
import org.jenkinsci.plugins.lucene.search.FreeTextSearchExtension;

@Extension
public class FitFreeTextSearchExtension extends FreeTextSearchExtension {

    @Override
    public String getKeyword() {
        return "fitsummary";
    }

    @Override
    public String getTextResult(AbstractBuild, ?&gt; build) {
        StringBuilder builder = new StringBuilder();

        for (FitSummaryAction action : build.getActions(FitSummaryAction.class)) {
            builder.append(action.getSummary() + "\n");
        }

        return builder.toString();
    }

    @Override
    public boolean isPersist() {
        return true;
    }
}</pre
```

Coming changes (completed/prioritized)

Planned

1. Improve console highlightings. "(exit code: 1)" highlights the word code and the number 1. Would be better if we only highlight the full match. - Command completed with `code 0`
2. Paginating results.
3. Find a solution for the bugs [JENKINS-30325](#) and [30665](#) where we can not use Lucene without "XX:-UseSplitVerifier" after Jenkins 1.602

Troubleshooting

JDK Version

The Lucene lib used by this plugin requires that we run Jenkins with Java 7 or later. However due to a bug in Jenkins byte code this plugin will not work if you use Java 8 with a Jenkins version later then 1.601. If you are running on a newer version you will need to run with Java 7 and add the java argument X X:-UseSplitVerifier when starting Jenkins or you will be unable to use the search function.

For more information see Jenkins defect [30665](#)

Example

```
JAVA_OPTS="XX:-UseSplitVerifier"
```

How to search

Just a type a string in the search box to search. If you do not include a suffix the search plugin will search in all keywords bellow not marked with a *. If you want to search for a string in a specific field just add the write <KEYWORDS>:<VALUE>.

IndexWriter is locked

This problem is seen for several reasons

- * The lucene folder on the Jenkins server has invalid user permissions, change the read/write permissions and reconfigure Lucene/restart the Jenkins server
- * Out of permgen memory on Jenkins
- * The solr server has crashed, often as you don't have enough memory on the server.

Versions

Version 4.2 (release Dec 21, 2015)

- Added missing extension on junit log indexer. It should now be possible to search in junit logs.

Version 4.1 (release Dec 19, 2015)

- Workaround for a bug where searching for "XXX YYY" results in a stacktrace.

Version 4.0 (release Dec 18, 2015)

- Updated Lucene to 5.3.1
- Fixed management GUI bugs
- Added exception handling when calling third party plugin extensions to prevent total log collection failure

Version 3.0 (release Oct 15, 2015)

- Support for workflow
- Added support to search in junit logs

Version 2.1 (release Sep 8, 2015)

- Improved search result presentation

Version 1.15 (released Apr 15, 2015)

- Bug fixes for Solr, primarily tests, but also minor changes to rebuild database backend.

Version 1.14 (released Apr 15, 2015)

- Bug fixes for Solr. It should now be possible to use it without restarting Jenkins and rebuild a clean database.

Version 1.13 (released Apr 7, 2015)

- For some reason, 1.12 is not being distributed, but hopefully this release will.
- No changes compared to 1.12 except in a test case. Solr now has working test cases.

Version 1.12 (released Apr 1, 2015)

- Crash bug fixed.
- Minor bug fixes for Lucene.
- When rebuilding, for each build, try to keep old data if no new data can be generated (e.g. if it's based on artifacts).
- Security check returned results. Slows down searches a lot, since all results returned needs to be deserialized and security checked.
- Add stability by adding test cases for Lucene backend.

Version 1.11 (released Mar 29, 2015)

- Display total elapsed time for rebuild database.
- Added plugin to index files. Path globbing works the same as for the artifact archiver.
- All errors when indexing are logged in the log4j output.

Version 1.10 (released Mar 26, 2015)

- Better support for Matrix build names.
- Less brutally ugly icon.
- Create links to full console log and console as text from search result page.
- Search help page with information about all fields indexed, and if persistent, example of what the values may be.
- Many fixes for Solr backend including highlighting, getting rebuilding to work again, avoid search exception.
- Many internal changes, but API is the same.

Version 1.9 (released Mar 18, 2015)

- **Bugfix** When rebuilding the database the plugin failed when trying to index new projects.
- **Management UI** Improved the Management GUI and added a history table when rebuilding the index.

- **Performance** When rebuilding the index the plugin will use several workers in order to improve performance. The number of workers can be set by the user.

Version 1.8 (released Mar 10, 2015)

- **searchModelObject** Now searches will include the results from the default Jenkins search such as direct links to users and jobs.
- **Delete old jobs** Delete all jobs somehow failed on Lucene backend (or never worked?). Fixed now.
- **Rebuild database** Rebuilding database now works as it should for Lucene backend. Test cases for this sorely needed.
- **Default Locale avoided** Avoid using default locale for Lucene for portability.

Version 1.6 (released Mar 5, 2015)

- **Lucene Management page:** Page to rebuild the database after for example upgrading plugins with Lucene extensions.
- **Extension points:** Fix for new extension point fields needing restart of Jenkins (only for Solr)
- **Maintenance feature:** clean up builds that are in search database but not in Jenkins anymore
- **Data Migration:** Migrate data when changing Solr <-> Lucene, especially Lucene -> Solr
- **Usability:** Prefill a dropdown with collections for Solr instead of a text field

Version 1.2 (released Sep 3, 2014)

- **New data format** (will probably cause problems until purged with Old Data)
- **Solr support**, i.e. external search server support. Solr needs to be set up in a managed schemaless configuration. Documentation above
- Removed Nisse feature (debug string that was printed on search)
- **Updated FreeTextSearchExtension**, renamed persist() to isPersist()

Version 1.1 (released Aug 31, 2014)

- Fixed No means Yes bug, also known as the rape bug.

Version 1.0 (released Aug 31, 2014)

- Lucene plugin was released to the world.