Technology Evaluation

# Scope

This document delineates technology evaluations executed by the former technical team of Kanbanana. Evaluations consist of management level overview, hands-on experience and comparison of the technologies with their competitors. The first evaluation focuses on search engines and their applicability towards office documents. The second evaluations focuses persistency solutions for office documents, meta data and articles written in a WYSIWYG web editor.

# Search Engines

The main purpose of a knowledge base is to share and search information gathered throughout the execution of projects within a company. Therefore, information once stored within the knowledge base has to be searchable in a convenient, fast and configurable way. Usually articles written for the knowledge base only represent an abstract of the detailed information contained in one or more documents attached to the article. As the abstract article might not cover all frequently used buzzwords or might be missing at all for some documents, it is important to also index uploaded files so that their content is not neglected in user inquiries. For this purpose, a search engine with an integrated file parser is integrated into the knowledge base. The following implementations for search engines are evaluated. All of them are open source and do not require the acquisition of licenses. Also none do handle the actual persistence of documents handed over for indexing. Therefore, various persistency options are evaluated and described later in this document.

## Apache Lucene & Tika

Apache Lucene is a text search engine library written in Java. For evaluation the library is combine with Apache Tika, a toolkit for detection and extraction of metadata and text content from various file types (e.g. DOCX, PPTX, TXT, PDF). Both Lucene and Tika can be used as standalone applications, but can also be embedded as JAR libraries into a Java project. In order to index a file with Lucene, it as to be abstracted to a java object instance of org.apache.lucene.document.Document. The conversion from various proprietary file formats into this more abstract version is handled by Tika. The requirement of this very specific document form limits the possibilities of connecting Lucene with non-Java technologies. Also the implementation of a search engine with Lucene and Tike requires an unhandy amount of glue code, which has to be implemented, updated and tested discretely.

## Apache Solr

Apache Solr is an open source platform built on Lucene. It adds new features to the search engine, abstracts away from the Java-only interface and covers the document parsing formerly done by external modules like Tika. Solr can be attached to other business logic by calling its REST service with either XML- or JSON-based data. Additionally, Solr provides an API for Java wrapping the REST service for more convenient integration. Solr can even be used to index data stored in databases, as long as there is a JDBC driver for respective database. Even though Solr decreases the implementation and testing effort compared to a bare Lucene solution, configuration effort increases a lot to make the new level of abstraction work.

## Open Search Server

Open Search Server is a platform containing multiple heavy weight components used in combination with a search engine. Besides the search engine itself it contains a file parser comparable to Tika and multiple crawlers able to crawl various data sources like SAMBA drives, FTP servers, JDBC-enabled databases and web pages. Open Search Server is a stand-alone solution delivered either with an integrated web server or as a WAR file to be embedded in a web container. Internally Open Search Server uses Lucene, just as Solr does. Besides its REST interface which can handle XML- and JSON-based data, API wrapper are available for PHP, Ruby, Perl and C#. The biggest benefit of Open Search Server is the set of crawlers it offers. The business logic does not have to handle explicit indexing of each uploaded document. Instead a crawler can be activated on the data storage (e.g. a dedicated directory). The crawler recognizes altered and newly added files and automatically executes the indexing. The downside of this convenience is the huge configuration effort for the crawlers and other components of open search server.

## Elastic Search

Elastic Search is another open source platform built on top of Lucene. It comes as a stand-alone server with RESTful API for JSON-based data. It focuses on massively distributed data sources and optimizes for analytics performance. Therefore, its drivers do not fit the requirements of the knowledge base project. Features like data visualization are not relevant for this project and just like bare Lucene, Elastic Search does require a parser module comparable to Tika in order to index content of files.

## Comparison of Search Engines

The search engines introduced above are compared for their qualities respective to following attributes:

* **Stand-alone:** Is the search engine capable of running on its own?
* **Embeddable:** Is the search engine embeddable into another project?
* **API:** How can other componentens communicate with the serach engine?
* **Data Format:** Which data formats are accepted by the sreach engine?
* **Data Source:** Which data sources does the search engine accept?
* **License:** Under what license is the search engine?

The following table delineates the differences between evaluated search engines towards the attributes explained above:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Apache Lucene & Tika** | **Apache Solr** | **Open Search Server** | **Elastic Search** |
| **Stand-alone** | Yes | Yes | Yes | Yes |
| **Embeddable** | Yes | Yes | (WAR File) | No |
| **API** | Native Java | REST service and service wrapper for Java | REST service and service wrapper for PHP, Ruby, Perl and C# | RESTful service |
| **Data Format** | org.apache.lucene. document.Document | XML,  JSON | XML, JSON | JSON |
| **Data Source** | Explicit from within Java program | Explicit over REST service or link to JDBC-enabled database | Explicit over REST service and crawlers for JDBC, FTP, SAMBA, File System | Explicit over RESTful service |
| **License** | Apache  License 2 | Apache  License 2 | GNU GPL 3 | Apache  License 2 |

After comparison of search engines towards these attributes the Apache Lucene & Tika approach is rated unsuitable for the knowledge base project. The effort which has to be put into the approach in terms of implementing and testing glue code is not justifiable. Also the Elastic Search approach is rated unsuitable, as its drivers and focus do not match with the goals of the knowledge base project. After this first decision

## Decision