Die Suche in Liferay Portal

Unternehmen brauchen für ihre Mitarbeiter Portale, die es ihnen ermöglichen, auf die verschiedenen relevanten Applikationen und deren Datenbestände schnell zuzugreifen. Eine komfortable Suche ist dazu eine Schlüsselfunktion. Mitarbeiter arbeiten effektiv und effizient, wenn sie Informationen suchen und schnell finden. Heterogene IT-Umgebungen und der Zwang zu Einsparungen machen es dabei schwer, eine geeignete und dennoch erschwingliche Lösung zu finden.

Die Suche in Liferay basiert auf dem Lucene Framework. Lucene ist allerdings keine fertige Suchmaschine, sondern stellt Klassen und Funktionen zur Verfügung, um für beliebige Projekte eine eigene Suchmaschine zu bauen.

Die Jakarta Lucene ist eine Open Source, leistungsstarke, voll funktionsfähige Text-Such-Engine, geschrieben in Java. Sie ist eine Technologie, die für fast jede mögliche Anwendung verwendbar ist, die Volltextsuche erfordert, besonders eine Cross-Plattform. Lucene unterstützt standardmäßig reichhaltige Suchoptionen. Allerdings besteht auch die Möglichkeit, eine eigene Suchsyntax zu entwickeln.

1. Operatoren

Welche Suchmöglichkeiten und Operatoren bietet die Lucene Engine in Liferay?

Operatoren	Lucene Search Engine in Liferay
Fuzzy Suche undeutliche Suche zu formulieren mit Tilde-Symbol ("~") Beispiel: seit~liefert: Seite, seinen, weitere, Leiter...	Ja Markierung des Suchwortes entfällt (Hit-High-Lighting)
Boolesche Suche Verknüpfen von Wörtern mit UND, OR, AND **Beispiel:** *Guten AND Tag* + für soll vorkommen **Beispiel:** oder aber *+Tag + guten* NOT nicht vorkommen	Ja
Proximity-Suche/Distanzsuche "Guten wiedersehen"~10 Zeigt Ergebnisse, indem die beiden Wörter	Ja

Guten und wiedersehen maximal 10 Wörter voneinander entfernt liegen.	
Wildcards Lucene erlaubt Wildcard-Suche für ein oder mehrere Zeichen. Um einen Platzhalter für ein einzelnes Zeichen anzugeben, wird ein "?" benutzt. *?aus* würde nach allen Wörtern wie "Maus" oder "Haus" suchen. Um ein Platzhalter für mehrere Zeichen anzugeben, wird ein "*" benutzt. *test** würde nach allen Wörtern wie "test" oder "tester" suchen.	Ja, sowohl innerhalb als auch am Ende des Ausdrucks, aber nicht am Anfang des Wortes: z. B. *wort, *gang **geht nicht** z. B. Haus*, Baum* **geht**
Verstärkungsfaktor um einen Verstärkungsfaktor zuzuordnen, wird das "^"-Symbol verwendet Tag^4 du = Tag ist 4 x stärker gewichtet	Ja
Feldsuche Lucene unterstützt Felddaten. Beispiel: *"title:index AND text: Tag".*	Ja, z. B. Suche kann über Volltext oder Seitentitel gesteuert werden. Dazu muss das Feld indiziert sein.
Bereichssuche	Nein, standardmäßig nicht. Dazu müssen die Felder definiert werden.
Weitere Funktionen	
Tokenisierung Zerlegen eines oder mehrerer Texte in kleine Einheiten, so genannte Tokens. Diese werden anschließend gespeichert in der Token-Dokument-Beziehung.	Ja
Indexierung Mit Indexierungsverfahren wird durch die Ermittlung von Worthäufigkeiten und Wortrelevanz eine Auswahl getroffen und	Ja Bei der Installation von Liferay erstellt die Lucene Engine automatisch aus den Inhalten in der Datenbank einen Index.

somit Wörter in den Index aufgenommen. Anhand des erstellten Index kann dann die Volltextsuche, wie später beschrieben, effizient durchgeführt werden.	Neue Dokumente werden sofort indiziert. Änderungen werden auch sofort berücksichtigt. Lucene untersucht den von Liferay übergebenen Text und schreibt Informationen über die Häufigkeit der im Text und der enthaltenen Worte sowie deren Relevanz in den Index. Das hat den Vorteil, dass bei einer Suche nicht das gesamte Dokument durchsucht werden muss (bei großen Dokumenten würde das sehr lange dauern), sondern nur nach dem Wort in dem Index geschaut werden muss und sofort die Trefferstelle und Häufigkeit feststeht.
Anfrageanalyse oder effiziente Suche Analyse von Text-, PDF-, Word-, Excel- und OpenOffice-Dokumenten.	Ja Für jedes Dokumentformat existiert ein „Analyzer“. Dieser durchsucht und indexiert in optimierter Weise verschiedene Dokumentformate.
Rekursives Crawling Suche durch mehrere Verzeichnisse und externe Laufwerke	Nein, dazu muss die Lucene Engine erweitert werden.
Paging Blättern durch Ergebnislisten.	Ja
TagCloud Eine Schlagwortwolke – ist eine Methode zur Informationsvisualisierung, bei der eine Liste aus Schlagworten, oft alphabetisch sortiert, flächig angezeigt wird, wobei Wörter, die häufiger gesucht werden, größer angezeigt werden als weniger gesuchte.	Ja
Syntaxzeichen ausschließen	Auch Zeichen, die für die Suchsyntax reserviert werden, können in der Suche mit angegeben werden. Ein Backslash ("\") vor einem solchen Zeichen bewirkt, dass es nicht für die Syntax interpretiert, sondern in den Suchausdruck einbezogen wird: \(1\+1\)\:2

Semantische Suche – Beispiel Eine Suchanfrage nach ‚Baum' an eine bedeutungserschließende Suchmaschine verwendet zur Suche auch Begriffe, die im Zusammenhang mit ‚Baum' genannt werden, auch wenn sie in der Anfrage selbst nicht genannt werden. Es werden Ergebnisse, die die Wörter (z. B. Ahorn, Eiche, Linde) beinhalten, angezeigt.	Nein

2. Das Grundkonzept

Das Grundkonzept in Lucene besteht aus Index, Dokument, Feld und Ausdruck.

- Ein Index enthält eine Folge von Dokumenten.
- Ein Dokument ist eine Folge von Feldern.
- Ein Feld ist eine benannte Folge von Ausdrücken.
- Ein Ausdruck ist eine Zeichenkette (String).

3. Indexierung

Die Indexierung dient der Aufbereitung des Suchraums und wird typischerweise zu Beginn und bei Änderungen des Suchraums durchgeführt. Anhand des erstellten Index kann dann die Volltextsuche effizient durchgeführt werden.

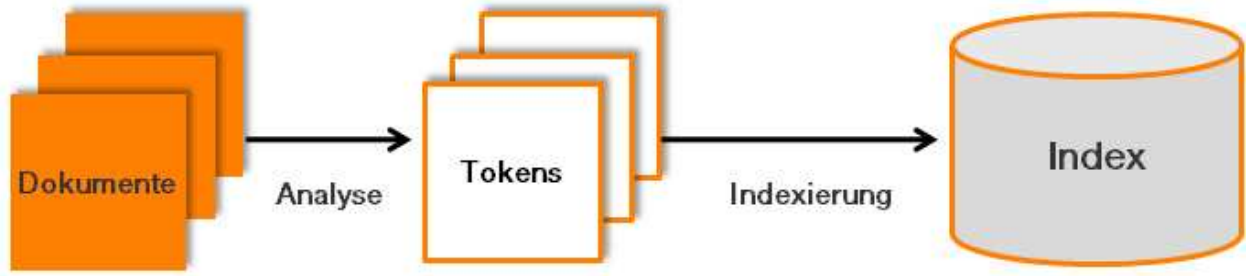

Abbildung: Allgemeiner Aufbau einer Volltextindexierung

Ein Index mehrerer Dokumente bezeichnet im Allgemeinen ein Abbild von Elementen (Tokens oder Terme) dieser Dokumente auf eine Liste mit den Dokumenten, in denen das jeweilige Token vorkommt. Unter der Indexierung versteht man die Generierung eines Index aus einem oder mehreren Texten. Der Indexierungsprozess in Lucene benötigt die folgenden Grundkonzepte mit entsprechenden Java-Klassen:

- **Dokument:** Die von Lucene zu indexierenden Dokumente sind in der Klasse Document abstrahiert. Ein Dokument kann hierbei alles, von einer Buchseite über eine Text- oder HTML-Datei bis zu einem Datenbankeintrag, umfassen.
- **Feld:** Ein Dokument besteht aus mehreren (benannten) Feldern, die auf Wunsch indexiert werden können. Diese Aufteilung ermöglicht eine feiner strukturierte Suche. Beispielsweise könnte man bei einem Word-Dokument neben dem eigentlichen Text als zusätzliche Felder Autor und Beschreibung indexieren.

- **Textvorverarbeitung (Analyse):** Der zu indexierende Text aus den Feldern der Dokumente wird nicht direkt in den Index geschrieben, sondern zunächst von einem sogenannten Analyzer geparst. Dadurch wird der in den Index einzutragende Text vorverarbeitet - das kann von so trivialen Dingen wie Normalisierung auf Kleinbuchstaben über das Weglassen bestimmter trivialer Wörter bis hin zum Eintrag nur des Wortstammes reichen. Alle Suchanfragen werden dann auch vom entsprechenden Analyzer vorverarbeitet.
- **Indexerstellung:** Der eigentliche Index wird von der Klasse org.apache.lucene.index.IndexWriter geschrieben. Dazu muss man angeben, wo der Index gespeichert werden soll - sei es direkt im Dateisystem, in einer Datenbank oder nur im Arbeitsspeicher - und welcher Analyzer verwendet werden soll. In den Index lassen sich dann mehrere Dokumente eintragen. Auch können mehrere Indizes zusammengefügt und optimiert werden.

Die Indexierung hat den Vorteil, dass bei einer Suche nicht das gesamte Dokument durchsucht werden muss, sondern nur nach dem Wort in dem Index geschaut werden muss und sofort die Trefferstelle und Häufigkeit feststeht. Bei großen Dokumenten bzw. Dokumentenmengen würde das sehr lange dauern und die Performance verschlechtern.

3.1. Indexierung Lucene

Es können große bzw. viele Dokumente im zweistelligen Mio.-Bereich in einem System indexiert werden.

4. Verhalten von Liferay und Lucene

1. Liferay erkennt, dass eine neue Ressource in die Suche aufgenommen werden soll.
2. Liferay schaut, um welche Ressourcenart es sich handelt und extrahiert suchrelevante Daten - bei XML-Content z. B. Titel, Body, Dateipfad des Anhangs etc., bei Bildern z. B. Bildname, Bildpfad etc. - als Text.
3. Liferay ruft Lucene auf und übergibt die aus der neuen Ressource suchrelevanten extrahierten Daten; dabei kann Liferay noch angeben, ob diese eine besondere Gewichtung - "boost" - haben sollen. Außerdem wird Text, welcher aus wenigen Einzelworten besteht, höher gewichtet, so z. B. beim Titel, der oft nur aus einem Wort besteht, im Gegensatz zum Content, der aus Tausenden Wörtern bestehen kann.
4. Lucene untersucht den von Liferay übergebenen Text und schreibt Informationen über die Häufigkeit der im Text enthaltenen Worte sowie deren Relevanz in den Index.

5. Die eigentliche Suche in Liferay

1. Benutzer gibt in Liferay einen Suchbegriff ein.
2. Liferay gibt den Begriff an Lucene weiter.
3. Lucene bereitet den Begriff ggf. auf (Operatoren wie AND, OR, Wildcards etc.).
4. Lucene holt sich Informationen über den Begriff aus dem Index und gibt eine Liste von Treffern zurück.
5. Die Liste ist dann so geordnet, dass relevante Treffer am Anfang stehen.
6. Nachdem ein Index erstellt ist, kann darüber gesucht werden.
7. Die Suche geschieht in Lucene mittels spezieller Anfragen (Queries).
8. Eine Anfrage in Lucene besteht aus Termen (einfachen Wörtern) und Phrasen (Sequenzen von Termen), die durch logische Operatoren verbunden sind.

9. Gesucht werden kann mit verschiedenen Operatoren wie Boosting, Proximity/Distanz, Fuzzy und Span Queries.
10. Zusätzlich kann man die Suche auf bestimmte Felder durch Voranstellen von *feldname:* einschränken.

5.1. Darstellung der Suchergebnisse

- Standardmäßig werden in Liferay die Suchergebnisse nach relevanten Treffern gelistet.
- Standardmäßig ist das Suchwort markiert oder hervorgehoben.
- Quelle ist ersichtlich, wenn die Suche über mehrere Websites geht.
- Standardmäßig wird die Trefferanzahl angezeigt.

5.2. Suchvorschläge

Standardmäßig macht Liferay keine Suchvorschläge. Dazu muss der Suchindex oder die Metadatenverwaltung an die Sucheingabemaske angebunden werden. Nur so wird während des Schreibens in das Suchfeld ein Wort aus dem Index oder der Metadatenverwaltung vorgeschlagen, das dort indexiert bzw. aufgenommen wurde. Den Vorschlägen liegt eine Technologie zugrunde, die sich daran orientiert, was der Nutzer mit seiner Suchanfrage am ehesten gemeint haben könnte. Die bereitgestellten Vorschläge laufen dabei im Hintergrund mit und werden erst eingeblendet, wenn der Nutzer das Drop-Down-Menü direkt unter dem Suchfenster öffnet. Die angezeigten weiteren Begriffe und Themenvorschläge sollen dabei helfen, das gesuchte Thema noch präziser einzugrenzen. Ziel wäre es, den User bei seiner Suche schneller und bequemer ans Ziel zu führen.

6. Option Solr

Mit einer Enterprise Search Lösung auf Basis der lizenzkostenfreien Software Solr können Sie das gesamte Wissen Ihres Unternehmens sekundenschnelle „googeln“. Solr ist eine hochskalierbare Suche auf der Basis der Lucene Java-Bibliothek und kann somit unabhängig vom Portal betrieben werden. Dies führt zu einer besseren Performance, Skalierbarkeit, Lastverteilung und Wartbarkeit von Solr.

Mit Solr durchsuchen Sie sämtliche Systeme Ihres Intranets: Datenbanken, CRM und ERP-Systeme, DMS und Content Management Systeme, E-Mails und brauchen keine speziellen Kenntnisse.

Zu den Schnittstellen von Solr gehört beispielsweise eine HTTP-API, mit der Dokumente hinzugefügt, geändert oder gelöscht werden können. Zu den weiteren Funktionen gehören XML/HTTP und JSON APIs, Hit-High-Lighting, facettierte Suche, Caching, Replikation sowie eine Web-Administrations-Oberfläche.

6.1. Vorteile von Solr

6.1.1. Facetting

Es bietet dem Suchenden eine Auswahl von Kategorien für das Eingrenzen der Freitextsuche. Ist der unspezifische Suchbegriff etwa "Restaurant", wären Facetten zur sinnvollen weiteren Eingrenzung typischerweise die Geographie (Städte, Stadtteile) oder Preiskategorien.

6.1.2. Portabilität

Lucene/Solr läuft auf allen Plattform-Systemen, welche Java unterstützen; die erstellten Indizes sind unabhängig vom Plattform-System und können somit ohne Probleme zwischen verschiedenen Plattformen ohne Anpassungen portiert werden.

6.1.3. Individualisierung

Die Solr ermöglicht die flexible Anpassung der Suchalgorithmen, somit können in den Applikationen, so auch bei Liferay, mit verschiedenen Operatoren gesucht werden.

6.1.4. Transparenz

Offene API, Protokolle, Formate und Suchalgorithmen bieten hohe Transparenz.

6.1.5. Sicherheit

Solr respektiert selbstverständlich individuelle Zugriffsregeln und läuft innerhalb Ihrer Firewall. Solr wird bereits in geschäftskritischen Anwendungen bei über 4.000 Unternehmen weltweit eingesetzt, darunter Branchengrößen wie MySpace, AOL, Nike, LinkedIn oder Monster.com.

6.1.6. Performance

Da keine Datenbankzugriffe nötig sind, verkürzen sich intern die Antwortzeiten oft auf unter 50 ms. Die Geschwindigkeit Ihrer Seite und gleichzeitig das Ranking in Suchmaschinen werden verbessert.

6.1.7. Skalierbarkeit

Auch bei wachsenden Datenbeständen sind keine überproportionalen Investitionen in Hardware nötig - dies schont Ihre IT-Budgets; große Anwender können von den Replikations-möglichkeiten und Load-Balancer-Systemen für Solr profitieren.

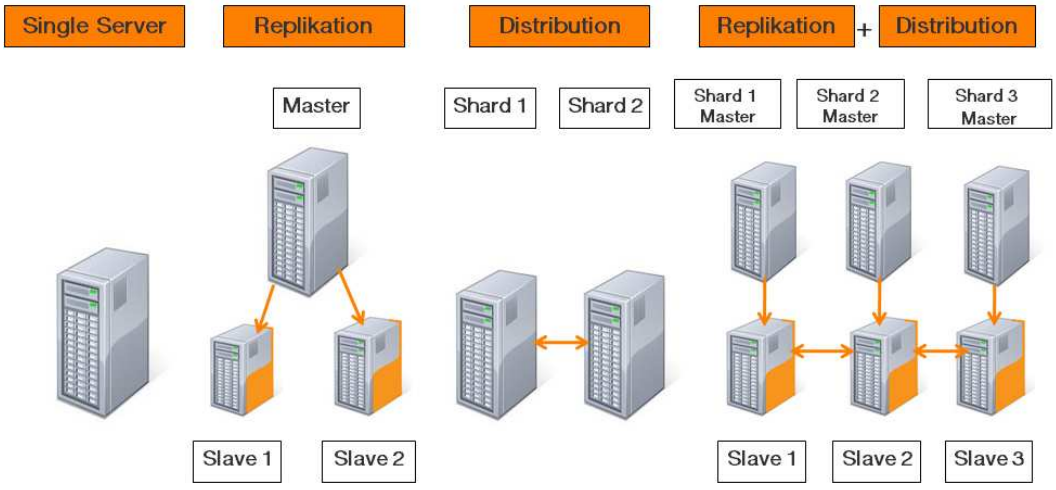

Abbildung: Vier Möglichkeiten, Lucene/Solr Applikation zu konfigurieren

6.1.8. Open-Source-Technologie

Mit Solr nutzen Sie alle Vorteile der Open Source Technologie:

- **Hohe Flexibilität**
  Anpassbarkeit und Erweiterbarkeit, die bei proprietärer Software teilweise nicht möglich ist.
- **Höhere Sicherheit**
  Experten aus der Entwicklergemeinde analysieren potenzielle Sicherheitslücken im Quellcode. Treten einmal Lücken auf, werden diese sofort geschlossen.
- **Keine Lizenzkosten**
  Trotz des höheren Anteils der Dienstleistungs- und Hardwarekosten am TOC ein spürbarer Vorteil.
- **Keine Abhängigkeit**
  Da die Software frei verfügbar ist, besteht keine Abhängigkeit vom Know-how eines bestimmten Herstellers.
- **Offene Standards**
  Hohe Interoperabilität und Kompatibilität durch offene Schnittstellen.

6.1.9. Indexierung Solr

Es können sehr große bzw. viele Dokumente im Mrd.-Bereich verteilter Systeme indexiert werden. Solr kann parallel Indexe/Suchanfragen verarbeiten. Arbeiten können im RAM durchgeführt werden. Die besseren Caching-Technologien liefern bessere Performance.

comundus realisiert auf Basis von Liferay Enterprise Portale, Mitarbeiterportale, Intranets und Internetauftritte. In allen Unternehmenslösungen spielt die Suche eine zentrale Rolle. Eine intelligente Suchfunktion kann darüber entscheiden, ob aus Besuchern Kunden werden und Mitarbeiter effizient und effektiv arbeiten und ihr Wissen mir anderen teilen.

Wir unterstützen Sie gerne, eine starke Suche in Ihrem Unternehmen als wichtigen Beitrag zum erfolgreichen Wissensmanagement zu implementieren.

comundus

comundus GmbH

Heerstraße 111
D-71332 Waiblingen
Telefon: +49 7151 96528-0
E-Mail: info@comundus.com
Internet: www.comundus.com