DOCUMENTS

DOCUMENTS
Volltextindizierung

DOCUMENTS 5.0

Inhaltsverzeichnis

1. Einführung

1.1 Indextypen allgemein

Texte lassen sich zum Zweck einer schnellen Recherche auf verschiedene Arten indizieren. Die einfachste Variante ist ein gewöhnlicher Datenbankindex für die gesamte Textspalte (Primärindex). Ein Volltextindex ist im Vergleich etwas komplexer im Aufbau. Allerdings ist er auch leistungsfähiger während der Recherche. Der Hauptunterschied der beiden Indextypen ist, dass ein Volltextindex den Feldinhalt in einzelne Wörter zerlegt. Der Primärindex kennt keine Wörter. Er behandelt jeden Text als unstrukturierte Zeichenfolge. Ein weiterer Unterschied ist, dass Volltextindizes in der Regel auch Satz- und Sonderzeichen erkennen und beseitigen. Die Suche nach einem einzelnen Komma oder einem Leerzeichen im Volltext ist folglich zwecklos.

1.2 Indextypen in DOCUMENTS

Alle Mappenfelder können in **DOCUMENTS** über einen Primärindex recherchiert werden. Ab **DOCUMENTS 5.0** besteht die Möglichkeit, bestimmte Felder zusätzlich in einen Volltextindex aufzunehmen.

Dateianhänge von Mappen indiziert **DOCUMENTS** ausschließlich als Volltext. Dabei kommen externe Hilfsprogramme zum Einsatz, um den Volltext aus unterschiedlichen Dateitypen zu extrahieren (siehe das Verzeichnis „`docfilter`“ im Installationspfad).

1.3 Die Stoppwortliste

Manche Wörter kommen in Texten so häufig vor, dass sie als Suchkriterium völlig ungeeignet sind. Volltextindizes filtern sie regelmäßig weg, damit der Index nicht zu umfangreich wird und effizient bleibt. Eine Suche nach einem Stoppwort liefert beim **DOCUMENTS**-Volltextindex keine Treffer. Viele Systeme filtern Stoppwörter allerdings auch bei Suchanfragen. Sie reagieren, als wären die Wörter nicht eingegeben worden. In **DOCUMENTS** kann dieses Verhalten bei Archivrecherchen auftreten. Es hängt von der jeweils eingesetzten Software ab.

Die Anpassung der **DOCUMENTS**-Stoppwortliste ist das Thema von Abschnitt 4.3.

1.4 Mustersuche und ihre Effizienz

Prinzipiell kann bei beiden Indextypen das Sonderzeichen „*“ als Platzhalter für eine beliebige Zeichenfolge benutzt werden. Auf diese Weise kann im Primärindex nach Teilen eines Feldwerts sowie im Volltextindex nach Teilen eines Worts gesucht werden. Dabei steigt der Zeitaufwand der Suche erheblich an, wenn das Platzhalterzeichen weit vorne im Suchbegriff steht. Wenn es ganz vorne steht, müsste die Datenbank den kompletten Index durchlesen. Somit ist der Index in diesem Fall nahezu nutzlos.

Die Suche in einem Volltextindex ist vor allem deshalb schneller, weil jedes einzelne Wort ohne vorangestelltes Platzhalterzeichen gefunden werden kann.

Beispiele für einige Recherchen mit und ohne Platzhalter folgen im Abschnitt 3.2 (Seite 9). **DOCUMENTS** kann Platzhalterzeichen auch automatisch anfügen. Dies wird im Abschnitt 4.1 näher erläutert.

2. Aktivierung des Volltextindexes

2.1 Auswahl der zu indizierenden Daten

DOCUMENTS unterscheidet bei der Volltextindizierung zwischen Dateianhängen (Anlagen) und Mappenfeldern. Die direkt in der Mappe gespeicherten Daten (Titel, Erstelldatum u.s.w.) und die Namen von Anlagen werden nicht volltextindiziert. Die Indizierung von Anlagen wird über den Mappentyp gesteuert.

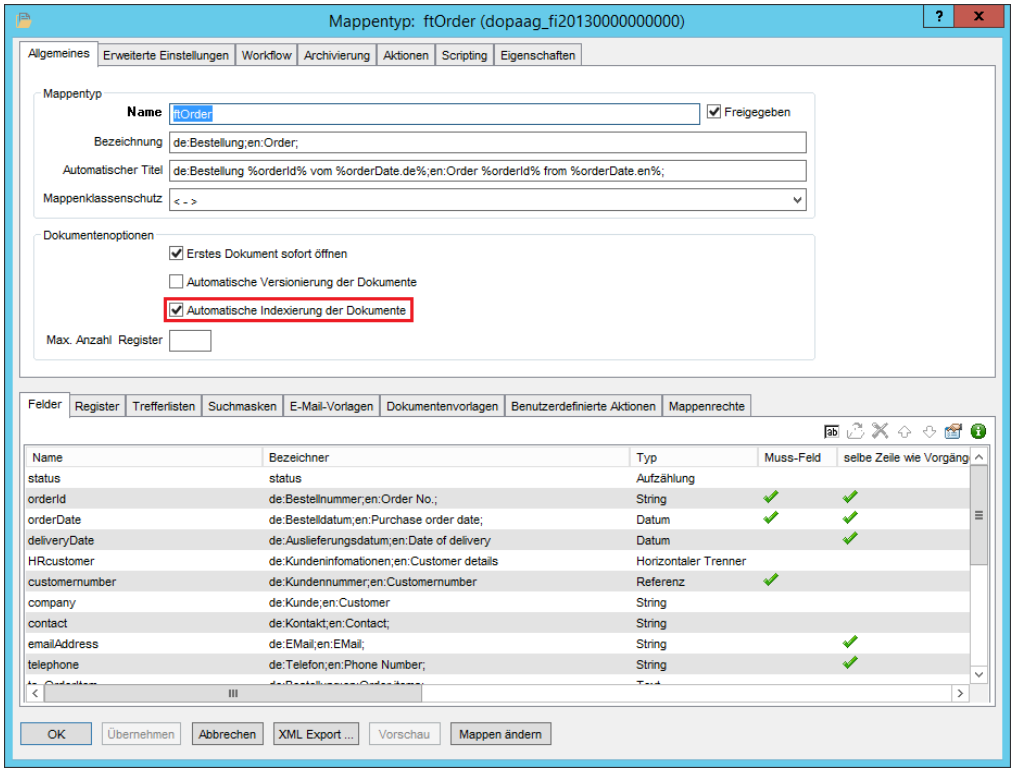

Abb. 1: Indexierung von Dokumenten am Mappentyp aktivieren

Für die Indizierung von Feldern gibt es eine entsprechende Checkbox im Felddialog.

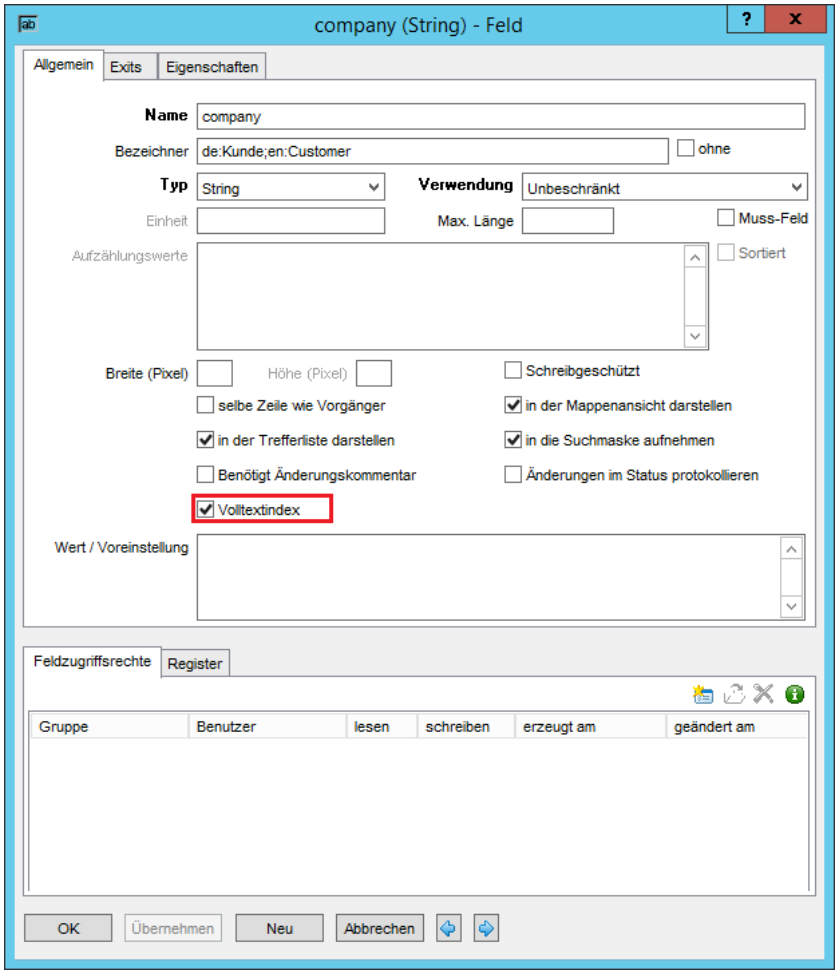

Abb. 2: Volltextindex für Felder aktivieren

Diese Einstellungen betreffen nur **DOCUMENTS**-Mappen *(„aktive Vorgänge“).* Für Archivmappen gelten systemspezifische Einstellungen, die der jeweiligen Dokumentation zu entnehmen sind.

Bei der Neudefinition eines Felds schlägt der **DOCUMENTS-Manager** die Volltextoption für Textfelder automatisch vor. Er entfernt sie bei anderen Feldtypen. Diese Vorauswahl ist nicht bindend. Es ist sinnvoll, von Fall zu Fall zu entscheiden, ob der Volltextindex gebraucht wird. Je weniger Daten indiziert werden müssen, desto schneller erfolgt das Speichern von Mappenänderungen.

2.2 Indizierung von Bestandsdaten

Für bestehende Mappen funktioniert der Volltextindex nicht unmittelbar. Es muss zunächst die Wartungsoperation „`reindex`“ ausgeführt werden. Wenn nur die Anlagen oder nur die Feldinhalte neu indiziert werden sollen, kann man „`reindex docs`“ bzw. „`reindex fields`“ angeben, um den Vorgang abzukürzen. Die Laufzeit dieser Operation kann im Stundenbereich liegen. Daher ist es vorteilhaft, zunächst alle gewünschten Einstellungen vorzunehmen, und die Neuindizierung erst zum Schluss auszulösen.

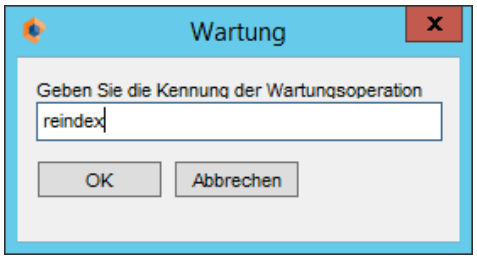

Abb. 3: Wartungsoperation "reindex"

Wichtiger Hinweis

Wenn vor Abschluss der Neuindizierung in dem System eine Suche läuft, die den Volltextindex benutzt, können Treffer ausbleiben. Bei Abfragen mit „enthält nicht“ können auch falsche Treffer entstehen. In ***sehr*** *speziellen Fällen kann das zu Folgefehlern in Portalskripten oder in externen Anwendungen führen. Ein extremes Beispiel wäre ein Skript, das mit einem „enthält nicht“-Filterausdruck falsche Mappen ermittelt, die es anschließend ohne weitere Sicherheitsabfragen löscht.*

*Als Gegenmaßnahmen bieten sich die Wartungskommandos „****jobs off****“ (vorher) und „****jobs on****“ (nachher) an sowie ein zeitweiliger Stopp des* DOCUMENTS-SOAP-Proxys*. Zur Sicherheit sollte auch der Web-Zugang gesperrt werden.*

3. Recherchen mit dem Volltextindex

3.1 Verwendung

Die Recherchefunktionen von **DOCUMENTS** benutzen den internen Volltextindex in folgenden Fällen.

1. Suche über die Weboberfläche bzw. mit dem Webrechercheoperator „*enthält*“ in einem volltextindizierten Feld. Davon ausgenommen sind Feldtypen mit numerischem oder binärem Charakter, bei denen eine Volltextrecherche zwecklos ist (z.B. Checkbox, Zeitstempel).

2. Suche in einem allgemeinen Volltextsuchfeld. Diese Suche schließt grundsätzlich alle Daten einer Mappe ein: Attribute wie z.B. Titel, Erstelldatum, sowie Feldinhalte, Dateinamen und Dateiinhalte. Innerhalb von Portalskripten und externen Anwendungen kann diese Form der Recherche mit dem Bezeichner „`Search_Fulltext`“ anstelle eines Feldnamens durchgeführt werden.

3.2 Beispiele mit Vergleich der Indextypen

In einem Textfeld einer Mappe sei der folgende, aus Abschnitt 1.4 übernommene Beispielsatz gespeichert.

> *Prinzipiell kann bei beiden Indextypen das Sonderzeichen „*“ als Platzhalter für eine beliebige Zeichenfolge benutzt werden.*

Die folgende Tabelle zeigt, ob die beiden Indextypen für verschiedene Suchausdrücke einen Treffer erzeugen und wie der Zeitaufwand der Recherche einzuschätzen ist. Dabei wird vorausgesetzt, dass **DOCUMENTS** keine Platzhalterzeichen automatisch ergänzt, d.h. alle Suchmethodeneinstellungen haben den Wert 1 (exakte Suche).

Ausdruck	Indextyp	Treffer	Suchzeit Einzelfeld*	Suchzeit Volltextfeld*	Bemerkungen
Prinzipiell	Primär	Nein	unter 1s	unter 1s	Werte werden exakt verglichen, daher kein Treffer
	Voll	Ja	unter 1s	unter 1s	
Prinzipiell*	Primär	Ja	unter 1s	unter 1s	
	Voll	Ja	unter 1s	unter 1s	
P*	Primär	Ja	unter 1s	1s bis 2 s	
	Voll	Ja	unter 1s	1s bis 2 s	
Indextypen*	Primär	Nein	unter 1s	unter 1s	Gesuchter Ausdruck steht nicht am Feldanfang
	Voll	Ja	unter 1s	unter 1s	
Z*	Primär	Nein	unter 1s	unter 1s	
	Voll	Ja	unter 1s	unter 1s	Treffer beim Wort ‚Zeichenfolge'
ka*	Voll	Nein	unter 1s	unter 1s	„kann" würde passen, doch es ist ein Stoppwort und steht nicht im Index
!Prinzipiell	Primär	Ja	unter 1s	unter 1s	Wirkt wie „ungleich". Mit „*" dahinter ist es kein Treffer mehr
	Voll	Nein	unter 1s	unter 1s **	Wirkt wie „enthält nicht". Vorsicht bei Verwendung in Skripten! (siehe Hinweis unter 2.2)
Zeichenfolge	Primär	Ja	2s bis 3s	9s	
	Voll	Ja	3s	9s	
Zeichenfolge (jeweils nach Neustart aller Dienste)	Primär	Ja	3s	33s	
	Voll	Ja	3s	33s	
"*Zeichenfolge benutzt*" (in einer Zeile)	Primär	Ja	3s	10s	Dieses Ersatzmuster für eine Phrasensuche funktioniert nur, wenn im Feld genau ein Leerzeichen zwischen den Wörtern steht. Bei Mehreren oder Zeilenumbruch ist es kein Treffer.
"Zeichenfolge benutzt" (in einer Zeile)	Voll	Ja	unter 1s	unter 1s	Phrasensuche mit Volltextindex. Je mehr Wörter in der Phrase desto langsamer wird die Suche.

Testsystem: 16 GB RAM, Datenbank SQL Server 2012 64 Bit, 111663 aktive Mappen im System, darunter 61412 vom Testmappentyp; Feldanzahl gesamt: 2.521.727; Einzelwörter im Feld-Volltextindex: 8.338.279. Keine Volltextindizierten Anlagen vorhanden. Kein Mappenklassenschutz; Ordnerrechte deaktiviert. Jobs deaktiviert. Nur ein angemeldeter Benutzer im System.

Die Zeitangaben beziehen sich auf das reine Ermitteln der Mappen-Ids. Die Zeit für den Aufbau der ersten Trefferseite kommt jeweils noch hinzu.

Abb. 4: Unterschiede der beiden Indextypen bei der Suche

4. Ergänzende Einstellungen

4.1 Suchmethoden

Der Parameter „Suchmethode“ in den **Documents**-Einstellungen dient dazu, einen Suchbegriff automatisch mit Platzhalterzeichen einzuschließen, wenn eine Mustersuche für das Feld sinnvoll ist (z.B. nicht bei Boolean). Somit müssen die Benutzer nicht mehr ständig „*“ für ein bestimmtes Suchmuster eintippen. Von den drei Basismethoden sind beim Volltextindex nur zwei praktikabel, weil die Anwendung sonst bei komplexen Anfragen regelrecht stehen bliebe. Die Methode 0 wird beim Volltexindex deshalb wie 2 behandelt. Es gibt separate Suchmethodeneinstellungen für **DOCUMENTS**-Mappen und für Archivmappen.

Suchmethoden in **DOCUMENTS**				
Nummer	Beschreibung	Benutzereingabe (Beispiel)	Suchmuster im Primärindex	Suchmuster im Volltextindex **
0	beliebiger Feldteil	foo	*foo*	foo*
1	exakte Suche	foo	foo	foo
2	Feldanfang / Wortanfang	foo	foo*	foo*
3*	beliebiger Feldteil; bei Volltextsuche vorangestellte Platzhalter unterdrücken	foo	*foo*	foo*
		*foo	*foo*	foo*
4*	exakte Suche; bei Volltextsuche vorangestellte Platzhalter unterdrücken	foo	foo	foo
		*foo	*foo	foo
5*	Feldanfang / Wortanfang; bei Volltextsuche vorangestellte Platzhalter unterdrücken	foo	foo*	foo*
*: Die Zusatzmethoden 3 bis 5 werden nur für aktive Mappen unterstützt. Sie sollten nicht als Archivsuchmethode angegeben werden. **: Spezielle Einstellung für EE.i für EE.x erforderlich; siehe Text.				

Abb. 5: Suchmethoden in DOCUMENTS

Wenn **DOCUMENTS** mit einem **EE.i** oder **EE.x**-Archiv verbunden ist, wird zwecks Abwärtskompatibilität mit der bisherigen Einstellung noch kein Suchmuster für Volltextanfragen erzeugt. Dazu muss in den **DOCUMENTS**-Einstellungen erst die Eigenschaft „`FulltextMethod`“ mit der Methodennummer 2 als Wert eingetragen werden. Sie gilt dann für alle Volltextrecherchen systemübergreifend. Umgekehrt bewirkt diese Eigenschaft mit dem Wert 1, dass beim Einsatz des **DOCUMENTS-Archivs (EAS)** die Volltextrecherche wortgenau erfolgt, solange kein Platzhalterzeichen manuell angegeben wird. Somit kann die Suchmethode für Primärindex und Volltextindex unabhängig gesteuert werden.

4.2 Parameter des Volltextindizierers

Im Unterverzeichnis „`docfilter`“ der **DOCUMENTS**-Installation befindet sich das Hilfsprogamm „`wpl`“, das **DOCUMENTS** zum Indizieren von Anlagen einer Mappe benutzt. In der zugehörigen Konfigurationsdatei „`wpl.ini`“ gibt es folgende Einstellungen.

- „`wordMode`“ legt fest, welche Zeichen als Teil eines Worts akzeptiert werden. Alle Übrigen gelten in der Regel als Trennzeichen. Im Modus 0 werden nur reine Buchstabenfolgen akzeptiert. Im Modus 1 sind auch Kombinationen aus Buchstaben und Zahlen erlaubt. Modus 2 erlaubt folgende Zeichen als Teil eines Worts: Punkt, Komma, Semikolon, Schrägstrich, Minuszeichen und Klammeraffe. Modus 3 kombiniert die Erweiterungen der Modi 1 und 2.
- „`minWordLength`“ legt fest, aus wie vielen Zeichen ein Wort mindestens bestehen muss, um im Volltextindex gespeichert zu werden. Die Voreinstellung ist 3.
- Die Einträge in dem Abschitt [filter] bestimmen, welches Kommandoteilentool zum Extrahieren der Textinformation aus einem bestimmten Dateityp benutzt werden soll. Die Einträge entsprechen folgendem Muster.

```
Dateierweiterung=Kommandozeile
```

Der Platzhalter *„%d“* steht hierbei für den Dateinamen inklusive Pfad.

4.3 Änderung der Stoppwortliste

Bei der Datei „`wpl.swl`“ im Verzeichnis „`docfilter`“ handelt es sich um die vorinstallierte Stoppwortliste. Sie kann nach Bedarf mit einem gewöhnlichen Texteditor bearbeitet werden. In diesem Fall sollte zuerst eine Sicherungskopie von der Originaldatei erstellt werden und später eine von der individuellen Version. Ferner darf die Zeichenkodierung der Stoppwortliste nicht verändert werden.

Bei einem Update von **DOCUMENTS** wird die Stoppwortliste eventuell überschrieben. Ein Vergleich der alten Originalliste mit der frisch installierten Version und mit der gesicherten individuellen Version erleichtert dann das Zusammenführen von Änderungen aus dem Update mit eigenen Anpassungen.

4.4 Spezielle Datenbankoptimierung

Zugunsten der Portierbarkeit verwendet **DOCUMENTS** für den internen Volltextindex keine Sonderfunktionen einer Datenbanksoftware, sondern gewöhnliche Datenbanktabellen. Pro **DOCUMENTS**-Mandant werden zwei Tabellen gleicher Struktur erzeugt (DlcDocumentI… für Anlagen und DlcFTII… für Felder). Zu diesen Tabellen gehört jeweils einen Datenbankindex „<Tabellenname>_word“.

Es hat sich herausgestellt, dass beim **Microsoft SQLServer** eine Vereinfachung der Ausführungspläne für die Volltextsuche erreicht werden kann, wenn dieser Index etwas erweitert wird. Die Erweiterung belegt ungefähr 80 Bytes pro Datensatz. In Produktivsystemen mit Millionen Indexeinträgen macht das allerdings hunderte Megabytes aus. Daher sollte die Erweiterung nur nach Abwägung der Vor- und Nachteile benutzt werden.

Der **DocumentsServer** verwaltet diesen Index selbst und nimmt nach jedem Start die datenbankseitige Anpassung vor, wenn die Konfiguration geändert wurde. Um die Erweiterung zu aktivieren, genügt folgende Ergänzung in der „`documents.ini`“.

```
$FTI_LARGE 1
```

Grundsätzlich funktioniert die Indexerweiterung auch mit **Oracle DB** und **MySQL**, jedoch konnte hier bisher keine Verbesserung der Geschwindigkeit festgestellt werden.

Abbildungsverzeichnis