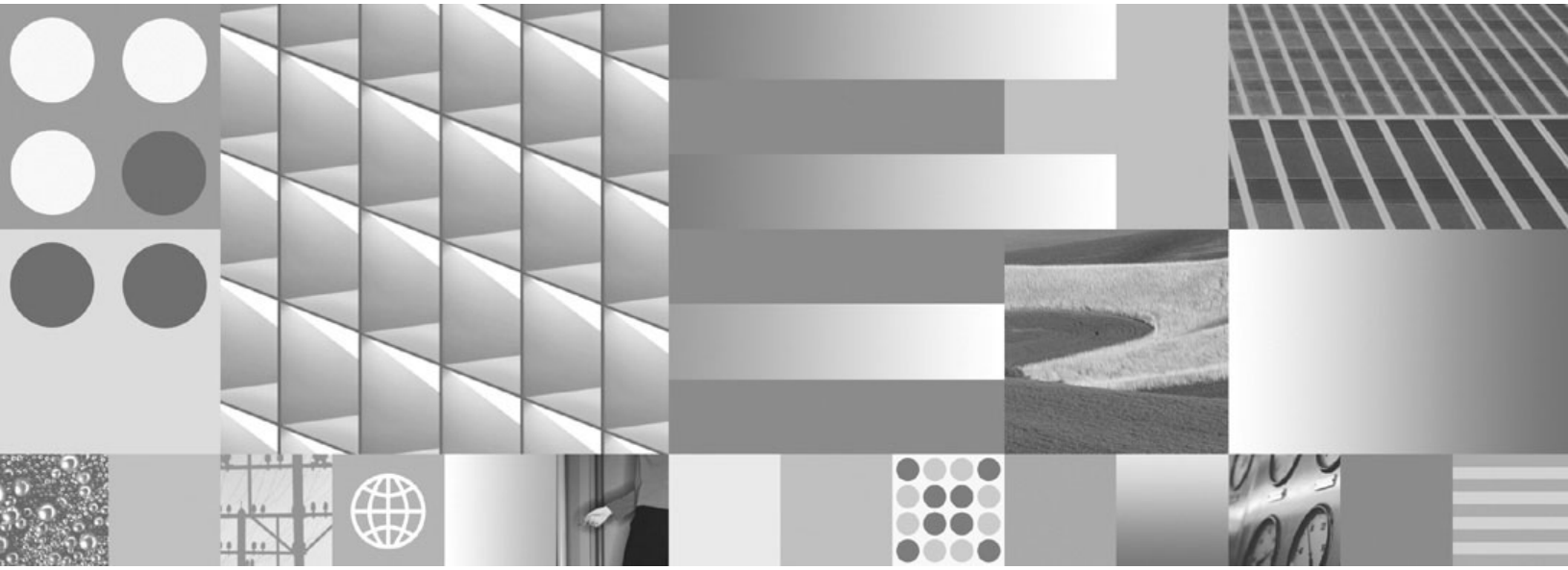
IBM OmniFind Enterprise Edition

Version 8.5

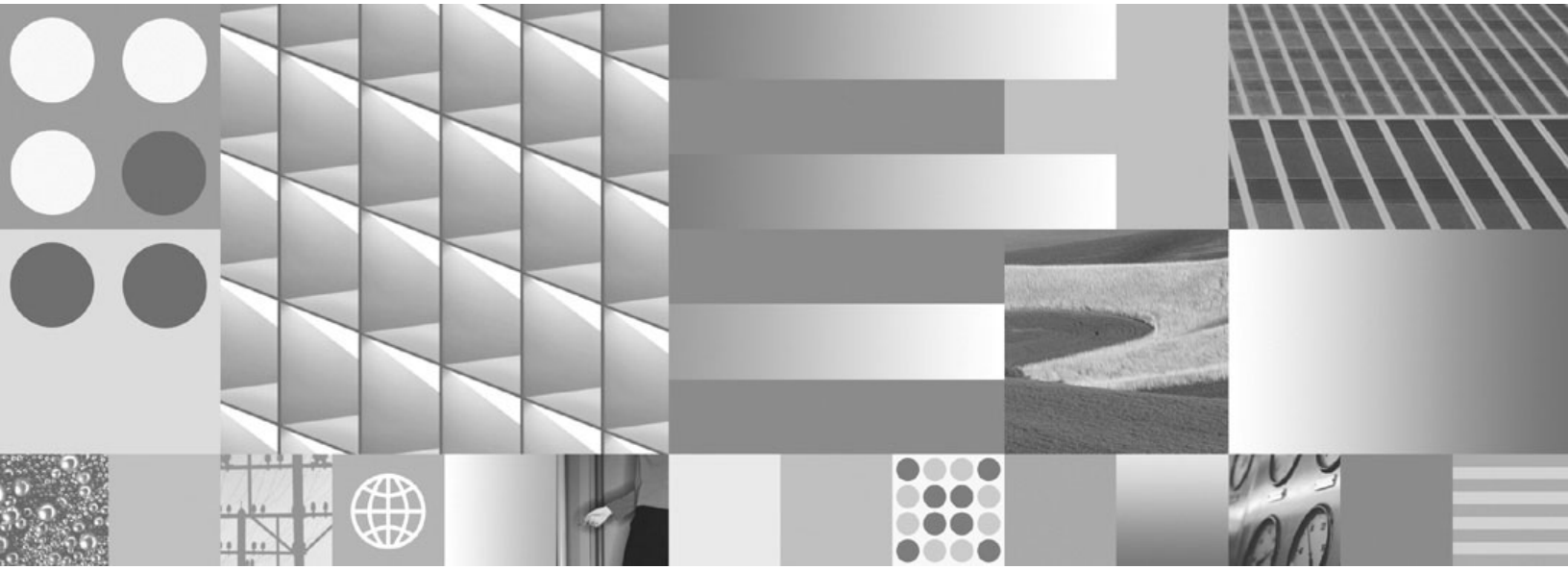


Integration der Textanalyse

SC12-3611-02

IBM OmniFind Enterprise Edition

Version 8.5



Integration der Textanalyse

SC12-3611-02

Hinweis

Lesen Sie vor Verwendung dieser Informationen und des darin beschriebenen Produkts die Informationen in „Bemerkungen und Marken" auf Seite 123 gelesen werden.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs
IBM OmniFind Enterprise Edition Version 8.5 Text Analysis Guide,
IBM Form SC18-9674-03,
herausgegeben von International Business Machines Corporation, USA

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:
SW TSC Germany
Kst. 2877
Februar 2008

Inhaltsverzeichnis

ibm.com und zugehörige Ressourcen . . v
Senden von Kommentaren. v
Kontaktaufnahme mit IBM vi

Linguistische Unterstützung der semantischen Suche 1

Integration der benutzerdefinierten Textanalyse 3
Grundlegende Konzepte für die Verarbeitung der Textanalyse 4
Textanalysealgorithmen. 5
Workflow für die Integration der benutzerdefinierten Analyse 6
Verwenden der Basisannotatoren für die Unternehmenssuche in UIMA 8
Verwenden der allgemeinen Analysestruktur für private Datenbankanwender in UIMA 11
Verwenden des Annotators für reguläre Ausdrücke in UIMA 13
Anzeigen der Ergebnisse des Basisannotators und der angepassten Textanalyse. 14
Typsystembeschreibung 16
Wechsel vom Basisanalysemodus in den erweiterten Analysemodus 17
Für die Unternehmenssuche definierte Typen und Komponenten 18
Bestimmte Typen und Komponenten für die Unternehmenssuche 23
Beispiel für eine Typsystembeschreibung . . . 26
XML-Markup-Formatierung in Analyse und Suche 29
Erstellen einer Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur 31
Ergebnisse der Textanalyse 36
Komponentenpfade. 37
Integrierte Komponenten 38
Filter 41
Indexzuordnung für benutzerdefinierte Analyseergebnisse 41
Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zum Index 43
Datenbankzuordnung für ausgewählte Analyseergebnisse 49
Speichern von Analyseergebnissen in einer Datenbank. 50
Verwenden von Ladedateigruppen 50
Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank 52
Zuordnung von Containertypen 57
Abrufen von Teilen eines Dokuments, die mit einer semantischen Suchabfrage übereinstimmen 61
Semantische Suchanwendungen 64
Begriff der semantischen Suchabfrage 64

Synonymunterstützung in Suchanwendungen 67
Erstellen einer XML-Datei für Synonyme 68
Erstellen eines Synonymverzeichnisses 69

Benutzerdefinierte Verzeichnisse von Stoppwörtern 71
Erstellen einer XML-Datei für Stoppwörter 72
Erstellen eines Verzeichnisses von Stoppwörtern . . 73

Benutzerdefinierte Verzeichnisse von Boostwörtern 75
Erstellen einer XML-Datei für Boostwörter 76
Erstellen eines Verzeichnisses von Boostwörtern . . 77

Textanalyse innerhalb der Unternehmenssuche 79
Spracherkennung 79
Linguistische Unterstützung der nicht wörterverzeichnisbasierten Segmentierung 81
Aufbereiten von numerischen Zeichen als N-Gram-Token 82
Linguistische Unterstützung der Segmentierung auf der Basis von Wörterverzeichnissen 83
Wortsegmentierung im Japanischen 85
Orthografische Varianten im Japanischen . . . 85
Stoppwortentfernung 85
Zeichennormalisierung 86

Annotator für reguläre Ausdrücke . . . 89
Einfache semantische Suche mithilfe des Annotators für reguläre Ausdrücke 90
Aktivieren der einfachen semantischen Suche mithilfe des Annotators für reguläre Ausdrücke . . . 91
Die Regelsatzdatei 93
Definieren von Regeln für reguläre Ausdrücke . . 94
Anpassen des Annotators für reguläre Ausdrücke 97
Der Annotatordeskriptor 99
Protokollierung. 102

Dokumentation für die Unternehmenssuche. 105

Funktionen zur behindertengerechten Bedienung 107

Glossar der Begriffe für die Unternehmenssuche. 109

Bemerkungen und Marken. 123
Bemerkungen 123
Marken 125

Index 127

ibm.com und zugehörige Ressourcen

Produktunterstützung und -dokumentation ist unter ibm.com verfügbar.

Unterstützung

Produktunterstützung ist im Web verfügbar.

IBM OmniFind Enterprise Edition
http://www.ibm.com/software/data/enterprise-search/omnifind-enterprise/support.html

IBM OmniFind Discovery Edition
http://www.ibm.com/software/data/enterprise-search/omnifind-discovery/support.html

IBM OmniFind Yahoo! Edition
http://www.ibm.com/software/data/enterprise-search/omnifind-yahoo/support.html

Informationszentrale

Sie können die Produktdokumentation mit einem Web-Browser in einer Eclipse-basierten Informationszentrale anzeigen. Sie finden die Informationszentrale unter http://publib.boulder.ibm.com/infocenter/discover/v8r5m0/.

PDF-Veröffentlichungen

Sie können die PDF-Dateien online anzeigen, indem Sie Adobe Acrobat Reader für Ihr Betriebssystem verwenden. Wenn Sie Acrobat Reader nicht installiert haben, können Sie ihn von der Adobe-Website unter http://www.adobe.com herunterladen.

Rufen Sie die folgenden Websites mit PDF-Veröffentlichungen auf:

Produkt	Adresse der Website
OmniFind Enterprise Edition Version 8.5	http://www.ibm.com/support/docview.wss?rs=63&uid=swg27010938
OmniFind Discovery Edition Version 8.4	http://www.ibm.com/support/docview.wss?rs=3035&uid=swg27008552
OmniFind Yahoo! Edition Version 8.4	http://www.ibm.com/support/docview.wss?rs=3193&uid=swg27008932

Senden von Kommentaren

Ihre Rückmeldung ist eine wichtige Hilfe dabei, präzise und qualitativ hochwertige Informationen bereitzustellen.

Senden Sie Ihre Kommentare, indem Sie das online verfügbare Formular für Leserkommentare unter https://www14.software.ibm.com/webapp/iwm/web/signup.do?lang=en_US&source=swg-rcf verwenden.

Kontaktaufnahme mit IBM

Unter 0180 3 313233 erreichen Sie Hallo IBM, wo Sie Antworten zu allgemeinen Fragen erhalten.

Telefonische Unterstützung erhalten Sie über folgende Nummern:

- Unter 0180 3 313233 erreichen Sie Hallo IBM, wo Sie Antworten zu allgemeinen Fragen erhalten.
- Unter 0180 5 426014 erreichen Sie die DB2 Helpline, wo Sie Antworten zu DB2-spezifischen Problemen erhalten.

Weitere Informationen dazu, wie Sie mit IBM in Kontakt treten, finden Sie auf der IBM Website **Kontakt** unter http://www.ibm.com/contact/de/.

Linguistische Unterstützung der semantischen Suche

Die Unternehmenssuche bietet die linguistische Unterstützung der Suche für Textdokumente in den meisten indo-europäischen und asiatischen Sprachen an, einschließlich Japanisch.

Sie können die linguistische Unterstützung verwenden, um die Qualität der Suchergebnisse zu verbessern.

Die Verarbeitung auf linguistischer Basis erfolgt in zwei Arbeitsabschnitten: Wenn ein Dokument verarbeitet wird, damit es dem Index hinzugefügt werden kann, und wenn ein Benutzer eine Suchabfrage eingibt.

Die Unternehmenssuche enthält nur eine grobe oder grundlegende linguistische Funktionalität, die verwendet wird, um die Sprache eines Eingabedokuments zu ermitteln und den Eingabedatenstrom des Dokuments in Wörter oder Token zu segmentieren.

Wenn Sie wissen, dass Sie sich hauptsächlich auf eine Suche beschränken werden, die die Dokumentstruktur verwendet, wie z. B. die Suche nach Basisschlüsselwörtern oder nach nativer XML, werden Ihre Bedürfnisse von der in der Unternehmenssuche enthaltenen Verarbeitung auf linguistischer Basis in ausreichendem Maße erfüllt.

Die meisten in Textdokumenten enthaltenen Informationen sind unstrukturiert und damit nur sehr schwer effektiv zu verwenden, da sich der Zugriff auf die Bedeutung der Informationen schwierig gestaltet.

Es ist zwar einfach, nach Schlüsselwörtern zu suchen, führt aber nicht immer zu zufriedenstellenden Ergebnissen, wenn Sie über die einzelnen, im Dokument enthaltenen Wörter hinausgehen wollen. Dies wird in den folgenden Beispielen verdeutlicht:

- Bei der Onlinezusammenarbeit sind Informationen nicht immer explizit gekennzeichnet, zum Beispiel wenn eine Adresse oder Telefonnummer in einer E-Mail angegeben wird. So kann es sein, dass der Begriff *Telefonnummer* gar nicht auftaucht. Statt dessen enthält die E-Mail möglicherweise einen Satz wie "Sie erreichen mich unter 0711 24798". Der Benutzer weiß häufig nicht, wie die Informationen, nach denen er sucht, im Dokument dargestellt werden, und würde im Idealfall eine gezielte Abfrage nach der Telefonnummer von Barbara ausführen, wenn er nach der Telefonnummer einer Person mit dem Namen Barbara sucht. Diese Abfrage wird jedoch nicht erfolgreich sein, weil das Wort *Telefonnummer* gar nicht im Dokument vorhanden ist.
- In der marktorientierten Informationsbeschaffung werden in Dokumenten Mitbewerber und die von ihnen gelieferten Waren erwähnt, oder die Website eines Mitbewerbers wechselte innerhalb der letzten Monate vom Verkauf einer Produktgruppe zu einer anderen. In diesem Fall kann der Benutzer eine Abfrage wie "Schmitt & Co. Waren" oder "Schmitt & Co. Waren Nov. 2004 bis Jan. 2005" eingeben. In der ersten Abfrage steht der Begriff *Waren* für ein Produkt oder einen Produktbereich, die Abfrage gibt jedoch nicht die Produkte zurück, die von Schmitt & Co. geliefert werden, weil sie nach dem Begriff *Waren* sucht. Dasselbe gilt für die Abfrage, die einen bestimmten Zeitraum enthält. Es ist fast unmöglich, mit der Schlüsselwortsuche einen Zeitraum abzufragen.

- Im Customer-Relationship-Management werden möglicherweise Probleme an Bremsen von Autos in Reparaturwerkstätten im Einzugsgebiet von Köln erwähnt. In den Berichten der Reparaturwerkstatt wird das Problem z. B. wie folgt beschrieben: ″Bremsbacke eingestellt wegen Leck in der Hydraulik″. Benutzer, die weitere Details abfragen, können z. B. eine Abfrage eingeben wie ″Reparaturwerkstätten für Bremsen in Niedersachsen″. Diese Abfrage gibt jedoch möglicherweise keinerlei Berichte zurück, in denen es um ″Bremsbacke eingestellt wegen Leck in der Hydraulik″ geht, da die Bezeichnungen *Bremsen* oder *Reparaturwerkstätten* als solche nicht in den Berichten vorkommen. Darüber hinaus wird in den Berichten möglicherweise nur der Name der Straße und des Stadtteils angegeben, in der sich die Werkstatt befindet, nicht die vollständige Adresse mit der Angabe von Köln.
- In der Forschung wird in Dokumenten ein bestimmtes Medikament beschrieben, das unter verschiedenen Markennamen weit verbreitet ist, und seine Beziehung zu mindestens einer Krankheit, die im selben Abschnitt genannt wird. Ein medizinisch weniger erfahrener Benutzer gibt möglicherweise eine der gängigen Bezeichnungen für das Medikament in seine Abfrage ein und hofft, dass daraufhin ein detaillierter Bericht über die verschiedenen Krankheiten, einschließlich der Symptome, angezeigt wird. Die Abfrage wird unter Umständen jedoch keine zufriedenstellenden Dokumente zurückgeben, da der gängige Begriff möglicherweise in den Dokumenten nicht verwendet wird. Außerdem wird häufig in den Dokumenten das Wort *Krankheit* gar nicht erwähnt, sondern nur der Name der Krankheit.

In diesen Beispielen werden Sie bei der Suche nach Ihren gewünschten Informationen in den umfangreichen Sammlungen an Informationsquellen, die es heute gibt, vor neue Herausforderungen gestellt, für die eine ausgereifte Analyse erforderlich ist, die die in der Unternehmenssuche angebotene Segmentierungsstufe und auf Basis von Wörterverzeichnissen ausgeführte Analyse übersteigt. Ein großer Teil der gesuchten Informationen ist in den Originaldokumenten nicht ausdrücklich gekennzeichnet oder hervorgehoben. Statt dessen muss der Dokumentinhalt analysiert werden, damit die betreffenden Konzepte erkannt und gefunden werden, z. B. benannte Entitäten wie Personen, Organisationen, Standorte, Einsatzmittel und Produkte und die möglichen zwischen diesen Entitäten bestehenden Beziehungen.

Die Informationen, die Sie in Textdokumenten aufspüren und extrahieren wollen, sind benutzer- und domänenspezifisch. Damit Sie Ihre eigenen Analysealgorithmen entwerfen und entwickeln können, stellt IBM Ihnen IBM Unstructured Information Management Architecture (UIMA) zur Verfügung, eine Architektur und Software-Rahmendefinition, mit deren Hilfe Sie die erweiterte Analysefunktionalität zum Finden von gesuchten Informationen in Dokumentobjektgruppen in der Unternehmenssuche erstellen können.

Zugehörige Konzepte

„Integration der benutzerdefinierten Textanalyse” auf Seite 3
Nachdem Sie Ihre benutzerdefinierte Analyse außerhalb der Unternehmenssuche mithilfe von Unstructured Information Management Architecture (UIMA) erstellt haben, können Sie die Analyselogik in die Unternehmenssuche integrieren. Verwenden Sie hierfür die Administrationskonsole für die Unternehmenssuche.

„Grundlegende Konzepte für die Verarbeitung der Textanalyse” auf Seite 4
Zu den grundlegenden, in der Verarbeitung der Textanalyse verwendeten Konzepten gehören Annotatoren, Analyseergebnisse, Komponentenstruktur, Typ, Typsystem sowie Annotationsstruktur und allgemeine Analysestruktur.

Integration der benutzerdefinierten Textanalyse

Nachdem Sie Ihre benutzerdefinierte Analyse außerhalb der Unternehmenssuche mithilfe von Unstructured Information Management Architecture (UIMA) erstellt haben, können Sie die Analyselogik in die Unternehmenssuche integrieren. Verwenden Sie hierfür die Administrationskonsole für die Unternehmenssuche.

UIMA ist eine offene Plattform, die für jede konzeptionell eindeutige Analysefunktion Komponenten angibt und sicherstellt, dass diese Komponenten ohne großen Aufwand wiederverwendet und kombiniert werden können.

Die erweiterte linguistische Analyse kann eine Kombination aus vielen verschiedenen Analysetasks enthalten. Die Analyse beginnt bei der Erkennung der Sprache und der Segmentierung, auf die eine Wortarterkennung und anschließend eine tiefgehende grammatikalische Syntaxanalyse folgt. Die letzten Tasks bestehen dann z. B. darin, dass der Zusammenhang zwischen bestimmten chemischen Substanzen und dem Auftreten bestimmter Symptome erkannt wird. Jeder Schritt im Analyseprozess ist von den Ergebnisses des vorherigen Schritts abhängig.

Die Analyselogik für jeden Schritt befindet sich in einem *Annotator*. Annotatoren werden zu einer Verarbeitungskette kombiniert, die für jedes Dokument der Objektgruppe eine Iteration ausführt, um neue Informationen zu erkennen und diese für die Downstream-Verarbeitung zu speichern.

Die Annotatoren zum Aufspüren und Darstellen des Analyseinhalts in Textdokumenten befinden sich in einer *Analysesteuerkomponente*, einem zentralen Konzept in UIMA. Eine Analysesteuerkomponente kann einen einzelnen Annotator enthalten, sie kann aber auch aus mehreren Steuerkomponenten zusammengesetzt sein, von denen jede Annotatoren enthält.

UIMA stellt nur die Grundbausteine bereit, mit denen Sie Ihre eigenen Analysesteuerkomponenten erstellen, testen und implementieren können. Es stellt Ihnen jedoch keine linguistische Analysefunktionalität in Form von vorkonfigurierten Analysesteuerkomponenten zur Verfügung, die Sie in Ihrer UIMA-Umgebung implementieren können. Die Verarbeitung auf linguistischer Basis, die in der Unternehmenssuche angewendet wird, ist jedoch als Annotatorengruppe verfügbar, mit der Sie in UIMA arbeiten können.

Wenn Sie mit UIMA arbeiten wollen, müssen Sie UIMA Software Development Kit installieren. Das Development-Kit ist auf der Site von IBM developerWorks verfügbar. Weitere Informationen finden Sie im WebSphere Information Integrator-Bereich unter http://www.ibm.com/developerworks/db2/zones/db2ii/. UIMA Software Development Kit (SDK) enthält eine Java-Implementierung der UIMA-Rahmendefinition für die Implementierung, die Beschreibung, die Zusammensetzung und den Einsatz von UIMA-Komponenten.

UIMA SDK stellt auch eine Gruppe von Tools und Dienstprogrammen für die Arbeit mit UIMA in einer Eclipse-basierten Entwicklungsumgebung (Eclipse-Plugins) bereit.

Informationen zu Eclipse finden Sie unter `www.eclipse.org`, und in der Dokumentation von UIMA finden Sie Anweisungen zur Installation von UIMA Software Development Kit in der interaktiven Eclipse-Entwicklungsumgebung.

Zugehörige Konzepte

„Linguistische Unterstützung der semantischen Suche" auf Seite 1
Die Unternehmenssuche bietet die linguistische Unterstützung der Suche für Textdokumente in den meisten indo-europäischen und asiatischen Sprachen an, einschließlich Japanisch.

„Grundlegende Konzepte für die Verarbeitung der Textanalyse"
Zu den grundlegenden, in der Verarbeitung der Textanalyse verwendeten Konzepten gehören Annotatoren, Analyseergebnisse, Komponentenstruktur, Typ, Typsystem sowie Annotationsstruktur und allgemeine Analysestruktur.

Grundlegende Konzepte für die Verarbeitung der Textanalyse

Zu den grundlegenden, in der Verarbeitung der Textanalyse verwendeten Konzepten gehören Annotatoren, Analyseergebnisse, Komponentenstruktur, Typ, Typsystem sowie Annotationsstruktur und allgemeine Analysestruktur.

Annotatoren enthalten die Logik zur Analyse eines Dokuments und zum Erkennen und Erfassen der beschreibenden Daten zum Gesamtdokument (die so genannten Metadaten) sowie zu Teilen des Dokuments. Diese beschreibenden Daten sind die *Analyseergebnisse*. Die Analyseergebnisse merken alle aufeinanderfolgenden Unterzeichenfolgen (Bereiche) des Textdokuments an. Im Idealfall entsprechen die Analyseergebnisse den Informationen, nach denen Sie suchen wollen.

Eine *Komponentenstruktur* ist die zugrunde liegende Datenstruktur, die ein Analyseergebnis darstellt. Eine Komponentenstruktur ist eine Attribut-Wert-Struktur. Jede Komponentenstruktur hat einen *Typ*, und jeder Typ hat eine bestimmte Menge gültiger Komponenten oder Attribute (Merkmale), ähnlich wie eine Java-Klasse. Komponenten haben einen Bereichstyp, der den Wertetyp angibt, den die Komponente aufweisen muss, zum Beispiel Zeichenfolge.

Der Textbereich "James Matthew Bloggs" kann z. B. von einer Annotation des Typs `Person` mit den Komponenten `personName`, `age`, `nationality` und `profession` eingeschlossen werden.

Das *Typsystem* definiert die Typen von Objekten (Komponentenstrukturen), die in einem Dokument erkannt werden können. Das Typsystem definiert alle möglichen Komponentenstrukturen nach Typen und Komponenten (Attributen), ähnlich wie eine Klassenhierarchie in Java. Sie können eine beliebige Anzahl unterschiedlicher Typen in einem Typsystem definieren. Ein Typsystem ist domänen- und anwendungsspezifisch.

Die meisten Textanalyseannotatoren stellen ihre Analyseergebnisse in der Form von *Annotationen* bereit. Annotationen sind eine besondere Art Komponentenstruktur, die für die linguistische Analyseverarbeitung bestimmt ist. Eine Annotation umfasst einen Teil des Eingabetexts und wird mithilfe von Anfangs- und Endpositionen im Eingabetext definiert.

So erstellt z. B. ein Annotator, der monetäre Ausdrücke erkennt, für den Text "100,55 US Dollar" eine Annotation des Typs `monetaryExpression`, der den Text mit der auf "$" gesetzten Komponente `currencySymbol` erfasst.

Alle Annotatoren von UIMA modellieren und speichern die Daten in Komponentenstrukturen.

Alle Komponentenstrukturen werden in einer zentralen Datenstruktur dargestellt, die als die *allgemeine Analysestruktur* bezeichnet wird. Der gesamte Datenaustausch erfolgt unter Verwendung der allgemeinen Analysestruktur.

Die allgemeine Analysestruktur enthält die folgenden Objekte:

- Das Textdokument
- Die Beschreibung des Typsystems, in der die Typen, Subtypen und ihre Komponenten angegeben sind
- Analyseergebnisse, die das Dokument oder Dokumentbereiche beschreiben
- Ein Indexrepository, das den Zugriff auf und die Iteration für die Analyseergebnisse unterstützt

Zugehörige Konzepte

„Linguistische Unterstützung der semantischen Suche" auf Seite 1
Die Unternehmenssuche bietet die linguistische Unterstützung der Suche für Textdokumente in den meisten indo-europäischen und asiatischen Sprachen an, einschließlich Japanisch.

„Integration der benutzerdefinierten Textanalyse" auf Seite 3
Nachdem Sie Ihre benutzerdefinierte Analyse außerhalb der Unternehmenssuche mithilfe von Unstructured Information Management Architecture (UIMA) erstellt haben, können Sie die Analyselogik in die Unternehmenssuche integrieren. Verwenden Sie hierfür die Administrationskonsole für die Unternehmenssuche.

Textanalysealgorithmen

UIMA Software Development Kit enthält APIs und Tools, mit denen Sie Annotatoren (Analysealgorithmen, einschließlich der Typsystembeschreibung) erstellen und in Analysesteuerkomponenten einbetten können.

Die UIMA-Dokumentation enthält einen Leitfaden, der wie ein Lernprogramm aufgebaut ist und mit dessen Hilfe Sie diese Komponenten erstellen können. Das Software-Development-Kit enthält Dienstprogramme zum Testen und Anzeigen Ihrer Ergebnisse und eine kleine semantische Suchmaschine zum Indexieren Ihrer Analyseergebnisse. Sie können auch eine erweiterte semantische Suche für die im Index gespeicherten Informationen ausführen.

Da UIMA Software Development Kit keine vorkonfigurierten Annotatoren bereitstellt und alle benutzerdefinierten Annotatoren, die Sie mithilfe von UIMA entwickeln und anschließend in die Unternehmenssuche integrieren, auf den Basisannotatoren der Unternehmenssuche aufbauen, können Sie das Basisannotatorpaket für Ihre UIMA-Umgebung verwenden. In der Dokumentation von UIMA finden Sie Informationen zum Implementieren der Funktionalität für Spracherkennung und Einteilung in Token vor der Ausführung der Textanalysealgorithmen in Ihrer UIMA-Umgebung.

Nachdem Sie Ihre Analysesteuerkomponenten unter Verwendung von UIMA Software Development Kit entwickelt und getestet haben, müssen Sie eine PEAR-Datei erstellen, damit Sie Ihre Algorithmen für eine Dokumentobjektgruppe in der Unternehmenssuche ausführen können. Diese Archivierungsdatei enthält alle erforderlichen Ressourcen für die Implementierung Ihrer benutzerdefinierten Analysefunktionalität als Analysesteuerkomponenten für die Unternehmenssuche. Die Vorgehensweise zum Erstellen eines Archivs ist in der Dokumentation von UIMA beschrieben, die im Software-Development-Kit bereitgestellt wird.

Das Archiv, das Sie für den Upload in die Unternehmenssuche erstellen, darf nur Ihre benutzerdefinierte Analyselogik enthalten. Es darf keinen Basisannotator der Unternehmenssuche enthalten, selbst wenn Ihre benutzerdefinierte Analyselogik anhand der Basisannotatorergebnisse erstellt wurde, da in der Unternehmenssuche die Basisannotatoren stets vor den benutzerdefinierten Analysen ausgeführt werden.

Wenn Sie lernen wollen, wie Sie eine semantische Suchlösung für die Unternehmenssuche konfigurieren und implementieren können, führen Sie das Lernprogramm aus. Sie finden es unter http://www.ibm.com/developerworks/db2/zones/db2ii/. Das Lernprogramm führt Sie durch die Schritte, die erforderlich sind, um benutzerdefinierte Textanalysealgorithmen für die Unternehmenssuche zu implementieren, und zeigt Ihnen, wie Sie die Analyseergebnisse in Abfragen verwenden können, um die Suchergebnisse zu verbessern.

Zugehörige Tasks

„Verwenden der Basisannotatoren für die Unternehmenssuche in UIMA" auf Seite 8
Sie können die Annotatoren des Basisannotatorpakets für die Unternehmenssuche verwenden, um neue Annotatoren innerhalb von UIMA Software Development Kit (SDK) zu entwickeln und um JDBC-Tabellen Analyseergebnisse zuzuordnen.

Workflow für die Integration der benutzerdefinierten Analyse

Algorithmen für die benutzerdefinierte Textanalyse erstellen und testen Sie mit UIMA Software Development Kit, und anschließend implementieren Sie sie in die Unternehmenssuche und führen sie für Dokumentobjektgruppen aus.

Gehen Sie wie folgt vor, um Analysealgorithmen zu entwickeln und in die Unternehmenssuche zu integrieren:

1. Planung und Design:
   a. Legen Sie fest, nach welchen Informationen Sie suchen wollen. Welche Dokumente wollen Sie abrufen? Welche Konzepte und Beziehungen sind für jede einzelne Suchtask erforderlich? So können z. B. die Namen von Produkten und Mitarbeitern erforderlich sein, um die allgemeine Suche auf der internen Website eines pharmazeutischen Unternehmens zu erweitern, während für den Bereich Forschung und Entwicklung Varianten von Medikamentennamen und die Beziehungen zwischen Medikament, Ursache und Heilung erforderlich sind.
   b. Geben Sie die Art Textanalyse an, die Sie benötigen, um die Informationen aus den zu durchsuchenden Dokumenten abzurufen.
   c. Wenn Ihre Objektgruppe XML-Dokumente enthält, müssen Sie entscheiden, ob Sie XML-Markup-Formatierung in Ihrer Lösung verwenden wollen. In der Unternehmenssuche gibt es zwei Möglichkeiten, wie Sie XML-Markup-Formatierung verwenden können:
      - Wenn Sie XML-Markup-Formatierung in Ihrer benutzerdefinierten Analyse verwenden können (z. B. wenn Ihre Dokumente die Elemente `<summary>` oder `<topic>` enthalten, die in einem Zusammenfassungs- oder Kategorisierungsannotator nützlich sein können), erstellen Sie XML-Elemente für die Datei für die Zuordnung der allgemeinen Analysestruktur und ordnen die Elemente dieser Datei zu.
      - Wenn Sie XML-Markup-Formatierung in Ihren Abfragen so verwenden wollen, wie es im Dokument angezeigt wird, müssen Sie die native XML-Zuordnung aktivieren.

d. Legen Sie fest, auf welche Textanalyseergebnisse, die in der allgemeinen Analysestruktur gespeichert sind, Sie über die semantische Suche zugreifen wollen. Erstellen Sie eine Datei für die Zuordnung der allgemeinen Analysestruktur zum Index.

e. Legen Sie fest, ob Sie Analyseergebnisse in einer relationalen Datenbank speichern wollen, z. B. um Trends und Beziehungen aufzuspüren, indem Sie Berichterstellungs- oder Data-Mining-Anwendungen verwenden. Erstellen Sie eine Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank.

f. Entwerfen Sie die semantische Suchanwendung. Legen Sie fest, wie der Benutzer der Suche die zusätzliche semantische Suchfunktionalität verwenden wird. Entwerfen Sie die Benutzeroberfläche.

2. Entwicklung: Aktivitäten mit UIMA Software Development Kit

   a. Definieren Sie die einzelnen Analyseschritte.

   b. Beschreiben Sie das Typsystem für Ihre Zuordnungen und Analysealgorithmen.

   c. Entwickeln Sie die Analysealgorithmen (Annotatoren) für jeden Analyseschritt, und betten Sie die Annotatoren in Analysesteuerkomponenten ein, indem Sie UIMA Software Development Kit verwenden. Erstellen Sie alle benutzerdefinierten Analysen unter Verwendung der Basisfunktionalität (Spracherkennung und Einteilung in Token) im Annotatorpaket für die Unternehmenssuche.

   d. Nachdem Sie die Analysealgorithmen in UIMA getestet haben, packen Sie die Analysesteuerkomponenten als PEAR-Datei in ein Verarbeitungsenginearchiv. Das Archiv darf nur Ihre Analysealgorithmen enthalten, nicht jedoch die linguistische Basisfunktionalität für die Unternehmenssuche.

      Wenn Sie eine Lösung für die Textanalyse entwerfen, kann diese mehrere Analysemodule enthalten, die in mehreren PEAR-Dateien bereitgestellt werden. UIMA stellt eine Möglichkeit bereit, zwei oder mehr PEAR-Dateien in eine PEAR-Datei zusammenzuführen, die Sie in die Unternehmenssuche hochladen und dort ausführen können. Die Funktion zum Zusammenführen von PEAR-Dateien gewährleistet, dass keine Benennungskollisionen auftreten, dass die Eingabe- und Ausgabefunktionalität ordnungsgemäß zusammengeführt werden und dass keine Parameter überschrieben werden, falls die Annotatordesktiptoren gleichnamige Parameter enthalten. In der Dokumentation von UIMA finden Sie Anweisungen zum Zusammenführen von PEAR-Dateien.

3. Implementierung: Aktivitäten für die Unternehmenssuche

   a. Übertragen Sie die Verarbeitungsenginearchivdatei (.pear) an die Unternehmenssuche. Geben Sie der Textanalysekomponente einen Namen, damit Sie diesen in der Unternehmenssuche verwenden können.

   b. Verknüpfen Sie mindestens eine Dokumentobjektgruppe mit Ihrer Textanalysekomponente.

   c. Übertragen Sie gegebenenfalls für jede Objektgruppe die Zuordnung der XML-Elemente zur allgemeinen Analysestruktur, die Sie für Ihre benutzerdefinierte Analyse definiert haben, und wählen Sie diese aus.

   d. Übertragen Sie gegebenenfalls für jede Objektgruppe die Zuordnung der allgemeinen Analysestruktur zur Datenbank, die Sie für Ihre benutzerdefinierte Analyse definiert haben, und wählen Sie diese aus.

   e. Übertragen Sie für jede Objektgruppe die Zuordnung der allgemeinen Analysestruktur zum Index, die Sie für die semantische Suche definiert haben, und wählen Sie diese aus.

f. Konfigurieren Sie gegebenenfalls Ihre benutzerdefinierte semantische Suchanwendung. Implementieren Sie z. B. Ihre browserbasierte Benutzeroberfläche zum Suchen in einem Anwendungsserver.

g. Führen Sie für die Dokumente in Ihrer semantischen Suchobjektgruppe, wie für eine stichwortbasierte Objektgruppe, eine Crawlersuche, eine syntaktische Analyse und eine Indexierung aus.

Zugehörige Tasks

„Verwenden der Basisannotatoren für die Unternehmenssuche in UIMA"
Sie können die Annotatoren des Basisannotatorpakets für die Unternehmenssuche verwenden, um neue Annotatoren innerhalb von UIMA Software Development Kit (SDK) zu entwickeln und um JDBC-Tabellen Analyseergebnisse zuzuordnen.

Verwenden der Basisannotatoren für die Unternehmenssuche in UIMA

Sie können die Annotatoren des Basisannotatorpakets für die Unternehmenssuche verwenden, um neue Annotatoren innerhalb von UIMA Software Development Kit (SDK) zu entwickeln und um JDBC-Tabellen Analyseergebnisse zuzuordnen.

Die Gruppe der Basisannotatoren umfasst folgende Elemente:

- **Sprach-ID-Annotator**

  Ermittelt die Sprache eines Dokuments. Informationen zu Funktionalität und Konfigurationsparametern finden Sie in der Deskriptordatei `jlangid.xml`.

- **Annotator für FROST-Wörterverzeichnissuche**

  Stellt Einteilung in Token und Satzerkennung auf der Basis der IBM LanguageWare-Wörterverzeichnisse bereit. Für Token werden zusätzliche linguistische Informationen generiert, z. B. Grundform oder Lemma. Informationen zu Funktionalität und Konfigurationsparametern finden Sie in der Deskriptordatei `jfrost.xml`.

- **Leerzeichentokenizer**

  Kann eine leerzeichenbasierte Einteilung in Token für alle in europäischen Sprachen abgefassten Dokumente oder andere durch Leerzeichen getrennte Scripts ausführen. Darüber hinaus kann der Annotator eine Einteilung in Token mithilfe von n-gram-Elementen für die folgenden Textscripts ausführen: Arabisch, Han, Hebräisch, Hiragana, Katakana, Lao, Mongolisch, Thailändisch, YI und Hangul. Diese Liste umfasst alle wichtigen asiatischen Textscripts und bedeutet, dass der Annotator Japanisch, Chinesisch und Koreanisch unterstützt.

  Informationen zu Funktionalität und Konfigurationsparametern finden Sie in der Deskriptordatei `jtok.xml`.

- **Annotator für reguläre Ausdrücke**

  Spürt Entitäten oder Informationsbereiche in einem Textdokument auf der Basis von regulären Ausdrücken auf. Sie können den Annotator für reguläre Ausdrücke anpassen, dass er die von Ihnen benötigten Textentitäten aufspürt, indem Sie Ihre eigenen Regeln definieren. Das Basisannotatorpaket enthält einen Beispielannotator für reguläre Ausdrücke zum Aufspüren von Telefonnummern, URL-Adressen und E-Mail-Adressen in Textdokumenten.

- **Allgemeine Analysestruktur für private Datenbankanwender**

  Der private Datenbankanwender einer allgemeinen Analysestruktur füllt eine relationale Datenbank mit bestimmten Textanalyseergebnissen.

Das Basisannotatorpaket für die Unternehmenssuche ist eine komprimierte Datei, die die Basisannotatoren für die Textanalyse, den Annotator für reguläre Ausdrücke und die allgemeine Analysestruktur für private Datenbankanwender enthält. Der Sprach-ID-Annotator, der Annotator für FROST-Wörterverzeichnissuche und der Leerzeichentokenizer sind die Basisannotatoren für die Textanalyse, die immer vor einer benutzerdefinierten Textanalyse ausgeführt werden, wenn Dokumente in der Unternehmenssuche syntaktisch analysiert werden.

Da die Basisannotatoren für die Textanalyse in der Unternehmenssuche immer vor einer benutzerdefinierten Textanalyse ausgeführt werden und die benutzerdefinierte Textanalyse auf der Ausgabe der Basisannotatoren basiert, können Sie diese Annotatoren verwenden, um Ihre benutzerdefinierten Annotatoren in Ihrer UIMA-Umgebung zu entwickeln und zu testen.

Der Annotator für reguläre Ausdrücke und die allgemeine Analysestruktur für private Datenbankanwender sind zusätzliche Optionen, die Sie während der Konfiguration Ihrer Textverarbeitungsoptionen in der Administrationskonsole für die Unternehmenssuche auswählen können. Sie können sie auch in UIMA verwenden. Für eine erweiterte Anpassung des Annotators für reguläre Ausdrücke ist es empfehlenswert, die im Produktumfang von UIMA SDK enthaltenen Tools für die Annotatoranpassung zu verwenden.

Damit Sie diese Annotatoren in UIMA ausführen können, muss UIMA Software Development Kit (SDK) installiert sein. Es ist auf der Website von IBM developerWorks unter http://www.ibm.com/developerworks/db2/zones/db2ii/ verfügbar.

Gehen Sie wie folgt vor, um das Annotatorpaket in Ihrer Installation von UIMA SDK zu installieren:

1. Suchen Sie in Ihrer Installation der Unternehmenssuche (OmniFind Enterprise Edition) im Verzeichnis *ES_INSTALL_ROOT*/packages/uima nach dem Annotatorpaket OF_base_annotators.zip.
2. Kopieren Sie die komprimierte Datei in das Stammverzeichnis Ihrer UIMA SDK-Installation.
3. Extrahieren Sie die komprimierte Datei, um die Basisannotatordateien für die Unternehmenssuche der angegebenen Verzeichnisstruktur Ihrer UIMA SDK-Installation hinzuzufügen. Die Datei tt_core_typesystem.xml wird überschrieben. Wenn Sie die alte Version dieser Datei aufbewahren wollen, speichern Sie sie, bevor Sie die komprimierte Datei extrahieren.
4. Wenn Sie den Klassenpfad festlegen möchten, öffnen Sie das Script setUIMAClasspath im Verzeichnis bin, und fügen Sie am Ende des Scripts eine Zeile hinzu, die das Script OFAnnotEnv startet.
5. Wenn Sie benutzerdefinierte Typen oder spezifische Typen der Unternehmenssuche in UIMA verwenden wollen, finden Sie die entsprechenden Informationen zum Definieren dieser Typen in der Dokumentation von UIMA SDK.

Nachdem Sie das Basisannotatorpaket installiert haben, finden Sie die Deskriptordateien im Verzeichnis *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine. Die Datei of_tokenization.xml listet die Basisannotatoren für die Textanalyse (Sprach-ID-Annotator, Annotator für FROST-Wörterverzeichnissuche und Leerzeichentokenizer) in der Reihenfolge auf, in der sie in der Unternehmenssuche verwendet werden.

Die Deskriptordateien enthalten dieselben Konfigurationswerte, die in der Unternehmenssuche verwendet werden. Sie können Werte zu Debugging-Zwecken in UIMA SDK ändern. Sie sollten jedoch nicht die Deskriptordateien in Ihrem System

für die Unternehmenssuche ändern. Änderungen an diesen Dateien können möglicherweise zu Systeminstabilität oder Leistungsproblemen führen.

Das Basisannotatorpaket für die Unternehmenssuche enthält nur die Wörterverzeichnisse, die für die Verarbeitung englischsprachiger Dokumente erforderlich sind. Wenn Sie andere Sprachen in Ihrer Entwicklungsumgebung verarbeiten wollen, gehen Sie wie folgt vor:

1. Suchen Sie in der Installation Ihrer Unternehmenssuche unter *ES_INSTALL_ROOT*/configurations/parserservice/jediidata/frost/resources nach den Wörterverzeichnissen für die Unternehmenssuche.
2. Kopieren Sie den Inhalt des Verzeichnisses in Ihre lokale Installation von UIMA SDK unter *UIMA_SDK_INSTALL*/data/frost/resources.

Gehen Sie wie folgt vor, um zu prüfen, ob das Annotatorpaket erfolgreich installiert wurde:

1. Öffnen Sie CAS-Visual Debugger (CVD) im folgenden Verzeichnis: *UIMA_SDK_INSTALL*/bin/cvd[.bat/.sh].
2. Klicken Sie **Run → load TAE** an.
3. Wählen Sie die Kennungsdatei of_tokenization.xml für die Textanalysesteuerkomponente im Verzeichnis *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine aus.
4. Laden Sie ein Beispieldokument, und führen Sie die Textanalysesteuerkomponente aus. In CVD werden Annotationen des Typs uima.tt.TokenAnnotation angezeigt.

Wenn Sie einen beliebigen Basisannotator für die Textanalyse vor Ihren benutzerdefinierten Annotatoren in Ihrer Entwicklungsumgebung ausführen und Ihre benutzerdefinierten Annotatoren Typen verwenden, die von der Basistextanalyse definiert werden, fügen Sie dem Abschnitt für das Typsystem Ihrer benutzerdefinierten Annotatorkennung einen Verweis auf die Datei tt_core_typesystem hinzu. Die Datei tt_core_typesystem befindet sich im Verzeichnis *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine. In der Datei jtok.xml im Verzeichnis analysis_engine finden Sie ein Beispiel, wie einer Deskriptordatei Verweise hinzugefügt werden.

Zugehörige Tasks

„Anzeigen der Ergebnisse des Basisannotators und der angepassten Textanalyse" auf Seite 14
Wenn Sie die Analyseergebnisse anzeigen wollen, die nach der Syntaxanalyse von einem beliebigen Annotator in der Unternehmenssuche erzeugt wurden, müssen Sie die Merkmale der Dokumentobjektgruppe aktualisieren, damit eine lesbare XML-Version der in der allgemeinen Analysestruktur gespeicherten Analyseergebnisse erzeugt wird.

„Aktivieren der einfachen semantischen Suche mithilfe des Annotators für reguläre Ausdrücke" auf Seite 91
Wenn Sie die einfache semantische Suche unter Verwendung von Synonymen aktivieren wollen, müssen Sie Ihrem System für die Unternehmenssuche den Annotator für reguläre Ausdrücke, die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index und das Beispielsynonymverzeichnis hinzufügen und Ihrer Objektgruppe diese Ressourcen zuordnen.

„Verwenden der allgemeinen Analysestruktur für private Datenbankanwender in UIMA" auf Seite 11
Bevor Sie die allgemeine Analysestruktur für private Datenbankanwender in UIMA verwenden können, müssen Sie die Änderungen an der Deskriptordatei

für den privaten Anwender vornehmen und die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank erstellen.

„Verwenden des Annotators für reguläre Ausdrücke in UIMA" auf Seite 13
Verwenden Sie den Annotator für reguläre Ausdrücke, um Entitäten oder Informationseinheiten in einem Textdokument aufzuspüren. Sie können den Annotator an Ihre betreffende Domäne anpassen, damit er Ihren Anforderungen an die Suche entspricht.

Verwenden der allgemeinen Analysestruktur für private Datenbankanwender in UIMA

Bevor Sie die allgemeine Analysestruktur für private Datenbankanwender in UIMA verwenden können, müssen Sie die Änderungen an der Deskriptordatei für den privaten Anwender vornehmen und die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank erstellen.

Bevor Sie die allgemeine Analysestruktur für private Datenbankanwender in Ihrer UIMA-Umgebung ausführen können, müssen Sie die folgenden Schritte ausführen:

1. Öffnen Sie die XML-Deskriptordatei `cas2jdbc.xml` im Verzeichnis *UIMA_SDK_INSTALL*`/docs/examples/descriptors/cas_consumer`. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden.
2. Modifizieren Sie den Parameter **mappingFile** so, dass er den absoluten Pfad enthält, in dem sich die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank befindet, z. B. `D:\temp\MyMapping.xml`.
3. Modifizieren Sie den Parameter **docMetadata_Type** so, dass er den UIMA-Typ angibt, aus dem alle Metadaten für die integrierten Komponenten abgerufen werden, z. B. `uima.tcas.DocumentAnnotation`.
4. Modifizieren Sie den Parameter **docId_Feature**, und fügen Sie ihm die Komponente oder den Komponentenpfad zu dem Metadatentyp hinzu, aus dem die numerische ID (ganzzahliger Typ) eines Dokuments abgerufen wird. Dies ist für alle integrierten Komponenten erforderlich, die die ID benötigen, z. B. `docId()`, `uniqueId()`, `objectId()` und `fsId()`.
5. Geben Sie den Parameter **encryptionClass** nicht an, da dieser nur in der Unternehmenssuche verwendet wird, um es der allgemeinen Analysestruktur für private Datenbankanwender zu ermöglichen, mit verschlüsselten Zuordnungsdateien zu arbeiten.
6. Speichern Sie die Datei.
7. Kopieren Sie die EMF-Bibliotheksdateien (`common.jar`, `ecore.jar` und `ecore.xmi.jar`) aus dem Verzeichnis `lib` der Installation der Unternehmenssuche in das Verzeichnis `lib` der UIMA-Installation. Die Datei `cc_cas2jdbc.jar` befindet sich bereits im Verzeichnis `lib` der UIMA-Installation.
8. Erstellen Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank, die definiert, welche Textanalyseergebnisse in einer Datenbank gespeichert werden sollen. Sie können die Zuordnungsdatei `sampleMapping.xml` unter *UIMA_SDK_INSTALL*`/docs/examples/descriptors/cas_consumer` als Muster für die Erstellung einer eigenen Zuordnungsdatei verwenden.

   Verwenden Sie die XML-Schemadatei mit dem Namen `CasToJDBCMapping.xsd` im Verzeichnis *UIMA_SDK_INSTALL*`/docs/examples/descriptors/cas_consumer`, um die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank zu überprüfen. Damit die Leistung nicht verringert wird, führt die allgemeine Analysestruktur für private Datenbankanwender keine Prüfung der Zuordnungsdatei aus. Die Prüfung müssen Sie selbst vornehmen.

Eine Beschreibung der Ausführung des privaten Anwenders in UIMA finden Sie in der Dokumentation von UIMA.

Das folgende Beispiel zeigt, wie Sie die obligatorischen Parameter im Deskriptor definieren müssen:

```
 ...
 <nameValuePair>
  <name>mappingFile</name>
  <value>
   <string>D:/temp/MyMapping.xml</string>
  </value>
 </nameValuePair>
<nameValuePair>
  <name>docMetadata_Type</name>
  <value>
   <string>uima.tcas.DocumentAnnotation</string>
  </value>
 </nameValuePair>
<nameValuePair>
  <name>docId_Feature</name>
  <value>
   <string>end</string>
  </value>
 </nameValuePair>

...
```

Die Tabelle enthält die Konfigurationsparameter in der Reihenfolge, in der sie in der Deskriptordatei aufgeführt sind, und gibt an, welche obligatorisch sind:

Tabelle 1. Die Konfigurationsparameter in der Deskriptordatei der allgemeinen Analysestruktur für private Datenbankanwender

Parameter	**Beschreibung**	**Obligatorisch**
mappingFile	Dies ist der absolute Pfad zur Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank, z. B. `D:/temp/sample.xml`. Unter Windows verwenden Sie "/" als Pfadtrennzeichen.	Wahr
encryptionClass	Geben Sie diesen Parameter nicht an, da er nur in der Unternehmenssuche verwendet wird, um es der allgemeinen Analysestruktur für private Datenbankanwender zu ermöglichen, mit verschlüsselten Zuordnungsdateien zu arbeiten.	Falsch
docMetadata_Type	Der UIMA-Typ, aus dem alle Metadaten für integrierte Komponenten abgerufen werden.	Wahr

Tabelle 1. Die Konfigurationsparameter in der Deskriptordatei der allgemeinen Analysestruktur für private Datenbankanwender (Forts.)

Parameter	Beschreibung	Obligatorisch
docId_Feature	Die Komponente oder der Komponentenpfad für den Metadatentyp, aus dem die numerische Dokument-ID abgerufen wird. Der Wert muss ganzzahlig sein und ist für alle integrierten Komponenten erforderlich, die die ID benötigen, z. B. `uniqeId()`, `objectId()` und `fsId()`.	Wahr
docUri_Feature	Die Komponente oder der Komponentenpfad für den Metadatentyp, aus dem die URI des Dokuments stammt. Der Typ muss `string` sein.	Falsch
IsCompleted_Feature	Die Komponente oder der Komponentenpfad für den Metadatentyp, der markiert, ob das aktuelle Dokument auf mehrere allgemeine Analysestrukturen aufgeteilt wird.	Falsch
chunkNumber_Feature	Die Komponente oder der Komponentenpfad für den Metadatentyp, der die nachfolgende Nummer des aktuellen Chunks angibt.	Falsch

Verwenden des Annotators für reguläre Ausdrücke in UIMA

Verwenden Sie den Annotator für reguläre Ausdrücke, um Entitäten oder Informationseinheiten in einem Textdokument aufzuspüren. Sie können den Annotator an Ihre betreffende Domäne anpassen, damit er Ihren Anforderungen an die Suche entspricht.

Damit Sie den Beispielannotator für reguläre Ausdrücke zum Aufspüren von Telefonnummern, URL-Adressen und E-Mail-Adressen ausführen oder den Beispielannotator als Grundlage für die Erstellung Ihrer eigenen angepassten Version eines Annotators für reguläre Ausdrücke in Ihrer UIMA-Umgebung verwenden können, brauchen Sie Folgendes:

1. Annotatordeskriptor für reguläre Ausdrücke im Verzeichnis *UIMA_SDK_INSTALL*`/docs/examples/descriptors/analysis_engine`
2. Beispielregelsatz und die Beschreibung des Typsystems im Verzeichnis *UIMA_SDK_INSTALL*`/docs/examples/regex`
3. Beispieltextdatei, auf die der Beispielregelsatz angewendet werden kann, im Verzeichnis *UIMA_SDK_INSTALL*`/docs/data` und mit dem Namen `of_sample_regex.txt`

Die Beschreibung der Ausführung des Annotators in UIMA finden Sie in der Dokumentation von UIMA.

Anzeigen der Ergebnisse des Basisannotators und der angepassten Textanalyse

Wenn Sie die Analyseergebnisse anzeigen wollen, die nach der Syntaxanalyse von einem beliebigen Annotator in der Unternehmenssuche erzeugt wurden, müssen Sie die Merkmale der Dokumentobjektgruppe aktualisieren, damit eine lesbare XML-Version der in der allgemeinen Analysestruktur gespeicherten Analyseergebnisse erzeugt wird.

Informationen zu dieser Task

Verwenden Sie die XML-Serialisierung der in der allgemeinen Analysestruktur gespeicherten Analyseergebnisse des Annotators für Folgendes:

- Anzeigen der Ergebnisse nach der Syntaxanalyse und vor der Verarbeitung der Basisannotatoren.
- Anzeigen der Ergebnisse, nachdem diese syntaktisch analysiert und mit Token versehen wurden (Ausführen der Basisannotatoren der Unternehmenssuche). Dies kann Ihnen helfen, die Struktur der Eingabedaten für jede beliebige benutzerdefinierte Analyse zu ermitteln, die Sie entwickeln wollen und die stets nach den Basisannotatoren ausgeführt wird.
- Anzeigen und Prüfen der Ergebnisse einer benutzerdefinierten Analyse, die für eine kleinere Dokumentobjektgruppe in der Unternehmenssuche zu Testzwecken ausgeführt wurde, bevor Sie sich dafür entscheiden, die Analyse für eine vollständige Objektgruppe auszuführen.

Die XML-Serialisierung erzeugt zwei Ergebnissätze:

- Die Ergebnisse nach der Syntaxanalyse. Hierzu gehören Feldzuordnungen und Dokumentmetadaten.
- Die Ergebnisse nach der Syntaxanalyse, der Aufbereitung und, falls ausgewählt, der benutzerdefinierten Textanalyse. Hierzu gehören alle erzeugten Token und Annotationen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine lesbare XML-Version der Analyseergebnisse zu erzeugen:

1. Öffnen Sie die Datei `collection.properties` in *`ES_NODE_ROOT`*`/master_config/<objektgruppen-id>.parserdriver`, bevor Sie mit der Syntaxanalyse der Dokumente beginnen.
2. Wenn Sie die Ergebnisse nach der Syntaxanalyse anzeigen wollen, fügen Sie der Datei `collection.properties` die Zeile `trevi.parser.dumpXCas=<your_dump_directory>` hinzu.

   Ihr Speicherauszugsverzeichnis muss bereits vorhanden sein.

   a. Wählen Sie den gewünschten Ausgabetyp aus. Die Ausgabe enthält immer die für die Ergebnisse der Syntaxanalyse verwendete Typsystembeschreibung namens `OmniFindParserTypeSystem.xml`. Fügen Sie eine der folgenden Zeilen hinzu:

      - Zur Anzeige der Ausgabe der letzten 25 verarbeiteten Dateien fügen Sie die Zeile `trevi.parser.maxXCasFileCount=25` hinzu.

        Sie können die Anzahl Dateien selbst bestimmen, obwohl es empfohlen wird, diesen Wert nicht zu hoch zu setzen.

Denken Sie daran, dass der Dateiausgabepuffer fortwährend überschrieben wird, nachdem die maximale Puffergröße erreicht ist. Dies bedeutet auch, dass das Dokument mit der höchsten Nummer nicht das zuletzt verarbeitete sein muss.

Die Ausgabe enthält die folgenden Dateien: `OmniFindParserXCasDump1.xml`, gefolgt von `OmniFindParserXCasDump2.xml` usw., bis 25 Dateien aufgelistet sind.

- Wenn Sie die Ausgabe bestimmter Dokumente anzeigen wollen, fügen Sie den URI des Dokuments hinzu: `trevi.parser.xCasURI.1=file://home/test/file1.txt`.

  Sie können eine beliebige Anzahl Dokumente hinzufügen, aber die Dokumente müssen in aufsteigender Reihenfolge beginnend mit 1 ohne Lücken zwischen den Nummern nummeriert sein. Das zweite Dokument ist z. B. `trevi.parser.xCasURI.2=file://home/test/file2.txt` und das dritte Dokument `trevi.parser.xCasURI.3=file://home/test/file3.txt`.

  Die Ausgabe enthält die folgenden Dateien: `OmniFindParserXCasDumpURI_1.xml`, `OmniFindParserXCasDumpURI_2.xml` und so weiter, bis alle von Ihnen aufgelisteten Dateinamen aufgeführt sind.

3. Wenn Sie die Ergebnisse nach der Aufbereitung anzeigen wollen, fügen Sie die folgende Zeile hinzu:
   `trevi.tokenizer.dumpXCas=<ihr_speicherauszugsverzeichnis>`

   Auch in diesem Fall muss Ihr Speicherauszugsverzeichnis bereits vorhanden sein.

   a. Wählen Sie den gewünschten Ausgabetyp aus. Die erstellte Ausgabe enthält auch immer die Typsystembeschreibung `OmniFindTypeSystem.xml`, die zum Einfügen der Token und für die Textanalyseergebnisse verwendet wurde. Fügen Sie eine der folgenden Zeilen hinzu:

      - Zur Anzeige der Ausgabe der letzten 25 verarbeiteten Dateien fügen Sie die Zeile `trevi.tokenizer.maxXCasFileCount=25` hinzu.

        Sie können die Anzahl Dateien selbst bestimmen, obwohl es empfohlen wird, diesen Wert nicht zu hoch zu setzen.

        Denken Sie daran, dass der Dateiausgabepuffer fortwährend überschrieben wird, nachdem die maximale Puffergröße erreicht ist. Dies bedeutet auch, dass das Dokument mit der höchsten Nummer nicht das zuletzt verarbeitete sein muss.

        Die Ausgabe enthält die folgenden Dateien: `OmniFindXCasDump1.xml`, gefolgt von `OmniFindXCasDump2.xml` usw., bis 25 Dateien aufgelistet sind.

      - Wenn Sie die Ausgabe bestimmter Dokumente anzeigen wollen, fügen Sie den URI des Dokuments hinzu: `trevi.tokenizer.xCasURI.1=file://home/test/file1.txt`.

        Sie können eine beliebige Anzahl Dokumente hinzufügen, aber die Dokumente müssen in aufsteigender Reihenfolge beginnend mit 1 ohne Lücken zwischen den Nummern nummeriert sein. Das zweite Dokument ist z. B. `trevi.tokenizer.xCasURI.2=file://home/test/file2.txt` und das dritte Dokument `trevi.tokenizer.xCasURI.3=file://home/test/file3.txt`.

        Die Ausgabe enthält die folgenden Dateien: `OmniFindXCasDumpURI_1.xml`, `OmniFindXCasDumpURI_2.xml` und so weiter, bis alle von Ihnen aufgelisteten Dateinamen aufgeführt sind.

In der Unternehmenssuche können Sie XCAS Annotation Viewer verwenden, um den Inhalt von XML-Dateien anzuzeigen. Starten Sie XCAS Annotation Viewer,

indem Sie die Scriptdatei `xcasAnnotationViewer` ausführen, die sich im Verzeichnis `ES_INSTALL_ROOT/bin` befindet. Sie werden aufgefordert, die folgenden Angaben zu machen:

- Ihr Speicherauszugsverzeichnis, in dem die Ergebnisse nach der Syntaxanalyse oder dem Einfügen der Token gespeichert werden
- Die Deskriptordatei, `OmniFindParserTypeSystem.xml` (für die Ergebnisse nach der Syntaxanalyse) oder `OmniFindTypeSystem.xml` (für die syntaktisch analysierten und mit Token versehenen Ergebnisse), in Ihrem Speicherauszugsverzeichnis.

Wenn Sie ein Dokument aus der Liste auswählen, werden die Analyseergebnisse für das betreffende Dokument angezeigt. Wenn Sie eine hervorgehobene Annotation im Dokument anklicken, werden die Details der Annotation angezeigt.

Typsystembeschreibung

Das Typsystem definiert die Typen von Objekten und Ihre Merkmale (oder Komponenten), die in einer allgemeinen Analysestruktur instanziiert werden können.

Jede Analysesteuerkomponente hat ihre eigenen Typsystembeschreibungen, die die Eingabeanforderungen und Ausgabetypen für die Annotatoren in der Analysesteuerkomponente beschreiben. Typsystembeschreibungen sind spezifisch für die Anwendungsdomäne.

Ein Typsystem enthält die Definitionen von Typen, ihre Merkmale und die Hierarchie der Einfachvererbung von Typen. Eine allgemeine Analysestruktur muss einem bestimmten Typsystem entsprechen.

Die Typen und Komponenten, die in der Typsystembeschreibung definiert sind, müssen auch in allen Zuordnungsdateien verwendet werden, die der Dokumentanalyse zugeordnet sind, einschließlich der Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur, der Datei für die Zuordnung der allgemeinen Analysestruktur zum Index und der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank.

Die Typsystembeschreibung eines Annotators kann Teil des Deskriptors des Annotators sein, oder sie kann in einer separaten Deskriptordatei für das Typsystem enthalten sein. Manchmal ist sie auch Teil des Deskriptors eines anderen Annotators, der in derselben Analysesteuerkomponente enthalten ist.

Wenn Sie Ihre Analysesteuerkomponente in Ihrer UIMA-Umgebung vollständig entwickelt und getestet haben, enthält die von Ihnen erstellte und in die Unternehmenssuche hochgeladene Archivierungsdatei (PEAR-Datei) neben Ihrer Analyselogikdatei auch Ihre Typsystembeschreibung.

Die Basisannotatoren für die Unternehmenssuche verwenden drei Typsystembeschreibungen: eine Beschreibung des Kerntypsystems, die immer vorhanden ist, und zwei weitere, die Sie optional aktivieren können, um in der Dokumentverarbeitung für die Objektgruppe von der Basisanalyse in den erweiterten Analysemodus zu wechseln. Ob Sie nur eine oder beide erweiterten Typsystembeschreibungen einschließen müssen, hängt davon ab, welche zusätzlichen Analyseergebnisse aus der Textverarbeitung Sie während der Verarbeitung der Basisanalyse aufnehmen wollen.

Sie können den erweiterten Analysemodus aktivieren, indem Sie eines oder beide der erweiterten Typsysteme einschließen. Im erweiterten Analysemodus sind wäh-

rend der Verarbeitung der Basisanalyse zusätzliche Analysefunktionen verfügbar, die auch in der allgemeinen Analysestruktur gespeichert werden. Wenn Sie z. B. weitere Informationen zu einem Token (weitere Komponenteninformationen) wünschen, wie z. B. alle möglichen Lemmata für das Token, ob es sich bei dem Lemma um ein Stoppwort handelt, die Wortart des Lemmas oder bestimmte Komponenten für die morphologische Verarbeitung (auch für Japanisch) müssen Sie den erweiterten Analysemodus aktivieren.

Zugehörige Tasks

„Wechsel vom Basisanalysemodus in den erweiterten Analysemodus"
Wenn Sie die Verarbeitung von Dokumentobjektgruppen, die von den Basisannotatoren der Unternehmenssuche im Basisanalysemodus ausgeführt wird, im erweiterten Analysemodus ausführen wollen, müssen Sie die Typsystembeschreibungen für den erweiterten Modus aufnehmen.

Zugehörige Verweise

„Für die Unternehmenssuche definierte Typen und Komponenten" auf Seite 18
Das für die Unternehmenssuche definierte Typsystem umfasst die Handhabung von Dokumentmetadaten und eine linguistische Basisanalyse.

Wechsel vom Basisanalysemodus in den erweiterten Analysemodus

Wenn Sie die Verarbeitung von Dokumentobjektgruppen, die von den Basisannotatoren der Unternehmenssuche im Basisanalysemodus ausgeführt wird, im erweiterten Analysemodus ausführen wollen, müssen Sie die Typsystembeschreibungen für den erweiterten Modus aufnehmen.

Einschränkungen

Es gibt zwei Typsystembeschreibungen, die Sie auswählen können, um den erweiterten Analysemodus zu aktivieren:

- Die Beschreibung `tt_extension_typesystem`, die ausführlichere lexikalisch typisierte Komponenteninformationen zu Lemmata enthält.
- Die Beschreibung `dlt_extension_typesystem`, die zusätzliche morphologische Komponenten und spezielle lexikalische Typen enthält.

Vorgehensweise

Gehen Sie wie folgt vor, um von der Basisverarbeitung der Objektgruppe in den erweiterten Analysemodus zu wechseln:

1. Öffnen Sie die Datei `tt_core_typesystem.xml` im Verzeichnis *ES_NODE_ROOT*`/master_config/`*objektgruppen-id*`.parserdriver/specifiers`. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden.
2. Entfernen Sie die Kommentartags, die das Element `<import>` einschließen, im Abschnitt `<imports>` so, dass sie entweder eine Beschreibungsdatei oder beide Beschreibungsdateien des Erweiterungstypsystems einschließen.

   ```
   <imports>
   <!-- imports the tt_extension_typsystem for advanced analysis -->
   <!-- <import location="tt_extension_typesystem.xml"/>-->
   <!-- imports the dlt extension typesystem -->
   <!-- <import location="dlt_extension_typesystem.xml"/> -->
   </imports>
   ```
3. Öffnen Sie die beiden Deskriptordateien `jfrost.xml` und `jfrost_ngram.xml`, und ändern Sie den Inhalt des Elements `<outputs>` so, dass die Typen (in einem

Element <type>) und Komponenten (in einem Element <feature>) eingeschlossen werden, die im Element <description> im Abschnitt <capabilities> aufgelistet werden, den Sie bei der Analyse aufnehmen möchten. Speichern Sie Ihre Änderungen.

4. Öffnen Sie die Deskriptordatei jtok.xml, und ändern Sie den Inhalt des Elements <outputs> so, dass die Komponenten (in einem Element <feature>) eingeschlossen werden, die im Element <description> im Abschnitt <capabilities> aufgelistet werden, den Sie bei der Analyse aufnehmen möchten. Speichern Sie Ihre Änderungen.
5. Öffnen Sie die Deskriptordatei es_tok_no_stw.xml, und ändern Sie auch darin den Inhalt des Elements <outputs> so, dass die Komponenten (in einem Element <feature>) eingeschlossen werden, die im Element <description> im Abschnitt <capabilities> aufgelistet werden, den Sie bei der Analyse aufnehmen möchten. Speichern Sie Ihre Änderungen.
6. Wenn Sie in den erweiterten Analysemodus wechseln, müssen Sie Ihre Dokumentobjektgruppe erneut syntaktisch analysieren.

Zugehörige Konzepte

„Typsystembeschreibung" auf Seite 16
Das Typsystem definiert die Typen von Objekten und Ihre Merkmale (oder Komponenten), die in einer allgemeinen Analysestruktur instanziiert werden können.

Zugehörige Verweise

„Für die Unternehmenssuche definierte Typen und Komponenten"
Das für die Unternehmenssuche definierte Typsystem umfasst die Handhabung von Dokumentmetadaten und eine linguistische Basisanalyse.

Für die Unternehmenssuche definierte Typen und Komponenten

Das für die Unternehmenssuche definierte Typsystem umfasst die Handhabung von Dokumentmetadaten und eine linguistische Basisanalyse.

Die in der Unternehmenssuche verwendeten Typen sind in drei separaten Beschreibungsdateien für das Typsystem definiert. An erster Stelle steht die Beschreibungsdatei für das Typsystem, die die Kerntypen enthält, die immer für die gesamte grundlegende linguistische Analyse erforderlich sind. Danach kommen die Typsystembeschreibungen, die erweiterte linguistische Komponenten definieren, die in der Regel nur für den erweiterten Analysemodus erforderlich sind.

Eine linguistische Basisanalyse in Form von Erkennung der Sprache und die Segmentierung eines Dokuments findet immer während der Dokumentindexierung statt, unabhängig davon, ob eine benutzerdefinierte Analyse ausgewählt ist oder nicht. Während der Basisanalyse wird die Beschreibung tt_core_typesystem verwendet, und der allgemeinen Analysestruktur werden die folgenden Informationen hinzugefügt, die Sie in der nachfolgenden benutzerdefinierten Analyse verwenden können:

- Dokumentmetadaten des Typs com.ibm.es.tt.DocumentMetaData.
- Informationen zur Dokumentstruktur, wie z. B. Annotationen für Sätze und Abschnitte der Typen uima.tt.SentenceAnnotation und uima.tt.ParagraphAnnotation.
- Lexikalische Annotationen, wie z. B. Token und zusammengesetzte Begriffe des Typs uima.tt.TokenAnnotation.

Die Beschreibung tt_core_typesystem ist für die meisten Textanalyseverarbeitungen adäquat.

Wenn Sie die Verarbeitung der Objektgruppe im erweiterten Analysemodus ausführen wollen, können Sie die folgenden zwei Typsysteme einschließen. Diese Typsysteme enthalten in erster Linie zusätzliche Komponenten, die während der grundlegenden Verarbeitung auf linguistischer Basis nicht erstellt werden.

- Das Typsystem tt_extension_typesystem enthält weitere Informationen zu Token, Lemmata sowie Abschnitten und Satzkomponenten.
- dlt_core_typesystem enthält einige der erweiterten IBM LanguageWare-Annotationstypen, z. B. URLs und Adressen. Weiterhin enthält es morphologische Komponenten, die nur selten verwendet werden.

tt_core_typesystem

Die folgenden Typen und Komponenten sind in der Beschreibung tt_core_typesystem definiert:

uima.tcas.DocumentAnnotation
Die Dokumentannotation enthält Dokumentmetadaten und hat die folgenden Komponenten:

- categories, der Dokumentkategorien von einer Textkategorisierungsfunktion hinzugefügt werden. Jede hinzugefügte Kategorie hat den Typ com.tt.CategoryConfidencePair.
- languageCandidates für die automatische Erkennung der Dokumentsprachen während der Syntaxanalyse. Die Sprachen werden einer Liste des Typs com.tt.LanguageConfidencePair hinzugefügt, in der die Sprache mit der höchsten Wahrscheinlichkeit an erster Stelle steht.
- id mit der Dokument-ID, wie z. B. der URL.

uima.tt.TTAnnotation
Das ist der Stammelementtyp für Annotationen, die in tt_core_typesystem definiert sind. Der übergeordnete Typ ist uima.tcase.Annotation. Das Element hat die folgenden Typen:

uima.tt.DocStructureAnnotation
Annotationen zur Dokumentstruktur. Das Element hat die folgenden Subtypen:

uima.tt.SentenceAnnotation
Sätze

uima.tt.ParagraphAnnotation
Dokumentabschnitt

uima.tt.LexicalAnnotation
Lexikalische Annotationen wie z. B. Token oder Mehrwortbegriffe. Das Element hat die folgenden Subtypen:

uima.tt.TokenLikeAnnotation
Annotationen zu einzelnen Token, die die folgenden Komponenten aufweisen können:

- tokenProperties mit den Tokenmerkmalen
- lemma mit dem Lemma oder Wortstamm des Begriffs
- normalizedCoveredText mit der normalisierten Darstellung des betreffenden Texts

Dieser Annotationstyp hat die folgenden Subtypen:

uima.tt.TokenAnnotation
Tatsächliche Token, die von zusammengesetzten Teilen zu unterscheiden sind.

uima.tt.CompPartAnnotation
Die zusammengesetzten Teile eines Begriffs.

uima.tt.CompoundAnnotation
Die Annotation eines zusammengesetzten Tokens. Das zusammengesetzte Token umfasst in der Regel mehrere Annotationen von Token.

uima.tt.MultiTokenAnnotation
Lexikalische Annotation, die aus mehreren Token besteht. Dieser Annotationstyp hat die folgenden Subtypen:

uima.tt.StopwordAnnotation
Annotationen von Stoppwörtern. Bei Stoppwörtern kann es sich auch um Mehrwortbegriffe handeln.

uima.tt.SynonymAnnotation
Die Annotation eines Begriffs, für den Synonyme vorhanden sind. Sie hat die Komponente `synonyms`, die die für den Begriff gefundenen Synonyme auflistet.

uima.tt.SpellCorrectionAnnotation
Die Annotation eines Begriffs, für den Rechtschreibkorrekturen vorhanden sind. Sie hat die Komponente `correctionTerms`, die mögliche Korrekturen in einer Sortierreihenfolge auflistet, die bei den wahrscheinlichsten Korrekturen beginnt.

uima.tt.MultiWordAnnotation
Die Annotation eines Mehrwortbegriffs.

uima.CAS.TOP
Stammelement des Typsystems. Das Element hat die folgenden Subtypen:

uima.tt.KeyStringEntry
Der abstrakte Typ für Datenstrukturen `String`. Er enthält die Komponente `key`, die den Zeichenfolgenschlüssel und den folgenden Subtyp enthält:

uima.tt.Lemma
Lemmaeinträge im Wörterverzeichnis.

uima.tt.CategoryConfidencePair
Der Übereinstimmungswert für die gefundene Kategorie. Er hat die folgenden Komponenten:

- `categoryString` mit dem Namen der Kategorie
- `categoryConfidence` mit dem Übereinstimmungswert für die Kategorie
- `mostSpecific` mit einer Markierung, die angibt, ob die Kategorie auch die spezifischste für das Dokument ist
- `taxonomy` mit dem Namen der Taxonomie, aus der die Kategorie abgeleitet wurde

uima.tt.LanguageConfidencePair
Der Übereinstimmungswert für die gefundene Kategorie. Dieser Typ enthält die Komponenten `languageConfidence`, `language` und `languageID`.

tt_extension_typesystem

Die Beschreibung `tt_extension_typesystem` enthält zusätzliche Textanalysekomponenten für die erweiterte Verarbeitung.

uima.tt.TokenLikeAnnotation
Dieser Annotationstyp in `tt_extension_typesystem` hat die folgenden Komponenten:

- `lemmaEntries` listet alle möglichen Lemmata für das Token auf. Die Listeneinträge haben den Typ `uima.tt.Lemma`.
- `tokenNumber`
- `stopwordToken`

uima.tt.Lemma
Diese Annotation des Typs `uima.tt.KeyStringEntry` hat die folgenden Komponenten:

- `isStopword` ist wahr, wenn das Lemma ein Stoppwort ist.
- `isDeterminer` ist wahr, wenn das Lemma ein Determinator ist.
- `partOfSpeech`. Die folgenden numerischen Beschreibungscodes für die Wortarten sind vorhanden:
  - 0: unbekannt
  - 1: Pronomen
  - 2: Verb
  - 3: Nomen
  - 4: Adjektiv
  - 5: Adverb
  - 6: Apposition
  - 7: Interjektion
  - 8: Konjunktion

uima.tt.DocStructureAnnotation
Annotationen zur Dokumentstruktur. Diese hat die folgenden Subtypen:

uima.tt.SentenceAnnotation
Dokumentsatz. Er hat die Komponente `sentenceNumber`.

uima.tt.ParagraphAnnotation
Dokumentabschnitt. Er hat die Komponente `paragraphNumber`.

dlt_extension_typesystem

Die Beschreibung `dlt_extension_typesystem` enthält zusätzliche Komponenten, die von IBM LanguageWare verwendet werden.

uima.tt.LexicalAnnotation
Diese Annotation hat die folgenden Subtypen:

uima.tt.TokenLikeAnnotation
In `dlt_extension_typesystem` hat diese Annotation die folgenden Komponenten:

- `synonymEntries`

- frost_TokenType
- inflectedForms
- spellAid
- decomposition

com.ibm.dlt.uimatypes.FilePath

com.ibm.dlt.uimatypes.Email

com.ibm.dlt.uimatypes.Number

com.ibm.dlt.uimatypes.URL

com.ibm.dlt.uimatypes.Date

com.ibm.dlt.uimatypes.Time

com.ibm.dlt.uimatypes.Tel

com.ibm.dlt.uimatypes.Currency

com.ibm.dlt.uimatypes.Acronym

uima.tt.TokenLikeAnnotation
Der Annotationstyp in dlt_extension_typesystem hat den folgenden Typ:

com.ibm.dlt.uimatypes.MWU
Dieser Typ wird von IBM LanguageWare verwendet, um Annotationen zu Mehrwortausdrücken zu erstellen.

uima.tt.KeyStringEntry
Zeichenfolgeannotationen. Dieser Typ hat die folgenden Subtypen:

uima.tt.Lemma
Er hat die folgenden Komponenten:
- frost_Constraints mit Markierungen für Einschränkungen
- frost_MorphBitMasks, die einen morphologischen Bitmaskenbereich enthält
- frost_ExtendedPOS mit einem erweiterten Teil für Sprachinformationen, wie z. B. JPOS für Japanisch und CPOS für Chinesisch
- frost_JKom mit morphologischen Daten für Japanisch
- frost_JPStart mit Analysestartdaten für Japanisch
- morphID mit Lemmamerkmalen

uima.tcas.Annotation
Dieser Typ hat den folgenden Subtyp:

com.ibm.dlt.uimatypes.Decomp_Analysis
Vollständige strukturelle Analyse eines zusammengesetzten Begriffs. Sie hat die folgenden Komponenten:
- headComponentIndex mit der Headerkomponente des zusammengesetzten Begriffs
- route mit einer Liste der Token, die aus einer Dekompositionsroute bestehen

Zugehörige Verweise

„Beispiel für eine Typsystembeschreibung" auf Seite 26
Die Typsystembeschreibung enthält die Komponentenstrukturen (die zugrunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Bestimmte Typen und Komponenten für die Unternehmenssuche

Zu den in der Beschreibung 'of_typesystem' definierten Typen und Komponenten gehören bestimmte Typen für OmniFind Enterprise Edition. Diese Typen werden für dokumentspezifische Metadaten verwendet. Sie beschreiben auch die Darstellung von Feldern und XML-Markup-Informationen oder HTML-Ankern.

Die Beschreibung 'of_typesystem' ist nicht im Software-Development-Kit (SDK) von UIMA definiert. Wenn Sie beim Schreiben eines Annotators in UIMA einen dieser Typen verwenden möchten, müssen Sie die Typen in der Typsystembeschreibung Ihrer Analysesteuerkomponente erneut definieren. Möglicherweise möchten Sie z. B. auf Dokumentsicherheitsinformationen oder auf den Crawlertyp oder Dokumenttyp zugreifen.

Die folgenden Typen und Komponenten werden in der Beschreibung 'of_typesystem' definiert:

uima.tcas.DocumentAnnotation
Die UIMA-Standarddokumentanmerkung wird durch die folgende Komponente erweitert:

esDocumentMetaData
Enthält Dokumentmetadaten des Typs `com.ibm.es.tt.DocumentMetaData`.

com.ibm.es.tt.DocumentMetaData
Der Dokumentmetadatentyp hat die folgenden Komponenten. Die Komponenten sind mit der Dokumentannotationskomponente `esDocumentMetaData` verknüpft.

crawlerId
Der Crawlername. Der Komponentenwert hat den Typ `uima.cas.String`.

dataSource
Einer der folgenden Datenquellentypen. Der Komponentenwert hat den Typ `uima.cas.String`.

- CM (für Dokumente, die vom DB2 Content Manager-Crawler durchsucht werden)
- Database (für Dokumente, die vom Crawler der JDBC-Datenbank durchsucht werden)
- DB2 (für Dokumente, die vom DB2-Crawler durchsucht werden)
- DominoDoc (für Dokumente, die vom Domino Document Manager-Crawler durchsucht werden)
- Exchange (für Dokumente, die vom Exchange Server-Crawler durchsucht werden)
- NNTP (für Dokumente, die vom NNTP-Crawler durchsucht werden)
- Notes (für Dokumente, die vom Notes-Crawler durchsucht werden)
- QuickPlace (für Dokumente, die vom QuickPlace-Crawler durchsucht werden)
- Seedlist (für Dokumente, die vom Crawler der Einstiegspunktliste durchsucht werden)

- UnixFS (für Dokumente, die vom Crawler des UNIX-Dateisystems durchsucht werden)
- VBR (für Dokumente, die vom Content Edition-Crawler durchsucht werden)
- WCM (für Dokumente, die vom Web Content Management-Crawler durchsucht werden)
- Web (für Dokumente, die vom Web-Crawler durchsucht werden)
- WinFS (für Dokumente, die vom Crawler des Windows-Dateisystems durchsucht werden)
- WP (für Dokumente, die vom WebSphere Portal-Crawler durchsucht werden)

dataSourceName
Der Name des Crawlers (Datenquelle). Der Komponentenwert hat den Typ `uima.cas.String`.

docType
Einer der folgenden Dokumenttypen. Der Komponentenwert hat den Typ `uima.cas.String`.

- text/html
- application/postscript
- application/pdf
- application/x-mspowerpoint
- application/msword
- application/x-msexcel
- application/rtf
- application/vnd.lotus-wordpro
- application/x-lotus-123
- application/vnd.lotus-freelance
- text/xml
- text/plain
- application/x-js-taro (Ichitaro)

securityTokens
Die Dokumentsicherheitstoken. Der Komponentenwert hat den Typ `uima.cas.StringArray`.

date Das Dokumentdatum. Der Komponentenwert hat den Typ `uima.cas.String`.

baseUri
Die Basis-URI der Seite. Der Komponentenwert hat den Typ `uima.cas.String`.

metaDataFields
Der Komponentenwert hat den Typ `uima.cas.FSArray`. Jedes Element in diesem Array hat den Typ `com.ibm.es.tt.MetaDataField`.

redirectUrl
Die umgeleitete URL. Der Komponentenwert hat den Typ `uima.cas.String`.

mimeType
Der MIME-Typ oder Dokumenttyp, z. B. XML. Der Komponentenwert hat den Typ `uima.cas.String`.

url Die Dokument-URL. Der Komponentenwert hat den Typ `uima.cas.String`.

com.ibm.es.tt.CommonFieldParameters
Zu den allgemeinen Feldparametern gehören folgende Parameter:

searchable
Eine Markierung, die angibt, dass das Feld für die freie Textsuche verfügbar ist.

fieldSearchable
Eine Markierung, die angibt, dass das Feld als Feld durchsucht werden kann.

parametric
Eine Markierung, die angibt, dass das Feld mit einer parametrischen Suche durchsucht werden kann.

showInSearchResult
Eine Markierung, die angibt, dass mit Annotationen versehene Daten in die Suchergebnisdetails aufgenommen werden.

resolveConflict
Eine Markierung zum Auflösen von Metadatenkonflikten zwischen `MetadataPreferred`, `ContentPreferred` und `Coexist`. Der Komponentenwert hat den Typ `uima.cas.String`.

name Der Name des Felds. Sie können unter Verwendung des Feldnamens nach diesem Feld suchen. Der Komponentenwert hat den Typ `uima.cas.String`.

sortable
Eine Markierung, die angibt, dass das Feld nach Zeichenfolgen sortiert werden kann.

exactMatch
Eine Markierung, die angibt, dass die Suche und die Abfragebegriffe exakt übereinstimmen müssen.

com.ibm.es.tt.ContentField
Die Inhaltsfeldannotation hat die folgende Komponente:

parameters
Die Inhaltsfeldparameter haben den Typ `com.ibm.es.tt.CommonFieldParameters`.

com.ibm.es.tt.MetaDataField
Die Daten in Metadatenfeldern sind nicht Teil des Dokumentinhalts, sie werden aber in der Komponente "text" gespeichert:

parameters
Parameter von Metadatenfeldern des Typs `com.ibm.es.tt.CommonFieldParameters`.

text Der Metadatentext wird in dieser Komponente gespeichert. Sie hat den Typ `uima.cas.String`.

com.ibm.es.tt.Anchor
Die Ankerannotation für Ankertext in HTML-Dokumenten. Sie hat die folgende Komponente:

uri Die Ziel-URI des Ankertexts. Der Komponentenwert hat den Typ `uima.cas.String`.

com.ibm.es.tt.MarkupTag
Die Annotationen zu Markupinformationen, z. B. eines XML-Tags. Die Markupinformationen sind in den folgenden Komponenten gespeichert:

name Der Name des Markuptags. Der Komponentenwert hat den Typ `uima.cas.String`.

depth Die Verschachtelungstiefe. Der Komponentenwert hat den Typ `uima.cas.Integer`.

attributeName
Der Name des Komponentenattributs. Der Komponentenwert hat den Typ `uima.cas.StringArray`.

attributeValues
Eine Wertezeichenfolge für das Attribut. Der Komponentenwert hat den Typ `uima.cas.StringArray`.

Beispiel für eine Typsystembeschreibung

Die Typsystembeschreibung enthält die Komponentenstrukturen (die zugrunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Die Typsystembeschreibung muss Teil des Analysesteuerkomponentenarchivs (PEAR-Datei) sein, die aus Ihrer UIMA-Umgebung in die Unternehmenssuche importiert wird.

Das folgende Beispiel einer Typsystembeschreibung enthält Polizeiberichte mit Informationen zu Verdächtigen, Tatorten, Tatzeiten und Datumsangaben:

Die folgende Beispielbeschreibung eines Typsystems wird in allen Textanalysethemen verwendet, in denen die unterschiedlichen Zuordnungstypen erläutert werden, die Sie mit der benutzerdefinierten Analyse auswählen können.

```
<?xml version="1.0" encoding="UTF-8"?>
<typeSystemDescription>
   <name>Police Reports Type System</name>
   <description>Typsystembeschreibung für
        Polizeiberichte</description>
   <version>1.0</version>
   <types>
     <typeDescription>
       <name>com.ibm.omnifind.types.PoliceReport</name>
       <description>Polizeibericht mit Annotationen versehen</description>
       <supertypeName>uima.tcas.Annotation</supertypeName>
       <features>
         <featureDescription>
           <name>time</name>
           <description>Tatzeit gemäß Bericht
              </description>
           <rangeTypeName>com.ibm.omnifind.types.Time</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>date</name>
           <description>Tatzeit</description>
           <rangeTypeName>com.ibm.omnifind.types.Date</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>location</name>
           <description>Tatort der Straftat</description>
           <rangeTypeName>com.ibm.omnifind.types.City</rangeTypeName>
         </featureDescription>
         <featureDescription>
```

```
      <name>knownSuspects</name>
      <description>Enthält Annotationen des Typs Verdächtiger</description>
      <rangeTypeName>uima.cas.FSArray</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>crimeDescription</name>
      <description>Kurzbeschreibung der Straftat</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.City</name>
  <description>Name einer Stadt</description>
  <supertypeName>uima.tcas.Annotation</supertypeName>
  <features>
    <featureDescription>
      <name>cityName</name>
      <description>Name der Stadt</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>cityDistrict</name>
      <description>Name des Stadtteils</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Person</name>
  <description>Annotation zu Person</description>
  <supertypeName>uima.tcas.Annotation</supertypeName>
  <features>
    <featureDescription>
      <name>role</name>
      <description>Rolle, z. B. Verdächtiger oder Zeuge</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>firstName</name>
      <description>Vorname der Person</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>surName</name>
      <description>Familienname der Person</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>title</name>
     <description>Anrede, z. B. Herr oder Frau</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>gender</name>
      <description>Männlich oder weiblich</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Suspect</name>
  <description>Gefundener Verdächtiger</description>
  <supertypeName>com.ibm.omnifind.types.Person</supertypeName>
  <features>
    <featureDescription>
      <name>description</name>
```

```
            <description>Beschreibung des Verdächtigen,
            z. B. Bartträger mit dunkler Brille</description>
            <rangeTypeName>uima.cas.String</rangeTypeName>
          </featureDescription>
        </features>
      </typeDescription>
      <typeDescription>
        <name>com.ibm.omnifind.types.Date</name>
        <description>Ein Datum</description>
        <supertypeName>uima.tcas.Annotation</supertypeName>
        <features>
          <featureDescription>
            <name>year</name>
            <description>Jahresangabe, z. B. 2005</description>
            <rangeTypeName>uima.cas.Integer</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>month</name>
            <description>Monat in Ziffern, z. B. 7</description>
            <rangeTypeName>uima.cas.Integer</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>day</name>
            <description>Tag in Ziffern</description>
            <rangeTypeName>uima.cas.Integer</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>dayOfWeek</name>
            <description>Wochentag, z. B. Montag</description>
            <rangeTypeName>uima.cas.String</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>quarter</name>
            <description>Quartal, z. B. Q1-2005</description>
            <rangeTypeName>uima.cas.String</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>englDate</name>
            <description>Datum im Format mm/tt/jjjj</description>
            <rangeTypeName>uima.cas.String</rangeTypeName>
          </featureDescription>
        </features>
      </typeDescription>
      <typeDescription>
        <name>com.ibm.omnifind.types.Time</name>
        <description>Eine Zeitangabe</description>
        <supertypeName>uima.tcas.Annotation</supertypeName>
        <features>
          <featureDescription>
            <name>hours</name>
            <description>Angabe der Stunde von 00 bis 23</description>
            <rangeTypeName>uima.cas.Integer</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>minutes</name>
            <description>Angabe der Minuten innerhalb der Stunde</description>
            <rangeTypeName>uima.cas.Integer</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>timeOfDay</name>
            <description>Tageszeit, z. B. Morgen, Mittag</description>
            <rangeTypeName>uima.cas.String</rangeTypeName>
          </featureDescription>
        </features>
      </typeDescription>
    </types>
  </typeSystemDescription>
```

XML-Markup-Formatierung in Analyse und Suche

Sie können Informationen in XML-Strukturen in einem Dokument direkt einer allgemeinen Analysestruktur zuordnen, ohne einen UIMA-Annotator zu schreiben.

Wenn die Dokumente in Ihrer Objektgruppe in XML vorliegen und Sie Markup-Formatierung in der Textanalyse oder semantischen Suche nutzen wollen, stehen Ihnen die folgenden Optionen zur Verfügung:

Native XML-Suche

Verwenden Sie diese Option, wenn Sie alle XML-Tags und -Attribute während der Suche so verwenden wollen, wie sie im Dokument angezeigt werden. Wenn Sie z. B. Dokumente für die Rechnungsstellung haben, die das Element `<addressee>` enthalten, können Sie durch Aktivieren der nativen XML-Suche diesen Tag in einer semantischen Suchabfrage verwenden, um in diesem Element nach einem bestimmten Kundennamen zu suchen.

Mit dieser Option wird die XML-Struktur des Dokuments in der allgemeinen Analysestruktur unter Verwendung des Typs `com.ibm.es.tt.MarkupTag` dargestellt. Für jeden XML-Tag wird eine Annotation dieses Typs erstellt. Die Annotation enthält den Namen des Tags, seine Attribute und den Attributinhalt. Diese Informationen werden immer indexiert und sind für die semantische Suche verfügbar.

Für die native XML-Suche ist keine Zuordnungskonfigurationsdatei erforderlich. Sie können die native XML-Suche über die Administrationskonsole für die Unternehmenssuche aktivieren.

Zuordnung von XML-Elementen zur allgemeinen Analysestruktur

Verwenden Sie diese Option in den folgenden Fällen:

- Die Semantik bestimmter XML-Elemente ist präzise und kann in späteren Textanalyseschritten verwendet werden. Die Analyseschritte können direkt für die von den XML-Strukturen erstellten Annotationen und Komponenten ausgeführt werden und sind gegen die möglicherweise abweichenden Formate der Originaldokumente abgeschirmt. Z. B. enthält das Element `<addressee>` in Dokumenten für die Rechnungsstellung in der Regel Kundennamen. Wenn Sie die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur verwenden, können Sie den Inhalt dieses Elements direkt Annotationen des Typs `Customer` zuordnen. Ein Annotator kann dann eine Beziehung zum Standort des Kunden ableiten, indem er die Informationen in der Umgebung der Annotation `Customer` verwendet.
- Sie wollen den Verarbeitungsbereich eines benutzerdefinierten Annotators auf angegebene Bereiche in der XML-Eingabe eingrenzen. Sie wollen z. B. die Analyse des Taginhalts von `<technicianComment>` nur in einem Annotator eingrenzen, der KFZ-Probleme feststellt.
- Sie wollen die Verarbeitung der Textanalyse und die nachfolgende Suche auf bestimmte Teile des XML-Dokuments beschränken und irrelevante Inhalte oder Inhalte herausfiltern, bei denen es sich nicht um Text handelt.
- Sie wollen XML-Tags, die untXML-Dokuments beschränken und irrelevante Inhalte oder Inhalte herausfiltern, bei denen es sich nicht um Text handelt.
- Sie wollen XML-Tags, dieerschiedliche Namen haben einem einheitlichen Bereich zuordnen, der bei der semantischen Suche verwendet wird. Sie können z. B. `<mainHeading>` oder `<doc>` dem Titel zuordnen.

In diesen Fällen, müssen Sie eine Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur erstellen, die definiert, welche XML-Elemente welchen Komponentenstrukturen zugeordnet werden. Die Komponentenstrukturen, die Sie in der Zuordnungsdatei definieren, werden während der syntaktischen Analyse der Dokumente erstellt. Der Zugriff erfolgt durch die benutzerdefinierten Annotatoren.

Sie können mehrere Dateien für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur für eine Dokumentobjektgruppe verwenden. Welche Zuordnungsdatei für welches XML-Dokument verwendet wird, wird vom Element `<identifier>` festgelegt. Das Element `<identifier>` in der Zuordnungsdatei muss mit dem Stammelement im XML-Dokument übereinstimmen. Wenn das Stammelement Ihres Dokuments z. B. `doc` ist, muss in der Zuordnungsdatei für das Element `<identifier>` ebenfalls der Wert "doc" angegeben sein.

Wird keine Übereinstimmung gefunden, sucht das Programm nach einer Zuordnungsdatei, in der das Element `<identifier>` auf den Standardwert gesetzt ist. Wird keine Standardzuordnung gefunden, werden die Textabschnitte des Dokuments (ohne Taginformationen) dem Dokumentkommentar in der allgemeinen Analysestruktur zugeordnet.

Wenn Sie Informationen extrahieren wollen, die nur in relevanten Teilen eines Dokuments enthalten sind, während die irrelevanten Teile ignoriert werden sollen, müssen Sie nur angeben, welche XML-Elemente des Dokuments relevante Informationen enthalten. Diese Vorgehensweise wird als Inhaltsextraktion bezeichnet. Sie können z. B. die in den Titel- und Hauptteilelementen angegebene Eingabe extrahieren, während Sie die Eingabe in den Elementen Autor, Datum, ID und Veröffentlichungskomponente ignorieren.

Die Inhaltsextraktion kann die Analyseverarbeitung für die folgenden XML-Dokumenttypen verbessern:

- Dokumente, deren Inhalt umfangreiche Teile enthält, die nicht analysiert werden sollen, z. B. binäre Anlagen. Die Verwendung der Inhaltsextraktion verringert die Dokumentgröße beträchtlich, erhöht dadurch die Verarbeitungsgeschwindigkeit und vermeidet gleichzeitig Analysefehler, die aufgrund ungeeigneter Daten entstehen.
- Dokumente, deren Text mit irrelevanten Textstellen durchsetzt ist, z. B. Dokumente, mit redaktionellen Informationen zwischen den Tags `<note>`. Durch das Ignorieren dieser Informationen erzielen Sie bessere Ergebnisse bei der Analyse des Dokumentinhalts.

Die Verwendung der nativen XML-Suche und die Optionen der Inhaltsextraktion in der Zuordnung von XML-Elementen zur allgemeinen Analysestruktur schließen einander aus, da nur der gesamte Inhalt oder nur ein angegebener Inhalt berücksichtigt werden kann. Wenn Sie die Inhaltsextraktion angeben, wird die native XML-Zuordnung ignoriert. Ohne Inhaltsextraktion können Sie die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur und die native XML-Suche verwenden.

Alle Typen und Komponenten, die Sie in Ihrer Konfigurationsdatei verwenden, müssen in der Typsystembeschreibung Ihrer benutzerdefinierten Analyseschritte beschrieben sein. Sie können einen Deskriptor für ein Typsystem in Ihrer UIMA-Umgebung erstellen, indem Sie das Eclipse-Plug-in "Component Descriptor Editor" verwenden. Dieses Plug-in ermöglicht es Ihnen, auch ohne Kenntnisse der erforderlichen XML-Syntax, eine Deskriptordatei zu erstellen.

Nachdem Sie die benutzerdefinierte Analyse fertig erstellt und getestet haben, verwenden Sie den Assistenten für die Generierung für PEAR-Dateien von UIMA, um ein Archiv zu erstellen, das die benutzerdefinierten Analysedateien einschließlich der Typsystembeschreibung enthält. Anschließend können Sie das benutzerdefinierte Analysearchiv und Ihre Dateien für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur in die Unternehmenssuche hochladen. Verwenden Sie hierfür die Administrationskonsole für die Unternehmenssuche.

Zugehörige Tasks

„Erstellen einer Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur"
In einer Datei für die Zuordnung von XML zur allgemeinen Analysestruktur können Sie die gesamte Palette an Konfigurationsoptionen für die Zuordnung von XML- zu UIMA-Datentypen verwenden.

Erstellen einer Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur

In einer Datei für die Zuordnung von XML zur allgemeinen Analysestruktur können Sie die gesamte Palette an Konfigurationsoptionen für die Zuordnung von XML- zu UIMA-Datentypen verwenden.

Informationen zu dieser Task

Im folgenden Beispiel ist eine Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur dargestellt.

Der Beispielbericht, ein Polizeibericht, enthält XML-Tags für die Art des Verbrechens, das Tatdatum, den Tatort, den diensthabenden Polizeibeamten, sein Polizeirevier, die Täterbeschreibung und eine Zusammenfassung. Danach folgt ein Abschnitt mit dem Hauptteil. Beispiel:

```
<report>
 <doc>
  <crimeType>Autodiebstahl</crimeType>
  <crimeDate>04/23/05 21:23</crimeDate>
  <crimeLocation>Hauptstraße 27, Stuttgart</crimeLocation>
  <reportingOfficer rank="Lt">Jakob
   <lastName>Meier</lastName>
  </reportingOfficer>
  <policePrecinct>14. Polizeirevier</policePrecinct>
  <suspectDescription>Männlich, dunkelhaarig, dunkle Brille
    Bluejeans und dunkles, möglicherweise schwarzes
    Jacket</suspectDescription>
  <abstract>Ein Mercedes CLK wurde am 04/23/2005 von einem Parkplatz
   vor dem Restaurant "Blaue Lagune", Hauptstraße 27 in Stuttgart
   gestohlen. (Seriennummer: 32 2761 50871)</abstract>
  <body>Ein Mercedes CLK wurde am 04/23/2004 von einem Parkplatz
   vor dem Restaurant "Blaue Lagune", Hauptstraße 27 in Stuttgart
   gestohlen. (Seriennummer: 32 2761 50871)

Er ist schwarz und hat Breitreifen der Marke Michelin.
Augenzeugen vor dem Restaurant sahen zwei dunkel gekleidete Männer mit hoher
Geschwindigkeit mit dem Auto wegfahren. Das Fahrzeug wurde verlassen in der
Ulmenallee in Bruchsal gefunden. Der Tank war leer. Die Sitze waren stark
verschmutzt und der Rücksitz wurde beschädigt. Aus dem Fahrzeug wurde
nichts gestohlen....</body>
 </doc>
 <image>
  <--! image of the crime scene as a base64-encoded string -->
 </image>
</report>
```

Auf der Basis dieses Beispielberichts, könnte eine Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur die folgende Struktur aufweisen. Das Beispiel verwendet das Typsystem, das für das Szenario des Polizeiberichts definiert wurde.

```
<?xml version="1.0"?>
<xmlCasInitializerConfiguration
 xmlns="http://www.ibm.com/2005/uima/jedii_ci_xml">

 <identifier>Default</identifier>
 <description>Beispielkonfiguration</description>

 <contentElements>
   <element>/report/doc</element>
 </contentElements>

 <elementToTypeMappings>
   <elementToTypeMapping>
     <element>//doc//reportingOfficer</element>
     <type>com.ibm.omnifind.types.Person</type>
     <featureValueAssignment>
       <feature>role</feature>
       <basicValue default="Reporting officer">
           </basicValue>
     </featureValueAssignment>
     <featureValueAssignment>
       <feature>gender</feature>
       <basicValue default="male"
          useAttributeValue="sex"/>
     </featureValueAssignment>
     <featureValueAssignment>
       <feature>surName</feature>
       <values concatenate="true" delimiter=" ">
         <basicValue useAttributeValue="rank"
                 default="Lt"/>
         <basicValue useElementContent="lastName"/>
       </values>
     </featureValueAssignment>
   </elementToTypeMapping>
   <elementToTypeMapping>
     <element>//doc</element>
     <type>com.ibm.omnifind.types.PoliceReport</type>
     <featureValueAssignment>
       <feature>crimeDescription</feature>
       <basicValue useElementContent="abstract"
                trim="true">
           </basicValue>
     </featureValueAssignment>
   </elementToTypeMapping>
 </elementToTypeMappings>

</xmlCasInitializerConfiguration>
```

Einschränkungen

Die Zuordnungsdatei besteht aus zwei Abschnitten:

Element <contentElements>

Verwenden Sie dieses Element, wenn Sie bestimmte Inhalte extrahieren wollen. Die Beispielzuordnungsdatei extrahiert den Inhalt im Abschnitt <doc> eines Dokuments und ignoriert die anderen Abschnitte des Dokuments. Im XML-Polizeibericht kann eine große, für die Textverarbeitung unbrauchbare Grafik enthalten sein. Indem Sie <doc> als Inhaltselement angeben und nicht <image>, wird die Grafik herausgefiltert bevor die Textverarbeitung beginnt.

<elementToTypeMappings>

Verwenden Sie dieses Element, um anzugeben, welche einzelnen XML-Elemente (in einem Element <elementToTypeMapping> angegeben) des Dokuments welchen Komponentenstrukturen der allgemeinen Analysestruktur zugeordnet werden sollen.

Wenn Sie die Option für die Inhaltsextraktion verwenden, müssen die XML-Elemente, die im Abschnitt <elementToTypeMappings> angegeben sind, in den XML-Elementen enthalten sein, die im Abschnitt <contentElements> angegeben sind.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool für die XML-Prüfung, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die Zuordnungsdatei heißt XMLCasInitSchema.xsd und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*/packages/uima/configuration_xsd/.
2. Nehmen Sie Ihre Zuordnungen in ein Element <xmlCasInitializerConfiguration xmlns="http://www.ibm.com/2005/uima/jedii_ci_xml"> auf. Der Namespace (im Attribut xmlns angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element <contentElements> hinzu, wenn Sie bestimmte Inhalte aus Dokumentabschnitten extrahieren wollen, und ein Element <elementToTypeMappings>, das angibt, welche einzelnen XML-Elemente des Dokuments Sie welchen Komponentenstrukturen des allgemeinen Analysebereichs zuordnen.
4. Fügen Sie ein Element <identifier> und ein Element <description> hinzu. Die Kennung legt fest, welche Zuordnung für welches XML-Dokument verwendet wird. Die Kennung muss das Stammelement des Dokuments enthalten, zum Beispiel doc. Wenn die Kennung auf den Standardwert gesetzt ist, ist das Stammelement des Dokuments irrelevant und die Zuordnung wird auf jedes XML-Dokument angewendet.
5. Wenn Sie Informationen extrahieren wollen, die nur in relevanten Teilen eines Dokuments enthalten sind, fügen Sie ein Element <contentElements> hinzu. Es enthält das folgende Komponentenelement:
   - Mindestens ein Element <element>, das den Pfad zu einem XML-Element des Dokuments enthält und die XPath-Syntax einhält, z. B. <element>/doc/crimeType</element>.
6. Wenn Sie angeben wollen, welche XML-Elemente des Dokuments welchen Komponentenstrukturen der allgemeinen Analysestruktur zugeordnet werden, fügen Sie ein Element <elementToTypeMappings> hinzu. Es enthält die folgenden Komponentenelemente:
   - Mindestens ein Element <elementToTypeMapping>. Dieses Element muss die folgenden verschachtelten Elemente aufweisen:
     – Ein Element <element>, das verwendet wird, um den Pfad eines XML-Elements anzugeben, und die XPath-Syntax befolgt: Ein vorangestellter Schrägstrich (/) bedeutet, dass ein vollständiger Pfad angegeben ist. Zum Beispiel abstract unter dem Stammelement doc. Zwei Schrägstriche (//) stehen für eine beliebige Untergruppe unter einem Pfad. So muss z. B. birthDate innerhalb von reportingOfficer enthalten sein, es können jedoch andere Elemente zwischen den beiden Elementen vorhanden sein.

- Ein Element <type>, das einen Typ angibt, der in der Typsystembeschreibung definiert ist. Es muss den Typ Annotation aufweisen.
- Null oder mehr Elemente <featureValueAssignment>.

7. In einem Element <featureValueAssignment> benennen Sie eine Komponente des Typs String im Element <feature>, und weisen Sie im Element <basicValue> einen Wert zu. Mehrere Elemente <basicValue> können einem Element <values> hinzugefügt werden.

   Das Element <basicValue> kann Attribute haben. Zu diesen gehören useAttributeValue, useElementContent, default und trim.

Verwenden Sie useAttributeValue, wenn Sie den Wert eines Attributs als Wert einer Komponente verwenden wollen. Beispiel:

```
  <elementToTypeMapping>
    <element>/doc//reportingOfficer</element>
<type>com.ibm.omnifind.types.Person</type>
    <featureValueAssignment>
      <feature>role</feature>
      <basicValue default="Reporting officer"/>
    </featureValueAssignment>
    <featureValueAssignment>
      <feature>gender</feature>
      <basicValue default="male" useAttributeValue="sex"/>
    </featureValueAssignment>
  </elementToTypeMapping>
```

Dieses Beispiel führt zum folgenden Ausgabeergebnis:

- Für jeden XML-Tag <reportingOfficer>, der an beliebiger Stelle innerhalb eines XML-Tags <doc> im Dokument vorhanden ist, wird eine Komponentenstruktur des Typs com.ibm.omnifind.types.Person erstellt.
- Wenn der Tag <reportingOfficer> das Attribut sex enthält, wird die Komponente gender der neu erstellten Komponentenstruktur auf diesen Attributwert gesetzt.

Verwenden Sie das Attribut useElementContent, um Inhalt als Wert einer Komponente hinzuzufügen. Das folgende Beispiel enthält einen Zuordnungsausschnitt:

```
<elementToTypeMapping>
  <element>//doc</element>
  <type>com.ibm.omnifind.types.PoliceReport</type>
  <featureValueAssignment>
    <feature>crimeDescription</feature>
    <basicValue useElementContent="abstract" trim="true"/>
  </featureValueAssignment>
</elementToTypeMapping>
```

In diesem Ausschnitt wird der Text, auf den sich das Element <abstract> in <doc> bezieht, zum Wert der Komponentenstruktur crimeDescription. Alle vorangehenden und folgenden Leerzeichen werden entfernt.

In den folgenden Fällen kann mehr als ein Wert im Element <values> angegeben werden:

- Die Komponente, die konfiguriert wird, hat den Typ StringArray.
- Durch die Verwendung des Begrenzerattributs sind viele Zeichenfolgen zu einer Zeichenfolge verknüpft und werden deshalb einer Komponente des Typs String zugeordnet. Beispiel: Der Titel Mr. ist eine Konstante, der Vorname ist ein Attributwert und der Nachname wird von einem XML-Element angegeben:

```
  <elementToTypeMapping>
    <element>//doc//reportingOfficer</element>
<type>com.ibm.omnifind.types.Person</type>
    <featureValueAssignment>
      <feature>surName</feature>
      <values concatenate="true" delimiter=" ">
        <basicValue default="Mr."/>
        <basicValue useAttributeValue="rank"
           default="Lt."/>
        <basicValue useElementContent="lastName"/>
      </values>
    </featureValueAssignment>
  </elementToTypeMapping>
```

Komponentenwerte, die Zeichenfolgen enthalten werden aus der Zuordnungsdatei so extrahiert, wie sie vorliegen. Die Werte behalten etwaige vorangehende oder folgende Leerzeichen bei. Aus Namen von Typen und Komponenten werden Leerzeichen jedoch gelöscht. So wird z. B. `<type> **com.ibm.omnifind.types.Person**</type>` zu `<type>**com.ibm.omnifind.types.Person**</type>`.

Verwenden Sie das Element `<condition>`, um Bedingungen für Attribute festzulegen. Die Komponentenstruktur `com.ibm.omnifind.types.Person` wird z. B. nur erstellt, wenn im Dokument `<suspectDescription>` mit dem auf `yes` gesetzten Attribut `armed` vorhanden ist:

```
  <elementToTypeMapping>
<element>//suspectDescription</element>
<type>com.ibm.omnifind.types.Person</type>
    <condition attribute="armed"  value="yes"/>
  </elementToTypeMapping>
```

Auf der Basis des Beispielpolizeiberichts und der definierten Zuordnungsdatei werden die folgenden Komponentenstrukturen erstellt:

com.ibm.omnifind.types.PoliceReport

- covered text: ″Autodiebstahl 04/23/05 21:23 Hauptstraße 27, Stuttgart, Jakob Meier 14. Polizeirevier Männlich, dunkelhaarig, dunkle Brille, Bluejeans und dunkles, möglicherweise schwarzes Jackett Ein Mercedes CLK wurde ... Aus dem Fahrzeug wurde nichts gestohlen.
- begin = 2
- end = 904
- knownSuspects = null
- crimeDescription = ″Ein Mercedes CLK wurde am 04/23/2005 von einem Parkplatz vor dem Restaurant ″Blaue Lagune″, Hauptstraße 27 in Stuttgart gestohlen. (Seriennummer: 32 2761 50871)″

com.ibm.omnifind.types.Person

- covered text = ″Jakob Meier″
- begin = 112
- end = 127
- role = ″Reporting officer″
- firstName = null
- surName = ″Lt Meier″
- gender = ″male″

Nachdem Sie die Zuordnungsdatei erstellt haben, müssen Sie diese in die Unternehmenssuche hochladen und die Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur mit Ihren anderen benutzerdefinierten Analyseauswahlen in der Administrationsskonsole für die Unternehmenssuche auswählen.

Zugehörige Konzepte

„XML-Markup-Formatierung in Analyse und Suche" auf Seite 29
Sie können Informationen in XML-Strukturen in einem Dokument direkt einer allgemeinen Analysestruktur zuordnen, ohne einen UIMA-Annotator zu schreiben.

Zugehörige Verweise

„Beispiel für eine Typsystembeschreibung" auf Seite 26
Die Typsystembeschreibung enthält die Komponentenstrukturen (die zugrunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Ergebnisse der Textanalyse

Alle Ergebnisse der Textanalyse werden in der allgemeinen Analysestruktur gespeichert.

Annotatoren lesen in der Regel die allgemeine Analysestruktur und schreiben auch in diese. Die privaten Anwender der allgemeinen Analysestruktur (*CAS-Anwender*) haben nur Lesezugriff auf die allgemeine Analysestruktur. Die CAS-Anwender führen die Endverarbeitung der Analyseergebnisse aus, die in der allgemeinen Analysestruktur gespeichert sind. In der Unternehmenssuche gibt es zwei Arten privater CAS-Anwender:

- Der private Anwender, der den Inhalt der allgemeinen Analysestruktur in einer Suchmaschine indiziert. Für diesen privaten Anwender ist eine Datei für die Zuordnung der allgemeinen Analysestruktur zum Index erforderlich, die Sie mit der benutzerdefinierten Textanalyse über die Administrationskonsole für die Unternehmenssuche auswählen.
- Der private Anwender, der eine relationale Datenbank mit bestimmten Analyseergebnissen füllt. Für diesen privaten Anwender ist eine Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank erforderlich, die Sie mit den benutzerdefinierten Textanalyseoptionen über die Administrationskonsole für die Unternehmenssuche auswählen.

Erforderlichenfalls können Sie benutzerdefinierte private CAS-Anwender in die Unternehmenssuche implementieren. Informationen dazu, wie Sie einen privaten Anwender schreiben, finden Sie in der Dokumentation von UIMA. Informationen dazu, wie Sie Ihren privaten Anwender in die Unternehmenssuche hochladen und verwenden, finden Sie auf der Website von IBM UIMA developerWorks unter http://www.ibm.com/developerworks/db2/zones/db2ii/.

Zugehörige Konzepte

„Indexzuordnung für benutzerdefinierte Analyseergebnisse" auf Seite 41
Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe ausgeführt haben, können Sie die Suchmaschine in der Unternehmenssuche verwenden, um einen Index zu erstellen. Verwenden Sie für den Index die Informationen, die in der allgemeinen Analysestruktur gespeichert sind, die von den benutzerdefinierten Analysealgorithmen erstellt wurde.

„Datenbankzuordnung für ausgewählte Analyseergebnisse" auf Seite 49
Nachdem Sie Ihre Dokumente in der Unternehmenssuche analysiert haben, können Sie ausgewählte Textanalyseergebnisse in einer JDBC-fähigen Datenbank speichern.

Komponentenpfade

Ein Komponentenpfad stellt eine Möglichkeit bereit, auf Komponentenwerte in allgemeinen Analysestrukturen zuzugreifen, ähnlich wie XPath-Anweisungen, die verwendet werden, um auf XML-Elemente in einem XML-Dokument zuzugreifen.

Komponentenpfade sind sinnvoll, wenn Sie auf eine Komponentenstruktur zugreifen wollen, die komplexe Komponenten kombiniert, z. B. Komponenten, die einen Bereichswert haben oder die auf eine andere Komponentenstruktur zeigen. Wenn Sie einen Komponentenpfad verwenden, können Sie den Wert einer Komponente einer Komponentenstruktur direkt zuweisen und diesen Wert im semantischen Suchindex oder in einer Datenbank speichern.

Betrachten Sie z. B. einen Annotator, der Autos und ihre Fabrikate angibt. Er erstellt Annotationen des Typs `car`, die das Attribut `make` aufweisen. In `make` ist jedoch nicht der Hersteller selbst enthalten (z. B. Chevrolet), sondern eine Komponentenstruktur des Typs `Company`, die wiederum das Attribut `companyname` enthält, dessen Wert eine Zeichenfolge sein muss. Wenn Sie eine semantische Abfrage aktivieren wollen, die Autonamen und Firmennamen kombiniert, verwenden Sie den Komponentenpfad `make/companyname`, um den Wert `companyname` dem Bereich `car` zuzuordnen, der für die Annotation `car` generiert wurde. Die Abfrage "Ich suche Dokumente, in denen Autos des Herstellers Chevrolet, vorkommen" wird aktiviert, indem Sie '/car[@make="Chevrolet"]' verwenden.

Ein Komponentenpfad ist eine Folge von Komponentennamen (f1/.../fn) mit den folgenden Merkmalen:

- Der Wert eines Komponentenpfads kann eine Zeichenfolge (`String`), eine ganze Zahl (`Integer`), eine Gleitkommazahl (`Float`) oder ein Bereich eines dieser Typen sein.
- Alle Komponenten dieses Pfads von f1 bis fn-1 müssen einen komplexen Typ aufweisen, das heißt die Typen `uima.cas.TOP`, `uima.cas.FSArray`, `uima.cas.FSList` oder einen ihrer Subtypen.
- Die letzte Komponente des Pfads, fn, kann einen komplexen Typ enthalten. Darüber hinaus kann sie einen der folgenden Typen oder einen seiner Subtypen enthalten: `uima.cas.Float`, `uima.cas.Integer`, `uima.cas.String`, `uima.cas.FloatArray`, `uima.cas.IntegerArray`, `uima.cas.StringArray`, `uima.cas.FloatList`, `uima.cas.IntegerList` oder `uima.cas.StringList`.
- Optional kann eine Komponente eingegeben werden. Der vollständig qualifizierte Name des Typs muss dem Komponentennamen vorangestellt und durch einen Doppelpunkt getrennt werden. Beispiel: `f1/com.ibm.es.SomeType:f2/.../fn` .

Sie können den Typbereich einer bestimmten Komponente eingrenzen. Nehmen Sie z. B. die Komponente `additionalInfo` des Typs `uima.cas.TOP`. Wenn Sie wissen, dass der Wert Ihrer Komponente `additionalInfo` tatsächlich den Typ `EmployeeInfo` hat, der die Komponente `salary` enthält, können Sie unter Verwendung von `additionalInfo/EmployeeInfo:salary` auf diese Komponente zugreifen. Beachten Sie, dass in diesem Beispiel der Komponentenpfad `additionalInfo/salary` zu einem Fehler führen würde, da `salary` nicht für den Typ `uima.cas.TOP` definiert wurde.

Komponenten mit Bereichs- oder Listenwerten haben die folgenden zusätzlichen Merkmale:

- Verwenden Sie eckige Klammern ([<*zahl*>]), um ein bestimmtes Element des Bereichs oder der Liste auszuwählen. Ein Bereich startet bei Null (0). Wenn Sie z. B. das erste Element im Bereich `companies` auswählen wollen, verwenden Sie

companies[0]. Die Sondermarkierung [last] kann verwendet werden, um unabhängig von der Größe den letzten Eintrag eines Bereichs auszuwählen, z. B. companies[last].

- Verwenden Sie leere eckige Klammern ([]), um alle Elemente anzugeben. In einem Komponentenpfad sind leere eckige Klammern ([]) nur einmal zulässig. Wenn Sie z. B. einen Bereich mit Verdächtigen haben, erfasst der Komponentenpfad knownSuspects[]/com.ibm.omnifind.types.Suspect:surName alle Nachnamen von Verdächtigen in einen Bereich String.
- Wird ein Komponentenpfad, der einen Bereich zurückgibt, während der Indexierung verwendet, werden die Bereichselemente verknüpft (durch Leerzeichen getrennt) und als ein aus mehreren Begriffen bestehendes Attribut oder Feld in den Index geschrieben.
- Das nächste Element des Komponentenpfads muss eingegeben werden. Der Name des Typs ist der Typ der Elemente des Bereichs. Nehmen Sie z. B. eine Komponentenstruktur des Typs Info. Dieser Typ hat eine Komponente namens companies, deren Geltungsbereich FSArray ist. Die Elemente des Bereichs haben den Typ Company. Company, wiederum hat eine Komponente namens profit. Wenn Sie nun den Gewinn des dritten Unternehmens ermitteln wollen, geben Sie folgendes ein (verwenden Sie dabei die vollständig qualifizierten Namen der Typen): companies[2]/Company:profit.

Integrierte Komponenten

Integrierte Komponenten sind vordefinierte Komponentennamen mit einer speziellen Semantik. Sie können verwendet werden, um auf Informationen zuzugreifen, die nicht in der Komponentenstruktur selbst enthalten sind, z. B. den Typ der Komponentenstruktur oder den Annotationstext, auf den sie sich beziehen. Sie können in einem Komponentenpfad als letztes oder einziges Element verwendet werden.

Die folgenden integrierten Komponenten können in beiden Zuordnungskonfigurationsdateien verwendet werden:

- fsId() gibt die ID der Komponentenstruktur zurück. Die zurückgegebene ID ist eine ganze Zahl (32 Bit). Verwenden Sie diese integrierte Komponente, um auf Teile eines Dokuments zuzugreifen, die genau mit der Abfrage übereinstimmen.
- typeName() gibt den Objekttyp der allgemeinen Analysestruktur als Zeichenfolge zurück. Der Typ ist der vollständig qualifizierte Name des Typs, einschließlich aller Namensbereichspräfixe, z. B. uima.tcas.Annotation. In einem Datenbankkontext ist typeName() besonders nützlich, wenn Sie Typen und Subtypen in derselben Spalte speichern und wissen wollen, welches der richtige Typ einer Annotation oder einer Komponentenstruktur ist. Im folgenden Beispiel wird der Typ person, wie *suspect* (Verdächtiger) oder *witness* (Zeuge), in der Spalte role gespeichert.

```
<explicitMappingRule applyToSubTypes="false">
   <type>com.ibm.omnifind.types.Person</type>
   <table>sample.person</table>
   <featureMappings>
     <featureMapping>
       <feature>typeName()</feature>
       <column>role</column>
     </featureMapping>
   </featureMappings>
 </explicitMappingRule>
```

- coveredText() gibt den Text zurück, den das allgemeine Analyseobjekt umfasst. coveredText() ist nur für Annotationen und ihre Subtypen verfügbar. Verwenden Sie diese integrierte Komponente nicht für Komponentenstrukturen, die

nicht unter den Typ annotation fallen. Im folgenden Beispiel wird der Name eines Verdächtigen in der Spalte suspectName gespeichert.

```
<implicitMappingRule applyToSubTypes="false">
   <type>com.ibm.omnifind.types.Suspect</type>
   <relation>sample.person</relation>
   <featureMappings>
     <featureMapping>
       <feature>coveredText()</feature>
       <column>suspectName</column>
       <length>128</length>
     </featureMapping>
   </featureMappings>
 </implicitMappingRule>
```

- [] gibt eine Kennung an den aktuellen Containereintrag (Bereich oder Liste) zurück. Die Komponente impliziert eine Iteration, das heißt, dass ein Eintrag in der Datenbanktabelle oder dem Index für jedes Element des Bereichs oder der Liste angelegt wird. Das folgende Beispiel stammt aus einer Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank, in der die integrierte Funktion [:index] auch zulässig ist.

```
<implicitMappingRule applyToSubTypes="false">
   <type>uima.cas.FSArray</type>
   <table>sample.knownSuspects</table>
   <featureMappings>
     <featureMapping>
       <feature>uniqueId()</feature>
       <column>arrayId</column>
     </featureMapping>
     <featureMapping>
       <feature>[:index]</feature>
       <column>arrayIndex</column>
     </featureMapping>
     <featureMapping>
       <feature>[]/com.ibm.omnifind.types.Suspect:uniqueId()</feature>
       <column>suspectId</column>
     </featureMapping>
   </featureMappings>
 </implicitMappingRule>
```

Die folgenden integrierten Komponenten können nur in der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank verwendet werden:

- uniqueId() gibt die globale eindeutige ID der Komponentenstruktur zurück. Bei der zurückgegebenen eindeutigen ID handelt es sich um eine Zeichenfolge mit fester Länge (27 Zeichen) und um eine Verkettung der Ergebnisse von fsId(), docId(), docTimestamp() und der Anzahl aktueller Chunks, da Dokumente in der Unternehmenssuche in mehrere Chunks mit allgemeinen Analysestrukturen geteilt werden können.

  Die zurückgegebene Zeichenfolge kann alle Buchstaben von ″a-z″ und ″A-Z″, die Zahlen von ″0-9″, das Semikolon (″;″) und den Doppelpunkt (″:″) enthalten.

  Das Ergebnis von uniqueId() kann als Primärschlüssel für Tabellen verwendet werden.

- objectId() gibt die ID der Annotation oder Komponentenstruktur zurück. objectId() ist ähnlich wie uniqueId(), enthält jedoch nicht das Ergebnis von docTimestamp(). Die zurückgegebene ID ist nur in einer Objektgruppe eindeutig, in der die Dokument einmal syntaktisch analysiert werden. Wenn für Sie Eindeutigkeit über alle Dokumente und Dokumentversionen hinweg erforderlich ist, müssen Sie uniqueId() verwenden.

  Die zurückgegebene Zeichenfolge der integrierten Komponente objectId() hat eine feste Länge von 16 Zeichen und kann alle Buchstaben von ″a-z″ und ″A-Z″, die Zahlen von ″0-9″, das Semikolon (″;″) und den Doppelpunkt (″:″) enthalten.

Wenn uniqueId() oder objectId() auf leere Komponentenstrukturen verweisen, wird der in der Datenbanktabellendefinition definierte Standardwert verwendet. Es werden keine leeren Objekte eines Typs gespeichert, auf den verwiesen wird.

- docId() gibt die Dokument-ID zurück. Der zurückgegebene Wert ist der ganzzahlige Typ integer (32 Bit).

  Im folgenden Beispiel werden die integrierten Komponenten angezeigt:

```
<explicitMappingRule applyToSubTypes="true">
   <type>com.ibm.omnifind.types.PoliceReport</type>
   <table>sample.PoliceReport</table>
   <featureMappings>
     <featureMapping>
       <feature>uniqueId()</feature>
       <column>policeReportId</column>
     </featureMapping>
     <featureMapping>
       <feature>docId()</feature>
       <column>policeReportDocId</column>
     </featureMapping>
   </featureMappings>
 </explicitMappingRule>
```

- docUri() gibt die Dokument-URI zurück.
- docTimestamp() gibt die Zeit (in Millisekunden) zurück, zu der das Dokument verarbeitet wurde. Diese integrierte Komponente ist sinnvoll für die Verfolgung von Dokumentversionen, z. B. wenn Sie wissen wollen, ob es sich bei der von Ihnen verwendeten Dokumentversion um die aktuellste handelt, die vom Crawler übergeben wurde.

```
<explicitMappingRule applyToSubTypes="false">
   <type>com.ibm.omnifind.types.PoliceReport</type>
   <relation>sample.PoliceReport</relation>
   <featureMappings>
     <featureMapping>
       <feature>uniqueId()</feature>
       <column>policeReportId</column>
     </featureMapping>
     <featureMapping>
       <feature>docTimestamp()</feature>
       <column>reportVersion</column>
     </featureMapping>
   </featureMappings>
 </explicitMappingRule>
```

- parentId() gibt die fsId() der Komponentenstruktur zurück, die eine Containerzuordnung enthält. parentId() ist nur im Kontext einer Containerzuordnung gültig.
- uniqueParentId() gibt die uniqueId() der Annotation oder Komponentenstruktur zurück, der oder die in einer Containerzuordnung enthalten ist. Auch diese integrierte Komponente ist nur im Kontext einer Containerzuordnung gültig.
- [:index] gibt den Index des aktuellen Containereintrags zurück (Bereich oder Liste).

Zugehörige Tasks

„Abrufen von Teilen eines Dokuments, die mit einer semantischen Suchabfrage übereinstimmen" auf Seite 61
Sie können nur die Teile eines Dokuments abrufen, die genau mit der Abfrage übereinstimmen, indem Sie die relevanten Komponentenstrukturen dem Index und der Datenbank zuordnen und den Bereich in der semantischen Suchabfrage angeben.

Filter

Filter werden verwendet, um Zuordnungsregeln in den Dateien für die Zuordnung der allgemeinen Analysestruktur zum Index und zur Datenbank zu beschränken. Nur wenn der Filter wahr ist, werden die Analyseergebnisse dem Index oder einer JDBC-Tabelle hinzugefügt.

Das Element `<filter>` ist optional und wird verwendet, um Zuordnungen nur auf Komponenten zu beschränken, die einen bestimmten Attributwert aufweisen. Das ist sinnvoll, wenn Sie wollen, dass ein Attribut als Schalter dafür fungiert, was indexiert oder was der Datenbank hinzugefügt werden soll. So könnten z. B. Personen und Unternehmen in einer Annotation des Typs `EntityAnnotation` erfasst werden. Ihre Komponente namens `type` wird auf `person` oder auf `organization` gesetzt. Wenn Sie nur die Personen, jedoch nicht die Unternehmen extrahieren wollen, können Sie der Zuordnungsregel den folgenden Filter hinzufügen:

```
<filter syntax="FeatureValue">type = "person"</filter>
```

Jeder Filterausdruck hat die folgende Form:

```
<FeaturePath> <Operator> <Literal>
```

Dabei gilt Folgendes:

- `FeaturePath` ist ein Komponentenpfad in der allgemeinen Analysestruktur.
- `Operator` ist =, !=, <, <=, > oder >=. Beachten Sie, dass < (und nur <) wie folgt ausgedrückt werden muss: `&lt;`.
- `Literal` ist eine ganze Zahl, eine Gleitkommazahl (Exponentensyntax wird nicht unterstützt) oder ein in Anführungszeichen eingeschlossenes Zeichenfolgeliteral. Eingebettete Anführungszeichen und umgekehrte Schrägstriche werden mit einem umgekehrten Schrägstrich als Escapezeichen verwendet.

`<FeaturePath>`, `<Operator>` und `<Literal>` müssen mit einem Leerzeichen getrennt sein.

Die folgenden Beispiele enthalten gültige Filter:

- `<filter syntax="FeatureValue"> foo = "Hallo Welt" </filter>`

  Die Komponente `foo` enthält die Zeichenfolge `Hallo Welt`.
- `<filter syntax="FeatureValue"> foo &lt; 42 </filter>`

  Die Komponente `foo` weist einen ganzzahligen Wert kleiner 42 auf.
- `<filter syntax="FeatureValue"> make/company = "Chevrolet" </filter>`

  Der Komponentenpfad `make/company`, in dem die Komponente `make` eine Komponentenstruktur mit der Komponente `company` enthält, weist den Wert `Chevrolet` auf.
- `<filter syntax="FeatureValue"> bar7 >= 0,5 </filter>`

  Die Komponente `bar7` weist einen Gleitkommawert größer oder gleich 0,5 auf.

Indexzuordnung für benutzerdefinierte Analyseergebnisse

Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe ausgeführt haben, können Sie die Suchmaschine in der Unternehmenssuche verwenden, um einen Index zu erstellen. Verwenden Sie für den Index die Informationen, die in der allgemeinen Analysestruktur gespeichert sind, die von den benutzerdefinierten Analysealgorithmen erstellt wurde.

Indem Sie Ihre Analyseergebnisse Feldern, Textbereichen und Attributen im Index für die Unternehmenssuche zuordnen, können Sie diese Informationen in Abfragen verwenden. Eine Kombination aus benutzerdefinierter Analyse und Unternehmenssuche, die in der Lage ist, sowohl Wörter als auch Textauszüge zu indexieren, ermöglicht die semantische Suche.

Indem Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index verwenden, können Sie festlegen, welche Analyseergebnisse in der allgemeinen Analysestruktur Sie indexieren wollen.

Sie können verschiedene Stile verwenden, um Komponentenstrukturen in der allgemeinen Analysestruktur dem Index für die Unternehmenssuche zuzuordnen.

Annotation
Wenn Sie Komponentenstukturen in der allgemeinen Analysestruktur unter Verwendung des Annotationsstils indexieren, werden alle Annotationen des angegebenen Typs im Index als durchsuchbare Bereiche gespeichert.

Wenn z. B. eine Komponentenstruktur, die einen bestimmten Textbereich umfasst, den Typ `person` aufweist und unter Verwendung des Annotationsstils indexiert wurde, sind die folgenden Abfragen möglich:

Tabelle 2. Beispiele für Abfragen

Gewünschte Information	Mögliche Abfrage
Alle Dokumente, die mindestens einen Personennamen enthalten	<person/>
Alle Dokumente, in denen in der Annotation zu einer Person "boss" vorkommt	<person>boss</person>
Alle Dokumente, in denen "Lang" im gleichen Satz wie einer meiner Konkurrenten genannt wird	<sentence><person>Lang</person> <competitor/></sentence>

Attribute von Komponentenstrukturen werden auch als Teil des Bereichs indexiert. Betrachten Sie z. B. einen Annotator, der Autos aufspürt und die Automarke als Komponente `make` der Annotation `car` speichert. Damit können Sie den folgenden Abfragetyp aktivieren: "Alle Dokumente, in denen Autos der Marke Chevrolet erwähnt werden".

Field Verwenden Sie diesen Stil, wenn Sie den Inhalt von Komponentenstrukturen während der Suche zugänglich machen wollen, indem Sie die feldspezifische Suchfunktionalität der Unternehmenssuche verwenden. Auf diese Weise kann der Inhalt einer Komponentenstruktur in den Suchergebnissen angezeigt oder in der parametrischen Suche verwendet werden.

Wenn Sie z. B. die Dosierungen von Medikamenten einem parametrischen Feld zuordnen, können Sie die folgende Abfrage verwenden: "Alle Dokumente, in denen ein bestimmtes Medikament erwähnt wird, das in einer Dosierung über 100 Milligramm eingenommen wurde."

Breaking
Verwenden Sie diesen Stil, wenn eine bestimmte Komponentenstruktur als deutlicher Begrenzer interpretiert werden soll, z. B. Abschnitte oder Absätze. Die Unternehmenssuche erkennt Sätze und Absätze standardmäßig. Verwenden Sie diesen Stil nur, wenn Ihre benutzerdefinierte Analyse zusätzliche strukturelle Elemente in einem Dokument erkennt, das Sie anders interpretieren wollen.

Analyseergebnisse können auch verwendet werden, um die Rangfolge der Dokumente in der Unternehmenssuche zu beeinflussen, selbst bei einfachen Schlüsselwortabfragen. Sie gehen hierzu in zwei Schritten vor:

1. Ordnen Sie Komponentenstrukturen durchsuchbaren Bereichen oder Feldern zu, indem Sie den Zuordnungsstil **Annotation** oder **Field** verwenden.
2. Definieren Sie eine Boostklasse in der Administrationskonsole für die Unternehmenssuche, und ordnen Sie dieser Boostklasse den Bereichs- oder Feldnamen zu.

Wenn ein Benutzer einen Suchbegriff eingibt, der in dieser Komponentenstruktur enthalten ist, wird das Dokument höher eingestuft. Betrachten Sie z. B. einen Annotator, der Personen- und Firmennamen aufspürt. Wenn Sie diese Komponentenstrukturen Bereichen (wie ″Person″ und ″Unternehmen″) zuordnen, und anschließend diese Bereiche Boostklassen zuordnen, werden Dokumente mit den Suchergebnissen ″Lücke″ höher eingestuft, wenn es sich dabei um das Unternehmen ″Lücke″ handelt, als wenn lediglich der Begriff ″Lücke″ erwähnt wird.

Nachdem Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index erstellt haben, können Sie diese in die Unternehmenssuche hochladen. Verwenden Sie dafür die Administrationskonsole.

Zugehörige Tasks

„Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zum Index″
Indem Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index verwenden, können Sie festlegen, welche Analyseergebnisse in der allgemeinen Analysestruktur Sie indexieren wollen, um die Suche zu aktivieren.

Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zum Index

Indem Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index verwenden, können Sie festlegen, welche Analyseergebnisse in der allgemeinen Analysestruktur Sie indexieren wollen, um die Suche zu aktivieren.

Informationen zu dieser Task

Die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index hat das Format XML. Die Beispieldatei für die Zuordnung der allgemeinen Analysestruktur zum Index basiert auf dem Typsystem, das für das Szenario des Polizeiberichts definiert wurde.

```
<?xml version="1.0" encoding="UTF-8"?>
<indexBuildSpecification
xmlns="http://www.ibm.com/of/822/consumer/index/xml">
 <skipCondition>
   <type>com.ibm.uima.tt.DocumentAnnotation</type>
   <filter syntax="FeatureValue">toBeprocessed = 0</filter>
 </skipCondition>

 <indexBuildItem>
   <name>com.ibm.omnifind.types.Person</name>
   <indexRule>
     <style name="Annotation">
       <attributemappings>
         <mapping>
           <feature>role</feature>
           <indexName>role</indexName>
         </mapping>
         <mapping>
```

```
          <feature>title</feature>
          <indexName>title</indexName>
        </mapping>
        <mapping>
          <feature>gender</feature>
          <indexName>gender</indexName>
        </mapping>
      </attributemappings>
    </style>
  </indexRule>
</indexBuildItem>
<indexBuildItem>
  <name>com.ibm.omnifind.types.Suspect</name>
  <indexRule>
    <style name="Annotation"/>
    <style name="Field">
      <attribute name="parametric" value="false"/>
      <attribute name="fieldSearchable"
        value="true"/>
      <attribute name="returnable" value="true"/>
    </style>
  </indexRule>
</indexBuildItem>
<indexBuildItem>
  <name>com.ibm.omnifind.types.City</name>
  <indexRule>
      <style name="Annotation">
      <attributemappings>
        <mapping>
          <feature>cityDistrict</feature>
          <indexName>district</indexName>
        </mapping>
      </attributemappings>
    </style>
  </indexRule>
</indexBuildItem>
<indexBuildItem>
  <name>com.ibm.omnifind.types.Date</name>
  <indexRule>
    <style name="Field">
      <attribute name="fixedName" value="Date"/>
      <attribute name="fieldSearchable"
          value="true"/>
      <attribute name="returnable" value="true"/>
    </style>
    <style name="Field">
      <attribute name="fixedName" value="hour"/>
      <attribute name="valueFeature"  value="hour"/>
      <attribute name="parametric" value="true"/>
    </style>
  </indexRule>
  <filter syntax="FeatureValue">year="2005"</filter>
</indexBuildItem>
<indexBuildItem>
  <name>com.ibm.omnifind.types.PoliceReport</name>
  <indexRule>
    <style name="Annotation">
      <attribute name="fixedName"
         value="PoliceReport"/>
      <attributemappings>
        <mapping>
          <feature>crimeDescription</feature>
          <indexName>crimeDescription</indexName>
        </mapping>
        <mapping>
          <feature>time/coveredText()</feature>
          <indexName>time</indexName>
```

```
        </mapping>
        <mapping>
          <feature>date/englDate</feature>
          <indexName>date</indexName>
        </mapping>
        <mapping>
          <feature>location/coveredText()</feature>
          <indexName>location</indexName>
        </mapping>
        <mapping>
          <feature>knownSuspects[]/com.ibm.omnifind.types.Suspect:surName</feature>
          <indexName>suspectsLastNames</indexName>
        </mapping>
      </attributemappings>
    </style>
  </indexRule>
 </indexBuildItem>
</indexBuildSpecification>
```

Einschränkungen

Die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index muss alle Analyseergebnisse enthalten, die Sie in Abfragen durchsuchen wollen.

Vorgehensweise

Gehen Sie wie folgt vor, um die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die Zuordnungsdatei heißt `CasToIndexMapping.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*`/packages/uima/configuration_xsd/`.
2. Nehmen Sie Ihre Zuordnungen in ein Element `<indexBuildSpecification xmlns="http://www.ibm.com/of/822/consumer/index/xml">` auf. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element `<skipCondition>` hinzu, um das Indexieren bestimmter Dokumente zu verhindern. Verwenden Sie als Basis einen bestimmten Komponentenwert. Dieses Element ist optional. Im Beispiel werden Dokument nicht indexiert, wenn sie eine Datenstruktur des Typs `com.ibm.uima.tt.DocumentAnnotation` enthalten, deren Komponente `toBeProcessed` auf Null gesetzt ist.
4. Fügen Sie der Struktur im Index mindestens ein Element `<indexBuildItem>` hinzu, das die Zuordnung einer bestimmten Komponentenstruktur in der allgemeinen Analysestruktur enthält.
5. Speichern und prüfen Sie die XML-Datei.

Element <indexBuildItem>

Die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index enthält mindestens ein Element `<indexBuildItem>`. Jedes Element beschreibt die Zuordnung einer bestimmten Komponentenstruktur in der allgemeinen Analysestruktur zu einer Struktur im Index (ein Bereich oder Feld).

Das Element `<name>` enthält den Typ der Komponentenstruktur. Es gibt zwei Möglichkeiten, einen Typ anzugeben:

- Den vollständigen Typnamen. Z. B. com.ibm.omnifind.types.Suspect
- Ein Platzhalterzeichen. Z. B. com.ibm.omnifind.types.*. Das Platzhalterzeichen kann nur am Ende der Typspezifikation hinzugefügt werden.

Verwenden Sie nur Subtypen von uima.tcas.Annotation als Elemente für die Indexerstellung. Wenn eine Komponentenstruktur den Typ uima.cas.TOP (statt uima.tcas.Annotation) aufweist, können Sie auf diese Komponentenstruktur zugreifen, indem Sie einen Komponentenpfad verwenden, der bei einer Annotation startet.

Wenn es sich bei Typ A um einen Subtyp von Typ B handelt (im Beispiel ist com.ibm.omnifind.types.Suspect ein Subtyp von com.ibm.omnifind.types.Person) und für beide Typen Elemente <indexBuildItem> Ia und Ib definiert sind, erfolgt die folgende Verarbeitung:

- Jede Indexierungsregel, die in Ib definiert ist, wird auf die Komponentenstrukturen des Typs B und die des Typs A angewendet
- Jede Indexierungsregel, die in Ia definiert ist, wird nur auf die Komponentenstrukturen des Typs A angewendet

Im Beispiel gilt das Element <indexBuildItem>, das für die Annotationen von com.ibm.omnifind.types.Person definiert ist, auch für die Annotationen von com.ibm.omnifind.types.Suspect. Für eine Annotation für einen Verdächtigen werden zwei Bereiche erstellt: einen für Person, der andere für Suspect.

Das Element <filter> ist optional und wird verwendet, um die Zuordnung von <indexBuildItem> nur auf Komponentenstrukturen zu beschränken, die einen bestimmten Attributwert aufweisen. Das ist sinnvoll, wenn Sie ein Attribut als Schalter verwenden wollen, um anzugeben, was indexiert werden soll. So könnten z. B. Personen und Unternehmen in einer Annotation des Typs EntityAnnotation erfasst werden. Ihre Komponente namens type wird auf person oder auf organization gesetzt. Wenn Sie nur die Personen, jedoch nicht die Unternehmen extrahieren wollen, können Sie den folgenden Filter hinzufügen:

```
<filter syntax="FeatureValue">type = "person"</filter>
```

Weiterhin könnten Sie Personen und Unternehmen unter unterschiedlichen Bereichsnamen indexieren, z. B. person und organization. Definieren Sie hierfür zwei Elemente <indexBuildItem> des Typs EntityAnnotation, und verwenden Sie zwei Filter für die Komponente type als Trigger für Personen oder Unternehmen.

Element <indexRule>

Jedes Element <indexBuildItem> enthält ein Element <indexRule>. Jedes Element <indexRule> enthält alle Informationen, die erforderlich sind, um dem Index eine Komponentenstruktur in der allgemeinen Analysestruktur als Feld-, Annotations- und Unterbrechungsstil zuzuordnen. Der Stil **Annotation** bzw. **Field** unterstützt eine Reihe von Attributen. Den Stil **Term**, der von UIMA Software Development Kit für die Unternehmenssuche unterstützt wird, können Sie nicht verwenden. (Der Stil **Term** wird übersprungen.)

Für den Stil **Annotation** bzw. **Field** gibt es die folgenden Alternativen, um den Annotations- oder Feldnamen im Index anzugeben:

- Verwenden Sie fixedName, wenn jede Komponentenstrukur im Index unter demselben Namen zugänglich sein soll. Im folgenden Beispiel wird jede Komponentenstruktur des Typs com.ibm.omnifind.types.Person einem Bereich "Person" im Index zugeordnet.

```
<indexBuildItem>
   <name>com.ibm.omnifind.types.Person</name>
   <indexRule>
     <style name="Annotation">
       <attribute name="fixedName" value="Person" />
     </style>
   </indexRule>
</indexBuildItem>
```

Damit können Sie Abfragen wie ″Alle Dokumente, in denen ″Boss″ als Name einer Person enthalten ist″ aktivieren. Die Abfrage wird unter Verwendung von XML-Fragmenten wie folgt ausgedrückt: `@xmlf2::'<Person>Boss</Person>'`

- Verwenden Sie `nameFeature`, wenn in der Annotation verschiedene Entitäten gespeichert sind, auf die Sie unter Verwendung verschiedener Bereiche zugreifen wollen, abhängig vom Wert einer bestimmten Komponente der Annotation. Im folgenden Beispiel ist `com.ibm.tt.EntityAnnotation` als Bereich `person` oder `organization` indexiert, was wiederum vom Wert der Komponente `type` abhängig ist. Bei der Komponente kann es sich auch um einen Komponentenpfad handeln.

  ```
  <indexBuildItem>
  <name>com.ibm.tt.EntityAnotation</name>
     <indexRule>
       <style name="Annotation">
         <attribute name="nameFeature" value="type" />
       </style>
     </indexRule>
  </indexBuildItem>
  ```

  Damit können Sie Abfragen wie ″Alle Dokumente über das Unternehmen WHO″ (im Gegensatz zum englischen Begriff ″who″) aktivieren. Die Abfrage wird wie folgt in der eingeschränkten XPath-Syntax ausgedrückt: `@xmlp::'/organization[ftcontains="WHO"]'`

- Wird keines der oben angegebenen Attribute verwendet, wird der Kurzname des Annotationstyps im Element `<indexBuildItem>` verwendet. Das ist der Standard. Beispiel:

  ```
  <indexBuildItem>
  <name>com.ibm.uima.tutorial.RoomNumber</name>
     <indexRule>
       <style name="Annotation" />
       <style name="Field" />
     </indexRule>
   </indexBuildItem>
  ```

  Als Ergebnis des Elements `<indexBuildItem>` sind die Annotationen und Felder namens `RoomNumber` mit dem Text gefüllt, auf den sich `com.ibm.uima.tutorial.RoomNumber` bezieht.

Element <style name="Annotation" />

Der Wert `Annotation` im Element `<style>` gibt an, wie Sie in der Unternehmenssuche auf Bereichsinformationen zugreifen können. Außer, dass Sie die Attribute `fixedName` und `nameFeature` verwenden können, unterstützt dieser Stil auch das Element `<attributemappings>`. In diesem Element ist es möglich, den Wert einer Komponente einem Attribut zuzuordnen, das aus dem resultierenden Bereich im Index stammt und das Sie anschließend in einem Suchausdruck verwenden können.

Jede Zuordnung erfolgt in einem separaten Element <mapping>. Das Element <feature> enthält einen Komponentenpfad, und das Element <indexName> enthält den Namen des Attributs, das im Index verwendet wird, um den Wert von <feature> zu speichern. Beispiel:

```
<mapping>
<feature>make/companyname</feature>
<indexName>company</indexName>
</mapping>
```

Im Element <mapping> wird der Wert der Komponente im Pfad make/companyname direkt im Indexattribut company gespeichert.

Die Zuordnung von Komponentenwerten zu Indexattributen ist besonders sinnvoll, wenn während der Textanalyse ein komplexes Typsystem verwendet wird, das viele verschachtelte Komponentenstrukturen enthält. Wenn Sie das Element <mapping> verwenden, sind die relevanten Attribute ohne Korrelationsnamen, sodass Sie diese in Abfragen verwenden können, ohne detaillierte Kenntnisse der Struktur des ursprünglichen Typsystems zu haben.

Element <style name="Field" />

Der Wert Field im Element <style> gibt an, wie Sie in der Unternehmenssuche auf Feldinformationen zugreifen können. Außer den Attributen fixedName und nameFeature können Sie die folgenden Attribute definieren.

parametric
: Wenn dieses Attribut auf "true" gesetzt ist, kann unter Verwendung der parametrischen Suche nach dem Feldwert gesucht werden, z. B. #dosage:>100

fieldSearchable
: Wenn dieses Attribut auf "true" gesetzt ist, kann der Feldwert in einer Suche verwendet werden, z. B. make:Bayer

returnable
: Wenn dieses Attribut auf "true" gesetzt ist, werden das Feld und seine Werte in den Suchergebnissen zurückgegeben

Bei Feldinformationen kann immer der Inhalt durchsucht werden, das heißt, Feldinformationen sind für die normale Schlüsselwortsuche zugänglich.

Das optionale Attribut valueFeature definiert, welcher Komponentenwert als Feldwert verwendet wird. Wenn es sich bei der Komponentenstruktur um eine Annotation handelt und kein Attribut gesetzt ist, wird der von der Annotation umfasste Text als Feldwert verwendet. Beispiel:

```
<indexBuildItem>
   <name>com.ibm.omnifind.types.Date</name>
   <indexRule>
     <style name="Field">
       <attribute name="fixedName" value="date"/>
       <attribute name="fieldSearchable"
           value="true"/>
       <attribute name="returnable" value="true"/>
     </style>
     <style name="Field">
       <attribute name="fixedName" value="hour"/>
       <attribute name="valueFeature"  value="hour"/>
       <attribute name="parametric" value="true"/>
```

```
    </style>
  </indexRule>
  <filter syntax="FeatureValue">year="2005"</filter>
</indexBuildItem>
```

Für `com.ibm.omnifind.types.Date` werden zwei Felder generiert. Das erste Feld `date` enthält den umfassten Text, z. B. 5:15pm. Das zweite Feld enthält den Wert des Attributs `hour`. Hier können Sie 'hour::<17' in einer Abfrage verwenden.

Element `<style name="Breaking" />`

Der Wert `Breaking` im Element `<style>` enthält keine weiteren Elemente.

Nachdem Sie die XML-Datei erstellt haben, müssen Sie diese in die Unternehmenssuche hochladen und die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index mit Ihren anderen benutzerdefinierten Analyseauswahlen in der Administrationskonsole für die Unternehmenssuche auswählen.

Zugehörige Konzepte

„Indexzuordnung für benutzerdefinierte Analyseergebnisse" auf Seite 41
Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe ausgeführt haben, können Sie die Suchmaschine in der Unternehmenssuche verwenden, um einen Index zu erstellen. Verwenden Sie für den Index die Informationen, die in der allgemeinen Analysestruktur gespeichert sind, die von den benutzerdefinierten Analysealgorithmen erstellt wurde.

„Komponentenpfade" auf Seite 37
Ein Komponentenpfad stellt eine Möglichkeit bereit, auf Komponentenwerte in allgemeinen Analysestrukturen zuzugreifen, ähnlich wie XPath-Anweisungen, die verwendet werden, um auf XML-Elemente in einem XML-Dokument zuzugreifen.

Zugehörige Verweise

„Filter" auf Seite 41
Filter werden verwendet, um Zuordnungsregeln in den Dateien für die Zuordnung der allgemeinen Analysestruktur zum Index und zur Datenbank zu beschränken. Nur wenn der Filter wahr ist, werden die Analyseergebnisse dem Index oder einer JDBC-Tabelle hinzugefügt.

„Beispiel für eine Typsystembeschreibung" auf Seite 26
Die Typsystembeschreibung enthält die Komponentenstrukturen (die zugrunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Datenbankzuordnung für ausgewählte Analyseergebnisse

Nachdem Sie Ihre Dokumente in der Unternehmenssuche analysiert haben, können Sie ausgewählte Textanalyseergebnisse in einer JDBC-fähigen Datenbank speichern.

Diese Version unterstützt DB2 Universal Database Version 8.2.2 (com.ibm.db2.jcc.DB2Driver Version 2.3) oder höher und Oracle 10g (oracle.jdbc.driver.OracleDriver Version 1.0).

Bei DB2 Universal Database und Oracle können Sie auswählen, ob Sie die Analyseergebnisse direkt in die Datenbank einfügen wollen oder ob Sie die funktional entsprechenden datenbankspezifischen Ladedateien und das zugehörige Script generieren wollen, das die Ladebefehle ausführt.

Wenn Sie Ihre Analyseergebnisse Tabellen in einer Datenbank zuordnen, können Sie diese Informationen in weiteren Business-Intelligence-Verarbeitungsschritten verwenden oder damit direkt auf die relevanten Teile eines Dokuments zugreifen, die mit einer semantischen Suchabfrage übereinstimmen.

Die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank enthält die Konfigurationsdaten für die Datenbankverbindung und gibt an, welche benutzerdefinierten Analyseergebnisse in welchen Tabellen und Spalten gespeichert werden. Die Tabellen- und Spaltennamen in Ihrer Zuordnungsdatei müssen den in der Datenbank erstellten Tabellen und Spalten entsprechen.

Nachdem Sie die Zuordnungsdatei erstellt haben, können Sie diese in die Unternehmenssuche hochladen. Verwenden Sie hierfür die Administrationskonsole.

Zugehörige Tasks

„Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank" auf Seite 52
Damit Sie einer Datenbank Analyseergebnisse hinzufügen können, müssen Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank erstellen, die die Konfigurationsdaten für die Datenbankverbindung und eine Beschreibung enthält, welche benutzerdefinierten Textanalyseergebnisse in welchen Datenbanktabellen und -spalten gespeichert werden sollen.

Speichern von Analyseergebnissen in einer Datenbank

Damit Sie ausgewählte Analyseergebnisse in einer JDBC-fähigen Datenbank speichern können, müssen Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank erstellen, die definiert, welche Analyseergebnisse in einer Datenbank gespeichert werden, und die erforderlichen JDBC-Treiberbibliotheken müssen sich in dem Pfad befinden, den Sie in der Zuordnungsdatei definiert haben.

Gehen Sie wie folgt vor, um Analyseergebnisse in einer JDBC-fähigen Datenbank zu speichern:

1. Entscheiden Sie, welche Analyseergebnisse Sie in der Datenbank speichern wollen. Erstellen Sie eine Datenbank, die die Tabellen mit allen erforderlichen Spalten der entsprechenden Datentypen enthält.
2. Verwenden Sie einen XML-Editor zum Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank für die Datenbankkonfigurationsdaten und die Analyseergebnisse, die Sie speichern wollen. Damit Sie ermitteln können, welche Analyseergebnisse in die Zuordnungsdatei aufgenommen werden sollen, müssen Sie wissen, welches zugrunde liegende Typsystem während der Verarbeitung der Dokumente verwendet wird.
3. Stellen Sie die JDBC-Treiberbibliotheken in ein Verzeichnis auf dem Indexierungsknoten, wo das System für die Unternehmenssuche darauf zugreifen kann.
4. Verwenden Sie die Administrationskonsole für die Unternehmenssuche, um die Zuordnungsdatei hochzuladen und auswählen.

Verwenden von Ladedateigruppen

Sie können die Analyseergebnisse direkt in einer JDBC-fähigen Datenbank speichern, oder Sie können die Verarbeitung für die Verwendung von Ladedateigruppen konfigurieren und die Daten zu einem späteren Zeitpunkt in die Datenbank laden.

Die Verwendung von Ladedateigruppen hat die folgenden Vorteile:

- Insgesamt kann eine Gruppe von Ladedateien nie größer sein, als die maximale, vom Betriebssystem unterstützte Dateigröße.
- Sobald eine Ladedateigruppe voll ist, können Sie damit beginnen, die Daten in die Datenbank zu laden, und Sie brauchen den Dokumentparser nicht zu stoppen und erneut zu starten, um Dateizugriffskonflikte zu vermeiden.

Der Wechsel von einer Ladedateigruppe zur nächsten erfolgt auf der Dokumentebene, auch wenn das Dokument auf mehrere allgemeine Analysestrukturen aufgeteilt wird. Nachdem ein Dokument verarbeitet wurde und wenn eine Ladedatei in der aktuellen Ladedateigruppe den Grenzwert überschreitet, wird eine neue Ladedateigruppe verwendet. Auf diese Weise wird die Konsistenz in der Ladedateigruppe gewährleistet. Nachdem der Inhalt einer Ladedateigruppe in die Datenbank geladen wurde, bleibt das Datenmodell konsistent, weil alle Einträge der Originaltabelle die übereinstimmenden Einträge der Datenbanktabelle enthalten.

Die Ladedateien und Scriptdateien werden durch die Dateierweiterung `.cur` angegeben. Wird eine Ladedateigruppe geschlossen, werden die Dateien umbenannt und haben nun die Erweiterung `.dat`. Damit wird angegeben, dass die Dateien auf einen Datenbankserver kopiert oder versetzt werden können, während der Dokumentparser noch ausgeführt wird.

Sie können die Größe der Ladedatei angeben. Wenn die Größenbegrenzung der Ladedatei erreicht ist, wird eine neue Ladedateigruppe gestartet. Sie geben die Größe der Ladedatei in der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank im Abschnitt für das XML-Element `<loadFile>` an. Der Parameter loadFileSize wird über das Element `<loadFileSize>` und in Megabyte angegeben: 10 <= loadFileSize <= 10240 (10 MB <= loadFileSize <= 10 GB). Das Element `<loadFileSize>` ist optional. Ist kein Wert angegeben, wird der Standardwert 1024 MB (1 GB) verwendet.

Die einzelnen Ladedateien einer Gruppe sind mit einer zehnstelligen Ziffer nummeriert, die angibt, welche Datei in welche Ladedateigruppe gehört. Eine Ladedateigruppe wird in den folgenden Fällen geschlossen:

- Wenn eine Ladedatei der Gruppe die definierte Größenbegrenzung überschreitet
- Wenn die Verarbeitung gestoppt wird, weil der Parser gestoppt wurde oder ein Fehler aufgetreten ist

Wird der Parser erneut gestartet, wird die Verarbeitung an der Stelle, an der sie gestoppt wurde, mit einer neuen Ladedateigruppe wieder aufgenommen.

Wichtig: Wenn Sie Cas2Jdbc verwenden, um Ladedateien zu generieren, müssen Sie sicherstellen, dass nur ein Parser-Thread konfiguriert ist. Das Verwenden mehrerer Parser-Threads für eine Objektgruppe, die für die Generierung von Cas2Jdbc-Ladedateien konfiguriert ist, kann ungültige Ladedateien zur Folge haben. Verwenden Sie die Administrationskonsole für die Unternehmenssuche zum Bearbeiten einer Objektgruppe, um anzugeben, wie viele Parser-Threads verwendet werden. Öffnen Sie die Seite **Syntaxanalyse**, wählen Sie die Option zum Konfigurieren der Syntaxanalyseoptionen aus, und geben Sie anschließend 1 für die Anzahl Parser-Threads an.

Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank

Damit Sie einer Datenbank Analyseergebnisse hinzufügen können, müssen Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank erstellen, die die Konfigurationsdaten für die Datenbankverbindung und eine Beschreibung enthält, welche benutzerdefinierten Textanalyseergebnisse in welchen Datenbanktabellen und -spalten gespeichert werden sollen.

Informationen zu dieser Task

Die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank hat das Format XML. Das folgende Beispiel basiert auf dem Typsystem, das für das Szenario des Polizeiberichts definiert wurde.

In diesem Beispiel werden nur die Polizeiberichte und die Städte, die in diesen Berichten angegeben werden, der Datenbank hinzugefügt. In diesem Beispiel wird auch die Verwendung integrierter Komponenten und die Zuordnung des Elements `<constant>` gezeigt.

```
<?xml version="1.0" encoding="UTF-8"?>
<cas2JdbcConfiguration xmlns="http://www.ibm.com/uima/consumer/jdbc/100/xml">
   <databaseConnection>
     <connectionUrl>db2://mein_system:mein_port/meine_datenbank</connectionUrl>
     <driver type="jdbc">com.ibm.db2.jcc.DB2Driver</driver>

     <driverLibraries>
       <driverLibrary>C:\db2\db2jcc.jar</driverLibrary>
       <driverLibrary>C:\db2\db2jcc_license_cu.jar</driverLibrary>
       <driverLibrary>C:\db2\db2jcc_license_cisuz.jar</driverLibrary>
     </driverLibraries>

     <authentication>
       <username>mein_benutzer</username>
       <password>mein_kennwort</password>
     </authentication>

     <loadFile>
       <loadFileDirectory>/home/cas2jdbc/load/</loadFileDirectory>
       <loadFileSize>1048</loadFileSize>
       <loadScript>/home/cas2jdbc/load/load.sh</loadScript>
     </loadFile>

   </databaseConnection>

   <cas2JdbcMappingSpec>
     <skipCondition>
       <name>com.ibm.uima.tt.DocumentAnnotation</name>
       <filter syntax="FeatureValue">toBeProcessed=0</filter>
     </skipCondition>

     <cas2JdbcMappings>
      <explicitMappings>
       <explicitMappingRule applyToSubtypes="false">
        <type>com.ibm.omnifind.types.PoliceReport</type>
        <table>sample.policeReport</table>
        <featureMappings>
         <featureMapping>
          <feature>uniqueId()</feature>
          <column>policeReportId</column>
         </featureMapping>
         <featureMapping>
          <feature>location/uniqueId()</feature>
          <column>crimeLocationId</column>
```

```
        </featureMapping>
       </featureMappings>
       <filter syntax="FeatureValue">location/coveredText()="Los Angeles"</filter>
      </explicitMappingRule>
     </explicitMappings>

     <implicitMappings>
      <implicitMappingRule applyToSubtypes="false">
       <type>com.ibm.omnifind.types.City</type>
        <table>sample.City</table>
       <featureMappings>
        <featureMapping>
         <feature>uniqueId()</feature>
         <column>crimeLocationId</column>
        </featureMapping>
        <featureMapping>
         <feature>coveredText()</feature>
         <column>cityName</column>
         <length>150</length>
        </featureMapping>
        <featureMapping>
         <constant>USA</constant>
         <column>country</column>
        </featureMapping>
       </featureMappings>
      </implicitMappingRule>
     </implicitMappings>

    </cas2JdbcMappings>
   </cas2JdbcMappingSpec>
</cas2JdbcConfiguration>
```

Vorgehensweise

Gehen Sie wie folgt vor, um die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die Zuordnungsdatei heißt `CasToJDBCMapping.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*`/packages/uima/configuration_xsd/`.
2. Nehmen Sie die Zuordnungen in ein Element `<cas2JdbcConfiguration xmlns="http://www.ibm.com/uima/consumer/jdbc/100/xml">` auf. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element `<databaseConnection>` hinzu, das alle Konfigurationsdaten für die Datenbankverbindung enthält, und ein Element `<cas2JdbcMappingSpec>`, das die Zuordnungsregeln für die Analyseergebnisse beschreibt, die in der Datenbank oder den Ladedateien gespeichert werden.
4. Fügen Sie dem Element `<databaseConnection>` die folgenden Komponentenelemente hinzu:
   - Obligatorisch: Ein Element `<connectionUrl>`. Dieses Element enthält die URL für die Datenbankverbindung. Je nachdem, welchen JDBC-Treiber Sie implementiert haben, haben Sie lokalen Zugriff oder Remotezugriff auf die Datenbank.
   - Obligatorisch: Ein Element `<driver>`. Dieses Element enthält den Namen der JDBC-Treiberklasse, z. B. `com.ibm.db2.jcc.DB2Driver` für DB2 oder `oracle.jdbc.driver.OracleDriver` für Oracle.

- Obligatorisch: Ein Element <driverLibraries>. Dieses Element listet die Treiberbibliotheken auf. Jede Bibliothek ist in einem Element <driverLibrary> aufgelistet. Die Bibliotheken befinden sich in Ihrem DB2- oder Oracle-Installationsverzeichnis. Für DB2 gibt es die Bibliotheken c:\ihr_db2-verz\db2jcc.jar, c:\ihr_db2-verz\db2jcc_license_cu.jar und c:\ihr_db2-verz\db2jcc_license_cisuz.jar. Für Oracle muss die Bibliothek c:\ihr_oracle-verz\classes12.zip vorhanden sein.

  Stellen Sie sicher, dass die Treiberbibliotheken stets dieselbe Wartungsstufe aufweisen wie der DB2-Applet-Server.
- Obligatorisch: Ein Element <authentication>. Dieses Element enthält den Benutzernamen und das Kennwort für die Datenbank.
- Optional: Ein Element <loadFile>. Dieses Element enthält die folgenden Komponentenelemente:
  - Das Verzeichnis der Ladedatei in einem Element <loadFileDirectory>.
  - Optional: Die Größe der Ladedatei in einem Element <loadFileSize>. Die Ladedatei hat eine Größenbegrenzung von 10 <= loadFileSize <= 10240 (10 MB <= loadFileSize <= 10 GB). Ist kein Wert definiert, wird der Standardwert 1024 MB (1 GB) verwendet.
  - Der Name des Ladescripts in einem Element <loadScript>.

  Wenn Sie kein Element <loadFile> angeben, werden die gesamten Daten unter Verwendung von JDBC direkt in der Datenbank gespeichert.

  Sie müssen außerdem alle Datenbankkonfigurationsparameter hinzufügen, wenn Sie datenbankspezifische Ladedateien und -scripts verwenden.

5. Fügen Sie dem Element <jdbcMappingSpec> die folgenden Komponentenelemente hinzu:
   - Optional: Ein Element <skipCondition>. Ist keine Bedingung zum Überspringen definiert, werden alle Dokumente verarbeitet.

     ```
     <skipCondition>
       <name>com.ibm.uima.tt.DocumentAnnotation</name>
       <filter syntax="FeatureValue">toBeProcessed=0</filter>
     </skipCondition>
     ```

     In diesem Beispiel werden die Dokumente nicht berücksichtigt, die eine Annotation des Typs com.ibm.uima.tt.DocumentAnnotation enthalten, deren Komponente toBeProcessed auf Null gesetzt ist.
   - Ein Element <cas2JdbcMappings>, das zeigt, welchen Datenbanktabellen und -spalten welche Typen und Komponenten zugeordnet sind. Das Element enthält einen Abschnitt für explizite und einen für implizite Zuordnungen.
6. Fügen Sie ein Element <explicitMappings> hinzu. Dieses Element ist obligatorisch. Es muss mindestens ein Element <explicitMappingRule> aufweisen, das die expliziten Zuordnungen definiert und kann nur für Annotationstypen und ihre Subtypen definiert werden. Ist im Abschnitt für explizite Zuordnungen eine Zuordnung definiert, werden alle Annotationen, die mit der Zuordnungsdefinition übereinstimmen, in der Datenbank gespeichert.
7. Optional: Fügen Sie ein Element <implicitMappings> hinzu. Dieses Element unterstützt alle Komponentenstrukturtypen. Wenn dieses Element vorhanden ist, muss es mindestens ein Element <implicitMappingRule> enthalten. Zuordnungen, die im Abschnitt für implizite Zuordnungen definiert sind, werden der Datenbank nur hinzugefügt, wenn die übereinstimmenden Annotationstypen in einer anderen Annotation angegeben sind, die mit einer expliziten oder einer impliziten Zuordnungsregel übereinstimmt.

   Eine implizite Zuordnung versetzt Sie in die Lage, nur die Analyseergebnisse zu speichern, die in einem bestimmten Kontext auftreten. Wenn z. B. die

Zuordnung für eine Annotation des Typs com.ibm.omnifind.types.City implizit ist, werden nur die Städte in der Datenbank gespeichert, auf die von der Zuordnungsdefinition com.ibm.omnifind.types.PoliceReport im Abschnitt für explizite Zuordnungen verwiesen wird. Das heißt, es werden der Datenbank nur Städte hinzugefügt, die in Polizeiberichten erwähnt werden.

Wenn es sich bei der Zuordnungsregel für die Annotation City um eine explizite Zuordnung handeln würde, würden der Datenbank alle Städte hinzugefügt. In beiden Fällen wird eine Stadt der Datenbank jedoch nur einmal hinzugefügt, auch wenn sie in mehreren Polizeiberichten angegeben wird.

8. Die Elemente <explicitMappingRule> und <implicitMappingRule> müssen das Attribut applyToSubtypes enthalten, das, wenn es auf true gesetzt ist, nicht nur die im Element <type> aufgelistete Komponentenstruktur speichert, sondern auch alle davon abgeleiteten Komponentenstrukturen. Fügen Sie den Elementen <explicitMappingRule> und <implicitMappingRule> die folgenden Komponentenelemente hinzu:
   - Ein Element <type>, das den Typ der Komponentenstruktur enthält.
   - Ein Element <table>, das das Datenbankschema und den Tabellennamen enthält. Die Syntax folgt der Regel schema.tabellenname oder nur tabellenname, wenn kein Schema definiert ist.
   - Ein Element <featureMappings> mit mindestens einem Element <featureMapping> oder einem Element <containerMapping>.
   - Optional: Ein Element <filter>, das eine Bedingung enthält, die immer dann ausgewertet wird, wenn die Zuordnungsregel übereinstimmt. Wenn die Bedingung bei der Auswertung wahr ist, wird die Annotation oder Komponentenstruktur in der Datenbank gespeichert. Im Beispiel werden nur Polizeiberichte in der Datenbank gespeichert, in denen Verbrechen erfasst sind, die in Stuttgart begangen wurden.
9. Die Komponentenstruktur des Elements <featureMapping> hängt davon ab, ob Sie eine Komponente oder eine Konstante zuordnen.

   Wenn Sie eine Komponente oder einen Komponentenpfad zuordnen, gehören die folgenden Elemente zu den Komponentenelementen:
   - Ein Element <feature> mit dem Namen der Komponente. Die Komponente muss für die Komponentenstruktur im Element type definiert sein. Sie können auch ein Komponentenpfadkonstrukt oder eine der integrierten Komponenten verwenden.
   - Optional: Ein Element <length>, das die Länge angibt, die eine Zeichenfolge in der angegebenen Datenbankspalte aufweisen darf. Längere Zeichenfolgen werden abgeschnitten.
   - Ein Element <column> mit dem Namen der Spalte, in der der Komponentenwert gespeichert wird. Datenbankspalten, die nicht in Komponentenzuordnungen verwendet werden, verwenden einen in der Datenbank konfigurierten Standardwert (normalerweise Null).

     Stellen Sie sicher, dass der Wert des Komponentenelements in einer Spalte des entsprechenden Typs gespeichert wird. Der folgenden Tabelle können Sie entnehmen, welche UIMA-Typen mit welchen Datenbanktypen übereinstimmen.

Tabelle 3. Zuordnung von UIMA-Typen zu den entsprechenden Datenbanktypen

UIMA-Typ oder integrierte Komponente	**Empfohlener DB2-Datentyp**	**Empfohlener Oracle-Datentyp**
Float	REAL	FLOAT
String	VARCHAR	VARCHAR2

Tabelle 3. Zuordnung von UIMA-Typen zu den entsprechenden Datenbanktypen (Forts.)

UIMA-Typ oder integrierte Komponente	Empfohlener DB2-Datentyp	Empfohlener Oracle-Datentyp
Integer	INTEGER	INTEGER
uniqueId(), uniqueParentId()	CHAR(27)	CHAR(27)
objectId(), parentId()	CHAR(16)	CHAR(16)
docTimestamp()	BIGINT	LONG
fsId()	INTEGER	INTEGER

Eine Konstante hat die folgenden Komponentenelemente für die Komponentenzuordnung:

- Ein Element `<constant>`, das den Wert einer Konstanten enthält.
- Ein Element `<column>` mit dem Namen der Spalte, der der Wert der Konstanten hinzugefügt wird.

10. Das Element `<containerMapping>` enthält die Zuordnung für eine Containertypkomponente (Bereich oder Liste). Dieses Element darf nur für Containertypen verwendet werden. Es enthält die folgenden Komponentenelemente:
    - Ein Element `<feature>` mit dem Namen der Komponente. Sie können auch ein Komponentenpfadkonstrukt oder eine der integrierten Komponenten verwenden.
    - Ein Element `<table>`, das das Datenbankschema und den Tabellennamen enthält. Die Syntax beachtet die Regel `schema.tabellenname` oder nur `tabellenname`, wenn kein Schema definiert ist.
    - Mindestens ein Element `<featureMapping>`, das die Namen der Komponentenstrukturen und Spalten enthält, denen die Komponenten hinzugefügt werden.
11. Speichern und prüfen Sie die XML-Datei mit dem bereitgestellten Schema.

Nachdem Sie die XML-Datei erstellt haben, müssen Sie diese in die Unternehmenssuche hochladen und die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank mit Ihren anderen benutzerdefinierten Analyseauswahlen in der Administrationskonsole für die Unternehmenssuche auswählen.

Zugehörige Konzepte

„Datenbankzuordnung für ausgewählte Analyseergebnisse" auf Seite 49
Nachdem Sie Ihre Dokumente in der Unternehmenssuche analysiert haben, können Sie ausgewählte Textanalyseergebnisse in einer JDBC-fähigen Datenbank speichern.

„Komponentenpfade" auf Seite 37
Ein Komponentenpfad stellt eine Möglichkeit bereit, auf Komponentenwerte in allgemeinen Analysestrukturen zuzugreifen, ähnlich wie XPath-Anweisungen, die verwendet werden, um auf XML-Elemente in einem XML-Dokument zuzugreifen.

Zugehörige Verweise

„Filter" auf Seite 41
Filter werden verwendet, um Zuordnungsregeln in den Dateien für die Zuordnung der allgemeinen Analysestruktur zum Index und zur Datenbank zu beschränken. Nur wenn der Filter wahr ist, werden die Analyseergebnisse dem Index oder einer JDBC-Tabelle hinzugefügt.

„Integrierte Komponenten" auf Seite 38
Integrierte Komponenten sind vordefinierte Komponentennamen mit einer speziellen Semantik. Sie können verwendet werden, um auf Informationen zuzu-

greifen, die nicht in der Komponentenstruktur selbst enthalten sind, z. B. den Typ der Komponentenstruktur oder den Annotationstext, auf den sie sich beziehen. Sie können in einem Komponentenpfad als letztes oder einziges Element verwendet werden.

„Beispiel für eine Typsystembeschreibung" auf Seite 26
Die Typsystembeschreibung enthält die Komponentenstrukturen (die zugrunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Zuordnung von Containertypen

Ein Containertyp gehört zu den integrierten Bereichs- oder Listentypen in der allgemeinen Analysestruktur. Die Zuordnung von Containertypen bietet eine Möglichkeit, einer relationalen Datenbank Bereichs- oder Listenwerte zuzuordnen.

Es gibt zwei Methoden für die Handhabung von Containertypen in der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank. Eine Methode verwendet die definierten integrierten Komponentenkonstrukte und eine generische Verknüpfungstabelle, die Bereiche oder Listen mit Werten einer Komponentenzuordnungsregel enthält. Da verschiedene Bereiche oder Listen in derselben Verknüpfungstabelle gespeichert werden, sagt die Tabelle nichts über die Relation der gespeicherten Informationen aus.

Bei der zweiten Methode wird die verwendete Definition der Verknüpfungstabelle mit einem Element `<containerMapping>` definiert und gibt die Relation zwischen den angegebenen Informationen an, nach denen Sie suchen.

So wie das folgende Beispiel könnte die Zuordnung für eine generische Verknüpfungstabelle aussehen. Es gibt eine Relation n:m zwischen Polizeiberichten und Verdächtigen, das heißt, ein Verdächtiger kann in mehreren Polizeiberichten angegeben sein, und ein Polizeibericht kann mehrere Verdächtige enthalten.

Die im Beispiel angegebene generische Tabelle `sample.fsarray` ist die Verknüpfungstabelle zwischen Polizeiberichten und Verdächtigen. Wenn ein anderer Zuordnungstyp außer `com.ibm.omnifind.types.PoliceReport` vorhanden ist, der eine Komponente des Typs `com.ibm.omnifind.types.FSArray` aufweist, wird dieser ebenfalls dieser Tabelle zugeordnet. Es ist immer noch möglich, die Tabelle nach der Relation zwischen einem Polizeibericht und einem Verdächtigen ordnungsgemäß abzufragen, es ist jedoch nicht möglich, durch reines Betrachten der Tabelle, den Schluss zu ziehen, dass sie eine Relation oder eine Verknüpfung zwischen Polizeiberichten und möglichen Verdächtigen enthält.

```
<cas2JdbcMappings>
  <explicitMappings>
    <explicitMappingRule applyToSubtypes="false">
      <type>com.ibm.omnifind.types.PoliceReport</type>
      <table>sample.policeReport</table>
      <featureMappings>
        <featureMapping>
          <feature>uniqueId()</feature>
          <column>policeReportId</column>
        </featureMapping>
        <featureMapping>
          <feature>knownSuspects/uniqueId()</feature>
          <column>suspectArrayId</column>
        </featureMapping>
        <featureMapping>
          <feature>location/cityName</feature>
          <column>city</column>
        </featureMapping>
```

```
    </featureMappings>
  </explicitMappingRule>
</explicitMappings>

<implicitMappings>
  <implicitMappingRule applyToSubtypes="false">
    <type>com.ibm.omnifind.types.Suspect</type>
    <table>sample.suspect</table>
    <featureMappings>
      <featureMapping>
        <feature>uniqueId()</feature>
        <column>suspectID</column>
      </featureMapping>
      <featureMapping>
        <feature>surName</feature>
        <column>lastName</column>
      </featureMapping>
      <featureMapping>
        <feature>description</feature>
        <column>description</column>
      </featureMapping>
    </featureMappings>
  </implicitMappingRule>

  <implicitMappingRule applyToSubtypes="false">
    <type>uima.cas.FSArray</type>
    <table>sample.fsarray</table>
    <featureMappings>
      <featureMapping>
        <feature>uniqueId()</feature>
        <column>arrayId</column>
      </featureMapping>
      <featureMapping>
        <feature>[:index]</feature>
        <column>arrayIndex</column>
      </featureMapping>
      <featureMapping>
        <feature>[]/uniqueId()</feature>
        <column>suspectId</column>
      </featureMapping>
    </featureMappings>
  </implicitMappingRule>
</implicitMappings>

</cas2JdbcMappings>
```

Im Folgenden werden die Datenbanktabellen auf der Basis der oben angegebenen generischen Zuordnungsregeln angezeigt.

Tabelle 4. Tabelle 'sample.policeReport'

policeReportId	**suspectArrayId**	**city**
aaa...1	**bbb...1**	Sindelfingen
aaa...2	**bbb...2**	Leonberg

Tabelle 5. Tabelle 'sample.fsarray'

arrayId	**arrayIndex**	**suspectId**
bbb...1	1	*ccc...1*
bbb...1	2	*ccc...2*
bbb...2	1	*ccc...3*

Tabelle 6. Tabelle 'sample.suspect'

suspectID	lastname	description
ccc...1	Braun	Dunkelhäutig
ccc...2	Schmidt	Brillenträger
...	...	...

Das Beispiel zeigt die Zuordnung für Komponentenstrukturbereiche. Sie können diesen Zuordnungstyp auch auf `StringArray`, `IntegerArray` und `FloatArray` anwenden. Wenn Sie für diese Bereiche mit einfachen Werten Zuordnungsregeln angeben, ersetzen Sie `[]/uniqueId()` mit `[]`.

Dieselbe Methode für generische Tabellen kann für Komponentenstrukturlisten verwendet werden, sowie für Listen mit einfachen Typen (`StringList`, `IntegerList` und `FloatList`).

Eine einfachere Möglichkeit, Relationen zu handhaben, besteht darin, ein Element für die explizite Containerzuordnung zu verwenden, das die Iteration für die in den Bereichen oder Listen enthaltenen Elementen definiert.

Im folgenden Beispiel wird gezeigt, wie eine Zuordnung aussieht, in der eine explizite Verknüpfungstabelle angegeben wird. Auch hier gibt es wieder die Relation n:m zwischen Polizeiberichten und Verdächtigen. Jedoch ist in diesem Fall die Tabelle `sample.reports_suspects` die Verknüpfungstabelle zwischen Polizeiberichten und Verdächtigen.

Bei dieser Methode brauchen Sie keine Bereichs-IDs oder Zuordnungen für Kopfsatz- und Nachsatzeinträge für Listentypen zu berücksichtigen. Die Verknüpfungstabelle enthält eine explizite Relation.

```
<cas2JdbcMappings>
  <explicitMappings>
    <explicitMappingRule applyToSubtypes="false">
      <type>com.ibm.omnifind.types.PoliceReport</type>
      <table>sample.policeReport</table>
      <featureMappings>
        <featureMapping>
          <feature>uniqueId()</feature>
          <column>policeReportID</column>
        </featureMapping>
        <featureMapping>
          <feature>location/cityName</feature>
          <column>city</column>
        </featureMapping>
        <featureMapping>
          <feature>knownSuspects</feature>
          <containerMapping>
            <table>sample.reports_suspects</table>
            <featureMapping>
              <feature>com.ibm.omnifind.types.PoliceReport
                /objectId()</feature>
              <column>policeReportId</column>
            </featureMapping>
            <featureMapping>
              <feature>knownSuspects/[]/objectId()</feature>
              <column>suspectId</column>
            </featureMapping>
          </containerMapping>
        </featureMapping>
      </featureMappings>
```

```
    </explicitMappingRule>
  </explicitMappings>

  <implicitMappings>
    <implicitMappingRule applyToSubtypes="false">
      <type>com.ibm.omnifind.types.Suspect</type>
      <table>sample.suspect</table>
      <featureMappings>
        <featureMapping>
          <feature>objectId()</feature>
          <column>suspectID</column>
        </featureMapping>
        <featureMapping>
          <feature>surName</feature>
          <column>lastName</column>
        </featureMapping>
        <featureMapping>
          <feature>description</feature>
          <column>description</column>
        </featureMapping>
      </featureMappings>
    </implicitMappingRule>
  </implicitMappings>

</cas2JdbcMappings>
```

Ein Element `<containerMapping>` wird verwendet, um die Iteration für Elemente zu definieren, die in dem Bereich enthalten sind. Im Beispiel enthält die Verknüpfungstabelle `sample.reports_suspects` eine Verknüpfung zu den Spalten `policeReportId` und `suspectId`. Die Elemente `<containerMapping>` dürfen nicht verschachtelt werden.

Im Folgenden werden die Datenbanktabellen auf der Basis der expliziten Zuordnungsregeln der Verknüpfungstabelle angezeigt.

Tabelle 7. Tabelle 'sample.policeReport'

policeReportId	**city**
aaa...1	Sindelfingen
aaa...2	Leonberg

Tabelle 8. Tabelle 'sample.reports_suspect'

policeReportId	**suspectId**
bbb...1	*ccc...1*
bbb...2	*ccc...2*
...	...

Tabelle 9. Tabelle 'sample.suspect'

suspectID	**lastname**	**description**
ccc...1	Braun	Dunkelhäutig
ccc...2	Schmidt	Brillenträger
...	...	...

Zugehörige Verweise

„Integrierte Komponenten" auf Seite 38
Integrierte Komponenten sind vordefinierte Komponentennamen mit einer spe-

ziellen Semantik. Sie können verwendet werden, um auf Informationen zuzugreifen, die nicht in der Komponentenstruktur selbst enthalten sind, z. B. den Typ der Komponentenstruktur oder den Annotationstext, auf den sie sich beziehen. Sie können in einem Komponentenpfad als letztes oder einziges Element verwendet werden.

Abrufen von Teilen eines Dokuments, die mit einer semantischen Suchabfrage übereinstimmen

Sie können nur die Teile eines Dokuments abrufen, die genau mit der Abfrage übereinstimmen, indem Sie die relevanten Komponentenstrukturen dem Index und der Datenbank zuordnen und den Bereich in der semantischen Suchabfrage angeben.

Wenn Sie auf alle Instanzen eines bestimmten Annotationstyps in den Suchergebnissen zugreifen wollen, z. B. um alle Personen abzurufen, fügen Sie dem Annotationstyp eine Zuordnung des Stils **Field** hinzu, und markieren Sie diesen in der Datei für die Zuordnung der allgemeinen Analysestruktur zum Index als zurückzugebend (`returnable`). Beispiel:

```
<indexBuildItem>
 <name>com.ibm.omnifind.types.Person</name>
 <indexRule>
  <style name="Annotation"/>
  <style name="Field">
   <attribute name="returnable" value="true"/>
  </style>
 </indexRule>
</indexBuildItem>
```

In diesem Beispiel werden die Annotationen des Typs `com.ibm.omnifind.types.Person` im Index für die Unternehmenssuche dem Bereich `Person` zugeordnet, wo während der semantischen Suche auf sie zugegriffen werden kann. Darüber hinaus wird der Text, auf den sich die Annotationen beziehen, z. B. der vollständige Name der Person, als zurückzugebendes Feld gespeichert. Wenn Sie diese Annotationswerte abrufen wollen, führen Sie den Aufruf `getFields("Person")` für jedes Ergebnisobjekt aus, das von der Suchabfrage (Schlüsselwort- oder semantische Suche) zurückgegeben wird. Bei dieser Methode wird ein Bereich `String` mit den Annotationswerten zurückgegeben, in diesem Fall mit den Personennamen.

Diese Methode gibt jedoch alle Instanzen eines angegebenen Annotationstyps zurück und ist deshalb nicht geeignet, wenn Sie Ihre Ergebnisverarbeitung auf Dokumente begrenzen wollen, die genau mit der Abfrage übereinstimmen. So können in einem Dokument z. B. fünf Personen erwähnt werden. Der Benutzer gibt jedoch in der semantischen Suchabfrage '`<sentence><person/>IBM</sentence>` an, dass er nur an einer Person interessiert ist, die im selben Satz erwähnt wird, in dem auch der Begriff IBM erwähnt wird. An den anderen Personen ist der Benutzer nicht interessiert.

Gehen Sie wie folgt vor, um auf Komponentenstrukturen zuzugreifen, die genau mit der Abfrage übereinstimmen und diese zu verarbeiten:

1. Ordnen Sie die relevanten Komponentenstrukturtypen dem Index für die Unternehmenssuche zu, indem Sie den Zuordnungsstil `Annotation` verwenden. Beispiel:

```
<indexBuildItem>
 <name>com.ibm.omnifind.types.Person</name>
 <indexRule>
  <style name="Annotation"/>
 </indexRule>
</indexBuildItem>
```

2. Ordnen Sie die relevanten Komponentenstrukturtypen den JDBC-Tabellen zu. Ein Teil der Zuordnung besteht darin, dass Sie zwei Spalten für den Dokument-URI und die Komponentenstruktur-ID einfügen müssen. Obwohl Sie alle Komponentenstrukturtypen einer einzigen Datenbanktabelle zuordnen können, sollten Sie jeden Typ einer anderen Tabelle zuordnen. Beispiel:

```
<explicitMappingRule applyToSubtypes="false">
 <type>com.ibm.omnifind.types.Person</type>
 <table>sample.person</table>
 <featureMappings>
  <featureMapping>
   <feature>objectId()</feature>
   <column>primaryId</column>
  </featureMapping>
  <!-- Contains the covered text of the annotation-->
  <featureMapping>
   <feature>coveredText()</feature>
   <column>personName</column>
  </featureMapping>
  <!-- Other mapping go in here-->
  <!-- To access the relevant person annotations in the query result-->
  <featureMapping>
   <feature>docUri()</feature>
   <column>docUri</column>
  </featureMapping>
  <featureMapping>
   <feature>fsId()</feature>
   <column>annotationId</column>
  </featureMapping>
 </featureMappings>
</explicitMappingRule>
```

3. Führen Sie für die Dokumente eine Crawlersuche, eine syntaktische Analyse und eine Indexierung aus.
4. Rufen Sie die IDs der Instanzen ab, die mit der Abfrage übereinstimmen. In Search and Index API (SIAPI) werden diese Instanzen als Zielelemente bezeichnet. Ein Zielelement gibt den zurückzugebenden Eingabebereich an. Es wird wie folgt definiert:
   - In XML-Fragmenten wird das Zielelement von einem vorangestellten Nummernzeichen (#) angegeben. Das Nummernzeichen ist nur einmal zulässig und kann an beliebiger Stelle in der XML-Fragmentabfrage stehen. Beispiel: `$xmlf2::'<sentence><#person/>IBM</sentence>'`
   - In XPath ist das Zielelement standardmäßig das letzte Feld im XPath-Ausdruck.
   - Verwenden Sie die folgende Methode, um auf diese Instanzen zuzugreifen: `Result.getProperty("TargetElement")`. Das zurückgegebene Merkmal ist eine Verkettung von Zeichenfolgen aller Vorkommen von IDs, die durch Leerzeichen getrennt sind. Jedes Vorkommen im Merkmal kann in einen ganzzahligen Wert umgesetzt werden.
5. SIAPI gibt nicht die Komponentenstrukturen selbst zurück, sondern nur die IDs ihrer Vorkommen. Diese IDs entsprechen dem Wert `fsId()`, der in der Datenbanktabelle gespeichert ist. Ihre Anwendung muss die folgenden Schritte ausführen, um diese Instanzen und ihre zugehörigen Informationen abzurufen:
   a. Abhängig vom Bereichsnamen des Zielelements die richtige Datenbanktabelle auswählen. Im Beispiel enthält die Anwendung eine Zuordnung von

person zur Tabelle sample.person. Diese Informationen werden von der Datei für die Zuordnung der allgemeinen Analysestruktur zum Index abgeleitet, die den Bereichsnamen enthält, und von der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank, die den Tabellennamen enthält.

b. Für jedes Ergebnisobjekt im Suchergebnis die folgenden Schritte ausführen:
   1) Die von Result.getProperty ("TargetElement") zurückgegebene Zeichenfolge syntaktisch analysieren, um allen Element-IDs zu suchen.
   2) Eine SELECT-Anweisung für die Tabelle absetzen, indem die Ergebnis-URI (verfügbar über Result.getDocumentId()) als Wert in der Spalte docUri und die Element-IDs als Wert in der Spalte annotationId verwendet werden. Die Spaltennamen richten sich nach Ihrer Zuordnungsdatei. Die Spaltennamen wurden dem oben angegebenen Beispiel entnommen.

   Die zurückgegebene Zeile enthält die Informationen, die für die Komponentenstruktur gespeichert wurden, z. B. der umfasste Text oder bestimmte Attribute der Komponentenstruktur, wie "Nachname" oder "Geburtsort".

Stellen Sie sicher, dass die Aktualisierungen Ihrer Datenbank mit den Aktualisierungen des Index für die Unternehmenssuche synchronisiert werden. Wenn die Datenbank veraltete Informationen enthält (z. B. wenn Sie Datenbankladedateien verwendet und die Datenbank nicht aktualisiert haben, aber den Index aktualisiert angezeigt oder reorganisiert haben), werden möglicherweise nicht alle Element-IDs in der Datenbank gefunden. Die Unternehmenssuche speichert nur die letzte Dokumentversion im Index. Das heißt, die Element-IDs gelten nur für das letzte Dokument.

Wenn Sie mehrere Versionen eines Dokuments in derselben Datenbanktabelle speichern, gibt es möglicherweise mehrere Zeilen, die übereinstimmende Element-IDs aufweisen, von denen jede aus einer anderen Version des Dokuments stammt. In diesem Fall müssen Sie eine Spalte für die Dokumentversion definieren und diese füllen, indem Sie die Anwendungslogik oder integrierte Komponenten wie docTimestamp() verwenden. Auf diese Weise können Sie die Ergebnisse filtern, sodass nur die letzte Dokumentversion abgerufen wird.

Zugehörige Konzepte

„Begriff der semantischen Suchabfrage" auf Seite 64
Der Begriff der semantischen Suchabfrage wird als nicht transparenter Begriff übertragen.

Zugehörige Tasks

„Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zum Index" auf Seite 43
Indem Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index verwenden, können Sie festlegen, welche Analyseergebnisse in der allgemeinen Analysestruktur Sie indexieren wollen, um die Suche zu aktivieren.

„Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank" auf Seite 52
Damit Sie einer Datenbank Analyseergebnisse hinzufügen können, müssen Sie die Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank erstellen, die die Konfigurationsdaten für die Datenbankverbindung und eine Beschreibung enthält, welche benutzerdefinierten Textanalyseergebnisse in welchen Datenbanktabellen und -spalten gespeichert werden sollen.

Semantische Suchanwendungen

Vier Typen von Dokumentinformationen sind im Index für die Unternehmenssuche gespeichert, die Sie mit Suchanwendungen abfragen können, indem Sie Search and Index API (SIAPI) verwenden.

Die vier verschiedenen Informationstypen umfassen folgende Elemente:

- Textwörter, die in einem Dokument vorkommen, z. B. der Ausdruck *Computers-oftware*.
- Bereichsnamen, z. B. ein XML-Dokument, das <author>James</author> enthält, gibt den Bereich <author> zurück.
- Attributnamen, z. B. ein XML-Dokument, das <author countryOfBirth=USA>James</author> enthält, gibt das Attribut "countryOf-Birth" zurück.
- Attributwerte, z. B. ist USA der Wert des Attributs "countryOfBirth".

Die SIAPI-Abfragesprache enthält den Begriff der semantischen Suchabfrage. Der Begriff gibt ein Zweigmuster an. Ein Zweig ist ein kleiner Baum mit Blättern. Jedes Blatt stellt die vier Informationstypen (Textwörter, Bereichsnamen usw.) dar. Die internen Knoten des Baums geben an, welche Beziehung ihre Vorkommen in einem Dokument zueinander haben. Es gibt fünf interne Knotentypen, die Beziehungen angeben:

- and
- or
- not
- in_the_span_of
- attribute_in_the_span_of

Ein Dokument entspricht einem angegebenen semantischen Suchbegriff, wenn es Vorkommen dieser Blätter aufweist und die von den internen Knoten angegebenen Integritätsbedingungen (die definierten Beziehungen) erfüllt sind.

Die semantische Suchabfrage trägt dazu bei, besser geeignete Dokumente abzurufen. Sie können nun, neben der Suche unter Verwendung Boolescher Kombinationen des Wortes mit Annotationen, auch Dokumente abrufen, in denen z. B. *James* im Bereich `Autor` vorkommt oder in denen die Begriffe *IBM* und *Suche* im selben Satz enthalten sind.

Begriff der semantischen Suchabfrage

Der Begriff der semantischen Suchabfrage wird als nicht transparenter Begriff übertragen.

Es gibt zwei Syntaxformen, um einen nicht transparenten Begriff in Search and Index API (SIAPI) auszudrücken:

- XML-Fragmente
- Xpath (eingeschränkt)

Der XML-Fragmentabfragebegriff sieht wie ein gut ausbalanciertes Fragment eines XML-Dokuments aus. Ein XML-Fragmentabfragebegriff hat als Präfix das nicht transparente Begriffszeichen `@xmlf2::`, dem der in einfache Anführungszeichen eingeschlossene XML-Fragmentausdruck folgt ('...').

Die eingeschränkten XPath-Abfragebegriffe haben das Präfix `@xmlxp::`, dem die in einfache Anführungszeichen eingeschlossene XPath-Abfrage folgt ('...').

Wie ein allgemeiner Abfragebegriff in SIAPI kann jeder Begriff einen Modifikator für seine Darstellung haben:

Pluszeichen (+)
Der Begriff muss enthalten sein.

Präfix =
Bei dem Begriff muss es sich um eine exakte Übereinstimmung handeln.

Tilde als Präfix (~)
Es dürfen auch Synonyme des Abfragebegriffs berücksichtigt werden.

Tilde als Erweiterung (~)
Es dürfen auch Wörter berücksichtigt werden, die dasselbe Lemma wie der Abfragebegriff haben.

Nummernzeichen (#)
Der Begriff ist hervorgehoben.

Die folgenden Beispiele enthalten XML-Fragmentabfragen.

@xmlf2::'<City>Stuttgart</City>'
Sucht nach Dokumenten, die den Bereich (die Annotation) `City` mit der Zeichenfolge *Stuttgart* enthalten.

@xmlf2::'<Person gender="female"/>'
Sucht nach Dokumenten, in deren Annotationen eine weibliche Person vorkommt.

@xmlf2::'<Person><.or><@gender>female</@gender> <@title>Mrs</@title><@title>Ms</@title></.or></Person>'
Sucht nach Dokumenten, in denen eine Person anhand der Geschlechtsangabe (`gender`) oder der Anrede (`title`) als Frau erkannt wird.

@xmlf2::'<Person gender="male" role="suspect"/> <PoliceReport><@crimeDescription><.or>Raub Diebstahl</.or>-Unfall </@crimeDescription></PoliceReport> <City>Stuttgart<.or> <@district>Botnang</@district><@district>Feuerbach</@district></.or></City>'
Sucht nach Dokumenten, in denen männliche Personen vorkommen, die als Verdächtige (`suspect`) eingestuft wurden, sowie eine Annotation `PoliceReport`, in der die Zeichenfolge *Raub* oder *Diebstahl* im Attribut `crimeDescription` enthalten ist, nicht jedoch die Zeichenfolge *Unfall*. Außerdem müssen Dokumente eine Annotation `city` mit dem Textwort *Stuttgart* und einer Annotation enthalten, die das Attribut `district` mit dem Wert *Botnang* oder *Feuerbach* umfasst.

Die entsprechenden XPath-Abfragen haben die folgende Struktur:

@xmlxp::'//City ftcontains ("Stuttgart")'
Sucht nach Dokumenten, die den Bereich (die Annotation) `City` mit der Zeichenfolge *Stuttgart* enthalten.

@xmlxp::'//PoliceReport[City ftcontains("Stuttgart")]'
Sucht nach Dokumenten, die den Bereich (die Annotation) `City` im Bereich `PoliceReport` mit der Zeichenfolge *Stuttgart* enthalten.

@xmlxp::'//Person[@gender="female" or @title ftcontains("Ms") or @title ftcontains("Mrs")]'
Sucht nach Dokumenten, in deren Annotationen eine weibliche Person vor-

kommt. Im Attribut `gender` muss der Wert genau übereinstimmen, während für das Attribut `title` *Ms* und *Mrs* nicht genau mit dem Attributwert übereinstimmen müssen.

Synonymunterstützung in Suchanwendungen

Sie können die Suchergebnisse erweitern, indem Sie nach Dokumenten suchen, die Synonyme der Abfragebegriffe enthalten.

Zu Synonymen zählen in der Regel Mehrwortbegriffe, z. B. Produktnamen wie *OmniFind Enterprise Edition*. Mehrwortbegriffe, die im Synonymverzeichnis enthalten sind, werden in Benutzerabfragen richtig identifiziert und müssen nicht in Anführungszeichen gesetzt werden.

Die SIAPI-Schnittstelle für die Unternehmenssuche unterstützt verschiedene Methoden, mit denen Benutzer nach Synonymen der Abfragebegriffe suchen können:

- Die SIAPI-Abfragesyntax unterstützt die Tilde (~) als Operator für die Synonymerweiterung. Wenn der Benutzer die Tilde einem Abfragebegriff voranstellt, wird für dieses Wort eine Synonymerweiterung durchgeführt. Die Abfrage `~WAS` beispielsweise gibt Dokumente zurück, die WebSphere Application Server und andere vorhandene Synonyme für diese Abkürzung behandeln.
- Die Synonymerweiterung kann über die SIAPI-Synonymerweiterungsschnittstelle in einer Suchanwendung aktiviert werden. Abfragebegriffe können automatisch so erweitert werden, dass sie Synonyme mit einschließen, oder die Suchanwendung kann Optionen enthalten, mit denen der Benutzer angeben kann, ob Synonyme der Abfragebegriffe als Suchergebnisse zurückgegeben werden sollen.

  Bei der automatischen Synonymerweiterung wird die Synonymsuche für alle Abfragewörter durchgeführt. Die Suchergebnisse enthalten Dokumente, die die Abfragebegriffe oder Synonyme der Abfragebegriffe enthalten. SIAPI unterstützt auch die Erzeugung einer Liste mit Synonymerweiterungen für die übergebene Abfrage.
- Die Synonymerweiterung in N-Gram-Objektgruppen ermöglicht die Ausdruckssegmentierung des Abfragetexts. Wenn im Synonymverzeichnis ein vollständiger Ausdruck enthalten ist, ist die Suche erfolgreich. Ein Ausdruck wird entsprechend der folgenden Begrenzungszeichen extrahiert:

  Interpunktion
  Die folgenden Zeichen sind Begrenzungszeichen: `- ( ) + . ,`

  Anführungszeichen werden ignoriert und dienen nicht als Begrenzer für Ausdrücke.

  Änderung im Alphabet
  Beispiel: Für eine N-Gram-Objektgruppe wird die Abfrage so erweitert, dass die Synonyme für `ABC` in den folgenden Musterabfragen eingefügt werden, wenn `ABC` sich im Synonymverzeichnis befindet:

  `ABC run DEF stand` (Dabei stehen `ABC` und `DEF` für japanischen Text.)
  `ABC+DCF+GHI`

Erstellen einer XML-Datei für Synonyme

Zum Erweitern von Abfragen in einer Unternehmenssuche um Synonyme der Abfragebegriffe müssen Sie in einer XML-Datei festlegen, welche Wörter als Synonyme voneinander gelten. Die XML-Datei wird zum Erstellen einer binären Wörterverzeichnisdatei verwendet, die Sie in die Unternehmenssuche hochladen und entsprechenden Objektgruppen zuordnen.

Informationen zu dieser Task

Die XML-Datei, die die Synonyme auflistet, muss einem bestimmten Schema entsprechen. XML-Beispieldatei für Synonyme:

```
<?xml version="1.0" encoding="UTF-8"?>
<synonymgroups xmlns="http://www.ibm.com/of/822/synonym/xml">
  <synonymgroup>
    <synonym>Think Pad</synonym>
    <synonym>Notebook</synonym>
    <synonym>Notebooks</synonym>
  </synonymgroup>
  <synonymgroup>
    <synonym>WebSphere Application Server</synonym>
    <synonym>WAS</synonym>
  </synonymgroup>
</synonymgroups>
```

Einschränkungen

Wörter, die synonym sind (Elemente `<synonym>`), müssen Sie in einem Element `<synonymgroup>` zusammenfassen. Ein Synonym kann Leerzeichen enthalten, jedoch keine Interpunktionszeichen wie z. B. Kommata (,) oder senkrechte Striche (|), da diese die Abfragesyntax der Unternehmenssuche stören würden.

Sie müssen alle möglichen Beugungen der Begriffe aufführen, die Sie als Synonym hinzufügen, wie z. B. die Singular- und Pluralformen der Wörter. Sie müssen jedoch weder die normalisierten Formen eines Begriffs, wie z. B. das Entfernen von Akzenten und Umlauten (die Unternehmenssuche normalisiert automatisch), noch groß und klein geschriebene Varianten eines Begriffs auflisten. Wenn Sie beispielsweise das Wort ″météo″ als Synonym hinzufügen möchten, müssen Sie nicht auch das Wort METEO aufnehmen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Synonymliste für die Unternehmenssuche zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die XML-Datei heißt `synonyms.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*`/packages/uima/configuration_xsd/`.
2. Fügen Sie ein Element `<synonymgroup>` hinzu, und fügen Sie dann für jedes Wort, das in der Synonymgruppe als Synonym für andere Wörter behandelt werden soll, ein Element `<synonym>` ein.

   Achten Sie darauf, Ihre Zuordnungen in ein Element `<synonymgroups xmlns=″http://www.ibm.com/of/822/synonym/xml″>` aufzunehmen. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.

3. Wiederholen Sie den vorigen Schritt, bis Sie alle Synonyme angegeben haben, die Sie für die Suche nach Dokumenten in einer Objektgruppe für die Unternehmenssuche verwenden wollen.
4. Speichen und beenden Sie die XML-Datei.

Nach dem Erstellen der XML-Datei müssen Sie sie in ein Synonymverzeichnis umwandeln, das Sie dem System für die Unternehmenssuche hinzufügen können.

Erstellen eines Synonymverzeichnisses

Nach dem Erstellen oder Aktualisieren einer Synonymliste in einer XML-Datei müssen Sie die XML-Datei in ein binäres Synonymverzeichnis umwandeln.

Informationen zu dieser Task

Verwenden Sie für die Erstellung eines Synonymverzeichnisses das Befehlszeilentool 'essyndictbuilder', das zusammen mit OmniFind Enterprise Edition geliefert wird. Das Tool befindet sich im Verzeichnis *ES_INSTALL_ROOT*/bin.

Als Eingabe für das Tool dient die XML-Datei, die die Synonyme auflistet. Die Ausgabe ist ein Synonymverzeichnis. Das Wörterverzeichnis muss das Suffix .dic haben. Beispiel: c:\eigeneVerzeichnisse\produkte.dic.

Standardspeicherort für beide Dateien ist das Verzeichnis, in dem das Script aufgerufen wird. Wenn bereits ein Verzeichnis mit dem gleichen Namen vorhanden ist, generiert das Script einen Fehler.

Die maximale Größe einer DIC-Datei bei der Unternehmenssuche ist 8 MB.

Vorgehensweise

Gehen Sie wie folgt vor, um ein Synonymverzeichnis für die Unternehmenssuche zu erstellen:

1. Melden Sie sich am Indexserver als Administrator für die Unternehmenssuche an. Diese Benutzer-ID wurde bei der Installation von OmniFind Enterprise Edition angegeben.
2. Geben Sie den folgenden Befehl ein. Dabei ist *xml-datei* der vollständig qualifizierte Pfad zur XML-Datei mit der Synonymliste und *dic-datei* der vollständig qualifizierte Pfad zum Synonymverzeichnis.

   AIX, Linux oder Solaris: essyndictbuilder.sh *xml-datei dic-datei*
   Windows: essyndictbuilder.bat *xml-datei dic-datei*

Fügen Sie nach dem Erstellen des Synonymverzeichnisses über die Administrationskonsole für die Unternehmenssuche das Synonymverzeichnis dem System für die Unternehmenssuche hinzu, und ordnen Sie es mindestens einer Objektgruppe zu.

Nur die generierte DIC-Datei wird auf das System für die Unternehmenssuche hochgeladen. Stellen Sie sicher, dass die XML-Datei in einer Umgebung mit Zugriffssteuerung aufbewahrt wird, und sichern Sie die Datei regelmäßig. Sie brauchen diese XML-Datei, um Ihr Synonymverzeichnis zu aktualisieren.

Benutzerdefinierte Verzeichnisse von Stoppwörtern

Sie können ein unternehmensspezifisches Vokabular definieren, das aus einer Abfrage entfernt wird, um die Suchrelevanz zu erhöhen.

Es gibt zwei Arten der Stoppwortunterstützung in der Unternehmenssuche:

- Die sprachspezifische Stoppworterkennung, die alle häufig verwendeten Wörter, wie *ein* und *der* aus einer Mehrwortabfrage entfernt. Die Verzeichnisse von Stoppwörtern, die für die einzelnen Sprachen vorhanden sind, können von den Benutzern nicht geändert werden. Diese Stoppworterkennung wird für alle Abfragen automatisch ausgeführt, um die Suchrelevanz zu verbessern.
- Die benutzerdefinierte oder angepasste Stoppworterkennung, die unternehmensspezifisches Vokabular aus Abfragen entfernt. Dieses Verzeichnis von Stoppwörtern, das vom Administrator definiert wird, kann nur ein spezielles Vokabular enthalten. Das benutzerdefinierte Verzeichnis von Stoppwörtern ersetzt in der Unternehmenssuche nicht die sprachspezifischen Verzeichnisse von Stoppwörtern, die allgemeine Wörter enthalten. Benutzerdefinierte Verzeichnisse von Stoppwörtern sind sprachunabhängig.

Zu benutzerdefinierten Stoppwörtern zählen in der Regel Mehrwortbegriffe, z. B. Produktnamen wie *OmniFind Enterprise Edition*. Mehrwortbegriffe, die im Verzeichnis von Stoppwörtern enthalten sind, werden in Benutzerabfragen richtig identifiziert und müssen nicht in Anführungszeichen gesetzt werden.

Auch die zusammengesetzten Begriffe der germanischen Sprachen werden in Abfragen richtig identifiziert. Ein zusammengesetzter Begriff ist eine Kombination aus mindestens zwei Begriffen, die wie ein einziger Begriff verwendet werden. Lexikalisierte Zusammensetzungen wie *Reisebüro* gelten nicht als zusammengesetzte Begriffe.

In einer Abfrage werden die zusammengesetzten Begriffe in die Einzelbegriffe zerlegt, aus denen sie zusammengesetzt sind. Falls einer der Einzelbegriffe eines zusammengesetzten Begriffs im Verzeichnis von Stoppwörtern enthalten ist, wird der zusammengesetzte Begriff nicht aus der Abfrage entfernt.

So gibt z. B. der Abfragebegriff `Versicherungspolice` Dokumente zurück, die die zusammengesetzten Begriffe *Lebensversicherungspolice* und *Haftpflichtversicherungspolice* enthalten. Selbst wenn der Begriff *Police* im Verzeichnis von Stoppwörtern aufgeführt ist, wird der zusammengesetzte Abfragebegriff `Versicherungspolice` nicht aus der Abfrage entfernt.

Sie müssen das unternehmensspezifische Vokabular in einer XML-Datei auflisten, die Sie anschließend in ein Verzeichnis von Stoppwörtern konvertieren müssen, damit dieses dem System für die Unternehmenssuche hinzugefügt werden kann.

Über die Administrationskonsole für die Unternehmenssuche können Sie auswählen, welches Verzeichnis von Stoppwörtern verwendet wird. Sie können für jede Objektgruppe ein Verzeichnis von Stoppwörtern auswählen. Ein Verzeichnis von Stoppwörtern kann von mehreren Objektgruppen gemeinsam genutzt werden.

Erstellen einer XML-Datei für Stoppwörter

Damit Sie unternehmensspezifisches Vokabular aus Abfragen entfernen können, müssen Sie in einer XML-Datei angeben, welche Wörter Sie als Stoppwörter verwenden wollen.

Informationen zu dieser Task

Die XML-Datei, die die Stoppwörter auflistet, muss dem im XML-Dokument angegebenen Schema entsprechen. Beispiel einer XML-Datei für Stoppwörter:

```
<?xml version="1.0" encoding="UTF-8"?>
<stopWords xmlns="http://www.ibm.com/of/83/stopwordbuilder/xml">
   <stopWord>OmniFind Edition</stopWord>
   <stopWord>WAS</stopWord>
   <stopWord>...</stopWord>
</stopWords>
```

Einschränkungen

Ein Stoppwort kann Leerzeichen enthalten, jedoch keine Interpunktionszeichen wie z. B. Kommata (,) oder senkrechte Striche (|), da diese die Abfragesyntax der Unternehmenssuche stören würden.

Sie müssen jedoch nicht die normalisierten Formen eines Begriffs auflisten, wie z. B. das Entfernen von Akzenten und Umlauten (die Unternehmenssuche normalisiert automatisch). Wenn Sie beispielsweise das Wort "météo" als Stoppwort hinzufügen möchten, müssen Sie nicht auch das Wort METEO aufnehmen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Liste von Stoppwörtern für die Unternehmenssuche zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool für die XML-Prüfung, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die XML-Datei heißt `stopWords.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*`/packages/uima/configuration_xsd/`.
2. Fügen Sie für jedes Wort, das als Stoppwort behandelt werden soll, ein Element `<stopWord>` hinzu.

   Achten Sie darauf, Ihre Zuordnungen in ein Element `<stopWords xmlns="http://www.ibm.com/of/83/stopwordbuilder/xml">` aufzunehmen. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Wiederholen Sie diesen Schritt, bis Sie alle Stoppwörter angegeben haben, die aus Abfragen entfernt werden sollen, wenn Benutzer die Objektgruppen für die Unternehmenssuche durchsuchen.
4. Speichen und beenden Sie die XML-Datei.

Nach dem Erstellen der XML-Datei müssen Sie sie in ein Verzeichnis von Stoppwörtern umwandeln, das Sie dem System für die Unternehmenssuche hinzufügen können.

Erstellen eines Verzeichnisses von Stoppwörtern

Nach dem Erstellen oder Aktualisieren einer Liste von benutzerdefinierten Stoppwörtern in einer XML-Datei müssen Sie die XML-Datei in ein Verzeichnis von Stoppwörtern umwandeln.

Informationen zu dieser Task

Verwenden Sie für die Erstellung eines Verzeichnisses von Stoppwörtern das Befehlszeilentool 'esstopworddictbuilder', das zusammen mit OmniFind Enterprise Edition geliefert wird. Das Tool befindet sich im Verzeichnis *ES_INSTALL_ROOT*/bin.

Als Eingabe für das Tool dient die XML-Datei, die die Stoppwörter auflistet. Die Ausgabe ist ein Verzeichnis von Stoppwörtern. Das Wörterverzeichnis muss das Suffix .dic haben. Beispiel: c:\eigeneVerzeichnisse\produktstoppwörter.dic.

Standardspeicherort für beide Dateien ist das Verzeichnis, in dem das Script aufgerufen wird. Wenn bereits ein Verzeichnis mit dem gleichen Namen vorhanden ist, generiert das Script einen Fehler.

Die maximale Größe einer DIC-Datei bei der Unternehmenssuche ist 8 MB.

Vorgehensweise

Gehen Sie wie folgt vor, um ein Verzeichnis von Stoppwörtern für die Unternehmenssuche zu erstellen:

1. Melden Sie sich am Indexserver als Administrator für die Unternehmenssuche an. Diese Benutzer-ID wurde bei der Installation von OmniFind Enterprise Edition angegeben.
2. Geben Sie den folgenden Befehl ein. Dabei ist *xml-datei* der vollständig qualifizierte Pfad zur XML-Datei mit der Liste mit Stoppwörtern und *dic-datei* der vollständig qualifizierte Pfad zum Verzeichnis von Stoppwörtern.

   AIX, Linux oder Solaris: esstopworddictbuilder.sh *xml-datei dic-datei*
   Windows: esstopworddictbuilder.bat *xml-datei dic-datei*

Fügen Sie nach dem Erstellen des Verzeichnisses von Stoppwörtern über die Administrationskonsole für die Unternehmenssuche das Verzeichnis dem System für die Unternehmenssuche hinzu, und ordnen Sie es mindestens einer Objektgruppe zu.

Nur die generierte DIC-Datei wird auf das System für die Unternehmenssuche hochgeladen. Stellen Sie sicher, dass die XML-Datei in einer Umgebung mit Zugriffssteuerung aufbewahrt wird, und sichern Sie die Datei regelmäßig. Sie brauchen diese XML-Datei, um Ihr Verzeichnis von Stoppwörtern zu aktualisieren.

Benutzerdefinierte Verzeichnisse von Boostwörtern

Sie können bestimmte Begriffe oder Mehrwortbegriffe definieren, die den Rangordnungswert des Dokuments, das einen dieser Begriffe enthält, höher oder niedriger einstufen.

Jedem Begriff des Verzeichnisses ist ein Boostfaktor in einem Bereich zwischen -10 und +10 zugeordnet. Den Begriffen, die Sie im Ergebnisdokument als besonders wichtig erachten, ordnen Sie einen höheren Boostfaktor zu, während Sie denjenigen, die gar nicht oder nur in Kombination mit höherwertigen Boostbegriffen angezeigt werden sollen, einen niedrigeren Wert zuordnen. Die Werte -1, 0 und 1 haben keine Boostwirkung.

Wird ein Abfragebegriff, der im Verzeichnis von Boostwörtern mit einem bestimmten Boostfaktor aufgelistet ist, in einem abgerufenen Dokument angezeigt, wird der Rangordnungswert des Dokuments abhängig vom Boostwert erhöht oder erniedrigt. Der einem Begriff zugeordnete Boostwert ist relativ, da er auch von anderen Faktoren beeinflusst wird. Das heißt, wenn der Begriff X den Boostwert B1 und der Begriff Y den Boostwert B2 hat, und B1 > B2 ist, ist die Boostwirkung (X) >= Boostwirkung (Y).

Zu Boostwörtern zählen in der Regel Mehrwortbegriffe, z. B. Produktnamen wie *OmniFind Enterprise Edition*. Mehrwortbegriffe, die im Verzeichnis von Boostwörtern enthalten sind, werden in Benutzerabfragen richtig identifiziert und müssen nicht in Anführungszeichen gesetzt werden.

Verzeichnisse von Boostwörtern sind sprachunabhängig.

Auch die zusammengesetzten Begriffe der germanischen Sprachen werden in Abfragen richtig identifiziert. Ein zusammengesetzter Begriff ist eine Kombination aus mindestens zwei Begriffen, die wie ein einziger Begriff verwendet werden. Lexikalisierte Zusammensetzungen wie *Reisebüro* gelten nicht als zusammengesetzte Begriffe.

In einer Abfrage werden die zusammengesetzten Begriffe in die Einzelbegriffe zerlegt, aus denen sie zusammengesetzt sind. Wenn Boostwerte aus den Einzelbegriffen eines zusammengesetzten Begriffs bestehen, werden die abgerufenen Dokumente eingestuft. Allerdings ist der zugeordnete Wert niedriger, als bei Begriffen, die alleine in einem Dokument auftreten (und nicht Teil eines zusammengesetzten Begriffs sind). Dadurch wird der Suchbereich erweitert, was immer dann sinnvoll ist, wenn nur wenige Dokumente gefunden werden, die den vollständigen zusammengesetzten Begriff enthalten.

So gibt z. B. der Abfragebegriff `Versicherungspolice` Dokumente zurück, die die zusammengesetzten Begriffe *Lebensversicherungspolice* und *Haftpflichtversicherungspolice* enthalten. Wenn das Wort *Police* im Verzeichnis von Boostwörtern vorhanden ist, wird dem Dokument, das den zusammengesetzten Abfragebegriff `Versicherungspolice` enthält, ein Boostwert zugeordnet.

Sie müssen die Begriffe mit ihrem Boostwert in einer XML-Datei auflisten und diese anschließend in ein Verzeichnis von Boostwörtern konvertieren, das Sie dem System für die Unternehmenssuche hinzufügen können.

Über die Administrationskonsole für die Unternehmenssuche können Sie auswählen, welches Verzeichnis von Boostwörtern verwendet wird. Für jede Objektgruppe können Sie ein Verzeichnis von Boostwörtern auswählen. Ein Verzeichnis von Boostwörtern kann von mehreren Objektgruppen gemeinsam genutzt werden.

Erstellen einer XML-Datei für Boostwörter

Damit Sie die Wertigkeit bestimmter Ergebnisdokumente erhöhen oder erniedrigen können, müssen Sie in einer XML-Datei angeben, welche Wörter die Rangordnung von Dokumenten beeinflussen können.

Informationen zu dieser Task

Die XML-Datei, die die Boostwörter auflistet, muss dem in der XML-Datei angegebenen Schema entsprechen. Beispiel einer XML-Datei für Boostwörter:

```
<?xml version="1.0" encoding="UTF-8"?>
<boostTerms xmlns="http://www.ibm.com/of/83/boostbuilder/xml">
    <!-- group boost terms by boost value-->
   <boostTermList boost="5">
    <!-- each term can specify the synonym expansion separately-->
     <term useVariants="true">OmniFind Edition</term>
     <term useVariants="false">Edition</term>
    <term>OmniFind</term>
   </boostTermList>
   <boostTermList boost="8">
        <term useVariants="true">WAS</term>
        <term>term9</term>
    </boostTermList>
</boostTerms>
```

Einschränkungen

Sie können Wörter, die denselben Boostwert aufweisen, in einem `<boostTermList>`-Element gruppieren; ein Boostwert kann jedoch mehrmals vorkommen, z. B. wenn Sie Boostwörter in der XML-Datei alphabetisch sortieren möchten.

Ein Boostwort kann Leerzeichen enthalten, jedoch keine Interpunktionszeichen wie z. B. Kommata (,) oder senkrechte Striche (|), da diese die Abfragesyntax der Unternehmenssuche stören würden.

Boostwörter können Varianten aufweisen, wie z. B. Akronyme oder Abkürzungen. Sie können alle Varianten im Verzeichnis von Boostwörtern aufführen. Wenn Sie jedoch beabsichtigen, auch ein Synonymverzeichnis zu verwenden, und Sie dem Synonymverzeichnis bereits Begriffe mit ihren Varianten hinzugefügt haben, ist es nicht erforderlich, diese Varianten ebenfalls dem Verzeichnis von Boostwörtern hinzuzufügen.

Sie können stattdessen für die Varianten, die Sie dem Verzeichnis von Boostwörtern hinzufügen, einfach das Attribut `useVariants` auf `true` setzen. Alle im Synonymverzeichnis aufgelisteten Varianten dieses Begriffs, die in einem der abgerufenen Dokumente enthalten sind, beeinflussen die Rangfolge, die diesen Dokumenten zugewiesen wird.

Sie müssen jedoch nicht die normalisierten Formen eines Begriffs auflisten, wie z. B. das Entfernen von Akzenten und Umlauten (die Unternehmenssuche normalisiert automatisch). Wenn Sie beispielsweise das Wort "météo" als Boostwort hinzufügen möchten, müssen Sie nicht auch das Wort METEO aufnehmen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Liste von Boostwörtern für die Unternehmenssuche zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die XML-Datei heißt `boostTerms.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*`/packages/uima/configuration_xsd/`.
2. Nehmen Sie ihre Zuordnungen in ein Element `<boostTerms xmlns="http://www.ibm.com/of/83/boostbuilder/xml">` auf. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element `<boostTermList>` hinzu, um alle Begriffe zu gruppieren, die denselben Boostwert verwenden.

   Die Boostwerte bewegen sich im Bereich zwischen -10 und 10. Beispiele: `<boostTermList boost="-5">` oder `<boostTermList boost="5">`.

   Die Wertigkeit der Dokumente, die die angegebenen Begriffe enthalten, wird anhand des angegebenen Boostwerts erhöht oder erniedrigt.
4. Fügen Sie für jeden Begriff, der den angegebenen Boostwert verwendet, ein Element `<term>` hinzu.

   Wenn Sie auch die Varianten eines Boostworts aufnehmen wollen, die in einem Synonymverzeichnis aufgelistet sind, setzen Sie das Attribut `useVariants` des Elements `<term>` auf `true`. Der Standardwert ist `false`. Es wird keine Fehlernachricht erzeugt, wenn keine Varianten im Synonymverzeichnis gefunden werden.
5. Wiederholen Sie diese Schritte, bis Sie alle Begriffe angegeben haben, die die Benutzer zum Durchsuchen der Objektgruppen für die Unternehmenssuche als Boostwörter verwenden sollen.
6. Speichen und beenden Sie die XML-Datei.

Nach dem Erstellen der XML-Datei müssen Sie sie in ein Verzeichnis von Boostwörtern umwandeln, das Sie dem System für die Unternehmenssuche hinzufügen können.

Erstellen eines Verzeichnisses von Boostwörtern

Nach dem Erstellen oder Aktualisieren einer Liste von Boostwörtern in einer XML-Datei müssen Sie die XML-Datei in ein Verzeichnis von Boostwörtern umwandeln.

Informationen zu dieser Task

Verwenden Sie für die Erstellung eines Verzeichnisses von Boostwörtern das Befehlszeilentool '`esboostworddictbuilder`', das zusammen mit OmniFind Enterprise Edition geliefert wird. Das Tool befindet sich im Verzeichnis `ES_INSTALL_ROOT/bin`.

Als Eingabe für das Tool dient die XML-Datei, die Ihre Boostwörter auflistet. Die Ausgabe ist ein Verzeichnis von Boostwörtern. Das Wörterverzeichnis muss das Suffix `.dic` haben. Beispiel: `c:\eigeneVerzeichnisse\produktboostwörter.dic`.

Standardspeicherort für beide Dateien ist das Verzeichnis, in dem das Script aufgerufen wird. Wenn bereits ein Verzeichnis mit dem gleichen Namen vorhanden ist, generiert das Script einen Fehler.

Die maximale Größe einer DIC-Datei bei der Unternehmenssuche ist 8 MB.

Vorgehensweise

Gehen Sie wie folgt vor, um ein Verzeichnis von Boostwörtern für die Unternehmenssuche zu erstellen:

1. Melden Sie sich am Indexserver als Administrator für die Unternehmenssuche an. Diese Benutzer-ID wurde bei der Installation von OmniFind Enterprise Edition angegeben.
2. Geben Sie den folgenden Befehl ein. Dabei ist *xml-datei* der vollständig qualifizierte Pfad zur XML-Datei mit der Liste mit Boostwörtern und *dic-datei* der vollständig qualifizierte Pfad zum Verzeichnis von Boostwörtern. Wenn Sie gleichzeitig ein Synonymverzeichnis verwenden wollen, fügen Sie den vollständig qualifizierten Pfad zum Synonymverzeichnis nach dem Namen des Verzeichnisses von Boostwörtern hinzu. Die Angabe von Synonymverzeichnissen ist optional.

   UNIX: `esboostworddictbuilder.sh` *`xml-datei dic-datei synverz-datei`*
   Windows: `esboostworddictbuilder.bat` *`xml-datei dic-datei synverz-datei`*

Fügen Sie nach dem Erstellen des Verzeichnisses von Boostwörtern über die Administrationskonsole für die Unternehmenssuche das Verzeichnis dem System für die Unternehmenssuche hinzu, und ordnen Sie es mindestens einer Objektgruppe zu.

Nur die generierte DIC-Datei wird auf das System für die Unternehmenssuche hochgeladen. Stellen Sie sicher, dass die XML-Datei in einer Umgebung mit Zugriffssteuerung aufbewahrt wird und eine geeignete Backup-Strategie aktiviert ist. Sie brauchen diese XML-Datei, um Ihr Verzeichnis von Boostwörtern zu aktualisieren.

Zugehörige Tasks

„Erstellen eines Synonymverzeichnisses" auf Seite 69
Nach dem Erstellen oder Aktualisieren einer Synonymliste in einer XML-Datei müssen Sie die XML-Datei in ein binäres Synonymverzeichnis umwandeln.

Textanalyse innerhalb der Unternehmenssuche

Zu der in der Unternehmenssuche enthaltenen Textanalyse gehört die Erkennung der Sprache und die Segmentierung des Dokuments.

Bei der Verarbeitung eines Dokuments ermittelt die Unternehmenssuche die Sprache dieses Dokuments und unterteilt den Eingabedatenstrom in eindeutige Einheiten oder Token.

Während einer Suche muss der Benutzer oder eine Anwendung die Abfragesprache manuell auswählen. Die Abfragezeichenfolge wird segmentiert, analysiert und im Index gesucht.

Sowohl die Analyse des Dokuments als auch die der Abfragezeichenfolge lassen sich wie folgt unterteilen:

- Basisunterstützung, die nicht auf einem Wörterverzeichnis basiert. Hierzu gehören Leerraumsegmentierung und N-Gram-Segmentierung. Zur Basisunterstützung, die nicht auf einem Wörterverzeichnis basiert, gehört auch die Satzsegmentierung.
- Linguistische Unterstützung, die auf einem Wörterverzeichnis basiert. Hierzu gehören die Wort- und Satzsegmentierung und die Reduktion auf die Grundform.

  Zur Verarbeitung auf linguistischer Basis gehört die lexikalische Analyse, ein Prozess, bei dem alternative Darstellungen des Eingabetexts erstellt und den im Eingabetext erkannten Token alle verfügbaren Verzeichnisdaten zugeordnet werden. Durch die Verwendung der erweiterten Sprachenverarbeitung wurde die Suchqualität sehr verbessert.

  Zugehörige Konzepte

  „Spracherkennung"
  Bevor eine Wort- und Satzsegmentierung, eine Zeichennormalisierung oder eine Reduktion auf die Grundform ausgeführt werden kann, muss die Unternehmenssuche die Sprache des Quellendokuments ermitteln.

  „Linguistische Unterstützung der nicht wörterverzeichnisbasierten Segmentierung" auf Seite 81
  Für Dokumente, die in Sprachen abgefasst sind, die nicht von der lexikalischen Analysetechnologie unterstützt werden, stellt die Unternehmenssuche eine Basisunterstützung in der Form von Unicode-basierter Segmentierung mit Leerzeichen und N-Gram-Segmentierung zur Verfügung.

Spracherkennung

Bevor eine Wort- und Satzsegmentierung, eine Zeichennormalisierung oder eine Reduktion auf die Grundform ausgeführt werden kann, muss die Unternehmenssuche die Sprache des Quellendokuments ermitteln.

Die Unternehmenssuche erkennt die folgenden Sprachen automatisch:

Tabelle 10. Unterstützte Sprachen der automatischen Spracherkennung

Afrikaans	Arabisch	Balinesisch
Baskisch	Katalanisch	Chinesisch (traditionell und vereinfacht)

Tabelle 10. Unterstützte Sprachen der automatischen Spracherkennung (Forts.)

Tschechisch	Dänisch	Niederländisch
Englisch	Finnisch	Französisch
Deutsch	Griechisch	Hebräisch
Isländisch	Irisch (Gälisch)	Italienisch
Japanisch	Koreanisch	Malaiisch
Norwegisch (Bokmål)	Polnisch	Portugiesisch
Rumänisch	Russisch	Spanisch
Schwedisch	Tagalog	Thailändisch
Türkisch	Vietnamesisch	

Die linguistischen Prozesse in der Unternehmenssuche ermitteln die Sprache eines Quellendokuments während der Indexierung, nicht während der Abfrageverarbeitung.

In der Unternehmenssuche können Sie die automatische Spracherkennung für ein Dokument angeben, Sie können aber auch eine Sprache auswählen, die verwendet werden soll.

Wenn Sie die automatische Spracherkennung ausgewählt haben, der Parser die Sprache des Dokuments jedoch nicht ermitteln kann, verwendet der Parser die Sprache, die Sie beim Erstellen des Crawlers in der Administrationskonsole für die Unternehmenssuche angeben.

Wenn Sie keine automatische Spracherkennung auswählen, wird immer die von Ihnen angegebene Sprache verwendet. Geben Sie die Dokumentsprache an, indem Sie die Crawlermerkmale über die Administrationskonsole für die Unternehmenssuche bearbeiten. Die voreingestellte Sprache ist Englisch.

Dokumente, für die es keine sprachspezifischen Wörterverzeichnisse gibt, werden unter Verwendung einer sprachunabhängigen Basistechnologie verarbeitet, wie z. B. Leerraumsegmentierung und N-Gram-Segmentierung.

Die Spracherkennung der Unternehmenssuche ist für einsprachige Dokumente bestens geeignet. Bei mehrsprachigen Dokumenten wird versucht zu ermitteln, welche Sprache im Dokument am häufigsten verwendet wird. Die Analyseergebnisse sind jedoch nicht immer zufriedenstellend.

Anhand der Sprache eines Dokuments können Sie Ihre Suchergebnisse so beschränken, dass nur Dokumente angezeigt werden, die in einer bestimmten Sprache abgefasst sind. Wenn Sie z. B. in einer Objektgruppe mit mehrsprachigen Dokumenten nach Dokumenten suchen, die über Jacques Chirac geschrieben wurden, können Sie angeben, dass nur auf Französisch verfasste Dokumente in die Suchergebnisse aufgenommen werden sollen. Das Einstellen der Sprache für die Ausgabedokumente ist eine erweiterte Suchoption, die Sie über die Administrationskonsole für die Unternehmenssuche auswählen können.

Zugehörige Konzepte

„Textanalyse innerhalb der Unternehmenssuche" auf Seite 79
Zu der in der Unternehmenssuche enthaltenen Textanalyse gehört die Erkennung der Sprache und die Segmentierung des Dokuments.

„Linguistische Unterstützung der nicht wörterverzeichnisbasierten Segmentierung"
Für Dokumente, die in Sprachen abgefasst sind, die nicht von der lexikalischen Analysetechnologie unterstützt werden, stellt die Unternehmenssuche eine Basisunterstützung in der Form von Unicode-basierter Segmentierung mit Leerzeichen und N-Gram-Segmentierung zur Verfügung.

Linguistische Unterstützung der nicht wörterverzeichnisbasierten Segmentierung

Für Dokumente, die in Sprachen abgefasst sind, die nicht von der lexikalischen Analysetechnologie unterstützt werden, stellt die Unternehmenssuche eine Basisunterstützung in der Form von Unicode-basierter Segmentierung mit Leerzeichen und N-Gram-Segmentierung zur Verfügung.

Unicode-basierte Segmentierung mit Leerzeichen
Dieses Verarbeitungsverfahren auf linguistischer Basis verwendet den Leerraum (oder die Leerzeichen) zwischen Wörtern als Wortbegrenzer.

N-Gram-Segmentierung
Dieses Verarbeitungsverfahren auf linguistischer Basis behandelt überlappende Sequenzen von *n* Zeichen als ein Wort. Dieses einfache Segmentierungsverfahren ist für viele Abruftasks ausreichend.

Diese Verfahren sind unabhängig von Wörterverzeichnissen in bestimmten Sprachen. Sie enthalten keine hoch entwickelte Technologie für die Verarbeitung auf linguistischer Basis, wie z. B. die Reduktion auf die Grundform.

Die N-Gram-Segmentierung wird für Sprachen wie Thailändisch verwendet, in denen es keine Leerstellen gibt, die als Begrenzer verwendet werden können. Dasselbe Verfahren wird für Hebräisch und Arabisch angewendet. Obwohl in diesen zwei Sprachen Leerzeichen als Begrenzer vorhanden sind, gibt die N-Gram-Segmentierung bessere Ergebnisse zurück als die Basisform der Unicode-basierten Segmentierung mit Leerzeichen.

Beim Erstellen Ihrer Objektgruppe können Sie auch optional auswählen, ob Sie chinesische und japanische Dokumente unter Verwendung der N-Gram-Segmentierung aufbereiten wollen.

Sie können alle Leerraumzeichen, wie z. B. Zeichen für Zeilenvorschub oder Tabulatorzeichen, bei einer N-Gram-Segmentierung entfernen, indem Sie die entsprechenden Parametereinstellungen in der Datei `collection.properties` in *ES_NODE_ROOT*`/master_config/<objektgruppen-id>.parserdriver` aktivieren, bevor Sie mit der Syntaxanalyse der Dokumente beginnen.

Unter anderem sind folgende Parameter erforderlich, um Leerzeichen zu entfernen:

- **removeCjNewLineChars**: Ist hier *true* angegeben, entfernt der Parameter alle Zeilenvorschübe und Tabulatorzeichen zwischen chinesischen oder japanischen Zeichen. Die Standardeinstellung ist `removeCjNewlineChars=false`.
- **removeCjNewLineCharsMode**: Ist hier *all* angegeben, entfernt der Parameter kontextunabhängig alle Leerzeichen. So werden z. B. auch aus einem englischen Text die Leerzeichen entfernt. Wenn Sie diese Option verwenden wollen, müssen Sie der Merkmaldatei den Parameter hinzufügen.

  Nur `removeCjNewlineCharsMode=all` ist gültig, alle anderen Werte werden ignoriert.

Zugehörige Konzepte

„Textanalyse innerhalb der Unternehmenssuche" auf Seite 79
Zu der in der Unternehmenssuche enthaltenen Textanalyse gehört die Erkennung der Sprache und die Segmentierung des Dokuments.

„Spracherkennung" auf Seite 79
Bevor eine Wort- und Satzsegmentierung, eine Zeichennormalisierung oder eine Reduktion auf die Grundform ausgeführt werden kann, muss die Unternehmenssuche die Sprache des Quellendokuments ermitteln.

Aufbereiten von numerischen Zeichen als N-Gram-Token

Wenn Sie numerischen Zeichen zusätzlich zu den Doppelbytezeichen als N-Gram-Token aufbereiten wollen, müssen Sie eine Parametereinstellung in der Annotatordeskriptordatei ändern.

Informationen zu dieser Task

Bei der Standardhandhabung von numerischen Zeichen im Leerraum- und N-Gram-Tokenizer werden alle numerischen Zeichen wie durch Leerraum segmentierte Token behandelt. Wenn Sie numerische Zeichen als N-Gram-Token aufbereiten wollen, müssen Sie die N-Gram-Moduseinstellung in der Deskriptordatei des Annotators ändern. Sie können diese Einstellung nicht über die Administrationskonsole für die Unternehmenssuche ändern.

Tipp: Es sind drei Modi für die N-Gram-Tokenisierung verfügbar: `normal`, `numeric` und `full`. In dieser Prozedur wird die Aktivierung der numerischen N-Gram-Tokenisierung beschrieben. Informationen zur vollständigen N-Gram-Tokenisierung in Objektgruppen für die Unternehmenssuche und zur Verarbeitung von Zeichen in Objektgruppen, die für die vollständige N-Gram-Unterstützung konfiguriert sind, finden Sie unter http://www.ibm.com/support/docview.wss?rs=63&uid=swg27011088.

Vorgehensweise

Die Standardeinstellung des N-Gram-Modus heißt `normal` und behandelt numerische Zeichen und Einzelbytezeichensatz-Zeichen wie durch Leerraum segmentierte Zeichen. Gehen Sie wie folgt vor, um den numerischen N-Gram-Modus (`numeric`) zu aktivieren:

1. Stoppen Sie den Parser für Ihre Objektgruppe.
2. Stoppen Sie die Laufzeitumgebung für Ihre Objektgruppe.
3. Öffnen Sie die Deskriptordatei des Annotators mit dem Namen `jtok.xml` im Verzeichnis *ES_NODE_ROOT*`/master_config/`*objektgruppen-id*`.parserdriver/specifiers`. Dabei gilt ist *objektgruppen-id* die ID, die bei der Erstellung der Objektgruppe für die Objektgruppe angegeben (oder vom System zugeordnet) wurde.
4. Ändern Sie den Parameter **NgramMode** von **normal** in **numeric**.
5. Starten Sie den Parser für Ihre Objektgruppe erneut.
6. Starten Sie die Laufzeitumgebung erneut.

Linguistische Unterstützung der Segmentierung auf der Basis von Wörterverzeichnissen

Wenn die Sprache eines Dokuments richtig erkannt wurde und sprachspezifische Wörterverzeichnisse verfügbar sind, wird die entsprechende Verarbeitung auf linguistischer Basis angewendet.

Die Segmentierung ist der Prozess, bei dem der Eingabetext in eindeutige lexikalische Texteinheiten unterteilt wird. Dazu gehören einige der folgenden Verarbeitungsmethoden auf linguistischer Basis:

Wortsegmentierung
Die Wortsegmentierung wird für Sprachen verwendet, die zwischen Wörtern keinen Leerraum (oder keine Begrenzer) verwenden, wie z. B. Japanisch und Chinesisch.

Reduktion auf die Grundform
Die Reduktion auf die Grundform ist eine Verarbeitungsform auf linguistischer Basis, bei der für jedes Wort, das im Text vorkommt, das Lemma ermittelt wird. Das *Lemma* eines Worts umfasst dessen Grundform sowie flektierte Formen derselben Wortart. So umfasst z. B. das Lemma von *gehen* die Formen *gehen*, *geht*, *ging*, *gegangen* und *gehend*. Lemmata von Substantiven gruppieren die Formen von Singular und Plural (wie *Kalb* und *Kälber*). Lemmata von Adjektiven gruppieren die Formen von Komparativ und Superlativ (wie *gut*, *besser* und *am besten*). Lemmata von Pronomen gruppieren den Kasus eines Pronomens (wie *ich*, *mich*, *mein* und *mir*).

Für die Reduktion auf die Grundform ist sowohl zum Indexieren als auch zum Suchen ein Wörterverzeichnis erforderlich.

Die Unternehmenssuche indexiert die Lemmata und flektierten Wörter und reduziert alle flektierten Wörter in einer Abfrage auf ihre Grundform. Durch die Reduktion auf die Grundform wird die Suchqualität verbessert, indem nach Dokumenten gesucht wird, die Varianten eines in der Abfrage enthaltenen flektierten Worts aufweisen. So wird z. B. auch nach Dokumenten gesucht, in denen das Wort *Mäuse* vorkommt, wenn die Abfrage das Wort *Maus* enthält.

Kontraktionssplitting
Die Suchqualität wird verbessert, indem Kontraktionen erkannt und in ihre einzelnen Komponenten gesplittet werden. Beispiel:

wouldn't wird gesplittet in
would + *not*
Horse's wird gesplittet in *Horse* + *'s*

Klitikerkennung
Klitika sind eine besondere Form der Kontraktion, und die Suchqualität wird verbessert, wenn ihre einzelnen Komponenten ermittelt werden. Ein *Klitik* ist ein Element, das sich wie ein Affix und ein Wort verhält. Klitika sind jedoch schwer zu ermitteln, da sie auch Teil der Wortbildung sind. Anders als andere morphologische Phänomene (Wortstrukturen) treten Klitika in einer syntaktischen Struktur auf, und ihr Anschluss an das Wort ist nicht Teil der Wortbildungsregeln. Beispiel:

reparti-lo-emos hat die Komponenten *repartir* + *lo* + *emos*
l'avenue hat die Komponenten *le* + *avenue*
dell'arte hat die Komponenten *dello* + *arte*.

Erkennung nicht alphabetischer Zeichen

Die Verarbeitungsprozesse auf linguistischer Basis erkennen nicht alphabetische Zeichen. Abhängig von der internen sprachabhängigen Logik, werden einige nicht alphabetische Zeichen als eigene lexikalische Einheiten unterschiedlichen Typs zurückgegeben, andere werden in Gruppen zusammengefasst.

So werden bei Klitika z. B. Hochkommata als Teil des Worts betrachtet, und bei unbekannten Abkürzungen werden sie wie ein Punkt behandelt. URL-Adressen, E-Mail-Adressen und Datumsangaben werden in mehrere Token aufgeteilt.

Erkennung von Abkürzungen

Die Verarbeitung auf linguistischer Basis erkennt Abkürzungen, die im Wörterverzeichnis als eine lexikalische Einheit angegeben sind. Ist eine Abkürzung nicht im Wörterverzeichnis vorhanden, wird sie zwar als lexikalisches Element erkannt, es werden ihr jedoch keine Informationen aus dem Wörterverzeichnis zugeordnet.

Das korrekte Erkennen von Abkürzungen ist wichtig für die Satzerkennung. So ist z. B. der Punkt hinter einer Abkürzung nicht unbedingt das Ende eines Satzes.

Erkennung der Markierung des Satzendes

Für die Satzsegmentierung können die Verarbeitungsprozesse auf linguistischer Basis die Markierungen des Satzendes richtig erkennen.

Linguistische Unterstützung auf Basis von Wörterverzeichnissen ist für die folgenden Sprachen verfügbar:

Tabelle 11. Unterstützte Sprachen

Arabisch	Italienisch
Chinesisch (traditionell und vereinfacht)	Japanisch
Tschechisch	Koreanisch
Dänisch	Norwegisch (Bokmål)
Niederländisch	Polnisch
Englisch	Portugiesisch (Portugal und Brasilien)
Finnisch	Russisch
Französisch (Frankreich und Kanada)	Spanisch
Deutsch (Deutschland und Schweiz)	Schwedisch
Griechisch	

Zugehörige Konzepte

„Wortsegmentierung im Japanischen" auf Seite 85
Wird ein Textdokument oder eine Abfragezeichenfolge als Japanisch erkannt, führt die Unternehmenssuche die relevante Wortsegmentierung aus, indem sie die für die japanische Sprache optimierte morphologische Analysetechnologie verwendet.

„Orthografische Varianten im Japanischen" auf Seite 85
Im Japanischen werden viele orthografischen Varianten verwendet. Die Katakana-Varianten sind am wichtigsten, da Katakana häufig verwendet wird, um fremdsprachige Wörter zu buchstabieren und auszusprechen. Viele der Katakana-Varianten werden häufig im Japanischen verwendet.

Wortsegmentierung im Japanischen

Wird ein Textdokument oder eine Abfragezeichenfolge als Japanisch erkannt, führt die Unternehmenssuche die relevante Wortsegmentierung aus, indem sie die für die japanische Sprache optimierte morphologische Analysetechnologie verwendet.

Ein Beispiel für diese Optimierung ist die Zerlegung von Wörtern. Im Japanischen werden viele zusammengesetzte Begriffe verwendet. Diese Begriffe werden in Token von einer optimalen Größe zerlegt, sodass bessere Suchergebnisse erzielt werden. Flektierte Wörter und Präpositionen werden ebenfalls zerlegt, um die Suchleistung zu verbessern.

Zugehörige Konzepte

„Linguistische Unterstützung der Segmentierung auf der Basis von Wörterverzeichnissen" auf Seite 83
Wenn die Sprache eines Dokuments richtig erkannt wurde und sprachspezifische Wörterverzeichnisse verfügbar sind, wird die entsprechende Verarbeitung auf linguistischer Basis angewendet.

„Orthografische Varianten im Japanischen"
Im Japanischen werden viele orthografischen Varianten verwendet. Die Katakana-Varianten sind am wichtigsten, da Katakana häufig verwendet wird, um fremdsprachige Wörter zu buchstabieren und auszusprechen. Viele der Katakana-Varianten werden häufig im Japanischen verwendet.

Orthografische Varianten im Japanischen

Im Japanischen werden viele orthografischen Varianten verwendet. Die Katakana-Varianten sind am wichtigsten, da Katakana häufig verwendet wird, um fremdsprachige Wörter zu buchstabieren und auszusprechen. Viele der Katakana-Varianten werden häufig im Japanischen verwendet.

Die Unternehmenssuche verwendet ein Wörterverzeichnis von Varianten, um die typischen Katakana-Varianten ihren Basisformen (ähnlich wie ein Lemma) zuzuordnen, sodass alle Dokumente, einschließlich derer mit orthografischen Varianten des Katakana-Worts in der Abfragezeichenfolge, gefunden werden.

Die Unternehmenssuche unterstützt außerdem auch die typischen Okurigana-Varianten, bei denen es sich um Kanji-Wortendungen handelt, die in Hiragana geschrieben werden.

Zugehörige Konzepte

„Linguistische Unterstützung der Segmentierung auf der Basis von Wörterverzeichnissen" auf Seite 83
Wenn die Sprache eines Dokuments richtig erkannt wurde und sprachspezifische Wörterverzeichnisse verfügbar sind, wird die entsprechende Verarbeitung auf linguistischer Basis angewendet.

„Wortsegmentierung im Japanischen"
Wird ein Textdokument oder eine Abfragezeichenfolge als Japanisch erkannt, führt die Unternehmenssuche die relevante Wortsegmentierung aus, indem sie die für die japanische Sprache optimierte morphologische Analysetechnologie verwendet.

Stoppwortentfernung

In der Unternehmenssuche werden alle Stoppwörter, z. B. häufig verwendete Wörter wie *ein* und *der*, aus Mehrwortabfragen entfernt, um die Suchleistung zu optimieren.

Die Stoppworterkennung im Japanischen basiert auf grammatikalischen Informationen. So erkennt die Unternehmenssuche z. B., ob es sich bei einem Wort um ein Substantiv oder ein Verb handelt. Für die anderen Sprachen verwendet die Unternehmenssuche Speziallisten.

Während der Abfrageverarbeitung werden in den folgenden Fällen keine Stoppwörter entfernt:

- Alle Wörter einer Abfrage sind Stoppwörter. Wenn während der Verarbeitung von Stoppwörtern alle Abfragebegriffe entfernt werden, ist die Ergebnisliste leer. Wenn alle Abfragebegriffe Stoppwörter sind, wird die Stoppwortentfernung inaktiviert, um zu gewährleisten, dass Suchergebnisse zurückgegeben werden. Wenn z. B. das Wort *Auto* ein Stoppwort ist und Sie nach *Auto* suchen, enthalten die Suchergebnisse Dokumente, die das Wort *Auto* enthalten. Wenn Sie nach *Auto Buick* suchen, enthalten die Suchergebnisse jedoch nur die Dokumente, die das Wort *Buick* enthalten.
- In der Abfrage ist dem Wort ein Pluszeichen (+) vorangestellt.
- Das Wort ist Teil einer exakten Übereinstimmung.
- Das Wort befindet sich innerhalb einer Wortfolge, z. B. in "Ich mag mein Auto".

Zugehörige Konzepte

„Zeichennormalisierung"
Die Zeichennormalisierung ist ein Prozess, der das Abrufen verbessern kann. Abrufverbesserung durch Zeichennormalisierung bedeutet, dass mehr Dokument abgerufen werden, selbst wenn die Dokument nicht genau mit der Abfrage übereinstimmen.

Zeichennormalisierung

Die Zeichennormalisierung ist ein Prozess, der das Abrufen verbessern kann. Abrufverbesserung durch Zeichennormalisierung bedeutet, dass mehr Dokument abgerufen werden, selbst wenn die Dokument nicht genau mit der Abfrage übereinstimmen.

Die Unternehmenssuche verwendet eine Unicode-kompatible Normalisierung, zu der auch die Normalisierung asiatischer Zeichen mit halber Breite auf volle Breite gehört.

Die Unternehmenssuche entfernt auch die mittleren Punkte aus Katakana-Schriftzeichen, die im Japanischen als Begrenzer von zusammengesetzten Begriffen verwendet werden.

Die anderen Formen der Zeichennormalisierung umfassen Folgendes:

Normalisierung von Groß-/Kleinschreibung
Damit z. B. Dokumente mit *USA* angezeigt werden, wenn Sie nach *usa* suchen.

Umlautauflösung
Damit z. B. Dokumente mit *schoen* angezeigt werden, wenn Sie nach *schön* suchen.

Entfernen von Akzenten
Damit z. B. Dokumente mit *é* angezeigt werden, wenn Sie nach *e* suchen.

Entfernen anderer diakritischer Zeichen
Damit z. B. Dokumente mit *ç* angezeigt werden, wenn Sie nach *c* suchen.

Ligaturauflösung

Damit z. B. Dokumente mit *Æ* angezeigt werden, wenn Sie nach *ae* suchen.

Alle Normalisierungen funktionieren in beide Richtungen. Sie finden auch Dokumente mit *usa*, wenn Sie nach *USA* suchen, und Dokumente, die Wörter mit *e* enthalten, wenn Sie nach *é* suchen und so weiter. Sie können die Normalisierungen auch kombinieren. Sie finden z. B. auch Dokumente, die *météo* enthalten, wenn Sie nach *METEO* suchen.

Die Normalisierungen basieren auf den Unicode-Zeichenmerkmalen und sind nicht sprachenabhängig. So unterstützt die Unternehmenssuche z. B. das Entfernen diakritischer Zeichen aus dem Hebräischen und die Ligaturauflösung im Arabischen.

Zugehörige Konzepte

„Stoppwortentfernung" auf Seite 85
In der Unternehmenssuche werden alle Stoppwörter, z. B. häufig verwendete Wörter wie *ein* und *der*, aus Mehrwortabfragen entfernt, um die Suchleistung zu optimieren.

Annotator für reguläre Ausdrücke

Mit dem Annotator für reguläre Ausdrücke können Sie eine benutzerdefinierte Textanalyse ausführen, ohne Ihre eigene Textanalysesteuerkomponente implementieren zu müssen. Auf der Basis eines Regelsatzes (reguläre Ausdrücke), den Sie selbst definieren können, erkennt der Annotator für reguläre Ausdrücke Informationsstrukturen in Textdokumenten und erstellt Annotationen zu den erkannten Informationen in der allgemeinen Analysestruktur.

Der Annotator für reguläre Ausdrücke erkennt auf der Basis von regulären Ausdrücken Entitäten oder Informationseinheiten in Textdokumenten, z. B. Telefonnummern, Produktschlüssel, Gebäude- und Zimmernummern oder Adressen. Wenn ein regulärer Ausdruck mit einem Teil des Dokumenttexts übereinstimmt, erstellt der Annotator für reguläre Ausdrücke die entsprechenden Annotationen zu den abgeglichenen Einzelinformationen. Diese Annotationen werden in der allgemeinen Analysestruktur gespeichert und können später gesucht werden, indem die Analyseergebnisse dem Index für die Unternehmenssuche zugeordnet werden. Hierzu wird eine Datei für die Zuordnung der allgemeinen Analysestruktur zum Index verwendet. Alternativ kann eine Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank erstellt werden, um die Annotationen in einer JDBC-fähigen Datenbank zu speichern.

Der von Ihnen definierte Regelsatz (reguläre Ausdrücke) werden in einer XML-Konfigurationsdatei (auch als Regelsatzdatei bezeichnet) gespeichert. Der Annotator für reguläre Ausdrücke enthält die Analyselogik, mit der die regulären Ausdrücke verarbeitet werden. Er unterstützt die Syntax für reguläre Ausdrücke in Java 1.4.

Das Typsystem des Annotators für reguläre Ausdrücke muss die Annotationstypen und -komponenten definieren, die vom Annotator für reguläre Ausdrücke verwendet und erstellt werden. Abhängig von der Komplexität des Anwendungsbereichs des Annotators für reguläre Ausdrücke (wenn z. B. mehr Typen im bereitgestellten Annotator für reguläre Ausdrücke erforderlich sind, als definiert sind), muss eine zusätzliche Eingabe- und Ausgabefunktionalität im Deskriptor des Annotators für reguläre Ausdrücke definiert werden. Die im Deskriptor verwendeten Typen müssen mit den Typen in der Typsystembeschreibung des Annotators übereinstimmen.

Der Annotator für reguläre Ausdrücke wird in die Unternehmenssuche als implementierbare PEAR-Datei (Processing Engine Archive - Verarbeitungsenginearchiv) eingeschlossen, die mit Beispielregeln für das Aufspüren von Telefonnummern, URL-Adressen und E-Mail-Adressen konfiguriert ist.

Zugehörige Konzepte

„Die Regelsatzdatei" auf Seite 93
Im Annotator für reguläre Ausdrücke definiert die XML-Regelsatzdatei in der Form von regulären Ausdrücken die Regeln, die verwendet werden, um das Textdokument syntaktisch zu analysieren.

Zugehörige Tasks

„Definieren von Regeln für reguläre Ausdrücke" auf Seite 94
Der Regelsatz definiert die regulären Ausdrücke, die mit dem Dokumenttext abgeglichen werden, und die Aktionen, die der Annotator für reguläre Ausdrücke ausführen muss, wenn ein Muster erkannt wird.

Zugehörige Verweise

„Der Annotatordeskriptor" auf Seite 99
Der XML-Annotatordeskriptor für reguläre Ausdrücke enthält beschreibende Informationen zum Annotator für reguläre Ausdrücke, die für die Ausführung des Annotators erforderlich sind.

„Protokollierung" auf Seite 102
Alle Protokollnachrichten des Annotators für reguläre Ausdrücke werden in die Protokolldatei der aktuellen Objektgruppe geschrieben.

Einfache semantische Suche mithilfe des Annotators für reguläre Ausdrücke

Zur Unternehmenssuche gehört die Analysesteuerkomponente für reguläre Ausdrücke, die bereits mit einem Regelsatz vorkonfiguriert ist, der das Erkennen von Telefonnummern, URL-Adressen und E-Mail-Adressen in Textdokumenten ermöglicht.

Sie können diese Beispielkonfiguration der Analysesteuerkomponente für reguläre Ausdrücke verwenden, um es der Unternehmenssuche zu ermöglichen, Telefonnummern in Dokumenten zu finden, ohne in diesen Dokumenten nach dem Schlüsselwort *Telefonnummer* zu suchen. Damit Sie die Konstrukte abfragen können, die vom Annotator für reguläre Ausdrücke erkannt werden, wird auch eine Beispieldatei für die Zuordnung der allgemeinen Analysestruktur zum Index bereitgestellt. Weiterhin wird eine einfache Methode veranschaulicht, mit der Sie über einfache Schlüsselwörter leistungsfähige semantische Abfragen absetzen können. Diese Methode verwendet die Synonymunterstützung der Unternehmenssuche, um einfache Schlüsselwortabfragen automatisch in semantische Abfragen zu erweitern. Zur Illustration dieses Mechanismus wird ein Beispielsynonymverzeichnis bereitgestellt. Alle Dateien, die Sie zur Verwendung des Annotators für reguläre Ausdrücke mit der Beispielkonfiguration benötigen, finden Sie im Verzeichnis `ES_INSTALL_ROOT/packages/uima/regex`.

Für viele Anwendungsszenarien reicht es unter Umständen aus, lediglich die mit der Beispielkonfiguration bereitgestellten Regeln für reguläre Ausdrücke leicht zu verändern, um den Annotator für reguläre Ausdrücke an Ihre Bedürfnisse anzupassen.

Wenn Sie den Annotator jedoch vollständig anpassen wollen, empfehlen wir Ihnen, UIMA SDK zu verwenden. Dazu ist der Annotator für reguläre Ausdrücke auch im Basisannotatorpaket für die Unternehmenssuche unter `ES_INSTALL_ROOT/packages/uima/` enthalten.

Zugehörige Tasks

„Aktivieren der einfachen semantischen Suche mithilfe des Annotators für reguläre Ausdrücke" auf Seite 91
Wenn Sie die einfache semantische Suche unter Verwendung von Synonymen aktivieren wollen, müssen Sie Ihrem System für die Unternehmenssuche den Annotator für reguläre Ausdrücke, die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index und das Beispielsynonymverzeichnis hinzufügen und Ihrer Objektgruppe diese Ressourcen zuordnen.

„Anpassen des Annotators für reguläre Ausdrücke" auf Seite 97
Sie können die Beispielkonfiguration des Annotators für reguläre Ausdrücke so anpassen, dass sie neue Entitäten erkennt (z. B. Seriennummern von Produkten), oder die Regeln für reguläre Ausdrücke an vorhandene Entitäten anpassen (z. B. zur Erkennung unternehmensspezifischer Telefonnummern), indem Sie an Dateien für Beispielregelsätze und Typsyteme kleinere Änderungen vornehmen.

„Anzeigen der Ergebnisse des Basisannotators und der angepassten Textanalyse" auf Seite 14
Wenn Sie die Analyseergebnisse anzeigen wollen, die nach der Syntaxanalyse von einem beliebigen Annotator in der Unternehmenssuche erzeugt wurden, müssen Sie die Merkmale der Dokumentobjektgruppe aktualisieren, damit eine lesbare XML-Version der in der allgemeinen Analysestruktur gespeicherten Analyseergebnisse erzeugt wird.

Aktivieren der einfachen semantischen Suche mithilfe des Annotators für reguläre Ausdrücke

Wenn Sie die einfache semantische Suche unter Verwendung von Synonymen aktivieren wollen, müssen Sie Ihrem System für die Unternehmenssuche den Annotator für reguläre Ausdrücke, die Datei für die Zuordnung der allgemeinen Analysestruktur zum Index und das Beispielsynonymverzeichnis hinzufügen und Ihrer Objektgruppe diese Ressourcen zuordnen.

Danach verarbeitet der Annotator für reguläre Ausdrücke Ihre Dokumente während der Parsing-Phase, die Indexierungskomponente fügt dem Index die Ergebnisse der benutzerdefinierten Analyse hinzu, und der Suchservice kann das bereitgestellte semantische Synonymverzeichnis verwenden, um über einfache Schlüsselwörter, die automatisch in semantische Abfragen erweitert werden, nach den Ergebnissen der benutzerdefinierten Analyse zu suchen.

Vorgehensweise

Gehen Sie wie folgt vor, um die einfache semantische Suche zu aktivieren:

1. Fügen Sie dem System für die Unternehmenssuche mithilfe der Administrationskonsole für die Unternehmenssuche die benutzerdefinierte Steuerkomponente für die Textanalyse mit regulären Ausdrücken hinzu, die `of_regex.pear` heißt und sich im Verzeichnis *ES_INSTALL_ROOT*`/packages/uima/regex` befindet.
2. Ordnen Sie Ihrer Objektgruppe die Textanalysesteuerkomponente für reguläre Ausdrücke zu.
3. Fügen Sie der Indexzuordnungsdatei `of_sample_regex_cas2index.xml` im Verzeichnis *ES_INSTALL_ROOT*`/packages/uima/regex` die allgemeine Analysestruktur hinzu. Damit werden die benutzerdefinierten Analyseergebnisse (Annotationen), die vom Annotator für reguläre Ausdrücke erzeugt werden, durchsuchbaren Bereichen im Index für die Unternehmenssuche zugeordnet. Anschließend können Sie XML-Fragment- oder XPath-Abfragen verwenden, um nach diesen Bereichen zu suchen.
4. Führen Sie für Ihre Objektgruppe eine Crawlersuche, eine syntaktische Analyse und eine Indexierung aus. Nun können Sie nach Abschluss des Indexierens einen XML-Suchbegriff eingeben und dabei mithilfe der Suchanwendung einen XML-Fragmentabfrageausdruck wie z. B. `@xmlf2::'<#phonenumber>'` verwenden. Der Zweck der Aktivierung der semantischen Suche besteht jedoch darin, die Verwendung von Abfragen wie `Barbara phone number` zu ermöglichen und diese vom System automatisch in `Barbara @xmlf2::'<#phonenumber>'` umsetzen zu lassen.
5. Fügen Sie dem System für Unternehmenssuche mithilfe der Administrationskonsole das bereitgestellte binäre Beispielverzeichnis für Synonyme hinzu, das `of_sample_synonym_dic.dic` heißt und sich im Verzeichnis *ES_INSTALL_ROOT*`/packages/uima/regex` befindet. Sie können Änderungen an der Quelle des XML-Beispielwörterverzeichnisses vornehmen oder es als Basis verwenden, um

Ihr eigenes Wörterverzeichnisses zu erstellen und dieses anschließend mit dem Tool `essyndictbuilder` in eine neue Wörterverzeichnisdatei zu konvertieren. Das XML-Beispielsynonymverzeichnis heißt `of_sample_synonym_dic.xml` und befindet sich ebenfalls im Verzeichnis *ES_INSTALL_ROOT*`/packages/uima/regex`.

6. Ordnen Sie Ihrer Objektgruppe das Synonymverzeichnis zu, und starten Sie den Suchservice für Ihre Objektgruppe bzw. starten Sie ihn erneut.
7. Wählen Sie in der Suchanwendung die Option für die automatische Suche nach Synonymen mit der semantischen Erweiterung aus. Nachdem Sie diese Option aktiviert haben, erstellt die Suchanwendung Ihre Basis-Schlüsselwortabfragen erneut als XML-Fragmentabfragen und schließt Ausdrücke zum Finden von durchsuchbaren Bereichen ein, die Telefonnummern, E-Mail-Adressen und URL-Adressen angeben.
8. Geben Sie in der Suchanwendung eine Abfrage einer Telefonnummer ein, z. B. `barbara telephon number`. Die Abfrage sucht nach Dokumenten, die die drei Schlüsselwörter *barbara*, *telephone* und *number* enthalten, wie auch nach Dokumenten, die das Schlüsselwort *barbara* und Bereiche mit Nummern und Zeichen im Dokument enthalten, die mit den regulären Ausdrücken übereinstimmen, die für eine Telefonnummer definiert wurden. Die gefundenen Schlüsselwörter und Telefonnummern sind in den Suchergebnissen hervorgehoben.

Im bereitgestellten Beispielsynonymverzeichnis können Sie sehen, welche Schlüsselwörter in welche semantischen Abfragen umgesetzt werden.

```
<?xml version="1.0" encoding="UTF-8"?>
<synonymgroups xmlns="http://www.ibm.com/of/822/synonym/xml">
  <synonymgroup>
   <synonym>telephone number</synonym>
   <synonym>phone number</synonym>
   <synonym>telephone nbr</synonym>
   <synonym>phone nbr</synonym>
   <synonym>@xmlf2::'&lt;#phonenumber/&gt;'</synonym>
  </synonymgroup>
  <synonymgroup>
   <synonym>facsimile number</synonym>
   <synonym>fax number</synonym>
   <synonym>facsimile nbr</synonym>
   <synonym>fax nbr</synonym>
   <synonym>@xmlf2::'&lt;#phonenumber/&gt;'</synonym>
  </synonymgroup>
  <synonymgroup>
   <synonym>e-mail address</synonym>
   <synonym>email address</synonym>
    <synonym>@xmlf2::'&lt;#email/&gt;'</synonym>
  </synonymgroup>
  <synonymgroup>
   <synonym>URL</synonym>
   <synonym>unified resource locator</synonym>
   <synonym>Web address</synonym>
   <synonym>@xmlf2::'&lt;#url/&gt;'</synonym>
  </synonymgroup>
</synonymgroups>
```

Zugehörige Konzepte

„Einfache semantische Suche mithilfe des Annotators für reguläre Ausdrücke" auf Seite 90
Zur Unternehmenssuche gehört die Analysesteuerkomponente für reguläre Ausdrücke, die bereits mit einem Regelsatz vorkonfiguriert ist, der das Erkennen von Telefonnummern, URL-Adressen und E-Mail-Adressen in Textdokumenten ermöglicht.

Die Regelsatzdatei

Im Annotator für reguläre Ausdrücke definiert die XML-Regelsatzdatei in der Form von regulären Ausdrücken die Regeln, die verwendet werden, um das Textdokument syntaktisch zu analysieren.

Die Regeln geben in sequenzieller Reihenfolge an, wo der Annotator im Dokumenttext nach was suchen soll und welche Aktionen im Fall einer Übereinstimmung auszuführen sind.

Wird der Annotator für reguläre Ausdrücke aufgerufen, wird die XML-Regelsatzdatei kompiliert, die die regulären Ausdrücke enthält, und mit Teilen des Dokumenttexts abgeglichen. Wird eine Übereinstimmung oder eine teilweise Übereinstimmung gefunden, wird die Annotation erstellt, die der betreffenden Regel zugeordnet ist, und in der allgemeinen Analysestruktur gespeichert.

Die in den Regeln verwendeten Typen müssen in der Typsystembeschreibung des Annotators für reguläre Ausdrücke definiert sein.

Der Annotator für reguläre Ausdrücke verarbeitet jeweils immer nur eine Regel und beginnt mit der ersten Regel in der XML-Regelsatzdatei. Für jede Regel wird der entsprechende kompilierte reguläre Ausdruck mit den in einem früheren Schritt erstellten Annotationen abgeglichen, z. B. mit Annotationen, die von Annotatoren erstellt wurden, die das Dokument vor dem Annotator für reguläre Ausdrücke verarbeitet haben. Die Annotationen, die mit den Regeln übereinstimmen, müssen denselben Typ aufweisen wie die im Annotatordeskriptor für reguläre Ausdrücke angegebenen Typen für die Eingabefunktionalität.

Wird eine Übereinstimmung gefunden, muss der in der Regel erstellte, angewendete Annotationstyp auch als gültiger Typ für die Ausgabefunktionalität im Annotatordeskriptor für reguläre Ausdrücke angegeben werden. Die neuen, von einer früheren Regel erstellten Annotationen können als Eingabeannotationen für Regeln verwendet werden, die später im XML-Regelsatz angewendet werden.

Zugehörige Konzepte

„Annotator für reguläre Ausdrücke" auf Seite 89
Mit dem Annotator für reguläre Ausdrücke können Sie eine benutzerdefinierte Textanalyse ausführen, ohne Ihre eigene Textanalysesteuerkomponente implementieren zu müssen. Auf der Basis eines Regelsatzes (reguläre Ausdrücke), den Sie selbst definieren können, erkennt der Annotator für reguläre Ausdrücke Informationsstrukturen in Textdokumenten und erstellt Annotationen zu den erkannten Informationen in der allgemeinen Analysestruktur.

Zugehörige Tasks

„Definieren von Regeln für reguläre Ausdrücke" auf Seite 94
Der Regelsatz definiert die regulären Ausdrücke, die mit dem Dokumenttext abgeglichen werden, und die Aktionen, die der Annotator für reguläre Ausdrücke ausführen muss, wenn ein Muster erkannt wird.

Zugehörige Verweise

„Der Annotatordeskriptor" auf Seite 99
Der XML-Annotatordeskriptor für reguläre Ausdrücke enthält beschreibende Informationen zum Annotator für reguläre Ausdrücke, die für die Ausführung des Annotators erforderlich sind.

„Protokollierung" auf Seite 102
Alle Protokollnachrichten des Annotators für reguläre Ausdrücke werden in die Protokolldatei der aktuellen Objektgruppe geschrieben.

Definieren von Regeln für reguläre Ausdrücke

Der Regelsatz definiert die regulären Ausdrücke, die mit dem Dokumenttext abgeglichen werden, und die Aktionen, die der Annotator für reguläre Ausdrücke ausführen muss, wenn ein Muster erkannt wird.

Informationen zu dieser Task

Die XML-Regelsatzdatei muss die im folgenden Beispiel dargestellte Regelsyntax befolgen. Hierbei handelt es sich um die Regelsatzdatei für den Beispielannotator für reguläre Ausdrücke zum Erkennen von Telefonnummern, URL-Adressen und E-Mail-Adressen.

Das Element der Ausgangsebene ist das Element `<ruleSet>`, das mindestens ein Element `<rule>` enthält. Jedes Element `<rule>` wiederum definiert einen regulären Java-Ausdruck, der aus einem Attribut `regEx` sowie aus den Attributen `matchStrategy` und `matchType` besteht. Die Aktion ist im Element `<createAnnotation>` definiert, das die Annotations-ID und den Annotationstyp angibt.

```
<?xml version="1.0" encoding="UTF-8"?>
<ruleSet xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ruleSet.xsd">
 <!-- Phone Number -->
 <!-- This rule matches different ways of writing telephone numbers,
   for example, 01234-12345, 01234 / 122-32, (001234)12345,
   +49 (0) 123412345, (123) 123 1234,
   1-800-IBM-4YOU -->
 <rule regEx="(?x)(\s|\b)(
0{1,2}[1-9]{1}[0-9]{1,5}\x20?[-/\\]\x20?[1-9]{1}([0-9]{1,8}-?)
  {1,3}[0-9]{1,}
|\(0[1-9]{1}[0-9]{1,3}\)\x20?[1-9]{1}[0-9]{2,8}
|\(00[1-9]{1}[0-9]{1,8}\)\x20?[1-9]{1}[0-9]{2,10}
|\((0\x20?[1-9]{1}[0-9]{1,3}|00\x20?[1-9]{1}[0-9]{1,8})\)\x20?[1-9]
   {1}[0-9] {1,3}(\x20[0-9]{2,4}){1,5}
|(0\x20?[1-9]{1}[0-9]{1,3}|00\x20?[1-9]{1}[0-9]{1,8})\x20?[/\\]\x20?
   [1-9]{1} [0-9]{1,3}(\x20[0-9]{2,4}){1,5}
|\(?\+[1-9]{1}[0-9]{0,3}\)?([-\x20]|\x20?\(0\))[-\x20]?[1-9]
   {1}[0-9]{1,10}
|\(?\+[1-9]{1}[0-9]{0,3}\)?([-\x20]|\x20?\(0\))[-\x20]?[1-9]
   {1}[0-9]{1,3}[-\x20]([0-9]{2,5}[-\x20]?){1,4}
|(1-)?[0-9]{3}-[0-9]{3}-[0-9]{4}
|\([1-9]{1}[0-9]{2}\)\x20[0-9]{3}[-\x20][0-9]{4}
|1-(800|888|877|866)-([A-Z0-9]{7}|[A-Z0-9]{3}-[A-Z0-9]
   {4}|[A-Z0-9]{4}-[A-Z0-9]{3})
)(?!(\d|\x20\d|-\d))(\s|\b)"
  matchStrategy="matchAll" matchType="uima.tcas.DocumentAnnotation">
  <createAnnotation id="phonenumber" type="com.ibm.es.uima.PhoneNumber">
   <begin group="0"/>
   <end group="0"/>
  </createAnnotation>
 </rule>

 <!-- potential Phone Number -->
 <!-- This rule matches numbers that resemble telephone numbers but could
 also be anything else. For example, 0123 1234 123,
 +123456789, 123 123 1234 -->
 <rule regEx="(?x)(\s|\b)(
  0[1-9]{1}[0-9]{1,3}\x20[1-9]{1}[0-9]*\x20?([0-9]{2,}\x20?)+
  |00\x20?[1-9]{1}[0-9]{0,3}\x20[1-9]{1}[0-9]{1,3}\x20?[1-9]
   {1}([0-9]{2,}\x20?)+
  |\+[1-9]{1}[0-9]{0,3}[1-9]{1}[0-9]{6,}
  |[1-9]{1}[0-9]{2}\x20[0-9]{3}\x20[0-9]{4}
  )(?!(\d|\x20\d|-\d))(\s|\b)"
```

```
 matchStrategy="matchAll" matchType="uima.tcas.DocumentAnnotation">
 <createAnnotation id="potential_phonenumber"
    type="com.ibm.es.uima.PotentialPhoneNumber">
  <begin group="0"/>
  <end group="0"/>
 </createAnnotation>
</rule>
<!-- URL Annotation -->
<!-- This rule matches URLs, for example, http://www.ibm.com -->
<rule regEx="(?x)(\s|\b)(
   http://[\w\-]+([\.][\w\-]+)+([/][\w\~\(\)\-\?=%\u0026\#]*)*
   |www.[\w\-]+([\.][\w\-]+)+([/][\w\~\(\)\-\?=%\u0026\#]*)*
  )(\s|\b)"
 matchStrategy="matchAll" matchType="uima.tcas.DocumentAnnotation">
 <createAnnotation id="url" type="com.ibm.es.uima.URL">
  <begin group="0"/>
  <end group="0"/>
 </createAnnotation>
</rule>
<!-- Email Annotation -->
<!-- This rule matches e-mail addresses, for example, yourName@domain.com -->
<rule regEx="(?x)(\s|\b)(
   [a-zA-Z0-9][\w\.-]*[a-zA-Z0-9]@[a-zA-Z0-9]([\.-]?\w)*\.[a-zA-Z]
   {2,3})(\s|\b)"
 matchStrategy="matchAll" matchType="uima.tcas.DocumentAnnotation">
 <createAnnotation id="email" type="com.ibm.es.uima.Email">
  <begin group="0"/>
  <end group="0"/>
 </createAnnotation>
</rule>
</ruleSet>
```

Vorgehensweise

Gehen Sie wie folgt vor, um den XML-Regelsatz für den Annotator für reguläre Ausdrücke zu erstellen, der Ihre benutzerdefinierten regulären Ausdrücke definiert:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die XML-Regelsatzdatei heißt `ruleSet.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche im Verzeichnis *ES_INSTALL_ROOT*`/packages/uima/regex/`.
2. Nehmen Sie die Zuordnungen in ein Element `<ruleSet xmlns="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="ruleSet.xsd">` auf. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element `<rule>` hinzu, das ein Attribut `regEx` mit dem Muster des regulären Ausdrucks, ein Attribut `matchStrategy` und ein Attribut `matchType` enthält.

   Der Annotator unterstützt die Syntax für reguläre Ausdrücke von Java 1.4 vollständig. Eine Einführung in reguläre Ausdrücke und die vollständige Syntax finden Sie in der Java-Dokumentation unter http://java.sun.com/j2se/1.4.2/docs/api/java/util/regex/Pattern.html.

   Das Attribut `matchStrategy` gibt an, wie zu suchen ist, wenn z. B. alle Übereinstimmungen im Dokument gefunden werden sollen oder wenn die Textübereinstimmung exakt ein muss. Es gibt drei Möglichkeiten, nach Übereinstimmungen zu suchen:

   - `matchFirst` stoppt bei der ersten Textsequenz, die mit dem regulären Muster übereinstimmt.

- `matchAll` sucht nach allen Textsequenzen eines Dokuments, die mit dem regulären Muster übereinstimmen.
- `matchComplete` gleicht nur Textsequenzen ab, bei denen es sich um exakte Übereinstimmungen handelt. Wenn z. B. das Muster ″foo″ gegeben ist, passt nur der Suchbegriff ″foo″, und ″foobar″ ergibt keine Übereinstimmung.

`matchType` legt den Annotationstyp fest, mit dem die Regel abgeglichen wird. Auf diese Weise können Sie die Übereinstimmung Ihres regulären Ausdrucks beschränken, z. B. innerhalb einer vorhandenen Tokenannotation. So können Sie vermeiden, dass zu viel Inhalt innerhalb einer Regel abgeglichen wird. Mögliche Typen sind die zulässigen Eingabeannotationstypen für den Annotator (im Annotatordeskriptor definiert), wie z. B. `uima.tt.DocumentAnnotation`, `uima.tt.ParagraphAnnotation` und benutzerdefinierte Typen wie z. B. `foo.bar.MeineAnnotation`. Manchmal wird der Ausgabetyp einer Regel als Eingabetyp für die nachfolgende Regel verwendet. `matchType` ermöglicht es Ihnen, den Suchbereich bestimmter Regeln zu beschränken.

4. Fügen Sie ein Element `<createAnnotation>` hinzu, das die Aktion definiert, die der Annotator für reguläre Ausdrücke ausführen soll, wenn eine Übereinstimmung gefunden wird.

   Jedes Element `createAnnotation` hat zwei Attribute:

   - `id` gibt die Annotation eindeutig an und wird als Verweis auf die Annotation verwendet.
   - `type` gibt den erstellten Annotationstyp an.

5. Fügen Sie die folgenden Komponentenelemente hinzu, die die Übereinstimmungsposition für das Element `<createAnnotation>` definieren:
   - Obligatorisch: `<begin>` gibt an, wo die Übereinstimmung beginnt. Dieses Element hat zwei Attribute:
     - Obligatorisch: `group` gibt die Java-Erfassungsgruppe an. Es können Werte zwischen 0 (Übereinstimmung einer vollständigen Textsequenz) und 9 (mehrere Erfassungsgruppen) angegeben werden.
     - Optional: `location` gibt (abhängig von der Position der runden Klammern) eine Position innerhalb der Übereinstimmungsgruppe an: `start` (linke runde Klammer) oder `end` (rechte runde Klammer).
   - Obligatorisch: `<end>` gibt an, wo die Übereinstimmung endet. Dieses Element hat zwei Attribute:
     - Obligatorisch: `group` gibt die Erfassungsgruppe an. Es können Werte zwischen 0 (Übereinstimmung einer vollständigen Textsequenz) und 9 (nachfolgende und immer kleinere Erfassungsgruppen) angegeben werden.
     - Optional: `location` gibt (abhängig von der Position der runden Klammern) eine Position innerhalb der Übereinstimmungsgruppe an: `start` (linke runde Klammer) oder `end` (rechte runde Klammer).
   - Optional: `<setFeature>` erstellt eine Komponente und ordnet diese der Annotation zu. Dieses Element hat zwei Attribute:
     - `name` ist der Name der Komponente, wie Sie ihn in der Typsystembeschreibung definiert haben.
     - `type` gibt den Typ des Komponentenwerts an: Zeichenfolge (`String`), eine ganze Zahl (`Integer`), eine Gleitkommazahl (`Float`) oder ein Verweis (`Referenz`). Der Typ muss mit dem in der Typsystembeschreibung des Annotators für die Komponente definierten Bereichstyp identisch sein.

       Komponenten des Typs `Reference` werden verwendet, um eine Verknüpfung zwischen zwei Annotationen herzustellen, die eine semantische

Beziehung angibt. Der Elementinhalt von <setFeature> muss mit der id des Elements <createAnnotation> übereinstimmen, mit dem die Verknüpfung hergestellt werden soll.

Zugehörige Konzepte

„Die Regelsatzdatei" auf Seite 93
Im Annotator für reguläre Ausdrücke definiert die XML-Regelsatzdatei in der Form von regulären Ausdrücken die Regeln, die verwendet werden, um das Textdokument syntaktisch zu analysieren.

Anpassen des Annotators für reguläre Ausdrücke

Sie können die Beispielkonfiguration des Annotators für reguläre Ausdrücke so anpassen, dass sie neue Entitäten erkennt (z. B. Seriennummern von Produkten), oder die Regeln für reguläre Ausdrücke an vorhandene Entitäten anpassen (z. B. zur Erkennung unternehmensspezifischer Telefonnummern), indem Sie an Dateien für Beispielregelsätze und Typsyteme kleinere Änderungen vornehmen.

Die Regelsatzdatei und die Typsystembeschreibung müssen nach der Modifizierung der Verarbeitungsenginearchivdatei (PEAR-Datei) hinzugefügt werden. Nachdem Sie die PEAR-Datei aktualisiert haben, können Sie die angepasste Textanalysesteuerkomponente für reguläre Ausdrücke erneut dem System für die Unternehmenssuche hinzufügen.

Für eine ausgefeilte Anpassung des Annotators für reguläre Ausdrücke empfehlen wir Ihnen dringend, die Tools in UIMA SDK zu verwenden. Mithilfe dieser Tools können Sie die Typsystembeschreibung und die Deskriptordateien so erstellen oder aktualisieren, dass es möglich ist, den Annotator mit anderen Annotatoren zu einer zusammengefassten Analysesteuerkomponente zusammenzuschließen und ein neues Verarbeitungsenginearchiv (PEAR-Datei) zu erstellen, das alle Ressourcen enthält, die erforderlich sind, um den Annotator in der Unternehmenssuche zu verwenden. Informationen zu den Tools, die Ihnen für diese Tasks zur Verfügung stehen, finden Sie in der Dokumentation von UIMA SDK.

Vorgehensweise

Wenn Sie den Annotator für reguläre Ausdrücke durch Hinzufügen neuer Regeln und Entitäten anpassen wollen oder wenn Sie vorhandene Regeln ändern wollen, können Sie die bereitgestellte PEAR-Datei des Beispielannotators für reguläre Ausdrücke wie folgt aktualisieren:

1. Erstellen Sie auf dem System ein neues Verzeichnis mit dem Namen xml.
2. Kopieren Sie die Beispielregeldatei of_sample_regex_rules.xml in das Unterverzeichnis *ES_INSTALL_ROOT*/packages/uima/regex/ im Verzeichnis xml, und ändern Sie die Datei so, dass sie die angepassten Mustererkennungsregeln enthält. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden.
3. Kopieren Sie die entsprechende Systembeschreibungsdatei of_sample_typesystem.xml aus dem Verzeichnis ES_INSTALL_ROOT/packages/uima/regex/ in das Verzeichnis xml, und ändern Sie die Datei so, dass sie die Definitionen für die Typen enthält, die für die neuen Regeln erforderlich sind.
4. Wenn Sie nur einige wenige neue Regeln hinzufügen oder vorhandene Regeln ändern, ist es nicht erforderlich, den Annotatordeskriptor zu ändern. Wenn Sie

weitere Änderungen beabsichtigen oder wenn Sie weitere benutzerdefinierte Analyseschritte verwenden, müssen Sie prüfen, ob der Annotatordeskriptor modifiziert werden muss.

5. Verwenden Sie ein Archivierungsdienstprogramm Ihrer Wahl, um eine Kopie der PEAR-Datei des Annotators für reguläre Ausdrücke zu aktualisieren, sodass Ihre beiden aktualisierten Dateien darin enthalten sind. Kopieren Sie z. B. die Datei `of_regex.pear` aus dem Verzeichnis `ES_INSTALL_ROOT/packages/uima/regex/` in das übergeordnete Verzeichnis des von Ihnen erstellten Verzeichnisses `xml`. Setzen Sie anschließend mit dem Java-Befehlszeilentool jar(z. B. als Bestandteil von IBM Java SDK) von diesem übergeordneten Verzeichnis aus die folgenden Befehle ab:

   ```
   "jar -uf of_regex.pear -C xml/ of_sample_regex_rules.xml"
   "jar -uf of_regex.pear -C xml/ of_sample_regex_typesystem.xml"
   ```

6. Fügen Sie den Annotator für reguläre Ausdrücke über die Administrationskonsole für die Unternehmenssuche dem System für die Unternehmenssuche als benutzerdefinierte Textanalysesteuerkomponente hinzu, und ordnen Sie ihn einer Testdokumentobjektgruppe zu.
7. Überprüfen Sie die Analyseergebnisse des Annotators für reguläre Ausdrücke, indem Sie die Merkmale der Dokumentobjektgruppe aktualisieren, sodass mithilfe der XCAS-Funktion zum Erstellen eines Speicherauszugs eine lesbare XML-Ausgabe der in der allgemeinen Analysestruktur gespeicherten Analyseergebnisse erzeugt wird.
8. Verarbeiten Sie die Testdokumente, und verwenden Sie XCAS Annotation Viewer, um den Inhalt der XML-Dateien anzuzeigen.
9. Wenn Sie mit den Annotationen zufrieden sind, die der Annotator auf der Basis Ihrer benutzerdefinierten regulären Ausdrücke erstellt hat, bearbeiten Sie die Merkmale der Dokumentobjektgruppe erneut, und inaktivieren Sie die XML-Ausgabe lesbarer Analyseergebnisse durch den Parser. Sind weitere Änderungen an der Regelsatzdatei erforderlich, müssen Sie die Schritte für die Aktualisierung der PEAR-Datei wiederholen.
10. Erstellen Sie eine Datei für die Zuordnung der allgemeinen Analysestruktur zum Index, um die Analyseergebnisse zu indexieren, oder eine Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank, um die Ergebnisse einer Datenbank hinzuzufügen. Sie können die bereitgestellte Beispieldatei für die Zuordnung der allgemeinen Analysestruktur zum Index als Ausgangspunkt verwenden, um Ihre eigene Datei für die Zuordnung der allgemeinen Analysestruktur zum Index zu erstellen.
11. Verwenden Sie die Administrationskonsole für die Unternehmenssuche, um Zuordnungsdateien hinzuzufügen und Ihren Dokumentobjektgruppen zuzuordnen.
12. Durchsuchen Sie Ihre Annotationen mithilfe von XML-Fragmenten oder alternativ unter Verwendung semantischer Erweiterung während der Synonymsuche.

Zugehörige Konzepte

„Einfache semantische Suche mithilfe des Annotators für reguläre Ausdrücke" auf Seite 90
Zur Unternehmenssuche gehört die Analysesteuerkomponente für reguläre Ausdrücke, die bereits mit einem Regelsatz vorkonfiguriert ist, der das Erkennen von Telefonnummern, URL-Adressen und E-Mail-Adressen in Textdokumenten ermöglicht.

Zugehörige Tasks

„Anzeigen der Ergebnisse des Basisannotators und der angepassten Textanalyse" auf Seite 14

Wenn Sie die Analyseergebnisse anzeigen wollen, die nach der Syntaxanalyse von einem beliebigen Annotator in der Unternehmenssuche erzeugt wurden, müssen Sie die Merkmale der Dokumentobjektgruppe aktualisieren, damit eine lesbare XML-Version der in der allgemeinen Analysestruktur gespeicherten Analyseergebnisse erzeugt wird.

Der Annotatordeskriptor

Der XML-Annotatordeskriptor für reguläre Ausdrücke enthält beschreibende Informationen zum Annotator für reguläre Ausdrücke, die für die Ausführung des Annotators erforderlich sind.

Wenn Sie nur den Annotator für reguläre Ausdrücke und keine weiteren benutzerdefinierten Analyseschritte verwenden, ist es nur in den folgenden Fällen erforderlich, den Deskriptor zu ändern:

- Wenn Sie den Dateinamen der Regelsatzdatei (im Element `<externalResourceDependencies>`) ändern wollen.
- Wenn Sie mehrere Regelsatzdateien verwenden wollen.
- Wenn Sie den Namen der Beschreibungsdatei des Typsystems ändern wollen.

Wenn Sie zusätzliche benutzerdefinierte Analyseschritte verwenden, müssen Sie in den folgenden Fällen den Deskriptor ändern:

- Wenn Ihre benutzerdefinierte Analyse Annotationen verwenden soll, die der Annotator für reguläre Ausdrücke erstellt hat. In diesem Fall müssen Sie die Ausgabefunktionalität im Annotatordeskriptor ändern.
- Wenn Sie Regeln für reguläre Ausdrücke definiert haben, die mit Annotationstypen übereinstimmen müssen, die in früheren benutzerdefinierten Analyseschritten erstellt wurden. In diesem Fall müssen Sie die Eingabefunktionalität im Annotatordeskriptor aktualisieren.

Verwenden Sie die Tools von UIMA SDK, um den Annotatordeskriptor zu erstellen oder zu aktualisieren, und erstellen Sie das Verarbeitungsenginearchiv (PEAR-Datei) erneut, das alle für die Verwendung des Annotators in der Unternehmenssuche erforderlichen Ressourcen enthält. Informationen zu den Tools, die Ihnen für diese Tasks zur Verfügung stehen, finden Sie in der Dokumentation von UIMA SDK unter http://www.alphaworks.ibm.com/tech/uima/.

Konfigurationsparameter

Der Annotator für reguläre Ausdrücke hat nur einen Konfigurationsparameter namens `String2NumberImpl`, der auf den Namen der Klasse gesetzt sein muss, die die Schnittstelle `com.ibm.uima.an_regex.String2Number` implementiert. Der Annotator für reguläre Ausdrücke muss eine Implementierung dieser Klasse aufweisen, da sonst eine Ausnahmebedingung auftritt. Wenn Sie den Annotator für reguläre Ausdrücke an Ihre Bedürfnisse anpassen wollen, können Sie Ihre eigene Implementierung der Schnittstelle `String2Number` bereitstellen, indem Sie Ihren Klassennamen an die XML-Deskriptordatei übergeben.

Die Schnittstelle `String2Number` deklariert zwei Methoden, `toInt(String)` und `toFloat(String)`, die eine Zeichenfolgedarstellung einer ganzen Zahl oder eines Gleitkommawerts in die entsprechende ganze Zahl oder den entsprechenden Gleitkommawert umsetzen. Diese beiden Methoden werden verwendet, um eine Zahl, die ein Trennzeichen enthält, in eine gültige ganze Zahl oder einen gültigen Gleitkommawert im Java-Format umzusetzen.

Die Standardimplementierung von com.ibm.uima.an_regex.String2Number_impl verwendet einen Punkt (.) als Dezimaltrennzeichen und ein Komma (,) als Tausendertrennzeichen. Wenn z. B. in einem Textdokument 1,999.00 gefunden wird, konvertiert toInt dies in 1999. toFloat gibt 1999.00 zurück.

Beispiel

Der Abschnitt für Konfigurationsparameter des Deskriptors sieht wie folgt aus:

```
<configurationParameters>
 <configurationParameter>
  <name>String2NumberImpl</name>
  <description>Implementierung der Schnittstelle
   com.ibm.uima.an_regex.String2Number</description>
  <type>String</type>
  <multiValued>false</multiValued>
  <mandatory>true</mandatory>
 </configurationParameter>

 <configurationParameterSettings>
  <nameValuePair>
   <name>String2NumberImpl</name>
   <value>
    <string>com.ibm.uima.an_regex.impl.String2Number_impl</string>
   </value>
  </nameValuePair>
 </configurationParameterSettings>
</configurationParameters>
```

Funktionalität

Die Eingabe- und Ausgabefunktionalität des Annotators für reguläre Ausdrücke und die von ihm unterstützten Sprachen werden im Annotatordeskriptor im Abschnitt für die Funktionalität definiert.

Die Eingabefunktionalität (Eingabetypen) in der Deskriptordatei muss den in der Regelsatzdatei verwendeten Abgleichungstypen entsprechen. Wenn die Regeln nur den Typ uima.tt.DocumentAnnotation verwenden, brauchen Sie keine weitere Eingabefunktionalität zu deklarieren, da dieser Typ immer definiert ist. Alle anderen Typen müssen definiert werden.

Die vom Annotator für reguläre Ausdrücke erstellten Annotationstypen werden im Abschnitt für die Ausgabefunktionalität angegeben. Diese Typen müssen den in der Regelsatzdatei deklarierten Ausgabetypen entsprechen.

Da der Annotator für reguläre Ausdrücke sprachunabhängig ist, geben Sie x-unspecified an, das für jede beliebige Sprache steht.

Typsystembeschreibung

Im Abschnitt für die Typsystembeschreibung im XML-Annotatordeskriptor für reguläre Ausdrücke ist das vom Annotator verwendete Typsystem definiert. Die in der XML-Regelsatzdatei verwendeten Typen, die in den Abschnitten für die Eingabe- und Ausgabefunktionalität des Annotatordeskriptors genannt werden, müssen mit den in der Typsystembeschreibung definierten Typen übereinstimmen.

Beispiel

Der Abschnitt für die Typsystembeschreibung des Deskriptors importiert die XML-Deskriptordatei für das Typsystem:

```
<typeSystemDescription>
 <imports>
  <import location="./xml/of_sample_regex_typesystem.xml"/>
 </imports>
</typeSystemDescription>
```

Externe Ressourcen

Der Abschnitt für externe Ressourcen enthält die Dateien und Klassen, die für den Annotator erforderlich sind.

Für den Annotator für reguläre Ausdrücke ist die Regelsatzdatei erforderlich. Die Regelsatzdatei ist für den Annotator für reguläre Ausdrücke über die Schnittstelle `com.ibm.uima.an_regex.FileResource` verfügbar, die von der Klasse `com.ibm.uima.an_regex.impl.FileResource_impl` implementiert wird. Wenn Sie Ihre benutzerdefinierten Regeln an den Annotator für reguläre Ausdrücke übergeben wollen, müssen Sie den Namen der Regelsatzdatei im Annotatordeskriptor bereitstellen und Ihrem Klassenpfad die Position der Datei hinzufügen. Für den Zugriff auf die Regelsatzdatei verwendet der Annotator für reguläre Ausdrücke den Schlüssel `RuleSetDefinition`. Ändern Sie diesen Schlüssel nicht, da sonst der Annotator für reguläre Ausdrücke den Regelsatz nicht findet und keine Initialisierung ausführen kann.

Benutzerdefinierte Annotatoren, die Sie für die Unternehmenssuche implementieren, können nicht die UIMA-Einstellung `datapath` verwenden, um externe Ressourcen zu durchsuchen. Wenn Sie externe Ressourcen durchsuchen möchten, müssen Sie im Klassenpfad des benutzerdefinierten Annotators die Pfadnamen für die Ressourcen angeben. Informationen zum Verwenden des PEAR-Generierungsassistenten zum Angeben von Klassenpfadeinstellungen für benutzerdefinierte Annotatoren finden Sie in der SDK-Dokumentation von UIMA unter http://www.alphaworks.ibm.com/tech/uima/.

Beispiel

Der Abschnitt für externe Ressourcen des Deskriptors sieht wie folgt aus:

```
<externalResourceDependencies>
 <externalResourceDependency>
  <key>RuleSetDefinition</key>
  <description>Regelsatzdefinition</description>
  <interfaceName>com.ibm.uima.an_regex.FileResource</interfaceName>
  <optional>false</optional>
 </externalResourceDependency>
</externalResourceDependencies>
<resourceManagerConfiguration>
 <externalResources>
  <externalResource>
   <name>of_samples_regex_rules</name>
   <description>Regelsatzdefinitionsdatei für Zimmernummern</description>
   <fileResourceSpecifier>
    <fileUrl>file:of_samples_regex_rules.xml</fileUrl>
   </fileResourceSpecifier>
   <implementationName>
    com.ibm.uima.an_regex.impl.FileResource_impl</implementationName>
  </externalResource>
 </externalResources>
 <externalResourceBindings>
  <externalResourceBinding>
   <key>RuleSetDefinition</key>
```

```
    <resourceName>of_samples_regex_rules</resourceName>
  </externalResourceBinding>
 </externalResourceBindings>
</resourceManagerConfiguration>
```

Zugehörige Konzepte

„Annotator für reguläre Ausdrücke" auf Seite 89
Mit dem Annotator für reguläre Ausdrücke können Sie eine benutzerdefinierte Textanalyse ausführen, ohne Ihre eigene Textanalysesteuerkomponente implementieren zu müssen. Auf der Basis eines Regelsatzes (reguläre Ausdrücke), den Sie selbst definieren können, erkennt der Annotator für reguläre Ausdrücke Informationsstrukturen in Textdokumenten und erstellt Annotationen zu den erkannten Informationen in der allgemeinen Analysestruktur.

„Die Regelsatzdatei" auf Seite 93
Im Annotator für reguläre Ausdrücke definiert die XML-Regelsatzdatei in der Form von regulären Ausdrücken die Regeln, die verwendet werden, um das Textdokument syntaktisch zu analysieren.

Zugehörige Verweise

„Protokollierung"
Alle Protokollnachrichten des Annotators für reguläre Ausdrücke werden in die Protokolldatei der aktuellen Objektgruppe geschrieben.

Protokollierung

Alle Protokollnachrichten des Annotators für reguläre Ausdrücke werden in die Protokolldatei der aktuellen Objektgruppe geschrieben.

Die Objektgruppenprotokolldateien befinden sich im Verzeichnis `ES_NODE_ROOT/logs/` und weisen Namen im Formt `<objektgruppen-id>_<aktuelles_datum>.log` auf. Sie können die Protokolldateien mit den Scripts `esviewlogs.sh` und `esviewlogs.bat` anzeigen.

Es gibt sieben mögliche Protokollebenen:

- Error
- Warning
- Info
- Config
- Fine
- Finer
- Finest

Sie können die Zuordnung für Fehlernachrichten und für Warnungen nicht ändern. Standardmäßig werden nur Nachrichten der Ebenen `Info`, `Warning` und `Error` in die Protokolldatei geschrieben. Hierbei handelt es sich um die Standardprotokollebenen, die von der Unternehmenssuche verwendet werden. Die anderen Protokollebenen können zugeordnet werden, wenn ausführliche Informationen gewünscht werden.

Damit Protokollnachrichten vom Annotator für reguläre Ausdrücke empfangen werden, muss die Protokollebene mindestens auf `Config` gesetzt sein. Auf dieser Ebene protokolliert der Annotator Konfigurationseinstellungen, wie z. B. die verwendete Regelsatzdatei und den Namen der Implementierungsklasse für die Schnittstelle `com.ibm.uima.an_regex.String2Number`.

Wenn Sie die Protokollebene z. B. auf `Finer` setzen, protokolliert der Annotator die Annotationen, die nicht erstellt werden konnten. Damit können Sie feststellen, warum nicht alle von Ihnen erwarteten Annotationen erstellt wurden. So kann z. B. ein Fehler in einem Ihrer regulären Ausdrücke vorliegen, oder eine optionale Erfassungsgruppe hat möglicherweise keinen Text im Dokument abgeglichen. Ähnlich wird, wenn Sie angeben, dass eine Komponente auf die Textsequenz gesetzt wird, die mit einer Erfassungsgruppe übereinstimmt, aber keine übereinstimmende Textsequenz vorhanden ist, die Komponente auf Null gesetzt.

Wenn Sie sehr ausführliche Informationen wünschen, setzen Sie die Protokollebene auf `Finest`. Auf dieser Ebene protokolliert der Annotator das aktuelle Muster des regulären Ausdrucks, den zurzeit analysierten Abschnitt des Dokumenttexts und alle erstellten Annotationen und Komponenten. Wenn Sie sehr detailliert protokollieren, besonders mit den Protokollebenen `Finer` und `Finest`, wirkt sich dies negativ auf die Gesamtleistung des Annotators aus.

Wenn Sie eine detaillierte Protokollebenenzuordnung benötigen, ändern Sie die Konfigurationsdatei mit dem Namen `tokenizer.properties` im Verzeichnis *ES_NODE_ROOT*`/master_config/parserservice/`, indem Sie z. B. die Konfigurationseinstellung `trevi.tokenizer.jedii.InformationalLevelMapping=Info` in `trevi.tokenizer.jedii.InformationalLevelMapping=Finest` ändern.

Damit die Änderungen der Protokollebene aktiviert werden, müssen Sie alle Parserprozesse über die Administrationskonsole stoppen. Anschließend müssen Sie die Parser-Servicesitzung stoppen und erneut starten, indem Sie den folgenden Befehl an der Befehlszeile eingeben:

```
>esadmin session parserservice stop
>esdamin session parserservice start
```

Danach können Sie die syntaktische Analyse erneut starten. Nun sollte die neue Protokollebene aktiv sein. Diese Schritte müssen Sie bei jeder Änderung der Protokollebene wiederholen.

Zugehörige Konzepte

„Annotator für reguläre Ausdrücke" auf Seite 89
Mit dem Annotator für reguläre Ausdrücke können Sie eine benutzerdefinierte Textanalyse ausführen, ohne Ihre eigene Textanalysesteuerkomponente implementieren zu müssen. Auf der Basis eines Regelsatzes (reguläre Ausdrücke), den Sie selbst definieren können, erkennt der Annotator für reguläre Ausdrücke Informationsstrukturen in Textdokumenten und erstellt Annotationen zu den erkannten Informationen in der allgemeinen Analysestruktur.

„Die Regelsatzdatei" auf Seite 93
Im Annotator für reguläre Ausdrücke definiert die XML-Regelsatzdatei in der Form von regulären Ausdrücken die Regeln, die verwendet werden, um das Textdokument syntaktisch zu analysieren.

Zugehörige Verweise

„Der Annotatordeskriptor" auf Seite 99
Der XML-Annotatordeskriptor für reguläre Ausdrücke enthält beschreibende Informationen zum Annotator für reguläre Ausdrücke, die für die Ausführung des Annotators erforderlich sind.

Dokumentation für die Unternehmenssuche

Die Dokumentation zu OmniFind Enterprise Edition steht im PDF- oder HTML-Format zur Verfügung.

Das Installationsprogramm von OmniFind Enterprise Edition installiert automatisch die Informationszentrale, die HTML-Versionen der Dokumentation für die Unternehmenssuche umfasst. Bei einer Installation auf mehreren Servern wird die Informationszentrale auf beiden Suchservern installiert. Wenn Sie die Informationszentrale nicht installieren, wird beim Anklicken von **Hilfe** die Informationszentrale auf einer IBM Website geöffnet.

Wechseln Sie in das Verzeichnis ES_INSTALL_ROOT/docs/*ländereinstellung*/pdf, um installierte Versionen der PDF-Dokumente anzuzeigen. Wenn Sie beispielsweise Dokumente in englischer Sprache suchen, wechseln Sie in das Verzeichnis ES_INSTALL_ROOT/docs/en_US/pdf.

Über die Site mit der Dokumentation zu OmniFind Enterprise Edition Version 8.5 können Sie auf die PDF-Versionen der Dokumentation in allen verfügbaren Sprachen zugreifen.

Über die Unterstützungssite für OmniFind Enterprise Edition können Sie außerdem auf Produktdownloads, Fixpacks, technische Hinweise und die Informationszentrale zugreifen.

In der folgenden Tabelle ist die verfügbare Dokumentation mit Dateinamen und Speicherposition aufgeführt.

Tabelle 12. Dokumentation für die Unternehmenssuche

Titel	Dateiname	Speicherposition
Informationszentrale		http://publib.boulder.ibm.com/infocenter/discover/v8r5/
Installationshandbuch für die Unternehmenssuche	iiysi.pdf	ES_INSTALL_ROOT/docs/*ländereinstellung*/pdf/
Schnelleinstieg (Dieses Dokument ist in Englisch, Französisch und Japanisch auch als Hardcopy verfügbar.)	OmniFindEE850_qsg_*zweibuchstabige*_*ländereinstellung*.pdf	ES_INSTALL_ROOT/docs/*ländereinstellung*/pdf/
Verwaltung der Unternehmenssuche	iiysa.pdf	ES_INSTALL_ROOT/docs/*ländereinstellung*/pdf/
Programming Guide and API Reference for Enterprise Search	iiysp.pdf	ES_INSTALL_ROOT/docs/en_US/pdf/
Fehlerbehebung und Nachrichtenreferenz	iiysm.pdf	ES_INSTALL_ROOT/docs/*ländereinstellung*/pdf/
Integration der Textanalyse	iiyst.pdf	ES_INSTALL_ROOT/docs/*ländereinstellung*/pdf/
Plug-in für Google Desktop Search	iiysg.pdf	ES_INSTALL_ROOT/docs/*ländereinstellung*/pdf/

Funktionen zur behindertengerechten Bedienung

Funktionen zur behindertengerechten Bedienung helfen Menschen mit Behinderungen, wie z. B. eingeschränkte Beweglichkeit oder eingeschränktes Sehvermögen, erfolgreich mit Softwareprodukten zu arbeiten.

IBM hat sich zum Ziel gesetzt, Produkte bereitzustellen, auf die jeder - unabhängig von Alter oder Behinderung - zugreifen kann.

Funktionen zur behindertengerechten Bedienung

Die folgende Liste enthält die wichtigsten Funktionen zur behindertengerechten Bedienung in OmniFind Enterprise Edition:

- Ausschließliche Bedienung über die Tastatur
- Häufig von Sprachausgabeprogrammen verwendete Schnittstellen

Die Informationszentrale von OmniFind Enterprise Edition und die zugehörigen Veröffentlichungen sind für die behindertengerechte Bedienung aktiviert. Die Funktionen zur behindertengerechten Bedienung der Informationszentrale werden unter http://publib.boulder.ibm.com/infocenter/discover/v8r5m0/topic/com.ibm.classify.nav.doc/dochome/accessibility_info.htm beschrieben.

Navigation über die Tastatur

Dieses Produkt verwendet Microsoft Windows-Standardnavigationstasten.

Sie können auch mithilfe der folgenden Direktaufrufe über die Tastatur im Installationsprogramm von OmniFind Enterprise Edition navigieren und die Schritte ausführen.

Tabelle 13. Direktaufrufe über die Tastatur für das Installationsprogramm

Aktion	Direktaufruf
Hervorheben eines Radioknopfs	Pfeiltaste
Auswählen eines Radioknopfs	Tabulatortaste
Hervorheben eines Druckknopfs	Tabulatortaste
Auswählen eines Druckknopfs	Eingabetaste
Wechseln zum nächsten oder vorhergehenden Fenster oder Ausführen eines Abbruchs	Heben Sie einen Druckknopf durch Drücken der Tabulatortaste hervor, und drücken Sie die Eingabetaste.
Inaktivieren des aktiven Fensters	Strg + Alt + Esc

Schnittstelleninformationen

Die Benutzerschnittstellen für die Administrationskonsole, die Mustersuchanwendung und die Anpassungsfunktion für die Suchanwendung sind browserbasierte Schnittstellen, die Sie über Microsoft Internet Explorer oder Mozilla FireFox anzeigen können. Eine Liste der Direktaufrufe über die Tastatur und andere Funktionen zur behindertengerechten Bedienung von Internet Explorer bzw. FireFox finden Sie in der Onlinehilfe des jeweiligen Browsers.

Zugehörige Informationen zur behindertengerechten Bedienung

Sie können die Veröffentlichungen für OmniFind Enterprise Edition mithilfe von Adobe Acrobat Reader im Adobe PDF-Format anzeigen. Die PDFs werden auf einer CD bereitgestellt, die zum Lieferumfang des Produkts gehört. Alternativ können Sie unter http://www.ibm.com/support/docview.wss?rs=63 &uid=swg27010938 auf die Informationen zugreifen.

IBM und behindertengerechte Bedienung

Weitere Informationen zum Engagement von IBM hinsichtlich der behindertengerechten Bedienung finden Sie im IBM Human Ability and Accessibility Center.

Glossar der Begriffe für die Unternehmenssuche

Dieses Glossar enthält Begriffe, die in den Schnittstellen für die Unternehmenssuche und in der zugehörigen Dokumentation verwendet werden.

Abschließendes Zeichen (Trailing Character)
Ein Zeichen an der letzten Position in einem Wort.

Administrator für die Unternehmenssuche (Enterprise Search Administrator)
Eine Verwaltungsrolle, mit der ein Benutzer das gesamte System für die Unternehmenssuche verwalten kann.

Allgemeine Analysestruktur (CAS - Common Analysis Structure)
Eine Struktur, die den Inhalt und die Metadaten eines Dokuments sowie alle von einer Textanalysesteuerkomponente erstellten Analyseergebnisse speichert. Der gesamte Datenaustausch während der Dokumentanalyse wird mithilfe der allgemeinen Analysestruktur gehandhabt.

Allgemeine Übertragungsschicht (CCL - Common Communication Layer)
Die Kommunikationsinfrastruktur, die die verschiedenen Komponenten (Controller, Parser, Crawler, Indexserver) von OmniFind Enterprise Edition verbindet.

Analyseergebnisse (Analysis Results)
Die von Annotatoren erstellten Informationen. Analyseergebnisse werden in eine Datenstruktur geschrieben, die *allgemeine Analysestruktur* genannt wird. Analyseergebnisse, die von den benutzerdefinierten Textanalysesteuerkomponenten (Annotatoren) erstellt werden, können für die Suche verfügbar gemacht werden, indem Sie sie in den Index für die Unternehmenssuche aufnehmen.

Analysesteuerkomponente (Analysis Engine)
Siehe Textanalysesteuerkomponente.

Annotation
Informationen zu einer Textpassage. Eine Annotation (ergänzender Kommentar) könnte beispielsweise darauf hinweisen, dass eine kurze Textpassage für einen Unternehmensnamen steht. In UIMA (Unstructured Information Management Architecture) ist eine Annotation eine besondere Merkmalstruktur.

Annotator
Eine Softwarekomponente, die bestimmte linguistische Analysetasks ausführt und Annotationen (ergänzende Kommentare) erstellt und erfasst. Der Annotator (Kommentatorfunktion) ist die Analyselogikkomponente einer Analysesteuerkomponente.

Annotator für reguläre Ausdrücke (Regular Expression Annotator)
Eine Softwarekomponente, die Entitäten oder Informationseinheiten, wie z. B. Produktnummern, in einem Textdokument erkennt. Dies geschieht anhand von regulären Ausdrücken, die die genauen Muster beschreiben, nach denen im Dokumenttext gesucht wird. Entspricht einer der regulären Ausdrücke Teilen des Dokumenttexts, erstellt der Annotator für reguläre Ausdrücke die entsprechenden Annotationen mit der Übereinstimmung oder einem Teil davon. Diese Ausdrücke mit Annotationen werden dann unter Verwendung einer Indexzuordnungsdatei im Index für die Unter-

nehmenssuche oder unter Verwendung einer Datenbankzuordnungsdatei in einer JDBC-fähigen Datenbank gespeichert.

Archiv der Verarbeitungssteuerkomponente (Processing Engine Archive)
Eine komprimierte .pear-Archivdatei, die eine UIMA-Analysesteuerkomponente (Unstructured Information Management Architecture) sowie sämtliche Ressourcen enthält, die zu ihrer Nutzung für eine benutzerdefinierte Analyse in der Unternehmenssuche erforderlich sind.

Aufbereitung (Tokenization)
Die Syntaxanalyse der Eingabe in Token.

Aufspürfunktion (Discoverer)
Eine Crawlerfunktion, die feststellt, welche Datenquellen dem Crawler zum Abrufen von Informationen zur Verfügung stehen.

Aus Warteschlange entfernen (Dequeue)
Einträge aus einer Warteschlange entfernen.

Basisannotatoren für die Unternehmenssuche (Enterprise Search Base Annotators)
Eine Gruppe von Standardtextanalysesteuerkomponenten, die in der Unternehmenssuche verwendet werden, um Standarddokumentanalysen zu verarbeiten.

Bediener (Operator)
Ein Benutzer der Unternehmenssuche, der über die Berechtigung zum Beobachten, Starten und Stoppen von Prozessen auf Objektgruppenebene verfügt.

Begriffsextraktion (Concept Extraction)
Eine Textanalysefunktion, die signifikante Vokabularelemente (z. B. Personen, Orte oder Produkte) in Textdokumenten identifiziert und eine Liste dieser Elemente erstellt. Siehe auch Themenextraktion.

Benutzeragent (User Agent)
Eine Anwendung, die das Internet durchsucht und Informationen zu sich selbst auf den besuchten Sites hinterlässt. In der Unternehmenssuche ist der Web-Crawler ein Benutzeragent.

Benutzerdefinierte Steuerkomponente für Textanalyse (Custom Text Analysis Engine)
Eine Steuerkomponente für Textanalyse, die mithilfe des UIMA Software-Development-Kits (Unstructured Information Management Architecture SDK) erstellt wird und der Gruppe von Standardtextanalysesteuerkomponenten für die Unternehmenssuche (auch Basisannotatoren für die Unternehmenssuche genannt) hinzugefügt werden kann. Siehe auch Textanalysesteuerkomponente.

Berechtigungsnachweis (Credential)
Während der Authentifizierung zugewiesene detaillierte Informationen, die den Benutzer, gegebenenfalls vorhandene Gruppenzuordnungen und andere sicherheitsrelevante Identitätsattribute beschreiben. Mit Berechtigungsnachweisen lassen sich eine Vielzahl von Services ausführen, wie z. B. Berechtigung, Prüfung und Delegierung. Die Anmeldeinformationen (Benutzer-ID und Kennwort) für einen Benutzer sind Berechtigungsnachweise, mit denen der Benutzer auf ein Konto zugreifen kann.

Bereich (Place)
Ein virtueller Ort, der im Portal angezeigt wird und in dem sich Einzelpersonen sowie Gruppen zur Zusammenarbeit online treffen. In einem Portal verfügt jeder Benutzer über einen eigenen Bereich für seine persönliche

Aufgaben, und darüber hinaus haben Einzelpersonen und Gruppen Zugang zu einer Reihe gemeinsam genutzter Bereiche, die allgemein zugängliche oder eingeschränkte Bereiche sein können. Siehe auch Lotus QuickPlace-Bereich.

Bibliothek (Library)
Ein Systemobjekt, das anderen Objekten als Verzeichnis dient. Siehe auch Domino Document Manager-Bibliothek.

Boolesche Suche (Boolean Search)
Eine Suche, bei der mehrere Suchbegriffe mithilfe von Operatoren wie AND, NOT und OR kombiniert werden.

Boostklasse (Boost Class)
Ein Objekt, das Spezifikationen enthält, mit denen der relative Rang eines Dokuments in den Suchergebnissen beeinflusst werden kann.

Boostwort (Boost Word)
Ein Wort, mit dem der relative Rang eines Dokuments in den Suchergebnissen beeinflusst werden kann. Bei der Abfrageverarbeitung erhält ein Dokument, das ein Boostwort enthält, möglicherweise einen höheren oder niedrigeren Rang, je nachdem welche Bewertung für das Wort vordefiniert wurde.

Crawler
Ein Softwareprogramm, das Dokumente aus Datenquellen abruft und Informationen zusammenstellt, mit denen Suchindizes erstellt werden können.

Crawlerbereich (Crawl Space)
Eine bestimmten Mustern (wie z. B. URL-Adressen (Uniform Resource Locators), Datenbanknamen, Dateisystempfaden, Domänennamen und IP-Adressen) entsprechende Gruppe von Quellen, die ein Crawler liest, um Elemente zum Indexieren abzurufen.

Datenquelle (Data Source)
Jedes Datenrepository, aus dem Dokumente abgerufen werden können, z. B. das Internet, relationale und nicht relationale Datenbanken sowie Content-Management-Systeme.

Datenquellentyp (Data Source Type)
Eine Zusammenfassung von Datenquellen nach dem Protokoll, mit dem auf die Daten zugegriffen wird.

Datenspeicher (Data Store)
Eine Datenstruktur, in der Dokumente in syntaktisch analysierter Form gespeichert werden.

Definierter Name (Distinguished Name)
Der Name, der einen Eintrag in einem Verzeichnis eindeutig identifiziert. Ein definierter Name besteht aus durch Kommata getrennten Attribut-Wert-Paaren. Außerdem eine Gruppe von Name-Wert-Paaren (z. B. CN=Name der Person und C=Land oder Region), die eine Entität in einem digitalen Zertifikat eindeutig identifizieren.

Deltaindexerstellung (Delta Index Build)
Das Hinzufügen neuer Informationen zu einem vorhandenen Index in einem System für die Unternehmenssuche. Gegensatz zu Hauptindexerstellung.

Diakritisches Zeichen (Diacritic)
Eine Markierung, die eine Änderung im phonetischen Wert eines Zeichens oder einer Zeichenkombination angibt.

Dokumentobjektmodell (Document Object Model)
Ein System, bei dem ein gegliedertes Dokument (z. B. eine XML-Datei) in Form einer Baumstruktur mit Objekten angezeigt wird, auf die über das Programm zugegriffen werden kann und die aktualisierbar sind.

Domino Document Manager-Aktenschrankdatei (Domino Document Manager Cabinet)
Eine Domino Document Manager-Datenbank, die zum Organisieren von Dokumenten verwendet wird. Aktenschrankdateien enthalten Domino-Datenbanken.

Domino Document Manager-Bibliothek (Domino Document Manager Library)
Eine Domino Document Manager-Datenbank, die den Einstiegspunkt in Domino Document Manager bildet.

Domino Internet Inter-ORB Protocol (DIIOP)
Eine Server-Task, die auf dem Server ausgeführt wird und mit dem Domino-Object-Request-Broker zusammenarbeitet, um eine Kommunikation zwischen Java-Applets, die mit Notes-Java-Klassen erstellt werden, und dem Domino-Server zu ermöglichen. Browserbenutzer und Domino-Server führen die Kommunikation und den Austausch von Objektdaten über DIIOP aus.

Dynamische Rangfolge (Dynamic Ranking)
Ein Rangfolgetyp, bei dem die Begriffe in der Abfrage in Hinblick auf die durchsuchten Dokumente analysiert werden, um die Rangfolge der Ergebnisse zu ermitteln. Siehe auch textbasierte Bewertung. Vergleiche statische Rangfolge.

Dynamische Zusammenfassung (Dynamic Summarization)
Eine Art der Zusammenfassung, bei der die Suchbegriffe hervorgehoben werden und die Suchergebnisse Ausdrücke enthalten, die die Konzepte des gesuchten Dokuments am besten darstellen. Vergleiche statische Zusammenfassung.

Escapezeichen (Escape Character)
Ein Zeichen, das eine spezielle Bedeutung für mindestens ein nachfolgendes Zeichen unterdrückt oder auswählt.

Externe Datenquellen (External Data Source)
Eine Datenquelle für die Föderation, die nicht von OmniFind Enterprise Edition durchsucht, syntaktisch analysiert oder indexiert wird. Das Durchsuchen von externen Datenquellen wird an die Anwendungsprogrammierschnittstelle für die Abfrage dieser Datenquellen delegiert.

Feld (Field)
Ein Bereich, in den eine bestimmte Kategorie von Daten oder Steuerinformationen eingegeben wird.

Feldspezifische Suche (Fielded Search)
Eine auf ein bestimmtes Feld beschränkte Abfrage.

Ferner Föderator (Remote Federator)
Ein Serverföderator, der eine Föderation für eine Gruppe durchsuchbarer Objekte ausführt.

Föderation (Federation)
Der Prozess des Kombinierens von Benennungssystemen, wodurch es dem

zusammengefassten System ermöglicht wird, zusammengesetzte Namen zu verarbeiten, die alle Benennungssysteme umfassen.

Föderierte Suche (Federated Search)
Eine Suchfunktionalität, die das Durchsuchen mehrerer Suchservices ermöglicht und eine konsolidierte Liste mit Suchergebnissen zurückgibt.

Freiformatsuche (Free Text Search)
Eine Suche, in der der Suchbegriff als unformatierter Text dargestellt wird.

Hauptindexerstellung (Main Index Build)
Der Prozess des Erstellens des gesamten Index bei der Unternehmenssuche. Gegensatz zu Deltaindexerstellung.

Hybridsuche (Hybrid Search)
Eine Kombination aus Boolescher Suche und Freiformatsuche.

Identitätsmanagement (Identity Management)
Eine Gruppe von Anwendungsprogrammierschnittstellen für die Unternehmenssuche, die den Zugriff auf sichere Daten steuern und es Benutzern ermöglichen, eine Objektgruppe zu durchsuchen, ohne eine Benutzer-ID und ein Kennwort für jedes Repository in der Objektgruppe angeben zu müssen.

Index Siehe Volltextindex.

Indexierungswarteschlange (Index Queue)
Eine Liste von zu verarbeitenden Anforderungen für die Haupt- und Deltaindexerstellung.

Informationsextraktion (Information Extraction)
Eine Art der Begriffsextraktion, bei der signifikante Vokabularelemente (z. B. Namen, Begriffe und Ausdrücke) in Textdokumenten automatisch erkannt werden.

In Warteschlange stellen (Enqueue)
Eine Nachricht oder einen Eintrag in eine Warteschlange einfügen.

IP-Adresse (IP Address)
Eine eindeutige Adresse für ein Gerät oder eine logische Einheit in einem Netz, das den IP-Standard verwendet.

Java Database Connectivity (JDBC)
Ein Industriestandard für datenbankunabhängige Konnektivität zwischen der Java-Plattform und einer großen Reihe von Datenbanken. Die JDBC-Schnittstelle bietet eine API auf Aufrufebene für SQL-Datenbankzugriffe.

JavaScript
Eine Web-Scripting-Sprache, die in Browsern und auf Web-Servern verwendet wird.

JavaServer Pages (JSP)
Eine Servertechnologie zur Scripterstellung, die es ermöglicht, Java-Code dynamisch in Webseiten (HTML-Dateien) einzubetten und diesen auszuführen, wenn die Seite bereitgestellt wird, um einem Client dynamischen Inhalt zurückzugeben.

Java Virtual Machine (JVM)
Softwareimplementierung eines Prozessors, die kompilierten Java-Code (Applets und Anwendungen) ausführt.

Katakana
Ein Zeichensatz aus Symbolen, die in einem der beiden gebräuchlichen

phonetischen Alphabete der japanischen Sprache verwendet werden. Dieser dient in erster Linie zum phonetischen Schreiben von Fremdwörtern.

Kategoriebaum (Category Tree)
Eine Hierarchie von Kategorien.

Klitik (Clitic)
Ein Wort, das syntaktisch eigenständig ist, phonetisch aber mit einem anderen Wort zusammenhängt. Ein Klitik kann mit dem Wort, an das es angelehnt ist, zusammengeschrieben oder davon getrennt geschrieben werden. Typische Beispiele für in der englischen Sprache vorkommende Klitika sind der hintere Teil einer Zusammenfügung (*would**n't*** oder *you'**re***).

Komponentenpfad (Feature Path)
Ein Pfad, über den auf den Wert eines Merkmals in einer UIMA-Merkmalstruktur (Unstructured Information Management Architecture) zugegriffen wird.

Lemma
Die Grundform eines Worts. Lemmata spielen vor allem in stark flektierten Sprachen wie dem Tschechischen eine wichtige Rolle.

Lexikalische Affinität (Lexical Affinity)
Die Beziehung von Suchbegriffen in einem Dokument, die in ihrer Bedeutung in engem Zusammenhang stehen. Mit der lexikalischen Affinität wird die Relevanz eines Ergebnisses berechnet.

Ligatur (Ligature)
Mindestens zwei Zeichen, die so miteinander verbunden werden, dass sie ein einzelnes Zeichen bilden. Bei ff und ffi handelt es sich beispielsweise um Zeichen, die als Ligaturen dargestellt werden können.

Lightweight Directory Access Protocol (LDAP)
Ein offenes Protokoll, das mithilfe von TCP/IP Zugriff auf Verzeichnisse ermöglicht, die ein X.500-Modell unterstützen, und das nicht die Ressourcenanforderungen des komplexeren X.500 Directory Access Protocol (DAP) aufweist. LDAP kann beispielsweise verwendet werden, um Personen, Organisationen und andere Ressourcen in einem Internet- oder Intranetverzeichnis zu suchen.

Linguistische Suche (Linguistic Search)
Eine Art der Suche, bei der ein Dokument mit auf ihre Grundformen reduzierten Begriffen durchsucht, abgerufen und indexiert (Beispiel: *Mäuse* wird als *Maus* indexiert) oder mit ihrer Grundform erweitert wird (wie bei zusammengesetzten Wörtern).

Linkanalyse (Link Analysis)
Ein Verfahren, das auf der Analyse von Hyperlinks zwischen Dokumenten basiert und mit dem festgestellt wird, welche Seiten in der Objektgruppe für Benutzer von Bedeutung sind.

Lokaler Föderator (Local Federator)
Ein Clientobjekt in einer Anwendung für die Unternehmenssuche, das über die SIAPIs (Search and Index APIs) erstellt wird und es Benutzern ermöglicht, eine Gruppe von heterogenen Objektgruppen zu durchsuchen und eine einheitliche Gruppe von Suchergebnissen zu erhalten.

Lotus QuickPlace-Bereich (Lotus QuickPlace Place)
Ein von Lotus QuickPlace bereitgestellter Arbeitsbereich im Web, der geographisch weit verteilten Teilnehmern die Möglichkeit bietet, zusammen an

Projekten zu arbeiten und online in einem strukturierten und sicheren Arbeitsbereich miteinander zu kommunizieren.

Lotus QuickPlace-Raum (Lotus QuickPlace Room)
Ein partitionierter Bereich in einem Lotus QuickPlace-Bereich, der ausschließlich berechtigten Mitgliedern vorbehalten ist, die eine gemeinsame Aufgabe und die Notwendigkeit zur Zusammenarbeit verbindet.

Merkmalstruktur (Feature Structure)
Die zugrunde liegende Datenstruktur, die dem Ergebnis der Textanalyse entspricht. Die Merkmalstruktur hat eine Attribut-Wert-Struktur. Jede Merkmalstruktur ist von einem bestimmten Typ, wobei jeder Typ, ähnlich wie eine Java-Klasse, über eine angegebene Gruppe gültiger Merkmale oder Attribute verfügt.

MIME-Typ (MIME Type)
Ein Internetstandard zur Angabe des Typs eines Objekts, das über das Internet übertragen wird.

N-Gram-Segmentierung (N-Gram Segmentation)
Eine Analysemethode, bei der nicht wie bei der Unicode-basierten Segmentierung mit Leerzeichen Wörter durch eine Leerstelle begrenzt sind, sondern sich überlappende Folgen einer bestimmten Anzahl Zeichen als ein Wort betrachtet werden.

No-Follow-Anweisung (No-Follow Directive)
Eine Anweisung auf einer Webseite, die Roboter (z. B. den Web-Crawler) anweisen, den Links auf dieser Seite nicht zu folgen.

No-Index-Anweisung (No-Index Directive)
Eine Anweisung auf einer Webseite, die Roboter (z. B. den Web-Crawler) anweisen, den Inhalt dieser Seite nicht in den Index einzuschließen.

Notes Remote Procedure Call (NRPC)
Ein für die gesamte Notes-zu-Notes-Kommunikation verwendeter Kommunikationsmechanismus von Lotus Notes.

Objektgruppe (Collection)
Eine Gruppe von Datenquellen und Optionen für die Crawlersuche, die Syntaxanalyse, das Indexieren und das Durchsuchen dieser Datenquellen.

Parametrische Suche (Parametric Search)
Eine Art der Suche, bei der Objekte gesucht werden, die einen numerischen Wert oder ein numerisches Attribut (wie z. B. Datumsangaben, ganze Zahlen oder andere numerische Datentypen in einem angegebenen Bereich) enthalten.

Parser Ein Programm, das Dokumente interpretiert, die dem Datenspeicher für die Unternehmenssuche hinzugefügt werden. Der Parser extrahiert Informationen aus den Dokumenten und bereitet sie für Indexierungs-, Such- und Abrufvorgänge vor.

Parser-Service (Parser Service)
Der Service für die Unternehmenssuche, der die gesamte Syntaxanalyse für Dokumente und die Verarbeitung der Textanalysen in Dokumentobjektgruppen handhabt. Zu jedem Zeitpunkt wird mindestens ein Parser-Service ausgeführt.

Parsertreiber (Parser Driver)
Ein Service für die Unternehmenssuche, der dem Parser-Service Dokumente zuführt. Pro Objektgruppe gibt es einen Parsertreiber. Der Parser-

treiberservice einer Objektgruppe entspricht dem Parser der Objektgruppe in der Administrationskonsole für die Unternehmenssuche.

Platzhalterzeichen (Masking Character)
Ein Zeichen, das optionale Zeichen am Anfang, in der Mitte und am Ende eines Suchbegriffs darstellt. Mit Platzhalterzeichen werden normalerweise Varianten eines Begriffs in einem Index gesucht.

Platzhalterzeichen (Wildcard Character)
Ein Zeichen, das optionale Zeichen am Anfang, in der Mitte oder am Ende eines Suchbegriffs darstellt.

Popularitätsrangfolge (Popular Ranking)
Ein Rangfolgetyp, der die vorhandene Rangfolge eines Dokuments gemäß der Popularität des Dokuments erhöht.

Portal Document Manager (PDM)
Ermöglicht es Benutzern, ein einzelnes zentrales Dokumentrepository für elektronisches Teamwork zu verwenden. Administratoren haben die Möglichkeit, ihre Dokumente effizient zu verwalten und zu steuern, wie Benutzer mit Informationen interagieren.

Privater Anwender der allgemeinen Analysestruktur (Common Analysis Structure Consumer)
Ein privater Anwender führt die abschließende Verarbeitung der Analyseergebnisse durch, die in der allgemeinen Analysestruktur gespeichert sind. Beispielsweise indexiert ein privater Anwender den Inhalt der allgemeinen Analysestruktur in einer Suchmaschine oder füllt eine relationale Datenbank mit bestimmten Analyseergebnissen.

Protokoll zum Sperren von Websitebereichen für Robots (Robots Exclusion Protocol)
Ein Protokoll mit dem Website-Administratoren durchsuchende Robots anweisen können, welche Bereiche ihrer Site vom Robot nicht besucht werden soll.

Proxy-Server (Proxy Server)
Ein Server, der als Mittler für HTTP-Webanforderungen von einer Anwendung oder von einem Web-Server fungiert. Ein Proxy-Server wird als Ersatzsystem für die Content-Server im Unternehmen verwendet.

Quick Link
Eine Zuordnung zwischen einem Uniform-Resource-Identifier (URI) und Schlüsselwörtern oder Ausdrücken.

Rangfolge (Ranking)
Die Zuordnung eines ganzzahligen Werts zu jedem Dokument in den Suchergebnissen einer Abfrage. Die Reihenfolge der Dokumente in den Suchergebnissen basiert auf der Relevanz für die Abfrage. Eine höhere Einstufung in der Rangfolge bedeutet eine größere Übereinstimmung. Siehe auch dynamische Rangfolge und statische Rangfolge.

Raum (Room)
Ein Programm, mit dem Benutzer die Möglichkeit haben, von anderen Benutzern zu lesende Dokumente zu erstellen, auf Kommentare anderer Personen zu antworten und den Status sowie den Endtermin eines Projekts anzuzeigen. Außerdem können die Benutzer mit anderen im selben Raum befindlichen Personen chatten. Siehe auch Lotus QuickPlace-Raum.

Reduktion auf Grundform (Lemmatization)
Ein Prozess, der die Grundform und verschiedene grammatische Formen eines Worts angibt. Wenn Sie beispielsweise nach 'Maus' suchen, werden

auch Dokumente gefunden, die das Wort 'Mäuse' enthalten; bei einer Suche nach dem Wort 'gehen' werden auch Dokumente gefunden, die die Wörter 'geht', 'ging' oder 'gegangen' enthalten.

Regelbasierte Kategorie (Rule-Based Category)
Durch Regeln erstellte Kategorien, die angeben, welche Dokumente welchen Kategorien zugeordnet werden. Sie können beispielsweise Regeln definieren, mit denen Dokumente, die bestimmte Wörter enthalten oder nicht enthalten oder die einem URI-Muster (Uniform-Resource-Identifier) entsprechen, bestimmten Kategorien zugeordnet werden.

Rohdatenspeicher (Raw Data Store)
Eine Datenstruktur, in der durchsuchte Dokumente gespeichert werden, bevor Sie an den Parser gesendet werden. Crawler schreiben in den Rohdatenspeicher, und der Parser liest aus dem Rohdatenspeicher. Wenn Dokumente syntaktisch analysiert wurden, werden sie aus dem Rohdatenspeicher entfernt. Der Rohdatenspeicher darf nicht mit dem Datenspeicher verwechselt werden.

Schlüsseldatenbankdatei (Key Database File)
Siehe Schlüsselring.

Schlüsselring (Key Ring)
Im Bereich der IT-Sicherheit eine Datei, die öffentliche Schlüssel, private Schlüssel, Trusted Roots und Zertifikate enthält. Siehe auch Schlüsselspeicherdatei.

Schlüsselspeicherdatei (Keystore File)
Ein Schlüsselring mit öffentlichen Schlüsseln, die als Unterzeichnerzertifikate gespeichert werden, und mit privaten Schlüsseln, die in persönlichen Zertifikaten gespeichert werden.

Secure Sockets Layer (SSL)
Ein Sicherheitsprotokoll zur Gewährleistung von Datenschutz bei der Kommunikation. Mit SSL können Client-/Serveranwendungen in einer Weise kommunizieren, die das Ausspionieren, die Manipulation von Daten während der Übertragung und das Fälschen von Nachrichten verhindert.

Segmentierung (Segmentation)
Die Unterteilung von Text in bestimmte lexikalische Einheiten. Die nicht auf Wörterbüchern basierende Verarbeitung schließt Leerzeichen und N-Gram-Segmentierung ein, während die auf Wörterbüchern basierende Unterstützung die Wort-, Satz- und Absatzsegmentierung sowie eine Reduktion auf die Grundform umfasst.

Seite mit detaillierten Fehlerhinweisen (Soft Error Page)
Ein Typ einer Webseite, die Informationen dazu bereitstellt, weshalb die angeforderte Webseite nicht zurückgegeben werden kann. Beispielsweise kann der HTTP-Server statt eines einfachen Statuscodes eine Seite zurückgeben, auf der der Statuscode detailliert beschrieben wird.

Seite mit Einstiegspunktliste (Seed List Page)
Eine XML-Seite in WebSphere Portal, die Links zu den in einem Portal verfügbaren Seiten enthält. Crawler verwenden die Einstiegspunktliste, um die zu durchsuchenden Dokumente zu identifizieren. Die Seite mit der Einstiegspunktliste enthält außerdem Metadaten, die mit den durchsuchten Dokumenten im Index für die Unternehmenssuche gespeichert werden.

Semantische Suche (Semantic Search)
Ein Typ der Schlüsselwortsuche, der die linguistische Analyse und die Kontextanalyse umfasst. Siehe auch Textanalyse.

Servlet
Ein Java-Programm, das auf einem Web-Server ausgeführt wird und die Funktionalität des Servers erweitert, indem es aufgrund von Web-Client-Anforderungen dynamischen Inhalt generiert. Servlets werden gewöhnlich verwendet, für Datenbanken eine Verbindung zum Web herzustellen.

Shingle
Eine Zeichenfolge mit fortlaufenden Token (Wörter), die aus einem Satz stammen. Beispielsweise können dem Satz ″Dies ist ein sehr kurzer Satz.″ die folgenden 3-Wort-Shingles (bzw. Trigramme) entnommen werden:

Dies ist ein
ist ein sehr
ein sehr kurzer
sehr kurzer Satz

Shingles können in der statistischen Liguistik verwendet werden. Wenn beispielsweise zwei verschiedene Text viele gemeinsame Shingles aufweisen, besteht wahrscheinlich ein gewisser Zusammenhang zwischen den Texten.

Sicherheitstoken (Security Token)
Informationen zu Identität und Sicherheit, mit denen der Zugriff auf Dokumente in einer Objektgruppe berechtigt wird. Verschiedene Datenquellentypen unterstützen verschiedene Sicherheitstokentypen. Beispiele: Benutzerrollen, Benutzer-IDs, Gruppen-IDs und andere Informationen für die Datenzugriffssteuerung.

Spracherkennung (Language Identification)
Eine Suchfunktion der Unternehmenssuche, die die Sprache eines Dokuments bestimmt.

Stammbildung (Stemming)
Siehe Wortstammbildung.

Start-URL (Start Uniform Resource Locator)
Der Ausgangspunkt einer Crawlersuche.

Statische Rangfolge (Static Ranking)
Ein Rangfolgetyp, bei dem Faktoren der eingestuften Dokumente (z. B. das Datum, die Anzahl der Links, die auf das Dokument verweisen, usw.) den Rang erhöhen. Vergleiche dynamische Rangfolge.

Statische Zusammenfassung (Static Summarization)
Ein Zusammenfassungstyp, bei dem die Suchergebnisse eine angegebene, gespeicherte Zusammenfassung aus dem Dokument enthalten. Vergleiche dynamische Zusammenfassung.

Steuerkomponente für Textanalyse (Text Analysis Engine)
Eine Softwarekomponente, deren Aufgabe es ist, Kontext und semantischen Inhalt in Text aufzufinden und darzustellen.

Stoppwortentfernung (Stop Word Removal)
Das Entfernen von Stoppwörtern aus der Abfrage, damit allgemeine Wörter ignoriert und auf diese Weise gezieltere Ergebnisse zurückgegeben werden.

Stoppwort (Stop Word)
Ein häufig verwendetes Wort (z. B. *der*, *ein*, *und*), das von einer Suchanwendung ignoriert wird.

Suchanwendung (Search Application)
Ein Programm bei der Unternehmenssuche, das Abfragen verarbeitet, den Index durchsucht, die Suchergebnisse zurückgibt und die Quellendokumente abruft.

Suchcache (Search Cache)
Ein Puffer, der die Daten und Ergebnisse vorangegangener Suchanforderungen enthält.

Suche mit Begriffsgewichtung (Weighted Term Search)
Eine Abfrage, bei der bestimmten Begriffen größere Bedeutung beigemessen wird.

Suche nach grober Übereinstimmung (Fuzzy Search)
Eine Suche, bei der Wörter zurückgegeben werden, deren Schreibweise der des Suchbegriffs ähnlich ist.

Suchergebnisse (Search Results)
Eine Liste der Dokumente, die der Suchanforderung entsprechen.

Suchindexdateien (Search Index Files)
Gruppe von Dateien, in der ein Index in der Suchmaschine gespeichert wird.

Suchmaschine (Search Engine)
Ein Programm, das eine Suchanforderung annimmt und eine Dokumentenliste an den Benutzer zurückgibt.

Synonymverzeichnis (Synonym Dictionary)
Ein Wörterverzeichnis, das es Benutzern ermöglicht, eine Objektgruppe auch nach Synonymen ihrer Abfragebegriffe zu durchsuchen.

Sprache XPath (XML Path) (XML Path Language (XPath))
Eine Sprache, die konzipiert wurde, um Teile der XML-Quellendaten für die Verwendung mit XML-Technologien wie z. B. XSLT-, XQuery- und XML-Parsern eindeutig anzugeben oder zu adressieren. Die Sprache XPath ist ein Standard des World Wide Web Consortium.

Taxonomie (Taxonomy)
Eine auf Ähnlichkeiten basierende Klassifikation von Objekten zu Gruppen. In der Unternehmenssuche fasst eine Taxonomie Daten zu Kategorien und Unterkategorien zusammen. Siehe auch Kategoriebaum.

Textanalyse (Text Analysis)
Das Extrahieren der Semantik und anderer Informationen aus Text, um die Abrufbarkeit von Daten in einer Objektgruppe zu verbessern. Siehe auch Semantische Suche.

Textbasierte Bewertung (Text-Based Scoring)
Die Zuordnung eines ganzzahligen Werts zu einem Dokument, der die Relevanz des Dokuments in Bezug auf die Abfragebegriffe anzeigt. Ein hoher ganzzahliger Wert zeigt eine große Übereinstimmung mit der Abfrage an. Siehe auch dynamische Rangfolge.

Text mit freiem Format (Free-Form Text)
Unstrukturierter Text, der aus Worten oder Sätzen besteht.

Textsegmentierung (Text Segmentation)
Siehe Segmentierung.

Themenextraktion (Theme Extraction)
Eine Art der Begriffsextraktion, bei der signifikante Vokabularelemente in

Textdokumenten automatisch erkannt werden, um das Thema eines Dokuments zu extrahieren. Siehe auch Begriffsextraktion.

Token Die Basistexteinheiten, die von der Unternehmenssuche indexiert werden. Token können aus den Wörtern einer Sprache oder aus anderen Texteinheiten bestehen, die sich für das Indexieren eignen.

Tokenizer
Ein Textsegmentierungsprogramm, das Text überprüft und ermittelt, wann und ob eine Zeichenfolge als Token erkannt werden kann.

Typsystem (Type System)
Das Typsystem definiert die Objekttypen (Komponentenstrukturen), die eine Textanalysesteuerkomponente in einem Dokument erkennen kann. Das Typsystem definiert alle möglichen Komponentenstrukturen in Form von Typen und Komponenten. Sie können eine beliebige Anzahl unterschiedlicher Typen in einem Typsystem definieren. Ein Typsystem ist domänen- und anwendungsspezifisch.

Überwachungsbeauftragter (Monitor)
Ein Benutzer der Unternehmenssuche, der über die Berechtigung zum Beobachten von Prozessen auf Objektgruppenebene verfügt.

Unicode-basierte Segmentierung mit Leerzeichen (Unicode-Based White Space Segmentation)
Ein Aufbereitungsverfahren, bei dem mittels Unicode-Zeichenmerkmalen zwischen Token und Trennzeichen unterschieden wird.

Uniform-Resource-Identifier (URI)
Eine kompakte Zeichenfolge, die eine abstrakte oder physische Ressource angibt.

Uniform-Resource-Locator (URL)
Die eindeutige Adresse einer Informationsressource, auf die in einem Netz wie dem Internet zugegriffen werden kann. Die URL enthält den abgekürzten Namen des Protokolls, mit dem auf die Informationsquelle zugegriffen wird, sowie die Informationen, mit denen das Protokoll die Informationsquelle lokalisiert.

Unstructured Information Management Architecture (UIMA)
Eine IBM Architektur, die ein Framework zur Implementierung von Systemen zur Analyse unstrukturierter Daten definiert.

Verknüpfte Suche (Proximity Search)
Eine Textsuche, die ein Ergebnis zurückgibt, wenn mindestens zwei übereinstimmende Begriffe in einem bestimmten Abstand voneinander auftreten, z. B. im selben Satz oder Absatz.

Verwaltungsrolle (Administrative Role)
Eine Klassifizierung eines Benutzers, die die Zugriffsberechtigungen eines Benutzers festlegt.

Volltextindex (Full-Text Index)
Eine Datenstruktur, die auf Datenelemente verweist, damit eine Suche Dokumente finden kann, die die Abfragebegriffe enthalten.

Web-Crawler (Web Crawler)
Ein Crawlertyp, der das World Wide Web durchsucht, indem er ein Webdokument abruft und den Links in diesem Dokument folgt.

Wortstammbildung (Word Stemming)
Ein Prozess der linguistischen Normalisierung, in dem die Varianten eines

Worts auf eine allgemeine Form reduziert werden. Beispielsweise werden Wörter wie *Speicherung*, *speichernd* und *gespeichert* zu *speich-* reduziert.

Zeichennormalisierung (Character Normalization)
Ein Prozess, bei dem die Varianten eines Zeichens (z. B. Großschreibung und diakritische Zeichen) auf eine gemeinsame Form reduziert werden.

Zeilenvorschubzeichen (Newline Character)
Ein Steuerzeichen, das bewirkt, dass die Druck- oder Anzeigeposition eine Zeile nach unten versetzt wird.

Zertifikat (Certificate)
Im Bereich der IT-Sicherheit ein digitales Dokument, mit dem der Identität des Zertifikatinhabers ein öffentlicher Schlüssel angefügt wird, um eine Authentifizierung des Zertifikatinhabers zu ermöglichen. Zertifikate werden von einer Zertifizierungsstelle ausgestellt und digital signiert.

Zertifizierungsstelle (Certificate Authority)
Ein anerkanntes Fremdunternehmen, das digitale Zertifikate ausstellt, die verwendet werden, um digitale Signaturen und öffentlich/private Schlüsselpaare zu erstellen. Die Zertifizierungsstelle gewährleistet die Identität der Einzelpersonen, denen das eindeutige Zertifikat erteilt wird.

Zugriffssteuerungsliste (ACL - Access Control List)
Im Bereich der IT-Sicherheit eine Liste, die einem Objekt zugeordnet ist und in der alle Personen, die auf das Objekt zugreifen können, und deren Zugriffsberechtigungen angegeben werden.

Zusammenfassung (Summarization)
Das Einfügen von nicht redundanten Sätzen in Suchergebnissen, die den Inhalt eines Dokuments kurz beschreiben. Siehe auch dynamische Zusammenfassung und statische Zusammenfassung.

Bemerkungen und Marken

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. An Stelle der IBM Produkte, Programme oder Services können auch andere ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremdservices liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing
IBM Europe, Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Defense
France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekannt gegeben. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter dienen lediglich als Benutzerinformationen und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
USA

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Dokument aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Die oben genannten Erklärungen bezüglich der Produktstrategien und Absichtserklärungen von IBM stellen die gegenwärtige Absicht der IBM dar, unterliegen Änderungen oder können zurückgenommen werden, und repräsentieren nur die Ziele der IBM.

Alle von IBM angegebenen Preise sind empfohlene Richtpreise und können jederzeit ohne weitere Mitteilung geändert werden. Händlerpreise können u. U. von den hier genannten Preisen abweichen.

Diese Veröffentlichung dient nur zu Planungszwecken. Die in dieser Veröffentlichung enthaltenen Informationen können geändert werden, bevor die beschriebenen Produkte verfügbar sind.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogrammes illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Musteranwendungsprogramme, die in Quellensprache geschrieben sind. Sie dürfen diese Musterprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungs-

programme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten.

Kopien oder Teile der Musterprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Musterprogrammen der IBM Corp. abgeleitet. © Copyright IBM Corp. _Jahr/Jahre angeben_. Alle Rechte vorbehalten.

Für Teile des vorliegenden Produkts gilt Folgendes:

- Oracle® Outside In Content Access, Copyright © 1992, 2008, Oracle. Alle Rechte vorbehalten.
- IBM XSLT-Prozessor Lizensiertes Material - Eigentum der IBM © Copyright IBM Corporation, 1999-2008. Alle Rechte vorbehalten.

Marken

Informationen zu IBM Marken finden Sie unter http://www.ibm.com/legal/copytrade.shtml.

Die folgenden Namen sind Marken oder eingetragene Marken anderer Unternehmen:

Adobe, Acrobat, Portable Document Format (PDF), PostScript und alle auf Adobe basierenden Marken sind Marken oder eingetragene Marken der Adobe Systems Incorporated in den USA und/oder anderen Ländern.

Intel, das Intel-Logo, Intel Inside, das Intel Inside-Logo, Intel Centrino, das Intel Centrino-Logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium und Pentium sind Marken oder eingetragene Marken der Intel Corporation oder deren Tochtergesellschaften in den USA oder anderen Ländern.

Java und alle auf Java basierenden Marken und Logos sind Marken von Sun Microsystems, Inc. in den USA und/oder anderen Ländern.

Linux ist eine Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

Weitere Unternehmens-, Produkt- oder Servicenamen können Marken anderer Hersteller sein.

Index

A

Analyse auf der Basis von Wörterverzeichnissen 83
Analyseergebnisse in einer JDBC-fähigen Datenbank zuordnen
  Beschreibung 49
  Schritte 50
Annotator für reguläre Ausdrücke
  Aktivieren der einfachen semantischen Suche 91
  Annotatordeskriptor 99
  anpassen 97
  Beschreibung 89
  Beschreibung des XML-Regelsatzes 93
  Definieren von Regeln für reguläre Ausdrücke 94
  einfache semantische Suche 90
  Protokollierung 102

B

Benutzerdefinierte Analyse
  Analyseergebnisse in einer JDBC-fähigen Datenbank zuordnen 49, 50, 52, 57
  Beispielbeschreibung, Typsystem 26
  Methoden für das Indexieren benutzerdefinierter Analyseergebnisse 42
  Methoden für die Verwendung von XML-Markup-Formatierung in Analyse und Suche 29
  Textanalysealgorithmen 5
  Typsystembeschreibung 16
  Wechsel vom Basismodus in den erweiterten Analysemodus 17
  Workflow 6

D

DIC-Dateien
  benutzerdefinierte Stoppwörter 73
  Boostwörter 77
  Synonyme 69
Dokumentation
  HTML 105
  PDF 105
  suchen 105

E

Einfache semantische Suche
  Annotator für reguläre Ausdrücke verwenden 90
esboostworddictbuilder.bat, Script 77
esboostworddictbuilder.sh, Script 77
esstopworddictbuilder.bat, Script 73
esstopworddictbuilder.sh, Script 73
essyndictbuilder.bat, Script 69
essyndictbuilder.sh, Script 69

F

Funktionen zur behindertengerechten Bedienung für dieses Produkt 107

H

HTML-Dokumentation für die Unternehmenssuche 105

I

Indexieren benutzerdefinierter Analyseergebnisse
  Beschreibung 42
  Erstellen der Datei für die Zuordnung der allgemeinen Analysestruktur zum Index 43

K

Klitika 83

L

Lemmata 83
Linguistische Unterstützung
  Beschreibung 1
  Klitika 83
  Lemmata 83
  N-Gram-Segmentierung 81
  N-Gram-Segmentierung von numerischen Zeichen 82
  nicht-wörterverzeichnisbasierte Segmentierung 81
  Okurigana-Varianten 85
  orthografische Varianten im Japanischen 85
  Reduktion auf die Grundform 83
  Segmentierung auf der Basis von Wörterverzeichnissen 83
  semantische Suche 64
  Spracherkennung 79
  Stoppwortentfernung 86
  systemdefinierte Typen und Komponenten 18
  Unicode-basierte Leerraumsegmentierung 81
  Unicode-Normalisierung 86
  unterstützte Sprachen 83
  Unterstützung vom System 79
  Wortsegmentierung im Japanischen 85
  Zeichennormalisierung 86

N

N-Gram-Segmentierung
  Beschreibung 81
  normal 82
  numerisch 82
  vollständig 82
Nicht-wörterverzeichnisbasierte Analyse 81
Nicht-wörterverzeichnisbasierte Segmentierung 81

O

Okurigana-Varianten 85
Orthografische Varianten im Japanischen 85

P

PDF-Dokumentation für die Unternehmenssuche 105

R

Reduktion auf die Grundform 83

S

Scripts
  esboostworddictbuilder 77
  esstopworddictbuilder 73
  essyndictbuilder 69
Segmentierung
  auf der Basis von Wörterverzeichnissen 83
  nicht-wörterverzeichnisbasiert 81
  Unicode-basierte Segmentierung mit Leerzeichen 81
Segmentierung auf der Basis von Wörterverzeichnissen 83
Semantische Suche
  Abrufen von Teilen eines Dokuments, die mit einer Abfrage übereinstimmen 61
  Beschreibung 64
  semantische Suchabfrage 64
Spracherkennung 79
Stoppwortentfernung 86
Stoppwörter 86
Suchanwendungen
  Synonymunterstützung 67
  Unterstützung von Boostwörtern 75
  Unterstützung von Stoppwörtern 71
Suchserver
  Synonymverzeichnisse erstellen 69
  Verzeichnis von Boostwörtern erstellen 77
  Verzeichnis von Stoppwörtern erstellen 73

Suchserver *(Forts.)*
XML-Datei mit Synonymen 68
XML-Dateien mit Boostwörtern 76
XML-Dateien mit Stoppwörtern 72
Synonymverzeichnisse
DIC-Datei erstellen 69
Unterstützung in Suchanwendungen 67
XML-Datei erstellen 68

U

UIMA
Annotator für reguläre Ausdrücke verwenden 13
Anzeigen der Ergebnisse des Basisannotators und der angepassten Textanalyse 14
Basisannotatoren für die Unternehmenssuche ausführen 8
Basisannotatoren für die Unternehmenssuche installieren 8
Basiskonzepte 4
Beschreibung 3
Unterstützung benutzerdefinierter Textanalyse 3
Verwenden der allgemeinen Analysestruktur für private Datenbankanwender 11
UIMA-Typen, XML-Dokumentstrukturen zuordnen
Beschreibung 29
Erstellen der Datei für die Zuordnung von XML-Elementen zur allgemeinen Analysestruktur 31
Unicode-basierte Segmentierung mit Leerzeichen 81
Unicode-Normalisierung 86
Unterstützte Sprachen
Spracherkennung 79
Verarbeitung auf linguistischer Basis auf der Basis von Wörterverzeichnissen 83

V

Verzeichnisse von Boostwörtern
DIC-Datei erstellen 77
Unterstützung in Suchanwendungen 75
XML-Datei erstellen 76
Verzeichnisse von Stoppwörtern
DIC-Datei erstellen 73
Unterstützung in Suchanwendungen 71
XML-Datei erstellen 72

W

Wortsegmentierung, Japanisch 85

Z

Zeichennormalisierung 86
Zugriff auf Ergebnisse einer benutzerdefinierten Analyse
Definition eines Komponentenpfads 37
Filter 41
integrierte Komponenten 38
Zugriff auf Ergebnisse einer Textanalyse
Definition eines privaten CAS-Anwenders 36
Zuordnen von Ergebnissen einer benutzerdefinierten Analyse in einer JDBC-fähigen Datenbank
Containertypen 57
Datei für die Zuordnung der allgemeinen Analysestruktur zur Datenbank 52
Verwenden von Ladedateigruppen 51
Zuordnung von Containertypen 57

SC12-3611-02