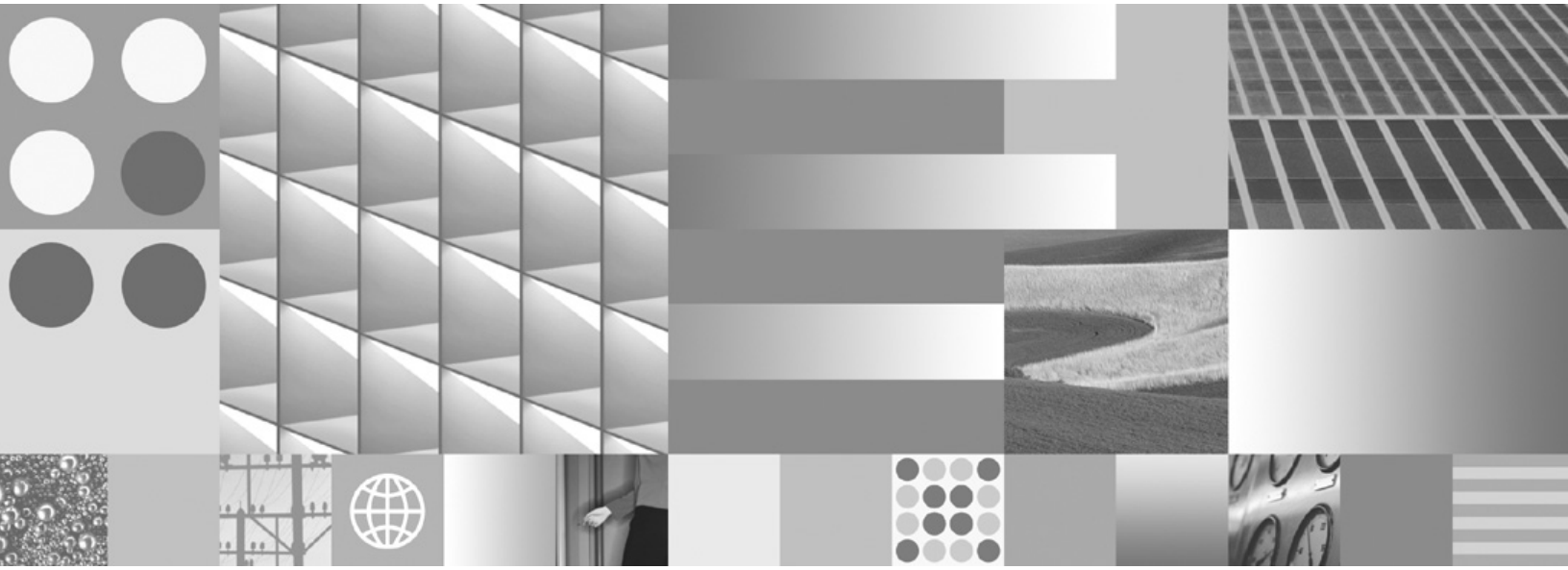
IBM OmniFind Enterprise Edition

Versie 8.4

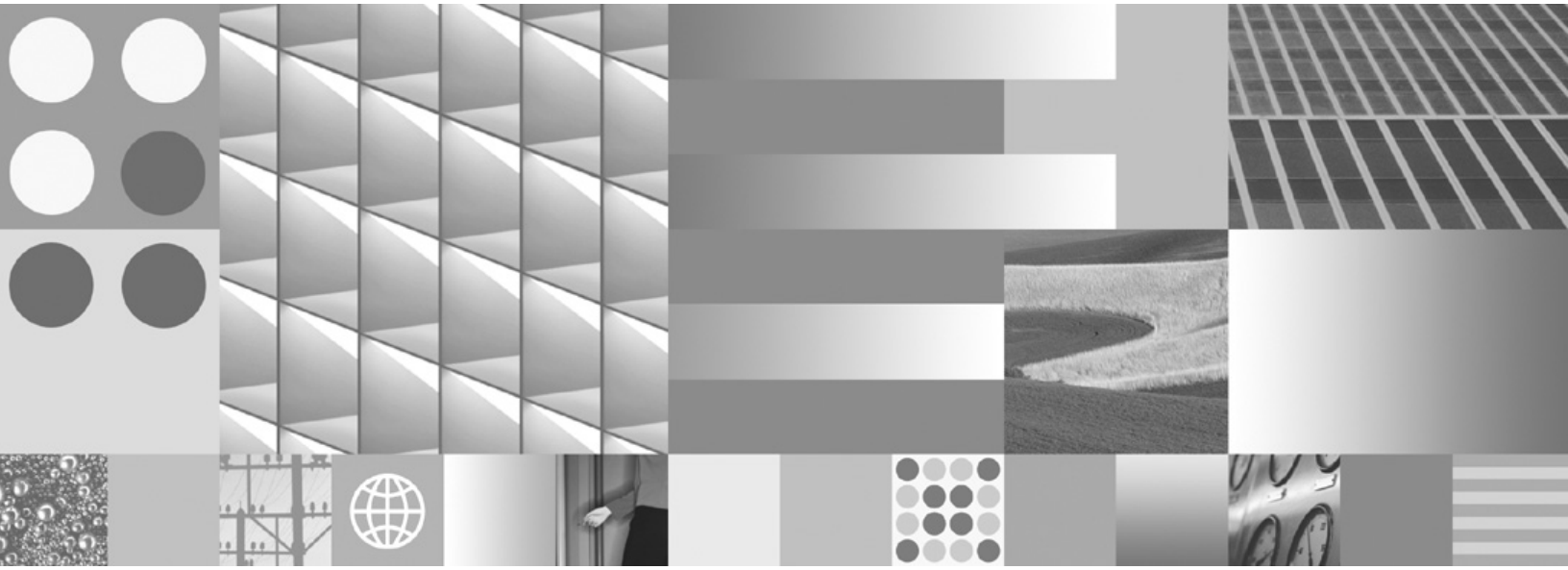


Integratie van tekstanalyses

SC18-9674-01

IBM OmniFind Enterprise Edition

Versie 8.4



Integratie van tekstanalyses

SC18-9674-01

Opmerking
Lees eerst de algemene informatie onder “Kennisgevingen en handelsmerken” op pagina 119.

Tweede uitgave (november 2006)

U kunt IBM-publicaties online of via uw plaatselijke IBM-vertegenwoordiger bestellen:

- Als u publicaties online wilt bestellen, gaat u naar het IBM Publications Center op www.ibm.com/shop/publications/order.
- Als u een IBM-vertegenwoordiger bij u in de buurt zoekt, gaat u naar de wereldwijde IBM-adressenlijst op www.ibm.com/planetwide.

Inhoudsopgave

Taalkundige ondersteuning voor semantische zoekopdrachten

Enterprise search biedt taalkundige ondersteuning tijdens zoekopdrachten voor tekstdocumenten in de meeste Indo-Europese en Aziatische talen, zoals Japans.

U kunt de taalkundige ondersteuning gebruiken om de kwaliteit van de zoekresultaten te verbeteren.

Taalkundige verwerking wordt uitgevoerd in twee stadia: wanneer een document wordt verwerkt om aan de index te kunnen worden toegevoegd en wanneer een gebruiker een zoekopdracht invoert.

Enterprise search bevat alleen een elementaire taalkundige functie die vereist is voor het bepalen van de taal van een invoerdocument en voor het segmenteren van de invoerstroom van het document in woorden of tokens.

Als u alleen basiszoekopdrachten gebruikt op basis van trefwoorden of alleen gebruikmaakt van native XML-zoekopdrachten waarin gebruik wordt gemaakt van de documentstructuur, hebt u voldoende aan de taalkundige verwerking in enterprise search.

De meeste informatie in tekstdocumenten is niet gestructureerd. Dit maakt het moeilijk om er effectief gebruik van te maken, want het is niet gemakkelijk om achter de betekenis van de informatie te komen.

Het zoeken naar trefwoorden is eenvoudig, maar dit is niet in alle gevallen voldoende, bijvoorbeeld als u verder wilt zoeken dan naar alleen woorden in het document, zoals in de volgende voorbeelden duidelijk wordt.

- Als u documenten gemeenschappelijk met andere personen gebruikt, worden de gegevens niet altijd expliciet aangegeven, bijvoorbeeld adressen of telefoonnummers in e-mails. Mogelijk wordt het woord *telefoonnummer* zelfs helemaal niet gebruikt. In plaats daarvan zou een e-mail een zin kunnen bevatten zoals ″U kunt mij bereiken op 020-5555555.″ De gebruiker weet vaak niet hoe de informatie waarnaar hij of zij op zoek is, in het document is vastgelegd en zou in het ideale geval het liefste een zoekopdracht opgeven als ″telefoonnummer Barbara″. Deze zoekopdracht levert echter niets op, want het woord *telefoonnummer* staat niet in het document.
- Op het gebied van ″competitive intelligence″ kunnen documenten informatie bevatten over concurrenten en de producten die zij leveren of de websites van concurrenten, waarbij de verschillen in de verkochte producten van de afgelopen drie maanden worden opgenomen. In dit geval zou de gebruiker een zoekopdracht opgeven als ″Smit & Zn. producten″ of ″Smit & Zn. producten nov. 2004 tot jan. 2005″. In de eerste zoekopdracht staat de term *producten* voor een product of groep producten, maar de query geeft geen overzicht van de producten die door Smit & Zn. worden geleverd, want er wordt gezocht naar de term *producten*. Datzelfde geldt voor de query waarin een bepaalde tijdsperiode wordt genoemd. Het is vrijwel onmogelijk om het zoeken met behulp van een trefwoordzoekopdracht te beperken tot een bepaalde tijdsperiode.
- In CRM (Customer Relationship Management) kunnen documenten bijvoorbeeld alle problemen aangeven die zijn opgetreden met de remmen van auto's in garages in het gebied rond Amsterdam. In de rapporten van de garages worden situaties beschreven zoals ″remblok vervangen als gevolg van lekkage hydrau-

lisch systeem″. Gebruikers die zoeken naar meer gedetailleerde informatie, zouden een zoekopdracht kunnen invoeren zoals ″remprobleem reparatie garages in Amsterdam-Noord″. Deze query levert echter waarschijnlijk geen rapporten op waarin sprake is van ″remblok vervangen als gevolg van lekkage hydraulisch systeem″ omdat de termen *remprobleem* en *garages* niet voorkomen in de rapporten. Deze rapporten vermelden mogelijk alleen de straat en postcode van de garages, niet het volledige adres met de vermelding Noord.

- Op het gebied van research kunnen documenten een bepaald medicijn beschrijven, en de relatie daarvan met ten minste één ziekte die in dezelfde alinea wordt vermeld. De gemiddelde gebruiker zou een zoekopdracht kunnen opgeven met alledaagse termen voor het medicijn, in de hoop een meer gedetailleerd verslag te vinden van de ziektes waartegen dat medicijn helpt, inclusief symptomen. Die zoekopdracht levert wellicht echter geen bevredigende resultaten op omdat de alledaagse term niet in de documenten wordt gebruikt en omdat in de documenten het woord *ziekte* helemaal niet wordt gebruikt, alleen de naam van de aandoening zelf.

Uit deze voorbeelden blijkt dat het zoeken naar de juiste termen in de enorme verzameling informatiebronnen van vandaag de dag een enorme uitdaging is die geavanceerde analyses vereisen waarvoor meer nodig is dan de op woordenboeken gebaseerde analyses met segmentering die enterprise search biedt. De meeste informatie die van belang is, wordt niet expliciet gemarkeerd in het oorspronkelijke document. In plaats daarvan moet de documentcontent worden geanalyseerd om de concepten van belang te kunnen herkennen en vinden, bijvoorbeeld benoemde entiteiten zoals personen, organisaties, locaties, faciliteiten en producten, en de mogelijke relatie tussen deze entiteiten.

De informatie die u in tekstdocumenten hoopt te ontdekken en extraheren, is afhankelijk van de gebruikers en van het domein. Als hulp bij het ontwerpen en ontwikkelen van uw eigen analysealgoritme is IBM uitgerust met de IBM Unstructured Information Management Architecture (UIMA), een architectuur- en softwareframework dat u helpt bij het bouwen van geavanceerde analysefuncties voor het zoeken naar interessante informatie in documentcollecties binnen enterprise search.

Verwante onderwerpen

“Integratie van aangepaste tekstanalyse” op pagina 3
Nadat u buiten enterprise search, met behulp van de Unstructured Information Management Architecture (UIMA) uw aangepaste analyseprogramma hebt gebouwd, kunt u de analyselogica in enterprise search integreren met behulp van de beheerconsole.

“Basisconcepten gebruikt bij verwerking van tekstanalyse” op pagina 4
Enkele van de basisbegrippen die bij tekstanalyse worden gebruikt, zijn annotators, analyseresultaten, featurestructuur, type, typesysteem, annotatie en Common Analysis Structure.

Integratie van aangepaste tekstanalyse

Nadat u buiten enterprise search, met behulp van de Unstructured Information Management Architecture (UIMA) uw aangepaste analyseprogramma hebt gebouwd, kunt u de analyselogica in enterprise search integreren met behulp van de beheerconsole.

UIMA is een open platform waarmee de componenten voor elke afzonderlijke analysefunctie kunnen worden geïdentificeerd. Dankzij dit platform kunnen deze componenten op een eenvoudige manier opnieuw worden gebruikt en worden gecombineerd.

Geavanceerde taalkundige analyses kan bestaan uit een combinatie van verschillende analysetaken. De analyse begint met de taaldetectie en -segmentering, gevolgd door herkenning van de woordsoort en het analyseren van de grammatica. De laatste stap bestaat bijvoorbeeld uit de identificatie van de relatie tussen bepaalde chemische stoffen en het voorkomen van bepaalde symptomen. Elke stap in het analyseproces hangt af van de resultaten van de vorige stap.

De analyselogica voor elke stap is ondergebracht in een *annotator*. Annotators worden gecombineerd tot een soort verwerkingsketen die elk document in de collectie aandoen om nieuwe informatie op te sporen en die informatie op te slaan voor verwerking verderop in het proces.

De annotators die verantwoordelijk zijn voor het ontdekken en beschrijven van de analysecontent in tekstdocumenten bevinden zich in een *analyseprogramma*, een centraal begrip in UIMA. Een analyseprogramma kan één annotator bevatten of kan zijn samengesteld uit verschillende programma's, die op hun beurt weer annotators bevatten.

UIMA bevat alleen de basisbouwstenen die nodig zijn om uw eigen analyseprogramma's te maken, te testen en te implementeren. Taalkundige analysefuncties in de vorm van vooraf geconfigureerde analyseprogramma's die u in de UIMA-omgeving kunt implementeren, zijn echter niet beschikbaar. Maar de taalkundige verwerking die in enterprise search wordt toegepast, is beschikbaar als een set annotators waarmee u kunt werken in UIMA.

Als u wilt werken met UIMA, moet u de UIMA Software Development Kit installeren. Deze kit vindt u via IBM developerWorks. Raadpleeg voor informatie de WebSphere Information Integrator-zone op http://www.ibm.com/developerworks/db2/zones/db2ii/. De UIMA Software Development Kit (SDK) bevat een Java-implementatie van het UIMA-framework voor de implementatie, beschrijving, samenstelling en het instellen van de UIMA-componenten.

Daarnaast vindt u in de UIMA SDK een set tools en hulpprogramma's voor het gebruik van UIMA in een ontwikkelomgeving op basis van Eclipse (Eclipse-plug-ins). Informatie over Eclipse vindt u op www.eclipse.org en in de documentatie van UIMA. Daar vindt u ook instructies voor het installeren van de UIMA Software Development Kit in de Eclipse Interactive Development Environment.

Verwante onderwerpen

Basisconcepten gebruikt bij verwerking van tekstanalyse

Enkele van de basisbegrippen die bij tekstanalyse worden gebruikt, zijn annotators, analyseresultaten, featurestructuur, type, typesysteem, annotatie en Common Analysis Structure.

Annotators bevatten de logica waarmee een document wordt geanalyseerd en waarmee beschrijvende gegevens over het document als geheel (de metagegevens van het document) en over onderdelen in het document worden ontdekt en vastgelegd. Deze beschrijvende gegevens worden de *analyseresultaten* genoemd. De analyseresultaten vormen een annotatie van elke aaneengesloten subreeks (ook wel spanne genoemd) van het tekstdocument. In het meest ideale geval komen de analyseresultaten overeen met de informatie die u zoekt.

Een *featurestructuur* is de onderliggende gegevensstructuur die een analyseresultaat vertegenwoordigt. Een featurestructuur is een structuur met kenmerken en de bijbehorende waarden. Elke featurestructuur heeft een bepaald *type* en elk type heeft een opgegeven set met geldige features of kenmerken (eigenschappen), vergelijkbaar met een Java-klasse. Features kunnen een reekstype bevatten waarmee het type waarde wordt aangegeven dat de feature moet hebben, zoals String.

Bijvoorbeeld, de tekstspanne ″Jan Jaap van Dam″ kan worden beschreven met een annotatie van het type `Person` met de features `personName`, `age`, `nationality` en `profession`.

Het *typesysteem* definieert de typen objecten (featurestructuren) die kunnen worden ontdekt in een document. Op basis van het typesysteem worden alle mogelijke featurestructuren gedefinieerd in de vorm van typen en features (kenmerken), vergelijkbaar met een klassehiërarchie in Java. U kunt in een typesysteem een willekeurig aantal verschillende typen definiëren. Een typesysteem is specifiek voor een bepaald domein en een bepaald programma.

Met de meeste analyseannotators worden de analyseresultaten in de vorm van *annotaties* gemaakt. Annotaties zijn een speciaal soort featurestructuur, speciaal ontworpen voor taalkundige analyseverwerking. Een annotatie omspant of dekt een bepaalde invoertekst en wordt gedefinieerd op basis van de begin- en eindpositie in de invoertekst.

Met een annotator waarmee bijvoorbeeld monetaire uitdrukkingen worden herkend, wordt voor de tekst ″100.55 US Dollars″ het toelichtingstype `monetaryExpression` gemaakt, dat de tekst dekt waarvoor de feature `currencySymbol` is ingesteld op ″$″.

Alle annotators in UIMA modelleren de gegevens in featurestructuren en slaan ze daar ook in op.

Alle featurestructuren worden weergegeven in een centrale gegevensstructuur, *Common Analysis Structure* genoemd. Alle gegevensuitwisselingen worden met behulp van deze structuur verwerkt.

De Common Analysis Structure bevat de volgende objecten:

- Het tekstdocument
- De beschrijving van het typesysteem waarmee de typen, de subtypen en de bijbehorende features worden aangegeven
- De analyseresultaten waarmee het document of gebieden in het document wordt beschreven
- Een indexopslagplaats die ondersteuning biedt voor de toegang tot en iteratie van de analyseresultaten

Verwante onderwerpen

"Taalkundige ondersteuning voor semantische zoekopdrachten" op pagina 1
Enterprise search biedt taalkundige ondersteuning tijdens zoekopdrachten voor tekstdocumenten in de meeste Indo-Europese en Aziatische talen, zoals Japans.

"Integratie van aangepaste tekstanalyse" op pagina 3
Nadat u buiten enterprise search, met behulp van de Unstructured Information Management Architecture (UIMA) uw aangepaste analyseprogramma hebt gebouwd, kunt u de analyselogica in enterprise search integreren met behulp van de beheerconsole.

Algoritmen voor tekstanalyses

De UIMA Software Development Kit bevat API's en tools waarmee u annotaties kunt maken (de beschrijving van het typesysteem is als analysealgoritme aanwezig) en deze annotaties kunt insluiten in de analyseprogramma's.

De documentatie bij UIMA bevat een zelfstudieprogramma dat u kan helpen bij het maken van deze componenten. De Software Development Kit bevat programma's waarmee u de resultaten kunt testen en bekijken en een kleinschalig programma voor semantische zoekopdrachten dat u kunt gebruiken voor het indexeren van de analyseresultaten. Daarnaast kunt u geavanceerdere semantische zoekopdrachten uitvoeren op de gegevens die in de index zijn opgeslagen.

Omdat de UIMA Software Development Kit geen vooraf geconfigureerde annotators bevat en omdat de aangepaste annotators die u met behulp van UIMA ontwikkelt en vervolgens in in enterprise search integreert, voortbouwen op de resultaten van de basisannotators van enterprise search, kunt u het pakket met basisannotators in uw UIMA-omgeving gebruiken. Raadpleeg de documentatie bij UIMA voor meer informatie over het opnemen van taaldetectie en de functie van tokens voordat u in een UIMA-omgeving gebruikmaakt van de algoritmen voor aangepaste tekstanalyses.

Nadat u uw analyseprogramma's hebt ontwikkeld en getest met behulp van de UIMA Software Development Kit, moet u een PEAR-bestand (Processing Engine ARchive) maken om uw algoritmen uit te voeren op een documentcollectie in enterprise search. Dit archiefbestand bevat alle vereiste bronnen voor het instellen van de functie voor aangepaste analyses en analyseprogramma's in enterprise search. Hoe u een archief maakt, wordt beschreven in de UIMA-documentatie in de Software Development Kit.

Het archief dat u maakt om te uploaden naar enterprise search mag alleen de aangepaste analyselogica bevatten. Het mag geen van de basisannotators van enter-

prise search bevatten, ook niet als uw aangepaste analyselogica voortbouwt op de resultaten van de basisannotators. De reden hiervan is dat de basisannotators altijd worden uitgevoerd voordat er in enterprise search aangepaste analyse wordt uitgevoerd.

Voor meer informatie over het configureren en instellen van oplossingen voor semantische zoekopdrachten in enterprise search voert u het zelfstudieprogramma uit via http://www.ibm.com/developerworks/db2/zones/db2ii/. Het zelfstudieprogramma begeleidt u door de stappen die nodig zijn voor het instellen van algoritmen voor aangepaste tekstanalyses en laat zien hoe u de analyseresultaten in query's kunt gebruiken om de zoekresultaten te verbeteren.

Verwante taken

"De basisannotators van enterprise search in UIMA gebruiken" op pagina 8
U kunt het pakket met basisannotators voor enterprise search gebruiken om nieuwe annotators te ontwikkelen in de UIMA Software Development Kit (SDK) en om de analyseresultaten toe te wijzen aan JDBC-tabellen.

Werkstroom voor integratie van aangepaste analyses

Met behulp van de UIMA Software Development Kit kunt u de algoritmen voor aangepaste tekstanalyses maken en testen, die u vervolgens kunt integreren in en uitvoeren op documentcollecties in enterprise search.

Ga als volgt te werk om analysealgoritmen te ontwikkelen en deze in enterprise search te integreren:

1. Plannings- en ontwerptaken:
   a. Bepaal naar welke gegevens u wilt zoeken. Welke documenten wilt u ophalen? Welke basisbegrippen en relaties zijn vereis voor elke specifieke zoektaak? Zo kunnen product- en werknemersnamen nodig zijn op de interne website van een farmaceutisch bedrijf om algemene zoekopdrachten te verbeteren, terwijl personen die werkzaam zijn op het gebied van Research Development varianten van medicijnnamen nodig hebben en de relatie willen weergeven tussen medicijnen, oorzaak en genezing.
   b. Geef aan welk soort tekstanalyse u nodig hebt om de gegevens op te halen in de documenten waarin u wilt zoeken.
   c. Als er XML-documenten aanwezig zijn, moet u bepalen of u de XML-markup in uw oplossing wilt gebruiken. In enterprise search kunt u de XML-markup op twee manieren gebruiken:
      - Als u de XML-markup in uw aangepaste analyses kunt gebruiken (als uw documenten bijvoorbeeld `<summary>`- of `<topic>`-elementen bevatten die handig kunnen zijn in een annotator voor overzichten of categorieën), maakt u een toewijzingsbestand voor het toewijzen van XML-elementen aan de Common Analysis Structure.
      - Als u in uw query's de XML-markup wilt gebruiken zoals deze in het document wordt weergegeven, schakelt u native XML-toewijzing in.
   d. Bepaal tot welke resultaatgegevens van de tekstanalyse die in de Common Analysis Structure zijn opgeslagen u toegang wilt met behulp van semantische zoekopdrachten. Maak een toewijzingsbestand voor het toewijzen van de Common Analysis Structure aan de index.
   e. Bepaald of u de analyseresultaten wilt opslaan in een relationele database (zodat u bijvoorbeeld trends en koppelingen kunt ontdekken met behulp van rapportage- of dataminingprogramma's). Maak een toewijzingsbestand voor het toewijzen van de Common Analysis Structure aan de database.

f. Ontwerp het programma voor semantische zoekopdrachten. Bepaal met welke aanvullende mogelijkheden van semantische zoekopdrachten de gebruiker werkt. Ontwerp de gebruikersinterface.

2. Ontwikkeling: UIMA Software Development Kit-activiteiten
   a. Definieer de afzonderlijke analysestappen.
   b. Beschrijf het typesysteem voor de toewijzingen en analysealgoritmen.
   c. Ontwikkel voor elke analysestap de analysealgoritmen (annotators) en neem de annotators op in de analyseprogramma's met behulp van de UIMA Software Development Kit. Maak de aangepaste analyses met behulp van de basisfunctionaliteit (taalidentificatie en tokenisatie) in het pakket met basisannotators in enterprise search.
   d. Na het testen van de analysealgoritmen in UIMA pakt u het analyseprogramma in als PEAR-bestand (Processing Engine Archive). Het archiefbestand mag alleen de analysealgoritmen bevatten en niet de taalkundige basisfunctionaliteit in enterprise search.

      Als u een oplossing voor tekstanalyse ontwerpt, kan deze diverse analysemodules bevatten die beschikbaar worden gesteld via meer dan één PEAR-bestand. UIMA kent een manier om twee of meer PEAR-bestanden samen te voegen in een enkel PEAR-bestand dat u kunt uploaden en in enterprise search kunt uitvoeren. De mogelijkheid voor het samenvoegen van PEAR-bestanden garandeert dat er geen doublures in de naamgeving zijn, dat de invoer- en uitvoermogelijkheden correct zijn samengevoegd en dat er geen parameters worden overschreven als de samengevoegde parameters in annotatordescriptors dezelfde naam hebben. In de documentatie van UIMA vindt u instructies voor het samenvoegen van PEAR-bestanden.

3. Implementatie: enterprise search-activiteiten
   a. Upload het PEAR-bestand naar enterprise search. Geef de tekstanalysecomponent een naam waarnaar u kunt verwijzen in enterprise search.
   b. Koppel een of meer documentcollecties aan de tekstanalysecomponent.
   c. Upload, indien van toepassing, voor elke collectie het XML-element naar de Common Analysis Structure-toewijzing die u voor uw aangepaste analyse hebt gemaakt, en selecteer deze toewijzing.
   d. Upload, indien van toepassing, voor elke collectie de toewijzing van Common Analysis Structure naar database die u voor uw aangepaste analyse hebt gemaakt, en selecteer deze toewijzing.
   e. Upload voor elke collectie de toewijzing van Common Analysis Structure naar index die u voor semantisch zoeken hebt gemaakt, en selecteer deze toewijzing.
   f. Stel het programma voor semantische zoekopdrachten in (indien noodzakelijk). Zo kunt u de op browsers gebaseerde gebruikersinterface voor zoekopdrachten bijvoorbeeld implementeren in een toepassingenserver.
   g. Vervolgens kunt u de documenten in de collectie van de semantische zoekopdracht op dezelfde manier crawlen, analyseren en indexeren als voor collecties van zoekopdrachten op trefwoord.

Verwante taken

"De basisannotators van enterprise search in UIMA gebruiken" op pagina 8
U kunt het pakket met basisannotators voor enterprise search gebruiken om nieuwe annotators te ontwikkelen in de UIMA Software Development Kit (SDK) en om de analyseresultaten toe te wijzen aan JDBC-tabellen.

De basisannotators van enterprise search in UIMA gebruiken

U kunt het pakket met basisannotators voor enterprise search gebruiken om nieuwe annotators te ontwikkelen in de UIMA Software Development Kit (SDK) en om de analyseresultaten toe te wijzen aan JDBC-tabellen.

De set met basisannotators omvat het volgende:

- **Annotator voor het taal-ID**

  Hiermee wordt de taal van een document gedetecteerd. Voor informatie over de mogelijkheden en de configuratieparameters raadpleegt u het descriptorbestand jlangid.xml.

- **Annotator voor FROST-woordenboekzoekopdrachten**

  Bevat tokenisatie en detectie van zinnen op basis van de IBM LanguageWare-woordenboeken. Voor tokens worden aanvullende taalkundige gegevens gegenereerd, zoals de basisvorm of het lemma. Voor informatie over de mogelijkheden en de configuratieparameters raadpleegt u het descriptorbestand jfrost.xml.

- **Witruimtetokenizer**

  Hiermee kunt u de witruimtetokenisatie (of andere door witruimte gescheiden scripts) voor alle Europese taaldocumenten uitvoeren. Daarnaast kan met de annotator n-gramtokenisatie worden uitgevoerd voor de volgende tekstscripts: Arabisch, Han, Hebreeuws, Hiragana, Katakana, Lao, Mongools, Thais, YI en Hangul. Deze lijst omvat alle grote Aziatische tekstscript, hetgeen betekent dat de annotator Japans, Chinees en Koreaans ondersteunt.

  Voor informatie over de mogelijkheden en de configuratieparameters raadpleegt u het descriptorbestand jtok.xml.

- **Expressieannotator**

  Detecteert entiteiten of tekstspannes in een tekstdocument op basis van expressies. U kunt de expressieannotator zodanig aanpassen dat deze de door u gezochte tekstentiteiten detecteert op basis van uw eigen regels. In het pakket van de basisannotator bevindt zich een expressieannotator die in tekstdocumenten telefoonnummers, URL's en e-mailadressen detecteert.

- **Common Analysis Structure to Database-consumer**

  De Common Analysis Structure to Database-consumer vult een relationele database met specifieke resultaten van tekstanalyse.

Het pakket met basisannotators voor enterprise search is opgenomen in een zip-bestand dat de basis-tekstanalyseannotators bevat, plus de expressieannotator en de Common Analysis Structure to Database-consumer. Als documenten in enterprise search worden geanalyseerd door de parser, zijn de Taal-ID-annotator, de FROST-woordenboekzoekannotator en de witruimtetokenizer de basis-tekstanalyseannotators die altijd worden uitgevoerd voordat er tekstanalyse op maat wordt uitgevoerd.

Omdat de basis-tekstanalyseannotators altijd worden uitgevoerd voordat er in enterprise search tekstanalyse op maat plaatsvindt, en omdat alle tekstanalyse op maat gebaseerd is op de uitvoer van de basisannotators, kunt u deze annotators gebruiken in uw UIMA-omgeving als u annotators op maat ontwikkelt en test.

De expressieannotator en de Common Analysis Structure to Database-consumer zijn extra opties die u bij het configureren van uw tekstverwerkingsopties kunt kiezen in de beheerconsole van enterprise search. U kunt ze tevens gebruiken in UIMA. Voor ingewikkelde aanpassingen in de expressieannotator kunt u het beste gebruik maken van de bijgeleverde UIMA SDK-tools.

Als u een van deze annotators wilt uitvoeren in UIMA, moet de UIMA Software Development Kit (SDK) zijn geïnstalleerd. De SDK is beschikbaar op de website IBM developerWorks via http://www.ibm.com/developerworks/db2/zones/db2ii/.

U kunt het pakket met annotators als volgt installeren in uw installatie van UIMA SDK:

1. Zoek het pakket met annotators OF_base_annotators.zip in enterprise search-installatie (WebSphere Information Integrator OmniFind Edition) in de directory *ES_INSTALL_ROOT*/packages/uima.
2. Kopieer het zip-bestand naar de hoofddirectory van de UIMA SDK-installatie.
3. Pak het zip-bestand uit om de basisannotatorbestanden van enterprise search naar de opgegeven directorystructuur van de UIMA SDK-installatie te kopiëren. Het bestand tt_core_typesystem.xml wordt daarbij overschreven. Als u de oude versie van dit bestand wilt bewaren, breng deze dan in veiligheid voordat u het ZIP-bestand uitpakt.
4. Om het klassenpad in te stellen, opent u het script setUIMAClasspath in de directory bin en voegt u aan het eind van het script een regel toe waarmee het script OFAnnotEnv begint.
5. Als u in UIMA gebruik wilt maken van op maat gemaakte typen of typen die specifiek zijn voor enterprise search, kijk dan in de documentatie van UIMA SDK hoe u deze kunt definiëren.

Als u het pakket hebt geïnstalleerd, kunt u de annotatordescriptorbestanden vinden in de directory *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine. In het bestand of_tokenization.xml worden de basisannotators (de Taal-ID-annotator, de FROST-woordenboekannotator en de witruimtetokenizer) weergegeven in de volgorde waarin ze in enterprise search worden gebruikt.

De descriptorbestanden bevatten dezelfde configuratiebestanden als de waarden die in enterprise search worden gebruikt. Om foutopsporingsredenen kunt u de waarden wijzigen in de UIMA SDK. U wordt echter aangeraden geen wijzigingen aan te brengen in deze descriptorbestanden in het enterprise search-systeem. Het aanbrengen van wijzigingen in deze bestanden kan ertoe leiden dat het systeem instabiel wordt of de prestaties verslechteren.

Het pakket met basisannotators voor enterprise search bevat alleen de woordenboeken die nodig zijn voor de verwerking van Engelse documenten. Ga als volgt te werk om andere talen te verwerken in de ontwikkelomgeving:

1. Zoek de enterprise search-woordenboeken in de enterprise search-installatiedirectory *ES_INSTALL_ROOT*/configurations/parserservice/jediidata/frost/resources.
2. Kopieer de inhoud van de directory naar de lokale installatiedirectory van UIMA SDK in *UIMA_SDK_INSTALL*/data/frost/resources.

Ga als volgt te werk om te controleren of het pakket met annotators op de juiste wijze is geïnstalleerd:

1. Open Visual Debugger (CVD) voor de Common Analysis Structure (CAS) in de volgende directory: *UIMA_SDK_INSTALL*/bin/cvd[.bat/.sh].
2. Klik op **Uitvoeren → laden TAE**.
3. Selecteer het specificatiebestand voor het tekstanalyseprogramma of_tokenization.xml in de directory *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine.

4. Laad een voorbeelddocument en voer het tekstanalyseprogramma uit. In de CVD worden annotaties van het type `uima.tt.TokenAnnotation` weergegeven.

Als u een van de basis-tekstanalyseannotators uitvoert voordat de aangepaste annotators in uw ontwikkelomgeving staan, en uw aangepaste annotators maken gebruik van typen die zijn gedefinieerd door de basistekstanalyse, neem dan een verwijzing naar het bestand `tt_core_typesystem` op in het gedeelte 'type system' van de aangepaste annotatorspecificatie. Het bestand `tt_core_typesystem` staat in de directory *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine. Kijk in het bestand jtok.xml in de directory analysis_engine als u een voorbeeld wilt zien van de manier waarop u verwijzingen naar descriptorbestanden kunt opnemen.

Verwante taken

"De resultaten van de basisannotator en aangepaste tekstanalyse bekijken" op pagina 13
Om te zien welke analyseresultaten de parser en de annotators in enterprise search hebben geproduceerd, moet u de eigenschappen van de document-collectie zodanig aanpassen dat er een leesbare XML-versie wordt gemaakt van de analyseresultaten die worden opgeslagen in de Common Analysis Structure.

"Eenvoudige semantische zoekopdracht met expressieannotator mogelijk maken" op pagina 85
Om eenvoudige semantische zoekopdrachten met behulp van synoniemen mogelijk te maken, moet u de expressieannotator, het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index en het voorbeeld synoniemenwoordenboek opnemen in uw enterprise search-systeem en deze resources vervolgens koppelen aan uw collectie.

"De Common Analysis Structure to Database-consumer in UIMA gebruiken"
Voordat u de Common Analysis Structure to Database-consumer in UIMA kunt gebruiken, moet u wijzigingen aanbrengen in het consumerdescriptorbestand en moet u het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database wegschrijven.

"De expressieannotator in UIMA gebruiken" op pagina 13
Met de expressieannotator kunt u entiteiten of eenheden aan informatie in een tekstdocument opsporen. U kunt de annotator aanpassen aan de eisen die de zoekfunctie binnen uw domein stelt.

De Common Analysis Structure to Database-consumer in UIMA gebruiken

Voordat u de Common Analysis Structure to Database-consumer in UIMA kunt gebruiken, moet u wijzigingen aanbrengen in het consumerdescriptorbestand en moet u het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database wegschrijven.

Voordat u de Common Analysis Structure to Database-consumer kunt uitvoeren in uw UIMA-omgeving, moet u:

1. Het XML-descriptorbestand cas2jdbc.xml in *UIMA_SDK_INSTALL*/docs/examples/descriptors/cas_consumer openen. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma.
2. De parameter **mappingFile** zodanig wijzigen dat deze het absolute pad van het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database bevat, bijvoorbeeld: D:\temp\MijnMapping.xml

3. De parameter **docMetadata_Type** aanpassen om het UIMA-type op te geven waaruit alle metagegevens voor de ingebouwde features worden opgehaald, bijvoorbeeld: uima.tcas.DocumentAnnotation.
4. De parameter **docId_Feature** zodanig aanpassen dat deze de feature of het pad bevat naar het metagegevenstype waaruit het numerieke ID (van het type geheel getal) van het document wordt opgehaald. Dit wordt verlangd door alle ingebouwde features die een ID nodig hebben, bijvoorbeeld `docId()`, `uniqueId()`, `objectId()` en `fsId()`.
5. Stel de parameter **encryptionClass** niet in, want deze wordt in enterprise search alleen gebruikt om de Common Analysis Structure to Database-consumer in staat te stellen met versleutelde (encrypted) toewijzingsbestanden te werken.
6. Sla het bestand op.
7. Kopieer de EMF-bibliotheekbestanden (common.jar, ecore.jar en ecore.xmi.jar) vanuit de directory lib van uw enterprise search-installatie naar de directory lib van uw UIMA-installatie. Het bestand cc_cas2jdbc.jar staat al in de directory lib van uw UIMA-installatie.
8. Maak een toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database waarin wordt gedefinieerd welke resultaten van tekstanalyse er moeten worden opgeslagen in een database. U kunt het toewijzingsbestand sampleMapping.xml in *UIMA_SDK_INSTALL*/docs/examples/descriptors/cas_consumer gebruiken als voorbeeld voor uw eigen toewijzingsbestand.

   Met het XML-schemabestand CasToJDBCMapping.xsd in *UIMA_SDK_INSTALL*/docs/examples/descriptors/cas_consumer kunt u de geldigheid van het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database controleren. Omwille van de snelheid van het systeem wordt de geldigheid van het toewijzingsbestand niet gecontroleerd door de Common Analysis Structure to Database-consumer; u moet dit zelf doen.

Hoe u de consumer uitvoert in UIMA wordt beschreven in de UIMA-documentatie.

In het volgende voorbeeld ziet u hoe de verplichte parameters moeten worden gedefinieerd in de descriptor:

```
 ...
 <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>

...
```

In de tabel ziet u de configuratieparameter in de volgorde waarin ze in het descriptorbestand staan en ziet u welke parameters verplicht zijn:

Tabel 1. De configuratieparameter in het descriptorbestand van de Common Analysis Structure to Database-consumer

Parameter	Beschrijving	Verplicht
mappingFile	Het absolute pad van het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database, bijvoorbeeld D:/temp/sample.xml. Op Windows-systemen gebruikt u "/" als scheidingsteken in het pad.	true
encryptionClass	Stel deze parameter niet in, want deze parameter wordt in enterprise search alleen gebruikt om de Common Analysis Structure to Database-consumer in staat te stellen met versleutelde (encrypted) toewijzingsbestanden te werken.	false
docMetadata_Type	Het UIMA-type waaruit alle metagegevens voor de ingebouwde features worden opgehaald.	true
docId_Feature	De feature of het pad naar het metagegevenstype waaruit het numerieke ID van het document wordt opgehaald. Dit ID moet van het type geheel getal zijn. Deze parameter is vereist voor alle ingebouwde features die een ID nodig hebben, zoals `uniqeId()`, `objectId()` en `fsId()`.	true
docUri_Feature	De feature of het pad naar het metagegevenstype waaruit de URI van het document wordt opgehaald. Deze moet van het type tekenreeks (string) zijn.	false
IsCompleted_Feature	De feature of het pad naar het metagegevenstype dat aangeeft of het actieve document is opgedeeld over verschillende Common Analysis Structures.	false
chunkNumber_Feature	De feature of het pad naar het metagegevenstype dat het volgende nummer van het huidige deel (chunck) aangeeft.	false

De expressieannotator in UIMA gebruiken

Met de expressieannotator kunt u entiteiten of eenheden aan informatie in een tekstdocument opsporen. U kunt de annotator aanpassen aan de eisen die de zoekfunctie binnen uw domein stelt.

Als u gebruik wilt maken van de voorbeeld-expressieannotator die telefoonnummers, URL's en e-mailadressen detecteert, of als u de voorbeeld-annotator wilt gebruiken als basis voor het maken van uw eigen aangepaste versie van de expressieannotator in uw UIMA-omgeving, hebt u het volgende nodig:

1. De descriptor van de expressieannotator in de directory *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine.
2. De voorbeeld-regelset en de voorbeeld-systeemtypebeschrijving in de directory *UIMA_SDK_INSTALL*/docs/examples/regex.
3. Een voorbeeld-tekstbestand of_sample_regex.txt in de directory *UIMA_SDK_INSTALL*/docs/data waarop de voorbeeld-regelset kan worden toegepast.

Hoe u de annotator uitvoert in UIMA wordt beschreven in de documentatie van UIMA.

De resultaten van de basisannotator en aangepaste tekstanalyse bekijken

Om te zien welke analyseresultaten de parser en de annotators in enterprise search hebben geproduceerd, moet u de eigenschappen van de documentcollectie zodanig aanpassen dat er een leesbare XML-versie wordt gemaakt van de analyseresultaten die worden opgeslagen in de Common Analysis Structure.

Over deze taak

U kunt de XML-serialisering van de in de Common Analysis Structure opgeslagen analyseresultaten van de annotator bekijken om:

- De resultaten van de analyse (parser) te zien voordat de basisannotators worden verwerkt.
- De resultaten van analyse en tokenisatie te zien (door de basisannotators van enterprise search uit te voeren). Dit helpt u te bepalen wat de invoergegevensstructuren zijn voor eventuele aangepaste analyse die u wilt ontwikkelen en die wordt uitgevoerd nadat de basisannotators hun werk hebben gedaan.
- De resultaten te bekijken en te valideren van aangepaste analyse die voor testdoeleinden is uitgevoerd op een kleinere documentcollectie in enterprise search, alvorens te beslissen of die analyse moet worden uitgevoerd op een hele collectie.

De XML-serialisering produceert twee groepen resultaten:

- De resultaten na analyse (parsing). Deze bestaan uit veldtoewijzingen en metagegevens van documenten.
- De resultaten van analyse en tokenisatie en, indien geselecteerd, aangepaste tekstanalyse. Deze bestaan uit alle geproduceerde tokens en annotaties.

Procedure

U kunt als volgt een leesbare XML-versie van de analyseresultaten produceren:

1. Open het bestand collection.properties in *ES_NODE_ROOT*/master_config/<Collectie-ID>.parserdriver voordat u begint met het analyseren van de documenten in de collectie.
2. Om de resultaten na de analyse te bekijken, kunt u de volgende regel toevoegen aan het bestand collection.properties:
   `trevi.parser.dumpXCas=<uw_dumpdirectory>`

   De dumpdirectory moet al bestaan.

   a. Selecteer het gewenste type uitvoer. De uitvoer omvat altijd de beschrijving van het typesysteem dat is gebruikt voor de analyseresultaten. Dit bestand heet OmniFindParserTypeSystem.xml. Voeg een van de volgende regels toe:

      - Als u de uitvoer van de laatste 25 bestanden wilt zien, voegt u `trevi.parser.maxXCasFileCount=25` toe.

        U kunt dit aantal zelf bepalen, maar het is niet verstandig om een erg hoge waarde te kiezen.

        Houd er rekening mee dat de bestandsuitvoerbuffer voortdurend wordt overschreven nadat de maximale buffergrootte is bereikt. Dit betekent ook dat het document met het hoogste nummer niet noodzakelijkerwijs het document is dat het laatst is verwerkt.

        De uitvoer bestaat uit de volgende bestanden: OmniFindParserXCasDump1.xml gevolgd door OmniFindParserXCasDump2.xml, enzovoort tot alle 25 bestanden worden opgesomd.
      - Als u de uitvoer van bepaalde documenten wilt zien, voegt u de document-URI `trevi.parser.xCasURI.1=file://home/test/file1.txt` toe.

        U kunt een willekeurig aantal documenten toevoegen, maar de documenten moeten in oplopende volgorde zijn genummerd, te beginnen met 1, zonder ontbrekende nummers. Bijvoorbeeld: het tweede document zou `trevi.parser.xCasURI.2=file://home/test/file2.txt` zijn, het derde `trevi.parser.xCasURI.3=file://home/test/file3.txt`

        De uitvoer bestaat uit de volgende bestanden: OmniFindParserXCasDumpURI_1.xml, OmniFindParserXCasDumpURI_2.xml, enzovoort, voor het aantal bestanden dat u hebt opgegeven.
3. Om de resultaten na de tokenisatie te bekijken, kunt u de volgende regel toevoegen: `trevi.tokenizer.dumpXCas=<uw_dumpdirectory>`

   Ook nu moet de dumpdirectory al bestaan.

   a. Selecteer het gewenste type uitvoer. De geproduceerde uitvoer omvat altijd de beschrijving van het typesysteem dat is gebruikt voor de resultaten van de tekstanalyse. Dit bestand heet OmniFindTypeSystem.xml. Voeg een van de volgende regels toe:

      - Als u de uitvoer van de laatste 25 bestanden wilt zien, voegt u `trevi.tokenizer.maxXCasFileCount=25` toe.

        U kunt dit aantal zelf bepalen, maar het is niet verstandig om een erg hoge waarde te kiezen.

        Houd er rekening mee dat de bestandsuitvoerbuffer voortdurend wordt overschreven nadat de maximale buffergrootte is bereikt. Dit betekent ook dat het document met het hoogste nummer niet noodzakelijkerwijs het document is dat het laatst is verwerkt.

        De uitvoer bestaat uit de volgende bestanden: OmniFindXCasDump1.xml gevolgd door OmniFindXCasDump2.xml, enzovoort tot alle 25 bestanden worden opgesomd.

- Als u de uitvoer van bepaalde documenten wilt zien, voegt u de document-URI `trevi.tokenizer.xCasURI.1=file://home/test/file1.txt` toe.

  U kunt een willekeurig aantal documenten toevoegen, maar de documenten moeten in oplopende volgorde zijn genummerd, te beginnen met 1, zonder ontbrekende nummers. Bijvoorbeeld: het tweede document zou `trevi.tokenizer.xCasURI.2=file://home/test/file2.txt` zijn, het derde `trevi.tokenizer.xCasURI.3=file://home/test/file3.txt`

  De uitvoer bestaat uit de volgende bestanden: OmniFindXCasDumpURI_1.xml, OmniFindXCasDumpURI_2.xml, enzovoort, voor het aantal bestanden dat u hebt opgegeven.

In enterprise search kunt u de content van de XML-bestanden bekijken met behulp van de XCAS Annotation Viewer. U start de XCAS Annotation Viewer door het scriptbestand `xcasAnnotationViewer` uit te voeren. Dit bestand bevindt zich in de directory ES_INSTALL_ROOT/bin. U wordt gevraagd naar:

- De dumpdirectory waarin de resultaten van de analyse en tokenisatie zijn geplaatst
- Het descriptorbestand, ofwel OmniFindParserTypeSystem.xml (voor de resultaten van de parser), ofwel OmniFindTypeSystem.xml (voor de resultaten van tokenisatie en analyse) in de dumpdirectory.

Als u een document in de lijst selecteert, worden de analyseresultaten voor dat document afgebeeld. Klikt u op een geaccentueerde annotatie in de document, dan verschijnen de details van die annotatie.

Typesysteembeschrijving

Het typesysteem definieert de typen objecten (inclusief de eigenschappen [of features] daarvan) die kunnen worden geïnstantieerd in een Common Analysis Structure.

Elk analyseprogramma bevat eigen typesysteembeschrijvingen waarmee de invoervereisten en uitvoertypen worden omschreven voor de annotators in het analyseprogramma. Typesysteembeschrijvingen zijn specifiek van toepassing op het programmadomein.

De typesystemen omvatten de definities van typen, hun eigenschappen en de hiërarchie van enkelvoudige overname (″single-inheritance hierarchy″) van typen. Een Common Analysis Structure moet voldoen aan een bepaald typesysteem.

Daarnaast moeten de typen en features die zijn gedefinieerd in de typesysteembeschrijving worden gebruikt in alle toewijzingsbestanden die betrekking hebben op de documentanalyse. Dit zijn: het toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure, het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index en het toewijzingsbestand voor toewijzing van de Common Analysis Structure aan een database.

De typesysteembeschrijving van een annotator kan deel uitmaken van de descriptor van de annotator of kan in een afzonderlijk descriptorbestand voor het typesysteem zijn opgenomen. Soms is de beschrijving onderdeel van de descriptor van een andere annotator die in hetzelfde analyseprogramma is opgenomen.

Als u het analyseprogramma hebt ontwikkeld en in uw UIMA-omgeving hebt getest, wordt de typesysteembeschrijving ook opgenomen in het archiefbestand (.pear-bestand) met de analyseprogrammabestanden dat u maakt en uploadt naar enterprise search.

De basis annotators van enterprise search maken gebruik van drie typesysteembeschrijvingen: een kernbeschrijving die altijd wordt opgenomen, en twee andere die u desgewenst kunt activeren om de verwerking van documentcollectie van basisanalyse om te zetten in geavanceerde analyse. Of het nodig is om een of beide uitgebreide typesysteembeschrijvingen op te nemen, hangt ervan af welke aanvullende resultaten van tekstanalyse u tijdens de basisanalyse wilt opnemen.

U kunt de geavanceerde analysewerkstand inschakelen door een of beide uitbreidingstypesystemen op te nemen. In de geavanceerde analysewerkstand worden er tijdens de basisanalyse extra analysefeatures beschikbaar gesteld en opgeslagen in de Common Analysis Structure. Als u bijvoorbeeld meer informatie nodig hebt over een token (meer featuregegevens), zoals alle mogelijke lemma's voor dat token, of als het lemma een stopwoord is, of de woordontleding van het lemma, of speciale features voor morfologische verwerking, ook voor het Japans, moet u de geavanceerde analysewerkstand activeren.

Verwante taken

"Overschakelen van basisanalyse naar geavanceerde analyse"
Om de verwerking van documentcollecties die door de basisannotators van enterprise search wordt uitgevoerd te veranderen van basisanalyse in geavanceerde analyse, moet u de typesysteembeschrijvingen voor de werkstand Geavanceerde Analyse opnemen.

Verwante verwijzing

"Gedefinieerde typen en features in enterprise search" op pagina 17
Met het typesysteem dat in enterprise search is gedefinieerd, kunnen metagegevens in documenten worden verwerkt en elementaire taalkundige analyses worden uitgevoerd.

Overschakelen van basisanalyse naar geavanceerde analyse

Om de verwerking van documentcollecties die door de basisannotators van enterprise search wordt uitgevoerd te veranderen van basisanalyse in geavanceerde analyse, moet u de typesysteembeschrijvingen voor de werkstand Geavanceerde Analyse opnemen.

Beperkingen

Er zijn twee typesysteembeschrijvingen die u kunt selecteren om de werkstand Geavanceerde Analyse te activeren:

- De beschrijving `tt_extension_typesystem`, met meer gedetailleerde lexicale typegegevens over lemma's.
- De beschrijving `dlt_extension_typesystem`, met aanvullende morfologische features en speciale lexicale typen.

Procedure

Om schakelt als volgt over van basisverwerking van collecties op geavanceerde analyse:

1. Open het bestand tt_core_typesystem.xml in de directory *ES_NODE_ROOT*/master_config/*Collectie-ID*.parserdriver/specifiers. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma.
2. Verwijder de commentaartags die om het element <import> in de sectie <imports> heen staan. Op die manier kunt u één of beide beschrijvingsbestanden voor uitgebreide typesystemen opnemen.

   ```
   <imports>
   <!-- tt_extension_typsystem importeren voor geavanceerde analyse -->
   <!-- <import location="tt_extension_typesystem.xml"/>-->
   <!-- het dlt uitgebreide typesysteem importeren-->
   <!-- <import location="dlt_extension_typesystem.xml"/> -->
   </imports>
   ```
3. Open de twee descriptorbestanden jfrost.xml en jfrost_ngram.xml en wijzig de content van het element <outputs> zodanig dat dit de typen (in een element <type>) en features (in een element <feature>) bevat die worden opgesomd in het element <description> van de sectie <capabilities> die u tijdens de analyse wilt opnemen. Sla uw wijzigingen op.
4. Open het descriptorbestanden jtok.xml en wijzig de content van het element <outputs> zodanig dat dit de features (in een element <feature>) bevat die worden opgesomd in het element <description> van de sectie <capabilities> die u tijdens de analyse wilt opnemen. Sla uw wijzigingen op.
5. Open het descriptorbestanden es_tok_no_stw.xml en wijzig ook hier de content van het element <outputs> zodanig dat dit de features (in een element <feature>) bevat die worden opgesomd in het element <description> van de sectie <capabilities> die u tijdens de analyse wilt opnemen. Sla uw wijzigingen op.
6. Nadat u bent overgestapt op de werkstand Geavanceerde Analyse, moet u uw documentcollectie opnieuw laten analyseren door de parser.

   Verwante onderwerpen

   "Typesysteembeschrijving" op pagina 15
   Het typesysteem definieert de typen objecten (inclusief de eigenschappen [of features] daarvan) die kunnen worden geïnstantieerd in een Common Analysis Structure.

   Verwante verwijzing

   "Gedefinieerde typen en features in enterprise search"
   Met het typesysteem dat in enterprise search is gedefinieerd, kunnen metagegevens in documenten worden verwerkt en elementaire taalkundige analyses worden uitgevoerd.

Gedefinieerde typen en features in enterprise search

Met het typesysteem dat in enterprise search is gedefinieerd, kunnen metagegevens in documenten worden verwerkt en elementaire taalkundige analyses worden uitgevoerd.

De typen die in enterprise worden gebruikt, zijn gedefinieerd in drie afzonderlijke beschrijvingsbestanden voor typesystemen, te beginnen met het beschrijvingsbestand dat een beschrijving bevat van de kerntypen die altijd vereist zijn voor elementaire taalkundige analyse. Daarna volgende te beschrijvingen van typesystemen waarmee de meer geavanceerde taalkundige functies worden gedefinieerd, functies die gewoonlijk alleen vereist zijn in de werkstand voor geavanceerde analyse.

De elementaire taalkundige analyse in de vorm van documenttaalherkenning en segmentering vindt altijd plaats wanneer het document wordt geïndexeerd, onge-

acht de vraag of de functie voor aangepaste analyses is geselecteerd. Tijdens de elementaire documentanalyse wordt de beschrijving `tt_core_typesystem` gebruikt en worden de volgende gegevens toegevoegd aan de Common Analysis Structure, die u vervolgens in de aangepaste analyse kunt gebruiken:

- Documentmetagegevens van het type `com.ibm.es.tt.DocumentMetaData`.
- Informatie over de documentstructuur zoals zins- en alinea-annotaties van het type `uima.tt.SentenceAnnotation` en `uima.tt.ParagraphAnnotation`.
- Lexicale annotaties zoals tokens en samenstellingen van het type `uima.tt.TokenAnnotation`.

De beschrijving `tt_core_typesystem` is geschikt voor de meeste vormen van tekstanalyse.

Als u de verwerking van een collectie wilt laten plaatsvinden in de werkstand voor geavanceerde analyse, kunt u de volgende twee typesystemen opnemen. Deze typesystemen bevatten hoofdzakelijke aanvullende functies die niet worden gemaakt tijdens de elementaire taalkundige verwerking.

- `tt_extension_typesystem` die meer informatie over tokens, lemma's, alinea's en zinnen bevat
- `dlt_core_typesystem` die enkele van de uitgebreide typen annotaties van IBM LanguageWare bevat, zoals URL's en adressen. Daarnaast zijn er morfologische features opgenomen die niet vaak worden gebruikt.

tt_core_typesystem

De volgende typen en features worden gedefinieerd in de beschrijving `tt_core_typesystem`:

uima.tcas.DocumentAnnotation
De annotatie voor documenten bevat metagegevens van documenten en heeft de volgende kenmerken:

- `categories` met documentcategorieën die door een tekstcategoriseringsprogramma zijn toegevoegd. Elke toegevoegde categorie is van het type `com.tt.CategoryConfidencePair`
- `languageCandidates` met de documenttalen die tijdens de analyse automatisch zijn gedetecteerd door de parser. De talen worden toegevoegd aan een lijst van het type `com.tt.LanguageConfidencePair`, met de meest waarschijnlijke talen bovenaan
- `id` met het document-ID, zoals de URL

uima.tt.TTAnnotation
Dit is het roottype voor de annotaties die zijn gedefinieerd in `tt_core_typesystem`. Het supertype is `uima.tcase.Annotation`. De volgende typen zijn aanwezig:

uima.tt.DocStructureAnnotation
Annotaties over de documentstructuur. Heeft de volgende subtypen:

uima.tt.SentenceAnnotation
Zinnen

uima.tt.ParagraphAnnotation
Documentalinea

uima.tt.LexicalAnnotation
Lexicale annotaties zoals tokens of uit meerdere woorden bestaande expressies. Heeft de volgende subtypen:

uima.tt.TokenLikeAnnotation
Annotaties met een enkel token die de volgende features kunnen hebben:

- `tokenProperties` met de tokeneigenschappen
- `lemma` met het lemma of de stam van de term
- `normalizedCoveredText` met de genormaliseerde weergave van de tekst in kwestie

Dit type annotatie heeft de volgende subtypen:

uima.tt.TokenAnnotation
De feitelijke tokens die moeten worden onderscheiden van samengestelde onderdelen.

uima.tt.CompPartAnnotation
De samengestelde onderdelen van een term.

uima.tt.CompoundAnnotation
De annotatie van een samengesteld token. Het samengestelde token omspant gewoonlijk meer dan één tokenannotatie.

uima.tt.MultiTokenAnnotation
Lexicale annotatie bestaande uit meer dan één token. Dit type annotatie heeft de volgende subtypen:

uima.tt.StopwordAnnotation
Annotaties van stopwoorden. Stopwoorden kunnen ook uit meerdere woorden bestaan.

uima.tt.SynonymAnnotation
De annotatie van een term waarvoor synoniemen bestaan. Deze heeft de feature `synonyms` die de synoniemen bevat welke voor de term zijn gevonden.

uima.tt.SpellCorrectionAnnotation
De annotatie van een term waarvoor spellingscorrecties bestaan. Deze heeft de feature `correctionTerms`, met een lijst van de mogelijke correcties, gesorteerd met de meest waarschijnlijke collecties eerst.

uima.tt.MultiWordAnnotation
De annotatie van een uit meerdere woorden bestaande term.

uima.CAS.TOP
De root van het typesysteem. Heeft de volgende subtypen:

uima.tt.KeyStringEntry
Het abstracte type voor gegevenssturen van het type String. Het bevat de feature `key` die de sleutel van de tekenreeks bevat en het volgende subtype heeft:

uima.tt.Lemma
Lemmavermeldingen voor het woordenboek.

uima.tt.CategoryConfidencePair
De betrouwbaarheidswaarde voor de gevonden categorie. Heeft de volgende features:

- `categoryString` met de naam van de categorie
- `categoryConfidence` met de betrouwbaarheidswaarde voor de categorie
- `mostSpecific` met een vlag die aangeeft of die de categorie is die het meest specifiek is voor het document
- `taxonomy` met de naam van de taxonomie waarvan de categorie is afgeleid

uima.tt.LanguageConfidencePair
De betrouwbaarheidswaarde voor de gevonden categorie. Dit type omvat de features `languageConfidence`, `language` en `languageID`.

tt_extension_typesystem

`tt_extension_typesystem` bevat aanvullende tekstanalysefeatures voor meer geavanceerde verwerking.

uima.tt.TokenLikeAnnotation
Dit type annotatie in `tt_extension_typesystem` heeft de volgende features:

- `lemmaEntries` met een lijst van alle mogelijke lemma's voor het token. De items in de lijst zijn van het type `uima.tt.Lemma`
- `tokenNumber`
- `stopwordToken`

uima.tt.Lemma
Deze annotatie van het type `uima.tt.KeyStringEntry` heeft de volgende features:

- `isStopword` is waar (″true″) als het lemma een stopwoord is
- `isDeterminer` is waar (″true″) als het lemma een determinator is
- `partOfSpeech`. De volgende numerieke beschrijvingscodes voor woordontleding worden gebruikt:
  - 0: onbekend
  - 1: voornaamwoord
  - 2: werkwoord
  - 3: zelfstandig naamwoord
  - 4: bijvoeglijk naamwoord
  - 5: bijwoord
  - 6: voorzetsel
  - 7: tussenwerpsel
  - 8: voegwoord

uima.tt.DocStructureAnnotation
Annotaties over de documentstructuur. Heeft de volgende subtypen:

uima.tt.SentenceAnnotation
Een zin in het document. Deze heeft de feature `sentenceNumber`.

uima.tt.ParagraphAnnotation
Een alinea in het document. Deze heeft de feature `paragraphNumber`.

dlt_extension_typesystem

`dlt_extension_typesystem` bevat aanvullende features die worden gebruikt door IBM LanguageWare.

uima.tt.LexicalAnnotation
Dit type annotatie heeft de volgende subtypen:

uima.tt.TokenLikeAnnotation
In `dlt_extension_typesystem` heeft deze annotatie de volgende features:

- `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
Dit type annotatie in `dlt_extension_typesystem` heeft de volgende typen:

com.ibm.dlt.uimatypes.MWU
Dit type wordt door IBM LanguageWare voor het annoteren van expressies die uit meerdere woorden bestaan.

uima.tt.KeyStringEntry
Tekenreeksannotaties. Heeft de volgende subtypen:

uima.tt.Lemma
Heeft de volgende features:

- `frost_Constraints` met vlaggen voor restricties (″constraints″)
- `frost_MorphBitMasks` met een morfologisch bitmaskerarray
- `frost_ExtendedPOS` met uitgebreide woordontledingsinformatie, zoals JPOS voor Japans en CPOS voor Chinees
- `frost_JKom` met morfologische gegevens voor Japans
- `frost_JPStart` met Japanse startanalysegegevens
- `morphID` met lemma-eigenschappen

uima.tcas.Annotation
Heeft het volgende subtype:

com.ibm.dlt.uimatypes.Decomp_Analysis
Volledige structurele analyse van een samenstelling. Heeft de volgende features:

- `headComponentIndex` met de hoofdcomponent van een samenstelling
- `route` met een lijst van tokens die een enkele ontledingsroute vormen

Verwante verwijzing

"Voorbeeld van typesysteembeschrijving"
Met de typesysteembeschrijving worden de featurestructuren omschreven (de onderliggende gegevensstructuren die de analyseresultaten aangeven) die in aangepaste analyses worden gebruikt.

Voorbeeld van typesysteembeschrijving

Met de typesysteembeschrijving worden de featurestructuren omschreven (de onderliggende gegevensstructuren die de analyseresultaten aangeven) die in aangepaste analyses worden gebruikt.

De typesysteembeschrijving moet deel uitmaken van het archief van het analyseprogramma (.pear-bestand) dat vanuit de UIMA-omgeving in enterprise search is geïmporteerd.

In het volgende voorbeeld van een typesysteembeschrijving worden de politieverslagen omschreven die informatie bevatten over de verdachten, de locatie van het misdrijf, de tijd van het misdrijf en de datum van het misdrijf:

Dezelfde typesysteembeschrijving wordt in alle onderwerpen gebruikt waarin de verschillende toewijzingstypen worden beschreven die u voor aangepaste analyses kunt selecteren.

```
<?xml version="1.0" encoding="UTF-8"?>
<typeSystemDescription>
   <name>Typesysteem politieverslagen</name>
   <description>Typesysteembeschrijving voor
         politieverslagen</description>
   <version>1.0</version>
   <types>
     <typeDescription>
       <name>com.ibm.omnifind.types.PoliceReport</name>
       <description>Toelichting van een politieverslag</description>
       <supertypeName>uima.tcas.Annotation</supertypeName>
       <features>
         <featureDescription>
           <name>time</name>
           <description>Tijdstip waarop misdrijf heeft plaatsgevonden
              </description>
           <rangeTypeName>com.ibm.omnifind.types.Time</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>date</name>
           <description>Datum waarop het misdrijf heeft plaatsgevonden</description>
           <rangeTypeName>com.ibm.omnifind.types.Date</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>location</name>
           <description>Plaats waar het misdrijf heeft plaatsgevonden</description>
           <rangeTypeName>com.ibm.omnifind.types.City</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>knownSuspects</name>
           <description>Annotaties van het type Verdachte</description>
           <rangeTypeName>uima.cas.FSArray</rangeTypeName>
         </featureDescription>
         <featureDescription>
```

```
      <name>crimeDescription</name>
      <description>Korte beschrijving van het misdrijf</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.City</name>
  <description>Een plaatsnaam</description>
  <supertypeName>uima.tcas.Annotation</supertypeName>
  <features>
    <featureDescription>
      <name>cityName</name>
      <description>De plaatsnaam</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>cityDistrict</name>
      <description>De districtsnaam</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Person</name>
  <description>Een toelichting voor personen</description>
  <supertypeName>uima.tcas.Annotation</supertypeName>
  <features>
    <featureDescription>
      <name>role</name>
      <description>Bijvoorbeeld: verdachte of getuige</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>firstName</name>
      <description>De voornaam van de persoon</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>surName</name>
      <description>De achternaam van de persoon</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>title</name>
      <description>Bijvoorbeeld: Dhr of Mevr</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>gender</name>
      <description>Man of vrouw</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Suspect</name>
  <description>Een gevonden verdachte</description>
  <supertypeName>com.ibm.omnifind.types.Person</supertypeName>
  <features>
    <featureDescription>
      <name>description</name>
      <description>Beschrijving van verdachte,
    bijvoorbeeld: met baard en donkere bril</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
```

```
    </typeDescription>
    <typeDescription>
      <name>com.ibm.omnifind.types.Date</name>
      <description>Een datum</description>
      <supertypeName>uima.tcas.Annotation</supertypeName>
      <features>
        <featureDescription>
          <name>year</name>
          <description>Het jaar, bijvoorbeeld 2005</description>
          <rangeTypeName>uima.cas.Integer</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>month</name>
          <description>De maand in cijfers, bijvoorbeeld 7</description>
          <rangeTypeName>uima.cas.Integer</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>day</name>
          <description>De dag in cijfers</description>
          <rangeTypeName>uima.cas.Integer</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>dayOfWeek</name>
          <description>De dag van de week, bijvoorbeeld: maandag</description>
          <rangeTypeName>uima.cas.String</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>quarter</name>
          <description>Het kwartaal, bijvoorbeeld Q1-2005</description>
          <rangeTypeName>uima.cas.String</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>englDate</name>
          <description>De datum in de vorm mm/dd/jjjj</description>
          <rangeTypeName>uima.cas.String</rangeTypeName>
        </featureDescription>
      </features>
    </typeDescription>
    <typeDescription>
      <name>com.ibm.omnifind.types.Time</name>
      <description>Een tijdstip</description>
      <supertypeName>uima.tcas.Annotation</supertypeName>
      <features>
        <featureDescription>
          <name>hours</name>
          <description>Het uur: 00-23</description>
          <rangeTypeName>uima.cas.Integer</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>minutes</name>
          <description>De minuten in het uur</description>
          <rangeTypeName>uima.cas.Integer</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>timeOfDay</name>
          <description>Tijdsperiode, bijvoorbeeld: 's morgens of 's middags</description>
          <rangeTypeName>uima.cas.String</rangeTypeName>
        </featureDescription>
      </features>
    </typeDescription>
  </types>
</typeSystemDescription>
```

XML-markup in analyses en zoekopdrachten

U kunt de gegevens in de XML-structuren die in een document staan, rechtstreeks aan een Common Analysis Structure toewijzen zonder dat u hiervoor een UIMA-annotator hoeft te schrijven.

Als de documenten in uw collectie in XML zijn geschreven en u de XML-markup wilt gebruiken in tekstanalyses of semantische zoekopdrachten, hebt u de volgende opties:

Native XML-zoekopdrachten

U kunt deze optie gebruiken als u alle XML-tags en -kenmerken in semantische zoekopdrachten wilt gebruiken zoals deze in het document worden weergegeven. Als u bijvoorbeeld werkt met factuurdocumenten die het element `<Geadresseerde>` bevatten, kunt u native XML-zoekopdrachten inschakelen zodat u deze tag in een semantische zoekopdracht kunt gebruiken om te zoeken naar een bepaalde klantnaam binnen dit element.

Met deze optie wordt de XML-structuur van het document in de Common Analysis Structure weergegeven met het type `com.ibm.es.tt.MarkupTag`. Voor elke XML-tag wordt een annotatie van dit type gemaakt. Deze annotatie bevat de naam van de tag, de bijbehorende kenmerken en de inhoud van de kenmerken. Deze gegevens worden altijd geïndexeerd en zijn beschikbaar voor semantische zoekopdrachten.

Voor native XML-zoekopdrachten is geen configuratiebestand voor de toewijzing nodig. U kunt de native XML-zoekopdracht inschakelen via de beheerconsole voor enterprise search.

XML-elementen toewijzen aan de Common Analysis Structure

U kunt deze optie in de volgende gevallen gebruiken:

- De semantiek van bepaalde XML-elementen is nauwkeurig en kan in verdere tekstanalysestappen worden gebruikt. Deze analysestappen kunnen direct van invloed zijn op de annotaties en features die op basis van de XML-structuren zijn gemaakt en worden afgeschermd van de mogelijk afwijkende indelingen van de oorspronkelijke documenten. Het element `<Geadresseerde>` in factuurdocumenten bevat bijvoorbeeld in de meeste gevallen de namen van klanten. Met behulp van de toewijzing van het XML-element aan de Common Analysis Structure kunt u de inhoud van dit element rechtstreeks aan annotaties van het type `Klant` toewijzen. Een annotator kan vervolgens de relatie voor de klantlocatie bevatten met behulp van de informatie die in de buurt van de annotatie `Klant` wordt weergegeven.
- U wilt het verwerkingsbereik van een aangepaste annotator beperken tot bepaalde gebieden die in de XML-invoer zijn opgegeven. U kunt de analyse bijvoorbeeld beperken tot de content van de `<technicianComment>`-tags in een annotator waarmee gebreken aan auto's worden gevonden.
- U wilt zowel de verwerking van tekstanalyses als de volgende zoekopdrachten beperken tot bepaalde delen van het XML-document en niet-relevante of niet-tekstuele inhoud weglaten.
- U wilt de XML-tags met verschillende namen toewijzen aan een algemene spanne die in semantische zoekopdrachten wordt gebruikt. Bijvoorbeeld door `<mainHeading>` of `<doc>` toe te wijzen aan title.

In deze gevallen moet u een toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure maken waarin de features-

tructuren zijn gedefinieerd. De featurestructuren die u in het toewijzingsbestand definieert, worden tijdens het analyseren van de documenten gemaakt en worden gebruikt door de aangepaste annotators.

U kunt meerdere toewijzingsbestanden voor toewijzing van XML-elementen aan de Common Analysis Structure gebruiken voor een documentcollectie. Welk toewijzingsbestand voor welk XML-document wordt gebruikt, wordt bepaald op basis van het element <identifier>. Het element <identifier> in het toewijzingsbestand moet overeenkomen met het rootelement in het XML-document. Als het rootelement van het document bijvoorbeeld doc is, moet de waarde van het element <identifier> in het toewijzingsbestand ook "doc" zijn.

Als deze waarden niet overeenkomen, zoekt het programma naar een toewijzingsbestand waarin het element <identifier> is ingesteld op Default (standaard). Als er geen standaardtoewijzing wordt gevonden, worden de tekstuele secties van het document (de secties zonder taggegevens) toegewezen aan de documenttoelichting in de Common Analysis Structure.

Als u gegevens wilt extraheren die alleen in de relevante gedeelten van een document voorkomen en de niet-relevante delen wilt negeren, geeft u aan welke XML-elementen in het document relevante gegevens bevatten. Dit wordt ook wel contentextractie genoemd. U kunt bijvoorbeeld de invoer extraheren die in de titel- en hoofdtekstelementen is opgegeven, terwijl de invoer voor de auteur, de datum, het ID en de uitgever wordt genegeerd.

Met contentextractie kunt u de analyseverwerking voor de volgende typen XML-documenten verbeteren:

- Documenten met grote hoeveelheden content die niet worden geanalyseerd (bijvoorbeeld binaire bijlagen). Met contentextractie wordt de documentgrootte aanzienlijk verkleind, wordt de verwerking sneller uitgevoerd en worden analysefouten voorkomen die het gevolg zijn van ongeschikte gegevens.
- Documenten waarin niet-relevante tekst voorkomt, zoals documenten met commentaar in de <note>-tags. Als u deze gegevens negeert, zorgt u voor betere resultaten bij het analyseren van de documentcontent.

Native XML-zoekopdrachten en de opties voor contentextractie in de toewijzing van XML-elementen aan de Common Analysis Structure zijn opties die elkaar wederzijds uitsluiten, aangezien ofwel alle content ofwel alleen opgegeven content in aanmerking kan worden genomen. Als u contentextractie opgeeft, wordt de native XML-toewijzing genegeerd. Zonder contentextractie kunt u zowel toewijzing van XML-elementen aan de Common Analysis Structure als native XML-zoekopdrachten gebruiken.

Alle typen en features die u in het configuratiebestand gebruikt, moeten in de beschrijving van het typesysteem van de aangepaste analysestappen zijn omschreven. U kunt een descriptor van het typesysteem in de UIMA-omgeving maken met behulp van de plugin Component Descriptor Editor Eclipse. Met deze plugin kunt u een descriptorbestand maken zonder dat u kennis nodig hebt van de benodigde XML-syntaxis.

Nadat u de aangepaste analyse hebt gebouwd en getest, kunt u de UIMA PEAR-wizard (Processing Engine ARchive) gebruiken om een archief te maken met de bestanden voor aangepaste analyse, inclusief de beschrijving van het typesysteem. Vervolgens kunt u uw archief voor aangepaste analyse en uw toewijzingsbestanden

voor toewijzing van XML-elementen aan de Common Analysis Structure met behulp van de beheerconsole uploaden naar enterprise search.

Verwante taken

"Een toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure maken"
In een toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure kunt u verschillende configuratieopties instellen voor het toewijzen van XML- aan UIMA-gegevenstypen.

Een toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure maken

In een toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure kunt u verschillende configuratieopties instellen voor het toewijzen van XML- aan UIMA-gegevenstypen.

Over deze taak

In het volgende voorbeeld ziet u een toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure.

Het voorbeeld-politieverslag bevat XML-tags voor het misdaadtype, de datum van het vergrijp, de locatie van het vergrijp, de naam van de rapporterende officier, het districtspolitiebureau waar de officier werkzaam is, een beschrijving van de verdachte en een samenvatting. Deze tags worden gevolgd door een sectie met de lopende tekst. Bijvoorbeeld:

```
<report>
 <doc>
<crimeType>Car theft</crimeType>
<crimeDate>04/23/05 09:23 pm</crimeDate>
<crimeLocation>27 Main Street, Brynston, Springfield, New Jersey</crimeLocation>
  <reportingOfficer rank="Lt">Jakob
<lastName>Collins</lastName>
  </reportingOfficer>
<policePrecinct>14th Precinct</policePrecinct>
<suspectDescription>Male, dark haired, dark glasses,
blue jeans with dark, probably black,
    jacket</suspectDescription>
<abstract>A Mercedes CLK was stolen on 04/23/2005 from a parking
lot in front of the Blue Lagoon restaurant on
27 Main Street, Brynston.(serial number: 32 2761 50871)</abstract>
<body>A Mercedes CLK was stolen on 04/23/2004 from a parking
lot in front of the Blue Lagoon restaurant on 27 Main Street,
Brynston.(serial number: 32 2761 50871)

It has a black color and wide Michelin tires.

Eyewitnesses in front of the restaurant saw two darkly dressed
males drive away in the car at high speed. The car was
found abandoned on Aliway Ave in Brooklyn. The fuel tank was empty.
The seats were badly stained and the back seat was vandalized.
Nothing was stolen out of the car....</body>
 </doc>
 <image>
  <--! image of the crime scene as a base64-encoded string -->
 </image>
</report>
```

Op basis van het voorbeeld kan het toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure over de volgende structuur beschikken. In het voorbeeld wordt het typesysteem gebruikt dat voor het politieverslag-scenario is gedefinieerd.

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

 <identifier>Default</identifier>
<description>Sample configuration</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>
```

Beperkingen

Het toewijzingsbestand wordt in twee secties onderverdeeld:

Het element <contentElements>

U kunt dit element gebruiken als u bepaalde inhoud wilt extraheren. Met het voorbeeldtoewijzingsbestand wordt de inhoud van de sectie <doc> van een document geëxtraheerd en worden de andere secties in het document genegeerd. In het XML-politieverslag kan de afbeelding groot zijn en hierdoor minder geschikt zijn voor tekstverwerkingsdoeleinden. Als u <doc> als inhoudselement opgeeft in plaats van <image>, wordt de afbeelding uit het bestand gefilterd voordat de tekstverwerking start.

<elementToTypeMappings>

U kunt dit element gebruiken om aan te geven welke afzonderlijke XML-elementen (opgegeven in een <elementToTypeMapping>-element) in het document aan welke featurestructuren in de Common Analysis Structure moeten worden toegewezen.

Als u de optie voor contentextractie gebruikt, moeten de XML-elementen die in de sectie <elementToTypeMappings> zijn opgegeven, zijn opgenomen in de XML-elementen die zijn opgegeven in de sectie <contentElements>.

Procedure

U maakt een toewijzingsbestand voor toewijzing van XML-elementen aan de Common Analysis Structure als volgt:

1. Maak een XML-bestand. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma om het XML-bestand te valideren.Het XSD-schema voor het toewijzingsbestand heet XMLCasInitSchema.xsd en is tijdens de enterprise search-installatie opgeslagen in *ES_INSTALL_ROOT*/packages/uima/configuration/.
2. Neem de toewijzingen op in het element <xmlCasInitializerConfiguration xmlns="http://www.ibm.com/2005/uima/jedii_ci_xml">. De naamruimte (opgegeven in het kenmerk xmlns) moet exact worden ingevoerd zoals weergegeven.
3. Voeg een <contentElements>-element toe als u specifieke inhoud wilt extraheren uit secties in het document en voeg een <elementToTypeMappings>-element toe waarmee wordt aangegeven welke afzonderlijke XML-elementen in het document aan welke featurestructuren in de Common Analysis Structure moeten worden toegewezen.
4. Voeg een <identifier>-element en een <description>-element toe. Op basis van de identificatie wordt bepaald welke toewijzing er voor welk XML-document moet worden gebruikt. De identificatie moet het rootelement van het document bevatten, zoals doc. Als de identificatie is ingesteld op Standaard, is het rootelement van het document niet relevant en wordt de toewijzing toegepast op alle XML-documenten.
5. Voeg een <contentElements>-element toe als u gegevens wilt extraheren die alleen in de relevante gedeelten van een document voorkomen. Het element bevat het volgende componentelementen:
   - Een of meer <element>-elementen die het pad bevatten van een XML-element in het document en de XPath-syntaxis volgen, bijvoorbeeld <element>/doc/crimeType</element>.
6. Voeg een <elementToTypeMappings>-element toe als u wilt aangeven welke XML-elementen aan welke featurestructuren aan de Common Analysis Structure moeten worden toegewezen. Het element bevat de volgende componentelementen:
   - Een of meer <elementToTypeMapping>-elementen. Dit element moet over de volgende geneste elementen beschikken:
     – Een <element>-element dat wordt gebruikt om het pad van een XML-element op te geven en die de XPath-syntaxis volgt. Een schuine streep naar rechts (/) vóór het pad betekent dat het volledige pad is opgegeven. Bijvoorbeeld: abstract onder het rootelement doc. Twee schuine strepen naar rechts (//) staan voor elke willekeurige padsubset. birthDate moet bijvoorbeeld in reportingOfficer vallen, terwijl andere elementen tussen deze twee kunnen voorkomen.

– Een <type>-element, waarmee een type wordt aangegeven dat in de beschrijving van het typesysteem is gedefinieerd. Deze waarde moet van het type Annotatie zijn.
– Nul of meer <featureValueAssignment>-elementen.

7. In een <featureValueAssignment>-element geeft u een feature van het type String een naam via het element <feature> en wijst u een waarde toe via het element <basicValue>. Meerdere <basicValue>-elementen kunnen tussen een <values>-element worden toegevoegd.

   Het element <basicValue> kan kenmerken bevatten. Voorbeelden hiervan zijn useAttributeValue, useElementContent, default en trim.

Gebruik useAttributeValue als u de waarde van een kenmerk als waarde voor een feature wilt gebruiken. Het volgende voorbeeld

```
<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>
```

resulteert in de volgende uitvoer:

- Voor elke <reportingOfficer>-XML-tag die in een <doc>-XML-tag in het document voorkomt, wordt een featurestructuur van het type com.ibm.omnifind.types.Person gemaakt.
- Als de tag <reportingOfficer> een sex-kenmerk bevat, wordt de feature gender van de zojuist gemaakte featurestructuur ingesteld op de waarde van het kenmerk.

Gebruik het kenmerk useElementContent om inhoud toe te voegen als de waarde van een feature. In het volgende toewijzingsfragment:

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

wordt de tekst die door het element <abstract> in <doc> wordt gedekt, de waarde van de featurestructuur crimeDescription. Alle spaties aan het begin of einde worden verwijderd.

Voor het element <values> kunnen in de volgende omstandigheden meerdere waarden worden opgegeven:

- De feature die u wilt instellen, is van het type StringArray.
- Met behulp van het kenmerk voor het scheidingsteken kunnen strings worden samengevoegd tot een string. Om deze reden worden dergelijke strings toegewezen aan een feature van het type String. De titel Mr. is bijvoorbeeld een constante, de voornaam is de waarde van een kenmerk en de achternaam wordt gedekt door een XML-element:

```
<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>
```

Stringfeaturewaarden worden ongewijzigd uit het toewijzingsbestand geëxtraheerd. Eventuele spaties aan het begin en einde blijven behouden. De spaties worden echter verwijderd uit de namen van typen en features.
`<type>`**`com.ibm.omnifind.types.Person`**`</type>` wordt bijvoorbeeld
`<type>`**`com.ibm.omnifind.types.Person`**`</type>`.

Stel de voorwaarden voor kenmerken in met behulp van het element `<condition>`. De featurestructuur van type `com.ibm.omnifind.types.Person` wordt bijvoorbeeld alleen gemaakt als `<suspectDescription>` in het document voorkomt als het kenmerk `armed` is ingesteld op `yes`:

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

Op basis van het voorbeeldpolitieverslag en het gedefinieerde toewijzingsbestand worden de volgende featurestructuren gemaakt:

com.ibm.omnifind.types.PoliceReport

- covered text: ″Car theft 04/23/05 09:23 pm 27 Main Street, Brynston, Springfield, New Jersey Jakob Collins 14th Precinct Male, dark haired, dark glasses, blue jeans with dark, probably black, jacket A Mercedes CLK was ... Nothing was stolen out of the car.
- begin = 2
- end = 904
- knownSuspects = null
- crimeDescription = ″A Mercedes CLK was stolen on 04/23/2005 from a parking lot in front of the Blue Lagoon restaurant on 27 Main Street, Brynston.(serial number: 32 2761 50871)″

com.ibm.omnifind.types.Person

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

Nadat u het toewijzingsbestand hebt gemaakt, moet u dit uploaden naar enterprise search en het toewijzingsbestand voor toewijzing van XML-elementen aan de Com-

mon Analysis Structure selecteren met de andere selecties voor aangepaste analyse. Hiervoor gebruikt u de beheerconsole van enterprise search.

Verwante onderwerpen

"XML-markup in analyses en zoekopdrachten" op pagina 25
U kunt de gegevens in de XML-structuren die in een document staan, rechtstreeks aan een Common Analysis Structure toewijzen zonder dat u hiervoor een UIMA-annotator hoeft te schrijven.

Verwante verwijzing

"Voorbeeld van typesysteembeschrijving" op pagina 22
Met de typesysteembeschrijving worden de featurestructuren omschreven (de onderliggende gegevensstructuren die de analyseresultaten aangeven) die in aangepaste analyses worden gebruikt.

Tekstanalyseresultaten

Alle tekstanalyseresultaten worden opgeslagen in de Common Analysis Structure.

Met annotators wordt meestal gelezen van en geschreven naar de Common Analysis Structure. Consumers van de Common Analysis Structure (*CAS-consumers*) gebruiken de Common Analysis Structure alleen om deze te lezen. CAS-consumers voeren de laatste verwerkingsstap uit op de analyseresultaten die zijn opgeslagen in de Common Analysis Structure. Enterprise search bevat twee CAS-consumers:

- Consumers die de inhoud van de Common Analysis Structure in een zoekprogramma indexeren. Deze consumers hebben een toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index nodig dat u samen met de aangepaste tekstanalyse selecteert in de beheerconsole van enterprise search.
- Consumers die relationele databases vullen met specifieke analyseresultaten. Deze consumers hebben tevens een toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een database nodig dat u samen met de opties voor aangepaste tekstanalyse selecteert in de beheerconsole van enterprise search.

Indien vereist kunt u aangepaste CAS-consumers in enterprise search instellen. Raadpleeg de documentatie bij UIMA voor meer informatie over het schrijven van consumers. Voor meer informatie over het uploaden en werken met gebruikers in enterprise search raadpleegt u de website van IBM UIMA developerWorks op http://www.ibm.com/developerworks/db2/zones/db2ii/.

Verwante onderwerpen

"Indextoewijzing voor aangepaste-analyseresultaten" op pagina 37
Als u de aangepaste analyse hebt uitgevoerd op een collectie documenten, kunt u het zoekprogramma in enterprise search gebruiken om een index samen te stellen op basis van de gegevens die zijn opgeslagen in de Common Analysis Structure die aan de hand van de algoritmen voor de aangepaste analyse is gemaakt.

"Databasetoewijzingen voor geselecteerde analyseresultaten" op pagina 45
Als u de aangepaste analyse hebt uitgevoerd uw documenten in enterprise search, kunt u de geselecteerde tekstanalyseresultaten opslaan in een JDBC-database.

Featurepaden

Met behulp van een featurepad kan toegang worden verkregen tot de featurewaarden in de Common Analysis Structures, vergelijkbaar met de toXPath-instructies waarmee toegang tot de XML-elementen in een XML-document kan worden verkregen.

Featurepaden zijn nuttig wanneer u toegang wilt tot een featurestructuur waarin complexe features worden gecombineerd (zoals features met arraywaarden of features waarmee naar een andere featurestructuur wordt verwezen). Met behulp van een featurepad kunt u de waarde van een feature direct koppelen aan een featurestructuur en deze waarde opslaan in de semantische zoekindex of in een database.

Stel dat u werkt met een annotator waarmee auto's en de bijbehorende merken worden aangegeven. Met de annotator worden annotaties gemaakt van type `Auto` met het kenmerk `Merk`. `Merk` bevat echter niet het werkelijke bedrijf (bijvoorbeeld Chevrolet) maar bevat een featurestructuur van het type `Bedrijf`, dat over het kenmerk `Bedrijfsnaam` beschikt. Als u een semantische query wilt inschakelen waarmee autonamen en bedrijfsnamen worden gecombineerd, wordt het featurepad `Merk/Bedrijfsnaam` gebruikt om de waarde van`Bedrijfsnaam` te koppelen aan het bereik met auto's dat voor de annotatie Auto wordt gegenereerd. Op basis van deze waarden wordt de query ingeschakeld, ″Documenten ophalen met auto's die zijn geproduceerd door Chevrolet″, met behulp van '/Auto[@Merk=″Chevrolet″]'.

Een featurepad is een reeks featurenamen (f1/.../fn) met de volgende eigenschappen:

- De waarde van een featurepad kan een string, geheel getal, drijvende waarde of een array van een van deze typen zijn.
- Alle features in het pad van f1 t/m fn-1 moeten over een complex type beschikken, het type `uima.cas.TOP`, `uima.cas.FSArray`, `uima.cas.FSList` of een van de bijbehorende subtypen.
- De laatste feature fn in het pad kan een complex type bevatten. Daarnaast kan deze een (sub-)type bevatten van `uima.cas.Float`, `uima.cas.Integer`, `uima.cas.String`, `uima.cas.FloatArray`, `uima.cas.IntegerArray`, `uima.cas.StringArray`, `uima.cas.FloatList`, `uima.cas.IntegerList` of `uima.cas.StringList`.
- Indien gewenst kan een feature worden ingevoerd. De volledige typenaam moet voor de featurenaam worden ingevoegd, gescheiden door een dubbele punt. Bijvoorbeeld: `f1/com.ibm.es.SomeType:f2/.../fn` .

U kunt het aantal typen van een bepaalde feature verkleinen. Stel dat u werkt met de feature `AanvullendeInfo` van het type `uima.cas.TOP`. Als u weet dat de waarde van de feature `AanvullendeInfo` van het type `WerknemersInfo` is, met de feature `Salaris`, hebt u toegang tot deze feature met behulp van `AanvullendeInfo/WerknemersInfo:Salaris`. In dit voorbeeld resulteert het featurepad `AanvullendeInfo/Salaris` in een fout, omdat `Salaris` niet is gedefinieerd voor het type `uima.cas.TOP`.

Features met array- of lijstwaarden beschikken over de volgende aanvullende eigenschappen:

- Gebruik punthaken ([<getal>]) om een bepaald element in de array of lijst te selecteren. Een array begint met nul (0). Als u bijvoorbeeld het eerste element in de array Bedrijven wilt selecteren, gebruikt u Bedrijven[0]. De speciale markering [last] kan worden gebruikt om het laatste item in een array te selecteren, ongeacht de grootte, bijvoorbeeld Bedrijven[last].

- Als u geen waarde tussen de punthaken plaatst ([]), worden alle elementen opgenomen. In een featurepad kunt u slechts eenmaal een waarde weglaten tussen de punthaken ([]). Als er bijvoorbeeld een array met verdachten aanwezig is, worden met het featurepad `knownSuspects[]/com.ibm.omnifind.types.Suspect:surName` alle achternamen van de verdachten verzameld in de array String.
- Als tijdens het indexeren een featurepad wordt gebruikt waarmee een array als resultaat wordt gegeven, worden de arrayelementen aaneengeschakeld (gescheiden door spaties) en naar de index geschreven als afzonderlijk kenmerk of veld met meerdere termen.
- Het volgende element in het featurepad moet worden ingevoerd. De typenaam is het type van de elementen in de array. Stel dat u werkt met de featurestructuur of type `Info`. Dit type beschikt over de feature `Bedrijven`, met bereik FSArray. De elementen van de array zijn van het type `Bedrijf`. `Bedrijf` beschikt weer over de feature `Winst`. Als u de winst van het derde bedrijf wilt bekijken, schrijft u `Bedrijven[2]/Bedrijf:Winst` (gebruik hierbij de volledige typenamen).

Geïntegreerde features

Geïntegreerde features zijn vooraf gedefinieerde featurenamen met een speciale semantiek. U kunt deze features gebruiken om toegang te krijgen tot informatie die niet in de featurestructuur zelf aanwezig is (bijvoorbeeld het type featurestructuur of de gedekte tekst van een annotatie). U kunt geïntegreerde features in een featurepad gebruiken als het laatste of het enige element.

De volgende geïntegreerde features kunnen in beide configuratiebestanden voor toewijzingen worden gebruikt:

- Met `fsId()` wordt het ID van de featurestructuur als resultaat gegeven. Het ID dat wordt teruggestuurd, is een geheel getal (32 bits). U kunt deze geïntegreerde feature gebruiken om toegang te krijgen tot de onderdelen van een document die niet exact overeenkomen met de query.
- Met `typeName()` wordt het objecttype Common Analysis Structure als een string als resultaat gegeven. Het type is de volledige typenaam, inclusief eventuele naamruimteprefixen, zoals `uima.tcas.Annotation`. In databasecontext is `typeName()` met name nuttig als u typen en subtypen in dezelfde kolom opslaat en het werkelijke type van een annotatie of featurestructuur wilt weten. Met het volgende voorbeeld wordt het type persoon, bijvoorbeeld *verdachte* of *getuige*, in de kolom Role opgeslagen.

```
<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>
```

- Met `coveredText()` wordt de tekst als resultaat gegeven die door het Common Analysis-object wordt omspannen. `coveredText()` is alleen beschikbaar voor annotaties en de bijbehorende subtypen. Gebruik deze geïntegreerde feature niet in featurestructuren die niet door het type annotatie worden gebruikt. Met het volgende voorbeeld wordt de naam van een verdachte in de kolom suspectName opgeslagen.

```
<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>
```

- Met [] wordt een handle naar het juiste containeritem (array of lijst) als resultaat gegeven. De feature geeft een iteratie aan, wat betekent dat voor elk element in de array of lijst een item in de databasetabel of index is ingevoerd. Het volgende voorbeeld is afkomstig uit een toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database, waarin de geïntegreerde functie [:index] ook geldig is.

```
<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>
```

De volgende geïntegreerde features kunnen alleen worden gebruikt in het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database:

- Met `uniqueId()` wordt het algemene, unieke ID van de featurestructuur als resultaat gegeven. Dit ID is een string met een vaste lengte (27 tekens) en is een aaneenschakeling van het resultaat van `fsId()`, `docId()`, `docTimestamp()` en het getal van het huidige segment, omdat documenten in enterprise search kunnen worden onderverdeeld in verschillende Common Analysis Structure-objecten.

  De teruggestuurde string kan tekens tussen ″a-z″ en ″A-Z″ bevatten, de cijfers ″0-9″, puntkomma's (″;″) en dubbele punten (″:″).

  Het resultaat van `uniqueId()` kan worden gebruikt als primaire sleutel voor tabellen.

- Met `objectId()` wordt het ID van de annotatie of de featurestructuur als resultaat gegeven. `objectId()` is vergelijkbaar met `uniqueId()`, met als uitzondering dat objectId() het resultaat van `docTimestamp()` niet bevat. Het teruggestuurde ID is alleen uniek in een collectie waarin documenten eenmaal worden geparseerd. Als u een uniek ID wilt gebruiken voor alle documenten en documentversies, moet u `uniqueId()` gebruiken.

  De teruggestuurde string van de geïntegreerde feature `objectId()` heeft een vaste lengte van 16 tekens en kan tekens bevatten tussen ″a-z″ en ″A-Z″, de cijfers ″0-9″, puntkomma's (″;″) en komma's (″:″).

  Als met `uniqueId()` of `objectId()` naar lege featurestructuren wordt verwezen, wordt de standaardwaarde gebruikt die in de databasetabeldefinitie is gedefinieerd. Er worden geen lege objecten opgeslagen.

- Met `docId()` wordt het document-ID als resultaat gegeven. De teruggestuurde waarde is een geheel getal (32 bits).

  In het volgende voorbeeld worden deze geïntegreerde features geïllustreerd:

```
<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>
```

- Met `docUri()` wordt de document-URI als resultaat gegeven.
- Met `docTimestamp()` wordt de tijd waarop het document is verwerkt als resultaat gegeven (in milliseconden). Deze geïntegreerde feature is met name nuttig voor het traceren van documentversies, bijvoorbeeld als u wilt weten of de documentversie die u gebruikt, de laatste versie is die door de crawler is verwerkt.

```
<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>
```

- Met `parentId()` wordt het `fsId()` van de featurestructuur die een containertoewijzing bevat als resultaat gegeven. `parentId()` is alleen geldig in de context van containertoewijzingen.
- Met `uniqueParentId()` wordt het `uniqueId()` van de annotatie of featurestructuur die in een containertoewijzing is opgenomen, als resultaat gegeven. Ook deze geïntegreerde feature is alleen geldig in de context van containertoewijzingen.
- Met `[:index]` wordt de index van het huidige containeritem (array of lijst) als resultaat gegeven.

  Verwante taken

  "Delen van een document ophalen die voldoen aan een semantische zoekopdracht" op pagina 56
  U kunt alleen de delen van een document ophalen die exact aan de zoekopdracht voldoen door de relevante featurestructuren toe te wijzen aan zowel de index als de database, en de spanne in de semantische zoekopdracht op te geven.

Filters

Filters worden gebruikt voor het beperken van de toewijzingsregels in toewijzingsbestanden voor de toewijzing van de Common Analysis Structure aan de index en

toewijzingsbestanden voor de toewijzing van de Common Analysis Structure aan een database. Alleen als het filter waar is, worden de analyseresultaten aan de index of een JDBC-tabel toegevoegd.

Het element `<filter>` is optioneel en wordt gebruikt om alleen voor features met een bepaalde kenmerkwaarde beperkingen in te stellen. Dit is met name nuttig als u wilt dat een kenmerk zich gedraagt als schakeloptie voor welke gegevens in de index moeten worden opgenomen of aan de database moeten worden toegevoegd. Personen en organisaties kunnen bijvoorbeeld worden vastgelegd in een annotatie van type `EntityAnnotation`. De bijbehorende feature `type` is ingesteld op `person` of `organization`. Als u alleen de personen wilt extraheren en niet de organisaties, kunt u het volgende filter aan de toewijzingsregel toevoegen:

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

Elke filterexpressie heeft de volgende indeling:

```
<Featurepad> <Operator> <Literaal>
```

waarbij geldt:

- Featurepad is een featurepad in de Common Analysis Structure
- Operator is =, !=, <, <=, > of >=. Onthoud dat < (en alleen <) moet worden uitgedrukt als `&lt;`.
- Literaal is een geheel getal, een getal met een drijvende komma (een exponentsyntaxis wordt niet ondersteund) of een string die tussen dubbele aanhalingstekens is geplaatst, met ingesloten enkele aanhalingstekens en schuine strepen naar links waarvoor nog een schuine streep naar links moet staan.

`<Featurepad>`, `<Operator>` en `<Literaal>` moeten worden gescheiden door een spatie.

In de volgende voorbeelden vindt u geldige filters:

- `<filter syntax="FeatureValue"> foo = "hallo mensen" </filter>`

  De feature foo bevat de string hallo mensen.
- `<filter syntax="FeatureValue"> foo &lt; 42 </filter>`

  De feature foo heeft als waarde een geheel getal kleiner dan 42.
- `<filter syntax="FeatureValue"> make/company = "Chevrolet" </filter>`

  Het featurepad merk/bedrijf, waarbij de feature merk een featurestructuur heeft die de feature bedrijf bevat met de waarde Chevrolet.
- `<filter syntax="FeatureValue"> bar7 >= 0.5 </filter>`

  De feature bar7 heeft als waarde een getal met drijvende komma, groter dan of gelijk aan 0,5.

Indextoewijzing voor aangepaste-analyseresultaten

Als u de aangepaste analyse hebt uitgevoerd op een collectie documenten, kunt u het zoekprogramma in enterprise search gebruiken om een index samen te stellen op basis van de gegevens die zijn opgeslagen in de Common Analysis Structure die aan de hand van de algoritmen voor de aangepaste analyse is gemaakt.

Als u de analyseresultaten toewijst aan velden, tekstspannen en kenmerken in de enterprise search-index, kunt u deze gegevens in query's gebruiken. Wanneer u aangepaste analyses in enterprise search gebruikt, waarin zowel woorden als tekstspannen kunnen worden geïndexeerd, wordt semantisch zoeken ingeschakeld.

Met het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index kunt u bepalen welke analyseresultaten in de Common Analysis Structure u wilt indexeren.

U kunt verschillende stijlen gebruiken om de featurestructuren in de Common Analysis Structure aan de enterprise search-index toe te wijzen.

Annotatie
Als u featurestructuren in de Common Analysis Structure indexeert met behulp van de stijl voor annotaties, worden alle annotaties van de opgegeven typen in de index opgeslagen als doorzoekbare spannen.

Als een featurestructuur waarmee een bepaald tekstgebied wordt omspannen van het type `Persoon` is en wordt geïndexeerd met behulp van de stijl voor annotaties, kunnen de volgende query's worden uitgevoerd:

Tabel 2. Voorbeeldquery's

Vereiste gegevens	Mogelijke query
Alle documenten ophalen die ten minste één naam van een persoon bevatten	<Persoon/>
Alle documenten ophalen waarin Baas voorkomt in een persoonsannotatie	<Persoon>Baas</Persoon>
Alle documenten ophalen waarin Lang in dezelfde zin voorkomt als een van mijn concurrenten	<Zin><Persoon>Lang</Persoon> <Concurrent/></Zin>

De kenmerken van featurestructuren worden ook geïndexeerd als onderdeel van de spanne. Stel dat u werkt met een annotator waarmee auto's worden gevonden en waarmee het automerk wordt opgeslagen als de feature `Merk` van de annotatie `Auto`. Hiermee wordt het volgende type query ingeschakeld: "Documenten ophalen waarin auto's van het merk Chevrolet voorkomen".

Veld Gebruik deze stijl als u de inhoud van de featurestructuren tijdens zoekopdrachten toegankelijk wilt maken. Hiertoe gebruikt u de zoekmogelijkheden voor velden in enterprise search. Op deze manier kan de inhoud van een featurestructuur in de zoekresultaten worden weergegeven of kunt u deze gebruiken in parametrische zoekopdrachten.

Als u de waarde Medicijndosis bijvoorbeeld aan een parametrisch veld toewijst, kunt u de volgende query gebruiken: "Alle documenten ophalen waarin een bepaald medicijn wordt genoemd waarvan de dosis hoger is dan 100 milligram."

Scheiding
U kunt deze stijl gebruiken als een bepaalde featurestructuur moeten worden geïnterpreteerd als scheidingsteken voor lege ruimten, zoals secties of alinea's. In enterprise search worden zinnen en alinea's standaard gedetecteerd. Gebruik deze stijl alleen als in de aangepaste analyse aanvullende structuurelementen in een document worden gevonden die u op een andere manier wilt interpreteren.

U kunt de analyseresultaten ook gebruiken om de documentranking in enterprise search te beïnvloeden, zelfs voor eenvoudige zoekopdrachten op trefwoord. Hiervoor moeten twee stappen worden uitgevoerd:

1. Wijs de featurestructuren met behulp van de toewijzingsstijl voor annotaties of velden toe aan doorzoekbare spannen of velden.
2. Definieer een wegingsklasse met behulp van de beheerconsole van enterprise search en wijs de spanne of veldnaam aan deze wegingsklasse toe.

Als de gebruiker vervolgens een zoekterm invoert die in deze featurestructuur voorkomt, wordt het document hoger in de ranking geplaatst. Stel dat u werkt met een annotator waarmee personen en bedrijfsnamen worden gevonden. Als u deze featurestructuren aan spannen toewijst (bijvoorbeeld: ″Persoon″ en ″Bedrijf″) en deze spannen vervolgens toewijst aan wegingsklassen, worden met het zoekresultaat voor ″Tekort″ de documenten waarin zowel ″Tekort″ als Bedrijf voorkomen, hoger in de ranking geplaatst dan de documenten waarin alleen de term ″Tekort″ voorkomt.

Als u het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index hebt geschreven, kunt u het naar enterprise search uploaden met behulp van de beheerconsole.

Verwante taken

“Een toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index maken”
Met het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index kunt u bepalen welke analyseresultaten in de Common Analysis Structure u wilt indexeren om zoeken mogelijk te maken.

Een toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index maken

Met het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index kunt u bepalen welke analyseresultaten in de Common Analysis Structure u wilt indexeren om zoeken mogelijk te maken.

Over deze taak

Het toewijzingsbestand voor het toewijzen van de Common Analysis Structure aan een index is een XML-bestand. Het toewijzingsbestand in het voorbeeld is gebaseerd op het typesysteem dat is gedefinieerd voor het politieverslagscenario.

```
<?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>
```

Beperkingen

Het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index moet alle analyseresultaten bevatten waarnaar in query's moet kunnen worden gezocht.

Procedure

U kunt als volgt een toewijzingsbestand maken voor het toewijzen van de Common Analysis Structure aan een index:

1. Maak een XML-bestand. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma. Het XSD-schema voor het toewijzingsbestand heet CasToIndexMapping.xsd en is tijdens de enterprise search-installatie opgeslagen in *ES_INSTALL_ROOT*/packages/uima/configuration/.
2. Neem de toewijzingen op in een element `<indexBuildSpecification xmlns="http://www.ibm.com/of/822/consumer/index/xml">`. De naamruimte (opgegeven in het kenmerk xmlns) moet exact worden ingevoerd zoals weergegeven.
3. Voeg een `<skipCondition>`-element toe om te voorkomen dat bepaalde documenten op basis van een specifieke featurewaarde in de index worden opgenomen. Dit element is optioneel. In het voorbeeld worden de documenten met een gegevensstructuur van het type `com.ibm.uima.tt.DocumentAnnotation` waarvoor de feature `toBeProcessed` op nul is ingesteld, niet geïndexeerd.
4. Voeg een of meer `<indexBuildItem>`-elementen toe waarmee een bepaalde featurestructuur in de Common Analysis Structure aan een structuur in de index wordt toegewezen.
5. Sla het XML-bestand op en valideer dit.

Het element `<indexBuildItem>`

Het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index bevat een of meer `<indexBuildItem>`-elementen. Met elk element wordt de toewijzing van een bepaalde featurestructuur in de Common Analysis Structuur aan een structuur in de index beschreven (een spanne of een veld).

Het element `<name>` bevat het featurestructuurtype. U kunt een type op twee manieren opgeven:

- U kunt de volledige naam van het type opgeven. Bijvoorbeeld: `com.ibm.omnifind.types.Suspect`
- U kunt een jokerteken opgeven. Bijvoorbeeld: `com.ibm.omnifind.types.*`. U kunt het jokerteken alleen aan het einde van de typespecificatie toevoegen.

Gebruik alleen subtypen van uima.tcas.Annotation als items voor het samenstellen van de index. Als een featurestructuur een subtype is van uima.cas.TOP (en niet van uima.tcas.Annotation), hebt u toegang tot deze featurestructuur via een featurepad, beginnend vanaf een annotatie.

Als type A een subtype is van type B (in dit voorbeeld com.ibm.omnifind.types.Suspect als subtype van com.ibm.omnifind.types.Person) en voor beide typen de <indexBuildItem>-elementen Ia en Ib zijn gedefinieerd, vindt de verwerking als volgt plaats:

- Elke indexregel die voor Ib is gedefinieerd, wordt toegepast op de featurestructuren van type B en de featurestructuren van type A
- Elke indexregel die voor Ia is gedefinieerd, wordt alleen toegepast op de featurestructuren van type A

In het voorbeeld wordt het element <indexBuildItem> dat voor de annotaties com.ibm.omnifind.types.Person is gedefinieerd, ook toegepast op de annotaties com.ibm.omnifind.types.Suspect. Voor de annotatie Suspect worden twee spannen gemaakt: een met de naam Person en een met de naam Suspect.

Het element <filter> is optioneel en wordt gebruikt om de <indexBuildItem>-toewijzing alleen te beperken tot de featurestructuren die over een bepaalde kenmerkwaarde beschikken. Dit is met name nuttig als u wilt dat een kenmerk zich gedraagt als schakeloptie voor welke gegevens in de index moeten worden opgenomen. Personen en organisaties kunnen bijvoorbeeld worden vastgelegd in een annotatie van type EntityAnnotation. De bijbehorende feature type is ingesteld op person of organization. Als u alleen de personen wilt extraheren en niet de organisaties, kunt u het volgende filter toevoegen:

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

Bovendien kunt u ervoor kiezen personen en organisaties onder verschillende spannamen te indexeren, bijvoorbeeld person en organization. Hiertoe definieert u twee <indexBuildItem>-elementen van het type EntityAnnotation en gebruikt u twee filters voor de feature type om de personen of de organisaties te activeren.

Het element <indexRule>

Elk <indexBuildItem>-element bevat een <indexRule>-element. Elk <indexRule>-element bevat alle informatie die nodig is om een featurestructuur in de Common Analysis Structure aan de index toe te wijzen als type voor veld, annotatie en scheiding. De stijlen voor annotaties en velden bieden ondersteuning voor verschillende kenmerken. U kunt de stijl voor termen die wordt ondersteund in de UIMA Software Development Kit niet gebruiken in enterprise search (deze stijl wordt overgeslagen).

Voor de stijlen voor annotaties en velden bestaan de volgende alternatieven als u de naam van de annotatie of het veld in de index opgeeft:

- Gebruik fixedName als elke featurestructuur in de index onder dezelfde naam toegankelijk moet zijn. In het volgende voorbeeld wordt elke featurestructuur van het type com.ibm.omnifind.types.Person toegewezen aan de spanne ″Person″ in de index.

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

```
      <attribute name="fixedName" value="Person" />
    </style>
  </indexRule>
</indexBuildItem>
```

Hiermee worden query's ingeschakeld zoals ″Documenten ophalen waarin Baas voorkomt als naam van persoon″. De query wordt als volgt uitgedrukt met behulp van XML-fragmenten: `@xmlf2::'<Persoon>Baas</Persoon>'`

- Gebruik `nameFeature` als met de annotatie verschillende entiteiten worden opgeslagen die toegankelijk moeten zijn met verschillende spannen, afhankelijk van de waarde van een bepaalde feature van de annotatie. In het volgende voorbeeld wordt `com.ibm.tt.EntityAnnotation` geïndexeerd als `person`- of `organization`-spanne, afhankelijk van de waarde van de feature `type`. De feature kan ook een featurepad zijn.

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

  Hiermee worden query's zoals ″Documenten ophalen over de WHO″ (in plaats van de Engelse term ″who″) ingeschakeld. De query wordt als volgt uitgedrukt met beperkte XPath-syntaxis: `@xmlp::'/organization[ftcontains="WHO"]'`

- Als geen van de bovenstaande kenmerken wordt gebruikt, wordt de korte naam van de annotatie in het element `<indexBuildItem>` gebruikt. Dit is de standaardwaarde. Bijvoorbeeld:

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

  Dit `<indexBuildItem>`-element resulteert in annotaties en velden met de naam `RoomNumber` waarin de tekst is geplaatst die wordt gedekt door `com.ibm.uima.tutorial.RoomNumber`.

Het element <style name="Annotation" />

Met de annotatie in het element `<style>` wordt aangegeven hoe u toegang kunt krijgen tot de spannegegevens in enterprise search. Naast het toestaan van het gebruik van de kenmerken `fixedName` en `nameFeature` biedt deze stijl ondersteuning voor het element `<attributemappings>`. In dit element kunt u de waarde van een feature toewijzen aan een kenmerk van de resulterende spanne in de index, die u vervolgens kunt gebruiken in een zoekexpressie.

Elke toewijzing wordt uitgevoerd in een afzonderlijk `<mapping>`-element. Het element `<feature>` bevat een featurepad en het element `<indexName>` bevat de naam van het kenmerk dat in de index wordt gebruikt om de waarde van `<feature>` op te slaan. Bijvoorbeeld:

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

Met dit <mapping>-element wordt de waarde van de feature in het pad make/companyname direct opgeslagen in het indexkenmerk company.

Het toewijzen van featurewaarden aan indexkenmerken is met name nuttig als het typesysteem dat tijdens tekstanalyses wordt gebruikt, complex is en er verschillende geneste featurestructuren worden gebruikt. Met behulp van het element <mapping> kunt u de relevante kenmerken weergegeven, zodat u deze in query's kunt gebruiken zonder dat u uitgebreide kennis nodig hebt van de structuur van het oorspronkelijke typesysteem.

Het element <style name="Field" />

Met het veld in het element <style> wordt aangegeven op welke manier toegang tot de veldgegevens kan worden verkregen in enterprise search. Naast de kenmerken fixedName en nameFeature kunt u de volgende kenmerken instellen.

parametric
Indien ingesteld op Waar, kan de veldwaarde worden gezocht via een parametrische zoekopdracht (bijvoorbeeld #dosering:>100)

fieldSearchable
Indien ingesteld op Waar, kan de veldwaarde worden gezocht via een zoekopdracht (bijvoorbeeld Merk:Bayer)

returnable
Indien ingesteld op Waar, worden het veld en de bijbehorende waarden in de zoekresultaten weergegeven

Veldgegevens zijn altijd doorzoekbaar (als de veldgegevens toegankelijk zijn via basiszoekopdrachten op trefwoord).

Met het optionele kenmerk valueFeature wordt gedefinieerd welke featurewaarde als veldwaarde moet worden gebruikt. Als de featurestructuur een annotatie is en het kenmerk niet is ingesteld, wordt de gedekte tekst van de annotatie als veldwaarde gebruikt. In het voorbeeld

```
<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>
```

worden twee velden voor com.ibm.omnifind.types.Date gegenereerd. Het veld date bevat de gedekte tekst, bijvoorbeeld 5:15pm. Een ander veld bevat de waarde van het kenmerk hour. Hier kunt u de query opgeven met 'hour::<17'.

Het element <style name="Breaking" />

De waarde voor scheidingen in het element <style> bevat geen overige elementen.

Als u het XML-bestand hebt gemaakt, moet u dit uploaden naar enterprise search en het toewijzingsbestand voor het toewijzen van de Common Analysis Structure aan een index selecteren met de andere selecties voor aangepaste analyse. Hiervoor gebruikt u de beheerconsole van enterprise search.

Verwante onderwerpen

"Indextoewijzing voor aangepaste-analyseresultaten" op pagina 37
Als u de aangepaste analyse hebt uitgevoerd op een collectie documenten, kunt u het zoekprogramma in enterprise search gebruiken om een index samen te stellen op basis van de gegevens die zijn opgeslagen in de Common Analysis Structure die aan de hand van de algoritmen voor de aangepaste analyse is gemaakt.

"Featurepaden" op pagina 33
Met behulp van een featurepad kan toegang worden verkregen tot de featurewaarden in de Common Analysis Structures, vergelijkbaar met de toXPath-instructies waarmee toegang tot de XML-elementen in een XML-document kan worden verkregen.

Verwante verwijzing

"Filters" op pagina 36
Filters worden gebruikt voor het beperken van de toewijzingsregels in toewijzingsbestanden voor de toewijzing van de Common Analysis Structure aan de index en toewijzingsbestanden voor de toewijzing van de Common Analysis Structure aan een database. Alleen als het filter waar is, worden de analyseresultaten aan de index of een JDBC-tabel toegevoegd.

"Voorbeeld van typesysteembeschrijving" op pagina 22
Met de typesysteembeschrijving worden de featurestructuren omschreven (de onderliggende gegevensstructuren die de analyseresultaten aangeven) die in aangepaste analyses worden gebruikt.

Databasetoewijzingen voor geselecteerde analyseresultaten

Als u de aangepaste analyse hebt uitgevoerd uw documenten in enterprise search, kunt u de geselecteerde tekstanalyseresultaten opslaan in een JDBC-database.

Deze versie ondersteunt DB2 Universal Database, Versie 8.2.2 (com.ibm.db2.jcc.DB2Driver Versie 2.3) of hoger en Oracle 10g (oracle.jdbc.driver.OracleDriver Versie 1.0).

Voor DB2 Universal Database en Oracle kunt u ervoor kiezen de analyseresultaten direct in de database in te voegen of de equivalente database-specifieke laadbestanden en het bijbehorende script waarmee de laadopdrachten worden uitgevoerd, te genereren.

Als u de analyseresultaten toewijst aan tabellen in een database, kunt u deze gegevens in volgende verwerkingsstappen voor bedrijfsinformatie gebruiken of direct toegang krijgen tot de relevante delen van een document die voldoen aan een semantische zoekopdracht.

Het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database bevat de configuratiegegevens voor de databaseverbinding en een beschrijving van welke aangepaste-analyseresultaten in welke tabellen en kolommen moeten worden opgeslagen. De tabel- en kolomnamen in het toewijzingsbestand moeten overeenkomen met de tabellen en kolommen die in de database zijn gemaakt.

Nadat u het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een database hebt weggeschreven, kunt u het bestand met behulp van de beheerconsole uploaden naar enterprise search.

Verwante taken

"Een toewijzingsbestand maken voor het toewijzen van de Common Analysis Structure aan een database" op pagina 47
Als u de analyseresultaten aan een database wilt toevoegen, moet u een toewijzingsbestand maken waarin de configuratiegegevens voor de databaseverbinding zijn opgenomen, en een beschrijving van welke aangepaste-tekstanalyseresultaten in welke tabellen en kolommen moeten worden opgeslagen.

Analyseresultaten opslaan in een database

Als u de geselecteerde analyseresultaten in een JDBC-database wilt opslaan, moet u een toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database schrijven, dat bepaalt welke analyseresultaten er moeten worden opgeslagen in een database, en moet u de benodigde JDBC-driverbibliotheken opnemen in het pad dat u in het toewijzingsbestand hebt gedefinieerd.

Ga als volgt te werk om analyseresultaten in een JDBC-database op te slaan:

1. Bepaal welke analyseresultaten u in de database wilt opslaan. Maak een database die de tabellen bevat met alle benodigde kolommen van de desbetreffende gegevenstypen.
2. Gebruik een XML-editor om het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database te schrijven met de database-configuratiegegevens en de analyseresultaten die u wilt opslaan. Als u wilt bepalen welke analyseresultaten er in het toewijzingsbestand moeten worden opgenomen, moet u weten welk onderliggend typesysteem wordt gebruikt wanneer de documenten worden verwerkt.
3. Plaats de JDBC-driverbibliotheken in een directory op het indexknooppunt waar ze vanuit het enterprise search-systeem toegankelijk zijn.
4. Vervolgens kunt u het toewijzingsbestand uploaden en selecteren met behulp van de beheerconsole van enterprise search.

Sets laadbestanden gebruiken

U kunt analyseresultaten ofwel rechtstreeks opslaan in een JDBC-database, of u kunt de verwerking zó configureren dat er sets laadbestanden worden gebruikt en dat de gegevens in een later stadium in de database worden geladen.

Het gebruik van sets laadbestanden heeft de volgende voordelen:

- In totaal kan een set laadbestanden nooit groter zijn dan de maximale bestandsgrootte die door het besturingssysteem wordt ondersteund
- U kunt beginnen met het laden van gegevens in de database zodra de set laadbestanden vol is, en u hoeft de documentparser niet te stoppen en opnieuw te starten om conflicten in de bestandstoegang te voorkomen

Het wisselen van de ene set laadbestanden naar de volgende gebeurt op documentniveau, zelfs als het document is verdeeld over meerdere Common Analysis Structures. Nadat er een document is verwerkt en als een laadbestand in de huidige set laadbestanden de gedefinieerde limiet overschrijdt, wordt er een nieuwe set laadbestanden gebruikt. Dit garandeert dat de set laadbestanden altijd consistent is. Nadat de content van de ene set laadbestanden in de database is

geladen, blijft het gegevensmodel consistent omdat alle vermeldingen in de mastertabel de overeenkomende vermeldingen in de databasetabel bevatten.

De laadbestanden en scriptbestanden worden aangegeven met de extensie .cur. Als een set laadbestanden wordt gesloten, krijgen de bestanden daarin de extensie .dat. Dit geeft aan dat de bestanden al naar een databaseserver kunnen worden gekopieerd of verplaatst terwijl de documentparser nog actief is.

U kunt de grootte van een laadbestand opgeven. Wanneer de groottelimiet voor laadbestanden is bereikt, wordt er een nieuwe set laadbestanden gestart. U geeft de grootte van laadbestanden op in de XML-elementsectie `<loadFile>` van het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database. De parameter loadFileSize wordt gedefinieerd met het element `<loadFileSize>` en wordt uitgedrukt in megabytes, waarbij geldt: 10 <= loadFileSize <= 10240 (10 MB <= loadFileSize <= 10 GB). Het element `<loadFileSize>` is optioneel. Als u geen waarde opgeeft, wordt de standaardwaarde gebruikt: 1024 MB (1 GB).

De afzonderlijke laadbestanden in een set worden genummerd met een tiencijferig getal dat aangeeft welk bestand tot welke set laadbestanden behoort. Een set laadbestanden wort gesloten wanneer:

- Een laadbestand in de set de gedefinieerde groottelimiet overschrijdt
- De verwerking is gestopt omdat de parser is gestopt of omdat er een fout is opgetreden

Als de parser opnieuw wordt gestart, wordt de verwerking hervat op het punt waar deze de vorige keer is gestopt, maar met een nieuwe set laadbestanden.

Een toewijzingsbestand maken voor het toewijzen van de Common Analysis Structure aan een database

Als u de analyseresultaten aan een database wilt toevoegen, moet u een toewijzingsbestand maken waarin de configuratiegegevens voor de databaseverbinding zijn opgenomen, en een beschrijving van welke aangepaste-tekstanalyseresultaten in welke tabellen en kolommen moeten worden opgeslagen.

Over deze taak

Het toewijzingsbestand voor het toewijzen van de Common Analysis Structure aan een database is een XML-bestand. Het volgende voorbeeld is gebaseerd op het typesysteem dat is gedefinieerd voor het politieverslagscenario.

In het voorbeeld worden alleen de politieverslagen en steden aan de database toegevoegd die in deze politiemisdaadverslagen voorkomen. In het voorbeeld wordt het gebruik getoond van geïntegreerde features en de toewijzing van het element `<constant>`.

```
<?xml version="1.0" encoding="UTF-8"?>
<cas2JdbcConfiguration xmlns="http://www.ibm.com/uima/consumer/jdbc/100/xml">
   <databaseConnection>
     <connectionUrl>db2://mijnMachine:mijnPoort/mijnDatabase</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>mijnGebruiker</username>
      <password>mijnWachtwoord</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>
```

Procedure

U kunt als volgt een toewijzingsbestand maken voor het toewijzen van de Common Analysis Structure aan een database:

1. Maak een XML-bestand. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma. Het XSD-schema voor het toewijzingsbestand heet CasToJDBCMapping.xsd en is tijdens de enterprise search-installatie opgeslagen in *ES_INSTALL_ROOT*/packages/uima/configuration/.
2. Neem de toewijzingen op in een element `<cas2JdbcConfiguration xmlns="http://www.ibm.com/uima/consumer/jdbc/100/xml">`. De naamruimte (opgegeven in het kenmerk xmlns) moet exact worden ingevoerd zoals weergegeven.
3. Voeg een `<databaseConnection>`-element toe dat alle configuratiegegevens voor de databaseverbinding bevat en voeg een `<cas2JdbcMappingSpec>`-element toe waarmee de toewijzingsregels worden beschreven voor de analyseresultaten die zijn opgeslagen in de database of de laadbestanden.
4. Voeg de volgende componentelementen toe aan het element `<databaseConnection>`:
   - Verplicht: een `<connectionUrl>`-element. Dit element bevat de URL van de databaseverbinding. Afhankelijk van de implementatie van het JDBC-driver kunt u lokale of niet-lokale toegang tot de database gebruiken.
   - Verplicht: een `<driver>`-element. Dit element bevat de naam van de JDBC-driverklasse, bijvoorbeeld `com.ibm.db2.jcc.DB2Driver` voor DB2, of `oracle.jdbc.driver.OracleDriver` voor Oracle.
   - Verplicht: een `<driverLibraries>`-element. Dit element bevat de driverbibliotheken. Elke bibliotheek wordt weergegeven in een `<driverLibrary>`-element. De bibliotheken zijn opgeslagen in de DB2- of Oracle-installatiedirectory. Voor DB2 zijn de bibliotheken `c:\your_db2_dir\db2jcc.jar`, `c:\your_db2_dir\db2jcc_license_cu.jar` en `c:\your_db2_dir\db2jcc_license_cisuz.jar`. Voor Oracle moet u de bibliotheek `c:\your_oracle_dir\classes12.zip` opnemen.

     Zorg dat de driverbibliotheken altijd hetzelfde onderhoudsniveau hebben als de DB2 appletserver.
   - Verplicht: een `<authentication>`-element. Dit element bevat de gebruikersnaam en het wachtwoord voor de database.
   - Optioneel: een `<loadFile>`-element. Dit element bevat de volgende componentelementen:
     - De directory voor het laadbestand, in een element `<loadFileDirectory>`.
     - Optioneel: De grootte van het laadbestand in het element `<loadFileSize>`. Het laadbestand is beperkt in grootte: 10 <= loadFileSize <= 10240 (10 MB <= loadFileSize <= 10 GB). Als u geen waarde opgeeft, wordt de standaardwaarde gebruikt: 1024 MB (1 GB).
     - De naam van het laadscript, in een element `<loadScript>`

     Als u geen `<loadFile>`-element opgeeft, worden alle gegevens direct in de database opgeslagen met behulp van JDBC.

     Als u databasespecifieke laadbestanden en -scripts gebruikt, moet u ook alle databaseconfiguratieparameters toevoegen.
5. Voeg de volgende componentelementen toe aan het element `<jdbcMappingSpec>`:
   - Optioneel: een `<skipCondition>`-element. Als er geen voorwaarde voor het overslaan van documenten is gedefinieerd, worden alle documenten verwerkt.

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

In het voorbeeld worden de documenten met een annotatie van het type com.ibm.uima.tt.DocumentAnnotation met de feature toBeProcessed ingesteld op nul, waarna deze documenten buiten beschouwing worden gelaten.

- Een <cas2JdbcMappings>-element waarin wordt aangegeven welke typen en features worden toegewezen aan welke databasetabellen en -kolommen. Het element bevat een sectie voor expliciete toewijzingen en een sectie voor impliciete toewijzingen.

6. Voeg een <explicitMappings>-element toe. Dit element is verplicht. Het element moet zijn voorzien van een of meer <explicitMappingRule>-elementen waarmee de expliciete toewijzingen worden gedefinieerd. Daarnaast kan het element alleen worden gedefinieerd voor annotatietypen en de bijbehorende subtypen. Als er een toewijzing is gedefinieerd in de sectie voor expliciete toewijzingen, worden alle annotaties die overeenkomen met de toewijzingsdefinitie in de database opgeslagen.
7. Optioneel: voeg een <implicitMappings>-element toe. Dit element biedt ondersteuning voor alle featurestructuurtypen. Als het element aanwezig is, moet dit minimaal één <implicitMappingRule>-element bevatten. Toewijzingen die in de sectie voor impliciete toewijzingen zijn gedefinieerd, worden alleen aan de database toegevoegd als naar de overeenkomende annotatietypen wordt verwezen door een andere annotatie die overeenkomt met een expliciete of een impliciete toewijzingsregel.

   Het doel van een impliciete toewijzing is het opslaan van de analyseresultaten die in een bepaalde context voorkomen. Als de toewijzing voor een annotatie van het type com.ibm.omnifind.types.City bijvoorbeeld impliciet is, worden alleen de steden waarnaar door de toewijzingsdefinitie com.ibm.omnifind.types.PoliceReport in de sectie met expliciete toewijzingen wordt verwezen, in de database opgeslagen. Dit betekent dat alleen de steden die in de misdaadverslagen voorkomen aan de database worden toegevoegd.

   Als er geen expliciete toewijzingsregel bestaat voor de annotatie City, worden alle steden aan de database toegevoegd. In beide gevallen geldt dat wanneer in verschillende politieverslagen naar een stad wordt verwezen, de stad slechts eenmaal aan de database wordt toegevoegd.
8. De elementen <explicitMappingRule> en <implicitMappingRule> moeten het kenmerk applyToSubtypes bevatten. Als dit kenmerk is ingesteld op Waar, wordt niet alleen de featurestructuur opgeslagen die in het <type>-element wordt weergegeven, maar ook alle afgeleide featurestructuren. Voeg de volgende componentelementen toe aan de elementen <explicitMappingRule> en <implicitMappingRule>:
   - Een <type>-element dat het featurestructuurtype bevat.
   - Een <table>-element dat het databaseschema en de tabelnaam bevat. De syntaxis komt na de regel schema.table_name, of alleen table_name als er geen schema is gedefinieerd.
   - Een element <featureMappings> met een of meer elementen <featureMapping> of één element <containerMapping>.
   - Optioneel: een <filter>-element dat een voorwaarde bevat die telkens wanneer de toewijzingsregel overeenkomt, wordt geëvalueerd. Als de voorwaarde is geëvalueerd en waar is, wordt de annotatie of de featurestructuur

in de database opgeslagen. In het voorbeeld worden alleen de politieverslagen met misdaden die in Los Angeles zijn gepleegd in de database opgeslagen.

9. De componentstructuur van het element <featureMapping> is afhankelijk van het feit of u een feature of een constante toewijst.

   Als u een feature of featurepad toewijst, omvatten de componentelementen:

   - Een <feature>-element met de naam van de feature. De feature moet zijn gedefinieerd voor de featurestructuur in het type-element. U kunt ook een gemaakt featurepad of een van de geïntegreerde features gebruiken.
   - Optioneel: een <length>-element met de lengte die voor strings is toegestaan in de opgegeven databasekolom. Langere strings worden afgekapt.
   - Een <column>-element met de naam van de kolom waarin de featurewaarde moet worden opgeslagen. Databasekolommen die niet in featuretoewijzingen worden gebruikt, maken gebruik van een standaardwaarde (meestal nul) die in de database is geconfigureerd.

     Zorg dat de waarde van het feature-element in het juiste type kolom is opgeslagen. In de volgende tabel ziet u welke UIMA typen overeenkomen met welke databasetypen.

Tabel 3. Toewijzingen tussen UIMA-typen en de bijbehorende databasetypen

UIMA-type of geïntegreerde feature	**Aanbevolen DB2-gegevenstype**	**Aanbevolen Oracle-gegevenstype**
Float	REAL	FLOAT
String	Varchar	Varchar2
Integer	INTEGER	INTEGER
uniqueId(), uniqueParentId()	CHAR(27)	CHAR(27)
objectId(), parentId()	CHAR(16)	CHAR(16)
docTimestamp()	BIGINT	LONG
fsId()	INTEGER	INTEGER

   Voor constanten zijn de featuretoewijzingselementen voor componenten als volgt:

   - Een <constant>-element dat de waarde van een constante bevat.
   - Een <column>-element met de naam van de kolom waaraan de constantewaarde is toegevoegd.

10. Het element <containerMapping> bevat de toewijzing voor een containertypefeature (array of lijst). Dit element moet alleen voor containertypen worden gebruikt. Het element bevat de volgende componentelementen:

    - Een <feature>-element met de naam van de feature. U kunt ook een gemaakt featurepad of een van de geïntegreerde features gebruiken.
    - Een <table>-element dat het databaseschema en de tabelnaam bevat. De syntaxis komt na de regel`schema.table_name` of alleen `table_name` als er geen schema is gedefinieerd.
    - Een of meer <featureMapping>-elementen die de namen bevatten van de featurestructuren en de kolomnamen waaraan de features zijn toegevoegd.

11. U kunt het XML-bestand nu opslaan en valideren met behulp van het geleverde schema.

Als u het XML-bestand hebt gemaakt, moet u dit uploaden naar enterprise search en het toewijzingsbestand voor het toewijzen van de Common Analysis Structure

aan een database selecteren met de andere selecties voor aangepaste analyse. Hiervoor gebruikt u de beheerconsole van enterprise search.

Verwante onderwerpen

"Databasetoewijzingen voor geselecteerde analyseresultaten" op pagina 45
Als u de aangepaste analyse hebt uitgevoerd uw documenten in enterprise search, kunt u de geselecteerde tekstanalyseresultaten opslaan in een JDBC-database.

"Featurepaden" op pagina 33
Met behulp van een featurepad kan toegang worden verkregen tot de featurewaarden in de Common Analysis Structures, vergelijkbaar met de toXPath-instructies waarmee toegang tot de XML-elementen in een XML-document kan worden verkregen.

Verwante verwijzing

"Filters" op pagina 36
Filters worden gebruikt voor het beperken van de toewijzingsregels in toewijzingsbestanden voor de toewijzing van de Common Analysis Structure aan de index en toewijzingsbestanden voor de toewijzing van de Common Analysis Structure aan een database. Alleen als het filter waar is, worden de analyseresultaten aan de index of een JDBC-tabel toegevoegd.

"Geïntegreerde features" op pagina 34
Geïntegreerde features zijn vooraf gedefinieerde featurenamen met een speciale semantiek. U kunt deze features gebruiken om toegang te krijgen tot informatie die niet in de featurestructuur zelf aanwezig is (bijvoorbeeld het type featurestructuur of de gedekte tekst van een annotatie). U kunt geïntegreerde features in een featurepad gebruiken als het laatste of het enige element.

"Voorbeeld van typesysteembeschrijving" op pagina 22
Met de typesysteembeschrijving worden de featurestructuren omschreven (de onderliggende gegevensstructuren die de analyseresultaten aangeven) die in aangepaste analyses worden gebruikt.

Containertypetoewijzing

Een containertype is een van de geïntegreerde array- of lijsttypen in de Common Analysis Structure. Een containertypetoewijzing is een manier om array- of lijstwaarden aan een relationele database toe te wijzen.

In het toewijzingsbestand voor de toewijzing van Common Analysis Structure aan database kunt u containertypen op twee manieren verwerken. Voor de eerste methode wordt gebruikgemaakt van de gedefinieerde, geïntegreerde features en een generieke koppelingstabel die de arrays of lijsten bevat die de waarden vormen voor een featuretoewijzingsregel. Als er verschillende arrays of lijsten in dezelfde koppelingstabel zijn opgeslagen, bevat de tabel geen informatie over de relatie van de opgeslagen gegevens.

Met de tweede methode wordt in de koppelingstabeldefinitie (gedefinieerd met een `<containerMapping>`-element) de relatie tussen de opgegeven informatie die u wilt weergegeven, expliciet aangegeven.

De toewijzing van een generieke koppelingstabel ziet er ongeveer als volgt uit. Er is een n:m-relatie tussen de politieverslagen en de verdachten, wat betekent dat één verdachte in meerdere politieverslagen kan voorkomen en dat in één politieverslag meerdere verdachten kunnen voorkomen.

De generieke tabel `sample.fsarray` in het voorbeeld is de koppelingstabel tussen politieverslagen en verdachten. Als er naast `com.ibm.omnifind.types.PoliceReport`

een ander toewijzingstype bestaat met een feature van type com.ibm.omnifind.types.FSArray, wordt dit ook aan deze tabel toegewezen. U kunt nog steeds een query uitvoeren op de tabel om de juiste relatie tussen een politieverslag en een verdachte op te halen, maar u kunt echter niet door naar de tabel te kijken, concluderen dat de tabel de relatie of koppeling tussen politieverslagen en mogelijke verdachten bevat.

```
<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>
```

In het volgende voorbeeld ziet u de databasetabellen op basis van de bovenstaande generieke toewijzingsregels.

Tabel 4. De tabel sample.policeReport

policeReportId	**suspectArrayId**	**city**
aaa...1	**bbb...1**	Springfield
aaa...2	**bbb...2**	Ladysmith

Tabel 5. De tabel sample.fsarray

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

Tabel 6. De tabel sample.suspect

suspectID	**lastname**	**description**
ccc...1	Brown	Dark complexion
ccc...2	Smith	Wears glasses
...	...	...

In het voorbeeld wordt de toewijzing voor featurestructuurarrays geïllustreerd. U kunt dit type toewijzing toepassen op StringArray, IntegerArray en FloatArray. Als u toewijzingsregels voor deze eenvoudige arrays opneemt, vervangt u `[]/uniqueId()` door `[]`.

Dezelfde methode voor generieke tabellen kan worden gebruikt voor lijsten van featurestructuren en eenvoudige, ingevoerde lijsten (StringList, IntegerList en FloatList).

Een eenvoudigere manier om relaties te verwerken, is gebruik te maken van een element voor expliciete containertoewijzingen waarmee de iteratie wordt gedefinieerd van de elementen in de arrays of lijsten.

Hieronder vindt u een voorbeeld van een toewijzing waarmee een expliciete koppelingstabel wordt aangegeven. Ook in dit voorbeeld bestaat er een n:m-relatie tussen de politieverslagen en de verdachten. Dit keer is de tabel `sample.reports_suspects` echter de koppelingstabel tussen politieverslagen en verdachten.

In dit scenario wordt geen rekening gehouden met array-ID's of de begin- en eindtoewijzingen voor lijsttypen. De koppelingstabel bevat één expliciete relatie.

```
<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>
```

Het element `<containerMapping>` wordt gebruikt om de iteratie te definiëren voor elementen in de array. In het voorbeeld bevat de koppelingstabel `sample.reports_suspects` een koppeling naar de kolommen `policeReportId` en `suspectId`. De elementen `<containerMapping>` moeten niet worden genest.

In volgende voorbeeld ziet u de databasetabellen op basis van expliciete toewijzingsregels voor de koppelingstabel.

Tabel 7. De tabel sample.policeReport

policeReportId	**city**
aaa...1	Springfield
aaa...2	Ladysmith

Tabel 8. De tabel sample.reports_suspect

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

Tabel 9. De tabel sample.suspect

suspectID	**lastname**	**description**
ccc...1	Brown	Dark complexion
ccc...2	Smith	Wears glasses
...	...	...

Verwante verwijzing

"Geïntegreerde features" op pagina 34
Geïntegreerde features zijn vooraf gedefinieerde featurenamen met een speciale semantiek. U kunt deze features gebruiken om toegang te krijgen tot informatie die niet in de featurestructuur zelf aanwezig is (bijvoorbeeld het type featurestructuur of de gedekte tekst van een annotatie). U kunt geïntegreerde features in een featurepad gebruiken als het laatste of het enige element.

Delen van een document ophalen die voldoen aan een semantische zoekopdracht

U kunt alleen de delen van een document ophalen die exact aan de zoekopdracht voldoen door de relevante featurestructuren toe te wijzen aan zowel de index als de database, en de spanne in de semantische zoekopdracht op te geven.

Als u alle exemplaren van een bepaald annotatietype in de zoekresultaten wilt weergeven (als u bijvoorbeeld alle personen wilt weergeven), neemt u een veldstijltoewijzing voor het annotatietype op en geeft u in het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index aan dat die veldstijltoewijzing als resultaat kan worden gegeven. Bijvoorbeeld:

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

In dit voorbeeld worden annotaties van het type `com.ibm.omnifind.types.Person` toegewezen aan de spanne Person in de enterprise search-index, waar deze annotaties kunnen worden weergegeven tijdens semantische zoekopdrachten. Daarnaast wordt de gedekte tekst van de annotaties, bijvoorbeeld de voor- en achternaam van personen, opgeslagen als veld dat als resultaat kan worden gegeven. Als u deze toelichtingswaarden wilt ophalen, roept u getFields(″Person″) op voor elk resultaatobject dat op basis van de zoekopdracht (op trefwoord of semantisch) als resultaat wordt gegeven. Met deze methode wordt een stringarray met de annotatiewaarden als resultaat gegeven, in dit voorbeeld de namen van personen.

Hoewel met deze methode alle exemplaren van het opgegeven annotatietype als resultaat worden gegeven, is de methode niet geschikt als u de resultaten-verwerking wilt beperken tot alleen de documenten die exact aan de zoekopdracht voldoen. Stel dat in een document vijf personen voorkomen. Met de semantische zoekopdracht '<sentence><person/>IBM</sentence> is de gebruiker echter alleen geïnteresseerd in de persoon die in dezelfde zin voorkomt als de term IBM. De gebruiker is niet geïnteresseerd in de overige personen.

Ga als volgt te werk om de featurestructuren weer te geven en te verwerken die exact voldoen aan de zoekopdracht:

1. Wijs de relevante featurestructuurtypen aan de enterprise search-index toe met behulp van de toewijzingsstijl voor annotaties. Bijvoorbeeld:

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

2. Wijs de relevante featurestructuurtypen toe aan JDBC-tabellen. Als onderdeel van de toewijzing moet u twee kolommen opnemen voor de document-URI en voor het featurestructuur-ID. Hoewel het mogelijk is alle featurestructuur-typen aan dezelfde databasetabel toe te wijzen, moet u elk type aan een andere tabel toewijzen. Bijvoorbeeld:

```
<explicitMappingRule applyToSubtypes="false">
 <type>com.ibm.omnifind.types.Person</type>
 <table>sample.person</table>
 <featureMappings>
  <featureMapping>
   <feature>objectId()</feature>
   <column>primaryId</column>
  </featureMapping>
  <!-- Bevat de gedekte tekst van de annotatie-->
  <featureMapping>
   <feature>coveredText()</feature>
   <column>personName</column>
  </featureMapping>
  <!-- Andere toewijzingen worden hier ingevoegd-->
  <!-- Voor toegang tot de relevante annotaties voor personen in de zoekresultaten-->
  <featureMapping>
   <feature>docUri()</feature>
   <column>docUri</column>
  </featureMapping>
  <featureMapping>
   <feature>fsId()</feature>
   <column>annotationId</column>
  </featureMapping>
 </featureMappings>
</explicitMappingRule>
```

3. Vervolgens kunt u de documenten crawlen, analyseren en indexeren.
4. Haal de ID's op van de exemplaren die voldoen aan de zoekopdracht. In de SIAPI (Search and Index API) wordt naar deze exemplaren verwezen als de doelelementen. Met een doelelement wordt de invoerspanne aangegeven die als resultaat moet worden gegeven. Het doelelement wordt als volgt gedefinieerd:
   - In XML-fragmenten wordt het doelelement aangegeven met behulp van het getalteken (#). Dit teken is slechts eenmaal toegestaan en kan op elke gewenste plek in de zoekopdracht voor XML-fragmenten voorkomen. Bij-voorbeeld: `$xmlf2::'<sentence><#person/>IBM</sentence>'`
   - In XPath is het doelelement standaard het laatste veld in de XPath-expressie.

- U kunt deze exemplaren weergeven met behulp van de methode Result.getProperty(″TargetElement″). De eigenschap die als resultaat wordt gegeven, is een string met de ID's van alle exemplaren, gescheiden door spaties. Elk exemplaar in de eigenschap kan worden omgezet in een geheel getal.

5. Met SIAPI worden de featurestructuren zelf niet als resultaat gegeven, alleen de ID's van de exemplaren. Deze ID's komen overeen met de waarde van fsId() die is opgeslagen in de databasetabel. Als u deze exemplaren en de bijbehorende gegevens wilt ophalen, moet de volgende handeling in het programma worden uitgevoerd:
   a. Selecteer de rechterdatabasetabel, afhankelijk van de spannenaam van het doelelement. In het voorbeeld bevat het programma een toewijzing van de persoon aan de tabel sample.Person. Deze informatie wordt opgehaald uit het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index, die de naam van de spanne levert, en het toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database, die de tabelnaam levert.
   b. Voor elk resultaatobject in het zoekresultaat moet u de volgende handelingen uitvoeren:
      1) Analyseer de string die op basis van Result.getProperty (″TargetElement″) als resultaat is gegeven om de ID's van de exemplaren te zoeken.
      2) Geef een SELECT-instructie op voor de tabel door de resulterende URI (toegankelijk via Result.getDocumentId()) als waarde in de kolom docUri te gebruiken en de ID's van de exemplaren als waarde in de kolom annotationId te gebruiken. De kolomnamen zijn afhankelijk van het toewijzingsbestand. In dit geval worden de kolomnamen uit het vorige voorbeeld gebruikt.

      De rijen die als resultaat worden gegeven, bevatten de gegevens die zijn opgeslagen voor de featurestructuur, bijvoorbeeld de gedekte tekst of specifieke kenmerken van de featurestructuur zoals de ″achternaam″ of ″geboorteplaats″.

Zorg ervoor dat de updates in de database worden gesynchroniseerd met de indexupdates in enterprise search. Als de database verouderde gegevens bevat (bijvoorbeeld omdat u databaselaadbestanden hebt gebruikt en u de database niet hebt bijgewerkt terwijl u de index wel hebt vernieuwd en opnieuw hebt georganiseerd), kunnen bepaalde ID's van exemplaren mogelijk niet meer worden gevonden in de database. In enterprise search wordt alleen de laatste documentversie in de index bijgehouden. Dit betekent dat de ID's van de exemplaren alleen geldig zijn voor het laatste document.

Als u meerdere versies van hetzelfde document in dezelfde databasetabel opslaat, kunnen er verschillende rijen aanwezig zijn die overeenkomen met dezelfde ID's voor exemplaren, maar voor verschillende versies van het document. In dit geval moet u een kolom voor de documentversie definiëren en de kolom van waarden voorzien met behulp van de programmalogica of geïntegreerde features zoals docTimestamp(). Op deze manier kunt u het resultaat filteren, zodat alleen de laatste documentversie wordt gebruikt.

Verwante onderwerpen

"Zoektermen in semantische zoekopdrachten" op pagina 60
Zoektermen in programma's voor semantische zoekopdrachten worden gecommuniceerd als ondoorzichtige termen.

Verwante taken

"Een toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index maken" op pagina 39
Met het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index kunt u bepalen welke analyseresultaten in de Common Analysis Structure u wilt indexeren om zoeken mogelijk te maken.

"Een toewijzingsbestand maken voor het toewijzen van de Common Analysis Structure aan een database" op pagina 47
Als u de analyseresultaten aan een database wilt toevoegen, moet u een toewijzingsbestand maken waarin de configuratiegegevens voor de databaseverbinding zijn opgenomen, en een beschrijving van welke aangepaste-tekstanalyseresultaten in welke tabellen en kolommen moeten worden opgeslagen.

Programma's voor semantische zoekopdrachten

In de enterprise search-index worden vier typen documentgegevens opgeslagen waarnaar u via zoekprogramma's kunt zoeken met behulp van de SIAPI-interface (Search and Index API).

De vier verschillende typen gegevens zijn:

- Tekstwoorden die in een document worden gevonden, bijvoorbeeld de zin *computersoftware*.
- Spannenamen, een XML-document met <Auteur>Jan</Auteur> levert bijvoorbeeld de spanne <Auteur>.
- Kenmerknamen, een XML-document met <Auteur Geboorteland=NL>Jan</Auteur> levert bijvoorbeeld het kenmerk ″Geboorteland″.
- Kenmerkwaarden, NL is bijvoorbeeld de waarde van het kenmerk ″Geboorteland″.

In de SIAPI-zoektaal kunt u gebruikmaken van zoektermen voor semantische zoekopdrachten. De zoekterm maakt gebruik van een twijgpatroon. Een twijg is een kleine boom met bladeren. Elk blad staat voor de vier typen gegevens (tekstwoorden, spannenamen, enzovoort). De interne knopen van de boom geven de relaties tussen de gegevens in een document aan. Er zijn vijf typen interne knopen waarmee relaties worden aangegeven:

- and
- or
- not
- in_the_span_of
- attribute_in_the_span_of

Een document voldoet aan de opgegeven semantische zoekterm als de bladeren in het document voorkomen en als de beperkingen worden gerespecteerd die op basis van de interne knopen (de gedefinieerde relaties) worden geleverd.

Met behulp van zoektermen in de semantische zoekopdracht kunt u documenten met hogere kwaliteit ophalen. U kunt niet alleen zoeken met behulp van booleaanse combinaties van woorden en annotaties, u kunt ook documenten ophalen waarin *Jan* bijvoorbeeld voorkomt in de spanne Auteur en waarin de termen *ibm* en *zoeken* in dezelfde zin voorkomen.

Zoektermen in semantische zoekopdrachten

Zoektermen in programma's voor semantische zoekopdrachten worden gecommuniceerd als ondoorzichtige termen.

In de SIAPI (Search and Index API) kunnen ondoorzichtige termen op twee manieren in de syntaxis worden uitgedrukt:

- Met XML- fragmenten
- Met Limited XPath

De zoekterm als XML-fragment ziet eruit als een evenwichtig fragment van een XML-document. Een dergelijke zoekterm wordt voorafgegaan door het teken van de ondoorzichtige term (`@xmlf2::`) gevolgd door de expressie van het XML-fragment, geplaatst tussen enkele aanhalingstekens ('...').

Limited XPath-zoektermen worden voorafgegaan door `@xmlxp::` gevolgd door de XPath-query, geplaatst tussen enkele aanhalingstekens ('...').

Net zoals voor algemene zoektermen in de SIAPI-interface (Search and Index API) geldt, kan elke term beschikken over een parameter waarmee de weergave kan worden gewijzigd:

Plusteken (+)
De term moet voorkomen.

Prefix =
De term moet exact overeenkomen.

Tilde: prefix (~)
Synoniemen van de zoekterm in aanmerking nemen.

Tilde: postfix (~)
Woorden met hetzelfde lemma als de zoekterm in aanmerking nemen.

Hekje (#)
De term wordt geaccentueerd.

In de volgende voorbeelden worden zoekopdrachten met XML-fragmenten geïllustreerd.

@xmlf2::'<City>Springfield</City>'
Hiermee worden documenten gevonden waarin de spanne (annotatie) City voorkomt met de string Springfield.

@xmlf2::'<Person gender="female"/>'
Hiermee worden documenten gevonden waarin een vrouwelijke persoon als annotatie voorkomt.

@xmlf2::'<Person><.or><@gender>female</@gender> <@title>Mrs</@title><@title>Ms</@title></.or></Person>'
Hiermee worden documenten gevonden waarin een persoon als vrouw op basis van geslacht of titel wordt aangegeven.

@xmlf2::'<Person gender="male" role="suspect"/> <PoliceReport><@crimeDescription><.or>robbery theft</.or>-accident </@crimeDescription></PoliceReport> <City>Springfield<.or> <@district>Brynston</@district><@district>Brooklyn</@district></.or></City>'
Hiermee worden documenten gevonden waarin mannelijke personen voorkomen die als verdachten worden beschouwd en waarin de toelichting PoliceReport voorkomt met de string *robbery* of *theft* in het kenmerk

crimeDescription, maar niet de string *accident*. De documenten moeten een annotatie City bevatten die het tekstwoord *Springfield* bevat en een annotatie die als kenmerk het district *Brynston* of *Brooklyn* heeft.

De bijbehorende XPath-zoekopdrachten bevatten de volgende structuur:

@xmlxp::'//City ftcontains ("Springfield")'
Hiermee worden documenten gevonden die de spanne (annotatie) City bevatten met de string *Springfield*.

@xmlxp::'//PoliceReport[City ftcontains("Springfield")]'
Hiermee worden documenten gevonden die in de spanne PoliceReport de spanne (annotatie) City bevatten met de string *Springfield*.

@xmlxp::'//Person[@gender="female" or @title ftcontains("Ms") or @title ftcontains("Mrs")]'
Hiermee worden documenten gevonden waarin een vrouwelijke persoon als annotatie voorkomt. In het kenmerk voor het geslacht ("gender") moet de waarde exact gelijk zijn, maar in het titelkenmerk ("title") hoeven *Ms* en *Mrs* niet exact overeen te komen met de kenmerkwaarde.

Ondersteuning voor synoniemen in zoekprogramma's

U kunt de zoekresultaten uitbreiden door te zoeken naar documenten die synoniemen van de zoektermen bevatten.

Synoniemen bevatten doorgaans termen die uit meerdere woorden bestaan, zoals productnamen (bijvoorbeeld *WebSphere Information Integrator OmniFind*). Termen die uit meerdere woorden bestaan en die in het synoniemenwoordenboek zijn opgenomen, worden op de juiste wijze geïdentificeerd in de gebruikersquery's en hoeven niet tussen aanhalingstekens te worden geplaatst.

De SIAPI (Search and Index API) voor enterprise search biedt ondersteuning voor verschillende manieren om synoniemen van zoektermen te zoeken:

- De SIAPI-querysyntaxis ondersteunt het gebruik van de tilde (~) voor de uitbreiding van synoniemen. Als de gebruiker deze operator invoegt voor een zoekterm, wordt de uitbreiding van synoniemen voor het woord uitgevoerd. Met de query `~WAS` worden bijvoorbeeld documenten als resultaat gegeven waarin WebSphere Application Server wordt besproken en andere synoniemen die voor deze afkorting bestaan.
- De uitbreiding van synoniemen kan worden ingeschakeld met behulp van de SIAPI-interface voor de uitbreiding van synoniemen in een zoekprogramma. Zoektermen kunnen automatisch worden uitgebreid zodat synoniemen worden opgenomen of u kunt ervoor zorgen dat het zoekprogramma opties bevat waarmee gebruikers kunnen aangeven of de synoniemen van de zoektermen in de zoekresultaten moeten worden opgenomen.

  Tijdens de automatische uitbreiding van synoniemen wordt het zoekproces voor synoniemen uitgevoerd voor alle zoekwoorden. In de zoekresultaten worden de documenten weergegeven die de zoektermen of de synoniemen van de zoektermen bevatten. De SIAPI biedt tevens ondersteuning aan het genereren van een lijst van synoniemuitbreidingen voor de ingediende query.

Gebruik de ondersteuning voor synoniemen niet voor tekst die wordt verwerkt met n-gramsegmentering.

Een XML-bestand voor synoniemen maken

Als u de query's in enterprise search wilt uitbreiden zodat deze synoniemen van de zoektermen bevatten, moet u in een XML-bestand aangeven welke woorden synoniemen van elkaar zijn. Dit XML-bestand wordt gebruikt voor het bouwen van een binair woordenboekbestand dat u naar enterprise search kunt uploaden en aan de gewenste collecties kunt toewijzen.

Over deze taak

Het XML-bestand met de synoniemen moet voldoen aan een bepaald schema. Dit is een voorbeeld van een XML-bestand voor synoniemen:

```
<?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>
```

Beperkingen

U moet de woorden die synoniemen van elkaar zijn (de <synonym>-elementen), groeperen in een <synonymgroup>-element. Een synoniem kan wel spaties, maar geen leestekens bevatten, zoals komma's (,) of verticale strepen (|), omdat deze tekens fouten in de enterprise search-querysyntaxis kunnen veroorzaken.

U moet alle mogelijke vervoegingen opnemen van de termen die u als synoniemen opneemt (bijvoorbeeld het enkelvoud en het meervoud van een woord). U hoeft de normalisatie van de term niet op te nemen, zoals het verwijderen van accenten of umlauten (normalisaties worden in enterprise search automatisch verwerkt) en het is ook niet nodig om hoofd- en kleine-lettervarianten van de term op te nemen. Als u bijvoorbeeld de term météo als synoniem wilt opnemen, hoeft u niet ook de term METEO op te nemen.

Procedure

Ga als volgt te werk om een lijst met synoniemen voor enterprise search te maken:

1. Maak een XML-bestand. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma. Het XSD-schema voor het XML-bestand heet synonyms.xsd en is tijdens de enterprise search-installatie opgeslagen in *ES_INSTALL_ROOT*/packages/uima/configuration/.
2. Voeg een <synonymgroup>-element toe, voeg vervolgens een <synonym>-element in voor elk woord dat moet worden beschouwd als synoniem van de andere woorden in de groep met synoniemen.

   Neem de toewijzingen op in een element <synonymgroups xmlns="http://www.ibm.com/of/822/synonym/xml">. De naamruimte (opgegeven in het kenmerk xmlns) moet exact worden ingevoerd zoals wordt weergegeven.
3. Herhaal de voorgaande stappen totdat u alle synoniemen hebt opgenomen die u wilt gebruiken voor het doorzoeken van documenten in een enterprise search-collectie.
4. Sla het XML-bestand op en sluit dit vervolgens af.

Als u het XML-bestand hebt gemaakt, moet u dit converteren naar een synoniemenwoordenboek, zodat het woordenboek aan het enterprise search-systeem kan worden toegevoegd.

Een synoniemenwoordenboek maken

Als u in een XML-bestand een lijst met synoniemen hebt gemaakt of bijgewerkt, moet u het XML-bestand converteren naar een binair synoniemenwoordenboek.

Over deze taak

Als u een synoniemenwoordenboek wilt maken, kunt u het opdrachtregelhulpprogramma essyndictbuilder gebruiken dat bij WebSphere II OmniFind Edition wordt geleverd. Dit hulpprogramma bevindt zich in de directory *ES_INSTALL_ROOT*/bin.

De invoer voor het hulpprogramma is het XML-bestand waarin uw synoniemen zijn opgenomen en de uitvoer van het hulpprogramma is een synoniemenwoordenboek. Aan het woordenboek moet de extensie `.dic` zijn toegewezen. Bijvoorbeeld: `c:\mijnwoordenboeken\producten.dic`.

Voor beide bestanden is de standaardlocatie de directory waarin het script is gestart. Als er al een woordenboek met dezelfde naam aanwezig is, treedt er een fout op in het script.

De maximale grootte van een `.dic`-bestand in enterprise search is 8 MB.

Procedure

Ga als volgt te werk om een synoniemenwoordenboek voor enterprise search te maken:

1. Meld u bij de indexserver aan als enterprise search-beheerder. Dit gebruikers-ID is opgegeven bij de installatie van WebSphere II OmniFind Edition.
2. Voer de volgende opdracht in, waarbij *XML-bestand* het volledige pad is naar het XML-bestand dat de lijst met synoniemen bevat en *DIC-bestand* het volledige pad is naar het synoniemenwoordenboek.

   AIX, Linux, or Solaris: `essyndictbuilder.sh` *XML-bestand DIC-bestand*
   Windows: `essyndictbuilder.bat` *XML-bestand DIC-bestand*

Als u een synoniemenwoordenboek hebt gemaakt, gebruikt u de beheerconsole van enterprise search om het woordenboek aan het enterprise search-systeem toe te voegen en te koppelen aan een of meer collecties.

Alleen het gegenereerde .dic-bestand wordt naar het enterprise search-systeem geüploaded. Zorg dat het bron-XML-bestand is opgeslagen in een omgeving waarvan de toegang wordt beheerd en dat u de juiste backupprocedure gebruikt. U hebt dit XML-bestand nodig als u het synoniemenwoordenboek wilt bijwerken.

Aangepaste stopwoordenboeken

U kunt een bedrijfsspecifieke woordenlijst definiëren met woorden die uit query's worden verwijderd, zodat de zoekrelevantie wordt verbeterd.

Enterprise search bevat twee typen ondersteuning voor stopwoorden:

- Taalspecifieke stopwoordenherkenning waarmee alle veelgebruikte algemene woorden zoals *een* en *de* uit query's met meerdere woorden worden verwijderd. Het stopwoordenboek dat voor elke taal aanwezig is, kan niet door gebruikers worden gewijzigd. De stopwoordenherkenning wordt automatisch uitgevoerd voor alle query's, zodat de zoekrelevantie wordt verbeterd.
- Door de gebruiker gedefinieerde of aangepaste stopwoordenherkenning waarmee de woorden uit de bedrijfsspecifieke woordenlijst uit query's worden verwijderd. Dit door de beheerder gedefinieerde stopwoordenboek kan alleen een speciale woordenlijst bevatten. Het door de gebruiker gedefinieerde stopwoordenboek is geen vervanging van de taalspecifieke stopwoordenboeken in enterprise search waarin de algemene woorden zijn opgenomen.Door de gebruiker gedefinieerde stopwoordenboeken zijn taal-onafhankelijk.

De door de gebruiker gedefinieerde stopwoordenboeken bevatten doorgaans termen die uit meerdere woorden bestaan, zoals productnamen (bijvoorbeeld *WebSphere Information Integrator OmniFind*). Termen die uit meerdere woorden bestaan en die in het stopwoordenboek zijn opgenomen, worden op de juiste wijze geïdentificeerd in de gebruikersquery's en hoeven niet tussen aanhalingstekens te worden geplaatst.

Samengestelde termen in Germaanse talen worden ook op de juiste wijze geïdentificeerd in query's. Een samengestelde term is een combinatie van twee of meer woorden die als één woord wordt gebruikt. Lexicale samengestelde termen zoals *Reisbureau* worden niet beschouwd als samengestelde termen.

Samengestelde termen in een query worden opgesplitst in de afzonderlijke termen waaruit de samengestelde term bestaat. Als een van deze afzonderlijke termen in het stopwoordenboek voorkomt, wordt de samengestelde term niet uit de query verwijderd.

Met de zoekterm `Verzekeringspolis` worden bijvoorbeeld documenten als resultaat gegeven waarin de samengestelde termen *Levensverzekeringspolis* en *Aansprakelijkheidsverzekeringspolis* voorkomen. Zelfs als het woord *Polis* in het stopwoordenboek voorkomt, wordt de samengestelde zoekterm `Verzekeringspolis` niet uit de query verwijderd.

U moet de bedrijfsspecifieke woordenlijst in een XML-bestand opnemen en dit bestand vervolgens converteren naar een stopwoordenboek, zodat het woordenboek aan het enterprise search-systeem kan worden toegevoegd.

U kunt aangeven welk stopwoordenboek u wilt gebruiken in de beheerconsole van enterprise search. Voor elke collectie kunt u één stopwoordenboek selecteren. Een stopwoordenboek kan door verschillende collecties gemeenschappelijk worden gebruikt.

Een XML-bestand voor stopwoorden maken

Als u bedrijfsspecifieke termen uit de query's wilt verwijderen, moet u in een XML-bestand aangeven welke woorden als stopwoorden moeten worden beschouwd.

Over deze taak

Het XML-bestand met de stopwoorden moet voldoen aan een bepaald schema dat in het XML-document is opgegeven. Dit is een voorbeeld van een XML-bestand voor stopwoorden:

```
<?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>
```

Beperkingen

Een stopwoord kan wel spaties, maar geen leestekens bevatten, zoals komma's (,) of verticale strepen (|), omdat deze tekens fouten in de enterprise search-querysyntaxis kunnen veroorzaken.

U hoeft de normalisatie van de term niet op te nemen, zoals het verwijderen van accenten of umlauten (normalisaties worden in enterprise search automatisch verwerkt). Als u bijvoorbeeld de term météo als stopwoord wilt opnemen, hoeft u niet ook de term METEO op te nemen.

Procedure

Ga als volgt te werk om een lijst met stopwoorden voor enterprise search te maken:

1. Maak een XML-bestand. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma om het XML-bestand te valideren.Het XSD-schema voor het XML-bestand heet stopWords.xsd en is tijdens de enterprise search-installatie opgeslagen in *ES_INSTALL_ROOT*/packages/uima/configuration/.
2. Voeg een `<stopWord>`-element toe voor elk woord dat als stopwoord moet worden beschouwd.

   Neem de toewijzingen op in een element `<stopWords xmlns="http://www.ibm.com/of/83/stopwordbuilder/xml">`. De naamruimte (opgegeven in het kenmerk xmlns) moet exact worden ingevoerd zoals wordt weergegeven.
3. Herhaal de voorgaande stappen totdat u alle stopwoorden hebt opgenomen die uit de query's moeten worden verwijderd wanneer gebruikers de enterprise search-collecties doorzoeken.
4. Sla het XML-bestand op en sluit dit vervolgens af.

Als u het XML-bestand hebt gemaakt, moet u dit converteren naar een stopwoordenboek, zodat het woordenboek aan het enterprise search-systeem kan worden toegevoegd.

Een stopwoordenboek maken

Als u in een XML-bestand een lijst met door de gebruiker gedefinieerde stopwoorden hebt gemaakt of bijgewerkt, moet u het XML-bestand converteren naar een stopwoordenboek.

Over deze taak

Als u een stopwoordenboek wilt maken, kunt u het opdrachtregelhulpprogramma esstopworddictbuilder gebruiken dat bij WebSphere II OmniFind Edition is geleverd. Dit hulpprogramma bevindt zich in de directory *ES_INSTALL_ROOT*/bin.

De invoer voor het hulpprogramma is het XML-bestand waarin de stopwoorden zijn opgenomen en de uitvoer van het hulpprogramma is een stopwoordenboek. Aan het woordenboek moet de extensie `.dic` zijn toegewezen. Bijvoorbeeld: `c:\mijnwoordenboeken\productstopwoorden.dic`.

Voor beide bestanden is de standaardlocatie de directory waarin het script is gestart. Als er al een woordenboek met dezelfde naam aanwezig is, treedt er een fout op in het script.

De maximale grootte van een `.dic`-bestand in enterprise search is 8 MB.

Procedure

Ga als volgt te werk om een stopwoordenboek te maken voor enterprise search:

1. Meld u bij de indexserver aan als enterprise search-beheerder. Dit gebruikers-ID is opgegeven bij de installatie van WebSphere II OmniFind Edition.
2. Voer de volgende opdracht in, waarbij *XML-bestand* het volledige pad is naar het XML-bestand dat de lijst met stopwoorden bevat en *DIC-bestand* het volledige pad is naar het stopwoordenboek.

   AIX, Linux, or Solaris: `esstopworddictbuilder.sh` *XML-bestand DIC-bestand*
   Windows: `esstopworddictbuilder.bat` *XML-bestand DIC-bestand*

Als u een stopwoordenboek hebt gemaakt, gebruikt u de beheerconsole van enterprise search om het woordenboek aan het enterprise search-systeem toe te voegen en te koppelen aan een of meer collecties.

Alleen het gegenereerde .dic-bestand wordt naar het enterprise search-systeem geüploaded. Zorg dat het bron-XML-bestand is opgeslagen in een omgeving waarvan de toegang wordt beheerd en dat u de juiste backupprocedure gebruikt. U hebt dit XML-bestand nodig als u het stopwoordenboek wilt bijwerken.

Aangepaste gewogen woordenboeken

U kunt bepaalde termen (die uit één of meerdere woorden bestaan) definiëren waarmee de rangwaarde van het document waarin de term voorkomt, wordt verhoogd of verlaagd.

Elke term in het gewogen woordenboek is gekoppeld aan een wegingsfactor die kan liggen tussen -10 en +10. Aan de termen die u afzonderlijk in de resultatendocumenten wilt weergeven, wordt een hogere wegingsfactor toegewezen, terwijl aan de termen die u niet in de documenten wilt weergeven of die u wilt combineren met termen met een hogere wegingsfactor, een lagere waarde wordt toegewezen. De waarden -1, 0 en 1 hebben geen effect op de weging.

Als een zoekterm die in het gewogen woordenboek met een bepaalde wegingsfactor is opgenomen in het opgehaalde document wordt weergegeven, is de rangwaarde van het document verhoogd of verlaagd (afhankelijk van de wegingsfactor). De wegingsfactor die aan een term wordt toegewezen, is relatief omdat deze ook wordt beïnvloed door andere factoren. Als de term X wordt gewogen door B1, de term Y door B2, en B1 > B2, dan is weging (X) >= weging (Y).

Een gewogen woord bevat doorgaans termen die uit meerdere woorden bestaan, zoals productnamen (bijvoorbeeld *WebSphere Information Integrator OmniFind*). Termen die uit meerdere woorden bestaan en die in het gewogen woordenboek zijn opgenomen, worden op de juiste wijze geïdentificeerd in de gebruikersquery's en hoeven niet tussen aanhalingstekens te worden geplaatst.

Gewogen woordenboeken zijn taal-onafhankelijk.

Samengestelde termen in Germaanse talen worden ook op de juiste wijze geïdentificeerd in query's. Een samengestelde term is een combinatie van twee of meer woorden die als één woord wordt gebruikt. Lexicale samengestelde termen zoals *Reisbureau* worden niet beschouwd als samengestelde termen.

Samengestelde termen in een query worden opgesplitst in de afzonderlijke termen waaruit de samengestelde term bestaat. Als er wegingsfactoren bestaan voor de afzonderlijke termen van een samengestelde term, worden de opgehaalde documenten gerangschikt, hoewel de toegewezen factor lager is dan wanneer de term op zichzelf zou staan (en dus geen deel zou uitmaken van een samengestelde term). Op deze manier wordt het zoekbereik vergroot, wat met name nuttig is wanneer slechts een aantal documenten is gevonden waarin de volledige samengestelde term voorkomt.

Stel dat met de zoekterm `Verzekeringspolis` documenten als resultaat worden gegeven waarin de samengestelde termen *Levensverzekeringspolis* en *Aansprakelijkheidsverzekeringspolis* voorkomen. Als het woord *Polis* voorkomt in het gewogen woordenboek, wordt een wegingsfactor toegewezen aan het document dat de samengestelde zoekterm `Verzekeringspolis` bevat.

U moet de termen met de bijbehorende wegingsfactoren in een XML-bestand opnemen en dit bestand vervolgens converteren naar een gewogen woordenboek, zodat het woordenboek aan het enterprise search-systeem kan worden toegevoegd.

U kunt aangeven welk gewogen woordenboek u wilt gebruiken in de beheerconsole van enterprise search. Voor elke collectie kan één gewogen woordenboek worden geselecteerd. Een gewogen woorden boek kan door verschillende collecties gemeenschappelijk worden gebruikt.

Een XML-bestand voor gewogen woorden maken

Als u het belang van bepaalde resultatendocumenten wilt verhogen of verlagen, moet u in een XML-bestand opgeven welke woorden van invloed kunnen zijn op de rangschikking van het document.

Over deze taak

Het XML-bestand met de gewogen woorden moet voldoen aan een bepaald schema dat in het XML-bestand is opgegeven. Dit is een voorbeeld van een XML-bestand voor gewogen woorden:

```
<?xml version="1.0" encoding="UTF-8"?>
<boostTerms xmlns="http://www.ibm.com/of/83/boostbuilder/xml">
    <!-- gewogen termen groeperen op wegingsfactor-->
   <boostTermList boost="5">
    <!-- voor elke term kan de uitbreiding van synoniemen afzonderlijk worden opgegeven-->
     <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>
```

Beperkingen

Termen met dezelfde wegingsfactor kunt u groeperen in een element `<boostTermList>`, maar een wegingsfactor kan meerdere keren voorkomen, bijvoorbeeld wanneer u de gewogen woorden alfabetisch wilt sorteren in het XML-bestand.

Een gewogen woord kan wel spaties, maar geen leestekens bevatten, zoals komma's (,) of verticale strepen (|), omdat deze tekens fouten in de enterprise search-querysyntaxis kunnen veroorzaken.

Gewogen termen kunnen varianten hebben, zoals acroniemen of afkortingen. U kunt alle varianten opnemen in het gewogen woordenboek. Als u echter van plan bent naast een gewogen woordenboek ook een synoniemenwoordenboek te gebruiken en u de termen en de bijbehorende varianten al aan het synoniemenwoordenboek hebt toegevoegd, hoeft u deze varianten niet nogmaals aan de lijst met gewogen woorden toe te voegen. In plaats daarvan stelt u het kenmerk `useVariants` in op Waar voor de variant die u aan het gewogen woordenboek wilt toevoegen. Alle varianten van deze term in het synoniemenwoordenboek die in een van de opgehaalde documenten worden weergegeven, zijn van invloed op de rangwaarde die aan deze documenten wordt toegewezen.

U hoeft de normalisatie van de term niet op te nemen, zoals het verwijderen van accenten of umlauten (normalisaties worden in enterprise search automatisch verwerkt). Als u bijvoorbeeld de term météo als gewogen woord wilt opnemen, hoeft u niet ook de term METEO op te nemen.

Procedure

Ga als volgt te werk om een lijst met gewogen woorden voor enterprise search te maken:

1. Maak een XML-bestand. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma. Het XSD-schema voor het XML-bestand heet boostTerms.xsd en is tijdens de enterprise search-installatie opgeslagen in *ES_INSTALL_ROOT*/packages/uima/configuration/.
2. Neem de toewijzingen op in een element `<boostTerms xmlns="http://www.ibm.com/of/83/boostbuilder/xml">`. De naamruimte (opgegeven in het kenmerk xmlns) moet exact worden ingevoerd zoals wordt weergegeven.
3. Voeg het element `<boostTermList>` toe om alle termen te groeperen die de opgegeven wegingsfactor gemeenschappelijk gebruiken.

   De wegingsfactor kan liggen tussen -10 en 10. Bijvoorbeeld `<boostTermList boost="-5">` of `<boostTermList boost="5">`.

   Op basis van de opgegeven wegingsfactor wordt het belang van de documenten die de opgegeven termen bevatten, verhoogd of verlaagd.
4. Voeg het element `<term>` toe voor elke term die de opgegeven wegingsfactor gebruikt.

   Als u varianten van een gewogen woord wilt opnemen die al in een synoniemenwoordenboek worden weergegeven, stelt u het kenmerk `useVariants` in het element `<term>` in op Waar. De standaardwaarde is Onwaar. Als er geen varianten in het synoniemenwoordenboek kunnen worden gevonden, verschijnt er geen foutbericht.
5. Herhaal de voorgaande stappen totdat u alle termen hebt opgegeven die u als gewogen woorden wilt gebruiken bij het doorzoeken van enterprise search-collecties .
6. Sla het XML-bestand op en sluit dit vervolgens af.

Als u het XML-bestand hebt gemaakt, moet u dit converteren naar een gewogen woordenboek, zodat het woordenboek aan het enterprise search-systeem kan worden toegevoegd.

Een gewogen woordenboek maken

Als u in een XML-bestand een lijst met gewogen woorden hebt gemaakt of bijgewerkt, moet u het XML-bestand converteren naar een gewogen woordenboek.

Over deze taak

Als u een gewogen woordenboek wilt maken, kunt u het opdrachtregelhulpprogramma esboostworddictbuilder gebruiken dat bij WebSphere II OmniFind Edition wordt geleverd. Dit hulpprogramma bevindt zich in de directory ES_INSTALL_ROOT/bin.

De invoer voor het hulpprogramma is het XML-bestand waarin uw gewogen woorden zijn opgenomen en de uitvoer van het hulpprogramma is een gewogen woordenboek. Aan het woordenboek moet de extensie `.dic` zijn toegewezen. Bijvoorbeeld: `c:\mijnwoordenboeken\gewogenwoordenproduct.dic`.

Voor beide bestanden is de standaardlocatie de directory waarin het script is gestart. Als er al een woordenboek met dezelfde naam aanwezig is, treedt er een fout op in het script.

De maximale grootte van een `.dic`-bestand in enterprise search is 8 MB.

Procedure

Ga als volgt te werk om een gewogen woordenboek te maken voor enterprise search:

1. Meld u bij de indexserver aan als enterprise search-beheerder. Dit gebruikers-ID is opgegeven bij de installatie van WebSphere II OmniFind Edition.
2. Voer de volgende opdracht in, waarbij *XML_file* het volledige pad is naar het XML-bestand dat de lijst met gewogen woorden bevat en *DIC_file* het volledige pad is naar het gewogen woordenboek. Als u ook een synoniemenwoordenboek wilt gebruiken, voegt u achter de naam van het gewogen woordenboek het volledige pad naar het synoniemenwoordenboek toe. U kunt synoniemenwoordenboeken een naam geven, maar dit is niet verplicht.

   UNIX: `esboostworddictbuilder.sh` *XML-bestand DIC-bestand SYNDIC-bestand*
   Windows: `esboostworddictbuilder.bat` *XML-bestand DIC-bestand SYNDIC-bestand*

Als u een gewogen woordenboek hebt gemaakt, gebruikt u de beheerconsole van enterprise search om het woordenboek aan het enterprise search-systeem toe te voegen en te koppelen aan een of meer collecties.

Alleen het gegenereerde .dic-bestand wordt naar het enterprise search-systeem geüploaded. Zorg dat het bron-XML-bestand is opgeslagen in een omgeving waarvan de toegang wordt beheerd en waarvoor de juiste backupprocedure wordt gebruikt. U hebt dit XML-bestand nodig als u het gewogen woordenboek wilt bijwerken.

Verwante taken

"Een synoniemenwoordenboek maken" op pagina 64
Als u in een XML-bestand een lijst met synoniemen hebt gemaakt of bijgewerkt, moet u het XML-bestand converteren naar een binair synoniemenwoordenboek.

Tekstanalyse in enterprise search

De tekstanalyse die bij enterprise search wordt geleverd, bevat functies voor de detectie en segmentering van de documenttaal.

Tijdens de verwerking van documenten in enterprise search wordt de taal van het document bepaald en wordt de invoertekst in afzonderlijke eenheden of tokens opgesplitst.

De querytaal moet tijdens zoekopdrachten handmatig worden geselecteerd door de gebruiker of het programma. De querystring wordt gesegmenteerd, geanalyseerd en doorzocht in de index.

De analyse van document- en querystrings kan worden opgesplitst in:

- Basisondersteuning zonder woordenboeken. Deze ondersteuning omvat witruimte- en n-gramsegmentering. Basisondersteuning zonder woordenboeken omvat ook zinsegmentering.
- Taalkundige ondersteuning op basis van woordenboeken. Deze ondersteuning bevat segmentering en lemmatisering van woorden en zinnen.

  Taalkundige verwerking omvat lexicale analyse, het proces voor het maken van alternatieve representaties van de invoertekst waarmee alle beschikbare woordenboekgegevens worden gekoppeld aan de tokens die in de invoertekst zijn herkend. Als u gebruikmaakt van geavanceerde taalverwerking, wordt de kwaliteit van zoekopdrachten aanzienlijk verbeterd.

  Verwante onderwerpen

  "Taalidentificatie"
  Voordat de processen voor woord- en zinsegmentering, tekennormalisatie of -lemmatisering kunnen worden uitgevoerd, moet de taal van het brondocument in enterprise search worden vastgesteld.

  "Taalkundige ondersteuning voor segmentering zonder woordenboeken" op pagina 77
  Voor documenten in talen die niet worden ondersteund door de lexicale analysetechnologie, bevat enterprise search basisondersteuning in de vorm van op Unicode gebaseerde witruimte- en n-gramsegmentering.

Taalidentificatie

Voordat de processen voor woord- en zinsegmentering, tekennormalisatie of -lemmatisering kunnen worden uitgevoerd, moet de taal van het brondocument in enterprise search worden vastgesteld.

In enterprise search kunnen de volgende talen automatisch worden gedetecteerd:

Tabel 10. Talen die worden ondersteund door automatische taalidentificatie

Afrikaans	Arabisch	Balinees
Baskisch	Catalaans	Chinees (Traditioneel en Vereenvoudigd)
Tsjechisch	Deens	Nederlands
Engels	Fins	Frans
Duits	Grieks	Hebreeuws

Tabel 10. Talen die worden ondersteund door automatische taalidentificatie (vervolg)

IJslands	Iers (Gaelic)	Italiaans
Japans	Koreaans	Maleis
Noors (Bokmål)	Pools	Portugees
Roemeens	Russisch	Spaans
Zweeds	Tagalog	Thais
Turks	Vietnamees	

Met de taalkundige processen in enterprise search wordt de taal van een bron-document gedetecteerd tijdens de indexering, niet tijdens de queryverwerking.

In enterprise search kunt u aangeven dat de taal van documenten automatisch moet worden gedetecteerd of u kunt de taal aangeven die moet worden gebruikt.

Als u automatische taaldetectie selecteert en de taal van een document niet kan worden vastgesteld door de parser, wordt de taal gebruikt die u hebt opgegeven tijdens het maken van de crawler in de beheerconsole van enterprise search.

Als u geen automatische taaldetectie selecteert, wordt altijd de opgegeven taal gebruikt. U kunt de documenttaal opgeven in de eigenschappen van de crawler in de beheerconsole van enterprise search. De standaardtaal is Engels.

Documenten waarvoor geen taalspecifieke woordenboeken bestaan, worden verwerkt met behulp van een basistaalonafhankelijke technologie zoals de witruimte- en n-gramsegmentering.

De enterprise search-taaldetectietechnologie is het meest geschikt voor documenten waarin één taal wordt gebruikt. Als in een document meerdere talen worden gebruikt, wordt geprobeerd vast te stellen wat de dominantste taal is in het document. De analyseresultaten zijn echter niet altijd even bevredigend.

De taal van een document kan worden gebruikt om de zoekresultaten te beperken tot alleen de documenten waarin een bepaalde taal wordt gebruikt. Als u in een meertalige collectie bijvoorbeeld documenten zoekt over Jacques Chirac, kunt u opgeven dat alleen de documenten in het Frans in de zoekresultaten moeten worden opgenomen. Het instellen van de taal van uw uitvoerdocumenten is een geavanceerde zoekoptie die kan worden geselecteerd in de beheerconsole van enterprise search.

Verwante onderwerpen

"Tekstanalyse in enterprise search" op pagina 75
De tekstanalyse die bij enterprise search wordt geleverd, bevat functies voor de detectie en segmentering van de documenttaal.

"Taalkundige ondersteuning voor segmentering zonder woordenboeken" op pagina 77
Voor documenten in talen die niet worden ondersteund door de lexicale analysetechnologie, bevat enterprise search basisondersteuning in de vorm van op Unicode gebaseerde witruimte- en n-gramsegmentering.

Taalkundige ondersteuning voor segmentering zonder woordenboeken

Voor documenten in talen die niet worden ondersteund door de lexicale analysetechnologie, bevat enterprise search basisondersteuning in de vorm van op Unicode gebaseerde witruimte- en n-gramsegmentering.

Op Unicode gebaseerde witruimtesegmentering
Voor deze methode van taalkundige verwerking wordt de witruimte (of spatie) tussen woorden als scheidingsteken tussen woorden gebruikt.

N-gram-segmentering
Voor deze methode van taalkundige verwerking worden overlappende reeksen met *n*-tekens als één woord beschouwd. Deze eenvoudige segmenteringsmethode is geschikt voor veel ophaaltaken.

Deze methoden werken onafhankelijk van de taalwoordenboeken en maken geen gebruik van geavanceerde taalkundige verwerkingstechnologie, zoals de basisvormreductie.

N-gramsegmentering wordt gebruikt voor talen zoals Thais waarin geen witruimten aanwezig zijn om te worden gebruikt als scheidingstekens. Dezelfde methode is van toepassing op het Hebreeuws en Arabisch. Hoewel deze twee talen gebruikmaken van scheidingstekens voor witruimten, zorgt u met n-gramsegmentering voor betere resultaten dan de standaardvorm van op Unicode gebaseerde witruimtesegmentering.

Bij het maken van een collectie kunt u desgewenst ook kiezen voor tokenisering van Chinese en Japanse documenten met behulp van n-gramsegmentering.

Om te zorgen dat tijdens de n-gramsegmentering alle witruimten worden verwijderd (bijvoorbeeld tekens voor tabs of nieuwe regels), moet u de parameterinstellingen in het bestand collection.properties in *ES_NODE_ROOT*/master_config/<Collectie-ID>.parserdriver aanzetten voordat u de analyse van het document start. De volgende parameters zijn vereist voor het verwijderen van witruimten:

- **removeCjNewLineChars**: Als u deze parameter instelt op *true*, worden reeksen nieuwe-regel- en tabtekens die tussen Chinese of Japanse tekens staan, verwijderd. De standaardinstelling is `removeCjNewlineChars=false`.
- **removeCjNewLineCharsMode**: Als u deze parameter instelt op *all*, worden alle witruimtetekens verwijderd, ongeacht hun context. Dit betekent dat witruimtetekens bijvoorbeeld ook uit Nederlandse teksten worden verwijderd. Als u met deze optie wilt werken, moet u de parameter toevoegen aan het eigenschappenbestand. Alleen `removeCjNewlineCharsMode=all` is geldig; alle andere waarden worden genegeerd.

  Verwante onderwerpen

  "Tekstanalyse in enterprise search" op pagina 75
  De tekstanalyse die bij enterprise search wordt geleverd, bevat functies voor de detectie en segmentering van de documenttaal.

  "Taalidentificatie" op pagina 75
  Voordat de processen voor woord- en zinsegmentering, tekennormalisatie of -lemmatisering kunnen worden uitgevoerd, moet de taal van het brondocument in enterprise search worden vastgesteld.

Numerieke tekens tokeniseren als n-gramtokens

Als u, naast dubbelbytetekens, ook numerieke tekens wilt tokeniseren als n-gramtokens, moet u een bepaalde parameter instellen in het descriptorbestand voor de witruimte- en n-gramtokenizer.

Over deze taak

Standaard worden alle numerieke tekens door de witruimte- en n-gramtokenizer behandeld als tokens die door witruimte worden gesegmenteerd. Als u numerieke tekens echter wilt segmenteren als n-gramtokens, moet u de instelling van de n-gramwerkstand in het descriptorbestand van de annotator wijzigen. Het is niet mogelijk om deze instelling te wijzigen met behulp van de beheerconsole van enterprise search.

Procedure

De standaardinstelling voor de n-gramwerkstand wordt ″normal″ genoemd en behandelt numerieke tekens en SBCS-tekens als tekens die worden gesegmenteerd door witruimte. U schakelt de numerieke n-gramwerkstand als volgt in:

1. Stop de parser voor uw collectie.
2. Stop de runtime voor uw collectie.
3. Open het annotatordescriptorbestand jtok.xml in de directory *ES_NODE_ROOT*/master_config/<Collectie-ID>.parserdriver/specifiers. Collectie-ID is het ID dat voor de collectie is opgegeven (of door het systeem is toegewezen) toen de collectie werd gemaakt.
4. Verander de instelling van de parameter **NgramMode** van normal in numeric.
5. Start de parser voor uw collectie opnieuw.
6. Start de runtime opnieuw.

Taalkundige ondersteuning voor op woordenboeken gebaseerde segmentering

Als de taal van een document op de juiste wijze is gedetecteerd en er taalspecifieke woordenboeken beschikbaar zijn, wordt de bijbehorende taalkundige verwerking toegepast.

Segmentering is het proces waarbij de ingevoerde tekst wordt onderverdeeld in afzonderlijke lexicale eenheden. Tijdens dit proces wordt een aantal van de volgende taalkundige verwerkingsactiviteiten uitgevoerd:

Woordsegmentering
Woordsegmentering wordt gebruikt voor talen waarin geen gebruikt wordt gemaakt van witruimten (of scheidingstekens) tussen woorden, zoals het Japans en Chinees.

Lemmatisering
Lemmatisering is een taalkundige verwerking waarmee het lemma wordt bepaald voor elke woordvorm die in de tekst voorkomt. Het *lemma* van een woord bestaat uit de basisvorm plus de vervoegde vormen die dezelfde woordsoort delen. Het lemma voor *gaan* omvat bijvoorbeeld onder meer *ga*, *gaat*, *ging*, *gegaan*, en *gaande*. Lemma's voor zelfstandige naamwoorden bestaan uit zowel het enkelvoud als het meervoud (zoals *kalf* en *kalveren*). Lemma's voor bijvoeglijke naamwoorden bestaan uit zowel de vergelijkende als de overtreffende vorm (zoals *goed*, *beter* en *best*).

Lemma's voor voornaamwoorden bestaan uit de verschillende vormen van hetzelfde voornaamwoord (zoals *ik*, *me*, *mij* en *mijn*).

Voor lemmatisering is een woordenboek voor zowel indexering als voor zoekopdrachten vereist.

In enterprise search worden de lemma's en de vervoegde woorden geïndexeerd en worden alle vervoegde woorden in een query gelemmatiseerd. Met lemmatisering verbetert u de kwaliteit van zoekopdrachten, omdat er documenten worden gevonden die varianten bevatten van een vervoegd woord in de query. Zo worden documenten met het woord *musea* gevonden als de query het woord *museum* bevat.

Samentrekkingen splitsen
U kunt de kwaliteit van zoekopdrachten verbeteren door samentrekkingen te identificeren en deze te splitsen in de verschillende onderdelen. Bijvoorbeeld:

wouldn't wordt gesplitst in *would + not*
Horse's wordt gesplitst in *Horse + 's*

Identificatie van cliticum
Een cliticum is een speciale vorm van een samentrekking. U kunt de kwaliteit van zoekopdrachten verbeteren door de onderdelen van de samentrekkingen vast te stellen. Een *cliticum* is een element dat zich gedraagt als een affix en een woord. Een cliticum is echter moeilijk te identificeren omdat dit ook deel uitmaakt van de woordformatie. In tegenstelling tot andere morfologische (woordstructuur) fenomenen, komt een cliticum voor in een syntactische structuur. Voor de verbinding van een cliticum aan een ander woord wordt geen rekening gehouden met de regels voor woordformaties. Bijvoorbeeld:

reparti-lo-emos bestaat uit de onderdelen *repartir + lo + emos*
l'avenue bestaat uit de onderdelen *le + avenue*
dell'arte bestaat uit de onderdelen *dello + arte*.

Niet-alfabetische tekenherkenning
In het taalkundige proces worden niet-alfabetische tekens herkend. Afhankelijk van de interne taalafhankelijke logica worden bepaalde niet-alfabetische tekens teruggestuurd als afzonderlijke lexicale eenheden van verschillende typen en worden bepaalde tekens samengevoegd.

In het geval van een cliticum worden apostroffen beschouwd als woordonderdelen, in het geval van onbekende afkortingen worden ze gezien als punten. URL's, e-mailadressen en datum worden opgesplitst in diverse tokens.

Herkenning van afkortingen
In de taalkundige processen worden afkortingen herkend die in het woordenboek zijn opgenomen als één lexicale eenheid. Als de afkorting niet in het woordenboek is opgenomen, wordt de afkorting herkend als een lexicaal item, maar is voor de afkorting geen bijbehorende informatie in het woordenboek aanwezig.

Het op de juiste wijze herkennen van afkortingen is van groot belang voor de herkenning van zinnen. Zo betekent de punt aan het einde van de afkorting niet noodzakelijkerwijs het einde van een zin.

Herkenning van markering voor einde van de zin
Markeringen voor het einde van zinnen worden in de taalkundige processen op de juiste wijze geïdentificeerd voor zinsegmentering.

Voor de volgende talen is op woordenboeken gebaseerde taalkundige ondersteuning beschikbaar:

Tabel 11. Ondersteunde talen

Arabisch	Italiaans
Chinees (Vereenvoudigd en Traditioneel)	Japans
Tsjechisch	Koreaans
Deens	Noors (Bokmål)
Nederlands	Pools
Engels	Portugees (Portugal en Brazilië)
Fins	Russisch
Frans (Frankrijk en Canada)	Spaans
Duits (Duitsland en Zwitserland)	Zweeds
Grieks	

Verwante onderwerpen

"Woordsegmentering in het Japans"
Als het tekstdocument of de querystring is herkend als Japanse tekst, wordt in enterprise search de relevante woordsegmentering uitgevoerd met behulp van een morfologische analysetechnologie die is geoptimaliseerd voor het Japans.

"Orthografische varianten in het Japans" op pagina 81
In het Japans worden vele orthografische varianten gebruikt. Katakana-varianten zijn het belangrijkst, omdat Katakana vaak wordt gebruikt voor de spelling en uitspraak van buitenlandse woorden. Veel Katakana-varianten worden algemeen gebruikt in het Japans.

Woordsegmentering in het Japans

Als het tekstdocument of de querystring is herkend als Japanse tekst, wordt in enterprise search de relevante woordsegmentering uitgevoerd met behulp van een morfologische analysetechnologie die is geoptimaliseerd voor het Japans.

Een voorbeeld van deze optimalisatie is het opsplitsen van woorden. In het Japans wordt een groot aantal samengestelde woorden gebruikt. Deze woorden worden opgesplitst in tokens van optimale grootte zodat betere zoekresultaten worden behaald. Vervoegde woorden en voorzetsels worden ook opgesplitst om de zoekprestaties te verbeteren.

Verwante onderwerpen

"Taalkundige ondersteuning voor op woordenboeken gebaseerde segmentering" op pagina 78
Als de taal van een document op de juiste wijze is gedetecteerd en er taalspecifieke woordenboeken beschikbaar zijn, wordt de bijbehorende taalkundige verwerking toegepast.

"Orthografische varianten in het Japans" op pagina 81
In het Japans worden vele orthografische varianten gebruikt. Katakana-varianten zijn het belangrijkst, omdat Katakana vaak wordt gebruikt voor de spelling en uitspraak van buitenlandse woorden. Veel Katakana-varianten worden algemeen gebruikt in het Japans.

Orthografische varianten in het Japans

In het Japans worden vele orthografische varianten gebruikt. Katakana-varianten zijn het belangrijkst, omdat Katakana vaak wordt gebruikt voor de spelling en uitspraak van buitenlandse woorden. Veel Katakana-varianten worden algemeen gebruikt in het Japans.

In enterprise search wordt een woordenboek met varianten gebruikt om typische Katakana-varianten aan de bijbehorende basisvormen te koppelen (vergelijkbaar met een lemma), zodat alle documenten worden gevonden, inclusief die met orthografische varianten van het Katakana-woord in de zoekstring.

Daarnaast biedt enterprise search ondersteuning voor typische Okurigana-varianten, Kanji-woordeinden die in Hiragana zijn geschreven.

Verwante onderwerpen

"Taalkundige ondersteuning voor op woordenboeken gebaseerde segmentering" op pagina 78
Als de taal van een document op de juiste wijze is gedetecteerd en er taalspecifieke woordenboeken beschikbaar zijn, wordt de bijbehorende taalkundige verwerking toegepast.

"Woordsegmentering in het Japans" op pagina 80
Als het tekstdocument of de querystring is herkend als Japanse tekst, wordt in enterprise search de relevante woordsegmentering uitgevoerd met behulp van een morfologische analysetechnologie die is geoptimaliseerd voor het Japans.

Stopwoorden verwijderen

In enterprise search worden alle stopwoorden, zoals veelgebruikte woorden (bijvoorbeeld *een* en *de*), uit query's met meerdere woorden verwijderd om de zoekprestaties te verbeteren.

In het Japans is de herkenning van stopwoorden gebaseerd op grammaticale informatie. Zo herkent enterprise search bijvoorbeeld of een woord een zelfstandig naamwoord of een werkwoord is. Voor andere talen maakt enterprise search gebruik van speciale lijsten.

Stopwoorden worden tijdens de verwerking van de query NIET verwijderd:

- Als alle woorden in de query stopwoorden zijn. Als alle zoektermen tijdens de verwerking van stopwoorden worden verwijderd, blijft er niets over. Om te zorgen dat er toch zoekresultaten kunnen worden afgebeeld, wordt het verwijderen van stopwoorden uitgeschakeld als alle zoektermen stopwoorden zijn. Als het woord *het* bijvoorbeeld een stopwoord is en u geeft alleen *het* op als zoekopdracht, dan bestaan de zoekresultaten uit documenten waarin het woord *het* voorkomt. Zoekt u naar *het wak*, dan bestaan de resultaten uit documenten waarin het woord *wak* voorkomt.
- Als het woord in de query wordt voorafgegaan door een plusteken (+).
- Als het woord deel uitmaakt van een exacte overeenkomst.
- Als het woord binnen een woordcombinatie staat, bijvoorbeeld "Ik reed in het wak".

Verwante onderwerpen

"Tekennormalisatie" op pagina 82
Tekennormalisatie is een proces waarmee het terughalen kan worden verbeterd. Dit houdt in dat meer documenten worden opgehaald, zelfs als de documenten niet exact aan de query voldoen.

Tekennormalisatie

Tekennormalisatie is een proces waarmee het terughalen kan worden verbeterd. Dit houdt in dat meer documenten worden opgehaald, zelfs als de documenten niet exact aan de query voldoen.

In enterprise search wordt Unicode-normalisatie gebruikt. Het gaat hierbij onder meer om de normalisatie van Aziatische tekens van halve breedte naar tekens met volledige breedte.

Daarnaast worden Katakana midden-punten verwijderd, die in het Japans worden gebruikt als scheidingstekens voor samenstellingen.

Andere vormen van tekennormalisatie zijn:

Normalisatie van hoofdletters
Hiermee worden bijvoorbeeld documenten met de tekst *VS* gevonden bij zoekopdrachten naar *vs*.

Umlautuitbreiding
Hiermee worden documenten met de tekst *schoen* gevonden bij zoekopdrachten naar *schön*.

Verwijderen van accenten
Hiermee worden documenten met het teken *é* gevonden bij zoekopdrachten naar *e*.

Verwijderen van andere diakritische tekens
Hiermee worden documenten met het teken *ç* gevonden bij zoekopdrachten naar *c*.

Ligatuuruitbreiding
Hiermee worden documenten met het teken *Æ* gevonden bij zoekopdrachten naar *ae*.

Alle normalisaties werken in beide richtingen. U kunt documenten zoeken die *vs* bevatten wanneer u zoekt naar *VS*, documenten die woorden bevatten met *e* wanneer u zoekt naar *é*, enzovoort. Deze normalisaties kunnen ook worden gecombineerd. U kunt bijvoorbeeld documenten zoeken die *météo* bevatten wanneer u zoekt naar *METEO*.

Deze normalisaties zijn gebaseerd op Unicode-tekeneigenschappen en zijn niet taalafhankelijk. Enterprise search biedt bijvoorbeeld ondersteuning voor het verwijderen van diakritische tekens voor Hebreeuws en Arabisch.

Verwante onderwerpen

"Stopwoorden verwijderen" op pagina 81
In enterprise search worden alle stopwoorden, zoals veelgebruikte woorden (bijvoorbeeld *een* en *de*), uit query's met meerdere woorden verwijderd om de zoekprestaties te verbeteren.

Expressieannotator

De expressieannotator maakt het mogelijk om aangepaste tekstanalyse uit te voeren zonder dat het nodig is uw eigen tekstanalyseprogramma te implementeren. Op basis van een set regels (expressies of ″regular expressions″) die u zelf kunt opstellen, is de expressieannotator in staat informatiestructuren in tekstdocumenten en annotaties van de gedetecteerde informatie te maken in de Common Analysis Structure.

De expressieannotator detecteert op entiteiten of eenheden informatie in tekstdocumenten, bijvoorbeeld telefoonnummers, productcodes, kamernummers en adressen. Dit gebeurt op basis van expressies. Als een van de expressies een match oplevert in de tekst van een document, maakt de expressieannotator de annotaties die het overeenkomende stuk informatie beslaan. Deze annotaties worden opgeslagen in de Common Analysis Structure en kunnen later worden doorzocht. Om dit doorzoeken mogelijk te maken, moeten deze analyseresultaten worden toegevoegd aan de enterprise search-index. Dit gebeurt met behulp van een toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index. Als alternatief kan er een toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database worden gemaakt, om de annotaties op te slaan in een JDBC-database.

De set regels (expressies) die u definieert, worden opgeslagen in een XML-configuratiebestand (ook wel een regelsetbestand genoemd). De expressieannotator bevat de analysesoftware waarmee deze expressies worden verwerkt en ondersteunt de expressiesyntaxis van Java 1.4.

De typesysteembeschrijving van de expressieannotator moet een definitie geven van de typen en features van de annotaties die worden gebruikt en gemaakt door de expressieannotator. Afhankelijk van de complexiteit van het toepassingsgebied van de expressieannotator (bijvoorbeeld: er zijn meer typen vereist dan er beschikbaar zijn in de meegeleverde expressieannotator), kan het zijn dat er aanvullende invoer- en uitvoermogelijkheden moeten worden gedefinieerd in de descriptor van de expressieannotator. De typen die in de descriptor worden gebruikt, moeten overeenkomen met de typen in de typesysteembeschrijving van de annotator.

De expressieannotator is in enterprise search opgenomen in de vorm van een PEAR-bestand (Processing Engine ARchive) dat is geconfigureerd met voorbeelden van regels voor het herkennen van telefoonnummers, URL's en e-mailadressen.

Verwante onderwerpen

"Het regelsetbestand" op pagina 86
In de expressieannotator bepaalt het XML-regelsetbestand de regels, in de vorm van expressies, die worden gebruikt voor het analyseren (parsing) van het tekstdocument.

Verwante taken

"Expressieregels definiëren" op pagina 87
De regelset bepaalt welke expressies er worden vergeleken met de tekst in het document en de acties die door de expressieannotator moeten worden ondernomen als het patroon overeenkomt.

Verwante verwijzing

"De annotatordescriptor" op pagina 92
De XML-descriptor van de expressieannotator bevat beschrijvende informatie over de expressieannotator. Die informatie is nodig om de annotator te kunnen laten werken.

"Loggen" op pagina 95
Alle logberichten van de expressieannotator worden bijgeschreven in het logbestand van de huidige collectie.

Eenvoudige semantische zoekopdracht met expressieannotator

Enterprise search wordt geleverd met een expressieanalyseprogramma dat vooraf is geconfigureerd met een groep regels die dat programma in staat stelt telefoonnummers, URL's en e-mailadressen in documenten te herkennen.

Deze voorbeeldconfiguratie van het expressieanalyseprogramma kunt u gebruiken om enterprise search in staat te stellen feitelijke telefoonnummers in documenten te vinden zonder in die documenten te zoeken naar het trefwoord *telefoonnummer*. Om te kunnen zoeken naar die door de expressieannotator worden herkend, wordt er tevens een voorbeeld toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index bijgeleverd. Bovendien wordt er een eenvoudige methode gedemonstreerd volgens welke u krachtige semantische zoekopdrachten kunt opgeven met behulp van eenvoudige trefwoorden. Deze methode maakt gebruik van de synoniemenfunctie van enterprise search, een functie die eenvoudige query's op trefwoorden automatisch uitbreidt tot semantische zoekopdrachten. Een voorbeeld synoniemenwoordenboek waarmee dit mechanisme wordt gedemonstreerd, is eveneens bijgeleverd. Alle bestanden die u nodig hebt om de expressieannotator met de voorbeeldconfiguratie te gebruiken, vindt u in ES_INSTALL_ROOT/packages/uima/regex.

Voor veel toepassingsscenario's hoeft u de expressieregels in de voorbeeldconfiguratie slechts een beetje te wijzigen om de expressieannotator volledig aan te passen aan uw wensen.

Wilt u de annotator volledig aanpassen, dan wordt u nadrukkelijk geadviseerd gebruik te maken van de UIMA SDK. Voor dit doel is de expressieannotator tevens opgenomen in het pakket met basisannotators van enterprise search, in ES_INSTALL_ROOT/packages/uima/.

Verwante taken

"Eenvoudige semantische zoekopdracht met expressieannotator mogelijk maken" op pagina 85
Om eenvoudige semantische zoekopdrachten met behulp van synoniemen mogelijk te maken, moet u de expressieannotator, het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index en het voorbeeld synoniemenwoordenboek opnemen in uw enterprise search-systeem en deze resources vervolgens koppelen aan uw collectie.

"De expressieannotator aanpassen" op pagina 90
U kunt de voorbeeldconfiguratie van de expressieannotator zodanig aanpassen dat deze nieuwe entiteiten herkent (bijvoorbeeld serienummers van producten). Het is mogelijk om de expressieregels voor bestaande entiteiten aan te passen (bijvoorbeeld voor het herkennen van bedrijfs-specifieke telefoonnummers). Dit laatste doet u door kleinere wijzigingen aan te brengen in de voorbeeldbestanden voor regelset en typesysteem.

"De resultaten van de basisannotator en aangepaste tekstanalyse bekijken" op pagina 13
Om te zien welke analyseresultaten de parser en de annotators in enterprise

search hebben geproduceerd, moet u de eigenschappen van de document-collectie zodanig aanpassen dat er een leesbare XML-versie wordt gemaakt van de analyseresultaten die worden opgeslagen in de Common Analysis Structure.

Eenvoudige semantische zoekopdracht met expressieannotator mogelijk maken

Om eenvoudige semantische zoekopdrachten met behulp van synoniemen mogelijk te maken, moet u de expressieannotator, het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index en het voorbeeld synoniemenwoordenboek opnemen in uw enterprise search-systeem en deze resources vervolgens koppelen aan uw collectie.

Daarna worden uw documenten tijdens de analysefase (parsing) door de expressieannotator verwerkt, worden de resultaten van de aangepaste analyse door het indexeerprogramma toegevoegd aan de index en kan het bijgeleverde semantische synoniemenwoordenboek door de zoekservice worden gebruikt voor het zoeken naar resultaten van de aangepaste analyse. Dit kan dan via eenvoudige trefwoorden die automatisch worden uitgebreid tot semantische zoekopdrachten.

Procedure

U maakt semantisch zoeken als volgt mogelijk:

1. Voeg het aangepaste tekstanalyseprogramma voor expressies, of_regex.pear in de directory *ES_INSTALL_ROOT*/packages/uima/regex, met behulp van de beheerconsole van enterprise search toe aan het enterprise search-systeem.
2. Koppel het tekstanalyseprogramma voor expressies aan uw collectie.
3. Voeg het toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index (of_sample_regex_cas2index.xml in de directory *ES_INSTALL_ROOT*/packages/uima/regex) toe. Hiermee worden de resultaten van aangepaste analyse (annotaties) die door de expressieannotator zijn geproduceerd, toegewezen aan doorzoekbare spannes in de enterprise search-index. Vervolgens kunt u deze spannes doorzoeken met XML-fragment- of XPath-query's.
4. Crawl, analyseer en indexeer uw collectie. Op dit punt, dus nadat het indexeren is voltooid, zou u via het zoekprogramma een XML-zoekopdracht kunnen opgeven met behulp van een XML-fragmentexpressie, bijvoorbeeld `@xmlf2::'<#phonenumber>'`. Het doel van semantisch zoeken op synoniemen is echter om het mogelijk te maken dat u zoekopdrachten kunt opgeven zoals `Barbara telefoonnummer` en te zorgen dat het systeem deze vertaalt in Barbara @xmlf2::'<#phonenumber>'.
5. Voeg het meegeleverde binaire voorbeeld synoniemenwoordenboek genaamd of_sample_synonym_dic.dic in de directory *ES_INSTALL_ROOT*/packages/uima/regex met behulp van de beheerconsole toe aan het enterprise search-systeem. U kunt de bron van het voorbeeld XML-woordenboek aanpassen, of u kunt dat woordenboek als basis gebruiken voor het maken van uw eigen woordenboek en dat woordenboek vervolgens, met behulp van het tool essyndictbuilder, converteren naar een nieuw woordenboekbestand. Het voorbeeld XML-synoniemenwoordenboek heet of_sample_synonym_dic.xml en is te vinden in *ES_INSTALL_ROOT*/packages/uima/regex.
6. Koppel het synoniemenwoordenboek aan uw collectie en start (of herstart) de zoekservice voor uw collectie.
7. Ga naar het zoekprogramma en selecteer de optie voor het automatisch zoeken naar synoniemen met behulp van semantische uitbreiding. Nadat u deze optie

hebt ingeschakeld, zorgt het zoekprogramma ervoor dat uw elementaire trefwoordquery's worden omgezet in XML-fragmentquery's en dat er expressies worden opgenomen waarmee doorzoekbare spannes waarmee telefoonnummers, e-mailadressen en URL's worden aangegeven, kunnen worden gevonden.

8. Geef in het zoekprogramma een zoekopdracht op waarmee u zoekt naar een telefoonnummer, bijvoorbeeld `barbara telefoonnummer`. De query zoekt nu naar documenten die de twee trefwoorden *barbara* en *telefoonnummer* bevat, en naar documenten die niet alleen het trefwoord *barbara* bevatten, maar ook spannes van van cijfers en andere tekens die voldoen aan de expressie die is gedefinieerd voor telefoonnummers. De trefwoorden en telefoonnummers die worden gevonden, worden in de zoekresultaten geaccentueerd.

In het bijgeleverde voorbeeld synoniemenwoordenboek kunt u zien welke trefwoorden er worden vertaald in semantische query's.

```
<?xml version="1.0" encoding="UTF-8"?>
<synonymgroups xmlns="http://www.ibm.com/of/822/synonym/xml">
  <synonymgroup>
<synonym>telefoonnummer</synonym>
<synonym>telefoon</synonym>
<synonym>telefoonnr.</synonym>
<synonym>tel.nr.</synonym>
<synonym>@xmlf2::'&lt;#phonenumber/&gt;'</synonym>
  </synonymgroup>
  <synonymgroup>
<synonym>facsimilenummer</synonym>
<synonym>faxnummer</synonym>
<synonym>faxnr</synonym>
<synonym>fax</synonym>
<synonym>@xmlf2::'&lt;#phonenumber/&gt;'</synonym>
  </synonymgroup>
  <synonymgroup>
<synonym>e-mailadres</synonym>
<synonym>emailadres</synonym>
<synonym>@xmlf2::'&lt;#email/&gt;'</synonym>
  </synonymgroup>
  <synonymgroup>
<synonym>URL</synonym>
<synonym>unified resource locator</synonym>
<synonym>internetadres</synonym>
<synonym>@xmlf2::'&lt;#url/&gt;'</synonym>
  </synonymgroup>
</synonymgroups>
```

Verwante onderwerpen

"Eenvoudige semantische zoekopdracht met expressieannotator" op pagina 84
Enterprise search wordt geleverd met een expressieanalyseprogramma dat vooraf is geconfigureerd met een groep regels die dat programma in staat stelt telefoonnummers, URL's en e-mailadressen in documenten te herkennen.

Het regelsetbestand

In de expressieannotator bepaalt het XML-regelsetbestand de regels, in de vorm van expressies, die worden gebruikt voor het analyseren (parsing) van het tekstdocument.

De regels geven in sequentiële volgorde aan op welke plaatsen in de documenttekst de annotator naar iets specifieks moet zoeken en welke actie er moet worden uitgevoerd als er een overeenkomst wordt gevonden.

Als de expressieannotator wordt aangeroepen, wordt het XML-regelsetbestand dat de expressiepatronen bevat, gecompileerd en vergeleken met delen van de documenttekst. Als er een overeenkomst of gedeeltelijke overeenkomst wordt gevonden, wordt er een annotatie gemaakt die bij een specifieke regel hoort en wordt deze annotatie opgeslagen in de Common Analysis Structure.

De typen die in de regels worden gebruikt, moeten worden gedefinieerd in de typesysteembeschrijving van de expressieannotator.

De expressieannotator verwerkt de regels een voor een, te beginnen met de eerste regel in het XML-regelsetbestand. Voor elke regel wordt de bijbehorende gecompileerde expressie vergeleken met de annotaties die in een eerdere stap zijn gemaakt, bijvoorbeeld annotaties die zijn gemaakt door annotators die het document hebben verwerkt voordat de expressieannotator aan de slag ging. De annotaties die overeenkomen met de regels, moeten van hetzelfde type zijn als de invoertypen die zijn opgegeven in de descriptor van de expressieannotator.

Als er een overeenkomst wordt gevonden, moet het type annotatie dat wordt gemaakt in een regel die aan de beurt is, eveneens worden opgegeven als een geldig uitvoertype in de descriptor van de expressieannotator. De nieuwe annotaties die door een eerdere regel zijn gemaakt, kunnen worden gebruikt als invoerannotaties voor regels die later in de XML-regelset aan de beurt komen.

Verwante onderwerpen

"Expressieannotator" op pagina 83
De expressieannotator maakt het mogelijk om aangepaste tekstanalyse uit te voeren zonder dat het nodig is uw eigen tekstanalyseprogramma te implementeren. Op basis van een set regels (expressies of ″regular expressions″) die u zelf kunt opstellen, is de expressieannotator in staat informatiestructuren in tekstdocumenten en annotaties van de gedetecteerde informatie te maken in de Common Analysis Structure.

Verwante taken

"Expressieregels definiëren"
De regelset bepaalt welke expressies er worden vergeleken met de tekst in het document en de acties die door de expressieannotator moeten worden ondernomen als het patroon overeenkomt.

Verwante verwijzing

"De annotatordescriptor" op pagina 92
De XML-descriptor van de expressieannotator bevat beschrijvende informatie over de expressieannotator. Die informatie is nodig om de annotator te kunnen laten werken.

"Loggen" op pagina 95
Alle logberichten van de expressieannotator worden bijgeschreven in het logbestand van de huidige collectie.

Expressieregels definiëren

De regelset bepaalt welke expressies er worden vergeleken met de tekst in het document en de acties die door de expressieannotator moeten worden ondernomen als het patroon overeenkomt.

Over deze taak

Het XML-bestand met de regelset moet de regelsyntaxis volgen die in het onderstaande voorbeeld wordt gebruikt. Dit is het regelsetbestand voor de voorbeeld expressieannotator die telefoonnummers, URL's en e-mailadressen herkent.

Op het hoogste niveau bevindt zich het element <ruleSet>, dat bestaat uit een of meer <rule>-elementen. Elk <rule>-element definieert op zijn beurt een Java-expressie die bestaat uit een kenmerk regEx en de kenmerken matchStrategy en matchType. De actie wordt gedefinieerd met het element <createAnnotation> element dat het annotatie-ID en het annotatietype aangeeft.

```
<?xml version="1.0" encoding="UTF-8"?>
<ruleSet xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ruleSet.xsd">
<!-- Telefoonnummer -->
<!-- Deze regel beschrijft verschillende manieren om telefoonnummers
te schrijven, bijvoorbeeld, 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-annotatie -->
<!-- Deze regel beschrijft URL's, bijvoorbeeld 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>
<!-- E-mailannotatie -->
<!-- Deze regel beschrijft e-mailadressen, bijvoorbeeld uwNaam@domein.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>
```

Procedure

U maakt als volgt een XML-regelset voor de expressieannotator die uw aangepaste expressies definieert:

1. Maak een XML-bestand. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma. Het XSD-schema voor het bestand met de XML-regelset heet ruleSet.xsd; u vindt het in uw installatie van enterprise search, in de directory *ES_INSTALL_ROOT*/packages/uima/regex/.
2. Neem uw toewijzingen op in een element `<ruleSet xmlns="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="ruleSet.xsd">`. De naamruimte is opgegeven in het kenmerk xmlns en moet exact worden ingevoerd zoals weergegeven.
3. Voeg een element `<rule>` toe dat een `regEx`-kenmerk met het expressiepatroon bevat, plus een kenmerk `matchStrategy` en een kenmerk `matchType`.

   De annotator biedt volledige ondersteuning aan de syntaxis voor expressies van Java 1.4. Een kennismaking met expressies en een overzicht van de volledige syntaxis vindt u in de Java-documentatie op http://java.sun.com/j2se/1.4.2/docs/api/java/util/regex/Pattern.html.

   `matchStrategy` geeft aan hoe er moet worden gezocht als bijvoorbeeld alle matches in het document moeten worden gevonden of als er sprake moet zijn van een exacte match. Er zijn drie verschillende strategieën beschikbaar:

   - `matchFirst` stopt bij de eerste tekstreeks die aan het patroon voldoet
   - `matchAll` zoekt naar alle tekstreeksen in het document die aan het patroon voldoen
   - `matchComplete` behandelt alleen tekstreeksen die exact overeenkomen als een match. Bijvoorbeeld: als we zoeken naar het patroon “bloem”, wordt alleen het woord ″bloem″ beschouwd als een match, ″bloemkool″ niet.

   `matchType` bepaalt met welk type annotatie de regel wordt vergeleken. Op die manier kunt u de overeenkomsten van uw expressie bijvoorbeeld beperken tot een bestaande tokenannotatie. Zo voorkomt u dat er met een regel teveel content wordt gevonden. De mogelijke typen zijn de typen annotaties die zijn toegestaan als invoer (gedefinieerd in de annotatordescriptor), zoals uima.tt.DocumentAnnotation, uima.tt.ParagraphAnnotation, en door de gebruiker gedefinieerde typen zoals foo.bar.MyAnnotation. Soms wordt het type uitvoer van de ene regel gebruikt als type invoer van een volgende regel. Met `matchType` kunt u het zoekbereik beperken tot bepaalde regels.

4. Voeg een element <createAnnotation> dat bepaalt wat de expressieannotator moet doen als er een match wordt gevonden.

   Elk element createAnnotation heeft twee kenmerken:

   - id geeft een unieke identificatie van de annotatie en wordt gebruikt om naar de annotatie te verwijzen
   - type geeft aan welk type annotatie er wordt gemaakt

5. Voeg de volgende componentelementen toe. Deze bepalen de positie van de match voor het element <createAnnotation>:
   - Verplicht: <begin> geeft aan waar de match begint. Dit element heeft twee kenmerken:
     - Verplicht: group geeft de ″Java capturing group″ aan. Hieraan kan een waarde worden toegewezen tussen 0 (match van volledige tekstreeks) en 9 (meerdere capturing groups)
     - Optioneel: location geeft de positie binnen de matchgroep aan (met betrekking tot de plaatsing van de haakjes), hetzij start (haakje openen) of end (haakje sluiten).
   - Verplicht: <end> geeft aan waar de match eindigt. Dit element heeft twee kenmerken:
     - Verplicht: group geeft de ″capturing group″ aan. Hieraan kan een waarde worden toegewezen tussen 0 (match van volledige tekstreeks) en 9 (volgende en nog kleinere matchgroepen)
     - Optioneel: location geeft de positie binnen de matchgroep aan (met betrekking tot de plaatsing van de haakjes), hetzij start (haakje openen) of end (haakje sluiten).
   - Optioneel: <setFeature> maakt een feature en wijst deze toe aan de annotatie. Dit element heeft twee kenmerken:
     - name is de naam van de feature zoals u die hebt opgegeven in de beschrijving van het typesysteem
     - type geeft het type van de featurewaarde aan: String, Integer, Float of Reference. Het type moet gelijk zijn aan het type bereik dat voor de feature is opgegeven in de beschrijving van het annotatortypesysteem.

       Features van het type Reference worden gebruikt voor het aanbrengen van een link tussen twee annotaties om een semantische relatie tot stand te brengen. De content van het element <setFeature> moet worden ingesteld op het id van het element <createAnnotation> waarheen u een link wilt maken.

Verwante onderwerpen

“Het regelsetbestand” op pagina 86
In de expressieannotator bepaalt het XML-regelsetbestand de regels, in de vorm van expressies, die worden gebruikt voor het analyseren (parsing) van het tekstdocument.

De expressieannotator aanpassen

U kunt de voorbeeldconfiguratie van de expressieannotator zodanig aanpassen dat deze nieuwe entiteiten herkent (bijvoorbeeld serienummers van producten). Het is mogelijk om de expressieregels voor bestaande entiteiten aan te passen (bijvoorbeeld voor het herkennen van bedrijfs-specifieke telefoonnummers). Dit laatste doet u door kleinere wijzigingen aan te brengen in de voorbeeldbestanden voor regelset en typesysteem.

Het gewijzigde regelsetbestand en de gewijzigde beschrijving van het typesysteem moeten worden toegevoegd aan het PEAR-bestand van de expressie. Nadat u het PEAR-bestand hebt bijgewerkt, kunt u het tekstanalyseprogramma met de bijgewerkte expressie weer toevoegen aan het enterprise search-systeem.

Voor ingewikkelde aanpassingen van de expressieannotator kunt u het beste gebruik maken van de UIMA SDK-tools. Deze tools helpen u bij het maken en bijwerken van de typesysteembeschrijving en de descriptorbestanden, bij het mogelijk combineren van de annotator met andere om een samengesteld analyseprogramma te vormen en bij het maken van een nieuw PEAR-bestand dat alle resources bevat die nodig zijn om de annotator te gebruiken in enterprise search. Voor informatie over de tools die beschikbaar zijn ter ondersteuning van deze taken kunt u de documentatie van UIMA SDK raadplegen.

Procedure

Als u nieuwe regels aan de expressieannotator wilt toevoegen of bestaande wilt wijzigen, kunt u het bijgeleverde PEAR-bestand van de voorbeeld expressieannotator als volgt bijwerken:

1. Maak in uw systeem een nieuwe directory met de naamxml.
2. Kopieer het voorbeeld regelbestand of_sample_regex_rules.xml in de directory *ES_INSTALL_ROOT*/packages/uima/regex/ naar uw directory xml en pas het bestand zodanig aan dat het uw aangepaste patroonmatchingregels bevat. Om XML-syntaxisfouten te voorkomen, kunt u het beste gebruikmaken van een XML-editor of XML-authoringprogramma.
3. Kopieer het overeenkomstige beschrijvingsbestand voor het typesysteem, of_sample_typesystem.xml, vanuit de directory ES_INSTALL_ROOT/ packages/uima/regex/ naar uw directory xml en pas het bestand zodanig aan dat het de definities bevat voor de typen die voor uw nieuwe regels vereist zijn.
4. Als u slechts enkele nieuwe regels toevoegt of slechts enkele bestaande regels wijzigt, is het niet nodig de annotatordescriptor te wijzigen. Bent u van plan andere wijzigingen aan te brengen, of voert u aanvullende aangepaste analyseprocedures uit, ga dan na of de annotatordescriptor moet worden aangepast.
5. Gebruik een archiveringsprogramma naar keuze om de twee bijgewerkte bestanden op te nemen in het PEAR-bestand van de expressieannotator. Bijvoorbeeld: kopieer het bestand of_regex.pear vanuit de directory ES_INSTALL_ROOT/packages/uima/regex/ naar de bovenliggende directory van de directory xml die u hebt gemaakt. Geef vervolgens met behulp van het jar-opdrachtregelprogramma van Java (bijvoorbeeld onderdeel van de IBM Java SDK) de volgende opdrachten op vanuit de bovenliggende directory:

   ```
   "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. Voeg de expressieannotator met behulp van de beheerconsole van enterprise search als aangepast tekstanalyseprogramma toe aan het enterprise search-systeem en koppel hem aan een test-documentcollectie.
7. Controleer de analyseresultaten die door de expressieannotator worden verkregen door de kenmerken van de documentcollectie zodanig aan te passen dat er met de functie XCAS-dump leesbare uitvoer wordt geproduceerd van de analyseresultaten die zijn opgeslagen in de Common Analysis Structure.
8. Verwerk de testdocumenten en bekijk de content van de XML-bestanden met behulp van de XCAS Annotation Viewer.

9. Als u tevreden bent met de annotaties die door de annotator worden gemaakt op basis van uw aangepaste expressies, dient u de eigenschappen van de documentcollectie nogmaals aan te passen, maar nu om te zorgen dat de parser GEEN leesbare XML-uitvoer meer produceert van de analyseresultaten. Als er verdere aanpassingen van het regelsetbestand vereist zijn, moet u de procedure voor het bijwerken van het PEAR-bestand nogmaals uitvoeren.
10. Maak een toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index als u de analyseresultaten wilt indexeren, of een toewijzingsbestand voor toewijzing van een Common Analysis Structure aan een database als u de resultaten wilt toevoegen aan een database. U kunt het bijgeleverde voorbeeld-toewijzingsbestand voor de toewijzing van de Common Analysis Structure aan een index gebruiken als uitgangspunt voor het maken van uw eigen toewijzingsbestand.
11. Gebruik de beheerconsole van enterprise search om de toewijzingsbestanden toe te voegen en ze te koppelen aan uw volledige documentcollectie.
12. Doorzoek uw annotaties met behulp van XML-fragment- of XPath-query's. U kunt hiervoor ook semantische uitbreiding gebruiken bij het zoeken naar synoniemen.

Verwante onderwerpen

"Eenvoudige semantische zoekopdracht met expressieannotator" op pagina 84
Enterprise search wordt geleverd met een expressieanalyseprogramma dat vooraf is geconfigureerd met een groep regels die dat programma in staat stelt telefoonnummers, URL's en e-mailadressen in documenten te herkennen.

Verwante taken

"De resultaten van de basisannotator en aangepaste tekstanalyse bekijken" op pagina 13
Om te zien welke analyseresultaten de parser en de annotators in enterprise search hebben geproduceerd, moet u de eigenschappen van de documentcollectie zodanig aanpassen dat er een leesbare XML-versie wordt gemaakt van de analyseresultaten die worden opgeslagen in de Common Analysis Structure.

De annotatordescriptor

De XML-descriptor van de expressieannotator bevat beschrijvende informatie over de expressieannotator. Die informatie is nodig om de annotator te kunnen laten werken.

Als u alleen gebruik maakt van de expressieannotator, en dus niet van aanvullende aangepaste analysemethoden, hoeft u de descriptor alleen te wijzigen als:

- U de bestandsnaam van het regelsetbestand wilt wijzigen (in het element `<externalResourceDependencies>`).
- U meer dan één regelsetbestand wilt gebruiken.
- U de naam van het beschrijvingsbestand voor het typesysteem wilt wijzigen.

Als u wél gebruik maakt van aanvullende aangepaste analysemethoden, moet u de descriptor wijzigen als:

- U wilt dat er bij uw aangepaste analyse gebruik wordt gemaakt van annotaties die zijn gemaakt door de expressieannotator. In dit geval moet u de uitvoermogelijkheden bijwerken in de annotatordescriptor.
- U expressieregels hebt gedefinieerd die overeen moeten komen met typen annotaties die zijn gemaakt in eerdere stappen van aangepaste analyse. In dit geval moet u de invoermogelijkheden bijwerken in de annotatordescriptor.

Voor het maken of bijwerken van de annotatordescriptor en het opnieuw maken van het PEAR-bestand met alle resources die nodig zijn om de annotator te gebruiken in enterprise search, kunt u gebruik maken van de UIMA SDK-tools. Zie de documentatie van UIMA voor informatie over de tools die beschikbaar zijn ter ondersteuning van deze taken.

Configuratieparameters

De nexpressieannotator kent slechts één configuratieparameter, genaamd `String2NumberImpl`, die moet worden ingesteld op de naam van de klasse waarmee de interface `com.ibm.uima.an_regex.String2Number` wordt geïmplementeerd. Aan de expressieannotator moet een implementatie van deze klasse beschikbaar worden gesteld, anders treedt er een uitzondering op. Als u de expressieannotator wilt aanpassen aan uw wensen, kunt u uw eigen implementatie van de interface `String2Number` aanleveren door uw klassenaam door te geven in het XML-descriptorbestand.

De interface `String2Number` declareert twee methoden, `toInt(String)` en `toFloat(String)`, die een tekenreeksweergave (string) van een geheel getal of een getal met drijvende komma (float) omzetten in het dienovereenkomstige gehele getal of een getal met drijvende komma. Deze twee methoden worden gebruikt om een getal dat scheidingstekens bevat, om te zetten in een geldig Java-getal (Integer of Float).

In de standaardimplementatie van `com.ibm.uima.an_regex.String2Number_impl` wordt een punt (.) beschouwd als decimaal scheidingsteken en een komma (,) als scheidingsteken voor duizendtallen. Als bijvoorbeeld 1,999.00 wordt aangetroffen in een tekstdocument, zet `toInt` dit getal om in 1999. `toFloat` zet het om in 1999.00.

Voorbeeld

De sectie met configuratieparameters van de descriptor ziet er als volgt uit:

```
<configurationParameters>
 <configurationParameter>
<name>String2NumberImpl</name>
<description>Implementatie van de interface
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>
```

Mogelijkheden

De invoer- en uitvoermogelijkheden van de expressieannotator en de talen die door de expressieannotator worden ondersteund, worden gedefinieerd in de mogelijkhedensectie (capabilities) van de annotatordescriptor.

De invoermogelijkheden (invoertypen) en het descriptorbestand moeten voldoen aan de typen matches die worden gebruikt in het regelsetbestand. Als er in de regels alleen gebruik wordt gemaakt van het type `uima.tt.DocumentAnnotation`, hoeft u geen invoermogelijkheden te declareren, want dit type is altijd gedefinieerd. Alle andere typen moeten wél worden gedeclareerd.

De typen annotaties die door de expressieannotator worden gemaakt, worden opgegeven in de sectie voor uitvoermogelijkheden. Deze typen moeten overeenkomen met de uitvoertypen die zijn gedeclareerd in het regelsetbestand.

Omdat de expressieannotator taal-onafhankelijk is, kunt u `x-unspecified` opgeven. Dit staat voor elke willekeurige taal.

Typesysteembeschrijving

In de sectie met de beschrijving van het typesysteem in de XML-descriptor van de expressieannotator wordt het door de annotator gebruikte typesysteem gedefinieerd. De typen die worden gebruikt in het XML-regelsetbestand en die worden genoemd in de secties met invoer- en uitvoermogelijkheden van de annotatordescriptor, moeten overeenkomen met de typen die zijn gedefinieerd in de beschrijving van het typesysteem.

Voorbeeld

De sectie met de beschrijving van het typesysteem in de descriptor importeert het XML-bestand met de typesysteemdescriptor:

```
<typeSystemDescription>
 <imports>
  <import location="./xml/of_sample_regex_typesystem.xml"/>
 </imports>
</typeSystemDescription>
```

Externe resources

De sectie met externe resources in de descriptor bevat de bestanden en klassen die vereist zijn voor de annotator.

De expressieannotator heeft een regelsetbestand nodig. Dat regelsetbestand wordt aan de expressieannotator beschikbaar gesteld via de interface `com.ibm.uima.an_regex.FileResource`, welke wordt geïmplementeerd door de klasse `com.ibm.uima.an_regex.impl.FileResource_impl`. Om uw aangepaste regels door te geven aan de expressieannotator, moet u de naam van het regelsetbestand opgeven in de annotatordescriptor en de locatie van het bestand toevoegen aan uw klassenpad. De sleutel die door de expressieannotator wordt gebruikt om toegang te krijgen tot het regelsetbestand heet `RuleSetDefinition`. Verander deze sleutel niet, want dan kan de expressieannotator de regelset niet meer vinden en kan de annotator niet meer worden geïnitialiseerd.

Voorbeeld

De sectie met externe resources van de descriptor ziet er als volgt uit:

```
<externalResourceDependencies>
 <externalResourceDependency>
<key>RuleSetDefinition</key>
<description>Regelsetdefinitie</description>
<interfaceName>com.ibm.uima.an_regex.FileResource</interfaceName>
<optional>false</optional>
```

```
 </externalResourceDependency>
</externalResourceDependencies>
<resourceManagerConfiguration>
 <externalResources>
  <externalResource>
<name>of_samples_regex_rules</name>
<description>Regelsetdefinitiebestand voor kamernummers</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>
```

Verwante onderwerpen

"Expressieannotator" op pagina 83
De expressieannotator maakt het mogelijk om aangepaste tekstanalyse uit te voeren zonder dat het nodig is uw eigen tekstanalyseprogramma te implementeren. Op basis van een set regels (expressies of ″regular expressions″) die u zelf kunt opstellen, is de expressieannotator in staat informatiestructuren in tekstdocumenten en annotaties van de gedetecteerde informatie te maken in de Common Analysis Structure.

"Het regelsetbestand" op pagina 86
In de expressieannotator bepaalt het XML-regelsetbestand de regels, in de vorm van expressies, die worden gebruikt voor het analyseren (parsing) van het tekstdocument.

Verwante verwijzing

"Loggen"
Alle logberichten van de expressieannotator worden bijgeschreven in het logbestand van de huidige collectie.

Loggen

Alle logberichten van de expressieannotator worden bijgeschreven in het logbestand van de huidige collectie.

De logbestanden van de collectie bevinden zich in ES_NODE_ROOT/logs/ en hebben namen van het type <*collectie-ID*>_<*actuele_datum*>.log. De logbestanden kunnen worden bekeken met de scripts esviewlogs.sh/.bat.

Er zijn zeven mogelijke logniveaus:

- Fout
- Waarschuwing
- Info
- Config
- Fine
- Finer
- Finest

De toewijzing van Fout- en Waarschuwingsberichten kan niet worden gewijzigd. Standaard worden alleen berichten van het type Info, Waarschuwing en Fout naar het logbestand geschreven. Dit zijn de standaard logniveaus die door enterprise search worden gebruikt. De andere logniveaus kunnen worden toegewezen omwille van meer gedetailleerde informatie.

Om logberichten van de expressieannotator te ontvangen, moet het logniveau minimaal zijn ingesteld op Config. Op dit niveau logt de annotator configuratie-instellingen, zoals het gebruikte regelsetbestand en de naam van de implementatieklasse voor de `com.ibm.uima.an_regex.String2Number`-interface.

Als u het logniveau bijvoorbeeld instelt op Finer, logt de annotator welke annotaties er niet konden worden gemaakt. Dit kan u helpen vast te stellen waarom niet alle annotaties die u verwachtte, daadwerkelijk zijn gemaakt. Er zou bijvoorbeeld een fout kunnen staan in een van uw expressies, of een optionele capturing group kwam wellicht niet overeen met enig stuk tekst in het document. Als u opgeeft dat een feature moet worden ingesteld op de tekstreeks die overeenkomt met een capturing group en er is geen overeenkomende tekstreeks, dan wordt de feature wordt ingesteld op null.

Voor de meest gedetailleerde informatie stelt u het logniveau in op Finest. Op dit niveau logt de annotator het actuele expressiepatroon, het deel van de documenttekst dat op dat moment wordt geanalyseerd en alle annotaties en features die zijn gemaakt. Het gebruik van zeer gedetailleerd loggen, met name de logniveaus Finer en Finest heeft duidelijk negatieve gevolgen voor de algehele snelheid van de annotator.

Mocht u gedetailleerde toewijzing van het logniveau verlangen, pas dan het configuratiebestand tokenizer.properties in *ES_NODE_ROOT*/master_config/parserservice/ aan en wijzig bijvoorbeeld de configuratie-instelling `trevi.tokenizer.jedii.InformationalLevelMapping=Info` in `trevi.tokenizer.jedii.InformationalLevelMapping=Finest`.

Om de wijzigingen in het logniveau te activeren, moet u met behulp van de beheerconsole alle parserprocessen stoppen. Vervolgens moet u de sessie van de parserservice stoppen en opnieuw starten. Dit doet u vanaf de opdrachtregel, met de volgende opdrachten:

```
>esadmin session parserservice stop
>esdamin session parserservice start
```

Daarna kan het analyseren opnieuw worden gestart en is het nieuwe logniveau actief. Elke keer dat u het logniveau wijzigt, moet u deze procedure uitvoeren.

Verwante onderwerpen

"Expressieannotator" op pagina 83
De expressieannotator maakt het mogelijk om aangepaste tekstanalyse uit te voeren zonder dat het nodig is uw eigen tekstanalyseprogramma te implementeren. Op basis van een set regels (expressies of ″regular expressions″) die u zelf kunt opstellen, is de expressieannotator in staat informatiestructuren in tekstdocumenten en annotaties van de gedetecteerde informatie te maken in de Common Analysis Structure.

"Het regelsetbestand" op pagina 86
In de expressieannotator bepaalt het XML-regelsetbestand de regels, in de vorm van expressies, die worden gebruikt voor het analyseren (parsing) van het tekstdocument.

Verwante verwijzing

“De annotatordescriptor” op pagina 92
De XML-descriptor van de expressieannotator bevat beschrijvende informatie over de expressieannotator. Die informatie is nodig om de annotator te kunnen laten werken.

Documentatie bij enterprise search

U kunt de documentatie bij OmniFind Enterprise Edition als PDF- of HTML-document lezen.

Het installatieprogramma van OmniFind Enterprise Edition installeert het Informatiecentrum van IBM Content Discovery automatisch. Dit omvat tevens HTML-versies van de documentatie van OmniFind Enterprise Edition, Versie 8.4 en WebSphere Information Integrator Content Edition Versie 8.3. Als u met meerdere servers werkt, wordt het Informatiecentrum op alle zoekservers geïnstalleerd. Als u het Informatiecentrum niet hebt geïnstalleerd en op Help klikt, wordt het Informatiecentrum geopend op de website van IBM.

Als u de geïnstalleerde versie van de PDF-documenten wilt bekijken, gaat u naar ES_INSTALL_ROOT/docs/*locale*/pdf. Als u de documenten bijvoorbeeld in het Engels wilt bekijken, gaat u naar ES_INSTALL_ROOT/docs/en_US/pdf.

Om de PDF-versie van de documentatie in alle beschikbare talen te zien, gaat u naar de website met de documentatie van OmniFind Enterprise Edition Versie 8.4.

Op de website OmniFind Enterprise Edition Support hebt u ook toegang tot downloads, fixpacks, technotes en het Informatiecentrum.

In de volgende tabel ziet u de beschikbare documentatie, bestandsnamen en locaties.

Tabel 12. Documentatie voor enterprise search

Title	Bestandsnaam	Locatie
Informatiecentrum		http://publib.boulder.ibm.com/infocenter/discover/v8r4/
Installation Guide for Enterprise Search	iiysi.pdf	ES_INSTALL_ROOT/docs/*locale*/pdf/
Quick Start Guide (Dit document is tevens beschikbaar als gedrukt boekje in het Engels, Frans en Japans.)	QuickStartGuide_*twee letters van taal*.pdf	ES_INSTALL_ROOT/docs/*locale*/pdf/
Installation Requirements for Enterprise Search	iiysr.txt of iiysr.htm	ES_INSTALL_ROOT/docs/*locale*/ (u kunt ook naar dit bestand gaan via het startvenster (launchpad) voor de installatie)
Enterprise Search beheren	iiysa.pdf	ES_INSTALL_ROOT/docs/*locale*/pdf/
Programming Guide and API Reference for Enterprise Search	iiysp.pdf	ES_INSTALL_ROOT/docs/en_US/pdf/
Troubleshooting Guide and Messages Reference	iiysm.pdf	ES_INSTALL_ROOT/docs/*locale*/pdf/
Integratie van tekstanalyse	iiyst.pdf	ES_INSTALL_ROOT/docs/*locale*/pdf/
Plugin voor Google Desktop Search	iiysg.pdf	ES_INSTALL_ROOT/docs/*locale*/pdf/

Tabel 12. Documentatie voor enterprise search (vervolg)

Title	Bestandsnaam	Locatie
Release Notes	iiysn.pdf	Beschikbaar op de website met documentatie voor OmniFind Enterprise Edition, Versie 8.4 (u kunt ook naar dit bestand gaan via het startvenster (launchpad) voor de installatie)

Toegankelijkheidsfuncties in WebSphere Information Integrator OmniFind Edition

De gebruikersinterfaces en documentatie voor IBM WebSphere Information Integrator OmniFind Edition zijn toegankelijk.

Installatieprogramma

U kunt gebruikmaken van sneltoetsen om door het installatieprogramma van WebSphere Information Integrator OmniFind Edition te navigeren. In de volgende tabel worden enkele sneltoetsen beschreven.

Tabel 13. Sneltoetsen voor het installatieprogramma

Actie	**Sneltoets**
Een keuzerondje markeren	Cursortoets
Een keuzerondje selecteren	Tabtoets
Een opdrachtknop markeren	Tabtoets
Een opdrachtknop selecteren	Enter-toets
Naar het volgende of vorige venster gaan of annuleren	Een opdrachtknop markeren met de tabtoets en vervolgens op Enter drukken
Het actieve venster deactiveren	Ctrl + Alt + Esc

Beheerconsole en Informatiecentrum van enterprise search

De beheerconsole en het Informatiecentrum bevatten een op browsers gebaseerde interface die u kunt bekijken in Microsoft Internet Explorer of Mozilla FireFox. Raadpleeg de online Help bij Internet Explorer of FireFox voor een overzicht van sneltoetsen en andere toegankelijkheidsopties voor uw browser.

documentatie in PDF-indeling

U kunt alle documentatie bij enterprise search bekijken in PDF-indeling. De PDF-documenten zijn toegankelijk via Adobe Acrobat versie 6.0. De PDF-documenten zijn gestructureerd en kunnen door de meeste schermlezers worden gelezen.

Woordenlijst met termen voor enterprise search

In deze woordenlijst vindt u de termen die in de interfaces en documentatie van enterprise search worden gebruikt.

Portal Document Manager (PDM)
Biedt gebruikers de mogelijkheid om met één centrale documentenrepository te werken, ten behoeve van samenwerking binnen een team. Beheerders hebben de mogelijkheid hun documenten effectief te beheren en kunnen invloed uitoefenen op de manier waarop gebruikers interactief kunnen werken met informatie.

aangepast tekstanalyseprogramma
Een tekstanalyseprogramma dat is gemaakt met behulp van de Unstructured Information Management Architecture (UIMA) software development kit (SDK) en dat kan worden toegevoegd aan de set van standaard tekstanalyseprogramma's van enterprise search (ook bekend als de basisannotators van enterprise search). Zie ook tekstanalyseprogramma.

Afsluitend teken
Een teken dat de laatste positie in een woord aangeeft.

Analyseprogramma
Zie Tekstanalyseprogramma.

Analyseresultaten
De gegevens die door de annotators worden geproduceerd. Analyseresultaten worden geschreven in een gegevensstructuur die Common Analysis Structure wordt genoemd. De resultaten van analyse die is uitgevoerd door tekstanalyseprogramma's op maat (annotators) kunnen beschikbaar worden gesteld voor zoeken door ze op te nemen in de enterprise search-index.

Annotator
Een softwarecomponent waarmee specifieke taalkundige analysetaken worden uitgevoerd en annotaties worden geproduceerd en vastgelegd. Een annotator is de analyselogicacomponent in een analyseprogramma.

Begripsextractie
Een tekstanalysefunctie waarmee significante vocabulaire-items (zoals personen, plaatsen of producten) in tekstdocumenten worden geïdentificeerd, waarna een lijst met deze items als resultaat wordt gegeven. Zie ook Thema-extractie.

beheerdersrol
Een classificatie van een gebruiker waarmee wordt bepaald welke functies de gebruiker kan uitvoeren in de beheerconsole van enterprise search. Daarnaast wordt met de rol bepaald welke collecties de gebruiker kan beheren.

Beveiligingstoken
Informatie over de identiteit en beveiliging die wordt gebruikt om toegang te verlenen tot de documenten in een collectie. Verschillende typen gegevensbronnen bieden ondersteuning voor verschillende typen beveiligingstokens. Voorbeelden zijn gebruikersrollen, gebruiker-ID's en andere informatie die kan worden gebruikt om de toegang tot bepaalde inhoud te bepalen.

bewaken
Een enterprise search-gebruiker die de machtiging heeft voor het bewaken van processen op collectieniveau.

Bibliotheek
Een systeemobject dat dienst doet als directory voor andere objecten. Zie ook Domino Document Manager-bibliotheek.

Booleaanse zoekopdracht
Een zoekopdracht waarin een of meer zoektermen worden gecombineerd door gebruik te maken van operatoren zoals EN, NIET en OF.

bouwen van delta-index
Proces waarbij nieuwe gegevens aan een bestaande index in een enterprise search-systeem worden toegevoegd. Tegenstelling met bouwen van hoofd-index.

bouwen van hoofdindex
Proces waarbij de hele index in een enterprise search-systeem wordt samengesteld. Tegenstelling met bouwen van delta-dindex.

CAS-consumer
Een CAS-consumer (CAS = Common Analysis Structure) voert de laatste verwerkingsstappen uit op de analyseresultaten die zijn opgeslagen in de Common Analysis Structure. Een consumer kan bijvoorbeeld de inhoud van de Common Analysis Structure in een zoekmachine indexeren of een relationele database vullen met specifieke analyseresultaten.

Categorie
Een groep documenten met vergelijkbare eigenschappen.

Categoriestructuur
Een hiërarchie met categorieën die in de beheerconsole van enterprise search wordt weergegeven.

Certificaat
Een digitaal document waarmee een openbare sleutel wordt verbonden aan de identiteit van de eigenaar van het certificaat, zodat de eigenaar kan worden geverifieerd. Een certificaat wordt uitgegeven door een certificaat-gever.

Certificaatgever
Een organisatie die certificaten uitgeeft en de entiteiten (individuen of organisaties) verifieert die betrokken zijn bij elektronische transacties. Certificaatgevers garanderen dat de twee partijen die gegevens uitwisselen, daadwerkelijk zijn wie ze claimen te zijn.

Cliticum
Een woord dat syntactisch gezien zelfstandig functioneert, maar fonetisch gezien verbonden is met een ander woord. Een cliticum kan verbonden zijn met het woord of zelfstandig worden geschreven van het woord waar-aan het is verbonden. Veelvoorkomende voorbeelden van een cliticum zijn het laatste deel van een samentrekking in het Engels (*would**n't*** of *you**'re***).

Collectie
Een set gegevensbronnen en opties voor het crawlen, analyseren, indexeren en doorzoeken van deze gegevensbronnen.

Common Analysis Structure (CAS)
Een structuur waarin de content en metagegevens van een document wor-den opgeslagen, alsmede alle analyseresultaten die worden geproduceerd

door een tekstanalyseprogramma. Alle gegevensuitwisseling tijdens de analyse van documenten vindt plaats met behulp van de Common Analysis Structure.

common communication layer (CCL)
De communicatie-infrastructuur waarin de verschillende componenten (controller, parser, crawler, indexserver) van WebSphere Information Integrator OmniFind Edition worden samengevoegd.

Crawler
Een softwareprogramma waarmee documenten uit gegevensbronnen worden opgehaald en gegevens worden verzameld die kunnen worden gebruikt voor het maken van zoekindexen.

Crawlruimte
Een set met bronnen waarvan de opgegeven patronen overeenkomen (zoals URL's, databasenamen, bestandssysteempaden, domeinnamen en IP-adressen) en waaruit de crawler gegevens leest om items op te halen voor indexeringsdoeleinden.

datastore
Een gegevensstructuur waarin de documenten in geanalyseerde vorm worden bewaard. De parser schrijft gegevens naar de datastore. De datastore wordt gebruikt voor het bouwen van de index en voor het genereren van samenvattingen. Niet te verwarren met de raw data store.

Diakritische tekens
Een teken dat aan een letter wordt toegevoegd om de uitspraak van een woord te wijzigen of om onderscheid te maken tussen vergelijkbare woorden, zoals een accentteken of de umlaut.

DN-naam
De unieke naam waarmee een vermelding in een woordenboek wordt aangegeven. Een DN-naam bestaat uit kenmerk:waardeparen, gescheiden door komma's. Daarnaast is de DN-naam een unieke set met naam/waardeparen (zoals CN=naam van persoon en C=land of regio) waarmee een entiteit in een digitaal certificaat wordt aangegeven.

Documentobjectmodel
Een systeem waarin een gestructureerd document, zoals een XML-bestand, wordt weergegeven als een structuur met objecten die via een programma kan worden geopend en bijgewerkt.

Domino Document Manager-bibliotheek
Een Domino Document Manager-database die het ingangspunt is voor Domino Document Manager.

Domino Document Manager-kabinet
Een Domino Document Manager-database die wordt gebruikt om documenten te organiseren. De Domino-databases worden opgeslagen in kabinetten.

Domino Internet Inter-ORB Protocol (DIIOP)
Een servertaak die op de server wordt uitgevoerd en samenwerkt met de Domino-objectaanvraagbroker om communicatie toe te staan tussen Java-applets die zijn gemaakt met de Notes Java-klassen en de Domino-server. Gebruikers van browsers en Domino-servers maken gebruik van DIIOP voor de communicatie en voor het uitwisselen van objectgegevens.

dynamische ranking
Een type ranking waarin de termen in de query worden geanalyseerd op

basis van de documenten die worden doorzocht om de rang van de resultaten te bepalen. Zie ook Op tekst gebaseerde scores. Vergelijk met Statische ranking.

Dynamisch samenvatten
Een manier van samenvatten waarbij de zoektermen worden gemarkeerd en de zoekresultaten zinnen bevatten die het meest overeenkomen met de begrippen in het document die de gebruiker zoekt. Vergelijk met Statisch samenvatten.

enterprise search basisannotators
Een set standaard tekstanalyseprogramma's die in enterprise search wordt gebruikt voor standaardanalyse van documenten.

enterprise search-beheerder
Een beheerdersrol waarmee een gebruiker het volledige enterprise search-systeem kan beheren.

Escapeteken
Een teken waarmee een speciale betekenis voor een of meer volgende tekens wordt onderdrukt of geselecteerd.

expressieannotator
D expressieannotator detecteert entiteiten of eenheden informatie in een tekstdocument, bijvoorbeeld telefoonnummers, productnummers, namen van werknemers, of adressen, op basis van een expressie die het exacte patroon beschrijft dat in de tekst van documenten moet worden gezocht. Als een van de expressies een match oplevert in de tekst van een document, brengt de expressieannotator de dienovereenkomstige annotaties die de match (of een deel daarvan) beslaan. Deze geannoteerde expressies worden dan opgeslagen, hetzij in de enterprise search-index (met behulp van een indextoewijzingsbestand), hetzij in een JDBC-database (met behulp van een databasetoewijzingsbestand).

Externe gegevensbron
Een federatieve gegevensbron die niet wordt gecrawld, geanalyseerd of geïndexeerd door WebSphere Information Integrator OmniFind Edition. Zoekopdrachten voor externe gegevensbronnen worden overgedragen aan de query-API van deze gegevensbronnen.

Featurepad
Een pad dat wordt gebruikt om toegang te krijgen tot de waarde van een feature in een UIMA-featurestructuur (UIMA = Unstructured Information Management Architecture).

Featurestructuur
De onderliggende gegevensstructuur waarin de resultaten van een tekstanalyse worden aangegeven. Een featurestructuur is een structuur met kenmerken en de bijbehorende waarden. Elke featurestructuur heeft een bepaald type en elk type heeft een opgegeven set met geldige features of kenmerken, vergelijkbaar met een Java-klasse.

Federatie
Een proces waarin de naamgevingssystemen worden gecombineerd, zodat in het samengevoegde systeem samengestelde namen kunnen worden verwerkt die in de verschillende naamgevingssystemen worden gebruikt.

Federatieve zoekopdracht
Een zoekoptie waarmee kan worden gezocht in verschillende zoekservices en waarmee een geconsolideerde lijst met zoekresultaten als resultaat wordt gegeven.

Gebruikersagent
Een programma waarmee op internet wordt gezocht en informatie over de agent zelf wordt achtergelaten op de bezochte websites. In enterprise search is de webcrawler een gebruikersagent.

Geen index (instructie)
Een instructie op een webpagina waarmee robots (zoals de webcrawler) worden geïnstrueerd de inhoud van deze pagina's niet in de index op te nemen.

Gegevensbron
Een opbergplaats voor gegevens waaruit documenten kunnen worden opgehaald, zoals het web, relationele en niet-relationele databases en contentbeheersystemen.

Gegevensbrontype
Een groepering gegevensbronnen volgens het protocol dat wordt gebruikt om toegang tot de gegevens te krijgen.

Gegevensextractie
Een type begripsextractie waarmee significante vocabulaire-items automatisch worden herkend in tekstdocumenten, zoals namen, termen en uitdrukkingen.

Gewogen woord
Een woord dat van invloed kan zijn op de relatieve ranking van een document in de zoekresultaten. Tijdens de queryverwerking kan het belang van een document waarin een gewogen woord voorkomt, worden vergroot of verkleind afhankelijk van de score die vooraf is gedefinieerd voor het woord.

Het vinden van de stam
Zie Het vinden van de woordstam.

Het vinden van de woordstam
Een proces van taalkundige normalisatie waarin de verschillende vormen van een woord worden gereduceerd tot een algemene vorm. Woorden als *verbindingen*, *verbindend* en *verbonden* worden bijvoorbeeld gereduceerd tot *verbind*.

Hybride zoekopdracht
Een combinatie van een booleaanse zoekopdracht en een vrije zoekopdracht.

Identiteitenbeheer
De mogelijkheid om de actuele legitimatiegegevens van een gebruiker te controleren met behulp van native toegangsbesturing. Als een gegevensbron wordt beschermd middels een product dat enkelvoudige aanmelding (single sign-on, SSO) ondersteunt en de crawler geconfigureerd is voor het gebruiken van SSO-beveiliging, worden er SSO-mechanismen gebruikt om de identiteit van de gebruiker te controleren. Is dat niet het geval, dan worden de legitimatiegegevens in versleutelde vorm vastgelegd in een veilige kluis die kan worden bijgewerkt wanneer de native toegangsbesturing verandert.

Index Zie Volledige tekstindex.

indexeerwachtrij
Een lijst van opdrachten voor het bouwen van hoofd- en delta-indexen.

In wachtrij plaatsen
Items in een wachtrij plaatsen.

IP-adres
Het unieke 32-bits adres waarmee een host in het netwerk wordt aangegeven.

Java Database Connectivity (JDBC)
Een industriestandaard voor databaseonafhankelijke connectiviteit tussen het Java-platform en een groot aantal databases. De JDBC-interface biedt een API op oproepniveau voor op SQL gebaseerde databasetoegang.

JavaScript
Een webscripttaal die in browsers en webservers wordt gebruikt.

JavaServer Pages (JSP)
Een serverscriptingtechnologie waarmee Java-code dynamisch in webpagina's (HTML-bestanden) wordt ingesloten en wordt uitgevoerd wanneer de pagina wordt verzonden, waarna dynamische inhoud aan een client als resultaat kan worden gegeven.

Java virtual machine (JVM)
Een software-implementatie van een processor waarmee gecompileerde Java-code wordt uitgevoerd (applets en toepassingen).

Jokerteken
Een teken dat wordt gebruikt om optionele tekens aan te geven aan het begin, in het midden of aan het einde van een zoekterm.

Katakana
Een tekenset die bestaat uit symbolen die worden gebruikt in een van de twee veelvoorkomende Japanse fonetische alfabetten en primair wordt gebruikt om buitenlandse woorden fonetisch te schrijven.

Koppelingsanalyse
Een methode die is gebaseerd op de analyse van hyperlinks tussen documenten en wordt gebruikt om te bepalen welke pagina's in de collectie belangrijk zijn voor gebruikers.

Legitimatiegegevens
Gedetailleerde informatie die tijdens de verificatie wordt verkregen en waarmee de gebruiker, groepskoppelingen en andere beveiligingsgerelateerde identiteitskenmerken worden beschreven. Met legitimatiegegevens kunnen verschillende services worden uitgevoerd, zoals autorisaties, controles en overdrachtstaken.

lemma
De basisvorm van een woord. Lemma's zijn van groot belang in talen waarin veel woordvervoegingen voorkomen, zoals het Tsjechisch.

Lemmatisering
Proces waarbij het lemma voor een bepaald woord in een woordenboek wordt gezocht. Het verschil tussen lemmatisering en het vinden van de stam is dat het vinden van de stam op algoritmische basis plaatsvindt en in het algemeen geen gebruikmaakt van een woordenboek met de woorden van een taal.

Lexicale verwantschap
De relatie tussen zoekwoorden in een document die qua betekenis dicht bij elkaar staan. Lexicale verwantschap wordt gebruikt om de relevantie van een resultaat te berekenen.

Ligatuur
Twee of meer tekens die verbonden zijn zodat ze als één teken worden weergegeven (bijvoorbeeld: de a en e vormen de ligatuur æ).

Lightweight Directory Access Protocol (LDAP)
Een open protocol waarvoor TCP/IP wordt gebruikt om toegang te verlenen tot directory's die ondersteuning bieden voor een X.500-model en waarvoor de bronvereisten van de complexere X.500-DAP (Directory Access Protocol) niet nodig zijn.

Lokale federator
Een clientfederator voor een set doorzoekbare objecten.

Lotus QuickPlace-place
Een webvenue die wordt geleverd door Lotus QuickPlace waarmee deelnemers die op verschillende plekken werken, kunnen samenwerken aan projecten en online kunnen communiceren in een gestructureerd en beveiligd werkgebied.

Lotus QuickPlace-room
Een gepartitioneerd gebied van een Lotus QuickPlace-place dat beperkt toegankelijk is voor bevoegde leden die een gemeenschappelijke interesse delen en moeten samenwerken.

Maskeringsteken
Een teken dat wordt gebruikt om optionele tekens aan te geven aan het begin, in het midden en aan het einde van een zoekterm. Maskeringstekens worden meestal gebruikt om variaties te vinden voor een term in een index. Zie ook Jokerteken.

MIME-type
Een internetstandaard voor de identificatie van het type object dat via internet wordt overgedragen.

Nabijheidszoekopdracht
Een type zoekopdracht waarin wordt gezocht naar bepaalde woorden in dezelfde zin, dezelfde alinea of hetzelfde document.

Natuurlijke-taalquery
Een type zoekopdracht waarmee geschreven expressies (zoals ″Wie is verantwoordelijk voor de financiële transacties?″) worden geanalyseerd, in plaats van een eenvoudige collectie trefwoorden.

n-gramsegmentering
Een analysemethode waarin overlappende reeksen van een bepaald aantal tekens als één woord worden beschouwd in plaats van witruimten te gebruiken om woorden te scheiden (zoals in op Unicode gebaseerde witruimtesegmentering).

Niet-lokale federator
Een serverfederator voor een set doorzoekbare objecten.

Niet volgen (instructie)
Een instructie op een webpagina waarmee robots (zoals de webcrawler) worden geïnstrueerd om de koppelingen op deze pagina's niet te volgen.

Notes Remote Procedure Call (NRPC)
De communicatiemechanisme van Lotus Notes dat wordt gebruikt voor alle Notes-naar-Notes-communicatie.

Ontdekker
Een functie van een crawler waarmee wordt bepaald uit welke gegevensbronnen de crawler gegevens kan ophalen.

operator
Een enterprise search-gebruiker die is gemachtigd voor het bewaken, starten en beëindigen van processen op collectieniveau.

Op regels gebaseerde categorie
Categorieën die zijn gemaakt op basis van regels waarmee wordt aangegeven welke documenten aan welke categorieën zijn gekoppeld. U kunt bijvoorbeeld regels definiëren om documenten te koppelen die bepaalde woorden bevatten of die voldoen aan een URI-patroon, met specifieke categorieën.

op tekst gebaseerde score
Proces waarbij een geheel getal aan een document wordt toegewezen en waarmee de relevantie van het document wordt aangegeven ten opzichte van de termen in een query. Een hoger getal geeft een grotere overeenkomst met de query aan. Zie ook Dynamische ranking.

Op Unicode gebaseerde witruimtesegmentering
Een tokenisatiemethode waarvoor Unicode-tekeneigenschappen worden gebruikt om een onderscheid te maken tussen tokens en scheidingstekens.

parametrische zoekopdracht
Een type zoekopdracht waarmee wordt gezocht naar objecten die een numerieke waarde of kenmerk bevatten, zoals een datum of geheel getal, of andere numerieke gegevenssoorten binnen een opgegeven reeks.

Parser Een programma waarin documenten worden geïnterpreteerd die aan de enterprise search-opslagplek zijn toegevoegd. Met de parser worden gegevens uit de documenten geëxtraheerd, waarna de documenten worden voorbereid voor indexerings-, zoek- en ophaaltaken.

parserdriver
Een enterprise search-service die de parserservice voorziet van documenten. Voor elke collectie is er één parserdriver. De parserdriverservice van een collectie correspondeert met de parser van de collectie in de beheerconsole van enterprise search.

parserservice
De enterprise search-service die alle documentanalyse (parsing) en tekstanalyse van documentcollecties afhandelt. Er is te allen tijde minimaal één parserservice actief.

PEAR-archief (Processing Engine ARchive)
Een .pear-archiefbestand (zip) dat een UIMA-analyseprogramma en alle bronnen bevat die nodig zijn voor aangepaste analyses in enterprise search.

Populariteitsranking
Een type ranking dat wordt toegevoegd aan de bestaande ranking van een document, op basis van de populariteit van het document.

Proxyserver
Een server die dienst doet als tussenliggend apparaat voor HTTP-webaanvragen van een programma of een webserver. Een proxyserver werkt als vervanger voor de inhoudsservers in het bedrijf.

Ranking
Proces waarin aan elk document dat als resultaat van een query wordt gegeven, een geheel getal wordt toegewezen. De volgorde van de documenten in de zoekresultaten is gebaseerd op de relevantie van de query. Een groter getal geeft een nauwkeurigere overeenkomst aan. Zie ook Dynamische ranking en Statische ranking.

raw data store
Een gegevensstructuur waarin gecrawlde documenten worden opgeslagen voordat ze naar de parser worden gezonden. Crawlers schrijven gegevens naar de raw data store en de parser leest de gegevens uit de raw data store. Nadat documenten door de parser zijn geanalyseerd, worden ze uit de raw data store verwijderd. Niet te verwarren met datastore.

Robots Exclusion Protocol
Een protocol waarmee websitebeheerders kunnen aangeven welke gedeelten van de site niet toegankelijk zijn voor de robot.

Room Een programma waarmee gebruikers documenten kunnen maken voor andere gebruikers, waarmee gebruikers kunnen reageren op de opmerkingen van andere gebruikers en de projectstatus en deadlines kunnen bekijken. Gebruikers kunnen ook met andere gebruikers in dezelfde room chatten. Zie ook Lotus QuickPlace-room.

Ruimte
Een virtuele locatie in de portal waar individuen en groepen elkaar ontmoeten zodat ze kunnen samenwerken. In een portal heeft elke gebruiker een persoonlijke ruimte voor eigen werk en hebben de individuen en groepen toegang tot verschillende gemeenschappelijke ruimten (openbare of beperkt toegankelijke ruimten). Zie ook Lotus QuickPlace-place.

Samenvatten
Proces waarbij zinnen in zoekresultaten worden opgenomen, zodat de inhoud van een document kort wordt beschreven. Zie ook Dynamisch samenvatten en Statisch samenvatten.

Scope Een groep gerelateerde URI's die worden gebruikt om de reikwijdte van een zoekopdracht te definiëren.

Secure Sockets Layer (SSL)
Een beveiligingsprotocol waarmee privacy tijdens communicatie wordt geleverd.

seedlistpagina
In WebSphere Portal: een XML-pagina die links bevat naar pagina's die beschikbaar zijn in de portal. Crawlers gebruiken de seedlist om erachter te komen welke documenten kunnen worden gecrawld. De seedlist-pagina bevat ook metagegevens die zijn opgeslagen bij de gecrawlde documenten in de enterprise search-index.

Segmentering
De verdeling van tekst in afzonderlijke lexicale eenheden. Niet op woordenboeken gebaseerde verwerking omvat witruimte- en n-gram-segmentering, en op woordenboeken gebaseerde segmentering omvat woord-, zin- en alineasegmentering, en lemmatisering.

semantische zoekopdracht
Semantische zoekopdrachten vormen een uitbreiding van zoekopdrachten met sleutelwoorden, in die zin dat ze meer kennis over taalkunde en de zoekoplossing omvatten. De technologie die deze kennis omvat en toepast, staat bekend als tekstanalyse.

Servlet
Een Java-programma dat op een webserver wordt uitgevoerd en de functionaliteit van de server uitbreidt door dynamische inhoud te genereren als antwoord op opdrachten van webclients. Servlets worden veel gebruikt voor databaseverbindingen met het web.

shingle
Een reeks opeenvolgende tokens (woorden) die uit een zin worden gehaald. Bijvoorbeeld: in ″Dit is een heel kort zinnetje″, komen de volgende 3-woords shingles (of trigrammen) voor:

Dit is een
is een heel
een heel kort
heel kort zinnetje

Shingles kunnen worden gebruikt voor statistische taalkunde. Als in twee verschillende teksten bijvoorbeeld zeer veel dezelfde shingles voorkomen, bestaat er waarschijnlijk een bepaald verband tussen die twee teksten.

Sleutelruimtebestand
Een databasebestand dat de openbare sleutels bevat die zijn opgeslagen als certificaten van de ondertekenaar en persoonlijke sleutels die zijn opgeslagen in persoonlijke certificaten.

snellink
Een koppeling tussen een URI en trefwoorden of termen.

Soft error-pagina
Een speciale pagina met een gedetailleerde beschrijving van het probleem die wordt weergegeven als een HTTP-server de pagina die door een client is aangevraagd, niet kan verzenden en waarmee de HTTP-server zodanig wordt geconfigureerd dat deze speciale pagina wordt verzonden (in plaats van een antwoord met alleen een code waarmee wordt aangegeven wat het probleem is).

start-URL
Het startpunt voor het crawlen.

statische ranking
Een type ranking waarmee de rang wordt verhoogd aan de hand van de factoren voor de documenten waarvoor de ranking wordt uitgevoerd, zoals de datum, het aantal koppelingen dat naar het document verwijzen, enzovoort. Vergelijk met Dynamische ranking.

Statisch samenvatten
Een manier van samenvatten waarbij de opgegeven, opgeslagen samenvatting van het document in de zoekresultaten wordt weergegeven. Vergelijk met Dynamisch samenvatten.

Stopwoord
Een woord dat veel wordt gebruikt, zoals *de*, *een* of *en*, en dat in het zoekprogramma wordt genegeerd.

Stopwoorden verwijderen
Proces waarbij stopwoorden uit de query worden verwijderd zodat veelgebruikte woorden worden genegeerd en relevantere resultaten worden weergegeven.

Synoniemenwoordenboek
Een woordenboek waarin gebruikers kunnen zoeken naar synoniemen van de query tijdens het doorzoeken van een collectie.

Taalidentificatie
Een functie in enterprise search waarmee de taal van een document wordt bepaald.

Taalkundige zoekopdracht
Een type zoekopdracht waarvoor documenten wordt doorzocht, opgehaald en geïndexeerd en de termen worden gereduceerd tot de basisvorm (zodat *mice* bijvoorbeeld wordt geïndexeerd als *muis*) of worden uitgebreid met de basisvorm (zoals bij samengestelde woorden).

taxonomy
Een classificatie van objecten in groepen op basis van de overeenkomsten In enterprise search worden gegevens in categorieën en subcategorieën verdeeld met een taxonomie. Zie ook Categoriestructuur.

Tekennormalisatie
Een proces waarin de verschillende vormen van een teken, zoals hoofdlettergebruik en diakritische tekens, worden omgezet in de algemene vorm.

Teken voor terugloop met regelopschuiving
Een stuurcode waarmee de positie in de afdruk of op het scherm één regel omlaag wordt verplaatst. Voor bepaalde systemen zijn meerdere tekens vereist.

tekstanalyse
Proces waarbij semantische gegevens en andere gegevens uit de tekst worden geëxtraheerd, zodat de gegevens in een collectie beter kunnen worden gevonden.

tekstanalyseprogramma
Een softwarecomponent die verantwoordelijk is voor het zoeken en weergeven van de context en de semantische inhoud in de tekst.

tekstsegmentering
Zie segmentering.

Thema-extractie
Een type begripsextractie waarmee significante vocabulaire-items automatisch worden herkend in tekstdocumenten, zodat het thema of onderwerp van een document kan worden opgehaald. Zie ook Begripsextractie.

toegangslijst (access control list, ACL)
Een lijst die bestaat uit een of meer gebruikers-ID's of -groepen en de bijbehorende bevoegdheden. Toegangslijsten worden gebruikt om de toegang van gebruikers tot items en objecten te besturen.

Toelichting
Informatie over een tekstspanne. Zo kan met een toelichting worden aangegeven dat een tekstspanne voor een bedrijfsnaam staat. In Unstructured Information Management Architecture (UIMA) is een annotatie een speciaal soort featurestructuur.

Token De tekstuele basiseenheden die in enterprise search worden geïndexeerd. Tokens kunnen de woorden in een taal zijn of andere teksteenheden die in aanmerking komen voor indexering.

tokenisering
Zie segmentering.

tokenizer
Een tekstsegmeteringsprogramma waarmee tekst wordt gescand en wordt bepaald of en wanneer een reeks tekens als token kan worden herkend.

typesysteem
Het typesysteem definieert de typen objecten (featurestructuren) die door

een tekstanalyseprogrammakunnen worden ontdekt in een document. Het typesysteem definieert alle mogelijke featurestructuren in termen van typen en features. U kunt in een typesysteem een willekeurig aantal verschillende typen definiëren. Een typesysteem is specifiek voor een bepaald domein en een bepaald programma.

Uit wachtrij verwijderen
Items uit een wachtrij verwijderen.

Uniform Resource Identifier (URI)
Een compacte tekenreeks waarmee een abstracte of fysieke bron wordt aangegeven.

Uniform Resource Locator (URL)
Een tekenreeks waarmee informatiebronnen op een computer of in een netwerk (zoals internet) worden aangegeven. Deze tekenreeks bestaat uit de afkorting van het protocol waarmee toegang wordt verkregen tot de informatiebron, alsmede de informatie die het protocol gebruikt om de informatiebron op te zoeken.

Universal Resource Name (URN)
Een internetprotocolelement dat bestaat uit een korte tekenreeks die voldoet aan een bepaalde syntaxis. De tekenreeks bevat een naam die kan worden gebruikt om naar een resource te verwijzen.

Unstructured Information Management Architecture (UIMA)
Een IBM-architectuur waarmee een framework wordt gedefinieerd voor de implementatie van systemen voor de analyse van niet-gestructureerde gegevens.

Veld Het kleinste identificeerbare deel van een record.

veldzoekopdracht
Een query die beperkt is tot een bepaald veld.

Volledige tekstindex
Een gegevensstructuur waarmee naar gegevensitems wordt verwezen zodat met de zoekopdracht snel de documenten worden gevonden waarin de zoektermen voorkomen.

vrije zoekopdracht
Een zoekopdracht waarin de zoekterm wordt uitgedrukt als vrije tekst.

webcrawler
Een robotsoftwareklasse waarmee het internet wordt doorzocht door een webdocument op te halen en de koppelingen in dat document te volgen.

wegingsklasse
Een specificatie die van invloed kan zijn op de relatieve ranking van een document in de zoekresultaten.

XML Path-taal (XPath)
Een taal waarmee de unieke adresgedeelten van een bron-XML-document worden aangegeven. XPath biedt ook basisvoorzieningen voor de manipulatie van strings, getallen en booleaanse operatoren.

zoekcache
Een buffer waarin de gegevens en resultaten van eerdere zoekopdrachten zijn opgeslagen.

Zoeken bij benadering
Een zoekopdracht waarmee woorden als resultaat worden gegeven waarvan de spelling vergelijkbaar is met de spelling van de zoekterm.

Zoekindexbestanden
De set met bestanden waarin een index wordt opgeslagen in de zoekmachine.

Zoekmachine
Een programma waarin een zoekopdracht wordt geaccepteerd, waarna een lijst met documenten voor de gebruiker als resultaat wordt gegeven.

Zoekopdracht met gewogen termen
Een query waarin aan bepaalde termen een groter belang wordt toegekend.

zoekprogramma
Een programma waarmee query's worden verwerkt, de index wordt doorzocht, de zoekresultaten worden teruggezonden en de brondocumenten worden opgehaald voor collecties in een enterprise search-systeem.

Zoekresultaten
Een lijst met de documenten die aan de zoekopdracht voldoen.

Toegang krijgen tot informatie over Content Management en Discovery

Informatie over IBM Content Management- en Discovery-producten is beschikbaar via de telefoon en op internet.

De telefoonnummers in dit document zijn van toepassing op de Verenigde Staten:

- Als u andere producten wilt bestellen of algemene informatie wilt opvragen, belt u 1-800-IBM-CALL (1-800-426-2255)
- Als u publicaties wilt bestellen, belt u 1-800-879-2755

Op internet kunt u informatie over IBM Content Management- en Discovery-producten vinden op http://www.ibm.com/software/sw-bycategory/subcategory/SWB40.html. Deze site bevat links waarmee u:

- Meer te weten kunt komen over de producten
- De producten kunt kopen
- Kunt deelnemen aan betatests van de producten
- Productondersteuning kunt aanvragen

Ga als volgt te werk om productdocumentatie op te vragen:

1. Ga naar de internetpagina http://www.ibm.com/software/sw-bycategory/subcategory/SWB40.html.
2. Selecteer een product waarover u meer informatie wenst, bijvoorbeeld WebSphere Information Integrator OmniFind Edition. Deze site bevat links naar:
   - Productdocumentatie, waaronder opmerkingen bij de release (″release notes″) en online informatiecentra
   - Systeemvereisten
   - Productdownloads
   - Fixpacks
   - Nieuws over producten
   - Ondersteunend informatiemateriaal, zoals white papers en IBM Redbooks
   - Nieuwsgroepen en gebruikersgroepen
   - Instructies voor het bestellen van boeken
3. Klik op de link Support (links op de pagina).
4. Selecteer in de sectie Learn het type documentatie dat u wilt zien. Als voor het geselecteerde product een informatielink beschikbaar is, kunt u de link voor het Informatiecentrum selecteren.

Commentaar op de documentatie

U kunt uw commentaar op deze informatie of andere documentatie van IBM insturen.

Dankzij uw feedback kan IBM de documentatie steeds verder verbeteren. Mocht u commentaar hebben op deze informatie of op de andere documentatie van die IBM Software Development bij producten levert, laat ons dat dan weten. Dit kunt u op de volgende manieren doen:

1. U kunt uw commentaar insturen met behulp van het online formulier dat u kunt vinden op www.ibm.com/software/awdtools/rcf/.
2. U kunt uw commentaar per e-mail verzenden naar comments@us.ibm.com. Vergeet niet de naam van het product te vermelden, alsmede het versienummer van het product en de naam en het onderdeelnummer van de informatie (indien van toepassing). Als u commentaar hebt op een specifiek stuk tekst, geef dan aan waar die tekst zich bevindt (bijvoorbeeld: de titel, het nummer van een tabel of het paginanummer).

Contact opnemen met IBM

Als u contact wilt opnemen met de klantenservice van IBM in de Verenigde Staten of Canada, belt u 1-800-IBM-SERV (1-800-426-7378).

Als u meer informatie wilt over de beschikbare serviceopties, belt u een van de volgende nummers:

- In de Verenigde Staten: 1-888-426-4343
- In Canada: 1-800-465-9600

Als u een IBM-vestiging in uw land of regio zoekt, raadpleegt u de wereldwijde adressenlijst van IBM op www.ibm.com/planetwide.

Kennisgevingen en handelsmerken

Kennisgevingen

Deze informatie is ontwikkeld voor producten en diensten die in de Verenigde Staten worden aangeboden. Mogelijk levert IBM niet alle in dit document genoemde producten, diensten of functies in alle andere landen. Neem contact op met uw IBM-vertegenwoordiger voor informatie over de producten en diensten die bij u beschikbaar zijn. Verwijzing in deze publicatie naar producten of diensten van IBM houdt niet in dat uitsluitend IBM-producten of -diensten gebruikt kunnen worden. Functioneel gelijkwaardige producten of diensten kunnen in plaats daarvan worden gebruikt, mits dergelijke producten of diensten geen inbreuk maken op intellectuele eigendomsrechten of andere rechten van IBM. De gebruiker is verantwoordelijk voor de samenwerking van IBM-producten of -diensten met producten of diensten van anderen, tenzij uitdrukkelijk anders aangegeven door IBM.

Mogelijk heeft IBM octrooien of octrooi-aanvragen met betrekking tot bepaalde in deze publicatie genoemde producten. Aan het feit dat deze publicatie aan u ter beschikking is gesteld, kan geen recht op licentie of ander recht worden ontleend. Voor vragen over licenties kunt u zich wenden tot: IBM Director of Licensing IBM Corporation North Castle Drive Armonk, NY 10504-1785 U.S.A.

Voor licentievragen over DBCS-informatie (Double Byte Character Set) neemt u contact op met het IBM Intellectual Property Department in uw land of stelt u de vragen schriftelijk aan: IBM World Trade Asia Corporation Licensing 2-31 Roppongi 3-chome, Minato-ku Tokyo 106-0032, Japan

De volgende alinea is niet van toepassing op het Verenigd Koninkrijk, noch op enig ander land waar dergelijke bepalingen niet stroken met de lokale wetgeving: INTERNATIONAL BUSINESS MACHINES BIEDT DEZE PUBLICATIE AAN OP "AS IS"-BASIS, ZONDER ENIGE GARANTIE, UITDRUKKELIJK NOCH STILZWIJGEND, MET INBEGRIP VAN, MAAR NIET BEPERKT TOT, IMPLICIETE GARANTIES VAN VERHANDELBAARHEID, GESCHIKTHEID VOOR EEN BEPAALD DOEL OF DE GARANTIE DAT DEZE PUBLICATIE GEEN INBREUK MAAKT OP DE RECHTEN VAN DERDEN. In bepaalde rechtsgebieden is het uitsluiten of beperken van uitdrukkelijke of stilzwijgende garanties niet toegestaan; zodat het bovenstaande mogelijk niet op u van toepassing is.

Deze informatie kan technische onjuistheden en/of drukfouten bevatten. IBM kan zonder voorafgaand bericht wijzigingen en/of verbeteringen aanbrengen in de producten en/of programma's die in deze publicatie worden beschreven.

Verwijzingen in deze publicatie naar niet-IBM websites mogen niet worden opgevat als een aanbeveling van die websites. Het materiaal op dergelijke websites maakt geen deel uit van het materiaal voor dit IBM-product en het gebruik van dergelijke websites is geheel voor eigen risico.

IBM mag informatie die door u wordt verstrekt gebruiken en distribueren op elke manier die haar goeddunkt zonder daarbij verplichtingen jegens u aan te gaan.

Licentiehouders die informatie over dit programma willen ontvangen over: (i) het uitwisselen van informatie tussen in eigen beheer gemaakte programma's en

andere programma's (waaronder dit programma) en (ii) het gemeenschappelijk gebruik van de uitgewisselde informatie, dienen contact op te nemen met:

IBM Corporation J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003 U.S.A.

Dergelijke informatie kan beschikbaar zijn onder bepaalde voorwaarden en bepalingen waaronder, in bepaalde gevallen, betaling van een vergoeding.

Het gelicentieerde programma dat in dit document wordt beschreven en al het bij dit programma behorende materiaal, wordt door IBM geleverd onder de voorwaarden omschreven in de IBM Klantenovereenkomst, de IBM Internationale programmalicentie-overeenkomst of enige andere gelijkwaardige overeenkomst.

Prestatiegegevens die hierin worden vermeld, zijn verzameld in een gecontroleerde omgeving. De resultaten die in een andere verwerkingsomgeving worden behaald, kunnen hiervan derhalve aanzienlijk afwijken. Bepaalde metingen zijn verricht op systemen die nog in ontwikkeling, waren en er kan geen garantie worden gegeven dat deze metingen op algemeen verkrijgbare systemen gelijk zullen zijn. Bovendien zijn sommige metingen niet meer dan schattingen die door extrapolatie tot stand zijn gekomen. De feitelijke resultaten kunnen hiervan afwijken. Gebruikers van dit document dienen de feitelijke gegevens in hun eigen specifieke omgeving vast te stellen.

Informatie over niet door IBM geleverde producten is verkregen van de leveranciers van de desbetreffende producten, uit de publicaties van deze leveranciers of uit andere publiek toegankelijke bronnen. IBM heeft die producten niet getest en kan niet bevestigen dat de gegevens op het gebied van prestaties, compatibiliteit of enig ander gebied, correct zijn. Vragen met betrekking tot de mogelijkheden van niet-IBM producten dienen te worden gericht aan de leveranciers van die producten.

Alle uitspraken met betrekking tot de strategie of plannen van IBM kunnen zonder voorafgaand bericht worden gewijzigd of ingetrokken. Dergelijke uitspraken geven uitsluitend doelstellingen aan.

Deze informatie bevat voorbeelden van gegevens en rapporten die in de dagelijkse bedrijfsvoering worden gebruikt. Voor een zo volledig mogelijke illustratie zijn in deze voorbeelden namen van personen, bedrijven, merken en producten opgenomen. Al deze namen zijn fictief en enige gelijkenis met de namen en/of adressen van werkelijke personen of bedrijven berust uitsluitend op toeval.

AUTEURSRECHTLICENTIE:

Deze informatie bevat voorbeeldprogramma's in brontaal ter illustratie van de programmeertechnieken op diverse besturingsplatforms. U mag deze voorbeeldprogramma's zonder betaling aan IBM in elke vorm kopiëren, aanpassen en distribueren, mits dit tot doel heeft het ontwikkelen, gebruiken, verkopen of distribueren van toepassingsprogramma's die voldoen aan de Application Programming Interface voor het besturingsplatform waarvoor de voorbeeldprogramma's zijn geschreven. Deze voorbeelden zijn niet onder alle omstandigheden uitvoering getest. IBM kan de betrouwbaarheid, onderhoudbaarheid en functionaliteit van deze programma's derhalve niet garanderen of impliceren. U mag deze voorbeeldprogramma's zonder betaling aan IBM in elke vorm kopiëren, aanpassen en distribueren, mits dit tot doel heeft het ontwikkelen, gebruiken, verkopen of distribue-

ren van toepassingsprogramma's die voldoen aan de Application Programming Interface voor het besturingsplatform waarvoor de voorbeeldprogramma's zijn geschreven.

Bij elk exemplaar of onderdeel van deze voorbeeldprogramma's, of afgeleide versies hiervan, moet een auteursrechtvermelding worden opgenomen, volgens het onderstaande voorbeeld:

Outside In (®) Viewer Technology, © 1992-2006 Stellent, Chicago, IL., Inc. Alle rechten voorbehouden.

IBM XSLT Processor Gelicentieerd materiaal - Eigendom van IBM ©Copyright IBM Corp., 1999-2006. Alle rechten voorbehouden.

Handelsmerken

In dit onderwerp vindt u een overzicht van de IBM-handelsmerken en enkele niet-IBM-handelsmerken.

Zie http://www.ibm.com/legal/copytrade.shtml voor meer informatie over IBM-handelsmerken.

De volgende termen zijn handelsmerken of geregistreerde handelsmerken van andere bedrijven:

Java en alle op Java gebaseerde merken en logo's zijn handelsmerken of geregistreerde handelsmerken van Sun Microsystems, Inc. in de Verenigde Staten en/of andere landen.

Microsoft, Windows, Windows NT en het Windows-logo zijn handelsmerken van Microsoft Corporation in de Verenigde Staten en/of andere landen.

Intel, Intel Inside (logo's), MMX en Pentium zijn handelsmerken van Intel Corporation in de Verenigde Staten en/of andere landen.

UNIX is een handelsmerk van The Open Group in de Verenigde Staten en andere landen.

Linux is een handelsmerk van Linus Torvalds in de Verenigde Staten en/of andere landen.

Andere namen van bedrijven, producten en diensten kunnen handelsmerken zijn van derden.

Trefwoordenregister

U

W

X

Z

Gedrukt in Nederland

SC18-9674-01