IBM WebSphere Information Integrator
OmniFind Edition

Intégration de l'analyse de texte

Version 8.3

SC11-2398-00

IBM WebSphere Information Integrator
OmniFind Edition

Intégration de l'analyse de texte

Version 8.3

SC11-2398-00

Important
Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section "Remarques".

Première édition - novembre 2005

Réf. US : SC18-9674-00

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- http://www.fr.ibm.com (serveur IBM en France)
- http://www.can.ibm.com (serveur IBM au Canada)
- http://www.ibm.com (serveur IBM aux Etats-Unis)

Compagnie IBM France
Direction Qualité
Tour Descartes
92066 Paris-La Défense Cedex 50

Table des matières

Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.

OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
(Pos1)		Home
Fin	Fin	End
(PgAr)		PgUp
(PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
(Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

A propos de ces rubriques

Ces informations permettent de créer et de déployer des solutions de recherche sémantique dans un système IBM WebSphere Information Integrator OmniFind Edition Version 8.3. La recherche sémantique permet de rechercher des concepts de niveau plus élevé et d'exprimer des relations dans une requête de recherche qui sont détectées à l'aide de l'analyse de texte.

WebSphere Information Integrator OmniFind Edition (WebSphere II OmniFind Edition) fournit une technologie appelée *recherche d'entreprise*. Les composants de la recherche d'entreprise sont installés en même temps que le produit WebSphere II OmniFind Edition. Le terme *recherche d'entreprise* est utilisé dans la documentation WebSphere II OmniFind Edition sauf dans le cas de références aux chemins d'installation et aux libellés des produits.

La documentation de l'analyse de texte de la recherche d'entreprise couvre les sections suivantes :

- Présentation du support linguistique dans l'entreprise
- Instructions concernant le mode d'intégration de l'analyse de texte personnalisée dans la recherche d'entreprise
- Instructions concernant le mode de mappage des structures de document XML
- Instructions concernant le mode d'ajout des résultats de l'analyse sélectionnés aux tables JDBC
- Instructions concernant le mode d'ajout des résultats de l'analyse à l'index de recherche d'entreprise afin d'activer la recherche sémantique
- Instructions concernant le mode d'inclusion des dictionnaires de mots avec degré de pondération, de mots vides, de synonymes lors de la recherche
- Présentation de l'analyse de texte effectuée automatiquement lors du traitement du document

A qui s'adressent ces sections ?

Ces informations s'adressent aux administrateurs système. Les développeurs de l'application de recherche sont chargés de la création et du déploiement des solutions de recherche sémantique dans la recherche d'entreprise.

La recherche d'entreprise offre un support de recherche sémantique pour les documents de texte. Les documents sont analysés, les résultats sont stockés et il est possible d'accéder à ces derniers lors de la recherche. L'association de l'analyse à la recherche d'entreprise pouvant indexer à la fois des mots et des étendues de texte active la recherche d'entreprise. Le but du support sémantique lors de la recherche est d'améliorer les résultats de la recherche de documents, autrement dit d'obtenir la meilleure collection de documents possibles correspondant à la requête.

Ces informations permettent de savoir comment intégrer l'analyse de texte personnalisée à la recherche d'entreprise et d'utiliser des dictionnaires de mots avec degré de pondération, de mots vides et de synonymes afin d'améliorer les résultats de la requête. Ces informations permettent également de savoir quel est le support linguistique de base inclus dans la recherche d'entreprise à tout moment lors du traitement de documents.

Pour utiliser ces informations de manière efficace, vous devez avoir une bonne connaissance des applications Web et des sources de données dans lesquelles effectuer des recherches.

Support linguistique pour la recherche sémantique

La recherche d'entreprise offre un support de recherche linguistique pour les documents dans la plupart des langues indo-européennes et dans les langues asiatiques, telles le japonais.

Le but du support linguistique lors de la recherche est d'améliorer les résultats de la recherche de documents et d'obtenir la collection de documents correspondant le plus possible à la requête.

Le traitement linguistique s'effectue en deux étapes : lors de l'ajout d'un document à l'index ou lorsqu'un utilisateur entre une requête pendant la recherche.

La recherche d'entreprise inclut uniquement la fonctionnalité granulaire linguistique qui est requise pour la détermination de la langue d'un document et pour la segmentation du flux d'entrée du document en mots ou en marqueurs sémantiques.

Si vous savez que la recherche sera restreinte à la recherche de mot clé ou à la recherche XML native qui utilise la structure du document, le traitement linguistique inclus dans la recherche d'entreprise est parfaitement adapté à vos besoins.

Toutefois, le traitement linguistique seul n'est pas toujours satisfaisant si vous souhaitez effectuer une recherche au delà des mots du document, comme dans les exemples suivants :

- Les informations ne sont pas toujours explicitement marquées, comme une adresse ou un numéro de téléphone. Le terme *numéro de téléphone* n'est pas utilisé. Le message électronique contient à la place une phrase du type ″vous pouvez me contacter au numéro suivant : 555-641-1805″.
- Dans le processus de veille à la concurrence, les documents mentionnent des concurrents et les biens qu'ils produisent. Ils indiquent également que le site Web d'un concurrent est passé d'une gamme de produits à une autre au cours des trois derniers mois.
- Dans la gestion de la relation client, des documents peuvent indiquer des problèmes liés aux freins automobiles dans des ateliers de réparation de la région de San Francisco. L'atelier de réparation décrit des situations, telles ″patin réglé suite à une fuite hydraulique″. De plus, ces rapports indiquent uniquement le nom de la rue de l'atelier de réparation et non l'adresse complète.
- Dans la recherche, les documents décrivent une protéine particulière et sa relation à une maladie, au moins, mentionnée dans le même paragraphe. Il existe plus de vingt noms différents de la protéine dans l'ensemble de la documentation disponible. Les documents ne mentionnent pas généralement le mot *maladie* mais uniquement le nom de la maladie elle-même.

Dans ces exemples, la recherche des éléments requis dans les vastes collections de sources d'informations qui existent aujourd'hui constitue de nouveaux défis pour lesquels une analyse sophistiquée est nécessaire. Cette analyse va au delà du niveau de segmentation et de l'analyse à l'aide de dictionnaires proposés dans la recherche d'entreprise. La plupart des informations intéressantes ne sont pas marquées dans le document d'origine. A la place, les informations doivent être analysées afin de reconnaître et trouver des sujets d'intérêt, par exemple, des

entités nommées, telles des personnes, des entreprise, des emplacements, des fonctions et des produits ainsi que des relations possibles entre ces entités.

IBM Unstructured Information Management Architecture (UIMA) est une architecture et une structure logicielle qui vous guide lorsque vous générez des fonctions d'analyse avancée dans la recherche d'entreprise. Ces fonctions sont nécessaires pour détecter et rechercher des informations importantes dans les collections de documents.

Concepts associés

«Intégration de l'analyse de texte personnalisée», à la page 3
UIMA (Unstructured information management architecture) est une architecture logicielle qui prend en charge la création, la reconnaissance, la composition et le déploiement des fonctions d'analyse de texte. A l'aide d'UIMA, vous pouvez générer une analyse de texte personnalisée.

«Présentation de l'architecture UIMA (Unstructured information management architecture)», à la page 4
UIMA (Unstructured information management architecture) est une architecture et une structure lgoicielle qui vous guide parmi les fonctions d'analyse avancée et qui vous aide à rechercher des informations dans les collections de documents.

Intégration de l'analyse de texte personnalisée

UIMA (Unstructured information management architecture) est une architecture logicielle qui prend en charge la création, la reconnaissance, la composition et le déploiement des fonctions d'analyse de texte. A l'aide d'UIMA, vous pouvez générer une analyse de texte personnalisée.

UIMA est une plateforme ouverte qui identifie les composants pour chaque fonction d'analyse distincte. Elle garantit que ces composants peuvent être facilement réutilisés et associés les uns aux autres.

Un des concepts centraux d'UIMA est le *moteur d'analyse*, qui est chargé de la reconnaissance et de la représentation du contenu de l'analyse dans les documents texte. Le composant logique d'analyse est appelé *annotateur*. Un annotateur se concentre sur une tâche d'analyse et n'est pas impliqué dans les autres étapes de traitement. Un moteur d'analyse peut contenir un seul annotateur ou peut être composé de plusieurs moteurs, chacun à son tour contenant des annotateurs.

Les informations générées par un moteur d'analyse sont appelées *résultats de l'analyse*. Dans l'idéal, les résultats de l'analyse correspondent aux informations que vous recherchez.

L'analyse linguistique avancée inclut une association des différentes tâches d'analyse. L'analyse commence, par exemple, par la détection de la langue et la segmentation et se poursuit par la reconnaissance de la classe des mots suivie d'une analyse grammaticale approfondie. La dernière étape inclut l'identification, par exemple, de la relation entre des substances chimiques et des symptômes particuliers. Chaque étape du processus d'analyse est suivie de l'étape suivante.

L'architecture UIMA fournit les blocs de génération de base permettant de créer, de tester et de déployer vos propres moteurs d'analyse. Elle ne fournit pas de fonctionnalité d'analyse linguistique sous la forme de moteurs d'analyse préconfigurés que vous pouvez déployer dans votre environnement UIMA.

Le SDK UIMA inclut une implémentation Java de la structure UIMA pour l'implémentation, la description, la composition et le déploiement des composants UIMA. Il fournit également un environnement de développement de type Eclipse qui inclut un ensemble d'outils et d'utilitaire pour l'utilisation de l'architecture UIMA. Pour plus d'informations sur Eclipse, consultez le site www.eclipse.org.

Pour pouvoir utiliser UIMA, vous devez installer le SDK UIMA. Le kit de développement est disponible dans le produit IBM developerWorks. Pour obtenir plus d'informations, reportez-vous à la zone WebSphere Information Integrator à l'adresse suivante http://www.ibm.com/developerworks/db2/zones/db2ii/. Pour obtenir des instructions sur le mode d'installation du SDK UIMA dans l'environnement Eclipse Interactive Development Environment, consultez la documentation UIMA.

Concepts associés

«Support linguistique pour la recherche sémantique», à la page 1
La recherche d'entreprise offre un support de recherche linguistique pour les documents dans la plupart des langues indo-européennes et dans les langues asiatiques, telles le japonais.

«Présentation de l'architecture UIMA (Unstructured information management architecture)»
UIMA (Unstructured information management architecture) est une architecture et une structure lgoicielle qui vous guide parmi les fonctions d'analyse avancée et qui vous aide à rechercher des informations dans les collections de documents.

Présentation de l'architecture UIMA (Unstructured information management architecture)

UIMA (Unstructured information management architecture) est une architecture et une structure lgoicielle qui vous guide parmi les fonctions d'analyse avancée et qui vous aide à rechercher des informations dans les collections de documents.

Une *structure de fonctions* est la structure de données sous-jacente qui représente le résultat d'une analyse. Il s'agit d'une fonction attribut-valeur. Chaque structure de fonctions est d'un type et chaque type a un ensemble défini de fonctions ou d'attributs valides (propriétés), tout comme une classe Java. Les fonctions ont une plage qui indique le type de valeur que la fonction doit avoir, par exemple String.

La plupart des algorithmes d'analyse, également appelés annotateurs, génèrent leurs résultats d'analyse sous la forme d'annotations. Les annotations sont un type spécial de structure de fonctions conçue pour le traitement de l'analyse linguistique. Une structure de fonctions s'étend sur un texte d'entrée et est définie en termes de positions de début et de fin du texte d'entrée.

Par exemple, un annotateur qui reconnaît des expressions monétaires crée pour le texte ″100,55 dollars des Etats-Unis″ une annotation de type `monetaryExpression` qui couvre le texte avec la fonction `currencySymbol` ayant la valeur ″$″.

Tous les annotateurs de UIMA utilisent les structures de fonctions pour stocker ou lire des informations. Autrement dit, toutes les données sont modelées en tant que structures de fonctions.

Le système type définit toutes les structures de fonctions possibles en termes de types et de fonctions, de la même manière qu'une hiérarchie de classes dans Java.

Toutes les structures de fonction sont représentées dans une structure de données centrale appelée *structure d'analyse commune*. Toutes les données d'échange sont gérées par l'utilisation de la structure d'analyse commune.

La structure d'analyse commune contient les objets suivants :

- Le document texte
- Description du système type qui indique les types, les sous-types et leurs fonctions
- Résultats de l'analyse qui décrivent le document ou une région du document
- Référentiel d'index qui prend en charge l'accès aux résultats de l'analyse et les itérations de ces derniers

Concepts associés

«Support linguistique pour la recherche sémantique», à la page 1
La recherche d'entreprise offre un support de recherche linguistique pour les documents dans la plupart des langues indo-européennes et dans les langues asiatiques, telles le japonais.

«Intégration de l'analyse de texte personnalisée», à la page 3
UIMA (Unstructured information management architecture) est une architecture logicielle qui prend en charge la création, la reconnaissance, la composition et le déploiement des fonctions d'analyse de texte. A l'aide d'UIMA, vous pouvez générer une analyse de texte personnalisée.

Flux de travaux pour l'intégration de l'analyse personnalisée

Des algorithmes d'analyse de texte personnalisée sont créés et testés à l'aide du SDK UIMA puis déployés et exécutés sur les collections de documents dans la recherche d'entreprise.

Pour développer des algorithmes d'analyse et les implémenter dans la recherche d'entreprise, procédez comme suit :

1. Planification et conception :
   a. Déterminez les informations à rechercher. Quels sont les documents que vous souhaitez extraire ? Quels sont les concepts et les relations nécessaires dans une tâche de recherche particulière ? Par exemple, des noms de produit et d'employé peuvent être nécessaires pour améliorer la recherche d'ordre général sur un site Web interne d'une entreprise pharmaceutique alors que les employés du service de recherche et développement ont besoin d'utiliser des variantes de noms de médicament et voir les relations médicament-cause-soin.
   b. Indiquez le type d'analyse de texte requis pour l'extraction des informations des documents dans lesquels effectuer la recherche.
   c. Si la collection contient des documents XML, déterminez si vous souhaitez exploiter le marquage XML dans votre solution. Dans la recherche d'entreprise, vous pouvez utiliser le marquage XML d'une des manières suivantes :
      - Si vous pouvez utiliser le marquage XML dans l'analyse personnalisée (par exemple, vos documents contiennent des éléments `<summary>` ou `<topic>` pouvant être utiles dans un annotateur de récapitulation ou de catégorisation), définissez des mappages XML à la structure d'analyse commune.
      - Si vous souhaitez utiliser le marquage XML, tel qu'il apparaît dans le document des requêtes, activez le mappage XML natif.
   d. Déterminez quelles sont les informations du résultat de l'analyse de texte stockées dans la structure d'analyse commune auxquelles vous souhaitez pouvoir accéder à l'aide de la recherche sémantique. Définissez le mappage de la structure à l'index.
   e. Déterminez si vous souhaitez stocker les résultats de l'analyse dans une base de données relationnelle, par exemple, afin de connaître les tendances et les associations en utilisant des applications d'exploration de données ou de génération de rapports. Définissez le mappage de la structure d'analyse commune aux tables JDBC.
   f. Concevez l'application de recherche sémantique. Déterminez l'utilisation des fonctions supplémentaires de la recherche sémantique. Concevez l'interface utilisateur.
2. Développez les activités du SDK UIMA
   a. Définissez les étapes de l'analyse individuelle.
   b. Décrivez le système type des mappages et des algorithmes de mappage.
   c. Développez les algorithmes d'analyse (annotateurs) pour chaque étape de l'analyse et intégrez les annotateurs aux moteurs d'analyse à l'aide du SDK

UIMA. Générez chaque analyse personnalisée à l'aide de la fonctionnalité de base (identification de langue et marquage sémantique) dans le module des annotateurs de base de la recherche d'entreprise.

d. Après avoir testé les algorithmes d'analyse dans UIMA, placez le moteur d'analyse dans un fichier PEAR (Processing Engine Archive). L'archive doit contenir uniquement vos algorithmes d'analyse et non la fonctionnalité linguistique de recherche d'entreprise de base.

3. Déployez les activités de recherche d'entreprise
   a. Téléchargez le fichier d'archive (.pear) du moteur d'analyse dans la recherche d'entreprise. Attribuez un nom au composant d'analyse afin que vous puissiez y faire référence dans la recherche d'entreprise.
   b. Associez une ou plusieurs collections de document au moteur d'analyse.
   c. Si possible, pour chaque collection, téléchargez et sélectionnez la configuration de mappage de l'élément XML au type UIMA définie pour l'analyse personnalisée.
   d. Lorsque cela est possible, pour chaque collection, téléchargez et sélectionnez la configuration de mappage de base de données définie pour l'analyse personnalisée.
   e. Pour chaque collection, téléchargez et sélectionnez la configuration de mappage d'index définie pour la recherche sémantique.
   f. Lorsque cela s'avère nécessaire, configurez l'application de recherche sémantique personnalisée. Déployez par exemple, l'interface utilisateur de recherche sur navigateur dans un serveur d'applications.
   g. Parcourez, analysez et indexez les documents de la collection de recherche sémantique comme vous le feriez pour une collection ayant recours à des mots clés.

Tâches associées

«Installation et exécution des annotateurs de base de recherche d'entreprise» Vous pouvez utiliser le module annotateur de base de recherche d'entreprise pour, d'une part, développer de nouveaux annotateurs basés sur le résultat des annotateurs de recherche d'entreprise et d'autre part, pour tester les annotateurs personnalisés dans le kit de développement logiciel (SDK) UIMA.

Installation et exécution des annotateurs de base de recherche d'entreprise

Vous pouvez utiliser le module annotateur de base de recherche d'entreprise pour, d'une part, développer de nouveaux annotateurs basés sur le résultat des annotateurs de recherche d'entreprise et d'autre part, pour tester les annotateurs personnalisés dans le kit de développement logiciel (SDK) UIMA.

L'ensemble des annotateurs de base inclut :

- **Annotateur d'ID de langue**

  Détecte la langue d'un document. Pour connaître les fonctions et les paramètres de configuration, reportez-vous au fichier du descripteur jlangid.xml.

- **Annotateur de recherche du dictionnaire FROST**

  Permet une création de marqueurs sémantiques et une détection de phrase effectuées en fonction des dictionnaires IBM LanguageWare. Pour les marques sémantiques, des informations linguistiques supplémentaires, par exemple la forme de base ou le lemme, sont générées. Pour connaître les fonctions et les paramètres de configuration, reportez-vous au fichier du descripteur jfrost.xml.

- **Marqueur sémantique d'espace**

  Effectue une marquage sémantique à base d'espaces sur tous les documents de langue européenne ou a recours à d'autres scripts utilisant la séparation par espace. De plus, l'annotateur peut effectuer le marquage sémantique n-gram sur les scripts de texte suivants : Arabe, han, hébreu, Hiragana, Katakana, laotien, mongolien, thaïlandais, yi et hangul. Pour connaître les fonctions et les paramètres de configuration, reportez-vous au fichier du descripteur jtok.xml.

Pour exécuter ces annotateurs dans l'architecture UIMA, le SDK UIMA doit être installé. Ce dernier est disponible sur le site Web IBM developerWorks à l'adresse http://www-128.ibm.com/developerworks/db2/zones/db2ii/.

Le module de l'annotateur de base de recherche d'entreprise est un fichier zip contenant les annotateurs d'analyse de texte utilisées dans la recherche d'entreprise. Ces annotateurs s'exécuteront toujours avant toute analyse personnalisée lors de l'analyse des documents dans la recherche d'entreprise.

Pour installer le module annotateur :

1. Recherchez le module d'annotateurs OF_base_annotators.zip dans l'installation de recherche d'entreprise (WebSphere Information Integrator OmniFind Edition) se trouvant dans le répertoire *RACINE_INSTALL_RECHERCHE_ENTREPRISE*/packages/uima.
2. Copiez le fichier zip dans le répertoire racine de l'installation du SDK UIMA.
3. Extrayez le fichier zip afin d'ajouter les fichiers d'annotateur de base de recherche d'entreprise à la structure de répertoire indiquée pour votre installation SDK UIMA.

Une fois que vous avez installé le module d'annotateurs de base, les fichiers du descripteur d'annotateur sont disponibles dans le dossier *INSTALL_SDK_UIMA*/docs/examples/descriptors/analysis_engine. Le fichier of_tokenization.xml répertorie les annotateurs de base dans l'ordre dans lequel ils sont utilisés dans la recherche d'entreprise.

Les fichiers de descripteur contiennent les valeurs de configuration autres que celles utilisées dans la recherche d'entreprise. Vous pouvez modifier les valeurs à des fins de débogage dans le SDK UIMA. Toutefois, ne modifiez pas les fichiers de descripteur dans le système de recherche d'entreprise. Les modifications apportées à ces fichiers peuvent générer une instabilité du système ou des incidents de performances.

Le module annotateur de base de recherche d'entreprise contient uniquement les dictionnaires requis pour le traitement des documents anglais. Si vous souhaitez traiter d'autres langues dans votre environnement de développement, suivez la procédure ci-après.

1. Recherchez les dictionnaires de recherche d'entreprise dans l'installation de recherche d'entreprise dans *RACINE_INSTALL_RECHERCHE_ENTREPRISE*/configurations/parserservice/jediidata/frost/resources.
2. Copiez le contenu du répertoire dans votre installation SDK UIMA locale dans *INSTALL_SDK_UIMA*/data/frost/resources .

Pour vérifier que l'installation du module d'annotateur a abouti, procédez comme suit :

1. Ouvrez le débogueur visuel CAS (CVD) dans le répertoire suivant : *INSTALL_SDK_UIMA*/bin/cvd[.bat/.sh].
2. Cliquez sur **Exécuter → charger TAE**.
3. Sélectionnez le fichier d'indicateur d'analyse de texte appelé of_tokenization.xml dans le répertoire *INSTALL_SDK_UIMA*/docs/examples/descriptors/analysis_engine.
4. Chargez un document exemple et exécutez le moteur d'analyse de texte. Vous verrez s'afficher des annotations du type `uima.tt.TokenAnnotation` dans le CVD.

Pour utiliser les annotateurs de recherche d'entreprise pour votre traitement :

1. Incluez une référence au fichier of_typesystem.xml dans la section typeSystem du spécificateur de votre annotateur personnalisé si celui-ci utilise des types définis par des annotateurs de recherche d'entreprise. Le fichier of_typesystem.xml se trouve dans le fichier of_typesystem.xml dans le répertoire *INSTALL_SDK_UIMA*/docs/examples/descriptors/analysis_engine. Pour obtenir un exemple du mode d'inclusion des références dans des fichiers de descripteur, voir le fichier jtok.xml dans le répertoire analysis_engine.

Référence associée

«Types et fonctions définis dans la recherche d'entreprise», à la page 45
Le système type défini dans la recherche d'entreprise couvre la gestion des métadonnées de document et l'analyse linguistique de base.

Algorithmes d'analyse de texte

Le SDK UIMA inclut des API et des outils qui permettent de créer des annotateurs (algorithmes d'analyse incluant la description du système type) et d'intégrer ces annotateurs dans les moteurs d'analyse.

La documentation UIMA inclut un tutoriel qui vous guide dans le processus de génération de ces composants. Le SDK inclut des utilitaires de test et d'affichage des résultats et un moteur de recherche sémantique à petite échelle pour l'indexation des résultats de la recherche. Vous pouvez également effectuer une recherche plus avancée pour les informations stockées dans l'index.

Le SDK UIMA ne fournit aucun moteur d'analyse pré-configuré. Toutefois, vous pouvez utiliser les annotateurs de base disponibles dans la recherche d'entreprise dans un environnement UIMA. Pour avoir comment inclure la fonctionnalité de marquage sémantique et de détection de langue avant d'utiliser les algorithmes d'analyse de texte lors de leur développement dans l'environnement UIMA, consultez la documentation UIMA.

Une fois que vous avez développé et testé les moteurs d'analyse à l'aide du SDK UIMA et que vous souhaitez utiliser ces algorithmes sur une collections de documents dans la recherche d'entreprise, vous devez créer un fichier PEAR (Processing Engine ARchive). Ce fichier d'archive inclut l'ensemble des ressources requises pour le déploiement de la fonctionnalité d'analyse personnalisée en tant que moteurs d'analyse dans la recherche d'entreprise. L'ensemble des procédures de traitement requises pour la création d'une archive sont décrites dans la documentation UIMA fournie dans le SDK.

L'archive doit contenir uniquement votre analyse personnalisée même si elle se fonde sur la fonctionnalité linguistique de base offerte dans la recherche

d'entreprise. Les étapes d'analyse de recherche d'entreprise de base sont toujours exécutées avant toute analyse personnalisée.

Pour savoir comment configurer et déployer une solution de recherche sémantique dans la recherche d'entreprise, suivez les instructions du tutoriel mentionné à l'adresse Web http://www.ibm.com/developerworks/db2/zones/db2ii/. Le tutoriel vous guide dans les procédures permettant de déployer des algorithmes d'analyse de texte personnalisé dans la recherche d'entreprise et vous indique également comment utiliser les résultats de l'analyse dans les requêtes afin d'améliorer les résultats de la recherche.

Tâches associées

«Installation et exécution des annotateurs de base de recherche d'entreprise», à la page 6
Vous pouvez utiliser le module annotateur de base de recherche d'entreprise pour, d'une part, développer de nouveaux annotateurs basés sur le résultat des annotateurs de recherche d'entreprise et d'autre part, pour tester les annotateurs personnalisés dans le kit de développement logiciel (SDK) UIMA.

Description du système type

Décrit les structures de fonctions (structures de données sous-jacentes qui représentent les résultats de l'analyse) utilisées dans l'analyse personnalisée.

Les mêmes types définis dans la description de système type doivent être utilisés par le moteur d'analyse qui contient les annotateurs (algorithmes d'analyse) et dans tous les fichiers de mappage associés à l'analyse personnalisée, qu'il s'agisse du fichier de configuration de mappage XML, du fichier de configuration de génération d'index ou du fichier de mappage de configuration de base de données compatible JDBC.

La description du système type d'un annotateur peut faire partie du descripteur de l'annotateur ou elle peut faire partie d'un fichier séparé du descripteur du type de système. Elle fait parfois partie du descripteur d'un autre annotateur contenu dans le même moteur d'analyse.

La description du système type doit faire partie de l'archive du moteur de balayage (fichier .pear) importé à partir de l'environnement UIMA dans la recherche d'entreprise.

La description du système type exemple qui suit est utilisée dans l'ensemble des sections qui décrivent les différents types de mappage que vous pouvez sélectionner avec l'analyse personnalisée.

L'exemple de description de système type suivant décrit des rapports de police qui contiennent des informations sur les suspects, le lieu du crime, l'heure du crime et la date :

```
<?xml version="1.0" encoding="UTF-8"?>
<typeSystemDescription>
   <name>Police Reports Type System</name>
   <description>Type system description for
        police reports</description>
   <version>1.0</version>
   <types>
     <typeDescription>
       <name>com.ibm.omnifind.types.PoliceReport</name>
       <description>Annotates a police report</description>
       <superTypeName>uima.tcas.Annotation</superTypeName>
```

```
    <features>
      <featureDescription>
        <name>time</name>
        <description>Time the crime was reported to have happened
           </description>
        <rangeTypeName>com.ibm.omnifind.types.Time</rangeTypeName>
      </featureDescription>
      <featureDescription>
        <name>date</name>
        <description>When the crime happened</description>
        <rangeTypeName>com.ibm.omnifind.types.Date</rangeTypeName>
      </featureDescription>
      <featureDescription>
        <name>location</name>
        <description>Where the crime took place</description>
        <rangeTypeName>com.ibm.omnifind.types.City</rangeTypeName>
      </featureDescription>
      <featureDescription>
        <name>knownSuspects</name>
        <description>Contains annotations of type Suspect</description>
        <rangeTypeName>uima.cas.FSArray</rangeTypeName>
      </featureDescription>
      <featureDescription>
        <name>crimeDescription</name>
        <description>Short description of the crime</description>
        <rangeTypeName>uima.cas.String</rangeTypeName>
      </featureDescription>
    </features>
  </typeDescription>
  <typeDescription>
    <name>com.ibm.omnifind.types.City</name>
    <description>The name of a city</description>
    <superTypeName>uima.tcas.Annotation</superTypeName>
    <features>
      <featureDescription>
        <name>cityName</name>
        <description>The name of the city</description>
        <rangeTypeName>uima.cas.String</rangeTypeName>
      </featureDescription>
      <featureDescription>
        <name>cityDistrict</name>
        <description>The name of the district</description>
        <rangeTypeName>uima.cas.String</rangeTypeName>
      </featureDescription>
    </features>
  </typeDescription>
  <typeDescription>
    <name>com.ibm.omnifind.types.Person</name>
    <description>A person annotation</description>
    <superTypeName>uima.tcas.Annotation</superTypeName>
    <features>
      <featureDescription>
        <name>role</name>
        <description>For example, suspect or witness</description>
        <rangeTypeName>uima.cas.String</rangeTypeName>
      </featureDescription>
      <featureDescription>
        <name>firstName</name>
        <description>The first name of the person</description>
        <rangeTypeName>uima.cas.String</rangeTypeName>
      </featureDescription>
      <featureDescription>
        <name>surName</name>
        <description>The surname of the person</description>
        <rangeTypeName>uima.cas.String</rangeTypeName>
      </featureDescription>
      <featureDescription>
```

```
      <name>title</name>
      <description>For example, Mr. or Ms.</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>gender</name>
      <description>Male or female</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Suspect</name>
  <description>A found suspect</description>
  <superTypeName>com.ibm.omnifind.types.Person</superTypeName>
  <features>
    <featureDescription>
      <name>description</name>
      <description>Suspect description,
    for example, bearded with dark glasses</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Date</name>
  <description>A date</description>
  <superTypeName>uima.tcas.Annotation</superTypeName>
  <features>
    <featureDescription>
      <name>year</name>
      <description>The year, for example, 2005</description>
      <rangeTypeName>uima.cas.Integer</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>month</name>
      <description>The month in digits, for example, 7</description>
      <rangeTypeName>uima.cas.Integer</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>day</name>
      <description>The day in digits</description>
      <rangeTypeName>uima.cas.Integer</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>dayOfWeek</name>
      <description>The day of the week, for example, Monday</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>quarter</name>
      <description>The quarter, for example, Q1-2005</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>englDate</name>
      <description>Date as mm/dd/yyyy</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Time</name>
  <description>A time</description>
  <superTypeName>uima.tcas.Annotation</superTypeName>
  <features>
    <featureDescription>
```

```
            <name>hours</name>
            <description>Hours from 00-23</description>
            <rangeTypeName>uima.cas.Integer</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>minutes</name>
            <description>Minutes in the hour</description>
            <rangeTypeName>uima.cas.Integer</rangeTypeName>
          </featureDescription>
          <featureDescription>
            <name>timeOfDay</name>
            <description>Time periods, such as morning, noon</description>
            <rangeTypeName>uima.cas.String</rangeTypeName>
          </featureDescription>
        </features>
      </typeDescription>
    </types>
</typeSystemDescription>
```

Référence associée

«Types et fonctions définis dans la recherche d'entreprise», à la page 45
Le système type défini dans la recherche d'entreprise couvre la gestion des métadonnées de document et l'analyse linguistique de base.

«Types et fonctions définis dans l'architecture UIMA», à la page 48
Le SDK UIMA définit des types linguistiques de base et des fonctions pouvant être reconnues dans un document lors de l'analyse du texte.

Marquage XML dans l'analyse et la recherche

Vous pouvez mapper les informations des structures XML d'un document directement à une structure d'analyse commune sans l'aide d'un annotateur UIMA.

Si les documents de votre collection sont au format XML et que vous souhaitez exploiter le marquage XML lors de l'analyse de texte ou de la recherche sémantique, vous disposez des options suivantes :

Recherche XML native

Employez cette option si vous souhaitez utiliser l'ensemble des balises et attributs XML, tels qu'ils apparaissent dans le document lors de la recherche sémantique. Par exemple, si vous avez des documents de facturation qui contiennent un élément `<addressee>`, l'activation de la recherche XML native permet d'utiliser cette balise dans une requête de recherche sémantique pour un nom de client particulier dans cet élément.

Avec cette option, la structure XML du document est représentée dans la structure d'analyse commune à l'aide du type `com.ibm.es.tt.MarkupTag`. Pour chaque balise XML, une annotation de ce type est créée. Cette annotation contient le nom de la balise, ses attributs et le contenu de l'attribut. Ces informations sont toujours indexées et sont accessibles pour la recherche sémantique.

La recherche XML native ne requiert aucun fichier de configuration de mappage. Vous pouvez activer la recherche XML native à partir de la console d'administration pour la recherche d'entreprise.

Mappage d'un élément XML au type UIMA

Utilisez cette option dans les situations suivantes :

- La sémantique de certains éléments XML est précise et peut être utilisée pour l'analyse de texte. Les étapes de l'analyse peuvent être effectuées directement sur les annotations et les fonctions créées à partir des

structures XML et sont dérivées des différents formats potentiels des documents d'origine. Par exemple, l'élément <addressee> des documents concernant la facturation contient généralement des noms de client. Lors de l'utilisation du mappage de l'élément XML au type, le contenu de cet élément peut être mappé directement aux annotations de type Customer. Un annotateur peut alors déduire une relation client-situé-à, à l'aide des informations entourant l'annotation Customer.

- Vous souhaitez limiter l'étendue du traitement d'un annotateur personnalisé à des zones spécifiques dans l'entrée XML. Vous pouvez, par exemple, limiter le contenu des balises <technicianComment> dans un annotateur qui détecte les problèmes liés aux voitures.
- Vous souhaitez restreindre le traitement d'analyse de texte ainsi que les recherches suivantes à certaines parties du document XML et supprimer le contenu non textuel ou inapproprié.
- Vous souhaitez mapper des balises XML ayant des noms différents (par exemple, <mainHeading> ou <doc>) à une étendue commune à utiliser dans la recherche sémantique (par exemple, title).

Dans ces cas, vous devez créer un fichier de configuration de mappage d'élément XML à un type UIMA qui définit des structures de fonctions. Les structures de fonctions que vous définissez dans le fichier de configuration sont créées lors de l'analyse des documents et il est possible d'y accéder à l'aide des annotateurs personnalisés.

Vous pouvez utiliser plusieurs fichiers de configuration pour une collection de documents. L'élément <identifier> permet de déterminer quelle configuration est utilisée pour quel document XML. L'élément <identifier> du fichier de configuration doit correspondre à l'élément principal du document XML. Par exemple, si l'élément principal du document est doc, la valeur de l'élément <identifier> du fichier de configuration doit également être "doc".

Si aucune correspondance n'est trouvée, le programme recherche un fichier de configuration avec l'élément <identifier> ayant la valeur par défaut. Si aucune configuration par défaut n'est trouvée, les sections de texte du document (sans informations de balise) sont mappées à l'annotation de document dans la structure d'analyse commune.

Si vous souhaitez extraire des informations qui se trouvent uniquement dans les parties appropriées d'un document tout en ignorant les parties inappropriées, il vous suffit d'indiquer quels éléments XML des documents contiennent les informations appropriées. Cette opération est appelée extraction. Par exemple, vous pouvez extraire les informations se trouvant dans les sections title et body tout en ignorant les informations des sections author, date, ID et publisher.

L'extraction du contenu peut améliorer le traitement de l'analyse pour les types de document XML suivants :

- Les documents incluant des grandes quantités de contenu non soumis à l'analyse, des pièces jointes binaires, par exemple. L'utilisation de l'extraction de contenu réduit de manière significative la taille du document, ce qui fait que le traitement est plus rapide et les erreurs d'analyse dues au fait que des données inappropriées sont utilisées sont évitées.
- Les documents contenant du texte inapproprié, par exemple, des documents contenant des informations éditoriales dans des balises <note>. Le fait d'ignorer ces informations génère un meilleur résultat lors de l'analyse du contenu du document.

La recherche XML native et les options d'extraction de contenu dans le mappage d'élément XML au type UIMA sont en contradiction car soit l'ensemble du contenu, soit le contenu indiqué uniquement peut être pris en compte. Si vous indiquez l'extraction de contenu, le mappage XML natif est ignoré. Sans l'extraction de contenu, vous pouvez avoir à la fois le mappage d'élément XML au type UIMA et la recherche XML native.

Tous les types et fonctions utilisés dans le fichier de configuration doivent être présentés dans la description du système type de la procédure d'analyse personnalisée. Vous pouvez créer un descripteur de système type dans votre environnement UIMA à l'aide du module d'extension Component Descriptor Editor Eclipse. Ce dernier permet de créer un fichier de descripteur sans qu'il soit nécessaire de connaître la syntaxe XML requise.

Une fois que vous avez généré et testé l'analyse personnalisée, utilisez l'assistant de génération UIMA PEAR (Processing Engine ARchive) pour créer une archive qui contient les fichiers d'analyse personnalisée, notamment la description du système type.

Vous pouvez alors télécharger l'archive d'analyse personnalisée et votre élément XML dans des fichiers de configuration de mappage de type UIMA dans la recherche d'entreprise à l'aide de la console d'administration pour la recherche d'entreprise.

Tâches associées

«Création d'un fichier de configuration de mappage de types XML en UIMA» Dans un fichier de configuration de mappage des types XML en UIMA, vous pouvez utiliser la gamme complète des options de configuration pour le mappage des types de données XML en UIMA.

Création d'un fichier de configuration de mappage de types XML en UIMA

Dans un fichier de configuration de mappage des types XML en UIMA, vous pouvez utiliser la gamme complète des options de configuration pour le mappage des types de données XML en UIMA.

A propos de cette tâche

Le fichier de configuration de mappage de types XML en UIMA doit être compatible avec le schéma affiché dans l'exemple suivant.

Le rapport de police XML exemple comporte des balises XML pour le type de crime, la date du crime, le lieu du crime, l'officier de police ayant effectué le rapport, le commissariat dont dépend l'officier de police, la description du suspect et un résumé des faits. Tous ces éléments sont suivis d'une section body. Par exemple :

```
<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>Une Mercedes modèle CLK  a été volée le 23 avril 2005 sur
   un parking devant le restaurant Blue Lagoon,
   27 Main Street, Brynston.(numéro de série : 32 2761 50871)</abstract>
  <body>Une Mercedes modèle CLK a été volée le 23 avril 2005 sur
   un parking devant le restaurant Blue Lagoon, 27 Main Street,
   Brynston.(numéro de série : 32 2761 50871)

Elle est de couleur noir et a des pneus Michelin.

Des témoins devant le restaurant ont vus des individus de sexe masculin
portant des vêtements de couleur sombre s'enfuir au volant de la voiture.
La voiture a été retrouvée abandonnées à l'adresse suivante à Brooklyn :
Aliway Ave. Le réservoir d'essence était vide.
Les sièges ont été tâchées et les sièges arrières saccagés.
Aucun objet n'a été volé dans la voiture....</body>
 </doc>
 <image>
  </-- image of the crime scene as a base64-encoded string -->
 </image>
</report>
```

Un fichier de configuration exemple créé à partir du rapport exemple peut avoir la structure suivante. L'exemple utilise le système type défini pour le scénario de rapport de police.

```
<?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>
```

Restrictions

Le fichier de configuration de mappage XML est fractionné en deux sections :

<contentElements> element
: Utilisez cet élément si vous souhaitez une extraction de contenu spécifique. Le fichier de configuration exemple extrait le contenu dans la section <doc> d'un document et ignore les autres sections du document. Dans le rapport de police XML, l'image peut être de grande taille et peu utile pour le traitement de texte. En indiquant <doc> en tant qu'élément de contenu et non <image>, l'image est supprimée avant qu'aucun traitement de texte ne commence.

<elementToTypeMappings>
: Utilisez cet élément pour indiquer quels éléments XML individuels (indiqué dans un élément <elementToTypeMapping>) du document doivent être mappés à quelles structures de fonctions dans la structure d'analyse commune.

  Si vous utilisez l'option d'extraction de contenu, les éléments XML indiqués dans la section <elementToTypeMappings> doivent être inclus dans les éléments XML indiqués dans la section <contentElements>.

Procédure

Pour créer un fichier de configuration de mappage des types XML en UIMA, procédez comme suit :

1. Créez un fichier XML. Pour éviter des erreurs de syntaxe XML, utilisez un éditeur XML ou un outil de création XML pour valider les éléments XML. Le schéma XSD du fichier de configuration est appelé configuration.xsd et se trouve dans le'installation de recherche d'entreprise à l'emplacement *RACINE_INSTALL_RECHERCHE_ENTREPRISE*/packages/uima/.
2. Incluez vos mappages dans un élément <xmlCasInitializerConfiguration xmlns="http://www.ibm.com/2005/uima/jedii_ci_xml">. L'espace de nom (indiqué dans l'attribut xmlns) doit être exactement identique à celui affiché.
3. Ajoutez un élément <contentElements> si vous souhaitez extraire un contenu spécifique des sections du document et un élément <elementToTypeMappings> qui définit quel élément XML individuel du document doit être mappé à quelles structures de fonctions dans la zone d'analyse commune.
4. Ajoutez un élément <identifier> et un élément <description>. L'identificateur détermine quelle configuration doit être utilisée avec quel document XML. L'identificateur doit contenir l'élément principal du document, tel que doc. Si l'identificateur a la valeur par défaut, l'élément principal du document est inapproprié et le mappage de configuration est appliqué à un document XML.
5. Ajoutez un élément <contentElements> si vous souhaitez extraire les informations qui se trouvent uniquement dans les parties appropriées d'un document. Il comporte l'élément de composant suivant :
   - Un ou plusieurs éléments <element> qui contiennent le chemin d'un élément XML du document et qui respectent la syntaxe XPath, par exemple <element>/doc/crimeType</element>.

6. Ajoutez un élément <elementToTypeMappings> si vous souhaitez indiquer quels éléments XML du document doivent être mappés à quelles structures de fonctions dans la structure d'analyse commune. Il comporte les éléments de composant suivants :
   - Un ou plusieurs éléments <elementToTypeMapping>. Cet élément doit avoir les éléments imbriqués suivants :
     - Un élément <element> permettant de définir le chemin d'un élément XML et respectant la syntaxe XPath. Une barre oblique en début de chaîne signifie qu'un chemin complet a été indiqué. Par exemple, abstract sous l'élément principal doc. Deux barres obliques (//) correspondent à un sous-ensemble de chemins. Par exemple, birthDate doit être placé dans reportingOfficer même si d'autres éléments peuvent être placés entre ces deux éléments.
     - Un élément <type>, qui indique un type défini dans la description de système type. Il doit être de type Annotation.
     - Aucun élément <featureValueAssignment> ou plusieurs éléments de ce type.
7. Dans un élément <featureValueAssignment>, attribuez un nom à la fonction de type String dans l'élément <feature> et attribuez une valeur dans l'élément <basicValue>. Plusieurs éléments <basicValue> peuvent être ajoutés entre un élément <values>.

   L'élément <basicValue> peut avoir des attributs, notamment useAttributeValue, useElementContent, default et trim.

Utilisez useAttributeValue si vous voulez utilisez la valeur d'un attribut en tant que valeur d'une fonction. Exemple :

```
<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>
```

Résultats dans la sortie suivante :

- Pour chaque balise XML <reportingOfficer> détectée dans une balise XML <doc> du document, une structure de fonctions de type com.ibm.omnifind.types.Person est créée.
- Si la balise <reportingOfficer> contient un attribut sex, la fonction gender de la structure de fonctions nouvellement créée a la valeur de l'attribut.

Utilisez l'attribut useElementContent pour ajouter le contenu en tant que valeur d'une fonction. Par exemple, dans le fragment de configuration suivant :

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

le texte couvert par l'élément <abstract> dans <doc> devient la valeur de la structure de fonctions crimeDescription. Tous les espaces de fin et de début sont supprimés.

Plusieurs valeurs peuvent être indiquées entre l'élément <values> dans les situations suivantes :

- La fonction à configurer est de type StringArray.
- Un grand nombre de chaînes sont concaténées en une chaîne à l'aide de l'attribut delimiter et donc mappent à une fonction de type String. Par exemple, le titre Mr. est une constante, le prénom est la valeur d'un attribut et le nom de famille est couvert par un élément XML :

```
<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>
```

Les valeurs de la fonction de chaîne sont extraites du fichier de configuration en l'état. Les valeurs conservent tous les espaces de début et de fin. Toutefois, les espaces sont supprimés des noms de type et de fonction. Par exemple, <type> **com.ibm.omnifind.types.Person** </type> devient <type>**com.ibm.omnifind.types.Person**</type>.

Définissez les conditions des attributs à l'aide de l'élément <condition>. Par exemple, la structure de fonctions de type com.ibm.omnifind.types.Person est créée uniquement si <suspectDescription> se trouve dans le document avec l'attribut armed ayant la valeur yes :

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

En fonction du rapport de police exemple et du fichier de configuration de mappage défini, les structures de fonctions suivantes sont créées :

com.ibm.omnifind.types.PoliceReport

- covered text: "Vol de voiture 23/04/05 09:23 27 Main Street, Brynston, Springfield, New Jersey Jakob Collins 14e circonscription Individu de sexe masculin, cheveux foncés, lunettes, pantalon jeans avec une veste foncée sans doute noire, une Mercedes modèle CLK a été ... Aucun objet n'a été volé dans la voiture.
- begin = 2
- end = 904
- knownSuspects = null
- crimeDescription = "Une Mercedes modèle CLK a été volée le 23 avril 2005 sur un parking devant le restaurant Blue Lagoon restaurant, 27 Main Street, Brynston.(numéro de série : 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″

Une fois que vous avez créé le fichier XML, vous devez le télécharger dans la recherche d'entreprise et sélectionner le fichier de configuration de mappage de document XML ainsi que d'autres options d'analyse personnalisée dans la console d'administration de recherche d'entreprise.

Concepts associés

«Marquage XML dans l'analyse et la recherche», à la page 12
Vous pouvez mapper les informations des structures XML d'un document directement à une structure d'analyse commune sans l'aide d'un annotateur UIMA.

Référence associée

«Description du système type», à la page 9
Décrit les structures de fonctions (structures de données sous-jacentes qui représentent les résultats de l'analyse) utilisées dans l'analyse personnalisée.

Résultats de l'analyse de texte

Tous les résultats de l'analyse de texte sont stockés dans la structure d'analyse commune.

Les annotateurs placent et lisent des données dans la structure d'analyse commune. Les clients de structure d'analyse commune (*clients CAS*) effectuent le traitement final sur les résultats de l'analyse stockés dans la structure d'analyse commune. La recherche d'entreprise contient deux clients CAS :

- Client qui indexe le contenu de la structure d'analyse commune dans un moteur de recherche. Ce client requiert un fichier de configuration de génération d'index que vous sélectionnez à l'aide de l'analyse de texte personnalisée dans la console d'administration de recherche d'entreprise.
- Client qui charge des résultats d'analyse spécifiques dans une base de données relationnelle. Ce client requiert également un fichier de configuration que vous sélectionnez à l'aide des options d'analyse de texte personnalisée dans la console d'administration de recherche d'entreprise.

Les clients CAS lisent les données à partir de la structure d'analyse commune.

Si nécessaire, vous pouvez déployer des clients CAS personnalisés dans la recherche d'entreprise. Pour savoir comment créer un client, consultez la documentation UIMA. Pour savoir comment télécharger et utiliser le client dans la recherche d'entreprise, voir le site Web IBM UIMA developerWorks à l'adresse http://www.ibm.com/developerworks/db2/zones/db2ii/.

Concepts associés

«Mappage d'index pour les résultats de l'analyse personnalisée», à la page 24
Après avoir exécuté l'analyse personnalisée sur une collection de documents, vous pouvez utiliser le moteur de recherche dans la recherche d'entreprise afin

de générer un index à partir des informations stockées dans la structure d'analyse commune créée par les algorithmes d'analyse personnalisée.

«Mappage de base de données pour les résultats de l'analyse sélectionnés», à la page 32
Une fois que vous avez exécuté l'analyse personnalisée sur une collection de documents dans la recherche d'entreprise, vous pouvez stocker les résultats de l'analyse de texte sélectionnés dans une base de données compatible JDBC.

Chemins de fonction

Un chemin de fonctions permet d'accéder à des valeurs de fonctions dans les structures d'analyse commune, de la même manière que les instructions XPath permettant d'accéder aux éléments XML d'un document XML.

Les chemins de fonctions sont utiles si vous souhaitez accéder à une structure de fonctions qui associe des fonctions complexes, par exemple des fonctions de valeurs de tableau ou qui désignent une autre structure de fonctions. A l'aide d'un chemin de fonction, vous pouvez associer la valeur d'une fonction directement à une structure de fonctions et stocker cette valeur dans l'index de recherche sémantique ou dans une base de données.

Prenons l'exemple d'un annotateur qui identifie les voitures et leurs marques. Il crée des annotations de type `car` ayant un attribut `make`. Toutefois, l'attribut `make` ne contient pas le nom réel de l'entreprise (par exemple, Chevrolet) mais une structure de fonctions de type `Company` qui contient elle-même un attribut `companyname`. Pour activer une requête sémantique qui associe des noms de voiture à des noms d'entreprise, un chemin de fonction `make/companyname` permet d'associer la valeur de `companyname` à l'étendue car générée pour l'annotation car. Permet d'activer la requête ″Extraire les documents qui contiennent les voitures fabriquées par Chevrolet″ en utilisant '/car[@make=″Chevrolet″]'.

Un chemin de fonctions est une suite de noms de fonction (f1/.../fn) avec les propriétés suivantes :

- La valeur d'un chemin de fonctions peut être String, Integer, Float ou un tableau d'un de ces types.
- Toutes les fonctions de ce chemin de f1 à fn-1 doit avoir un type complexe, c'est-à-dire de type `uima.cas.TOP`, `uima.cas.FSArray`, `uima.cas.FSList` ou d'un de ses sous-types.
- La dernière fonction du chemin peut inclure un type complexe. Elle peut également inclure un type ou un sous-type de `uima.cas.Float`, `uima.cas.Integer`, `uima.cas.String`, `uima.cas.FloatArray`, `uima.cas.IntegerArray`, `uima.cas.StringArray`, `uima.cas.FloatList`, `uima.cas.IntegerList` ou `uima.cas.StringList`.
- Vous pouvez également saisir une fonction. Le nom de la fonction doit être ajouté avant le nom du type complet et être séparé par le caractère deux points. Par exemple, `f1/com.ibm.es.SomeType:f2/.../fn`.

Vous pouvez restreindre la portée type d'une fonction particulière. Par exemple, utilisons une fonction `additionalInfo` de type `uima.cas.TOP`. Si vous savez que la valeur de la fonction `additionalInfo` est de type `EmployeeInfo` qui a la fonction `salary`, vous pouvez accéder à cette fonction en utilisant `additionalInfo/EmployeeInfo:salary`. Dans cet exemple, le chemin de fonction `additionalInfo/salary` génère une erreur car `salary` n'a pas été défini pour le type `uima.cas.TOP`.

Les fonctions qui ont des valeurs de type tableau ou liste ont les propriétés supplémentaires suivants :

- Utilisez des crochets ([<number>]) pour sélectionner un certain élément dans le tableau ou dans la liste. Un tableau commence à zéro (0). Par exemple, pour sélectionner le premier élément du tableau des entreprises, utilisez companies[0]. Le marqueur spécial [last] peut permettre de sélectionner la dernière entrée d'un tableau, quelle que soit sa taille, par exemple companies[last].
- Utilisez des crochets vides ([]) pour obtenir tous les éléments. Un seul crochet vide ([]) est admis dans un chemin de fonctions. Par exemple, dans un tableau de suspects, le chemin de fonctions `knownSuspects[]/com.ibm.omnifind.types.Suspect:surName` rassemble tous les noms des suspects dans un tableau String.
- Lorsqu'un chemin de fonctions qui renvoie un tableau est utilisé lors de l'indexation, les éléments de tableau sont concaténés (séparés par des espaces) et placés dans l'index en tant qu'attribut unique ou comportant plusieurs termes ou que zone.
- Vous devez entrer l'élément suivant du chemin de fonctions. Le nom de type est le type d'éléments du tableau. Par exemple, utilisons une structure de fonctions de type `Info`. Ce type a une fonction nommée `companies`, dont la plage est un élément FSArray. Les éléments du tableau sont de type `Company`. `Company` a une fonction nommée `profit`. Pour obtenir le bénéfice de la troisième entreprise, entrez (à l'aide des noms de type complets) `companies[3]/Company:profit`.

Fonctions intégrées

Les fonctions intégrées sont des noms de fonction prédéfinies avec des sémantiques particulières. Elles peuvent permettre d'accéder aux informations qui ne sont pas contenues dans la structure de fonctions elles-mêmes, par exemple, le type de la structure de fonctions ou le texte couvert d'une annotation. Vous pouvez les utiliser dans un chemin de fonctions en tant que dernier ou seul élément.

Les fonctions intégrées suivantes peuvent être utilisées dans les deux fichiers de configuration de mappage :

- `fsId()` renvoie l'ID de la structure de fonctions. L'ID renvoyé est un entier (32 bits). Utilisez cette fonction intégrée pour accéder aux parties d'un document qui correspondent exactement à la requête.
- `typeName()` renvoie le type d'objet de structure d'analyse commune sous la forme de chaîne Le type correspond au nom de type complet incluant des préfixes d'espace de nom, par exemple `uima.tcas.Annotation`. Dans un contexte de base de données, `typeName()` est particulièrement utile lorsque vous stockez des types et des sous-types dans la même colonne et que vous souhaitez connaître un type réel d'une annotation ou d'une structure de fonctions. L'exemple suivant stocke le type de personne, tel que *suspect* ou *witness*, dans la colonne des rôles.

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

- `coveredText()` renvoie le texte couvert par l'objet d'analyse commune. `coveredText()` est disponible uniquement pour les annotations et leurs

sous-types. N'utilisez pas cette fonction intégrée sur les structures de fonction qui ne sont pas incluses dans une classification par le type d'annotation. L'exemple suivant stocke le nom d'un suspect dans la colonne suspectName.

```
<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>
```

- [] renvoie un descripteur de l'entrée de conteneur en cours (tableau ou liste). La fonction implique une itération, ce qui signifie qu'une entrée est créée dans l'index ou la table de base de données pour chaque élément dans le tableau ou la liste. L'exemple suivant est extrait d'un fichier de configuration JDBC dans lequel la fonction intégrée [:index] est admise.

```
<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>
```

Les fonctions intégrées suivantes peuvent être utilisées uniquement dans le fichier de configuration de mappage JDBC :

- `uniqueId()` renvoie l'ID unique global de la structure de fonctions. L'ID unique renvoyé est une chaîne de longueur fixe (27 caractères) et une concaténation du résultat de `fsId()`, `docId()`, `docTimestamp()` et du nombre de segments car les documents peuvent être morcelés en plusieurs structures d'analyse commune dans la recherche d'entreprise.

  La chaîne renvoyée peut inclure des caractères compris dans les plages ″a-z″ et ″A-Z″, les nombres ″0-9″, le point-virgule (″;″) et le caractère deux points (″:″).

  Le résultat de l'élément `uniqueId()` peut être utilisé en tant que clé principale pour les tables.

- `objectId()` renvoie l'ID de l'annotation ou de la structure de fonctions. `objectId()` est similaire à `uniqueId()` mais il ne contient pas le résultat de `docTimestamp()`. L'ID renvoyé est unique uniquement dans une collection dans laquelle les documents sont analysés une seule fois. Si tous les documents et versions de document doivent être uniques, vous devez utiliser `uniqueId()`.

  La chaîne renvoyée de la fonction intégrée `objectId()` a une longueur fixe de 16 caractères et peut inclure des caractères compris dans les plages ″a-z″ et ″A-Z″, les nombres ″0-9″, le point-virgule (″;″) et le caractère deux points (″:″).

Si uniqueId() ou objectId() référence des structures de fonctions vides, la valeur par défaut définie dans la définition de table de base de données est utilisée, aucun objet vide d'un type référencé n'est stocké.

- docId() renvoie l'ID document. La valeur renvoyée est de type integer (32 bits).

  L'exemple suivant affiche ces fonctions intégrées :

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

- docUri() renvoie l'URI du document.
- docTimestamp() renvoie l'heure (en millisecondes) à laquelle le document a été traité. Cette fonction intégrée est utile pour le suivi des versions de document, par exemple, si vous souhaiter savoir si la version de document que vous utiliser est la dernière à avoir été transmise par le moteur de balayage.

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

- parentId() renvoie l'élément fsId() de la structure de fonctions qui comporte un mappage de conteneur. parentId() est valide uniquement dans le contexte d'un mappage de conteneur.
- uniqueParentId() renvoie l'élément uniqueId() de l'annotation ou de la structure de fonctions comprise dans un mappage de conteneur. La fonction intégrée est valide uniquement dans le contexte d'un mappage de conteneur.
- [:index] renvoie l'index de l'entrée de conteneur en cours (tableau ou liste).

Tâches associées

«Extraction des parties d'un document qui correspondent à une requête de recherche sémantique», à la page 42
Vous pouvez extraire uniquement les parties d'un document qui correspondent exactement à la requête en mappant les structures de fonctions appropriées à l'index et à la base de données et en indiquant l'étendue dans la requête de recherche sémantique.

Filtres

Les filtres permettent de restreindre les règles de mappage dans les fichiers de configuration JDBC et d'index. Les résultats de l'analyse sont ajoutés à l'index ou à une table JDBC uniquement si le filtre est appliqué.

L'élément `<filter>` est facultatif et il permet de restreindre les mappages aux fonctions qui ont une certaine valeur d'attribut. Cet élément est utile si vous voulez qu'un attribut se comporte comme commutateur pour les éléments à indexer ou à ajouter à la base de données. Par exemple, les personnes et les entreprises peuvent être enregistrées dans une annotation de type `EntityAnnotation`. Sa fonction appelée `type` a la valeur `person` ou `organization`. Pour extraire uniquement les personnes et non les entreprises, vous pouvez ajouter le filtre suivant à la règle de mappage :

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

Chaque expression de filtre a la forme :

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

où :

- FeaturePath est un chemin de fonction dans la structure d'analyse commune
- L'opérateur est =, !=, <, <=, > ou >=. < (et uniquement <) doit être exprimé sous la forme de `&lt;`.
- Le littéral est un entier, un nombre à virgule flottante (aucun syntaxe d'exposant n'est prise en charge) ou un un littéral de chaîne placé entre guillemets.

`<FeaturePath>`, `<Operator>` et `<Literal>` doivent être séparés par un espace.

Les exemples suivants sont des filtres valides :

- `<filter syntax="FeatureValue"> foo = "hello world" </filter>`

  La fonction foo contient la chaîne hello world.
- `<filter syntax="FeatureValue"> foo &lt; 42 </filter>`

  La fonction foo contient la valeur d'entier 42.
- `<filter syntax="FeatureValue"> make/company = "Chevrolet" </filter>`

  Le chemin de fonction make/company dans lesquel la fonction make contient une structure de fonctions qui a une fonction company avec la valeur Chevrolet.
- `<filter syntax="FeatureValue"> bar7 >= 0.5 </filter>`

  La fonction bar7 contient la valeur flottante 0.5.

Mappage d'index pour les résultats de l'analyse personnalisée

Après avoir exécuté l'analyse personnalisée sur une collection de documents, vous pouvez utiliser le moteur de recherche dans la recherche d'entreprise afin de générer un index à partir des informations stockées dans la structure d'analyse commune créée par les algorithmes d'analyse personnalisée.

Le mappage des résultats de l'analyse à des zones, des étendues de texte et des attributs dans l'index de recherche d'entreprise permet d'utiliser ces informations dans les requêtes. L'association de l'analyse personnalisée à la recherche d'entreprise pouvant indexer à la fois des mots et des étendues de texte, active la recherche d'entreprise.

A l'aide d'un fichier de configuration de génération d'index, vous pouvez déterminer les résultats de l'analyse de la structure d'analyse commune à indexer.

Vous pouvez utiliser différents styles pour mapper des structures de fonction de la structure d'analyse commune à l'index de recherche d'analyse.

Annotation
Si vous indexez des structures de fonction dans la structure d'analyse commune à l'aide du style d'annotation, toutes les annotations des types spécifiés sont stockées dans l'index en tant qu'étendues pouvant être recherchées.

Par exemple, si une structure de fonctions qui étend une certaine zone de texte est de type `person` et qu'elle est indexée à l'aide du style d'annotation, les requêtes suivantes sont possibles :

Tableau 1. Requêtes exemple

Informations requises	**Requête possible**
Extraire tous les documents qui contiennent au moins un nom de personne	<person/>
Extraire tous les documents dans lequel un supérieur hiérarchique est indiqué dans une annotation de personne	<person>boss</person>
Extraire tous les documents dans lesquels la mention de langue est indiquée dans la même phrase qu'un de mes concurrents	<sentence><person>Lang</person> <competitor/></sentence>

Les attributs des structures de fonctions sont également indexés comme partie de l'étendue. Prenons l'exemple d'un annotateur qui détecte les voitures et stocke la marque de voiture en tant que fonction `make` de l'annotation `car`. Permet d'activer le type suivant de requête : ″Extraire les documents qui mentionnent les voitures de marque Chevrolet″.

Zone Utilisez ce style si vous souhaitez que le contenu des structures de fonctions soient accessibles lors de la recherche en utilisant les fonctions de recherche de zone de la recherche d'entreprise. De cette manière, le contenu d'une structure de fonctions peut être affiché dans les résultats de la recherche ou utilisé dans la recherche paramétrique.

Par exemple, si vous mappez des dosages de médicaments à une zone paramétrique, vous pouvez utiliser la requête suivante : ″Extraire tous les documents mentionnant un médicament pris à un dosage supérieur à 100 milligrammes″.

Interruption
Utilisez ce style si vous souhaitez qu'une structure de fonctions particulière soit interprété en tant que délimiteur, par exemple des sections ou des paragraphes. La recherche d'entreprise détecte les phrases et les paragraphes par défaut. Utilisez ce style uniquement si l'analyse personnalisée détecte des éléments structurels supplémentaires dans un document que vous souhaitez interpréter différemment.

Vous pouvez également avoir recours aux résultats de l'analyse pour influencer le classement des documents dans la recherche d'entreprise, même pour des requêtes de mot clé simples. Cette action s'effectue en deux étapes :

1. Mappage des structures de fonction à des étendues ou à des zones pouvant être recherchées, à l'aide du style de mappage d'annotation ou de zone.
2. Définition d'une classe à l'aide de la console d'administration de recherche d'entreprise et mappage du nom de zone ou d'étendue à cette classe de pondération.

Si l'utilisateur entre un terme de recherche contenu dans la structure de fonctions, le classement du document est plus élevé. Prenons l'exemple d'un annotateur qui identifie les personnes et les noms d'entreprise. En mappant ces structures de fonctions à des étendues (″person″ et ″company″, par exemple) puis en mappant ces étendues aux classes de pondération, le résultat de recherche ″gap″ classe les documents parlant de l'entreprise ″Gap″ à un niveau plus élevé que ceux contenant à peine le terme ″gap″.

Une fois que vous avez créé le fichier de configuration de génération d'index, vous pouvez le télécharger dans la recherche d'entreprise à l'aide de la console d'administration.

Tâches associées

«Création du fichier de configuration de génération d'index»
A l'aide d'un fichier de configuration de génération d'index, vous pouvez déterminer les résultats de l'analyse de la structure d'analyse commune à indexer afin d'activer la recherche.

Création du fichier de configuration de génération d'index

A l'aide d'un fichier de configuration de génération d'index, vous pouvez déterminer les résultats de l'analyse de la structure d'analyse commune à indexer afin d'activer la recherche.

A propos de cette tâche

Le fichier de configuration de génération d'index doit être compatible avec le schéma décrit dans l'exemple suivant. Le fichier de configuration exemple se fonde sur le système type défini pour le scénario de rapport de police.

```
<?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>
```

Restrictions

Le fichier de configuration de mappage d'index doit contenir l'ensemble des résultats de l'analyse que vous souhaitez pouvoir rechercher dans les requêtes.

Procédure

Pour créer un fichier de configuration de mappage d'index, procédez comme suit :

1. Créez un fichier XML. Pour éviter des erreurs de syntaxe XML, utilisez un éditeur XML ou un outil de création XML de votre choix. Le schéma XSD du fichier de configuration est appelé CasToIndexMapping.xsd et se trouve dans l'installation de recherche d'entreprise à l'emplacement *RACINE_INSTALL_RECHERCHE_ENTREPRISE*/packages/uima/.
2. Incluez vos mappages dans un élément `<indexBuildSpecification xmlns="http://www.ibm.com/of/822/consumer/index/xml">`. L'espace de nom (indiqué dans l'attribut xmlns) doit être exactement identique à celui affiché.
3. Ajoutez un élément `<skipCondition>` pour empêcher que certains documents soient indexés, en fonction d'une certaine valeur de fonction. Cet élément est facultatif. Dans l'exemple, les documents contenant une structure de données de type `com.ibm.uima.tt.DocumentAnnotation` avec une fonction nommée `toBeProcessed` ayant la valeur zéro ne seront pas indexés.
4. Ajoutez un ou plusieurs éléments `<indexBuildItem>` qui contiennent le mappage d'une structure de fonctions particulière dans la structure d'analyse commune à une structure de l'index.
5. Sauvegardez et validez le fichier XML.

Elément <indexBuildItem>

Le fichier de configuration de spécification de génération d'index contient un ou plusieurs éléments `<indexBuildItem>`. Chaque élément décrit le mappage d'une structure de fonctions particulière dans la structure d'analyse commune à une structure de l'index (une étendue ou une zone).

L'élément `<name>` contient le type de structure de fonctions. Il existe deux méthodes permettant de définir un type :

- Nom de type complet. Par exemple, `com.ibm.omnifind.types.Suspect`
- Caractère générique. Par exemple, `com.ibm.omnifind.types.*`. Le caractère générique peut être ajouté uniquement à la fin de la spécification de type.

Utilisez uniquement des sous-types `uima.tcas.Annotation` en tant qu'éléments de génération d'index. Si une fonction est un sous-type `uima.cas.TOP` (et non `uima.tcas.Annotation`), vous pouvez accéder à cette structure de fonctions à l'aide d'un chemin de fonctions commençant à partir d'une annotation.

Si le type A est un sous-type du type B (dans l'exemple, `com.ibm.omnifind.types.Suspect` en tant que sous-type de

com.ibm.omnifind.types.Person) et qu'il existe des éléments <indexBuildItem> Ia et Ib définis pour les deux types, le traitement s'effectue comme suit :

- Chaque règle d'index définie dans Ib est appliquée aux structures de fonctions de type B et aux structures de traits de type A
- Chaque règle d'index définie dans Ia est appliquée aux structures de fonctions de type A

Dans l'exemple, l'élément <indexBuildItem> défini pour les annotations com.ibm.omnifind.types.Person s'applique également aux annotations com.ibm.omnifind.types.Suspect. Deux étendues sont créées pour une annotation suspect : Person et Suspect.

L'élément <filter> est facultatif et il permet de restreindre le mappage <indexBuildItem> aux structures de fonctions qui ont une certaine valeur d'attribut. Cet élément est utile si vous voulez qu'un attribut se comporte comme commutateur pour les éléments à indexer. Par exemple, les personnes et les entreprises peuvent être enregistrées dans une annotation de type EntityAnnotation. Sa fonction appelée type a la valeur person ou organization. Pour extraire uniquement les personnes et non les entreprises, vous pouvez ajouter le filtre suivant :

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

Vous pouvez également choisir d'indexer les personnes et les entreprises en utilisant des noms d'étendue différents, par exemple person et organization. Pour cela, définissez deux éléments <indexBuildItem> de type EntityAnnotation et utilisez deux filtres sur la fonction type pour déclencher les personnes ou les entreprises.

Elément<indexRule>

Chaque élément <indexBuildItem> contient un élément <indexRule>. Chaque élément <indexRule> contient l'ensemble des informations requises pour le mappage d'une structure de fonctions se trouvant dans la structure d'analyse commune à l'index en tant que zone, annotation ou style d'interruption. Les styles de zone et d'annotation prennent en charge plusieurs attributs. Vous ne pouvez pas utiliser le terme style, qui est pris en charge dans le SDK UIMA pour la recherche d'entreprise (le terme style est ignoré).

Pour les styles d'annotation et de zone, il existe les alternatives suivantes lorsque vous indiquez le nom d'annotation ou de zone dans l'index :

- Utilisez fixedName si vous souhaitez que chaque structure de fonctions soit accessible dans l'index portant le même nom. Dans l'exemple suivant, chaque structure de fonctions de type Person est mappée à une étendue nommée ″Person″ dans l'index.

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

  Active des requêtes telles ″Extraire des documents dans lequel un supérieur hiérarchique est inclus en tant que nom de personne″. La requête est exprimée de la manière suivante à l'aide de fragments XML :
  `@xmlf2::'<Person>Boss</Person>'`

- Utilisez nameFeature si l'annotation stocke des entités différentes auxquelles vous souhaitez pouvoir accéder à l'aide de différentes étendues en fonction de la valeur d'une certaine fonction de l'annotation. Dans l'exemple suivant, EntityAnnotation est indexé en tant qu'étendue person ou organization, en fonction de la valeur nommée type. La fonction peut également être un chemin de fonctions.

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

  Active des requêtes, telles "Extraire des documents sur l'entreprise WHO" (en opposition au terme anglais "who"). La requête est exprimée de la manière suivante dans la syntaxe XPath limitée :
  `@xmlp::'/organization[ftcontains="WHO"]'`

- Si aucun des attributs ci-dessus n'est utilisé, le nom abrégé du type d'annotation de l'élément <indexBuildItem> est utilisé. Il s'agit de la valeur par défaut. Par exemple :

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

  Cet élément <indexBuildItem> génère des annotations et des zones appelées RoomNumber chargées avec le texte couvert par com.ibm.uima.tutorial.RoomNumber.

Elément <style name="Annotation" />

L'annotation de l'élément <style> indique comment accéder aux informations d'étendue dans la recherche d'entreprise. Outre l'utilisation des attributs fixedName et nameFeature, ce style prend également en charge l'élément <attributemappings>. Dans cet élément, il est possible de mapper la valeur d'une fonction à un attribut de l'étendue en résultant dans l'index, que vous pouvez ensuite utiliser dans une expression de recherche.

Chaque mappage est effectué dans un élément <mapping> séparé. L'élément <feature> contient un chemin de fonction et l'élément <indexName> contient le nom de l'attribut utilisé dans l'index pour stocker la valeur de <feature>. Par exemple,

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

Cet élément <mapping> stocke la valeur de la fonction du chemin make/companyname directement dans l'attribut d'index company.

Le mappage des valeurs de fonction aux attributs d'index est particulièrement utile si le système type utilisé lors de l'analyse de texte est complexe, y compris un grand nombre de structures de fonctions imbriquées. A l'aide de l'élément

<mapping>, des attributs appropriés peuvent être exposés, ce qui vous permet de les utiliser dans des requêtes sans bien connaître la structure du système type d'origine.

Elément <style name="Field" />

La zone de l'élément <style> indique comment accéder aux informations de zone dans la recherche d'entreprise. Outre les attributs fixedName et nameFeature, vous pouvez définir les attributs suivants.

parametric
Si la valeur de la zone est true, elle peut être recherchée de manière paramétrique, par exemple, #dosage:>100

fieldSearchable
Si la valeur de la zone est true, elle peut être utilisée dans la recherche, par exemple, make:Bayer

returnable
Si la valeur de la zone est true, la zone et ses valeurs sont renvoyées dans le résultat de la recherche

Il est toujours possible de rechercher le contenu des informations de zone. Autrement dit, les informations de zone sont accessibles dans les recherches de mot clé normales.

L'attribut facultatif valueFeature définit la valeur de fonction à utiliser en tant que valeur de zone. Si la structure de fonctions est une annotation et que l'attribut n'est pas défini, le texte couvert de l'annotation est utilisé en tant que valeur de zone. Dans l'exemple,

```
<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>
```

deux zones sont générées pour com.ibm.omnifind.types.Date. Une zone nommée date contient le texte couvert, par exemple, 17:15. Une autre zone contient la valeur de l'attribut hour. Vous pouvez effectuer une requête ici de la manière suivante : 'hour::<17'.

Elément <style name="Breaking" />

La valeur Breaking de l'élément <style> n'inclut pas d'autres éléments.

Une fois que vous avez créé le fichier XML, vous devez le télécharger dans la recherche d'entreprise et sélectionner le fichier de configuration de mappage

d'index ainsi que d'autres options d'analyse personnalisée dans la console d'administration de recherche d'entreprise.

Concepts associés

«Mappage d'index pour les résultats de l'analyse personnalisée», à la page 24
Après avoir exécuté l'analyse personnalisée sur une collection de documents, vous pouvez utiliser le moteur de recherche dans la recherche d'entreprise afin de générer un index à partir des informations stockées dans la structure d'analyse commune créée par les algorithmes d'analyse personnalisée.

«Chemins de fonction», à la page 20
Un chemin de fonctions permet d'accéder à des valeurs de fonctions dans les structures d'analyse commune, de la même manière que les instructions XPath permettant d'accéder aux éléments XML d'un document XML.

Référence associée

«Filtres», à la page 24
Les filtres permettent de restreindre les règles de mappage dans les fichiers de configuration JDBC et d'index. Les résultats de l'analyse sont ajoutés à l'index ou à une table JDBC uniquement si le filtre est appliqué.

«Description du système type», à la page 9
Décrit les structures de fonctions (structures de données sous-jacentes qui représentent les résultats de l'analyse) utilisées dans l'analyse personnalisée.

Mappage de base de données pour les résultats de l'analyse sélectionnés

Une fois que vous avez exécuté l'analyse personnalisée sur une collection de documents dans la recherche d'entreprise, vous pouvez stocker les résultats de l'analyse de texte sélectionnés dans une base de données compatible JDBC.

Cette version prend en charge uniquement DB2 Universal Database, Version 8.2.2 (com.ibm.db2.jcc.DB2Driver Version 2.3) et Oracle 10g (oracle.jdbc.driver.OracleDriver Version 1.0).

Pour DB2 Universal Database et Oracle, vous pouvez choisir d'insérer les résultats de l'analyse directement dans la base de données ou de générer les fichiers de chargement propres à la base de données équivalents et le script correspondant qui exécute les commandes de chargement.

Le mappage des résultats de l'analyse à des tables d'une base de données vous permet d'utiliser ces informations dans les étapes de traitement de veille économique ou d'accéder directement aux parties appropriées d'un document qui correspondent à une requête de recherche sémantique.

Un fichier de configuration de mappage XML contient des informations de configuration de connexion aux bases de données et décrit quels résultats de l'analyse personnalisée doivent être stockés dans quelles tables et colonnes. Les noms des tables et des colonnes du fichier de configuration doivent correspondre aux tables et aux colonnes créées dans la base de données.

Une fois que vous avez créé le fichier de configuration, vous pouvez télécharger le fichier dans la recherche d'entreprise à l'aide de la console d'administration.

Tâches associées

«Création du fichier de configuration de mappage XML», à la page 33
Pour ajouter les résultats de l'analyse à une base de données, vous devez créer

un fichier de configuration. Ce dernier contient les informations concernant la configuration de connexion à la base de données et une description des résultats de l'analyse de texte personnalisée à stocker dans des tables et colonnes spécifiques.

Stockage des résultats de l'analyse dans une base de données

Pour stocker les résultats de l'analyse sélectionnés dans une base de données compatible JDBC, vous devez créer un fichier de configuration pour la recherche d'entreprise, et les bibliothèques du pilote JDBC nécessaires doivent se trouver dans le chemin défini dans le fichier de configuration.

Pour stocker les résultats de l'analyse dans une base de données compatible JDBC, procédez comme suit :

1. Déterminez les résultats de l'analyse à stocker dans la base de données. Créez une base de données qui contient les tables avec toutes les colonnes nécessaires des types de données appropriés.

   Important : Créez votre propre base de données DB2 pour stocker les résultats de l'analyse sélectionnés. N'utilisez pas la base de données DB2 inclus dans l'installation de la recherche d'entreprise.
2. Dans un éditeur XML, placez les données de configuration de la base de données et les résultats de l'analyse à stocker dans le fichier de configuration. Pour déterminer les résultats de l'analyse à inclure dans le fichier de configuration, vous devez connaître le système type sous-jacent utilisé par l'analyse personnalisée.
3. Placez les bibliothèques du pilote JDBC dans un répertoire accessible à partir du noeud de l'index du système de recherche d'entreprise.
4. Téléchargez et sélectionnez le fichier de configuration avec l'analyse de texte personnalisée en utilisant la console d'administration de recherche d'entreprise.

Création du fichier de configuration de mappage XML

Pour ajouter les résultats de l'analyse à une base de données, vous devez créer un fichier de configuration. Ce dernier contient les informations concernant la configuration de connexion à la base de données et une description des résultats de l'analyse de texte personnalisée à stocker dans des tables et colonnes spécifiques.

A propos de cette tâche

Le fichier de configuration de mappage XML doit être compatible avec le schéma présenté dans l'exemple suivant. L'exemple se fonde sur le système type défini pour le scénario de rapport de police.

Dans l'exemple, seuls les rapports de polices et les villes sont ajoutés à la base de données. L'exemple présente l'utilisation de fonctions intégrées et du mappage de l'élément<constant>.

```
<?xml version="1.0" encoding="UTF-8"?>
<cas2JdbcConfiguration xmlns="http://www.ibm.com/uima/consumer/jdbc/100/xml">
   <databaseConnection>
     <connectionUrl>db2://myMachine:myPort/myDatabase</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>myUser</username>
      <password>myPassword</password>
    </authentication>

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

  </databaseConnection>

  <jdbcMappingSpec>
    <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>
   </jdbcMappingSpec>
</cas2JdbcConfiguration>
```

Restrictions

Créez votre propre base de données DB2 pour stocker les résultats de l'analyse sélectionnés. N'utilisez pas la base de données DB2 intégrée à l'installation de recherche d'entreprise.

Procédure

Pour créer un fichier de configuration de base de données XML, procédez comme suit :

1. Créez un fichier XML. Pour éviter des erreurs de syntaxe XML, utilisez un éditeur XML ou un outil de création XML de votre choix. Le schéma XSD du fichier de configuration est appelé CasToJDBCMapping.xsd et se trouve dans l'installation de recherche d'entreprise à l'emplacement *RACINE_INSTALL_RECHERCHE_ENTREPRISE*/packages/uima/.
2. Incluez vos mappages dans un élément `<cas2JdbcConfiguration xmlns="http://www.ibm.com/uima/consumer/jdbc/100/xml">`. L'espace de nom (indiqué dans l'attribut xmlns) doit être exactement identique à celui affiché.
3. Ajoutez un élément `<databaseConnection>` qui contient toutes les informations de configuration de connexion à une base de données et un élément `<jdbcMappingSpec>` qui décrit les règles de mappage pour les résultats de l'analyse qui sont stockés dans la base de données ou dans des fichiers de chargement.
4. Ajoutez les éléments de composant suivants à l'élément `<databaseConnection>` :
   - Obligatoire : Un élément `<connectionUrl>`. Cet élément contient l'URL de connexion à la base de données. En fonction de l'implémentation du pilote JDBC, vous pouvez utiliser un accès distant ou local à la base de données.
   - Obligatoire : Un élément `<driver>`. Cet élément contient le nom de la classe du pilote JDBC, par exemple `com.ibm.db2.jcc.DB2Driver` pour DB2, ou `oracle.jdbc.driver.OracleDriver` pour Oracle.
   - Obligatoire : Un élément `<driverLibraries>`. Cet élément permet d'établir une liste des bibliothèques du pilote. Chaque bibliothèque est répertoriée dans un élément `<driverLibrary>`. Les bibliothèques se trouvent dans le répertoire d'installation DB2 ou Oracle. Pour DB2, les bibliothèques sont `c:\your_db2_dir\db2jcc.jar`, `c:\your_db2_dir\db2jcc_license_cu.jar` et `c:\your_db2_dir\db2jcc_license_cisuz.jar`. Pour Oracle, la bibliothèque à inclure est `c:\votre_rép_oracle\classes12.zip`.
   - Obligatoire : Un élément `<authentication>`. Cet élément contient le nom d'utilisateur et le mot de passe pour la base de données.
   - Facultatif : Un élément `<loadFile>`. Cet élément contient le répertoire de fichiers de chargement dans un élément `<loadFileDirectory>` et le nom du script de chargement dans un élément `<loadScript>`. Si vous n'indiquez pas d'élément `<loadFile>`, toutes les données sont stockées directement dans la base de données à l'aide de JDBC.

     Vous devez également ajouter l'ensemble des paramètres de configuration de base de données lorsque vous utilisez des fichiers de chargement et des scripts propres à la base de données.
5. Ajoutez les éléments de composant suivants à l'élément `<jdbcMappingSpec>` :
   - Facultatif : Un élément `<skipCondition>`. Si aucune condition skip n'est définie, tous les documents sont traités.

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

Dans l'exemple, les documents qui contiennent une annotation de type com.ibm.uima.tt.DocumentAnnotation avec une fonction nommée toBeProcessed dont la valeur est égale à zéro ne sont pas pris en compte.

- Un élément <cas2JdbcMappings> qui indique les types et les fonctions mappés à des tables et à des colonnes de base de données spécifiques. L'élément contient une section de mappages explicites et implicites.

6. Ajoutez un élément <explicitMappings>. Cet élément est obligatoire. Il doit avoir un ou plusieurs éléments <explicitMappingRule> qui définissent les mappages explicites et peut être défini uniquement pour les types d'annotation et leurs sous-types. Si un mappage est défini dans une section de mappages explicites, toutes les annotations qui correspondent à la définition de mappage sont stockées dans la base de données.
7. Facultatif : Ajoutez un élément <implicitMappings>. Cet élément prend en charge tous les types de structure de fonctions. Si cet élément existe, il doit contenir au moins un élément <implicitMappingRule>. Les mappages définis dans la section des mappages implicites sont ajoutés à la base de données uniquement si les types d'annotation sont référencés par une autre annotation qui respecte une règle de mappage implicite ou explicite.

   Le but du mappage implicite est de permettre le stockage uniquement des résultats de l'analyse qui apparaissent dans un contexte particulier. Par exemple, si le mappage d'une annotation de type com.ibm.omnifind.types.City est implicite, seules les villes référencées par la définition de mappage com.ibm.omnifind.types.PoliceReport de la section de mappages explicites sont stockées dans la base de données. Autrement dit, seules les villes mentionnées dans des rapports de police sont ajoutées à la base de données.

   S'il existe une règle de mappage explicite pour l'annotation City, toutes les villes sont ajoutées à la base de données. Dans tous les cas, si une ville est référencée par plusieurs rapports de police, elle est ajoutée une seule fois à la base de données.
8. Les éléments <explicitMappingRule> et <implicitMappingRule> doivent contenir l'attribut applyToSubtypes qui lorsqu'il a la valeur true stocke non seulement la structure de la fonction indiquée dans l'élément <type> mais également toutes les structures qui en sont dérivées. Ajoutez les éléments de composant suivants aux éléments <explicitMappingRule> et <implicitMappingRule> :
   - Un élément <type> qui contient le type de structure de fonctions.
   - Un élément <table> qui contient le schéma de base de données et le nom de la table. La syntaxe suit la règle schema.table_name, ou uniquement table_name si aucun schéma n'est défini.
   - Elément <featureMappings> avec un ou plusieurs éléments <featureMapping> ou un élément <contanerMapping>.
   - Facultatif : Un élément <filter> qui contient une condition évaluée dès concordance de la règle de mappage. Si la condition a la valeur true, l'annotation ou la structure de fonctions est stockée dans la base de données. Dans cet exemple, seuls les rapports de police concernant des crimes commis à Los Angeles seront stockés dans la base de données.
9. La structure de composant de l'élément <featureMapping> diffère selon que vous mappez une fonction ou une constante.

   Si vous mappez une fonction ou un chemin de fonction, les éléments du composant incluent :

- Un élément <feature> avec le nom de la fonction. La fonction doit être définie pour la structure de la fonction dans l'élément type. Vous pouvez également utiliser une construction de chemin de fonction ou une des fonctions intégrées du système.
- Facultatif : Un élément <length> avec la longueur d'une chaîne admise dans la colonne de base de données indiquée. Les chaînes plus longues sont tronquées.
- Un élément <column> avec le nom de la colonne dans laquelle la valeur de la fonction doit être stockée. Les colonnes de base de données qui ne sont pas utilisées dans des mappages de fonction utilisent une valeur par défaut (généralement, la valeur null) qui est configurée dans la base de données.

  Assurez-vous que la valeur de l'élément feature est stockée dans une colonne du type approprié. Le tableau ci-dessous indique quels types UIMA correspondent à quels types de base de données.

Tableau 2. Mappage entre les types UIMA et les types de base de données correspondants

Type UIMA ou fonction intégrée	Type de données DB2 recommandé	Type de données Oracle recommandé
Float	REAL	FLOAT
String	VARCHAR	VARCHAR2
Integer	INTEGER	INTEGER
uniqueId(), uniqueParentId()	CHAR(27)	CHAR(27)
objectId(), parentId()	CHAR(16)	CHAR(16)
docTimestamp()	BIGINT	LONG

Pour une constante, les éléments de mappage de fonction de composant sont les suivants :

- Un élément <constant> qui contient la valeur d'une constante.
- Un élément <column> avec le nom de la colonne dans lequel la valeur de la constante est ajoutée.

10. L'élément <containerMapping> contient le mappage d'une fonction de type de conteneur (tableau ou liste). Cet élément doit être utilisé uniquement pour les types de conteneur. Il comporte les éléments de composant suivants :
    - Un élément <feature> avec le nom de la fonction. Vous pouvez également utiliser une construction de chemin de fonction ou une des fonctions intégrées du système.
    - Un élément <table> qui contient le schéma de base de données et le nom de la table. La syntaxe suit la règle schema.table_name, ou uniquement table_name si aucun schéma n'est défini.
    - Un ou plusieurs éléments <featureMapping> qui contiennent le nom des structures de fonction et les noms des colonnes dans lesquelles les fonctions sont ajoutées.
11. Sauvegardez et validez le fichier XML à l'aide du schéma fourni.

Une fois que vous avez créé le fichier XML, vous devez le télécharger dans la recherche d'entreprise et sélectionner le fichier de configuration de mappage de base de données ainsi que d'autres options d'analyse personnalisée dans la console d'administration de la recherche d'entreprise.

Concepts associés

«Mappage de base de données pour les résultats de l'analyse sélectionnés», à la page 32

Une fois que vous avez exécuté l'analyse personnalisée sur une collection de documents dans la recherche d'entreprise, vous pouvez stocker les résultats de l'analyse de texte sélectionnés dans une base de données compatible JDBC.

«Chemins de fonction», à la page 20
Un chemin de fonctions permet d'accéder à des valeurs de fonctions dans les structures d'analyse commune, de la même manière que les instructions XPath permettant d'accéder aux éléments XML d'un document XML.

Référence associée

«Filtres», à la page 24
Les filtres permettent de restreindre les règles de mappage dans les fichiers de configuration JDBC et d'index. Les résultats de l'analyse sont ajoutés à l'index ou à une table JDBC uniquement si le filtre est appliqué.

«Fonctions intégrées», à la page 21
Les fonctions intégrées sont des noms de fonction prédéfinies avec des sémantiques particulières. Elles peuvent permettre d'accéder aux informations qui ne sont pas contenues dans la structure de fonctions elles-mêmes, par exemple, le type de la structure de fonctions ou le texte couvert d'une annotation. Vous pouvez les utiliser dans un chemin de fonctions en tant que dernier ou seul élément.

«Description du système type», à la page 9
Décrit les structures de fonctions (structures de données sous-jacentes qui représentent les résultats de l'analyse) utilisées dans l'analyse personnalisée.

Mappage de type de conteneur

Un type de conteneur est un des types de liste ou de tableau intégré de la structure d'analyse commune. Le mappage de type de conteneur permet de mapper des valeurs de liste ou de tableau à une base de données relationnelle.

Vous disposez de deux approches pour la gestion des types de conteneur dans le fichier de configuration. Une méthode utilise des constructions de fonctions intégrées et une table de liens génériques incluant des tableaux ou des listes qui constituent les valeurs d'une règle de mappage de fonction. Etant donnée que différents tableaux ou listes sont stockés dans la même table de liens, cette dernière ne comporte aucune informations sur la relation des informations stockées.

Dans la deuxième méthode, la définition de table de liens générée à l'aide de l'élément `<containerMapping>` indique explicitement la relation entre les informations indiquées requises.

Un exemple de mappage de table de liens générique est disponible ci-dessous. Il existe une relation n:m entre les rapports de police et les suspects. Ce qui signifie qu'un suspect peut être mentionné dans plusieurs rapports de police et qu'un rapport de police peut mentionner plusieurs suspects.

La table `sample.fsarray` générique de l'exemple constitue un lien entre les rapports de police et les suspects. S'il existe un type de mappage autre que `com.ibm.omnifind.types.PoliceReport` qui comporte une fonction de type `com.ibm.omnifind.types.FSArray`, il est également mappé à cette table. Vous pouvez toujours interroger la table pour connaître la relation entre un rapport de police et un suspect. Toutefois, vous ne pouvez pas conclure en consultant uniquement la table que cette dernière contient la relation ou le lien entre les rapports de police et les suspects possibles.

```
<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>
    </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>
```

L'élément suivant affiche les tables de base de données créées à l'aide des règles de mappage génériques ci-dessus.

Tableau 3. Table sample.policeReport

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

Tableau 4. Table sample.fsarray

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

Tableau 5. Table sample.suspect

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

L'exemple affiche le mappage des tableaux de structures de fonctions. Vous pouvez également appliquer ce type de mappage à StringArray, IntegerArray et FloatArray. Si vous incluez des règles de mappage pour ces tableaux de valeurs simples, remplacez `[]/uniqueId()` par `[]`.

L'approche de table générique peut être utilisée pour les listes de structures de fonctions ainsi que pour les listes de types simples (StringList, IntegerList et FloatList).

Une méthode plus simple de gestion des relations consiste à utiliser un élément de mappage de conteneur explicite qui définit l'itération des éléments se trouvant dans les tableaux ou les listes.

Un exemple de mappage qui décrit une table de liens explicites apparaît ci-dessous. Il existe à nouveau une relation n:m entre les rapports de police et les suspects. Toutefois, ici la table `sample.reports_suspects` constitue la table de liens entre les rapports de police et les suspects.

En utilisant cette approche, vous n'avez pas à prendre en compte les ID de tableau ou les mappages d'entrée de début et de fin pour les types de liste. La table de liens contient une relation explicite.

```
<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>
```

Un élément <containerMapping> permet de définir les itérations des éléments contenus dans le tableau. Dans l'exemple, la table de liens sample.reports_suspects contient un lien vers les colonnes policeReportId et suspectId. N'imbrique pas les éléments <containerMapping>.

L'élément suivant affiche les tables de base de données créées à l'aide de règles de mappage de tables de liens explicites.

Tableau 6. Table sample.policeReport

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

Tableau 7. Table sample.reports_suspect

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

Tableau 8. Table sample.suspect

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

Référence associée

«Fonctions intégrées», à la page 21
Les fonctions intégrées sont des noms de fonction prédéfinies avec des sémantiques particulières. Elles peuvent permettre d'accéder aux informations qui ne sont pas contenues dans la structure de fonctions elles-mêmes, par exemple, le type de la structure de fonctions ou le texte couvert d'une annotation. Vous pouvez les utiliser dans un chemin de fonctions en tant que dernier ou seul élément.

Extraction des parties d'un document qui correspondent à une requête de recherche sémantique

Vous pouvez extraire uniquement les parties d'un document qui correspondent exactement à la requête en mappant les structures de fonctions appropriées à l'index et à la base de données et en indiquant l'étendue dans la requête de recherche sémantique.

Accès à toutes les instances d'un type d'annotation spécifique dans le résultat de la recherche. Par exemple, pour obtenir toutes les personnes, incluez un mappage de style pour le type d'annotation et indiquez qu'elle peut être renvoyée dans le fichier de configuration d'index. Par exemple :

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

Dans cet exemple, les annotations de type `com.ibm.omnifind.types.Person` sont mappées à une étendue nommée Person dans l'index de recherche d'entreprise. Vous pouvez utiliser ce dernier pour accéder aux annotations lors de la recherche sémantique. De plus, le texte couvert des annotations (des noms de personnes, par exemple) est stocké sous la forme de zone pouvant être envoyée. Pour extraire ces valeurs d'annotation, appelez getFields(″Person″) sur chaque objet de résultat renvoyé de la requête de recherche (mot clé ou sémantique). Cette méthode renvoie un tableau de chaînes avec les valeurs d'annotation. Dans le cas présent, les noms des personnes.

Toutefois, cette approche renvoie toutes les instances d'une annotation donnée et n'est pas appropriée si vous souhaitez limiter le traitement du résultat aux documents qui correspondent exactement à la requête. Par exemple, un document peut mentionner cinq personnes. Toutefois, dans la requête de recherche sémantique '`<sentence><person/>IBM</sentence>`, l'utilisateur est intéressé uniquement par la personne mentionnée dans la même phrase que celle dans laquelle apparaît dans le terme IBM. Les autres personnes ne l'intéressent pas.

Pour accéder et traiter les structures de fonctions qui correspondent exactement à la requête, procédez comme suit :

1. Mappez les types de structure de fonctions appropriés à l'index de recherche d'entreprise à l'aide du style de mappage d'annotation. Par exemple :

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

2. Mappez les types de structure de fonctions appropriés aux tables JDBC. Au cours du processus de mappage, vous devez inclure deux colonnes, une pour l'URI de document et l'autre pour l'ID de structure de fonctions. Bien que vous puissiez mapper l'ensemble des types de structure de fonctions à la même table de base données, vous devez mapper chaque type à une table différente. Par exemple :

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

3. Parcourez, analysez et indexez les documents.
4. Extrayez les ID des instances correspondant à la requête. Dans l'interface SIAPI (search and index API), ces éléments sont appelés éléments cibles. Un élément cible indique l'étendue cible à renvoyer. Il est défini de la manière suivante :
   - Dans les fragments XML, l'élément cible est identifié par un signe numéro (#). Ce signe est autorisé uniquement une fois et peut apparaître à tout emplacement de la requête de fragment XML. Par exemple : `$xmlf2::'<sentence><#person/>IBM</sentence>'`
   - Dans XPath par défaut, l'élément cible est la dernière zone de l'expression XPath.
   - Accédez à ces instances en utilisant la méthode Result.getProperty("TargetElement"). La propriété renvoyée est une concaténation de chaînes de tous les ID d'occurrence séparés par des espaces. Chaque occurrence de la propriété peut être convertie en une valeur entière.

5. SIAPI ne renvoie pas les structures de fonctions elles-mêmes, uniquement leurs ID d'occurrence. Ces ID correspondent à la valeur fsId() stockée dans la table de base de données. Pour extraire ces instances et les informations associées, votre application doit :
   a. sélectionner la table de base de données correcte, en fonction du nom d'étendue de l'élément cible. Dans l'exemple, l'application contient un mappage d'une personne à la table sample.Person. Ces informations sont déduites des fichiers de configuration pour le mappage d'index, qui génère le nom de l'étendue et pour le mappage JDBC, qui génère le nom de la table.
   b. Pour chaque résultat de la recherche :
      1) Analysez la chaîne renvoyée par Result.getProperty ("TargetElement") pour trouver les ID d'occurrence.
      2) Emettez une instruction SELECT pour la table en utilisant l'URI de résultat (accessible à l'aide de Result.getDocumentId()) en tant que valeur de la colonne docUri et les ID d'occurrence en tant que valeur de colonne annotationId. Les noms de colonne dépendent du fichier de configuration. Les noms de colonne utilisés proviennent de l'exemple précédent.

      Les lignes renvoyées contiennent les informations stockées pour la structure de fonctions, par exemple, le texte couvert ou les attributs spécifiques de la structure de fonctions, telle que "last name" ou "city of birth."

Vérifiez que les mises à jour apportées à la base de données sont synchronisées aux mises à jour d'index dans la recherche d'entreprise. Si la base de données contient des informations obsolètes (par exemple, vous avez utilisé des fichiers de chargement de base de données et vous n'avez pas mis à jour cette dernière mais vous avez régénéré ou réorganisé l'index), certains ID d'occurrence peuvent ne pas se trouver dans la base de données. La recherche d'entreprise conserve un enregistrement de la dernière version du document dans son index. C'est pourquoi, les ID d'occurrence sont valides uniquement pour le dernier document.

Si vous stockez plusieurs versions du même document dans la même table de base de données, plusieurs lignes peuvent correspondre aux mêmes ID d'occurrence, pour les différentes versions du document. Vous devez alors définir une colonne de version de document et la charger en utilisant la logique de l'application ou les fonctions intégrées, telles docTimestamp(). Ainsi, vous pouvez filtrer le résultat afin d'obtenir uniquement la dernière version du document.

Concepts associés

«Terme de requête de recherche sémantique», à la page 52
Le terme de requête de recherche sémantique est communiqué sous la forme d'un terme opaque.

Tâches associées

«Création du fichier de configuration de génération d'index», à la page 26
A l'aide d'un fichier de configuration de génération d'index, vous pouvez déterminer les résultats de l'analyse de la structure d'analyse commune à indexer afin d'activer la recherche.

«Création du fichier de configuration de mappage XML», à la page 33
Pour ajouter les résultats de l'analyse à une base de données, vous devez créer un fichier de configuration. Ce dernier contient les informations concernant la configuration de connexion à la base de données et une description des résultats de l'analyse de texte personnalisée à stocker dans des tables et colonnes spécifiques.

Types et fonctions définis dans la recherche d'entreprise

Le système type défini dans la recherche d'entreprise couvre la gestion des métadonnées de document et l'analyse linguistique de base.

L'analyse linguistique de base sous la forme de reconnaissance de la langue du document et de segmentation est toujours effectuée lors de l'indexation d'un document, que l'analyse personnalisée soit ou non sélectionnée. Lors de l'analyse de document de base, les informations suivantes sont ajoutées à la structure d'analyse de base que vous pouvez utiliser dans l'analyse personnalisée :

- Métadonnées de document de type `com.ibm.es.tt.DocumentMetaData`
- Annotations de marqueur sémantique, de phrase et de paragraphe du type `uima.tt.TokenAnnotation`, `uima.tt.SentenceAnnotation` et `uima.tt.ParagraphAnnotation`. L'annotation de marqueur sémantique inclut la fonction de `lemme`.

Le système type défini dans la recherche d'entreprise n'inclut aucune fonction ou aucun type sophistiqué propre à l'analyse de texte. Ces éléments sont inclus dans le système type UIMA que vous pouvez utiliser et développer lors de la définition des fonctions et types d'analyse personnalisée dans votre environnement UIMA. Il est fort possible que vous n'ayez pas besoin de développer le système type de recherche d'entreprise.

Le système type de recherche d'entreprise n'est pas défini dans le SDK UIMA. Si vous souhaitez utiliser un de ces types lors de la création d'un annotateur dans l'architecture UIMA, par exemple, si vous souhaitez accéder aux informations de sécurité du document ou au type de moteur de balayage ou au type de document, vous devez définir une nouvelle fois les types dans la description du système type du moteur d'analyse.

Les fonctions et types suivants sont définis dans la recherche d'entreprise :

uima.tcas.Annotation
Une annotation est composée des types suivants :

uima.tcas.DocumentAnnotation
L'annotation de document a la fonction suivante :

esDocumentMetaData
Contient des métadonnées de document du type `com.ibm.es.tt.DocumentMetaData`

com.ibm.es.tt.ContentField
L'annotation de zone de contenu a la fonction suivante :

parameters
Les paramètres de zone de contenu sont de type `com.ibm.es.tt.CommonFieldParameters`.

com.ibm.es.tt.Anchor
Annotation d'ancrage pour le texte d'ancrage dans les documents HTML. Elle dispose de la fonction suivante :

uri URI cible du texte d'ancrage. La valeur de la fonction est de type `uima.cas.String`.

com.ibm.es.tt.MarkupTag

Annotations d'informations de marquage, par exemple, d'une balise XML. Les informations de marquage sont stockées dans les fonctions suivantes :

name Nom de la balise de marquage. La valeur de la fonction est de type `uima.cas.String`.

depth Profondeur d'imbrication. La valeur de la fonction est de type `uima.cas.Integer`.

attributeName

Nom de l'attribut de la fonction. La valeur de la fonction est de type `uima.cas.StringArray`.

attributeValues

Chaîne de valeurs pour l'attribut. La valeur de la fonction est de type `uima.cas.StringArray`.

uima.CAS.TOP

Elément principal du système type. Il dispose des types suivants :

com.ibm.es.tt.DocumentMetaData

Les métadonnées de document ont les fonctions suivantes. Les fonctions sont connectées à la fonction d'annotation de document `esDocumentMetaData`.

crawlerId

Nom du moteur de balayage. La valeur de la fonction est de type `uima.cas.String`.

dataSource

Un des types de source de données suivants :

- Web (pour les documents provenant du moteur de balayage Web)
- NNTP (pour les documents provenant du moteur de balayage de forum)
- DB2 (pour les documents provenant du moteur de balayage DB2)
- Notes (pour les documents provenant du moteur de balayage Notes
- CM (pour les documents provenant du moteur de balayage Content Management)
- FS (pour les documents provenant du moteur de balayage du système de fichiers UNIX)
- WinFS (pour les documents provenant du moteur de balayage du système de fichiers Windows)
- Exchange (pour les documents provenant du moteur de balayage Exchange)
- VBR (pour les documents provenant du moteur de balayage VeniceBridge)

La valeur de la fonction est de type `uima.cas.String`.

dataSourceName

Nom du moteur de balayage (source de données). La valeur de la fonction est de type `uima.cas.String`.

docType
Un des types de document suivants :

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

La valeur de la fonction est de type `uima.cas.String`.

securityTokens
Jetons de sécurité du document. La valeur de la fonction est de type `uima.cas.StringArray`.

date Date du document. La valeur de la fonction est de type `uima.cas.String`.

baseUri
URI de base de la page. La valeur de la fonction est de type `uima.cas.String`.

metaDataFields
La valeur de la fonction est de type `uima.cas.FSArray`. Chaque élément de ce tableau est de type `com.ibm.es.tt.MetaDataField`.

redirectUrl
URL redirigée. La valeur de la fonction est de type `uima.cas.String`.

mimeType
Type Mime, ou type de document (XML, par exemple). La valeur de la fonction est de type `uima.cas.String`.

url URL du document. La valeur de la fonction est de type `uima.cas.String`.

com.ibm.es.tt.CommonFieldParameters
Les paramètres de zone commune incluent :

searchable
Indicateur définissant s'il est possible d'effectuer une recherche sans texte dans la zone.

fieldSearchable
Indicateur définissant s'il est possible d'effectuer une recherche dans la zone.

parametric
Indicateur de recherche paramétrique.

showInSearchResult
Indicateur définissant si les données annotées sont incluses dans les détails des résultats de la recherche.

resolveConflict
Indicateur permettant de résoudre les conflits de métadonnées entre `MetadataPreferred`, `ContentPreferred` et `Coexist`. La valeur de la fonction est de type `uima.cas.String`.

name Nom de la zone. Vous pouvez rechercher cette zone en indiquant le nom de la zone. La valeur de la fonction est de type `uima.cas.String`.

com.ibm.es.tt.MetaDataField
Les données de la zone de métadonnées ne font pas partie du contenu du document mais elles sont stockées dans la fonction ″text″ :

parameters
Paramètres de la zone de métadonnées de type `com.ibm.es.tt.CommonFieldParameters`.

text Le texte des métadonnées est stocké dans cette fonction de type `uima.cas.String`.

Référence associée

«Types et fonctions définis dans l'architecture UIMA»
Le SDK UIMA définit des types linguistiques de base et des fonctions pouvant être reconnues dans un document lors de l'analyse du texte.

Types et fonctions définis dans l'architecture UIMA

Le SDK UIMA définit des types linguistiques de base et des fonctions pouvant être reconnues dans un document lors de l'analyse du texte.

Chaque moteur d'analyse disposant de ses propres descriptions de système type décrit les éléments d'entrée requis et les types de sortie pour les annotateurs du moteur d'analyse. Les descriptions de système type sont propres au domaine et à l'application.

Vous pouvez développer le système type UIMA afin d'inclure vos propres types et fonctions. Dans l'environnement UIMA, il existe un module d'extension Eclipse qui vous guide dans le processus de modification des descripteurs de système type pour les annotateurs. Pour obtenir plus de détails sur l'installation et l'utilisation du module d'extension Component Descriptor Editor, consultez la documentation UIMA.

Lorsque vous avez fini de développer et de tester votre moteur d'analyse dans l'environnement UIMA, le fichier d'archive (fichier .pear) créé qui contient les fichiers du moteur d'analyse inclut également la description du système type.

Les types et fonctions suivants sont définis dans l'architecture UIMA :

uima.tcas.Annotation
Une annotation comprend les types suivants :

uima.tcas.DocumentAnnotation

uima.tt.TTAnnotation

uima.tcas.DocumentAnnotation
Une annotation de document inclut les fonctions suivantes :

categories
Liste de noms de catégorie ou libellés du document. La valeur de la fonction est de type `uima.cas.FSList`.

languageCandidates
Liste de références de la langue du document. La valeur de la fonction est de type `uima.cas.FSList`.

id Forme d'identification de document, telle une URL. La valeur de la fonction est de type `uima.cas.String`.

uima.tt.TTAnnotation
Une annotation TT inclut les types suivants :

uima.tt.DocStructureAnnotation
Informations structurelles sur le document. L'annotation de structure de document inclut les types suivants :

uima.tt.SentenceAnnotation
Phrase qui inclut des signes de ponctuation d'ouverture et de fermeture. Inclut la fonction :

sentenceNumber
Numéro de séquence de la phrase dans le paragraphe. Attribuez à nouveau la valeur 1 au début de chaque paragraphe. La valeur de la fonction est de type `uima.cas.Integer`.

uima.tt.ParagraphAnnotation
Paragraphe. Ses fonctions incluent :

paragraphNumber
Numéro de séquence du paragraphe. La valeur de la fonction est de type `uima.cas.Integer`.

uima.tt.LexicalAnnotation
Informations de contenu sur le document. Une annotation lexicale inclut les types suivants :

uima.tt.CompPartAnnotation
Partie d'un mot composé. Les mots composés d'un grand nombre de langues germaniques sont des mots sans espace. Par exemple, le mot allemand ″Abteilungsleiter″ (chef de service) est composé des termes ″Abteilung″ (service) et ″Leiter″ (chef).

uima.tt.TokenAnnotation
Marque sémantique sans espace l'entourant. Ses fonctions incluent :

lemmaEntries
Liste de tous les lemmes pour une marque sémantique données. Chaque entrée est une entrée de dictionnaire possible pour la marque sémantique.

lemma
Lemme de la liste de tous les lemmes possibles pour une marque sémantique dans `lemmaEntries`. Ce lemme est utilisé lors de la recherche.

tokenNumber
Numéro de séquence de marque sémantique dans la séquence. Attribuez à nouveau la valeur 1 au début de chaque phrase. La valeur de la fonction est de type `uima.cas.Integer`.

tokenProperties
Propriété de marque sémantique, par exemple, correspond à la marque sémantique en majuscules ou sous forme numérique. La valeur de la fonction est de type `uima.cas.Integer`.

stopwordToken
Marque sémantique définie comme mot vide. La valeur de la fonction est de type `uima.cas.Integer`.

synonymEntries
Liste de références des entrées de type `uima.tt.Synonym`. Chaque entrée est une entrée de synonyme possible pour la marque sémantique.

normalizedCoveredText
Représentation normalisée du texte couvert par l'annotation. La valeur de la fonction est de type `uima.cas.String`.

uima.CAS.TOP
Elément principal du système type. Il dispose des types suivants :

uima.tt.KeyStringEntry
Chaîne avec la fonction suivante :

key Chaîne.

uima.tt.Lemma
Entrée de dictionnaire avec les informations morphologiques suivantes :

partOfSpeech
Codage intégral de la classe de mots du lemme.

morphID
Codage intégral des informations morphologiques.

uima.tt.Synonym
Entrée de synonyme pour un mot donné de type `uima.tt.keyStringEntry`.

uima.tt.LanguageConfidencePair
Type avec les fonctions suivantes qui décrit la sélection de la langue du document.

uima.tt.LanguageConfidencePair

languageConfidence
Indication (valeur flottante comprise entre 0 et 1) de l'adaptation de la langue choisie à la langue du document.

language
Langue du document (valeur ISO). La valeur est de type `uima.cas.String`.

languageID
ID de langue. La valeur est de type `uima.cas.Integer`.

uima.tt.CategoryConfidencePair
Type avec les fonctions suivantes qui décrit la sélection de catégorie pour le document.

uima.tt.CategoryConfidencePair
Une catégorie a les valeurs suivantes :

categoryString
Nom de la catégorie. La valeur est de type `uima.cas.String`.

categoryConfidence
Indication de l'adaptation de la catégorie au document. La valeur est de type float.

mostSpecific
Indicateur (de type `uima.cas.Integer`) définissant si la catégorie est la plus appropriée au document.

taxonomy
Nom de la taxinomie à laquelle appartient la catégorie. Les documents peuvent avoir des catégories provenant de différentes taxinomies. La valeur est de type `uima.cas.String`.

Référence associée

«Types et fonctions définis dans la recherche d'entreprise», à la page 45
Le système type défini dans la recherche d'entreprise couvre la gestion des métadonnées de document et l'analyse linguistique de base.

Applications de recherche sémantique

Quatre types d'informations de document sont stockés dans l'index de recherche d'entreprise que vous pouvez interroger dans les applications de recherche à l'aide de l'interface SIAPI.

Les quatre différents types d'informations incluent :

- Des mots qui se trouvent dans un document, par exemple une chaîne telle *logiciel*.
- Des noms d'étendue, par exemple, un document XML qui inclut <author>James</author> génère l'étendue <author>.
- Des noms d'attribut, par exemple, un document XML qui inclut <author countryOfBirth=USA>James</author> génère l'attribut "countryOfBirth".
- Des valeurs d'attribut, par exemple USA est la valeur de l'attribut "countryOfBirth."

La langue de la requête SIAPI inclut le terme de la requête de recherche sémantique. Le terme spécifie un motif de brindille. Une brindille est un petit arbre avec des feuilles. Chaque feuille représente les quatre types d'informations (mots, noms d'étendue, etc). Les modes internes de l'arborescence indiquent comment les occurrences d'un document sont liées les unes aux autres. Il existe cinq types de noeuds internes qui définissent des relations :

- et
- ou

- non
- dans_étendue_de
- attribut_dans_portée_de

Un document satisfait une recherche sémantique donnée s'il inclut les occurrences des feuilles et que les contraintes définies par les noeuds internes (relations définies) sont respectées.

La requête de recherche sémantique vous aide à extraire des documents de meilleure qualité. Maintenant, vous pouvez non seulement effectuer une recherche en utilisant des combinaisons booléennes de mots et d'annotations mais vous pouvez également extraire des document dans lesquels, par exemple *James* apparaît dans l'étendue nommée author ou dans lesquels les termes *ibm* et *search* apparaissent dans la même phrase.

Terme de requête de recherche sémantique

Le terme de requête de recherche sémantique est communiqué sous la forme d'un terme opaque.

Il existe deux formes de syntaxe permettant d'exprimer un terme opaque dans l'interface SIAPI :

- Fragments XML
- XPath limité

Le terme de requête de fragment XML a l'aspect d'un fragment équilibré d'un document XML. Un terme de requête de fragment XML est préfixé par le signe de terme opaque `@xmlf2::` suivi de l'expression de fragment XML incluse entre apostrophes ('...').

Toutefois, les termes de requête XPath limités sont préfixés par `@xmlxp::` suivi de la requête XPath incluse entre apostrophes ('...').

De la même manière qu'avec des termes de requête généraux de l'interface SIAPI, chaque terme peut avoir un modificateur d'apparence :

Signe plus (+)
Le terme doit apparaître.

Préfixe =
Le terme doit être une correspondance exacte.

Caractère tilde (~) en préfixe
Prise en compte du terme de la requête.

Caractère tilde (~) en suffixe
Prise en compte des mots qui ont le même lemme que le terme de la requête.

Les exemples suivants affichent des requêtes de fragment XML.

@xmlf2::'<City>Springfield</City>'
Recherche des documents qui incluent l'étendue (annotation) contenant la chaîne Springfield.

@xmlf2::'<Person gender="female">'
Recherche des documents dans lesquels une personne de sexe féminin est annotée.

**@xmlf2::'<Person><.or><@gender>female</@gender>
<@title>Mrs</@title><@title>Ms</@title></.or></Person>'**

Recherche des documents qui définissent une personne en tant que femme à l'aide du sexe ou de la civilité.

**@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>'**

Recherche des documents qui indiquent des individus de sexe masculin considérés comme suspects et une annotation policeReport qui contient les chaînes *robbery* et *theft* dans crimeDescription mais pas la chaîne *accident*. Les documents doivent également contenir une annotation city avec les quartiers *Brynston* et *Brooklyn*.

Les requêtes XPath correspondantes ont les structures suivantes :

@xmlxp::'//City[ftcontains(Springfield)]'

Recherche des documents qui incluent l'étendue (annotation) city contenant la chaîne *Springfield*.

@xmlxp::'//Person[@gender="female"]'

Recherche des documents dans lesquels une personne de sexe féminin est annotée.

**@xmlxp::'//Person[@gender="female" or @title ftcontains(Ms) or @title
ftcontains(Mrs)]'**

Recherche des documents qui définissent une personne en tant que femme à l'aide du sexe ou de la civilité.

**@xmlxp::'//Person[@gender="male" and @role="suspect"] //PoliceReport
[@crimeDescription ftcontains(robbery) or @crimeDescription ftcontains(theft)]
//City [(@district="Brynston" or @district="Brooklyn") and
ftcontains(Springfield)]'**

Recherche des documents qui indiquent des individus de sexe masculin considérés comme suspects et une annotation policeReport qui contient les chaînes robbery et theft dans crimeDescription. Les documents doivent également contenir une annotation city avec les quartiers *Brynston* et *Brooklyn*.

Prise en charge des synonymes dans les applications de recherche

Les utilisateurs peuvent étendre les résultats de la recherche en recherchant des documents contenant des synonymes de termes de la requête.

Les synonymes incluent généralement des termes incluant plusieurs mots, des noms par produit par exemple, tels *WebSphere Information Integrator OmniFind*. Les termes incluant plusieurs mots contenus dans le dictionnaire de synonymes sont correctement identifiés dans les requêtes utilisateur et ne sont pas placés entre guillemets.

L'API SIAPI (Search and Index API) pour la recherche d'entreprise prend en charge plusieurs méthodes de recherche de synonymes des termes de la requête :

- La syntaxe des requêtes SIAPI prend en charge l'opérateur tilde (~) pour le développement des synonymes. Si l'utilisateur ajoute cet opérateur à un terme de la requête, le développement de synonymes est effectué pour ce mot. Par exemple, la requête `~WAS` renvoie des documents qui comportent WebSphere Application Server et tout autre synonyme de cette abréviation.
- Le développement de synonymes peut être activé à l'aide de l'interface d'expansion de synonymes SIAPI à partir d'une application de recherche. Les termes de la requêtes peuvent être automatiquement développés afin d'inclure des synonymes ou l'application de recherche peut inclure des options qui permettent à l'utilisateur de déterminer si les synonymes des termes de la requête doivent être renvoyés dans les résultats de la recherche.

  Lors du développement automatique de synonymes, la recherche de synonymes est effectuée sur tous les mots de la requête et zones de contenu. Les résultats de la recherche incluent des documents qui contiennent, soit les termes de la requête, soit des synonymes des termes de la requête. Les résultats de la recherche indiquent également quels termes correspondent à quels synonymes.

  Dans un scénario géré par l'utilisateur, l'application de recherche indique à l'utilisateur les synonymes qui ont été trouvés pour chaque mot de la requête avant que cette dernière ne soit effectuée. L'utilisateur sélectionne alors les termes à inclure dans la recherche ou reformule la recherche afin de supprimer les termes de la requête d'origine. Dans ce scénario, l'utilisateur contrôle les termes à inclure dans la requête, soit les équivalences strictes, soit les variantes du sens et de l'utilisation des mots.

Création d'un fichier XML pour les synonymes

Pour développer des requêtes dans la recherche d'entreprise afin d'inclure les synonymes des termes de la requête, vous devez indiquer quels mots sont synonymes de quels autres dans un fichier XML.

A propos de cette tâche

Le fichier XML qui répertorie les synonymes doivent se conformer au schéma présenté dans l'exemple suivant :

```
<?xmlversion="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>
```

Restrictions

Vous devez grouper des mots qui sont synonymes les uns des autres (éléments <synonym>) dans un élément <synonymgroup>. Un synonyme peut inclure des espaces mais ne peut pas inclure de caractères de ponctuation, tels une virgule (,) ou une barre verticale (|), car ces caractères peuvent entrer en conflit avec la syntaxe de requête de recherche d'entreprise.

Vous devez énumérer toutes les déclinaisons possibles des termes ajoutés en tant que synonymes, telles la forme au singulier et au pluriel d'un mot. Il n'est pas nécessaire d'énumérer les normalisations du terme, comme la suppression des accents ou des trémas allemands (Umlaut). La recherche d'entreprise gère automatiquement la normalisation. Par exemple, si vous souhaitez inclure le terme météo en tant que synonyme, il n'est pas nécessaire d'inclure également le terme METEO.

Procédure

Pour créer une liste de synonymes pour la recherche d'entreprise, procédez comme suit :

1. Créez un fichier XML. Pour éviter des erreurs de syntaxe XML, utilisez un éditeur XML ou un outil de création XML de votre choix.
2. Ajoutez un élément <synonymgroup> puis un élément <synonym> pour chaque mot devant être traité comme un synonyme d'autres mots dans le groupe de synonymes.

   Vérifiez que vos mappages sont inclus dans un élément <synonymgroups xmlns="http://www.ibm.com/of/822/synonym/xml">. L'espace de nom (indiqué dans l'attribut xmlns) doit être exactement identique à celui affiché.
3. Répétez la procédure jusqu'à ce que vous ayez spécifié tous les synonymes à utiliser pour la recherche de documents dans une collection de recherche d'entreprise.
4. Sauvegardez et quittez le fichier XML.

Une fois que vous avez créé le fichier XML, vous devez le convertir en un dictionnaire de synonymes afin qu'il puisse être ajouté au système de recherche d'entreprise.

Création d'un dictionnaire de synonymes

Une fois que vous avez créé ou mis à jour une liste de synonymes dans un fichier XML, vous devez convertir ce dernier en un dictionnaire de synonymes.

A propos de cette tâche

Pour créer un dictionnaire de synonymes, utilisez l'outil de ligne de commande appelé essyndictbuilder, fourni avec WebSphere II OmniFind Edition. Il se se trouve dans le répertoire *RACINE_INSTALL_RECHERCHE_ENTREPRISE*/bin.

L'outil traite un fichier XML qui répertorie les synonymes et génère un dictionnaire de synonymes. Le dictionnaire doit avoir le suffixe `.dic`. Par exemple, `c:\mydictionaries\products.dic`.

L'emplacement par défaut des fichiers est le répertoire à partir duquel le script est appelé. S'il existe un dictionnaire portant le même nom, le script génère une erreur.

Procédure

Pour créer un dictionnaire de synonymes pour la recherche d'entreprise, procédez comme suit :

1. Sur le serveur d'index, connectez-vous en tant qu'administrateur de recherche d'entreprise. Cet ID utilisateur a été indiqué lors de l'installation de WebSphere II OmniFind Edition.
2. Entrez la commande suivante, où *fichier_XML* correspond au chemin complet du fichier XML qui contient la liste des synonymes et *fichier_DIC* au chemin complet du dictionnaire de synonymes.

   AIX Linux ou Solaris : `essyndictbuilder.sh` *fichier_XML fichier_DIC*
   Windows : `essyndictbuilder.bat` *fichier_XML fichier_DIC*

Une fois que vous avez créé un dictionnaire de synonymes, utilisez la console d'administration de recherche d'entreprise pour ajouter le dictionnaire au système de recherche d'entreprise et l'associer à une ou à plusieurs collections.

Seul le fichier .dic généré est téléchargé vers le système de recherche d'entreprise. Assurez-vous que le fichier XML source est stocké dans un environnement dont l'accès est contrôlé et effectuez une sauvegarde régulière du fichier. Ce fichier XML est requis pour la mise à jour du dictionnaire de synonymes.

Personnalisation des dictionnaires de mots vides

Les utilisateurs peuvent définir un vocabulaire propre à l'entreprise qui est retiré d'une requête afin d'augmenter la pertinence de la recherche.

Il existe deux types de prise en charge de mots vides dans la recherche d'entreprise :

- La reconnaissance de mots vides propres à une lange qui supprimer tous les mots fréquemment utilisés, tels *a* et *the* d'une requête comportant plusieurs mots. Le dictionnaire de mots vides qui existe pour chaque langue ne peut pas être modifié par les utilisateurs. Cette reconnaissance des mots vides est effectuée automatiquement pour toutes les requêtes afin d'améliorer la pertinence de la recherche.
- La reconnaissance des mots vides personnalisée ou définie par l'utilisateur qui supprime du vocabulaire propre à l'entreprise des requêtes. Ce dictionnaire de mots vides défini par l'administrateur peut contenir uniquement du vocabulaire spécial. Le dictionnaire de mots vides défini par l'utilisateur ne remplace les dictionnaires de mots vides propres à chaque langue de la recherche d'entreprise qui contiennent des mots communs.

Les mots vides définis par l'utilisateur incluent généralement des termes incluant plusieurs mots, des noms par produit par exemple, tels *WebSphere Information Integrator OmniFind*. Les termes incluant plusieurs mots contenus dans le dictionnaire de mots vides sont correctement identifiés dans les requêtes utilisateur et ne sont pas placés entre guillemets.

Les termes composés des langues germaniques sont également correctement identifiés dans les requêtes. Un terme composé est constitué de deux ou plusieurs mots utilisés dans un seul mot. Les termes composés lexicalisés, tels *Reisebüro* (agence de voyage) ne sont pas considérés comme étant des termes composés.

Les termes composés d'une requête sont fractionnés en termes individuels. Si un des termes qui compose le terme se trouve dans le dictionnaire de mots vides, le terme composé est alors supprimé de la requête.

Par exemple, le terme de requête `Versicherungspolice` (police d'assurance) renvoie des documents qui contiennent les termes composés *Lebensversicherungspolice* (police d'assurance vie) et *Haftpflichtversicherungspolice* (police d'assurance responsabilité civile). Le deuxième terme est également renvoyé pour une requête de type `Haftpflicht` (assurance de responsabilité civile). Même si le mot *Police* est répertorié dans le dictionnaire de mots vides, le terme composé de la requête `Versicherungspolice` n'est pas supprimé de la requête.

Vous devez dresser la liste du vocabulaire propre à l'entreprise dans un fichier XML que vous devez ensuite convertir en dictionnaire de mots vides afin qu'il puisse être ajouté au système de recherche d'entreprise.

Vous pouvez sélectionner dans la console d'administration de recherche d'entreprise le dictionnaire de mots vides à utiliser. Vous pouvez sélectionner un dictionnaire de mots vides pour chaque collection. Un dictionnaire de mots vides peut être partagé par plusieurs collections.

Création d'un fichier XML pour les mots vides

Pour retirer du vocabulaire propre à l'entrepris des requêtes, vous devez définir des mots vides dans un fichier XML.

A propos de cette tâche

Le fichier XML qui répertorie les mots vides doit se conformer au schéma présenté dans l'exemple suivant :

```
<?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>
```

Restrictions

Un mot vide peut inclure des espaces mais ne peut pas inclure de caractères de ponctuation, tels une virgule (,) ou une barre verticale (|) car ces caractères peuvent entrer en conflit avec la syntaxe de requête de recherche d'entreprise.

Il n'est pas nécessaire d'énumérer les normalisations du terme, comme la suppression des accents ou des trémas allemands (Umlaut). La recherche d'entreprise gère automatiquement la normalisation. Par exemple, si vous souhaitez inclure le terme météo en tant que mot vide, il n'est pas nécessaire d'inclure également le terme METEO.

Procédure

Pour créer une liste de mots vides pour la recherche d'entreprise, procédez comme suit :

1. Créez un fichier XML. Pour éviter des erreurs de syntaxe XML, utilisez un éditeur XML ou un outil de création XML pouvant valider les éléments XML.
2. Ajoutez un élément `<stopWord>` pour chaque mot devant être traité comme un mot vide.

   Vérifiez que vos mappages sont inclus dans un élément `<stopWords xmlns="http://www.ibm.com/of/83/stopwordbuilder/xml">`. L'espace de nom (indiqué dans l'attribut xmlns) doit être exactement identique à celui affiché.
3. Répétez la procédure précédente jusqu'à ce que vous ayez spécifié l'ensemble des mots vides à retirer des requêtes lorsque les utilisateurs effectuent des recherches dans des collections.
4. Sauvegardez et quittez le fichier XML.

Une fois que vous avez créé le fichier XML, vous devez le convertir en un dictionnaire de mots vides afin qu'il puisse être ajouté au système de recherche d'entreprise.

Création d'un dictionnaire de mots vides

Une fois que vous avez créé ou mis à jour une liste de mots vides dans un fichier XML, vous devez convertir ce dernier en un dictionnaire de mots vides.

A propos de cette tâche

Pour créer un dictionnaire de mots vides, utilisez l'outil de ligne de commande appelé esstopworddictbuilder, fourni avec WebSphere II OmniFind Edition. Il se trouve dans le répertoire *RACINE_INSTALL_RECHERCHE_ENTREPRISE*/bin.

L'outil traite un fichier XML qui répertorie les mots vides et génère un dictionnaire de mots vides. Le dictionnaire doit avoir le suffixe `.dic`. Par exemple, `c:\mydictionaries\productstopwords.dic`.

L'emplacement par défaut des fichiers est le répertoire à partir duquel le script est appelé. S'il existe un dictionnaire portant le même nom, le script génère une erreur.

Procédure

Pour créer un dictionnaire de mots vides pour la recherche d'entreprise, procédez comme suit :

1. Sur le serveur d'index, connectez-vous en tant qu'administrateur de recherche d'entreprise. Cet ID utilisateur a été indiqué lors de l'installation de WebSphere II OmniFind Edition.
2. Entrez la commande suivante, où *fichier_XML* correspond au chemin complet du fichier XML qui contient la liste des mots vides et *fichier_DIC* au chemin complet du dictionnaire de mots vides.

   AIX, Linux ou Solaris : `esstopworddictbuilder.sh` *fichier_XML fichier_DIC*
   Windows : `esstopworddictbuilder.bat` *fichier_XML fichier_DIC*

Une fois que vous avez créé un dictionnaire de mots vides, utilisez la console d'administration de recherche d'entreprise pour ajouter le dictionnaire au système de recherche d'entreprise et l'associer à une ou à plusieurs collections.

Seul le fichier .dic généré est téléchargé vers le système de recherche d'entreprise. Assurez-vous que le fichier XML source est stocké dans un environnement dont l'accès est contrôlé et effectuez une sauvegarde régulière du fichier. Ce fichier XML est requis pour la mise à jour du dictionnaire de mots vides.

Personnalisation des dictionnaires de mots avec degré de pondération

Les utilisateurs peuvent définir des termes spécifiques ou des termes comportant plusieurs mots qui augmentent ou réduisent la valeur de classement du document dans lequel apparaît le terme.

Chaque terme du dictionnaire de mots avec degré de pondération est associé à un facteur de pondération pouvant aller de -10 à +10. Un facteur de pondération élevé est attribué aux termes que vous souhaitez particulièrement voir dans les documents de résultat. Une valeur faible est attribuée aux termes que vous ne souhaitez pas inclure dans les résultats ou qui sont associés à des termes possédant un degré de pondération plus élevé. Les valeurs -1, 0 et 1 n'ont aucun effet de pondération.

Si un terme de requête répertorié dans le dictionnaire de mots avec degré de pondération avec un facteur de pondération particulier apparaît dans un document extrait, la valeur de classement du document est augmentée ou diminuée en fonction de la valeur de pondération. La valeur de pondération attribuée à un terme est relative car elle peut varier en fonction d'autres facteurs. Si la pondération B1 est attribuée au terme X et que la pondération B2 est attribuée au terme Y, alors pondération(X) >= pondération(Y).

Un mot avec degré de pondération inclut généralement des termes incluant plusieurs mots, des noms de produit par exemple, tels *WebSphere Information Integrator OmniFind*. Les termes incluant plusieurs mots contenus dans le dictionnaire de mots avec degré de pondération sont correctement identifiés dans les requêtes utilisateur et ne sont pas placés guillemets.

Les termes composés des langues germaniques sont également correctement identifiés dans les requêtes. Un terme composé est composé de deux ou de plusieurs mots utilisés comme un seul mot. Les termes composés lexicalisés, tels *Reisebüro* (agence de voyage) ne sont pas considérés comme étant des termes composés.

Les termes composés d'une requête sont fractionnés en termes individuels. Si des valeurs de pondération sont associées aux termes individuels d'un mot composé, les documents extraits sont classés bien que la valeur attribuée soit inférieure à la valeur que le terme aurait s'il ne faisait pas partie d'un terme composé. Ainsi, la portée de la recherche est élargie, ce qui permet de meilleurs résultats lorsque peu de documents contiennent le terme composé complet.

Par exemple, le terme de requête `Versicherungspolice` (police d'assurance) renvoie des documents qui contiennent les termes composés *Lebensversicherungspolice* (police d'assurance vie) et *Haftpflichtversicherungspolice* (police d'assurance responsabilité civile). Le dernier terme est également renvoyé pour une requête de type `Haftpflicht` (assurance responsabilité civile). Si le mot *Police* (police) existe dans le dictionnaire de mots avec degré de pondération, une valeur de pondération est attribuée au document contenant le terme de requête composé `Versicherungspolice`.

Vous devez dresser la liste des termes avec leur valeur de pondération dans un fichier XML que vous pouvez alors convertir en dictionnaire de mots avec degré de pondération, afin qu'il puisse être ajouté au système de recherche d'entreprise.

Vous pouvez avoir recours à la console d'administration de recherche d'entreprise pour sélectionner le dictionnaire de mots avec degré de pondération à utiliser. Vous pouvez sélectionner un dictionnaire de mots avec degré de pondération par collection. Un dictionnaire de mots avec degré de pondération peut être partagé par plusieurs collections.

Création d'un fichier XML pour les mots avec degré de pondération

Pour réduire ou augmenter l'importance de certains documents de résultat, vous devez indiquer les mots qui influencent le classement des documents dans un fichier XML.

A propos de cette tâche

Le fichier XML qui répertorie les mots avec degré de pondération doit se conformer au schéma présenté dans l'exemple suivant :

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

Restrictions

Vous devez grouper des termes qui partagent la même valeur de pondération dans un élément `<boostTermList>`. Une valeur de pondération peut survenir plusieurs fois, par exemple, si vous souhaitez trier les mots avec facteur de pondération par ordre alphabétique dans le fichier XML.

Un mot avec degré de pondération peut inclure des espaces mais ne peut pas inclure de caractères de ponctuation, tels une virgule (,) ou une barre verticale (|) car ces caractères peuvent entrer en conflit avec la syntaxe de requête de recherche d'entreprise.

Les termes avec degré de pondération ont généralement des variantes, tels des acronymes ou des abréviations. Vous pouvez énumérer toutes les variantes dans le dictionnaire de mots avec degré de pondération. Toutefois, si vous envisagez d'utiliser un dictionnaire de synonymes ainsi qu'un dictionnaire de mots avec degré de pondération et que vous avez déjà ajouté des termes et leurs variantes au dictionnaire de synonymes, il n'est pas nécessaire d'ajouter ces variantes à la liste des mots avec degré de pondération. A la place, il vous suffit d'attribuer la valeur true à l'attribut `useVariants` pour la variante que vous ajoutez au dictionnaire de mots avec degré de pondération. Toutes les variantes de ce terme répertoriées dans le dictionnaire de synonymes qui apparaissent dans un des documents extraits influencent la valeur de classement attribuée à ces documents.

Il n'est pas nécessaire d'énumérer les normalisations du terme, comme la suppression des accents ou des trémas allemands (Umlaut). La recherche d'entreprise gère automatiquement la normalisation. Par exemple, si vous souhaitez inclure le terme météo en tant que terme avec degré de pondération, il n'est pas nécessaire d'inclure également le terme METEO.

Procédure

Pour créer une liste de mots avec degré de pondération pour la recherche d'entreprise, procédez comme suit :

1. Créez un fichier XML. Pour éviter des erreurs de syntaxe XML, utilisez un éditeur XML ou un outil de création XML de votre choix.
2. Incluez vos mappages dans un élément `<boostTerms xmlns="http://www.ibm.com/of/83/boostbuilder/xml">`. L'espace de nom (indiqué dans l'attribut xmlns) doit être exactement identique à celui affiché.
3. Ajoutez un élément `<boostTermList>` pour regrouper tous les termes qui partagent la valeur de pondération indiquée.

   Les valeurs de pondération vont -10 à 10. Par exemple, `<boostTermList boost="-5">` ou `<boostTermList boost="5">`.

   L'importance des documents qui contiennent les termes indiqués est augmentée ou réduite en fonction de la valeur de pondération indiquée.
4. Ajoutez un élément `<term>` pour chaque terme qui utilise la valeur de pondération indiquée.

   Si vous souhaitez inclure des variantes d'un mot avec degré de pondération répertoriées dans le dictionnaire de synonymes, affectez la valeur true à l'attribut `useVariants` de l'élément `<term>`. La valeur par défaut est false. Si aucune variante n'est disponible dans le dictionnaire de synonymes, aucun message d'erreur n'est généré.
5. Répétez la procédure précédente jusqu'à ce que vous ayez indiqué l'ensemble des termes utilisés comme mots avec degré de pondération lorsque les utilisateurs effectuent des recherches dans les collections de recherche d'entreprise.
6. Sauvegardez et quittez le fichier XML.

Une fois que vous avez créé le fichier XML, vous devez le convertir en un dictionnaire de mots avec degré de pondération afin qu'il puisse être ajouté au système de recherche d'entreprise.

Création d'un dictionnaire de mots avec degré de pondération

Une fois que vous avez créé ou mis à jour une liste de mots avec degré de pondération dans un fichier XML, vous devez convertir ce dernier en un dictionnaire de mots avec degré de pondération.

A propos de cette tâche

Pour créer un dictionnaire de mots avec degré de pondération, utilisez l'outil de commande appelé esboostworddictbuilder, fourni avec WebSphere II OmniFind Edition. Il se trouve dans le répertoire RACINE_INSTALL_RECHERCHE_ENTREPRISE/bin .

L'outil traite un fichier XML qui répertorie les mots avec degré de pondération et génère un dictionnaire de mots avec degré de pondération. Le dictionnaire doit avoir le suffixe `.dic`. Par exemple, `c:\mydictionaries\productboostwords.dic`.

L'emplacement par défaut des fichiers est le répertoire à partir duquel le script est appelé. S'il existe un dictionnaire portant le même nom, le script génère une erreur.

Procédure

Pour créer un dictionnaire de mots avec degré de pondération pour la recherche d'entreprise, procédez comme suit :

1. Sur le serveur d'index, connectez-vous en tant qu'administrateur de recherche d'entreprise. Cet ID utilisateur a été défini lors de l'installation de WebSphere II OmniFind Edition.
2. Entrez la commande suivante, où *fichier_XML* correspond au chemin complet du fichier XML qui contient la liste des mots avec degré de pondération et *fichier_DIC*au chemin complet du dictionnaire de mots avec degré de pondération. Si vous souhaitez également utiliser un dictionnaire de synonymes, ajoutez le chemin complet de ce dernier après le nom du dictionnaire de mots avec degré de pondération. L'attribution d'un nom au dictionnaire de synonymes est facultative.

   UNIX : `esboostworddictbuilder.sh` *fichier_XML fichier_DIC fichier_SYNDIC*
   Windows : `esboostworddictbuilder.bat` *fichier_XML fichier_DIC fichier_SYNDIC*

Une fois que vous avez créé un dictionnaire de mots avec degré de pondération, utilisez la console d'administration de recherche d'entreprise pour ajouter le dictionnaire au système de recherche d'entreprise et l'associer à une ou à plusieurs collections.

Seul le fichier .dic généré est téléchargé vers le système de recherche d'entreprise. Assurez-vous que le fichier XML source est stocké dans un environnement dont l'accès est contrôlé avec la stratégie de sauvegarde appropriée appliquée. Ce fichier XML est requis pour la mise à jour du dictionnaire de mots avec degré de pondération.

Tâches associées

«Création d'un dictionnaire de synonymes», à la page 56
Une fois que vous avez créé ou mis à jour une liste de synonymes dans un fichier XML, vous devez convertir ce dernier en un dictionnaire de synonymes.

Analyse de texte incluse dans la recherche d'entreprise

L'analyse de texte de la recherche d'entreprise inclut la détection de la langue du document et la segmentation.

Lorsqu'un document est traité, la recherche d'entreprise détermine la langue du document et fragmente le texte d'entrée en unités distinctes ou marqueurs sémantiques.

Lors d'une recherche, l'utilisateur ou une application doit sélectionner manuellement la langue de la requête. La chaîne de la requête est segmentée, analysée et recherchée dans l'index.

L'analyse de la chaîne de requête et du document peut être fractionnée de la manière suivante :

- Support n'utilisant pas le dictionnaire de base. Inclut une segmentation n-gram et par espaces.
- Support linguistique utilisant des dictionnaires. Inclut une segmentation par mots et par phrases et la lemmatisation.

  Le traitement linguistique implique l'analyse lexicale, processus de création de représentations alternatives du texte d'entrée qui associe toutes les données des dictionnaires disponibles aux marqueurs sémantiques reconnus dans le texte d'entrée. Vous pouvez améliorer la pertinence de la recherche en utilisant un traitement avancé de la langue.

  Concepts associés

  «Identification de la langue»
  Avant que la segmentation par mot et par phrase, la normalisation des caractères ou la lemmatisation puissent avoir lieu, la recherche d'entreprise doit déterminer la langue du document source.

  «Support linguistique pour la segmentation effectuée sans dictionnaire», à la page 68
  Pour les documents rédigés dans des langues qui ne sont pas prises en charge par la détection de langue et la technologie d'analyse lexicale, la recherche d'entrepris fournit un support de base sous la forme de segmentation n-gram et d'espace de type Unicode.

Identification de la langue

Avant que la segmentation par mot et par phrase, la normalisation des caractères ou la lemmatisation puissent avoir lieu, la recherche d'entreprise doit déterminer la langue du document source.

La recherche d'entreprise peut automatiquement détecter les langues suivantes :

Allemand	Finnois	Néerlandais
Anglais	Français	Polonais
Arabe	Grec	Portugais
Chinois (traditionnel et simplifié)	Hébreu	Russe
Coréen	Hongrois	Suédois
Danois	Italien	Tchèque
Espagnol	Japonais	Turc

Le processus linguistique de la recherche d'entreprise détecte la langue d'un document source lors de l'indexation et non lors du traitement de la requête.

Dans la recherche d'entreprise, vous pouvez choisir l'option de détection automatique de la langue ou sélectionner une langue à utiliser.

Si vous sélectionnez la détection automatique de la langue et que l'analyseur syntaxique ne peut pas la déterminer, il utilise la langue indiquée lors de la création du moteur de balayage dans la console d'administration de recherche d'entreprise.

Si vous ne sélectionnez pas la détection automatique de la langue, la langue que vous indiquez est toujours utilisée. Par défaut, l'anglais est utilisé.

Les documents pour lesquels il n'existe aucun dictionnaire spécifique à une langue seront traités à l'aide d'une technologie linguistique de base, telle que la segmentation à l'aide d'espaces ou la segmentation n-gram.

La technologie de détection de langue de recherche d'entreprise est la plus adaptée pour les documents en une seule langue. Si un document est rédigé en plusieurs langues, une tentative de détection de la langue dominante du document est effectuée. Toutefois, les résultats de l'analyse ne sont pas toujours satisfaisants.

La langue d'un document peut être utilisée afin de restreindre les résultats de la recherche aux documents rédigés dans une langue spécifique. Par exemple, si vous recherchez des documents sur Jacques Chirac, vous pouvez indiquer que seuls les documents rédigés en français seront inclus dans les résultats de la recherche'.

Concepts associés

«Analyse de texte incluse dans la recherche d'entreprise», à la page 67
L'analyse de texte de la recherche d'entreprise inclut la détection de la langue du document et la segmentation.

«Support linguistique pour la segmentation effectuée sans dictionnaire»
Pour les documents rédigés dans des langues qui ne sont pas prises en charge par la détection de langue et la technologie d'analyse lexicale, la recherche d'entrepris fournit un support de base sous la forme de segmentation n-gram et d'espace de type Unicode.

Support linguistique pour la segmentation effectuée sans dictionnaire

Pour les documents rédigés dans des langues qui ne sont pas prises en charge par la détection de langue et la technologie d'analyse lexicale, la recherche d'entrepris fournit un support de base sous la forme de segmentation n-gram et d'espace de type Unicode.

Segmentation d'espace de type Unicode
Cette méthode de traitement linguistique utilise les espaces entre les mots comme délimiteurs.

Segmentation N-gram
Cette méthode de traitement linguistique traite les séquences de *n* caractères comme un seul mot. Cette méthode simple de segmentation est suffisante pour la plupart des tâches d'extraction.

Ces méthodes n'utilisent aucun dictionnaire et n'incluent aucune technologie de traitement linguistique sophistiquée, telles la réduction à la forme de base.

La segmentation N-gram est utilisée pour des langues, telles le thaïlandais, qui n'ont pas recours aux espaces comme délimiteurs. La même méthode s'applique à l'hébreu et à l'arabe. Bien que ces deux langues utilisent des espaces comme délimiteurs, la segmentation n-gram renvoie de meilleurs résultats que ne le fait la forme de base de la segmentation à l'aide d'espaces de type Unicode.

Concepts associés

«Analyse de texte incluse dans la recherche d'entreprise», à la page 67
L'analyse de texte de la recherche d'entreprise inclut la détection de la langue du document et la segmentation.

«Identification de la langue», à la page 67
Avant que la segmentation par mot et par phrase, la normalisation des caractères ou la lemmatisation puissent avoir lieu, la recherche d'entreprise doit déterminer la langue du document source.

Support linguistique pour la segmentation effectuée à l'aide d'un dictionnaire

Si la langue d'un document est correctement détectée et que des dictionnaires correspondant à cette langue sont disponibles, alors le traitement linguistique approprié est appliqué.

Le processus permettant de séparer le texte d'entrée en des unités lexicales distinctes est appelé segmentation. Ce processus inclut certaines des activités de traitement linguistique suivantes :

Segmentation de mots
La segmentation de mots est employée pour les langues qui n'utilisent pas d'espace (ou délimiteur) entre les mots, pour le japonais ou le chinois, par exemple.

Lemmatisation
La lemmatisation est une forme de traitement linguistique qui détermine le lemme de chaque mot du texte. Le *lemme* d'un mot inclut sa forme de base ainsi que les versions déclinées qui partagent la même classe de mots. Par exemple, le lemme de *go* inclut *go*, *goes*, *went*, *gone* et *going*. Les lemmes de noms comportent le singulier et le pluriel (*calf* et *calves*, par exemple). Les lemmes d'adjectifs incluent les formes comparatives et superlatives (*good*, *better* et *best*, par exemple). Les lemmes des pronoms rassemblent différentes formes du même pronom (*I*, *me*, *my* et *mine*, par exemple).

La lemmatisation nécessite une dictionnaire pour l'indexation et pour la recherche.

La recherche d'entreprise indexe les lemmes et les mots décline et effectue une lemmatisation de tous les mots déclinés dans une requête. La lemmatisation améliore la qualité de la recherche en recherchant des documents qui contiennent des variantes d'un terme décliné dans la requête. Par exemple, les documents contenant le mot *mice* sont trouvés lorsque la requête inclut le mot *mouse*.

Fragmentation des contractions
La qualité de la recherche est améliorée en identifiant les contractions et en les fractionnant. Par exemple :

wouldn't est fractionné en *would* + *not*
Horse's est fractionné en *Horse* + *is* ou *'s*
(pour prendre en compte l'ambiguïté de la requête)

Identification clitique

Les clitiques constituent une forme spéciale de contractions et la qualité de la recherche est améliorée en déterminant les différentes parties de composant. Un *clitique* est un élément qui se comporte comme un affixe et comme un mot. Toutefois, il est difficile d'identifier les clitiques car ils font également partie de la formation du mot. Contrairement à d'autres phénomènes morphologiques (structure des mots), les clitiques se trouvent dans une structure syntaxique et leur lien aux mots ne fait pas partie des règles de formation de mots. Par exemple :

reparti-lo-emos est constitué des composants *repartir + lo + emos*
l'avenue est constitué des composants *le + avenue*
dell'arte est constitué des composants *dello + arte*.

Reconnaissance des caractères non alphabétiques

Les processus linguistiques reconnaissent les caractères non alphabétiques. Selon la logique interne dépendant de la langue, certains caractères non alphabétiques sont renvoyés en tant qu'unités lexicales de types différents et d'autres sont regroupées.

Par exemple, les apostrophes ou les traits d'union sont considérées comme parties intégrantes d'un mot dans le cas de clitiques mais comme des points dans le cas d'abréviations inconnues. Le traitement linguistique peut également reconnaître certaines séquences spéciales de caractères en tant que marques sémantiques, par exemple les URL, les adresses électroniques et les dates.

Reconnaissance des abréviations

Les processus linguistiques reconnaissant les abréviations qui se trouvent dans le dictionnaire en tant qu'une seule unité lexicale. Si l'abréviation ne se trouve pas dans le dictionnaire, l'abréviation est reconnus comme élément lexical mais aucune information de dictionnaire n'est associé à l'abréviation.

Une reconnaissance correcte des abréviations est primordiale pour la reconnaissance des phrases. Par exemple, le point placé à la fin d'une abréviation ne représente pas forcément la fin d'une phrase.

Reconnaissance du marqueur de fin de phrase

Les processus linguistiques identifient correctement les marqueurs de fin de phrase pour la segmentation des phrases.

Le support linguistique à l'aide de dictionnaires est disponible pour les langues suivantes :

```
Allemand (National et Suisse)
Anglais
Arabe
Chinois (simplifié et traditionnel)
Coréen
Tchèque
Danois
Espagnol
Finnois
Français (National et Canadien)
Grec
Italien
Japonais
Néerlandais
Norvégien (bokmal et nynorsk)
Polonais
Portugais (National et Brésilien)
Russe
Suédois
```

Concepts associés

«Segmentation des mots en japonais»
S'il a été détecté que le document ou la chaîne de requête est en japonais, la recherche d'entreprise effectue une segmentation des mots appropriée en utilisant la technologie d'analyse morphologique optimisée pour le japonais.

«Variantes orthographiques en japonais»
La langue japonaise utilise un grand nombre de variantes orthographiques. Les variantes Katakana sont les plus importantes car Katakana est souvent utilisé pour orthographier et prononcer des mots étrangers. Un grand nombre de variantes Katakana sont souvent utilisées en japonais.

Segmentation des mots en japonais

S'il a été détecté que le document ou la chaîne de requête est en japonais, la recherche d'entreprise effectue une segmentation des mots appropriée en utilisant la technologie d'analyse morphologique optimisée pour le japonais.

Cette optimisation est par exemple illustrée par la décomposition des mots. La langue japonaise utilise un grand nombre de mots composés. Ces mots sont décomposés en marques sémantiques de taille optimale afin d'obtenir de meilleurs résultats de recherche. Les mots déclinés et les prépositions sont également décomposés afin d'améliorer les performances de la recherche.

Concepts associés

«Support linguistique pour la segmentation effectuée à l'aide d'un dictionnaire» , à la page 69
Si la langue d'un document est correctement détectée et que des dictionnaires correspondant à cette langue sont disponibles, alors le traitement linguistique approprié est appliqué.

«Variantes orthographiques en japonais»
La langue japonaise utilise un grand nombre de variantes orthographiques. Les variantes Katakana sont les plus importantes car Katakana est souvent utilisé pour orthographier et prononcer des mots étrangers. Un grand nombre de variantes Katakana sont souvent utilisées en japonais.

Variantes orthographiques en japonais

La langue japonaise utilise un grand nombre de variantes orthographiques. Les variantes Katakana sont les plus importantes car Katakana est souvent utilisé pour orthographier et prononcer des mots étrangers. Un grand nombre de variantes Katakana sont souvent utilisées en japonais.

La recherche d'entreprise utilise un dictionnaire de variantes pour mapper des variantes Katakana classiques à leurs formes de base afin que tous les documents, y compris ceux avec des variantes orthographiques du caractère Katakana dans la chaîne de requête, soient trouvés.

La recherche d'entreprise prend également en charge les variantes Okurigana classiques, fins de mots Kanji écrits en Hiragana.

Concepts associés

«Support linguistique pour la segmentation effectuée à l'aide d'un dictionnaire» , à la page 69
Si la langue d'un document est correctement détectée et que des dictionnaires correspondant à cette langue sont disponibles, alors le traitement linguistique approprié est appliqué.

«Segmentation des mots en japonais»
S'il a été détecté que le document ou la chaîne de requête est en japonais, la

recherche d'entreprise effectue une segmentation des mots appropriée en utilisant la technologie d'analyse morphologique optimisée pour le japonais.

Suppression des mots vides

Dans la recherche d'entreprise, tous les mots vides, par exemple des mots communs, *un* et *le* par exemple, sont supprimés des requêtes comportant plusieurs mots afin d'améliorer les performances de la recherche.

La reconnaissance des mots vides pour le japonais s'effectue à l'aide d'informations grammaticales. Par exemple, la recherche d'entreprise reconnaît si le mot est un nom ou une verbe alors que pour les autres langues, des listes spéciales sont utilisées.

Concepts associés

«Normalisation des caractères»
La normalisation des caractères est un processus qui peut améliorer le rappel. En améliorant le rappel à l'aide de la normalisation des caractères, un plus grand nombre de documents est extrait même si les documents ne correspondent pas exactement à la requête.

Normalisation des caractères

La normalisation des caractères est un processus qui peut améliorer le rappel. En améliorant le rappel à l'aide de la normalisation des caractères, un plus grand nombre de documents est extrait même si les documents ne correspondent pas exactement à la requête.

La recherche d'entreprise utilise la normalisation de compatibilité Unicode qui inclut la normalisation des caractères asiatiques demi-largeur et de largeur standard.

Par exemple, en japonais, un caractère alphanumérique de largeur standard est normalisé en caractère demi-largeur, un caractère Katakana demi-largeur en largeur standard, etc. La recherche d'entreprise supprime également les points qui sont utilisés en tant que délimiteurs de mots composés en japonais.

D'autres formes de normalisation de caractères incluent :

Normalisation majuscules/minuscules
Par exemple, la recherche de documents comportant *USA* lorsque la chaîne *usa* a été indiquée.

Développement des trémas allemands (Umlaut)
Exemple : Recherche des documents qui contiennent *schoen* lors de la recherche de *schön*.

Suppression des accents
Exemple : Recherche des documents contenant le caractère *é* lors de la recherche du caractère *e*.

Suppression d'autres signes diacritiques
Exemple : Recherche de documents contenant le caractère *ç* lors de la recherche du caractère *c*.

Développement de la ligature
Exemple : Recherche des documents contenant les caractères *Æ* lors de la recherche des caractères *ae*.

Toutes les normalisations fonctionnent dans les deux sens. Vous pouvez trouver des documents contenant *usa* lorsque vous recherchez *USA*, des documents contenant des mots avec le caractère *e* lorsque vous recherchez le caractère *é*, etc. Ces normalisations peuvent être associées les unes aux autres. Par exemple, vous pouvez trouver des documents comportant le terme *météo* lorsque vous recherchez *METEO*.

Les normalisations sont effectuées en fonction des propriétés des caractères Unicode et ne dépendent pas des langues. Par exemple, la recherche d'entreprise prend en charge la suppression des caractères diacritiques pour l'hébreu et le développement de la ligature pour l'arabe.

Concepts associés

«Suppression des mots vides», à la page 72
Dans la recherche d'entreprise, tous les mots vides, par exemple des mots communs, *un* et *le* par exemple, sont supprimés des requêtes comportant plusieurs mots afin d'améliorer les performances de la recherche.

Documentation relative à la recherche d'entreprise

La documentation WebSphere Information Integrator OmniFind Edition (recherche d'entreprise) est disponible au format PDF ou HTML.

Le programme d'installation WebSphere Information Integrator OmniFind Edition peut automatiquement installer le centre de documentation. Le programme d'installation installe le centre de documentation sur le serveur de recherche. Si vous n'installez pas le centre de documentation, ce dernier s'affiche dans le site Web IBM lorsque vous cliquez sur l'aide. Pour voir les rubriques HTML pour la recherche d'entreprise, démarrez le centre de documentation.

Pour voir les documents PDF, accédez au répertoire docs/*locale*/pdf. Pour afficher par exemple les documents en anglais, accédez au répertoire docs/en_US/pdf. Vous pouvez également consulter la documentation PDF la plus récente à partir du site de support WebSphere Information Integrator OmniFind Edition.

Le tableau suivant présente la documentation disponible, les noms de fichier et les emplacements.

Tableau 9. Documentation PDF pour la recherche d'entreprise

En-tête	En-tête	En-tête
Installation Guide for Enterprise Search (les sections de ce document sont également disponibles dans le centre de documentation)	iiysi.pdf	docs/*locale*/pdf/
Administering Enterprise Search (les sections de ce document sont également disponibles dans le centre de documentation.)	iiysa.pdf	docs/*locale*/pdf/
Programming Guide and API Reference for Enterprise Search (Les sections de ce document sont également disponibles dans le centre de documentation.)	iiysp.pdf	docs/*locale*/pdf/
Référence des messages pour Enterprise Search (Les sections de ce document sont également disponibles dans le centre de documentation.)	iiysm.pdf	docs/*locale*/pdf/
Installation Requirements for Enterprise Search (Les sections de ce document sont également disponibles dans le centre de documentation.)	iiysr.txt ou iiysr.htm	docs/*locale*/ (ce fichier peut également être lancé à partir du programme Premiers pas.)
Notes sur l'édition	iiysn.pdf	Disponible uniquement sur le site Web de la documentation IBM WebSphere Information Integrator OmniFind Edition.

Tableau 9. Documentation PDF pour la recherche d'entreprise (suite)

En-tête	En-tête	En-tête
Centre de documentation WebSphere Information Integrator	Non disponible	

WebSphere II OmniFind Edition - Accessibilité

Les interfaces utilisateur et la documentation IBM WebSphere Information Integrator OmniFind Edition sont accessibles.

Programme d'installation

Vous pouvez utiliser les raccourcis clavier pour parcourir le programme d'installation WebSphere II OmniFind Edition Le tableau suivant décrit certains raccourcis clavier.

Tableau 10. Raccourcis clavier du programme d'installation

Action	**Raccourci**
Mise en évidence d'un bouton d'option	Touche de direction
Sélection d'un bouton d'option	Touche Tabulation
Mise en évidence d'un bouton de fonction	Touche Tabulation
Sélection d'un bouton de fonction	Touche Entrée
Accès à la fenêtre précédente ou suivante ou annulation	Mise en évidence d'un bouton de fonction en appuyant sur la touche Tabulation puis sur la touche Entrée
Rendre la fenêtre active inactive	Ctrl + Alt + Esc

Console d'administration de recherche d'entreprise et centre de documentation

La console d'administration et le centre de documentation sont des interfaces utilisant un navigateur que vous pouvez afficher dans Microsoft Internet Explorer ou Mozilla FireFox. Reportez-vous à l'aide en ligne d'Internet Explorer ou de FireFox pour obtenir une liste des raccourcis clavier et des autres fonctions d'accessibilité de votre navigateur.

Documentation PDF

Vous pouvez afficher l'ensemble de la documentation de la recherche d'entreprise au format PDF. Adobe Acrobat Version 6.0 permet d'afficher les documents PDF. Les documents PDF sont structurés et doivent être lisibles par la plupart des lecteurs d'écran.

Glossaire de termes pour la recherche d'entreprise

Ce glossaire d├®finit les termes utilis├®s dans les interfaces de recherche d'entreprise et dans la documentation.

administrateur de recherche d'entreprise
Rôle administratif qui permet à un utilisateur de gérer l'intégralité du système de recherche d'entreprise.

adresse IP
Adresse 32 bits unique qui identifie un hôte sur le réseau.

adresse URL de départ
Point de départ d'une exploration.

affinité lexicale
Relation des mots de la recherche proches les uns des autres dans le document. L'affinité lexicale permet de calculer la pertinence d'un résultat.

agent utilisateur
Application qui parcourt le Web et laisse des informations sur le site visité. Dans la recherche d'entreprise, le moteur de balayage Web est un agent utilisateur.

analyse des liens
Méthode basée sur l'analyse des liens hypertexte entre les documents et permettant de déterminer les pages importantes pour l'utilisateur dans la collection.

analyse de texte
Processus d'extraction de la sémantique et d'autres informations à partir du texte afin d'améliorer la capacité d'extraction des données dans une collection.

analyseur syntaxique
Programme qui interprète des documents ajoutés au magasin de données de la recherche d'entreprise. L'analyseur syntaxique extrait des informations des documents et les prépare pour l'indexation, la recherche et l'extraction.

annotateur
Composant logiciel qui effectue des tâches d'analyse syntaxique et qui génère et enregistre des annotations. Un annotateur correspond au composant logique d'analyse dans un moteur d'analyse.

annotation
Informations sur une étendue de texte. Par exemple, une annotation peut indiquer qu'une étendue de texte représente un nom d'entreprise. Dans l'architecture UIMA, une annotation est un type spécial de structure de fonctions.

application de recherche
Programme qui traite des requêtes, effectue des recherches dans l'index, renvoie les résultats de la recherche et extrait les documents source pour les collections dans un système de recherche d'entreprise.

arborescence des catégories
Hiérarchie des catégories affichée dans la console d'administration de recherche d'entreprise.

archive du moteur de traitement
Fichier d'archive zip .pear qui inclut un moteur d'analyse UIMA et l'ensemble des ressources requises pour l'analyse personnalisée dans la recherche d'entreprise.

attribution d'un score en fonction du texte
Processus permettant d'attribuer une valeur d'entier à un document qui définit la pertinence du document par rapport aux termes d'une requête. Une valeur d'entier élevée correspond à une correspondance plus proche de la requête. Voir aussi classement dynamique.

autorité de certification
Organisation qui émet des certificats et authentifie les entités (individus ou entreprises) impliquées dans des transactions électroniques. Les autorités de certification garantissent que les deux parties échangeant des informations sont vraiment ce qu'elles affirment être.

bibliothèque
Objet système ayant la fonction de répertoire pour d'autres objets. Voir aussi bibliothèque Domino Document Manager.

bibliothèque Domino Document Manager
Une base de données Domino Document Manager qui constitue le point d'entrée de Domino Document Manager.

caractère d'échappement
Caractère qui supprime ou sélectionne une signification particulière pour un ou plusieurs caractères.

caractère de fin
Caractère indiquant le dernier emplacement d'un mot.

caractère de masquage
Caractère permettant de représenter des caractères facultatifs au début, au milieu et à la fin d'un terme de recherche. Les caractères de masquage permettent généralement de rechercher des variations d'un terme dans un index. Voir aussi caractère générique.

caractère de nouvelle ligne
Caractère de contrôle qui provoque le déplacement d'une ligne de l'emplacement d'impression ou d'affichage. Pour certains systèmes, plusieurs caractères sont requis.

caractère générique
Caractère utilisé pour représenter des caractères facultatifs au début, milieu et la fin d'un terme de recherche.

catégorie
Groupe de documents ayant des propriétés similaires.

catégorie de type modèle
Taxinomie des termes prédéfinis permettant de déterminer le sujet d'un document afin que ce dernier soit indexé et que des recherches y soient effectuées en même temps que dans les documents ayant un contenu similaire.

catégorie de type règle
Catégories qui sont créées par des règles définissant quels documents sont associés à quelles catégories. Par exemple, vous pouvez définir des règles à associer aux documents qui contiennent ou excluent certains mots ou qui correspondent à un masque d'URI, avec des catégories spécifiques.

certificat
Document numérique qui associe une clé publique à l'identité du propriétaire du certificat, activant ainsi l'authentification du propriétaire du certificat. Un certificat est émis par une autorité de certification.

chemin de fonctions
Chemin permettant d'accéder à la valeur d'une fonction dans une structure de fonctions UIMA.

classe de pondération
Spécification pouvant influencer le classement relatif d'un document dans les résultats de la recherche.

classement
Processus permettant d'attribuer une valeur d'entier à chaque document dans les résultats de la recherche à partir d'une requête. L'ordre des documents dans les résultats de la recherche se fonde sur la pertinence de la requête. Un classement plus élevé génère une correspondance plus approchante. Voir aussi classement dynamique et classement statique.

classement dynamique
Type de classement dans lequel les termes de la requête sont analysés en fonction des documents recherchés afin de déterminer le classement des résultats. Voir aussi attribution d'un score en fonction du texte. Opposé à classement statique.

classement populaire
Type de classement ajouté à un classement existant d'un document en fonction de la popularité du document.

classement statique
Type de classement dans lequel les facteurs concernant les documents classés, tels la date, le nombre de liens qui désignent le document, etc., augmentent le classement. Opposé à classement dynamique.

collection
Ensemble de sources de données et d'options permettant de parcourir, d'analyser, d'indexer ces sources de données et d'y effectuer des recherches.

contrôleur
Utilisateur de recherche d'entreprise qui dispose du droit d'observation des processus de niveau collection.

couche CCL (Common Communication Layer)
Infrastructure de communications qui unit les différents composants (contrôleur, analyseur, moteur de balayage, outil d'indexation) de WebSphere Information Integrator OmniFind Edition.

diacritique
Marque ajoutée à une lettre pour modifier la prononciation d'un mot ou pour effectuer la distinction entre des mots similaires, telle un accent ou le caractère tréma allemand (Umlaut).

dictionnaire de synonymes
Dictionnaire qui permet aux utilisateurs de rechercher des synonymes des termes de la requête lors de la recherche dans une collection.

DIIOP (Domino Internet Inter-ORB Protocol)
Tâche serveur qui s'exécute sur le serveur et fonctionne avec Domino Object Request Broker afin de permettre des communications entre les applets Java créées avec les classes Notes Java et le serveur Domino. Les

utilisateurs de navigateur et les serveurs Domino utilisent DIIOP pour communiquer et pour échanger des données d'objet.

directive no-follow
Directive d'une page Web qui indique aux robots (tels le moteur de balayage Web) de ne pas suivre les liens de ces pages.

directive no-index
Directive d'une page Web qui indique aux robots (tels le moteur de balayage Web) de ne pas inclure le contenu de ces pages dans l'index.

DOM (Document Object Model)
Système dans lequel un document structuré, tel un fichier XML, est affiché sous la forme d'arborescence d'objets. Il est possible d'accéder à ces objets ou de les mettre à jour à l'aide d'un programme.

données d'identification
Informations détaillées, acquises lors de l'authentification, qui décrivent l'utilisateur, des associations de groupe et d'autres attributs d'identité liés à la sécurité. Vous pouvez utiliser des données d'identification pour effectuer un grand nombre de services, tels l'autorisation, l'audit et la délégation.

dossier Domino Document Manager
Base de données Domino Document Manager permettant d'organiser les documents. Les dossiers contiennent des bases de données Domino.

élément clitique
Mot qui fonctionne syntaxiquement séparément mais qui est phonétiquement connecté à un autre mot. Un élément clitique peut être représenté comme associé ou séparé du mot auquel il est lié. Des exemples communs d'éléments clitiques incluent la dernière partie d'une contraction en anglais (*would**n't*** ou *you**'re***).

emplacement
Programme qui permet aux utilisateurs de créer des documents, de répondre à des commentaires d'autres utilisateurs et de consulter le statut et le délai des projets. Les utilisateurs peuvent également discuter avec d'autres utilisateurs qui se trouvent dans le même espace. Voir aussi emplacement Lotus QuickPlace.

emplacement Lotus QuickPlace
Zone partitionnée d'un espace Lotus QuickPlace restreinte à des membres autorisés qui partagent un centre d'intérêt commun et qui ont besoin de travailler ensemble.

espace Emplacement virtuel visible dans le portail dans lequel des individus et des groupes se retrouvent pour collaborer. Dans un portail, chaque utilisateur dispose d'un espace personnel pour un travail privé et les individus et les groupes disposent d'un accès à un ensemble d'espaces partagés qui peuvent être des espaces publics ou des espaces restreints. Voir aussi espace Lotus QuickPlace.

espace Lotus QuickPlace
Emplacement Web fourni par Lotus QuickPlace qui permet à des participants se trouvant à différents lieux géographiques de collaborer sur des projets et de communiquer en ligne dans un espace de travail structuré et sécurisé.

espace d'exploration
Ensemble de sources qui correspondent aux masques indiqués (tels des noms de base de données, des chemins de système de fichiers, des noms

de domaine, des adresses IP et des URL) qu'un moteur de balayage utilise pour extraire des éléments pour l'indexation.

extraction de concept
Fonction de recherche qui identifie des éléments de vocabulaire significatifs (tels personnes, emplacements ou produits) dans des documents de texte et qui génère une liste de ces éléments. Voir aussi extraction de thème.

extraction de thème
Type d'extraction de concept qui reconnaît automatiquement des éléments de vocabulaire significatifs dans des documents de texte pour extraire le thème ou le sujet d'un document. Voir aussi extraction de concept.

extraction d'informations
Type d'extraction de concept qui reconnaît automatiquement des éléments de vocabulaire significatifs, tels des noms, des termes et des expressions dans des documents de texte.

fédérateur distant
Fédérateur serveur qui fédère un ensemble d'objets pouvant être recherchés.

fédérateur local
Fédérateur client qui fédère un ensemble d'objets pouvant être recherchés.

fédération
Processus d'association de systèmes d'attribution de noms afin que le système agrégé puisse traiter des noms composites qui étendent les systèmes d'attribution de noms.

fichier de stockage de clés
Fichier de base de données de clés qui contient des clés publiques stockées en tant que certificats de signataire et des clés privées stockées dans des certificats personnels.

fichiers d'index de recherche
Ensemble de fichiers dans lesquels un index est stocké dans le moteur de recherche.

file d'attente d'index
Liste de requêtes pour la réorganisation d'index ou de requêtes pour la régénération d'index à traiter.

gestion des identités
Fonction permettant de chiffrer les autorisations d'accès utilisateur dans un magasin sécurisé.

identification de la langue
Fonction de recherche d'entreprise qui détermine la langue d'un document.

index Voir index de recherche.

index de texte complet
Structure de données qui référence des éléments de données afin que la recherche trouve rapidement des documents qui contiennent les termes de la requête.

JavaScript
Langage de scriptage Web utilisé dans des navigateurs et des serveurs Web.

JDBC (Java Database Connectivity)
Norme de l'industrie pour la connectivité indépendante de la base de

données entre la plateforme Java et une gamme importante de bases de données. L'interface JDBC fournit une API de niveau appel pour l'accès à la base de données de type SQL.

jeton de sécurité
Informations sur l'identité et la sécurité utilisées pour l'autorisation d'accès aux documents d'une collection. Les différents types de source de données prennent en charge différents types de jeton de sécurité. Les exemples incluent des rôles utilisateur, des ID utilisateur, des ID de groupe et d'autres informations pouvant être utilisées pour contrôler l'accès au contenu.

Katakana
Ensemble de caractères composé de symboles utilisés dans un des deux alphabets phonétiques japonais communs, utilisé principalement pour représenter phonétiquement des mots étrangers.

lemmatisation
Processus de recherche du lemme pour un mot donné dans un dictionnaire. La lemmatisation est différente de la recherche de radical, la recherche de radical étant algorithmique et n'utilisant généralement pas de dictionnaire qui répertorie les mots d'une langue.

lemme
Forme canonique d'un mot. Les lemmes sont importants dans des langues à fortes déclinaisons, telles le tchèque.

lien rapide
Association entre un URI et des mots clés et des phrases.

ligature
Au moins deux caractères associés afin qu'ils apparaissent sous la forme d'un seul caractère. Exemple : l'association des caractères f et i forme la ligature ?.

liste de contrôle d'accès
Liste identifiant les utilisateurs pouvant accéder à l'objet associé et qui indique les droits d'accès de l'utilisateur à cet objet.

Langage de chemin XML (XPath)
Langage qui identifie de manière unique ou adresse les parties d'un document XML source. XPath fournit également les fonctions de base pour la manipulation de chaînes, de nombres et d'opérateurs booléens.

machine virtuelle Java (JVM)
Implémentation logicielle d'un processeur qui exécute du code Java compilé (applets et applications).

marque sémantique
Unités textuelles de base indexées par la recherche d'entreprise. Les marques sémantiques peuvent être des mots d'un langue ou d'autres unités de texte convenant à l'indexation.

marqueur sémantique
Programme de segmentation de texte qui analyse le texte et détermine si un ensemble de caractères peut être reconnu en tant que marque sémantique.

mémoire cache de recherche
Mémoire tampon qui conserve les données et les résultats des requêtes de recherche précédentes.

mot avec degré de pondération
Mot pouvant influencer le classement relatif d'un document dans les résultats de la recherche. Lors du traitement de la requête, l'importance d'un document qui contient un mot avec degré de pondération peut être augmentée ou diminuée en fonction d'un score prédéfini pour le mot.

moteur d'analyse
Voir moteur d'analyse de texte.

moteur d'analyse de texte
Composant logiciel chargé de la recherche et de la représentation du contexte et du contenu sémantique dans le texte.

moteur de balayage
Programme logiciel qui extrait des documents de sources de données et qui rassemble des informations pouvant être utilisées pour créer des index de recherche.

moteur de balayage Web
Classe de logiciel robot qui explore le Web en extrayant un document Web et en suivant les liens de ce document.

moteur de recherche
Programme qui accepte une requête de recherche et renvoie une liste de documents à l'utilisateur.

mot vide
Mot souvent utilisé, tel que *the*, *an* ou *and*, ignoré par une application de recherche.

nom distinctif
Nom qui identifie de manière unique une entrée dans un annuaire. Un nom distinctif est composé de paires attribut:valeur, séparées par des virgules. Il peut également s'agir de paires nom-valeur (telles CN=nom de la personne et C=pays ou région) qui identifie de manière unique une entité dans un certificat numérique.

normalisation des caractères
Processus dans lequel les formes variantes d'un caractère, tels la capitalisation et les marques diacritiques, sont réduites à une forme commune.

NRPC (Notes remote procedure call)
Couche d'architecture Lotus Notes utilisée pour toutes les communications Notes-à-Notes.

opérateur
Utilisateur de recherche d'entreprise qui dispose du droit d'observation, de démarrage et d'arrêt des processus de niveau collection.

outil de reconnaissance
Fonction d'un moteur de balayage qui détermine les sources de données disponibles pour que le moteur de balayage puisse en extraire des informations.

page d'erreur temporaire
Page spéciale qui explique de manière détaillée pourquoi un serveur HTTP ne peut pas renvoyer la page demandée par un client et qui configure le serveur HTTP afin qu'il renvoie ces pages à la place d'une réponse composée uniquement d'un en-tête avec un code retour qui explique le problème.

pages JSP (JavaServer Pages)
Technologie de scriptage de serveur qui active le code Java pour qu'il soit dynamiquement intégré à des pages Web (fichiers HTML) et exécuté lorsque la page est servie afin de renvoyer un contenu dynamique à un client.

placer en file d'attente
Placer des éléments dans une file d'attente.

portée Groupe d'URI connexes permettant de définir la plage d'une requête de recherche.

protocole d'exclusion de robots
Protocole qui permet aux administrateurs de site Web d'indiquer aux robots de visite les parties du site auxquelles il ne doit pas accéder.

protocole LDAP (Lightweight Directory Access Protocol)
Protocole ouvert qui utilise TCP/IP pour offrir un accès aux répertoires qui prennent en charge un modèle X.500 et qui n'est pas concerné par les besoins en ressources du protocole X.500 plus complexe.

récapitulation
Processus permettant d'inclure des phrases dans les résultats afin de décrire brièvement le contenu d'un document. Voir aussi récapitulation dynamique et récapitulation statique.

récapitulation dynamique
Type de récapitulation dans laquelle les termes de la recherche sont mis en évidence et les résultats de la recherche contiennent des phrases qui représentent le mieux des concepts du document que l'utilisateur recherche. Opposé à récapitulation statique.

récapitulation statique
Type de récapitulation dans laquelle les résultats de la recherche contiennent un récapitulatif stocké spécifique provenant du document. Opposé à récapitulation dynamique.

recherche à texte libre
Recherche dans laquelle le terme de recherche est exprimé en tant que texte à forme libre.

recherche booléenne
Recherche dans laquelle un ou plusieurs termes de recherche sont associés à l'aide d'opérateurs, tels que NOT, AND et OR.

recherche dans une zone
Requête restreinte à une zone particulière.

recherche de proximité
Type de recherche de certains mots dans la même phrase, le même mot ou document.

recherche de radical
voir recherche de radical de mot.

recherche de radical de mot
Processus de normalisation linguistique dans lequel les formes variantes d'un mot sont réduites à une forme commune. Par exemple, les mots tels *connexions*, *connectivité*, et *connecté* sont réduits à la forme *connect*.

recherche fédérée
Fonction de recherche qui permet la recherche dans plusieurs services de recherche et renvoie une liste consolidée de résultats de recherche.

recherche floue
Recherche qui renvoie des mots avec des orthographes similaires à celle du terme de la recherche.

recherche hybride
Recherche booléenne combinée associée à la recherche de texte libre.

recherche linguistique
Type de recherche qui parcourt, extrait et indexe un document avec les termes réduits à leur forme de base (par exemple *mice* est indexé en tant que *mouse*) ou développés avec leur forme de base (comme avec les mots composés).

recherche paramétrique
Type de recherche qui recherche des objets qui contiennent un attribut ou une valeur numérique, tels des dates, des chiffres entiers ou d'autres types de données numériques dans une plage indiquée.

régénération d'index
Processus permettant d'ajouter des informations à un index existant dans un système de recherche d'entreprise. Opposé à réorganisation d'index.

réorganisation d'index
Processus de génération d'index dans un système de recherche d'entreprise. Opposé à régénération d'index.

requête de langage naturel
Type de recherche qui analyse des expressions rédigées (telles ″Qui est responsable du service finances ?″) et non une collection simple de mots clés.

résultats d'analyse
Informations générées par les annotateurs. Les résultats d'analyse qui correspondent aux informations recherchées sont placés dans une structure de données appelée structure d'analyse commune.

résultats de la recherche
Liste de documents qui correspondent à la requête de recherche.

retirer de la file d'attente
Retirer des éléments d'une file d'attente.

rôle administratif
Classification d'un utilisateur qui détermine les fonctions que l'utilisateur peut effectuer dans la console d'administration de recherche d'entreprise. Le rôle détermine également les collections que l'utilisateur peut administrer.

segmentation
Processus grâce auquel le contrôle de chemin divise les informations de base en plus petites unités, appelées segments BIU, afin de gérer des mémoires tampon de plus petite taille dans des serveurs adjacents.

Segmentation d'espace de type Unicode
Méthode de marquage sémantique qui utilise les propriétés de caractère Unicode pour effectuer une distinction entre les caractères de séparateur et de marque sémantique.

segmentation n-gram
Méthode d'analyse qui considère les séquences de chevauchement d'un

nombre de caractères comme un seul mot au lieu d'utiliser les espaces pour délimiter des mots comme dans la segmentation à l'aide d'espaces Unicode.

serveur proxy
Serveur qui se comporte comme intermédiaire pour les requêtes Web HTTP hébergées par une application ou un serveur Web. Un serveur proxy se comporte comme un représentant des serveurs de contenu dans l'entreprise.

servlet
Programme Java qui s'exécute sur un serveur Web et développe la fonctionnalité du serveur en générant un contenu dynamique en réponse aux requêtes client Web. Les servlets sont généralement utilisés pour la connexion des bases de données au Web.

source de données
Tout référentiel de données à partir duquel des documents peuvent être extraits, tel le Web, des bases de données relationnelle ou non relationnelle et des systèmes de gestion de contenu.

source de données externe
Source de données pour la fédération qui n'est pas explorée, anlaysée ou indexée par WebSphere Information Integrator OmniFind Edition. Les recherches de sources de données externes sont déléguées à l'interface de programmation d'application de requête de ces sources de données.

SSL (Secure Sockets Layer)
Protocole de sécurité qui permet la confidentialité des communications.

structure d'analyse commune
Structure qui stocke un document analysé par un moteur d'analyse de texte. Les informations sont stockées dans la structure d'analyse commune sous la forme d'annotations et d'autres structures de fonctions.

structure de fonctions
Structure de données sous-jacentes qui représente le résultat de l'analyse de texte. Une structure de fonctions est une structure attribut-valeur. Chaque structure de fonctions est d'un type et chaque type a un ensemble défini de fonctions ou attributs valides, de la même manière qu'une classe Java.

suppression des mots vides
Processus de suppression des mots vides de la requête afin d'ignorer les mots communs et d'obtenir des résultats plus pertinents.

taxinomie
Classification d'objets dans des groupes selon des similarités. Dans la recherche d'entreprise, une taxinomie organise les données en catégories et en sous-catégories. Voir aussi arborescence des catégories.

terme de recherche pondéré
Requête dans laquelle une importance plus élevée est donnée à certains termes.

Type de MIME
Norme Internet permettant d'identifier le type d'objet transféré via Internet.

type de source de données
Regroupement de sources de données en fonction du protocole utilisé pour l'accès aux données.

UIMA (Unstructured Information Management Architecture)
Architecture IBM architecture qui définit une structure d'implémentation des systèmes pour l'analyse des données non structurées.

URI (Uniform Resource Identifier)
Chaîne compacte de caractères qui identifie une ressource abstraite ou physique.

URL (Uniform Resource Locator)
Suite de caractères qui représente des ressources sur un ordinateur ou un réseau, tel l'Internet. Cette suite de caractères inclut le nom abrégé du protocole utilisé pour accéder aux ressources ainsi que les informations utilisées par le protocole pour localiser les informations.

URN (Universal Resource Name)
Elément de protocole Internet constitué d'une petite chaîne de caractères respectant une syntaxe donnée. La chaîne est composée d'un nom ou d'une adresse pouvant faire référence à une ressource.

zone La plus petite partie identifiable d'un enregistrement.

Accès aux informations concernant WebSphere Information Integration

Vous pouvez obtenir des informations sur les produits WebSphere par téléphone ou via notre site Web.

Les numéros de téléphone suivants sont valables pour les Etats-Unis :

- Pour commander des produits ou pour obtenir des informations générales, composez le 1-800-426-2255
- Pour commander des publications, composez le 1-800-879-2755

Des informations sur WebSphere Information Integration sont également disponibles sur le Web à l'adresse www.ibm.com/software/data/integration/db2ii/. Ce site contient les toutes dernières informations sur :

- La documentation du produit
- Les téléchargements du produit
- Les groupes de correctifs (Fix packs)
- Les notes sur l'édition et la documentation sur un autre support
- WebSphere Information Integration
- Les liens vers les ressources Web, tels que livres blancs et IBM Redbooks
- Les liens vers des forums et des groupes d'utilisateurs
- Les liens vers des centres d'informations des produits WebSphere Information Integration
- La commande de manuels

Pour accéder à la documentation du produit, procédez comme suit :

1. Visitez le site Web à l'adresse suivante : www.ibm.com/software/data/integration/db2ii/.
2. Sélectionnez un produit dans la liste déroulante, par exemple WebSphere Information Integrator OmniFind Edition.
3. Cliquez sur le lien Support sur le côté gauche de la page.
4. Dans la section Learn, sélectionnez le lien souhaité. S'il existe un centre de documentation pour le produit choisi, vous pouvez sélectionner le lien correspondant. Pour obtenir un exemple, voir figure 1, à la page 92.

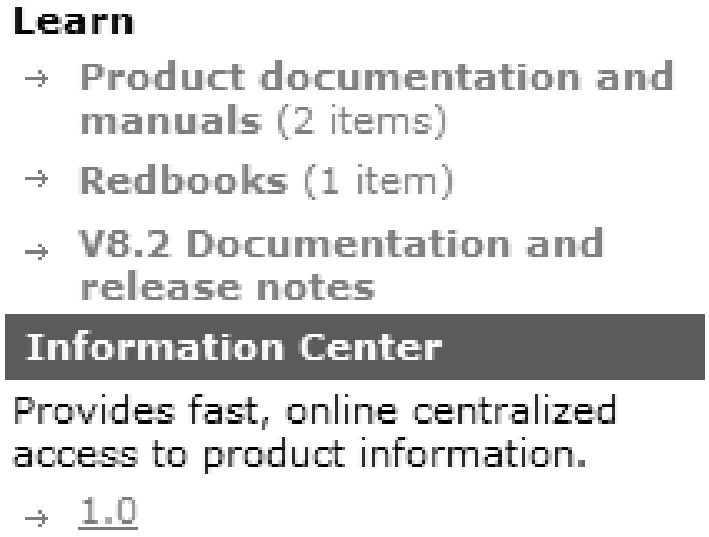

Figure 1. Example de liens vers la documentation du produit sur un site Web WebSphere Information Integration Support

Commentaires sur la documentation

N'hésitez pas à nous envoyer vos commentaires concernant le présent manuel ou la documentation IBM WebSphere Information Integration.

Vos remarques permettent à IBM d'améliorer la qualité des informations fournies. N'hésitez pas à nous envoyer vos commentaires concernant le présent manuel ou la documentation WebSphere Information Integration. Vous pouvez procéder d'une des manières suivantes pour effectuer des commentaires :

1. Envoyez-nous vos commentaires à l'aide du formulaire en ligne disponible à la page suivante www.ibm.com/software/awdtools/rcf/ .
2. Vous pouvez également nous les envoyer par courrier électronique à l'adresse comments@us.ibm.com. Veuillez indiquer le nom du produit, sa version et le nom et la référence du manuel (le cas échéant). Pour tout commentaire relatif à un texte spécifique, veuillez nous en indiquer l'emplacement (titre, numéro de table ou numéro de page, par exemple).

Comment prendre contact avec IBM

Pour contacter le service client IBM aux Etats-Unis ou au Canada, composez le 1-800-426-7378.

Pour connaître les options de service disponibles, contactez IBM aux numéros suivants :

- Aux Etats-Unis : 1-888-426-4343
- Au Canada : 1-800-465-9600

Pour trouver un bureau IBM dans votre pays ou votre région, consultez l'annuaire en ligne des contacts internationaux d'IBM sur le Web à l'adresse suivante : http://www.ibm.com/planetwide

Marques

Cette rubrique recense les marques IBM et certaines marques non IBM.

Pour obtenir des informations sur les marques IBM, voir http://www.ibm.com/legal/copytrade.shtml.

Les termes qui suivent sont des marques d'autres sociétés :

Java, ou toutes les marques et logos incluant Java, sont des marques de Sun Microsystems, Inc. aux Etats-Unis et/ou dans certains autres pays.

Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

Intel, Intel Inside (logos), MMX et Pentium sont des marques de Intel Corporation aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque enregistrée de The Open Goup aux Etats-Unis et/ou dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

D'autres sociétés sont propriétaires des autres marques, noms de produits ou logos qui pourraient apparaître dans le présent document.

Remarques

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans tous les pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous souhaitez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM EMEA Director of Licensing
IBM Europe Middle-East Africa
Tour Descartes
La Défense 5
2, avenue Gambetta
92066 - Paris-La Défense CEDEX
France

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations
IBM Canada Ltd.
3600 Steeles Avenue East
Markham, Ontario
L3R 9Z7
Canada

Les informations sur les licences concernant les produits utilisant un jeu de caractères à deux octets (DBCS) peuvent être obtenues par écrit à l'adresse suivante :

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome,
Minato-ku Tokyo 106-0032,
Japan

Le paragraphe suivant ne s'applique ni au Royaume-Uni ni dans aucun autre pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE «EN L'ETAT». IBM DECLINE TOUTE RESPONSABILITE, EXPRESSE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE QUALITE MARCHANDE OU D'ADAPTATION A VOS BESOINS. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003 U.S.A.

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'ICA, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins

illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

LICENCE DE COPYRIGHT :

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation IBM.

Toute copie totale ou partielle de ces programmes exemples et des oeuvres qui en sont dérivées doit comprendre une notice de copyright, libellée comme suit :

Outside In (®) Viewer Technology, ©1992-2005 Stellent, Chicago, IL., Inc. All Rights Reserved.

Elément processeur XSLT IBM sous licence - Propriété d'IBM ©Copyright IBM Corp., 1999-2005. All Rights Reserved.

Index

U

V

W

SC11-2398-00

Spine information:

WebSphere II OmniFind Edition

WebSphere II OmniFind Edition - Intégration de l'analyse de texte

Version 8.3