Oracle Knowledge Intelligent Search Language Tuning Guide

A Guide to Tuning Oracle Knowledge Intelligent Search

Oracle Knowledge Version 8.6
Document Number OKIS-LTUN86-00
March 2015

COPYRIGHT INFORMATION

Copyright © 2002, 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS:

Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Contents

Preface About This Guide 1

In This Guide 1
Screen and Text Representations 2
References to World Wide Web Resources 2

Chapter 1 The Language Workbench 3

Opening Oracle Knowledge Language Workbench 3
Dictionary Manager 6
Personalized Navigation 7
Test Drive 8
Quality Monitor 9
User Manager 9
Keyword Search 10
Overview 10
NLP Search vs. Keyword Search Overview 10
Benefits and Features 10
Easy Language Management 10
String Match 11
Word Separation 11
Non-white-space Languages 11
Industry Terminology 11
Global Terms and Phrases Work Automatically 11
Automatic or Manual Language Detection 11
Information Manager Content 11
Non-Information Manager Content (e.g. Web-content) 11
Encoding 12
Findability 12
Configuration 12
Enabling a pre-configured Language or Locale for Keyword Search 12
Adding a Language or Locale for Keyword Search 13
Language Model For Global Customers 14

Locales In Content Authoring . . . 14
Automatic Language-to-Locale Mapping . . . 15
Optional Region/Country-specific Language Modeling . . . 16
Global Terms . . . 17
Automatic language configuration . . . 17
The Inheritance Model . . . 17

Chapter 2 Dictionary Manager . . . **19**

Concepts . . . 19
Role of Concepts in Unstructured Search Results . . . 20
Adjusting the Importance Level of Concepts . . . 21
Role of Concepts in Rules and Intents . . . 22
Basic Concept Relationships . . . 23
Parent / Child Relationships . . . 23
Cluster Relationships . . . 23
Additional IML Tips . . . 23
Create Root Concepts . . . 23
Create New Concept . . . 24
Add Synonyms . . . 26
Add Synonym Set (synset) . . . 27
Add Related Concepts . . . 28
Add Cluster Relationship . . . 29
Concept Tools . . . 29
Ontology Viewer . . . 31
Example of Creating a New Concept . . . 32
Example of Creating Synonyms in a Concept . . . 36
Example of Creating a Synonym Set (synset) . . . 38
Spell Check List . . . 41
Operating Modes . . . 41
How Spell Checking Works . . . 42
Learning from the Ontology . . . 42
Contextual Spell-Checking . . . 42
Learning from the Content . . . 42
User-defined Corrections . . . 42
Additional Features . . . 43
Modifying a Spell Check List . . . 43
Clusters . . . 45
Create a new Cluster . . . 45
Example of Creating Clusters . . . 46
Intents, Intent Hierarchies, Intent Responses . . . 49
Intent Responses and Managed Answers . . . 49
Create and Assign Intent Responses . . . 50
Intent Response Tools . . . 54
Validate Response . . . 54
Reset IM Document Cache . . . 55
Create a New Intent in the Intent Hierarchy . . . 56
Define Concept Parameters . . . 59
Intent Scores . . . 59
Intent Tools . . . 60

Check IML . . . 60
Validate Intent . . . 61
Example of Using an Intent and Creating a New Intent Response . . . 61
Adding an additional language Intent Response . . . 64
Rules . . . 66
Create a New Rule- Managed Answer Rule (default Custom Content) . . . 67
Create a New Rule- Search Component Rule . . . 71
Rule Tools . . . 77
Match Against a Question . . . 77
Check IML . . . 78
Validate Rule . . . 79
Example of Creating a Managed Answer . . . 79
Example of Creating a Search Component Booster Rule . . . 81
Types of Booster Rules . . . 81
Booster Rules Issue . . . 83
Example . . . 84
Domains and Domain Lists . . . 88
Concept Domains . . . 88
Non-Concept Domains . . . 88
Create a new Domain . . . 89
Delete a Domain . . . 90
Domain Lists . . . 91
Create a new Domain List . . . 91
Active Domain List . . . 93
Finding Dictionary Objects . . . 94
Displaying Dictionary Search Results . . . 95
Adjust Objects from the Dictionary Search Results . . . 96
Commit Dictionary Changes to Central Repository (vcroot) . . . 96
Resolving Dictionary Conflicts, Commit Failed . . . 99
Reviewing Dictionary Data Conflicts . . . 100
Update Dictionary from Central Repository . . . 101
Validate Dictionary . . . 102
Reviewing Processed Workbench Modifications . . . 103
Export and Import Dictionary Objects, Ontology Objects, Process Wizards, and Taxonomies 105
Exporting Dictionary Objects . . . 105
Importing Dictionary Objects . . . 105
Importing Ontology Objects . . . 106
Exporting Ontology Objects . . . 109
Clusters . . . 110
Synonyms . . . 110
Exporting Process Wizards . . . 110
Importing Process Wizards . . . 111
Exporting Taxonomies . . . 111
Importing Taxonomies . . . 111

Chapter 3 Test Drive . . . **112**

Using Test Drive . . . 112
Rules Engine Tab . . . 114

Annotations Tab 116
Concept Synonyms 117
Label 118
No Concept Identified 119
Skip Lists 119
#SKIP_INDEX 119
Adding or Removing Tokens in #SKIP_INDEX 119
Updating Installed Search Component Rules Affected by #SKIP_INDEX Logic 120
Maintaining Custom Search Component Rules Affected by #SKIP_INDEX Logic 120
Search Impact 120
Part of Speech 121
Phrase 122
Search Component 123
Stem 124
Unstructured Results Tab 125
Structured Retrieval Tab 126
All Messages 127
XML Results 127
Parameter Element 128
Responses Element 128
Facets Element 128
Constraint Element 128
Query Element 128
Configuration Element 129

Chapter 4 Quality Monitor. **130**

Using Quality Monitor 131
Question Sets 132
Add Question Set 132
Import / Export Questions 133
Creating a Question List to import 133
Import Question Set 134
Import Intent Questions 134
Import Questions and Grades 134
Export Questions and Grades 134
Add a New Question Manually 135
Test Profiles 135
Create a New Test Profile 136
Delete a Test Profile 137
Test a Question Set 137
Grade a Question Set 137
Grading Guidelines 139
Analyze a Question Set 139
Compare Test Results 139
Test History 141
Run Batch Jobs 142

Appendix A Supplementary Tables **143**

Purpose (portlet) 143
IML Basics 143
IML Color Code 144
Search Criteria Operators 144
Specifying General Criteria 145
Specifying Concept-Specific Criteria 146
Specifying Rule-Specific Criteria 146
Specifying Intent or Intent Hierarchy-Specific Criteria 147
Specifying Intent Response-Specific Criteria 147
Dictionary Validation Tests 148
For Alias Lists, the Dictionary Validation Tool tests: 148
For Concepts, the Dictionary Validation Tool tests: 148
For Domain Lists, the Dictionary Validation Tool tests: 148
For Intents, the Dictionary Validation Tool tests: 148
For Intent Hierarchies, the Dictionary Validation Tool tests: 148
For Intent Responses, the Dictionary Validation Tool tests: 149
For Response Templates, the Dictionary Validation Tool tests: 149
For Rules, the Dictionary Validation Tool tests the syntax: 149
For Process Wizards, the Dictionary Validation Tool tests: 150
Language Analysis Labels 150
Part of Speech Annotations 151
Phrase Annotations 152

About This Guide

The Oracle Knowledge Language Workbench is the Oracle Knowledge application used to ensure Oracle Knowledge Intelligent Search continuously returns the best possible answers to a user query as your company's needs and content change.

The Language Tuning Guide reviews the following tools within Oracle Knowledge Language Workbench used to tune the Dictionary:

- Dictionary Manager
- Test Drive
- Quality Monitor

For information on additional tools and features of the Language Workbench, please see *Oracle Knowledge Intelligent Search Optimization Guide*.

In This Guide

The *Oracle Knowledge Language Tuning Guide* is divided into the following sections:

Chapter 1, The Language Workbench	Describes the Oracle Knowledge Language Workbench that contains the tools used to tune your search results.
Chapter 2, Dictionary Manager	Describes the Oracle Knowledge Dictionary Manager accessed from within the Language Workbench that allows you to customize and maintain the Dictionary Objects used by Oracle Knowledge Intelligent Search.
Chapter 3, Test Drive	Describes the Test Drive component accessed from within the Language Workbench that allows you to test language processing and the effects of changes made in Dictionary Manager prior to committing to the Central Repository and synchronizing.
Chapter 4, Quality Monitor	Describes the Quality Monitor accessed from within the Language Workbench that allows you to evaluate and monitor the quality of search results. You can use Quality Monitor and the Grading Tool to analyze unsatisfactory search results, identify problems with Dictionary Objects, gaps in your company's content, and tune Oracle Knowledge Intelligent search accordingly.
Chapter A, Supplementary Tables	Contains the supplementary tables and information to the previous chapters

Examples of Product Screens and Text

The product screens, screen text, and file contents depicted in the documentation are examples. We attempt to convey the product's appearance and functionality as accurately as possible; however, the actual product contents and displays may differ from the published examples.

Operating System Variations in Examples and Procedures

We generally use Linux screen displays and naming conventions in our examples and procedures. We include other operating system-specific procedures or steps as noted in section headings, or within topics, as appropriate.

We present command syntax, program output, and screen displays:

- in Linux format first
- in other Unix-specific variants only when necessary for proper operation or to clarify functional differences
- in Windows format only when necessary for clarity

References to Web Content

For your convenience, this guide refers to Uniform Resource Locators (URLs) for resources published on the World Wide Web, when appropriate. We attempt to provide accurate information; however, these resources are controlled by their respective owners and are therefore subject to change at any time.

CHAPTER 1

The Language Workbench

The Oracle Knowledge Language Workbench is the Oracle Knowledge application used to ensure Oracle Knowledge Intelligent Search continuously returns the best possible answers to a user query as your company's needs and content change.

The chapter describes the components of the Language Workbench:

- *Dictionary Manager* on page 6
- *Personalized Navigation* on page 7
- *Test Drive* on page 8
- *Quality Monitor* on page 9
- *User Manager* on page 9

In addition, it provides information on:

- Understanding Keyword Search in *Keyword Search* on page 10.
- Enabling Locales for Search in *Language Model For Global Customers* on page 14.

For the purpose of language tuning we focus on key aspects of *Dictionary Manager*, *Test Drive*, and *Quality Monitor* in this guide. For more information on other Oracle Knowledge Language Workbench components please see *Oracle Knowledge Intelligent Search Optimization Guide*.

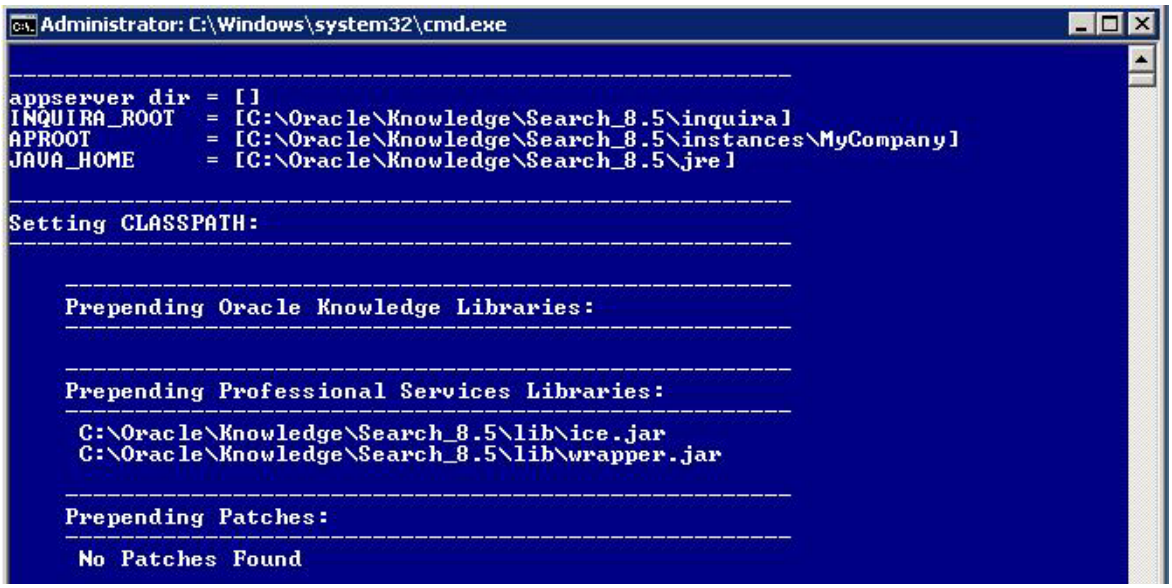
Opening Oracle Knowledge Language Workbench

To access Oracle Knowledge Language Workbench:

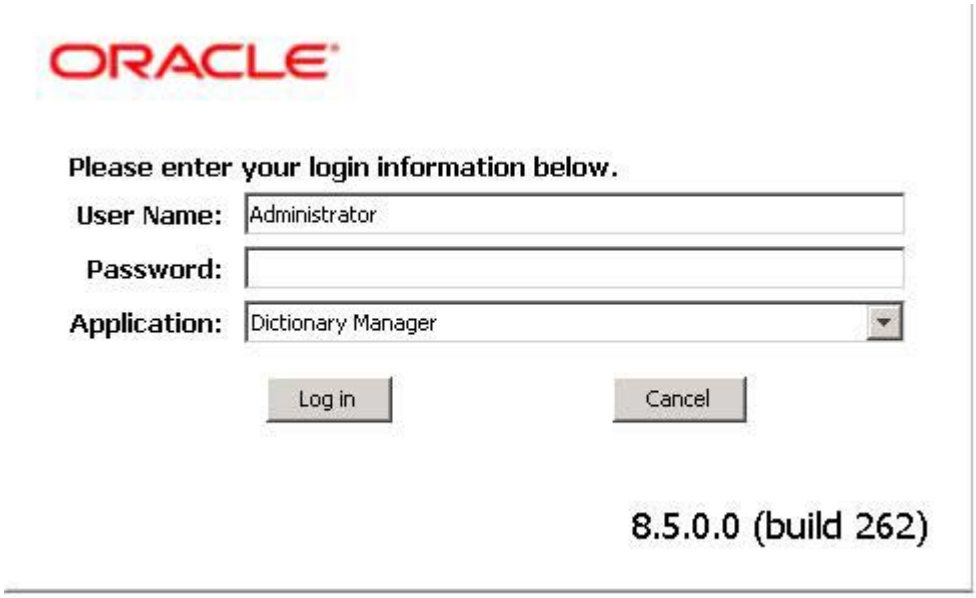
1. From the development environment, open the InQuira Common Environment.
   In Windows select: Start > InQuira 8.1 > MLD Development > Indexing Environment > ICE
   For Unix/Linux: Start Shell, use "cd" to go to the instance directory, type `sh setenv.sh`

   Important! We recommend that you perform language development only in the content processing instance of the development environment.

2. Enter the following command at the Installation Configuration Environment command prompt: **workbench** (**workbench .sh** for Unix/Linux systems).



The Oracle Knowledge Language Workbench launcher displays:



3 Enter a valid User Name and Password.

The default user name is Administrator. The default user password is Administrator. You will be prompted to change the password before you can continue.

For more information on creating User Names see the *Oracle Knowledge Intelligent Search Optimization Guide*.

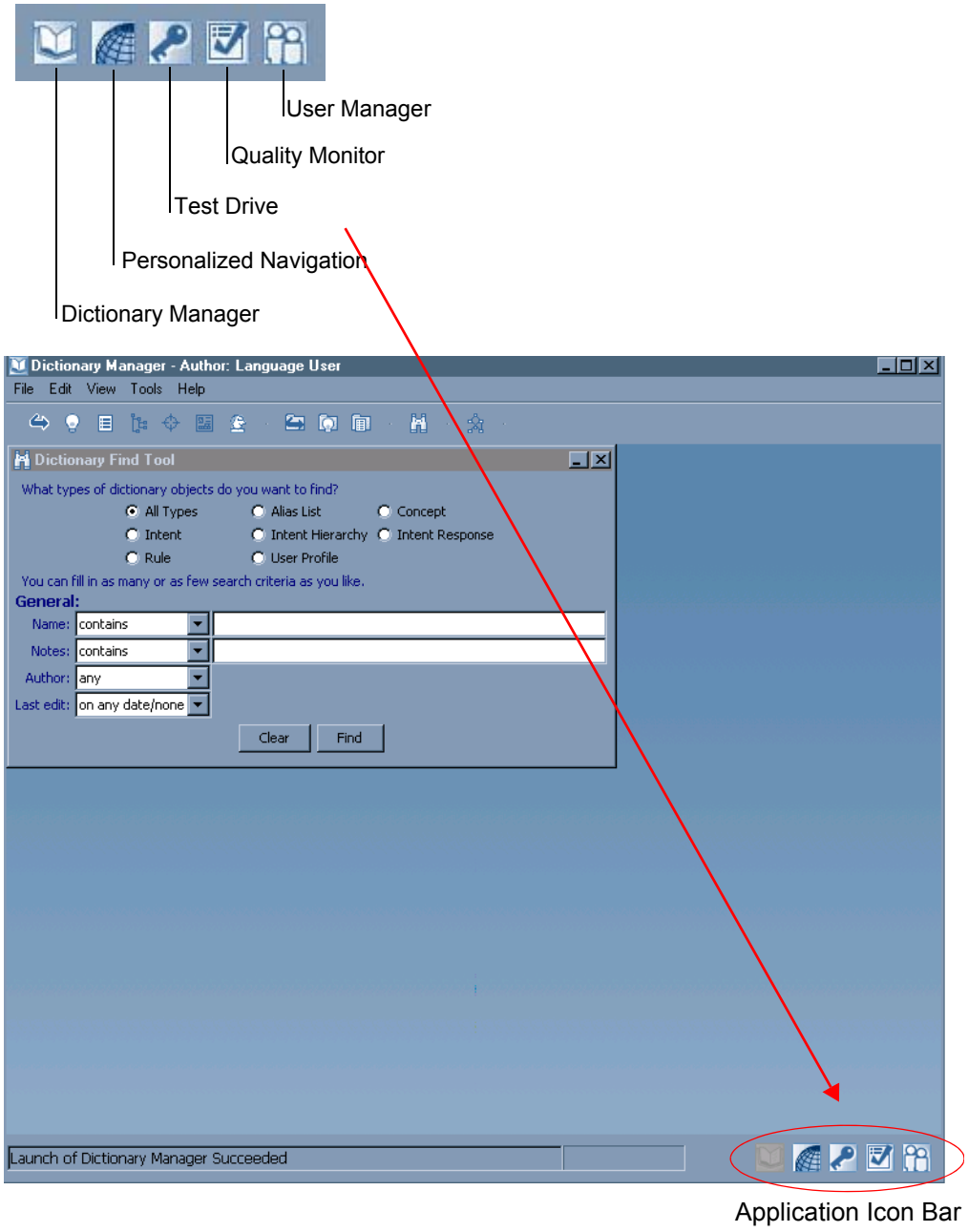
4 Select the application for Oracle Knowledge Language Workbench to open from the drop down menu. The default is Dictionary Manager or the last application started.

Note: All Oracle Knowledge Language Workbench applications are accessible after Oracle Knowledge Language Workbench is opened.

5 Select **Log In**.

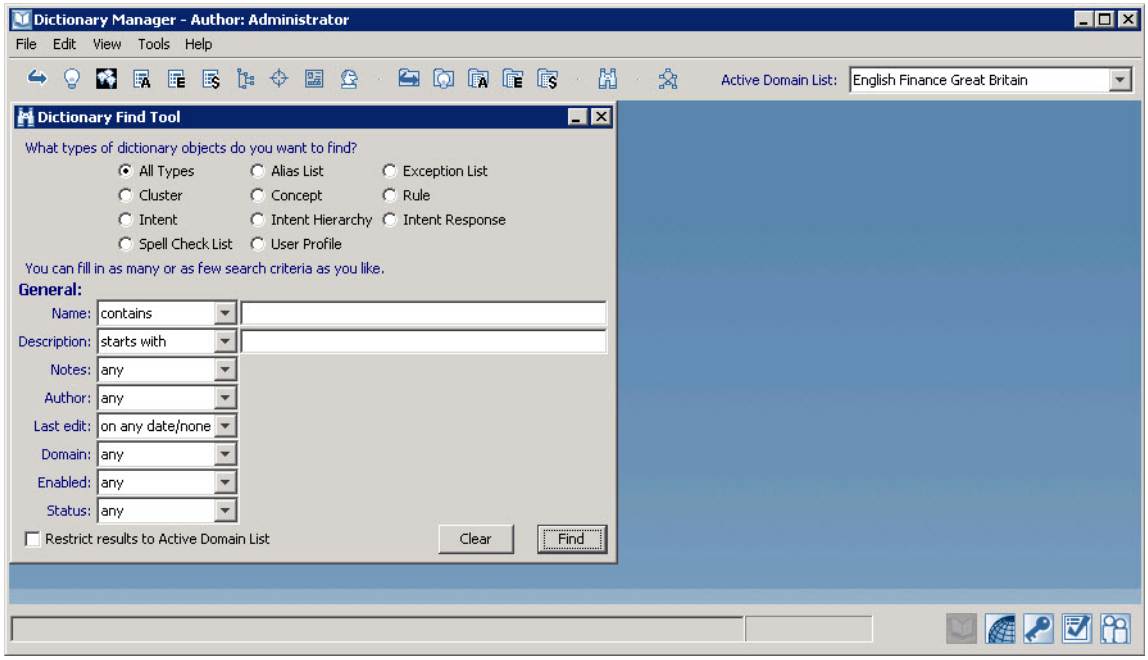
The Language Workbench opens to the selected application.

In the Language Workbench, you use the Application Icon Bar to navigate to other Oracle Knowledge Language Workbench applications.



Dictionary Manager

After opening the Oracle Knowledge Language Workbench to Dictionary Manager, or by selecting the Dictionary Manager icon from inside the application, the following screen displays:

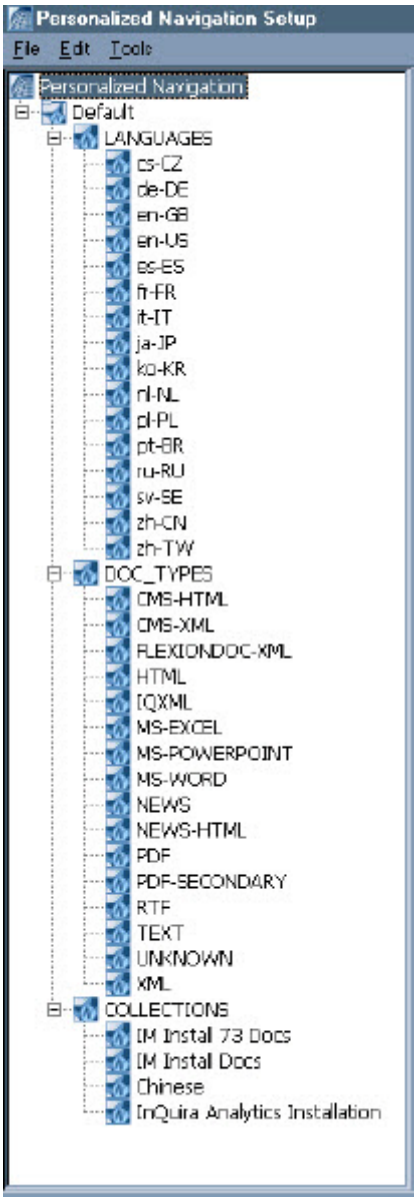


Dictionary Manager is used to create, find, and tune Dictionary Objects such as concepts, rules, and intents that determine what results Oracle Knowledge Intelligent Search returns as the best possible answer to a user query.

For more information see *Dictionary Manager*.

Personalized Navigation

After opening the Oracle Knowledge Language Workbench to Personalized Navigation, or by selecting the Personalized Navigation icon from inside the application, the following screen displays:

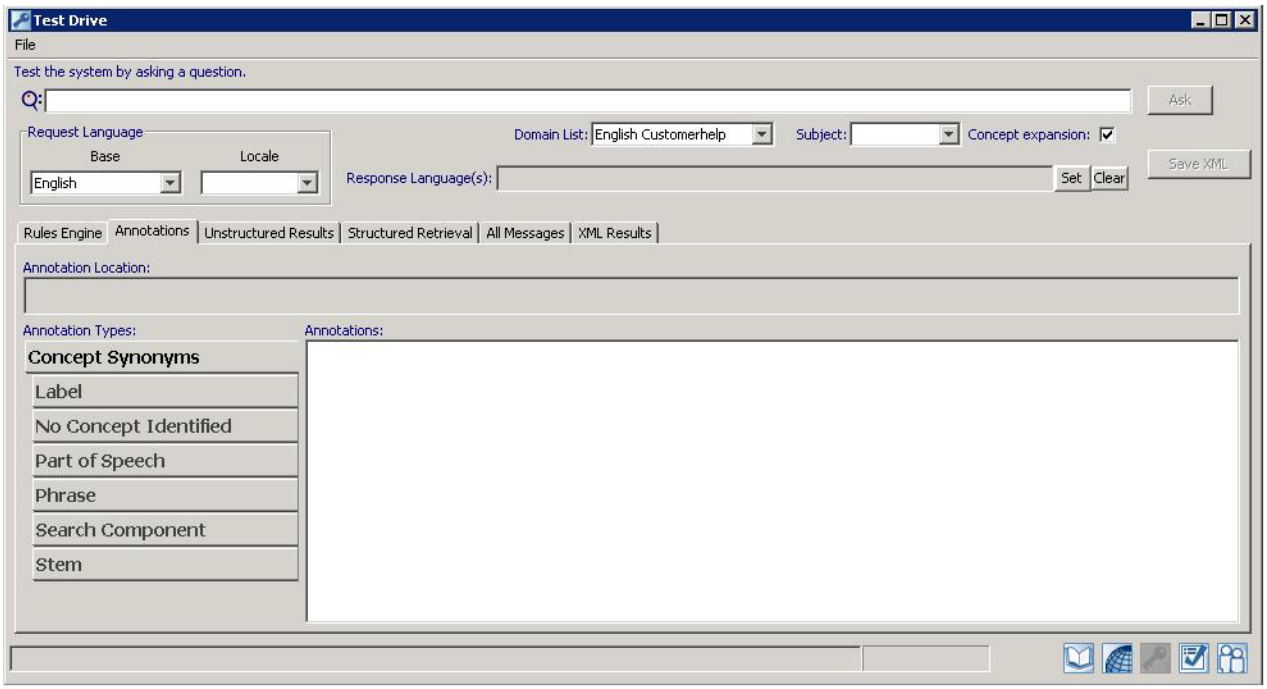


Personalized Navigation is used to define and create taxonomies to classify your content and facets in Oracle Knowledge Intelligent Search results.

For more information see *Oracle Knowledge Intelligent Search Optimization Guide*.

Test Drive

After opening the Oracle Knowledge Language Workbench to Test Drive, or by selecting the Test Drive icon from inside the application, the following screen displays:

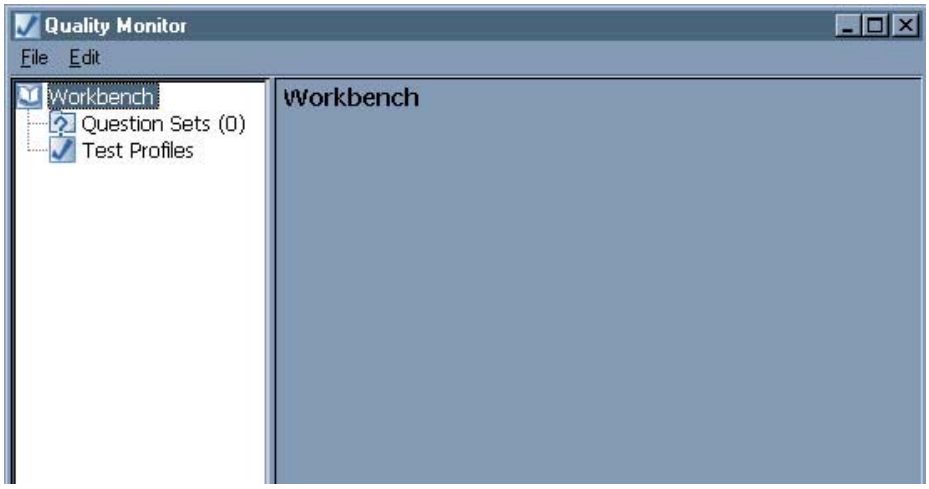


Test Drive is used to test language processing and the effects of changes made in Dictionary Manager prior to affecting the live User Interface. Test Drive provides detailed information about Dictionary Objects and language analysis processing, associated with the response to an individual question.

For more information see *Test Drive*.

Quality Monitor

After opening the Oracle Knowledge Language Workbench to Quality Monitor, or by selecting the Quality Monitor icon from inside the application, the following screen displays:

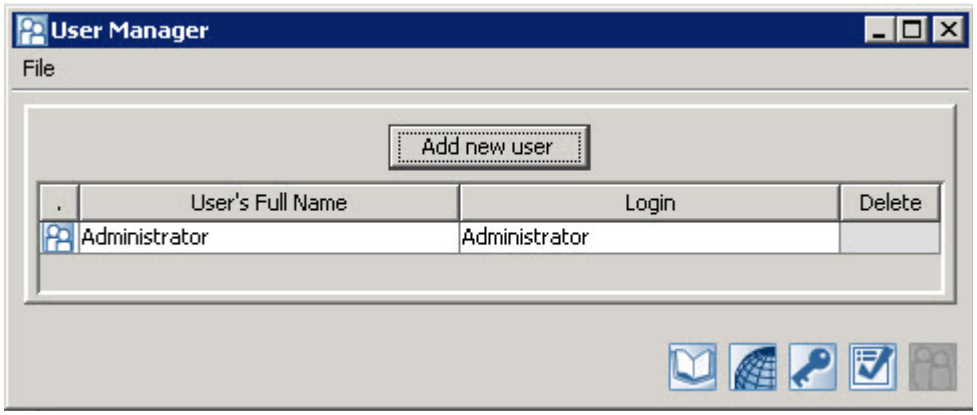


Quality Monitor is used to evaluate and maintain the quality of the answers returned by Oracle Knowledge Intelligent Search.

For more information see *Quality Monitor*.

User Manager

After opening the Oracle Knowledge Language Workbench to User Manager, or by selecting the User Manager icon from inside the application, the following screen displays:



User Manager is used to create and set permissions for Language Workbench users. For more information see *Oracle Knowledge Intelligent Search Optimization Guide*.

Keyword Search

This section provides an overview of Keyword Search functionality and review of benefits and features.

Overview

As Oracle Knowledge supports content authoring in most languages. To support customers with extended language needs, Oracle Knowledge Search supports:

- Natural Language Processing (NLP) Search in many languages (see the *Oracle Knowledge Platform and Language Support Reference* for details).
- Keyword Search for most other languages.
- Ontology and Intent support for NLP Languages and Keyword Languages.

Findability with Keyword Search varies from language to language. Keyword Search provides better user experience with some languages than with others. While the use of ontology and intents can improve findability and user experience, Keyword Search is generally designed to support search for a customer's secondary languages, where non-NLP Search quality can be accepted.

NLP Search vs. Keyword Search Overview

Keyword Search differs from Natural Language Search as follows:

	Natural Language Search	Keyword Search
Translated Industry Terminology	Yes[a]	Can be supplied by customer
Spell-check	Yes[b]	No
Part-of-Speech Analysis	Yes	No
Morphological Analysis	Yes	No
Ontology & Intent Support	Yes	Yes[c,d]
Tokenization	Language-specific Tokenizer	Generic Tokenizer[e]
Language Coverage	Many Languages[f]	Most Languages[g]

a. Oracle Knowledge provides terminology translations for all major languages in the priority industries. For other languages and industries, coverage may be partial or can be supplied by the customer as needed.
b. Provided for most NLP languages.
c. Global terms increase Keyword Search findability automatically.
d. Customer can increase findability and configure intents by providing industry and company terminology in the Keyword Search Language.
e. Except for Thai Keyword Search where a Language-specific Tokenizer is provided.
f. See the *Oracle Knowledge Intelligent Search Administration Guide* for language coverage details.
g. See the following sections for limitations.

Benefits and Features

The following sections discuss the various benefits and features of Keyword Search.

Easy Language Management

A Language Management Interface enables customers to easily enable a pre-configured Language for Keyword Search, or to add a new Language.

String Match

Keyword Search will match the exact strings as provided in the question. For instance, Keyword Search for υπολογιστή (Greek: 'computer') will match υπολογιστή but not inflected forms such as υπολογιστές ('computers').

Word Separation

Keyword Search uses a Generic Tokenizer, which tokenizes on white space. Strings between white spaces are therefore the atomic units for Keyword Search. For instance Keyword Search for αποτυχία υπολογιστή (Greek: 'computer failure') identifies αποτυχία and υπολογιστή as tokens and will retrieve content that contains either αποτυχία ('failure') or υπολογιστή ('computer') or both.

Non-white-space Languages

Some writing systems, such as e.g. Chinese, Japanese, Thai, Lao, and others, do not separate words by spaces. Oracle Knowledge Search provides Language-specific Tokenizers for Chinese, Japanese and Thai, which help identify words despite the lack of white-space separation.

As for Lao and other non-white-space languages, enabling Keyword Search is not recommended: Keyword Search for a Lao word will only match content where that string occurs in isolation; occurrences of that word as part of a sentence or phrase without white-spaces will not be found. Hence findability may be unsatisfactory.

Industry Terminology

While industry terminology is not provided upon installation for Keyword Search languages, they can be supplied by the customer as concept synonyms in order to increase findability and search success.

Global Terms and Phrases Work Automatically

Terms and phrases defined to apply globally will work for Keyword Search Languages automatically. For instance, the product name synonyms ‘bmw 323’ and ‘323i’ will automatically increase findability and search success for most Keyword Languages if defined as global synonyms.

Global Synonyms are synonyms that apply across languages, such as product names. For instance the model name Toyota Yaris is used across regions. Such cross-language terminology can be maintained in one location as Global Terms and automatically apply to all languages and locales.Global Synonyms are defined in the **Any** synset for a concept.

Automatic or Manual Language Detection

INFORMATION MANAGER CONTENT

Keyword Search automatically identifies the Language of content maintained in Information Manager from the content's Locale.

NON-INFORMATION MANAGER CONTENT (E.G. WEB-CONTENT)

- Manual Detection: A non-Information Manager Content Collection can be manually assigned to a specific Keyword Search Language.

- Automatic Detection: Automatic Language Detection is supported through the Basis Rosette Language Identifier (RLI), which covers over 50 languages (see Basis documentation for actual coverage).

Encoding

The Language must be UTF-8 encoded.

Findability

Findability and search success with Keyword Search will vary from question to question and from language to language. Keyword Search is designed primarily for a customer's secondary languages, where non-NLP Search quality can be accepted.

Findability can be increased by:

- Adding terms in the Keyword Language to the ontology.
- Adding terms that are applicable globally as global synonyms to the ontology.

Configuration

See the *Oracle Knowledge Platform and Language Support Reference* to determine which languages are supported with NLP Search and which languages are supported by Keyword Search. Languages can be enabled with Keyword Search within the limitations described herein.

Important! **Enabling a language cannot be undone.** Only enable languages that are needed.

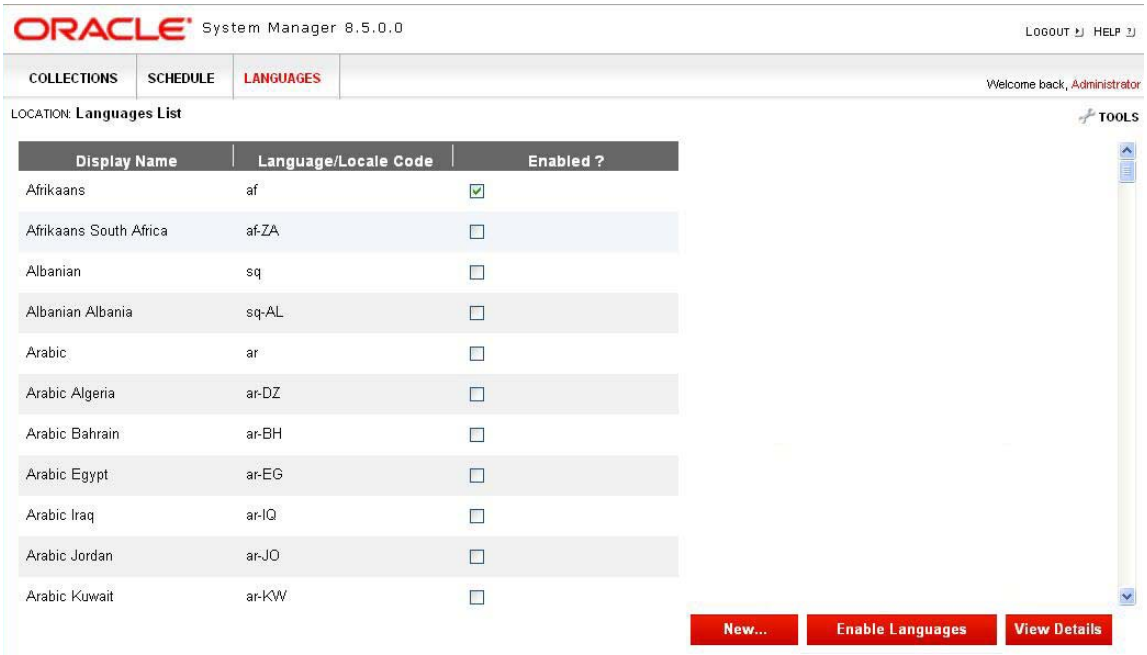
Important! **The Language configurations in Search and Information Manager must match**. When enabling a Language for Search in combination with Information Manager content, the configuration in Search must match the configuration in Information Manager. As Information Manager is typically locale-based, a locale may have to be enabled in addition to the language. For instance, after enabling the language Romanian, the Locale "Romanian Romania" may have to be enabled as well in order to match the Information Manager configuration.

Important! **Define Languages in the Development Environment.** As a general practice any configuration changes, including Language configuration, shall be performed in the Development Environment. Any configuration at a destination environment, such as the Staging or Production Environments, will be overwritten during propagation. For example, if you are propagating from development to staging, all the languages in the staging environment will be overwritten by the languages in the development environment.

Enabling a pre-configured Language or Locale for Keyword Search

1. Open the "Languages" tab in System Manager.
2. Select the language(s) you want to enable by selecting the **Enabled?** checkbox.
3. Click **Enable Languages** (Caution: this step cannot be undone).

   Note: Language properties are inherited from the Base Language to its Locales unless overridden in the Locales.



The language is now configured for Search: Crawl content in that language and it will return in Search results.

Note: The new language is available in the Language Workbench under Synonym Sets but does not appear in the Domain List and Subject Map until a synonym has been defined. See *Add Synonym Set (synset)* on page 27 for more information.

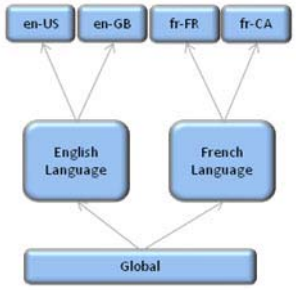
Adding a Language or Locale for Keyword Search

In case the desired language is not preconfigured in the “Languages” tab in System Manager, the language can be added to the configuration:

1. Open the “Languages” tab in System Manager.
2. Verify that the desired language is not preconfigured in the “Languages” tab.
3. Click **New**.
4. Provide the Language Display Name and the Language Code.
5. Click **Save**.

Language Model For Global Customers

Oracle Knowledge features an extended Language Model with 3 levels: Global, Language and Locale:



For example: French France “fr-FR” and French Canada “fr-CA” are Locales of the Language French, differentiated by region- or country-specific variations. Analogously, English United States “en-US” and English Great Britain “en-GB” are Locales of the Language English. While the Locale level hosts country and region specific variations, the Global level captures terminology that is shared across languages, most typically product names.

This model improves user experience by:

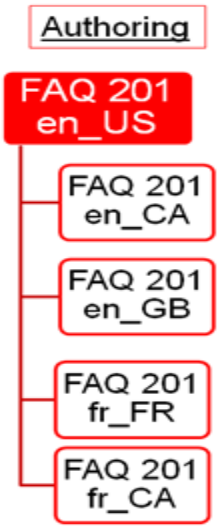
- Reducing maintenance as terminology (concepts) and other linguistic knowledge is automatically inherited to the next, more specialized level.
- Using Simplified Business Modeling as global terminology that apply across languages, such as Product Names, can be maintained centrally in one location. They automatically apply to all Languages and Locales.
- Increasing Search Accuracy by adding region- and country-specific terms, phrases, product names at the Locale level.
- Allowing decentralized, country-level dictionary tuning.

Locales In Content Authoring

From the content authoring perspective, the term “Locale” refers to regionalized content, differentiated across countries and regions by:

- Products and product names
- Policy and procedure
- Government regulations (e.g. IRS vs. Inland Revenue)
- Terminology
- Time/date/number format

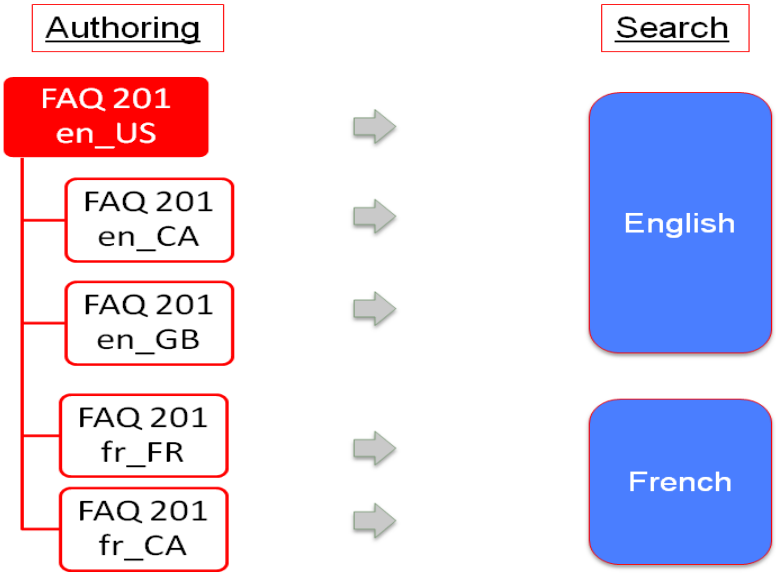
Content can be regionally adapted to Locales by the translation workflow within the Authoring Console. For instance, English content can be adapted to the Locales “English United States”, “English Great Britain”, as well as “French France”, “French Canada”, etc.



Automatic Language-to-Locale Mapping

In Search, a Locale automatically inherits the Linguistic Knowledge from the Language Level: For instance, en-US, en-GB, en-CA inherit the Linguistic Knowledge from English; French provides the Linguistic Knowledge for fr-FR, fr-CA, etc.

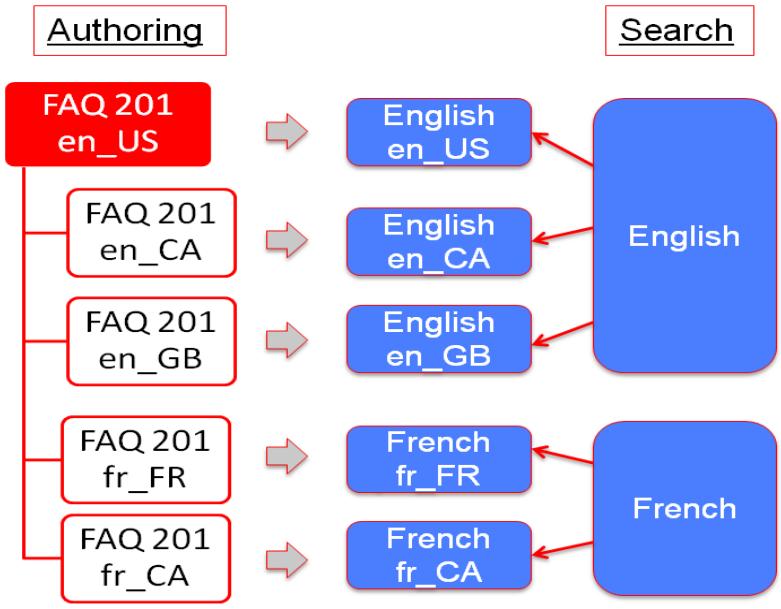
The inherited Linguistic Knowledge includes the entire linguistic spectrum as defined for that language: from Tokenization, Morphological Analysis, Part-of-Speech Analysis, Spell-check, to Industry Terminology Translations in the Ontology.



Optional Region/Country-specific Language Modeling

Search relevancy can be enhanced by adding regional/country-specific terms, phrases and product names at the Locale-level.

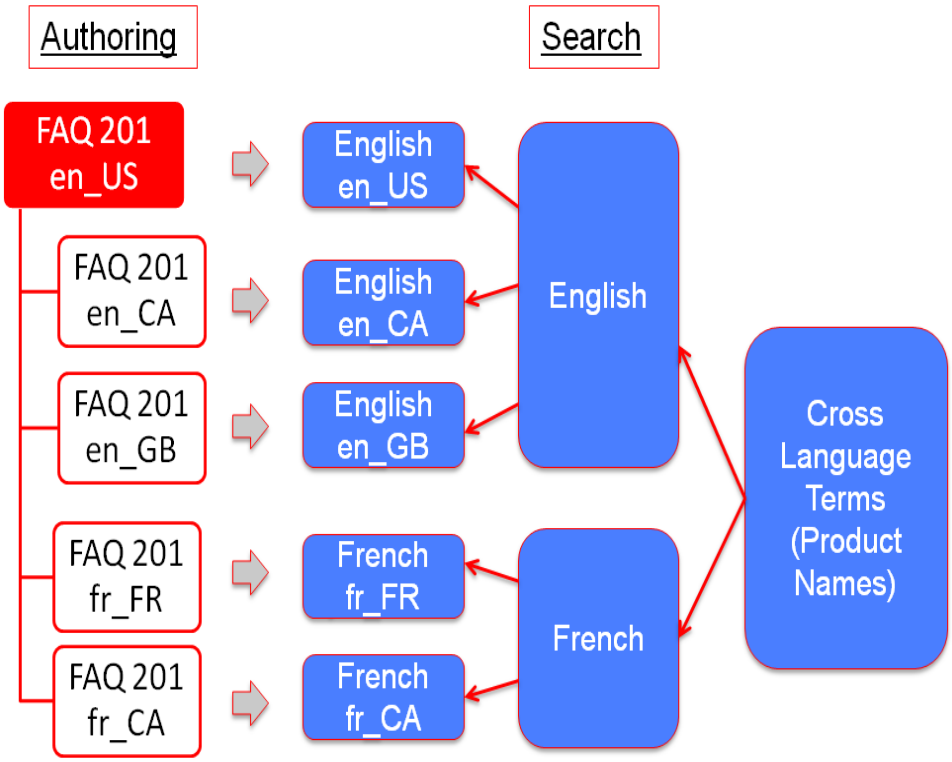
For instance, regional UK terminology such as ‘boot’ (for trunk), ‘accumulator’ (for battery), ‘saloon’ (for sedan), etc, for Automotive, has its place at the Locale-Level of en-GB.



The following illustrates the concept ‘car battery’ with ‘accumulator’ being added as an “en-GB” concept synonym: While the generic synonym ‘car battery’ applies across the English Language and is inherited by all English Language Locales, ‘accumulator’ is specific to en-GB.

Global Terms

Global terms that apply across languages, typically product, feature or model names such as 'bmw 323i', can be maintained centrally in one location. They will automatically apply to all Languages and Locales enabled for Search:



AUTOMATIC LANGUAGE CONFIGURATION

A Locale in Search inherits its linguistic knowledge from the Language level. Therefore every Locale must be associated with a Language. Oracle Knowledge assures that this is the case automatically. For instance, as the user enables e.g. the Locale "Romanian Romania", the system verifies if the Language "Romanian" is enabled, and if not, will enable it automatically.

The Inheritance Model

When a new language or locale is enabled, it is available for use in Dictionary Manager, Personalized Navigation and Content Processing.

Locales inherit from their base languages by the principles below.

Component	Inheritance Principle
Tokenization	inherit base tokenization
Spell Checking	inherit base spell checking; inherit and override base spell check lists
Stemming	inherit base stemming; inherit and override base exception list
Synonym Matching	merge all synonyms in ANY, base and locale
Domain List	default to base domain list if undefined; override base domain list if defined

Component (Continued)	Inheritance Principle (Continued)
Intent Response	default to base intent response if undefined; override base intent response if defined
Language Based Label For Facets	default to base label if undefined; override base label if defined
Content Inheritance	merge base content and locale-specific content

Note: When request language and result language are different, intent responses are determined by request language only.

CHAPTER 2

Dictionary Manager

The Oracle Knowledge Dictionary Manager is a component accessed from within the Language Workbench that allows you to customize and maintain the Dictionary Objects used by Oracle Knowledge Intelligent Search to return the best possible answers. To access Dictionary Manager see *Dictionary Manager* on page 6.

This chapter describes the following Dictionary Objects:

- *Concepts* on page 19.
- *Clusters* on page 39.
- *Intents, Intent Hierarchies, Intent Responses* on page 44.
- *Rules* on page 61.
- *Domains and Domain Lists* on page 83
- *Spell Check List* on page 89

This chapter also explains:

- *Finding Dictionary Objects* on page 95.
- *Commit Dictionary Changes to Central Repository (vcroot)* on page 97.
- *Update Dictionary from Central Repository* on page 102.
- *Validate Dictionary* on page 103.

For more information on additional features of the Oracle Knowledge Dictionary Manager not included in this book, please see *Oracle Knowledge Intelligent Search Optimization Guide*.

Concepts

Concepts are Dictionary Objects that define meanings of words and phrases that occur both in user queries and in your company's crawled content. Concepts are unique, symbolic, abstract representations of meaning. By creating a list of synonyms, a Concept can represent multiple words and phrases that have the same meaning within the context of your company’s business.

Creating Concepts in Dictionary Manager allows Oracle Knowledge Intelligent Search to:

- Treat multiple words and phrases with the same meaning, regardless of language type, as instances of a single Concept.
- Create relationships and clusters of ideas that expand, contract, or create associations between other words and phrases that have been created as Concepts.
- Establish different levels of importance with respect to search. Concepts can be set to high, mid, or low, indicating how relevant they are for the purposes of returning search results.

When a user query is analyzed, Oracle Knowledge Intelligent Search looks at each word in the query and determines whether it matches any synonyms or Clusters of the Concept in the Dictionary. (Clusters are

discussed in the following section *Clusters* on page 39). If there is a match, Intelligent Search annotates each word or phrase with the associated Concept(s). Similarly, Intelligent Search annotates your company's crawled content with the appropriate Concepts. Matches on the exact string of the Concept are considered the strongest, followed by alternative forms of the word, and then associated synonyms, for example:

- A noun matches on the corresponding plural form. Manager->managers
- A verb matches on the corresponding conjugated form. Install->installed, installs, installing
- An adjective matches on the corresponding comparative and superlative form. Low->lower, lowest

Note: Matches on Concepts that are connected as Clusters are also considered, but with less scoring weight.

For example, a search on "dictionary" first matches on "dictionary", followed by any variations of the word, such as; "dictionaries." After these, Oracle Knowledge Intelligent Search matches on any synonyms listed to the Concept "dictionary" such as "lexicon," "lexicons" "glossary," or "glossaries." Finally, any Clusters or relationships that have been established match. These may include, for example, "ontology", "concept," "dictionary manager."

In addition to determining how an unstructured search is conducted, Concepts may also be used to write Rules and Intents.

For information on the features of Concepts listed under the Advanced menu, please see *Oracle Knowledge Intelligent Search Optimization Guide*.

Role of Concepts in Unstructured Search Results

A Concept has three features which play a role in evaluating potential search results in the indexed content:

- Concept Level
- Synonyms
- Clusters (discussed in *Clusters* on page 39.)

For example, suppose the user query is "closing the dictionary manager." this query may be annotated with the Concepts <close> and <dictionary manager>. (Concepts are indicated by angle brackets.) The query is not annotated with <manager>, as the complete phrase is considered a better match.

These annotated Concepts may have the following features:

```
<close>
Concept level: low
Synonyms: close, close down, shut
Clustered with: <switch off> (synonyms: switch off, deactivate)

<dictionary manager>
Concept level: high
Synonyms: dictionary manager
Clustered with: <workbench> (synonyms: workbench)
```

When Oracle Knowledge Intelligent Search evaluates a document to determine how well it matches a user query, these are two of the criteria used:

Note: The following assume the effects of the criteria if all other scoring procedures are equal.

1. How close the word or phrase in the content matches the annotated Concept from the user query:
   a. Best match: The content is annotated with the same Concept, and has the same form. For example, "closing" on "closing."

b Next best: The content is annotated with the same Concept, but has an alternative form. For example, “closing” matches on “closes.”

c Next best: The content is annotated with the same Concept, but uses an associated synonym. For example, “closing” matches on “shut” or “shutting.”

d Next best: The content is annotated with a Concept that is clustered with a Concept annotated in the query. For example, “dictionary” matches on “ontology” which is annotated with the Concept <ontology>, clustered with the Concept <dictionary>.

e Next best: A literal match between word forms which are not annotated with related concepts. For example matching “dictionary” from a user query of “dictionary manager” with “dictionary” in a piece of content such as “dictionary objects.”

2 The weight of each Concept match (high, medium, or low).

- First result: “How to close the dictionary manager”

  “closing” has a low level Concept match on “close” as described in 1b.

  “dictionary manager” has a high level Concept match on “dictionary manager” as described in 1a.

- Second result: “Closing the workbench”

  “closing” has a low level Concept match on “closing” as described in 1a.

  “dictionary manager” has a high level Concept match on “workbench” as described in 1d.

- Third result: “Deactivating a workbench account”

  “closing” has a low level Concept match on “deactivating” as described in 1d.

  “dictionary manager” has a high level Concept match on “workbench” as described in 1d.

- Fourth result: “Setting up the dictionary manager”

  “closing” has no match.

  “dictionary manager” has a high level Concept match on “dictionary manager” as described in 1a.

- Fifth result: “Deleting closed and ignored tasks”

  “closing” has a low level Concept match on “closed” as described in 1b.

  “dictionary manager” has no match.

The ordering of these results reflects the overall closeness of the match between query and content. For example, the first result scores higher than the second result because it has the best match on :dictionary manager:, which is annotated with a high level Concept; whereas the second result matches on “closing”, a low level Concept. The fourth and fifth results are ranked last because they each only have one match, but the fourth result is ranked higher because the match is on the Concept with the higher level, <dictionary manager>. The net effect is to produce an ordering which corresponds to actual relevancy of the result to the user’s query.

In addition to determining how an unstructured search is conducted, Concepts may also be used to write Rules and Intents. See *Role of Concepts in Rules and Intents*.

Adjusting the Importance Level of Concepts

If you are not getting the desired best match and you have ruled out other language tuning issues, try downgrading the importance level of a high concept to medium to see if that helps resolve the issue. Sometimes decreasing the importance level of a concept improves search accuracy.

The three importance levels that are assigned to concepts—high, medium, and low contribute to the overall score that a document receives by being matched to the Search Component rules *RequiredConcept SC*, *VeryImportantConcept SC*, and *ImportantConcept SC*, respectively.

These rules contribute the following points when they match:

Importance Level	Points
High	100,000
Medium	10,000
Low	1,000

Since a high level concept adds substantial points to the overall score of the document, it is important to assign this level only to concepts that are likely to produce meaningful matches to documents from the search perspective. Concepts that appear in many documents without discriminating the content of the document need to be assigned to lower levels of importance. For example, the name of a company, which may appear in every document without adding anything to the content of the document or identifying it as relevant.

For this reason, sometimes search results are improved when a concept is moved from high down to medium, or medium down to low. This happens when a good match is overtaken by less desirable matches that takes place at a higher level.

Role of Concepts in Rules and Intents

In addition to determining how an unstructured search is conducted, Concepts may also be used to write Rules and Intents that match on user queries and provide some specific information in addition to the unstructured search results. Concepts have three features which can be leveraged for this purpose:

- Synonyms
- Cluster relationships
- “Type of” relationships, also referred to as a parent/child relationship.

For example, the Information Manager is an Oracle Knowledge product; this relationship can be represented by making the Concept <information manager> a type of <oracle product>. We may also want to cluster <information manager> with the Concept <information center>, as they are related ideas.

Assuming this Concept hierarchy, <information manager> may have the following features relevant to writing Rules and Intents:

```
<information manager>
Synonyms: information manager, im
Clustered with: <information center> (synonyms: information manager, ic) in
a Cluster object named [information manager]
Type of: <oracle product>
```

Using the InQuira Matching Language (IML), we can use the following expressions in a Rule or Intent:

```
<information manager>
Matches on all synonyms of the Concept

[information center]
Matches on all synonyms of the Concepts which are members of this Cluster.

CHILDREN(<oracle products>)
Matches on all synonyms of the Concepts which are a type of <oracle product>.
```

Whenever a change is made to the synonyms or relationships of a Concept, any Rules or Intents which reference the Concept change their matching behavior accordingly. For more information on Rules, please see *Rules* on page 61. For Intents, please see *Intents, Intent Hierarchies, Intent Responses* on page 44.

For more information on the InQuira Matching Language, please see *Chapter 1, IML and VIL Reference* in the *Intelligent Search Language Reference*.

Basic Concept Relationships

This section discusses the following basic concept relationships:

- *Parent / Child Relationships*
- *Cluster Relationships*

This section also provides *Additional IML Tips* for working with concept relationships.

PARENT / CHILD RELATIONSHIPS

There are two symmetrical relationship constructs to build parent/child relationships among concepts:

- is a type of / has these types
- is a part of / has these parts

These relationship definitions do not influence default search results by themselves, but can be leveraged in the rule system. For example, you can write a rule with IML referencing `CHILDREN(<concept>)` to match all the direct children of that concept (everything in the **has these types** field) or, you can use `ALLCHILDREN(<concept>)` to match both direct children, all grandchildren, etc.

CLUSTER RELATIONSHIPS

Concepts can also belong to a cluster with other concepts. This can influence default search results if a document contains a clustermate of a concept found in the query, that is considered a valid match. Cluster matches are given less weight than an exact match or a match on a synonym of the same concept. You can reference a cluster in IML with the SIMILAR operator, e.g. `SIMILAR(<concept>)` matches on the synonyms of that concept plus any concept that it's in a cluster with.

ADDITIONAL IML TIPS

- If you reference a concept without the CHILDREN or SIMILAR operators, then the match is on just that concept, not on related concepts.
- Some strings are required to be the sole contents of the query (i.e. they're surrounded by BEGIN/END), while others are not. So IML of:

  `BEGIN <401_k> END"`

  matches on "401k" as a synonym of <noun.finance.401_k>, but not if it is part of a larger query, e.g. "my 401k". Whereas for IML "<noun.finance.401_k>", the synonym of <noun.finance.401_k> does not have further restrictions on it, so this IML matches "401k", but also, "my 401k" and, "I'm writing to ask for more information about the 401k packages you offer."

Create Root Concepts

The Dictionary Manager requires at least one Root Concept to be defined in the application configuration in order to display the Concept tree. You can configure one or more Root Concepts using the Workbench page of the Advanced Configuration Facility.

To add or change a Root Concept in the Advanced Configuration Facility:

1. Select **Workbench** from the System menu of the Advanced Configuration Facility main menu, then click **Edit**

   The Workbench page lists any currently configured Root Concepts. The order of the list does not affect the Concept Tree display.

2. Under Root Concepts:, click **Add New Item**, or select an existing Root Concept to modify.

   The Editing: Workbench > Root Concepts page displays a blank Concept field.

3. Enter the new Root Concept, in the Concept field or, edit an existing concept name.

4. Click **OK** to apply your changes.

5. Click **OK** to exit the Editing page.

6. Click **Save** to save your chnages.

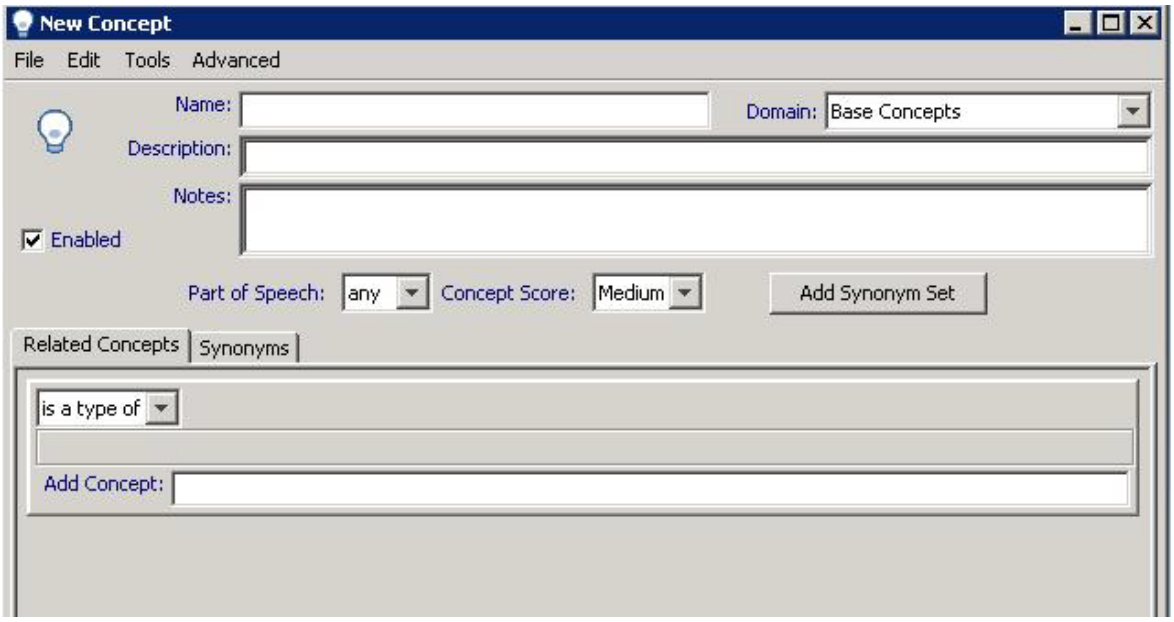
Create New Concept

By creating new Concepts, you allow Oracle Knowledge Intelligent Search to return results that match not only on a particular word but also their synonyms and related Concepts.

To create a new Concept:

1. In Dictionary Manager, select the icon or **New Concept** from the File menu.

   The New Concept window displays:



2. Name the Concept in the **Name** field. You may use any alpha-numeric combination.

   The Concept name is used to reference the Concept in other Dictionary Objects, such as Rules.

Note: The only valid characters allowed in the name field are: letters, digits, underscores, percent signs, single quotes, periods, spaces. You may not use special characters, such as Chinese characters in the **Name** field.

3 (Optional). Enter a definition into the **Description** field.

The **Description** field is used to clarify the meaning among users, and can be used to find dictionary objects.

Note: The Description field may be used as the display form of the Concept, and for that reason it is advised to enter a display name for the Concept into the Meaning field.

4 (Optional). Enter any information regarding the Concept in the **Notes** field.

The **Notes** field has no impact on Oracle Knowledge Intelligent Search results. It is used to track comments on Concept modification among users.

5 Check the **Enabled** box. (This is checked by default.) If the **Enabled** box is not checked, Oracle Knowledge Intelligent Search ignores the Concept.

6 Select a domain from the **Domain List**. For more information, see *Concept Domains* on page 83. The default domain list is the last Domain List selected.

7 Select the **Part of Speech** from the drop down list. The part of speech is used by Oracle Knowledge Intelligent Search in stemming. It is recommended that you select **ANY**, unless you need to reduce the forms of the word synonyms match.

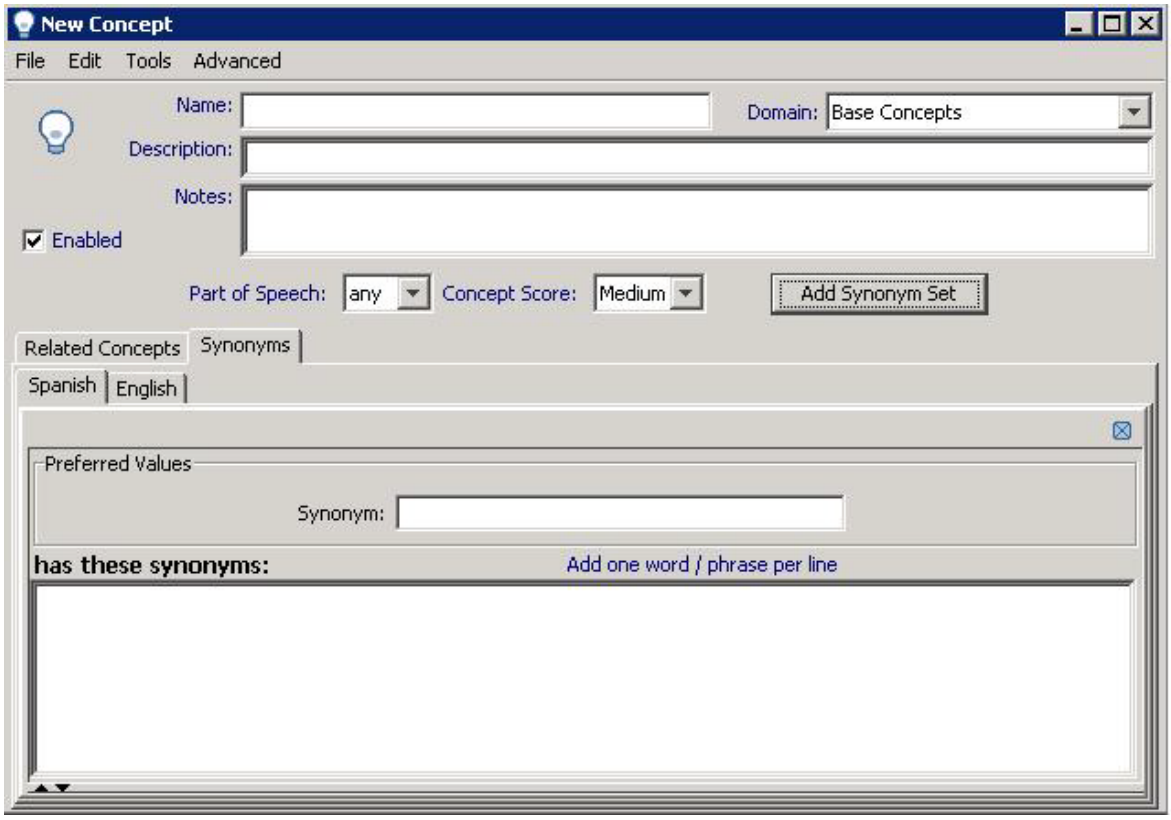
8 Select the **Concept Level** from the drop down menu. The default is the last Concept Level selected.

The Concept Level is used for scoring. High corresponds to 100,000 points or Required, Medium to 10,000 or Very Important, Low to 1,000 or Important.

Concepts that are highly relevant to search results should be set to high. However, Concepts that occur frequently in your company's content such as the company name should be set to low.

Add Synonyms

9 Select the **Synonyms tab**. The Concept window with the default language synonyms tab displays:



a (Optional). Complete the **Preferred Values** fields.

In the **Synonym** field, enter the preferred name of the synonym based on the selected language.

Note: The **Preferred Values: Synonym** overrides the Concept name, and is used in display purposes in the Ontology Navigation. For more information, see the Personalized Navigation Setup in *Oracle Knowledge Intelligent Search Optimization Guide*.

It is recommended that you leave the **Abbreviation** field blank, as it has no impact on Oracle Knowledge Intelligent Search.

10 Enter all the synonyms for the Concept under **has these synonyms**.

Enter one synonym per line. You must enter at least one synonym.

Note: You must include the name of the Concept itself as a synonym, as the Concept name is NOT matched by default.

It is best to enter all Concept synonyms in lowercase and in their unmarked (stemmed) form. Oracle Knowledge automatically matches the marked forms of that word, like plural nouns or conjugated verbs. You may enter special characters, such as Chinese characters, into the **has these synonyms** field.

Note: English and Japanese synonyms (typed in Latin characters) use a specific type of case sensitivity regarding uppercase usage. Lowercase letters in synonyms match on either upper- or lowercase, but uppercase letters match only on uppercase. So, the synonym "EZPay" matches on "EZPay", "EZPAY", "EZPAy", etc., whereas "ezpay" matches on any type of capitalization. This allows

you to specify case matching for short abbreviations that might otherwise get confused with common words, like "OR" for "Oregon".
In other languages, a synonym with a marked form or with capitalization matches on other forms and capitalizations of the word.

Include as synonyms only words that truly have the same meaning. If the words are similar but not the same, it is best to create separate Concept. For example, The Concept <web site> may have the synonyms “web site” and “web page,” but non-synonymous terms such as “home page” and “URL” belong to a different Concept.

We recommend you add common misspellings to the Spell Check List, for more information see *Spell Check List* on page 40.

11 (Optional). Enter IML into **has synonyms matching this IML**.

Use this field to narrow the context in which a concept is annotated. For more information on IML see *Oracle Knowledge Intelligent Search Language Reference*.

Add Synonym Set (synset)

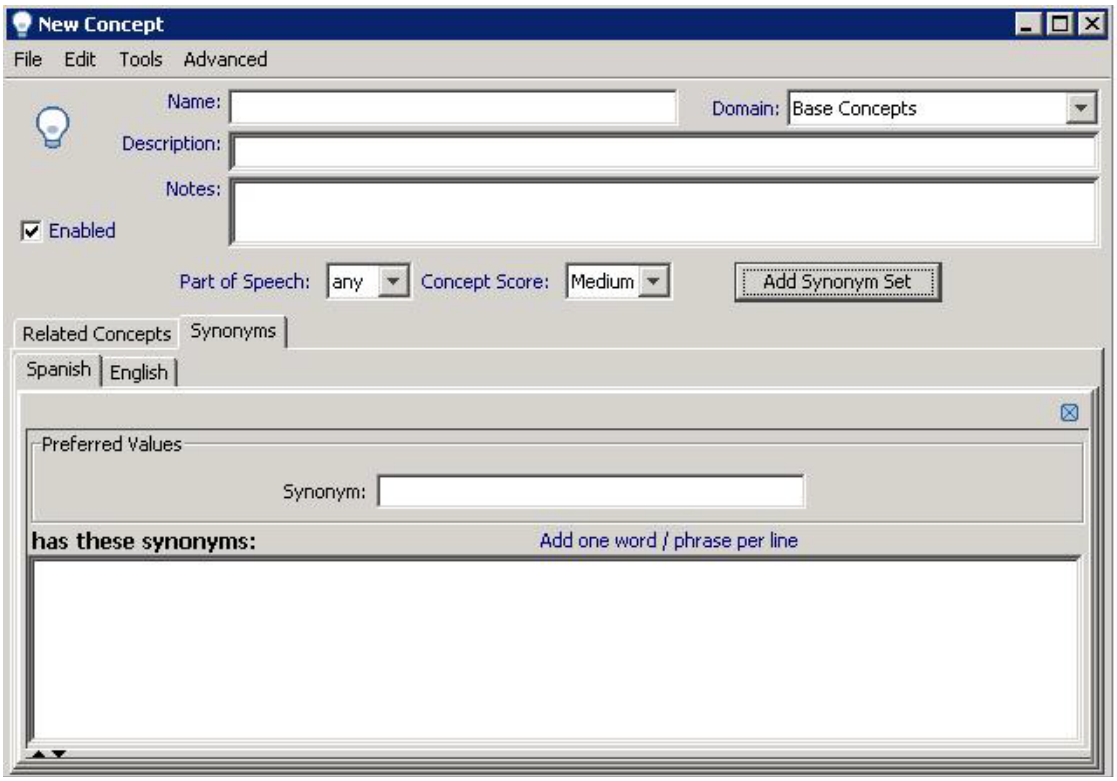
By adding a Synonym Set, you allow Oracle Knowledge Intelligent search to return results of various languages.

12 (Optional.) Select **Add Synonym Set** to associate synonyms of another language with this concept. The Select language of Synonym Set window displays:

13 Select a language from the drop down list.

Note: Select **Any** if the synonym is valid for any language. For example, Product Names, Proper Names, Cities.

14 Select **OK**. An additional language tab displays.

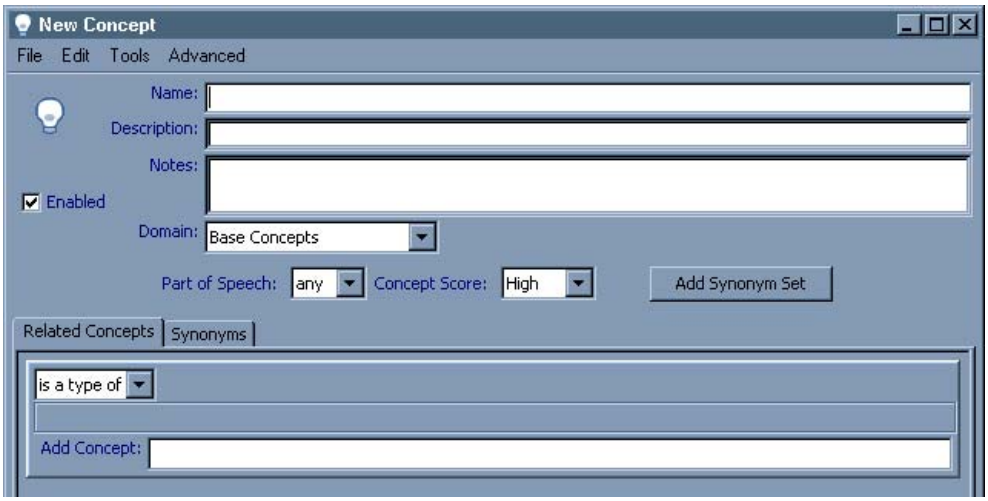


15 Complete the fields as described above in Steps 9 , 10, and 11.

Add Related Concepts

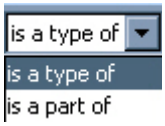
16 (Optional.) To add any relationships the new Concept has with existing Concepts.

Select the **Related Concepts** tab.The **Related Concepts** tab displays:



Note: You may add additional Related Concepts by selecting **Add Concept relationship** from the **Edit** menu of the Concept Window.

- Select the type of Concept relationship from the drop down menu.

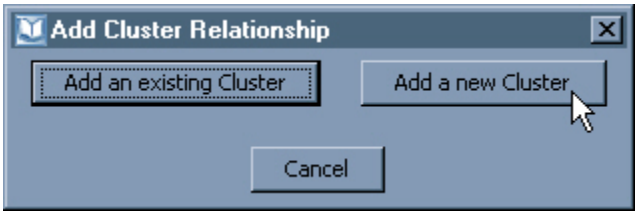


- Drag and drop the existing Concept(s) you want to relate, or type the Concept name in the Add Concept blank.

Note: "Type of" and "Part Of" Concept Relationships does not affect which Concepts are annotated by Intelligent Search or scoring unless triggered by custom Rules or other Dictionary Objects that are created to specifically search on associated relationships.

Add Cluster Relationship

17 (Optional.) You may add a Cluster relationship through the Concept window For more information on Cluster Relationships see *Clusters* on page 39. From the Concept Window, select **Edit**, select **Add Cluster relationship**. The following dialog displays:



- Select **Add a new Cluster**. The **belongs to this cluster** section adds to the Concept window.

- Drag and drop the existing Concept(s) you want to Cluster with, or type the Concept name in the **Add Concept** field.

Note: You may also create a Cluster Object. For more information, see *Clusters* on page 39.

18 Save the Concept.

Select **Save** from the **File** menu in the Concept window.

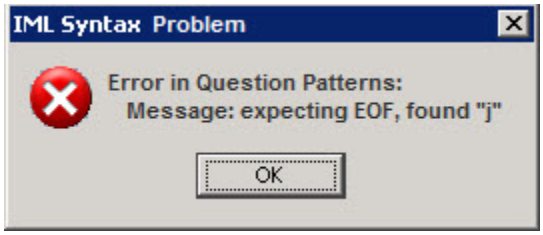
Important! A new Concept does not affect live Intelligent Search results until you *Commit Dictionary Changes to Central Repository (vcroot)* on page 97, a full content re-index is performed, and the changes have been synchronized to the runtime environment, for more information see *Oracle Knowledge Intelligent Search Administration Guide*. To see the effects of new or modified Concepts without committing, use *Test Drive* with the Concept expansion box checked.

Concept Tools

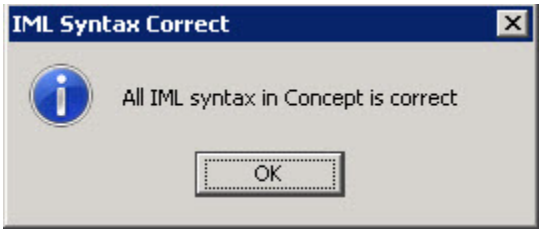
If you entered IML into the **has synonyms matching this IML** field, you may check the validity of the IML prior to saving.

19 Select **Check IML** from the **Tools** menu in the Concept window.

Oracle Knowledge Intelligent Search automatically checks the validity of your IML. If a problem is detected, the **IML Syntax Problem** dialog displays explaining the IML error:



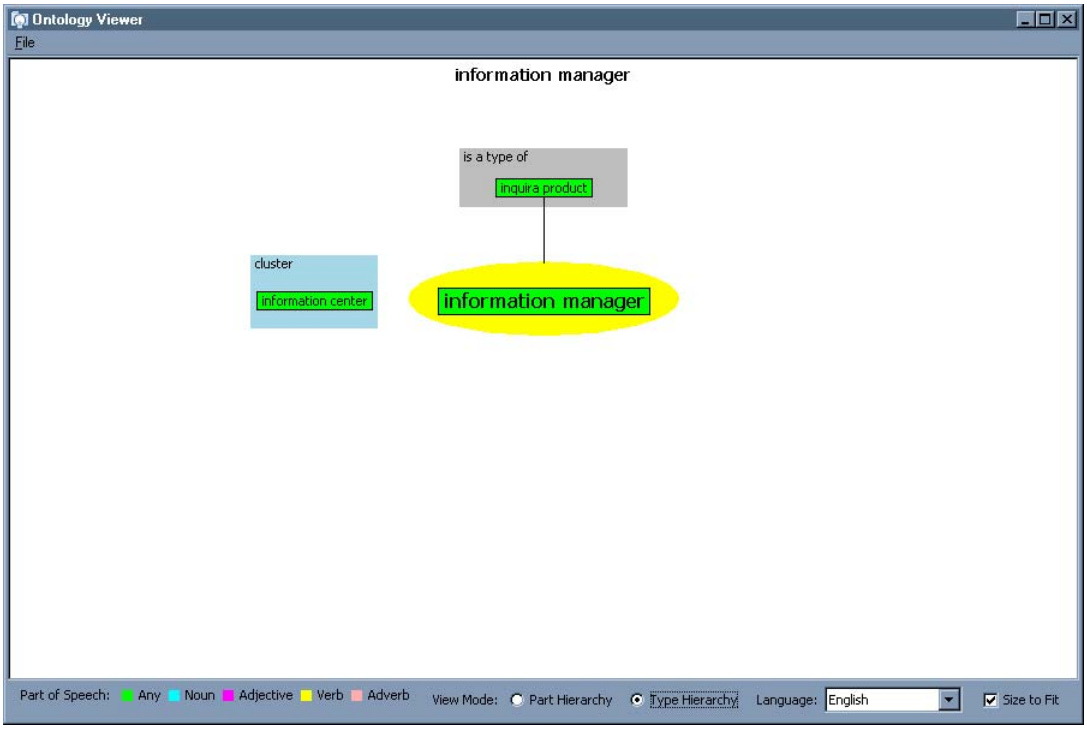
If no problem with the IML is detected, then the **IML Syntax Correct** dialog displays:



Ontology Viewer

The Ontology Viewer allows you to see the Concept and the existing relationships it has with other Concepts and Clusters.

To use the Ontology Viewer, select **Graph** from the **File** menu in the Concept window. The Ontology Viewer displays:

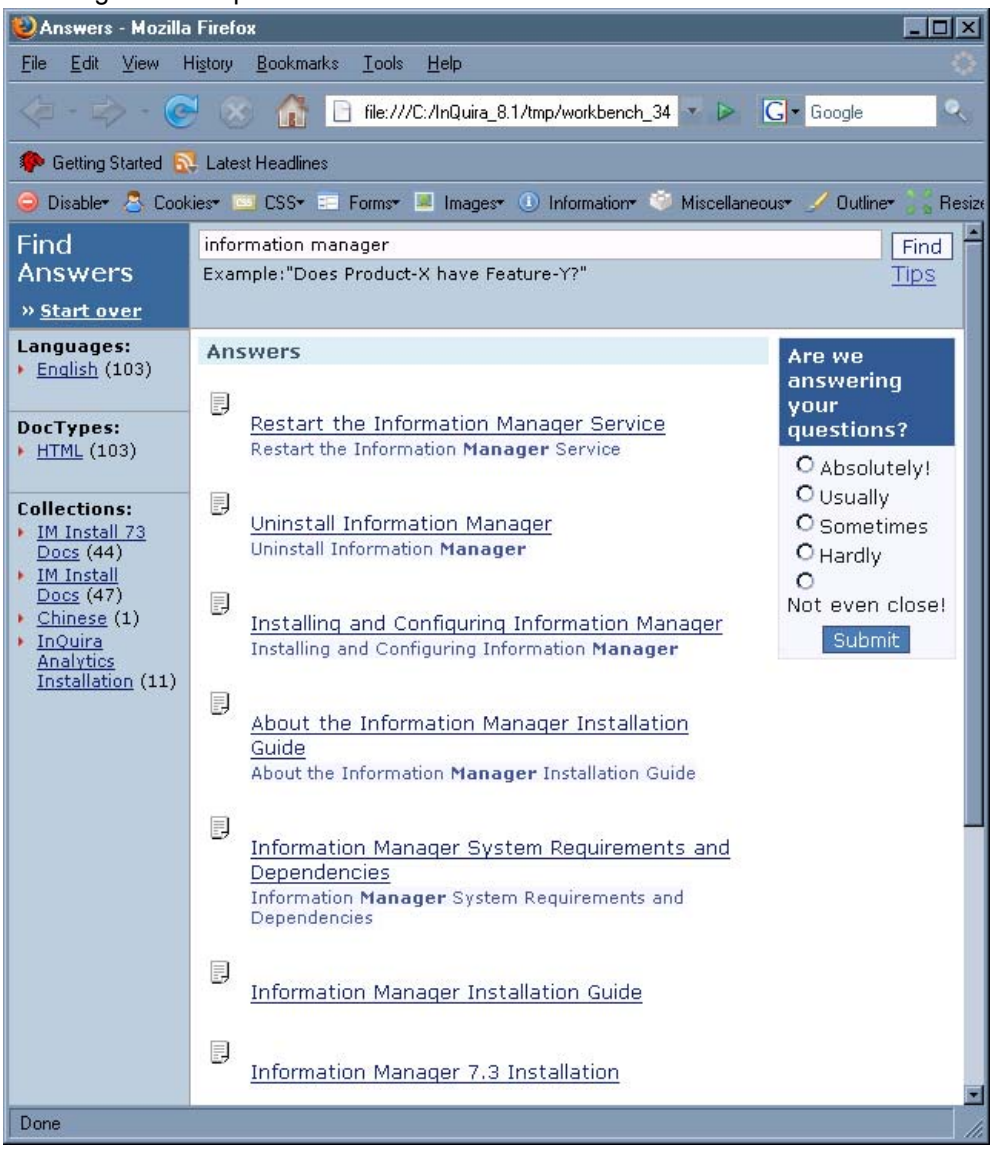


- You may select **Part Hierarchy** to switch the view mode from Type Hierarchy.

- Hovering over a Concept displays the associated synonyms.
- You may select a language from the **Language** menu to see the synonym set of the Concept.

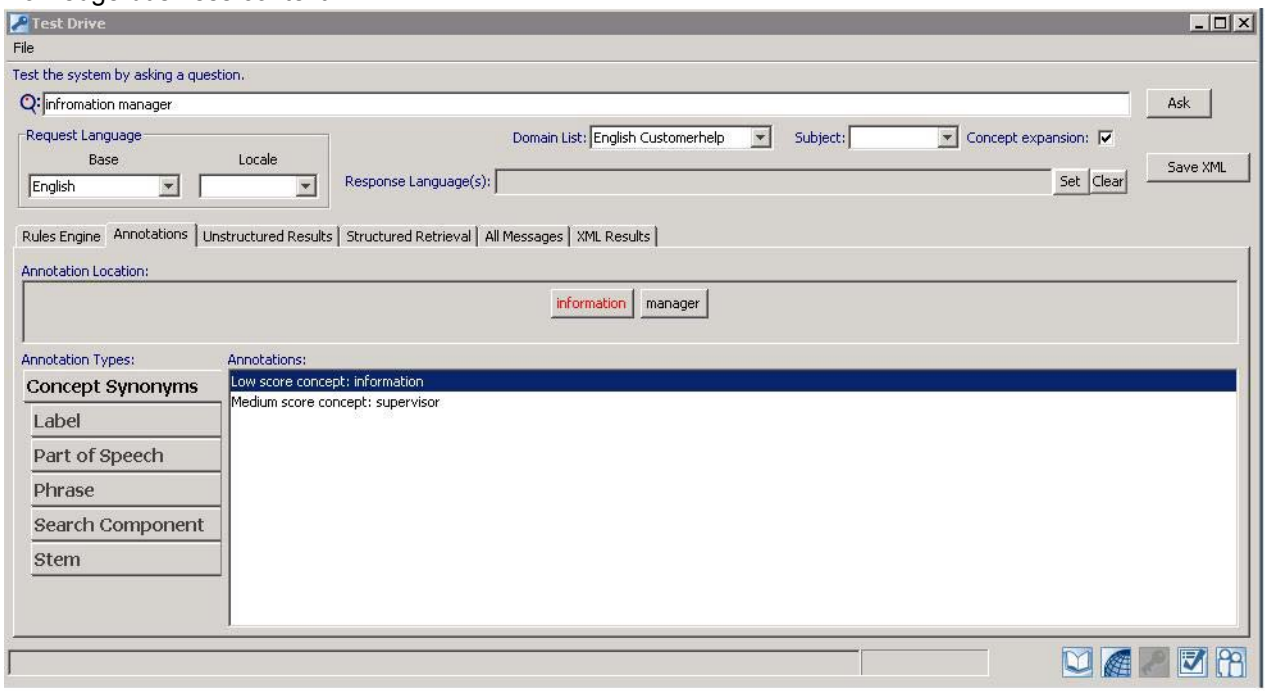
Example of Creating a New Concept

The following screen shows the results page for a search on Information Manager prior to the creation of an <information manager> Concept:

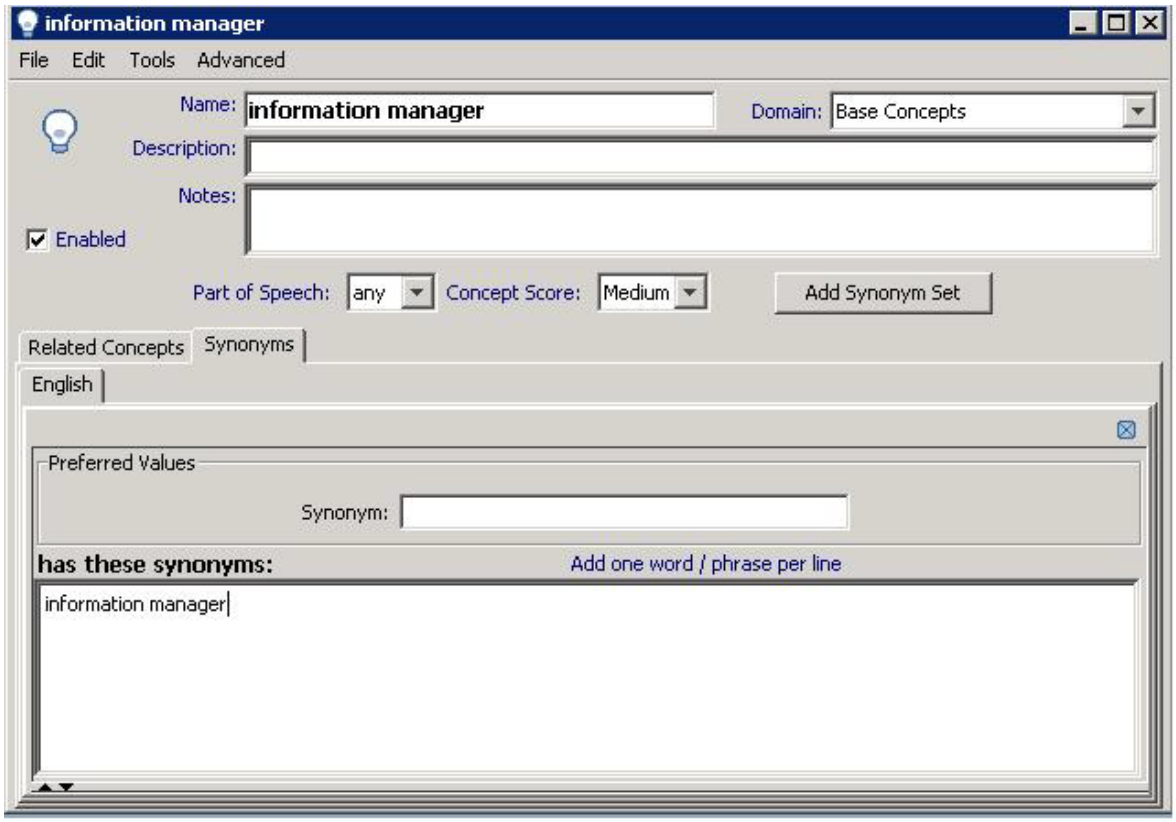


Note: Only *manager* is bold. Bold face words indicate what Oracle Knowledge matched between the query and the documents.

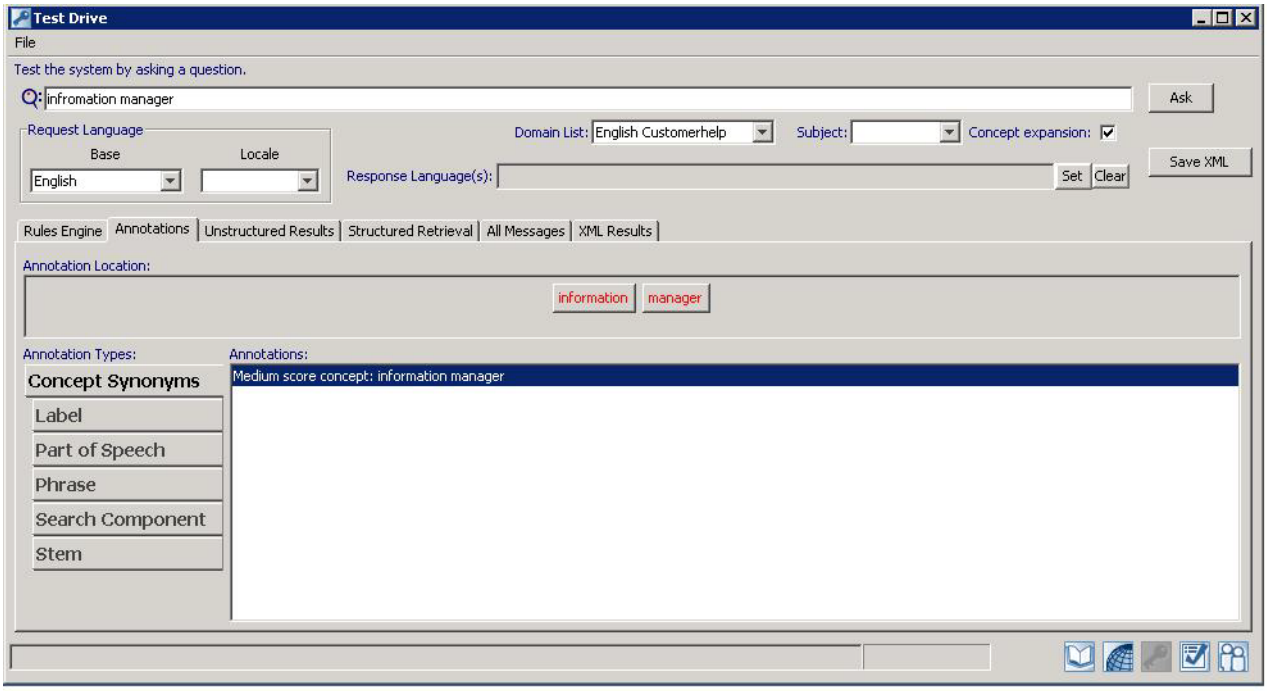
The following screen shot of the **Annotations** tab in Test Drive shows that Information Manager is being treated as two separate Concepts, despite the fact that Information Manager is a single concept in the Oracle Knowledge business context.



In order for Oracle Knowledge Intelligent Search to recognize that Information Manager is a single Concept, we add and save the term as a new Concept in Dictionary Manager.

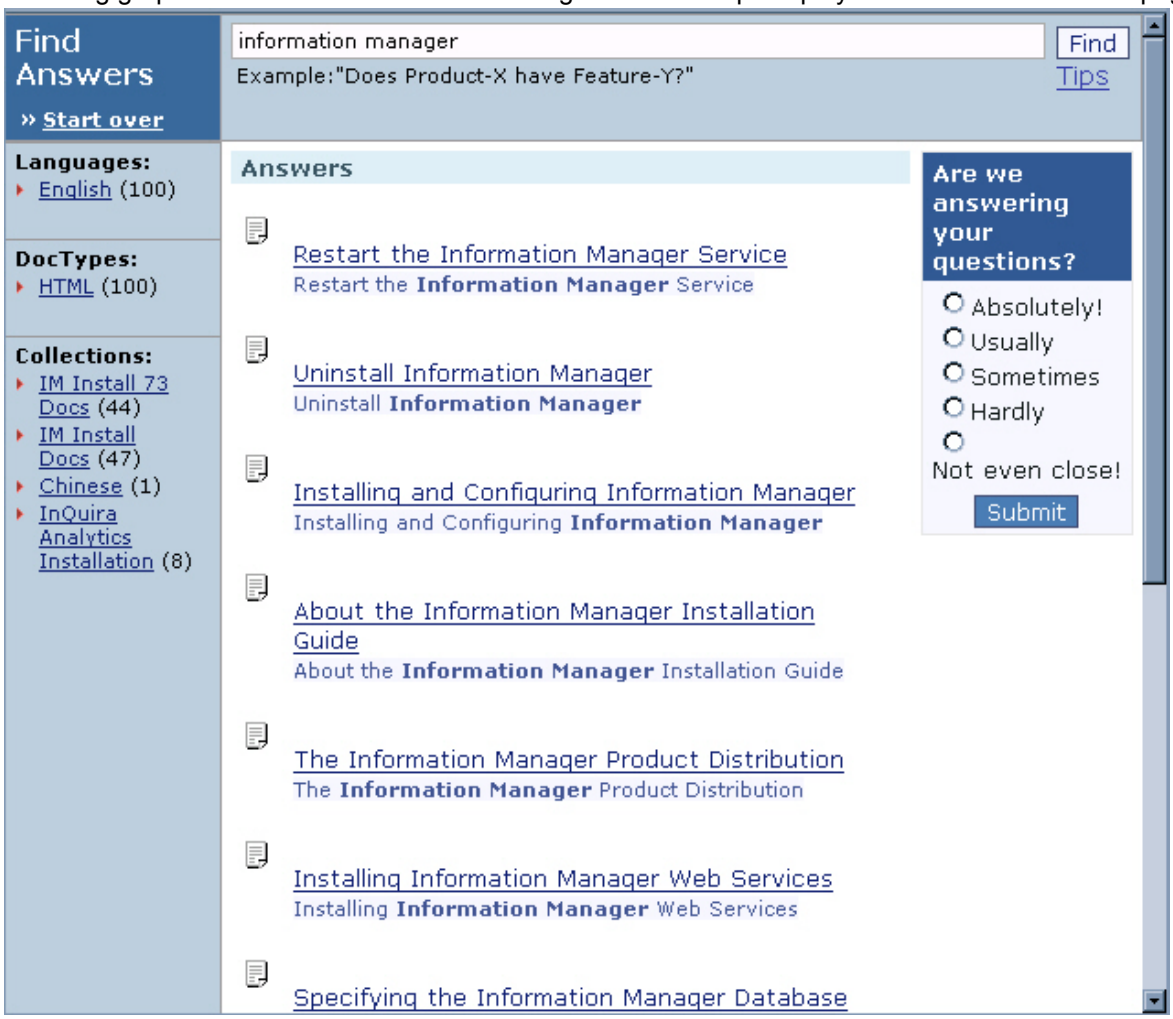


Using Test Drive, confirm that the Concept expansion box is checked (see *Using Test Drive* on page 112 for more information), we can see the effects of creating the new Concept <information manager>.



After creating the Concept <information manager>, it can be seen that Oracle Knowledge Intelligent Search no longer treats the term as two separate Concepts but now as one single Concept.

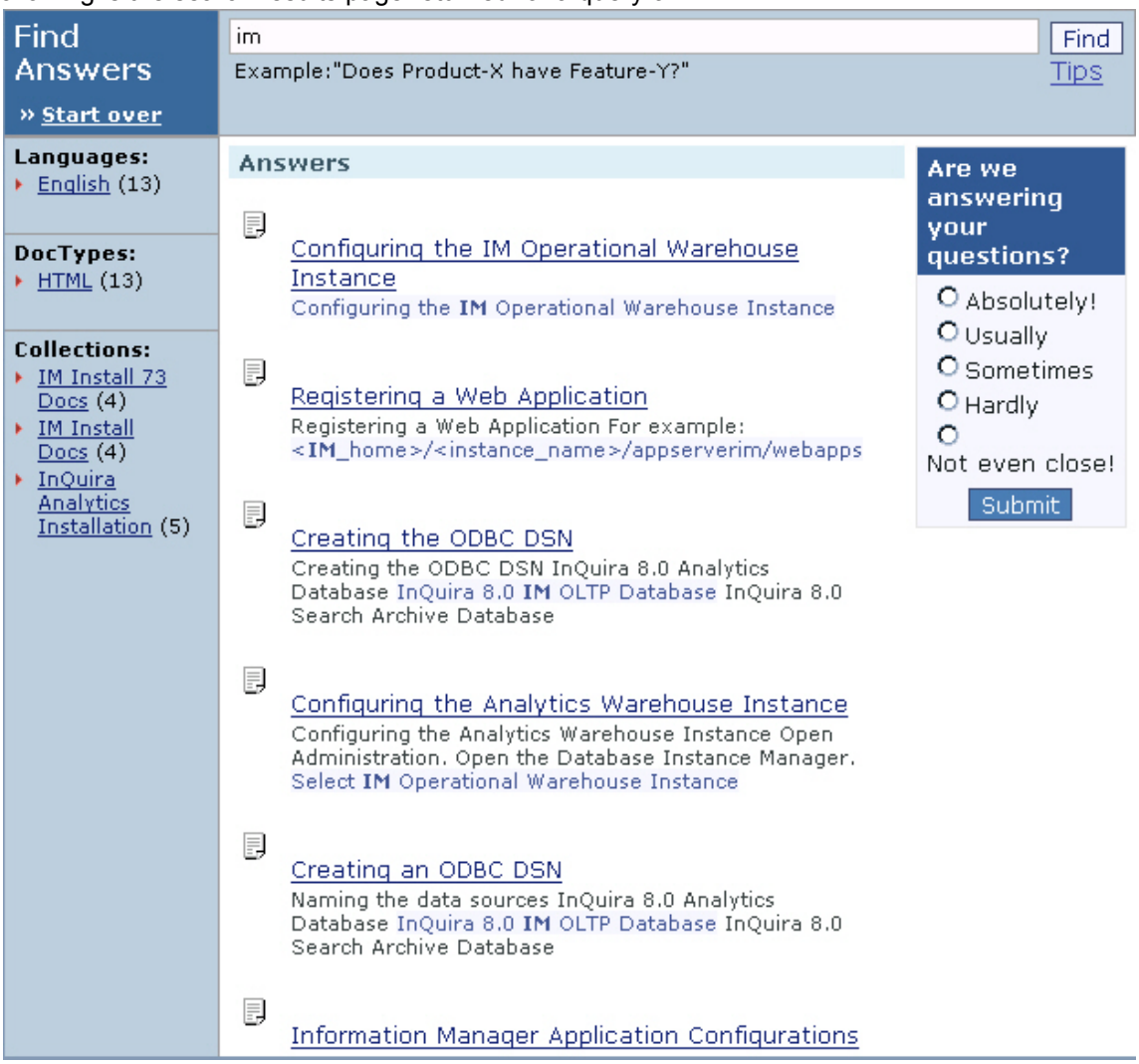
The following graphic illustrates the effect of creating a new Concept displayed on the search results page.



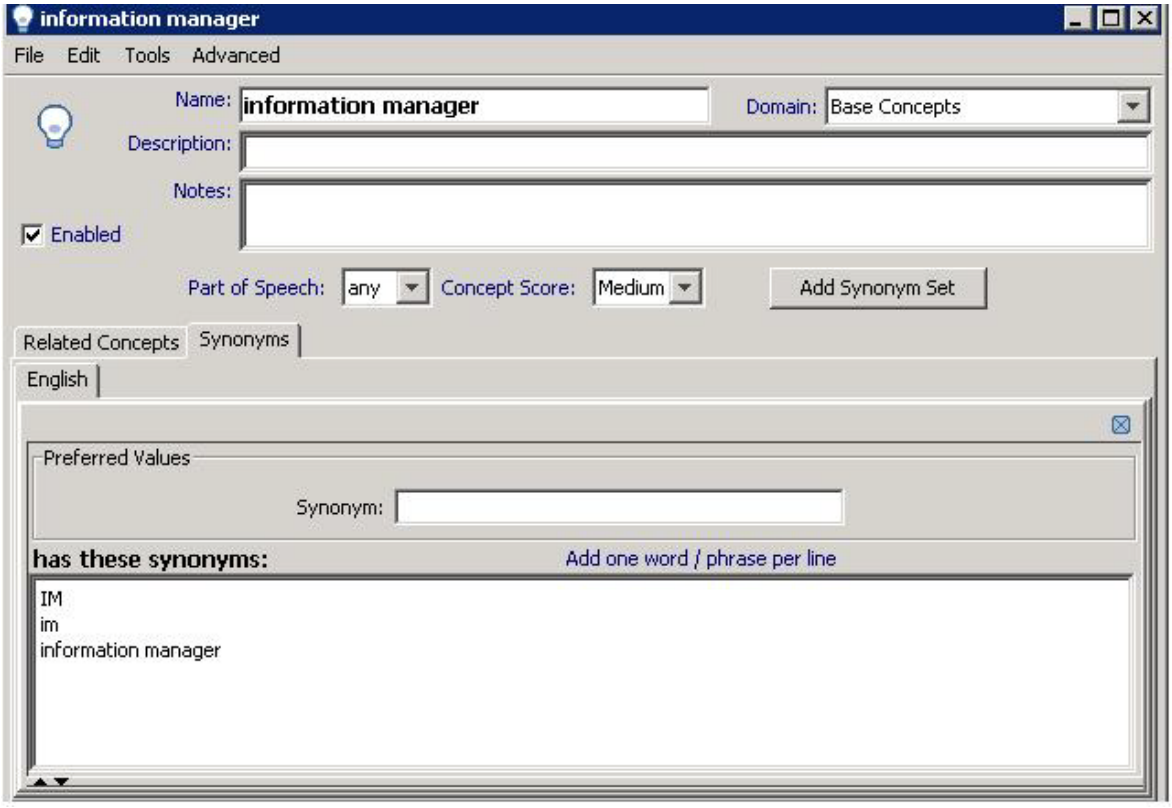
Note: The result have changed and now display “Information Manager” in bold. Additionally, the fifth result has changed to no longer include manager individually, as is now is recognized as a part of a larger phrase.

Example of Creating Synonyms in a Concept

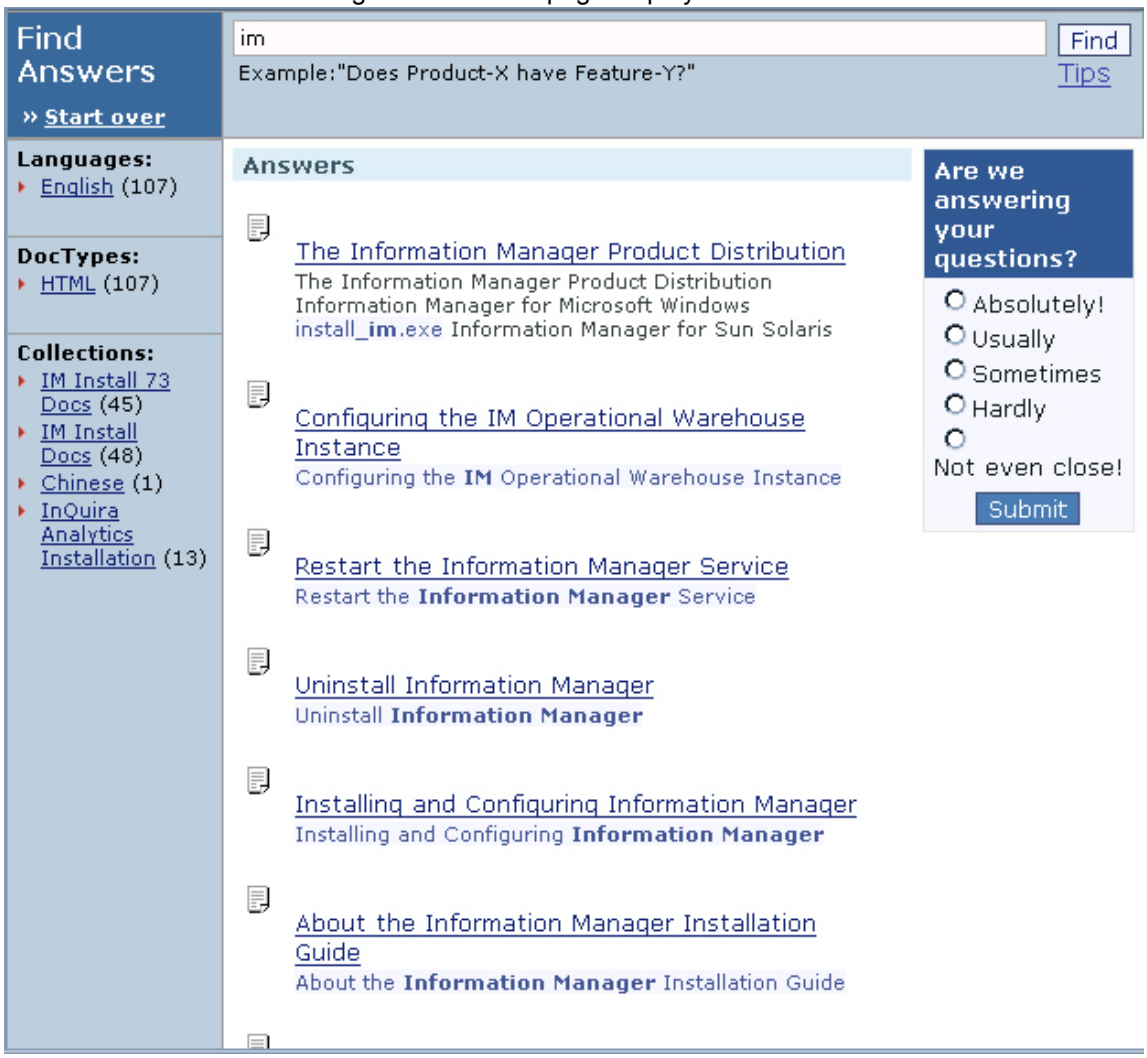
The following is the search results page returned for a query on “im”.



These results answer the query, however at Oracle Knowledge it is known that users often use “IM” to refer to Information Manager, therefore we modify the Concept <information manager> to include “im” as a synonym in order to return queries on im with results related to Information Manager.



After creating and saving the modified Concept, we check the effects in Test Drive, making sure Concept expansion is checked. The following search results page displays:

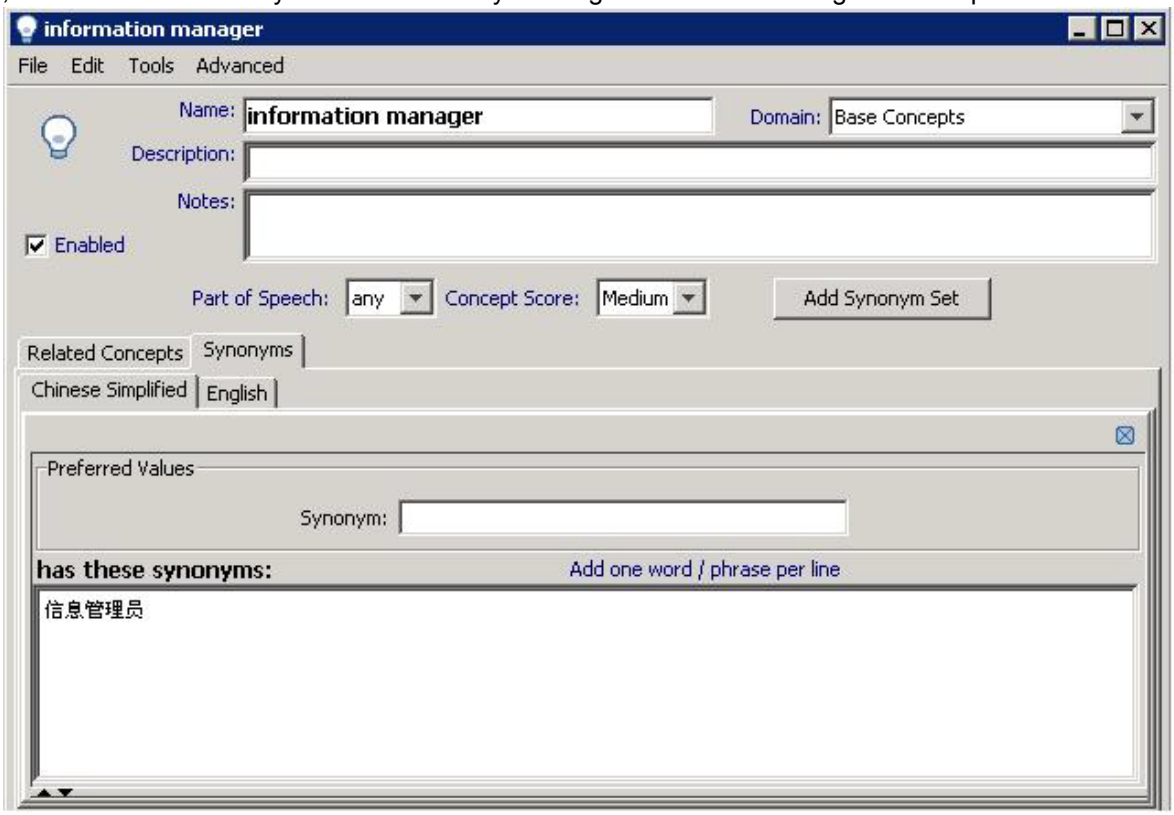


InQuira Intelligent Search matches on results that contain both “im” and “Information Manager.”

Example of Creating a Synonym Set (synset)

In the following example, we create a Chinese synonym to the <information manager> Concept. By creating a synset, we expect Oracle Knowledge Intelligent Search to return documents in Chinese that contain the Concept <information manager>.

First, we add the Chinese synset to the already existing <information manager> Concept.



After creating and saving the modified Concept, we check the effects in Test Drive, making sure Concept expansion is checked. Considering we expect a user to enter in the Chinese synonym for <information manager> from a Chinese-based search page, we select the **language** as Chinese Simplified, and the **Domain List** as Chinese Simplified Customerhelp.

The following search results page displays:

Note: You must select the **Language** and **Domain List** according to the language of the synset you want to test in Test Drive.

Clusters

Clusters are Dictionary Objects that allow you to group similarly related Concepts. For example, the Concept <cat> could be clustered with the Concept <kitten>, as the two are not strictly synonyms, but can be used with roughly the same meaning. Like synonyms of Concepts, Clusters are matched by Oracle Knowledge Intelligent Search when evaluating unstructured search results but are assigned less scoring weight.

Creating Clusters in Dictionary Manager allows Oracle Knowledge Intelligent Search to:

- Expand, contract, or create associations between Concepts.

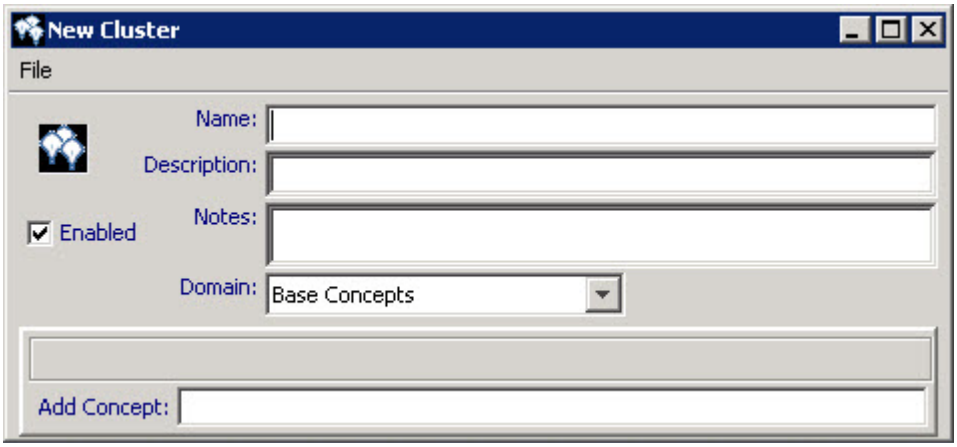
In addition to determining how an unstructured search is conducted, Clusters may also be used to write Rules and Intents. For more information see *IML Basics* on page 143.

Create a new Cluster

By creating new Clusters you allow Oracle Knowledge Intelligent Search to return results that match not only on a particular Concept, but also similarly related Concepts.

To create a new Cluster:

1. In Dictionary Manager select the icon or **New Cluster** from the File menu.
   The Cluster window displays:



2 Name the Cluster in the **Name** field. You may use any alpha-numeric combination.

The Cluster name is used to identify the Cluster. For example, a Rule may use the Cluster name in the IML question pattern.

Note: The only valid characters allowed in the name field are: letters, digits, underscores, percent signs, single quotes, periods, spaces. You may not use special characters, such as Chinese characters in the **Name** field.

3 (Optional). Enter a definition into the **Description** field.

The **Description** field has no impact on Oracle Knowledge Intelligent Search results. It is used to clarify the meaning among users, and can be used to find dictionary objects.

4 (Optional). Enter any information regarding the Clusters in the **Notes** field.

The **Notes** field has no impact on Oracle Knowledge Intelligent Search results. It is used to track comments on Cluster modification among users.

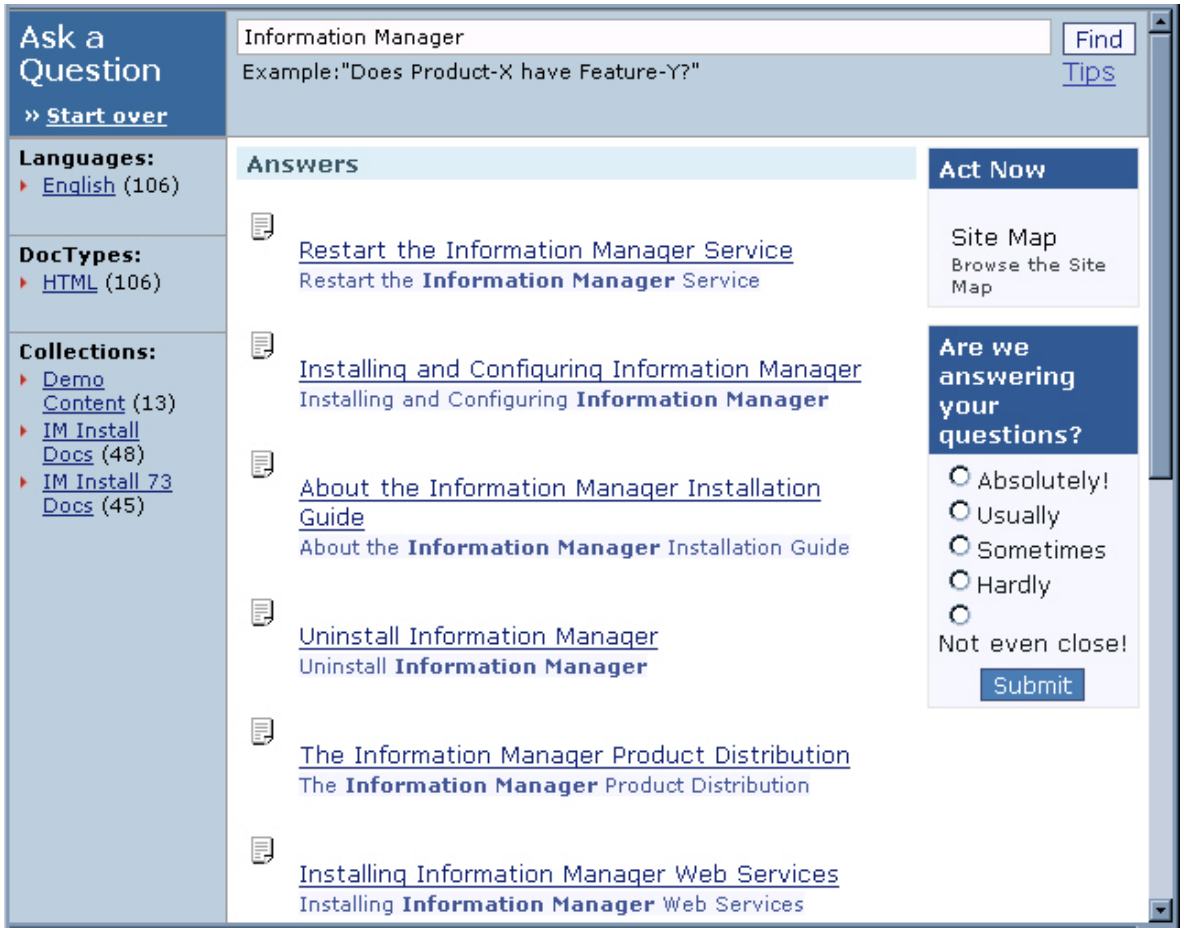
5 Check the **Enabled** box. (This is selected by default.) If the Enabled box is not checked, Oracle Knowledge Intelligent Search ignores the Cluster.

6 Select a domain from the **Domain** menu. For more information see *Concept Domains* on page 83. The default domain list is the last Domain selected.

7 Add the Concepts you want to cluster together by typing the Concept names into the **Add Concept** field, or drag and drop the Concepts into the field.

Example of Creating Clusters

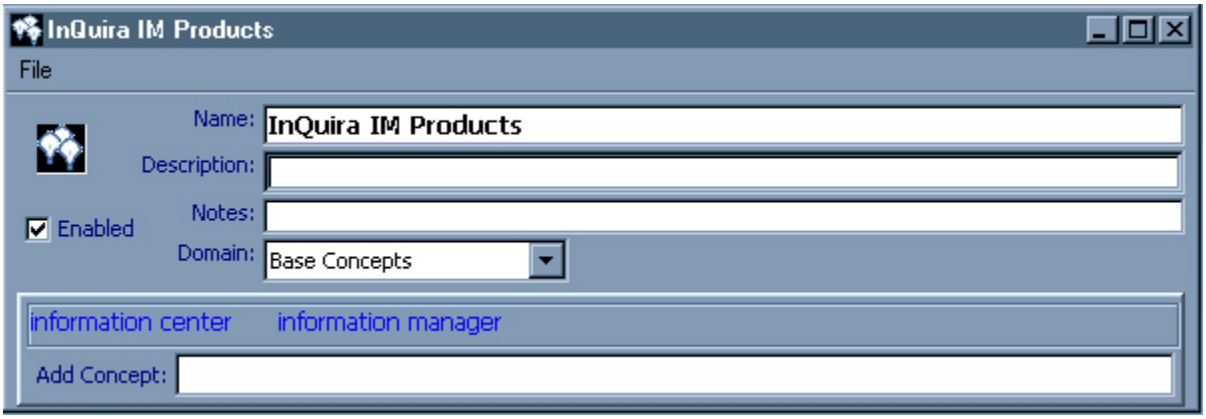
The following screen shows the results page for a search on Information Manager.



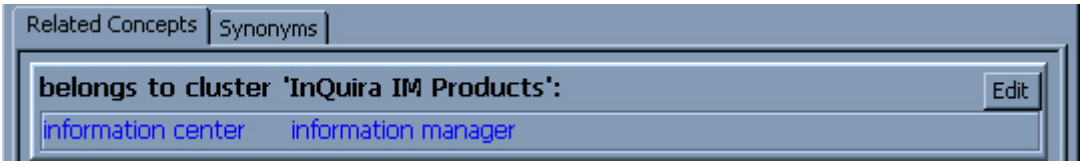
Note: The fifth answer.

In the Oracle Knowledge business context we know that Information Center is the web client for Information Manager. We can safely assume that when a user queries for Information Manager they may also benefit from results on Information Center (and vice versa). By creating Information Manager and Information Center as synonyms we ensure results on both appear on the results page. However, our knowledge of the Oracle Knowledge business context and the English language dictate that although Information Manager and Information Center are related, they do not have the same meaning. Considering the two terms are not synonymous, we should NOT create them as synonyms. Instead, the best and simplest way to ensure information on both Information Manager and Information Center are returned when a user query contains either is to Cluster <information manager> and <.information center>.

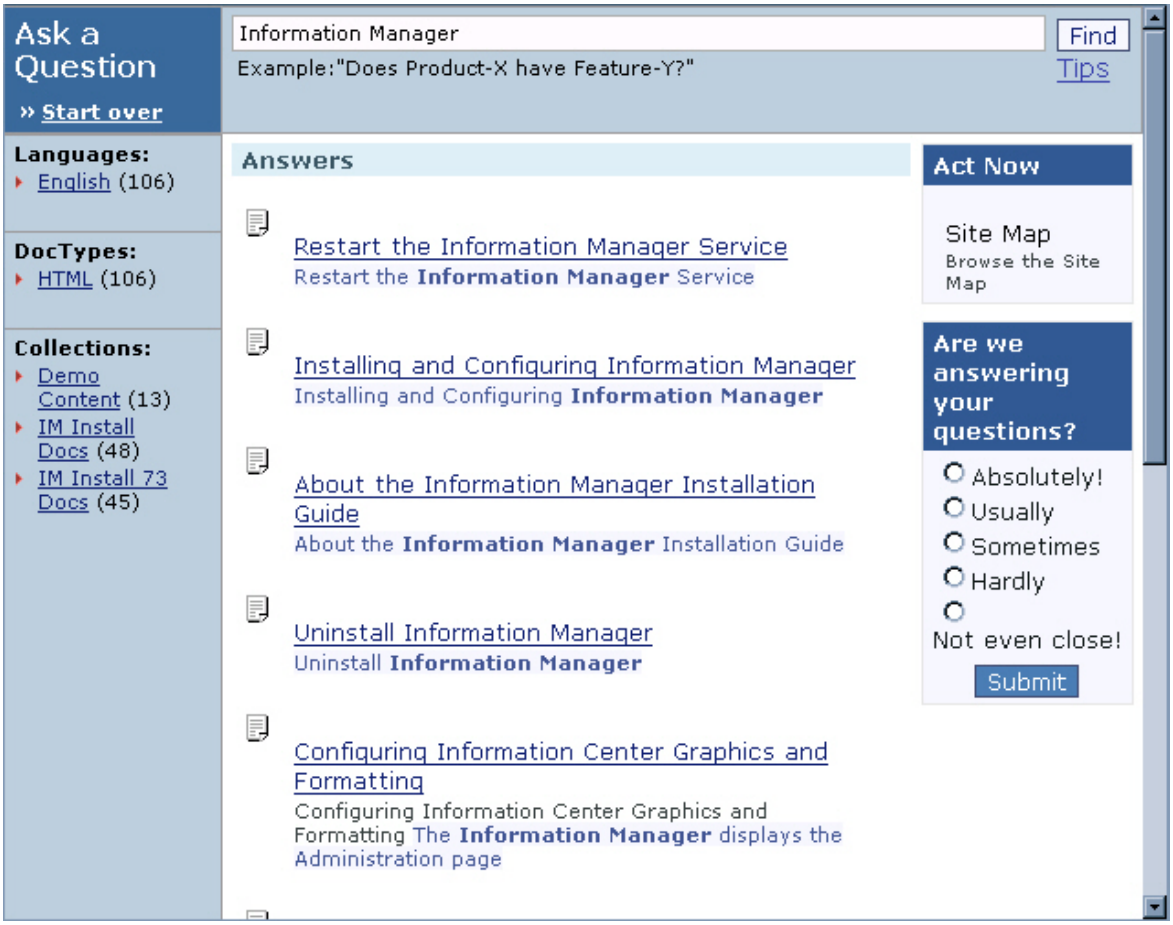
To Cluster <information manager> and <information center>, we must first create Concepts for both in Dictionary Manager. After the Concepts have been created, select to Create a new Cluster. After entering a name, drag and drop both the Concepts to the **Add Concept** field.



Then save the Cluster by selecting **Save** from the **File** menu. Saving the changes automatically adds the Cluster relationship to Concepts <information manager> and <information center>.



After saving, we check the effects of clustering the two Concepts using Test Drive.



By comparing the Oracle Knowledge Intelligent Search results we can see the effects of clustering <information manager> and <information center>. The first four answers remain the same, however the fifth answer has changed due to the cluster relationship. The fifth answer appears higher in the search results as it contains both “Information Manager” and “Information Center.” As expected, the results are not drastically changed as the scoring process used by Oracle Knowledge Intelligent Search first matches on exact matches, then synonyms, and to a much lesser degree Clusters.

Intents, Intent Hierarchies, Intent Responses

Intents are Dictionary Objects that, like Rules, determine how Oracle Knowledge Intelligent Search interprets and responds to user queries. Intents are packaged, installed, and configured as part of the industry-specific dictionaries provided with Oracle Knowledge. default Intents were determined through linguistic and statistical analysis of question logs from leading industry operators. A separate intent has been created for every category that received more than .5% of search questions.

Intents are defined by the language patterns for similar types of queries or requests. An Intent typically captures a family of queries that Oracle Knowledge Intelligent Search recognizes as a unitary Dictionary Object. For example, The “Bill Inquiry” Intent matches on queries about checking a bill balance, how much is owed, and payment deadlines.

Intents are associated with *Intent Responses*. Intent Responses are collections of answer content and additional associated information that you can create and manage as single entities. Intent Responses can be assigned to one or more specified Intents. For example, the Intent Response associated with the Balance Inquiry Intent may provide instructions on how to log in and check an account balance. To liken Intents to Rules, Intents correlate to the question pattern condition, whereas Intent Responses are the action similar to that of a Managed Answer Rule.

Creating Intents and Intent Responses in Dictionary Manager allows Oracle Knowledge Intelligent Search to:

- Return the best possible answers to common user queries.
- Return specific results to satisfy users queries and your company’s needs.
- Group various questions under one common intent.

Intents are organized as parent, siblings, and child nodes in an Intent Hierarchy. Intent Hierarchies descend from a single node at the highest level called *All Intents*. The position of the Intent within the Intent Hierarchy determines its precedence in being selected as an answer to a user query. The Rules Engine selects the lowest-level matching Intent in the hierarchy as the best possible response. If a low-level Intent does not satisfy a user query, the Rule Engine works backwards to match on a higher-level Intent. This process, referred to as *degradation*, ensures that Oracle Knowledge Intelligent Search returns an appropriate Intent Response even if there is none configured for the matched Intent. For example, the Bill Inquiry Intent is a child of the Billing Inquiry and Support Intent. If there is no Intent Response configured for the Bill Inquiry Intent, then the Intent Response for the Billing Inquiry and Support Intent displays.

Note: All Intent Responses use the Answer template, which means they all contain the title, url, picture, and text fields.

Intent Responses and Managed Answers

Functionally, intent responses and managed answers are very similar. They are both a means to push a piece of predetermined content to the user as a response to a query. They are organized differently though—Managed Answers are straightforward rules. Managed Answers have a condition section (question pattern, business conditions) and an action section (answer action -> custom content).

For more on Rules and Managed Answers, see *Rules* on page 61.

Intents have a different, but corresponding, design. The intent section corresponds to the question pattern of a managed answer rule. The dictionary object Intent Response—which is where the predetermined content is specified— corresponds to the action part of a rule. When intents match in a search nothing really happens unless an Intent Response is configured for that intent. When an Intent Response is configured for an intent, the predetermined content gets pushed out to the user.

However, there are several differences in how Intent Response and Managed Answers implemented and administered:

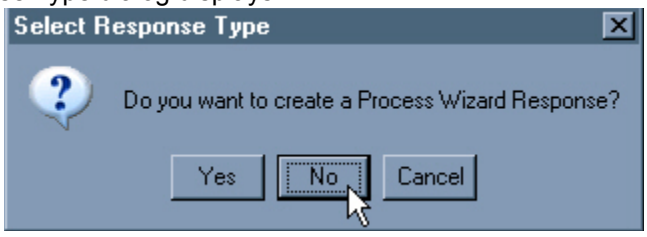
- Oracle delivers hundreds of intents already configured for online customer support in four industries (telecom, automotive, finance, and customer help) available at install. The preconfigured intents can be used as is or modified as needed.
- Intents are organized in a hierarchy such that if a query matches a very specific intent it also matches a more generic one. So “online bank transaction” intent would be more specific than “bank transactions”; so that if a query matches the “online bank transaction” intent, it also matches “bank transactions” intent. The advantage of this is you can choose to configure responses at different levels of specificity.
- Intents are more general in their formulations and the specific term in the query that triggers it is usually captured as a parameter. So if there is a general product search intent in the dictionary, the particular product that triggered it would be captured as a parameter, so you would have a car model search triggered by a Honda Civic search, for instance. In this case the Civic concept becomes the parameter of the product search intent, so that the Intent Response can be configured not only with regard to the product search intent, but more specifically for the Civic search, as opposed to Accord search etc. This allows you to use the same intent to configure different responses based on what is being matched.
- Intent Responses can get their content from specifically configured Information Manager channels, which makes it possible for Information Manager workflows to be involved in the creation and modification of the content of the Intent Response. The presence of the Intent Response itself and what intent it is associated with is determined in the workbench.The content of Managed Answers can only be created and modified inside the Workbench. It is not possible to do this in IM.
- Managed Answers operate with ad hoc business conditions (you can add them to the rule in any combination), which means if the business condition does not match, the Managed Answer does not fire. With Intent Responses, you cannot add business conditions directly. To add business conditions in the intent system, you must first package them in Test Profiles and then make the Intent Response conditional on the Test Profile. Thus, the intent always matches regardless of business conditions, but you control the Intent Response through contextual information.

Create and Assign Intent Responses

By creating new Intent Responses, you allow Oracle Knowledge Intelligent Search to return a set answer to a variety of user queries that essentially have the same intent.

To create a new Intent Response:

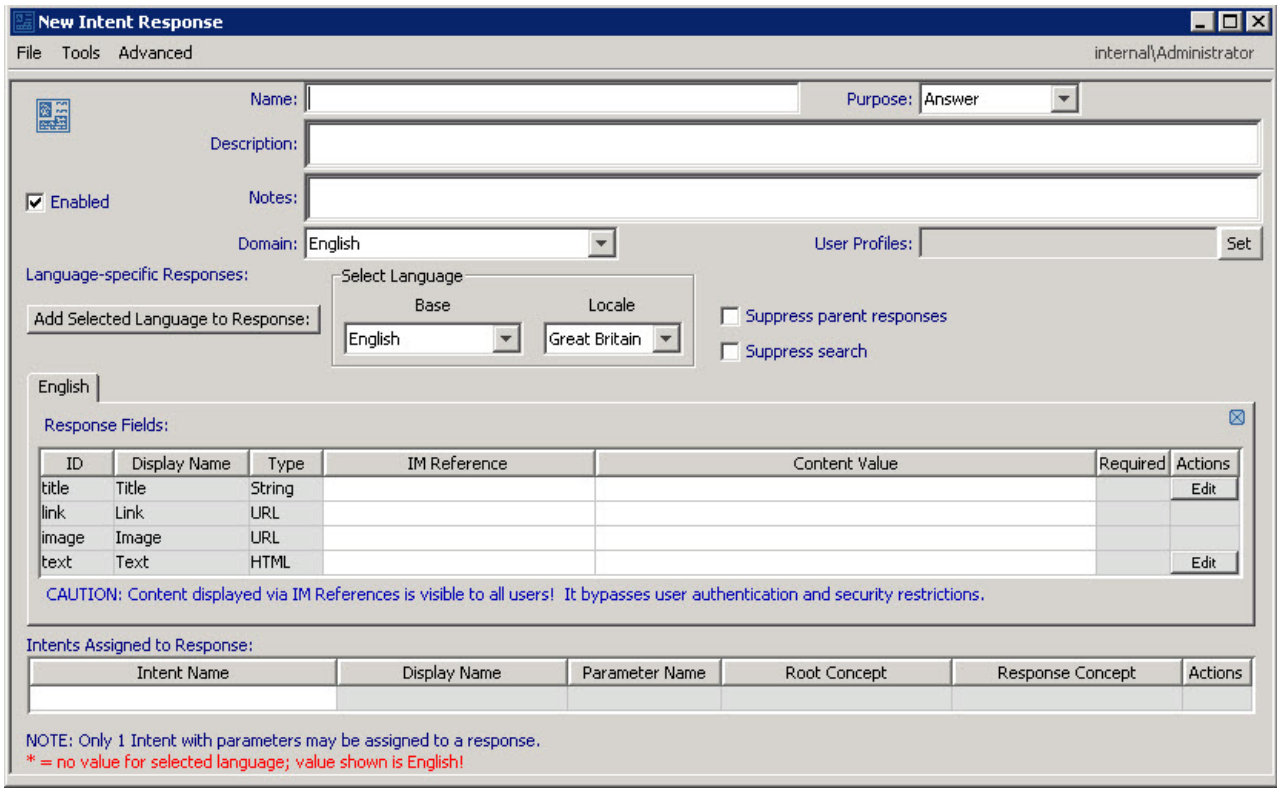
1. In Dictionary Manager, select the icon or **New Intent Response** from the File menu.
   The Select Response Type dialog displays:



2. Select **Yes** or **No**. (The following instructions assume you select **No**.)

   Note: If you want to create an Intent Response that uses a Process Wizard, you must have already created the Process Wizard you intend to use. For more information on creating Process Wizards see *Oracle Knowledge Intelligent Search Optimization Guide*.

:The Intent Response window displays:



3 Enter a descriptive name for the Intent Response in the **Name** field.

Note: An Intent Response cannot have the same name as an Intent or a Rule.

4 Select a **Purpose** from the drop down list.

The Purpose corresponds to display characteristics defined in the User Interface, enabling you to establish consistent, focused and targeted presentation for various types of content. For more information, see *Purpose (portlet)* on page 143.

5 (Optional.) Fill in the **Description** field.

The Description field has no impact on Oracle Knowledge Intelligent Search results. It is used to clarify the intended function of the Intent Response among users.

6 (Optional.) Enter any information regarding the Intent Response in the **Notes** field.

The Notes field has no impact on Oracle Knowledge Intelligent Search results. It is used to track comments on Intent Response modification among users.

7 Check the **Enabled** box. (This is selected by default.) If the Enabled box is not checked, Oracle Knowledge Intelligent Search ignores the Intent Response.

8 Select the **Domain** from the drop down list. Only Domains you have permission for are displayed. The Domain field is set to the most recently saved Domain by default. For more information on Domains, please see *Non-Concept Domains* on page 83.

9 (Optional.) Select a **User Profile**.

Click Set and select a user profile from the drop down list if you want Oracle Knowledge Intelligent Search to restrict the return of the Intent Response based on business conditions associated with the query. To establish User Profiles see *Oracle Knowledge Intelligent Search Optimization Guide*.

10 (Optional.) Check the **Suppress parent responses** box. (This is deselected by default.)

Checking the **Suppress parent responses** box configures Oracle Knowledge Intelligent Search to not display Intent Responses configured for a parent Intent unless the query directly matched the parent Intent.

11 (Optional.) Check the **Suppress search** box. (This is deselected by default.)

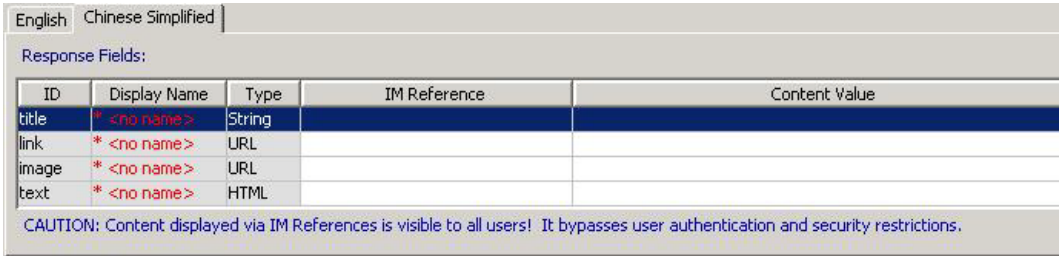
Checking the **Suppress search** box configures Oracle Knowledge Intelligent Search to stop the search process once an Intent associated with this Intent Response is matched. Therefore, the search results page display only one result, the Intent Response.

12 (Optional.) Select an additional language from the **Add Selected Language** drop down list.

The additional language selection refers to the language used in the specified response.

The default language is English.

Upon selecting **Add Selected Language**, an additional language tab displays:



13 Enter Content Values in the Response Fields. If you want to associate the Intent Response with an Information Manager document enter the information in the following format in the **IM Reference** field: *DOCID:Channel_name/@field*

Note: Using an IM Reference makes the Information Manager document visible to all users when the Intent Response displays.

a Enter text in the **Title** Field.

Type the text as you want to see it displayed as hypertext on the Oracle Knowledge Intelligent Search results page.

b Enter a complete URL in the **Link** field

Type the full URL link associated with the title specified above, and the optional graphic specified below.

c (Optional.) Enter a complete URL in the **Image** filed.

Type the full URL of an image you want to appear on the Oracle Knowledge Intelligent Search results page. The image takes the URL specified above.

d (Optional.) Enter any additional text in the **Text** field.

Use this field to create custom text or HTML to display along with the title at the top of the Oracle Knowledge Intelligent Search results page.

14 (Optional.) Repeat Steps 12 and 13 for each additional language.

15 Assign an Intent to the Intent Response.

Select an Intent from the drop down **Intent Name** list. Only Intents that are included in the Intent Hierarchy are available for selection. You may associate more than one Intent with the Intent Response, unless the Intent has specified Concept parameters, and you have specified a Response Concept for the Intent Response.

Note: Only one Intent can be associated with an Intent Response if the Intent has specified Concept parameters, and a specified a Response Concept for the Intent Response. For more information see *Define Concept Parameters* on page 54.

After you select an Intent from drop down list, the Display Name field automatically completes with the display name of the Intent.

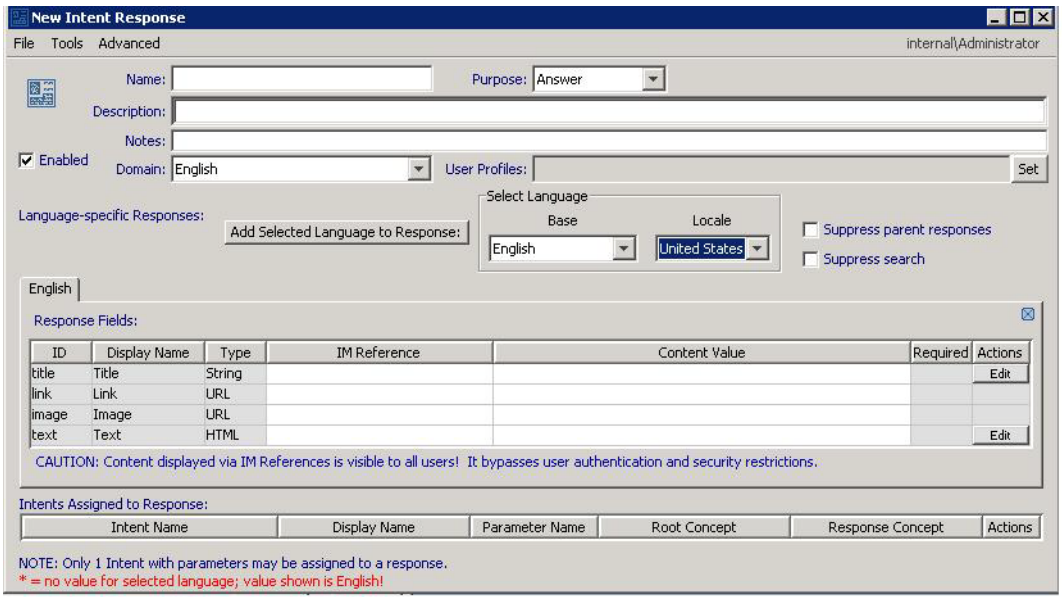
If the Intent you selected has specified Concept parameters, the Parameter Name and Root Concept fields automatically complete with information from the Intent. If you are using an Intent with a Root Concept, you may enter the Root Concept or any children of the Root Concept in the Response Concept field to limit the return of the Intent Response. A *Root Concept* is any concept that has associated relationships where the Concept is the Parent or Grandparent of other Concepts.

Note: The Intent Response does not appear on the Oracle Knowledge Intelligent Search results page unless it is assigned to an Intent, and that Intent is included in the Intent Hierarchy.

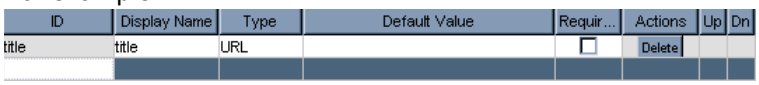
16 Save the Intent Response.

Select **Save** from the **File** menu in the Intent Response window.

Note: A new Intent Response does not affect live Intelligent Search results until you *Commit Dictionary Changes to Central Repository (vcroot)*, and the changes have been synchronized to the runtime environment, for more information, see *Oracle Knowledge Intelligent Search Administration Guide*. To see the effects of new or modified Intent without committing, use *Test Drive*.



Field	Description
Name	Specify the Response Template name.
Description	Enter an optional description of the template. The content of the Description field is for Dictionary Manager users' reference only.

Field (Continued)	Description (Continued)
Notes	Enter any optional notes. The content of the Notes field is for Dictionary Manager users' reference only.
Domain	Specify the Domain to which this template belongs. The pull-down menu displays only domains for which you have permission. For new templates, the Domain field is set to the domain most recently saved to.
ID	Specify an ID for the Response field. The ID must be unique, and cannot be changed after the parameter is saved. For example:  The Dictionary Manager automatically populates additional fields when you specify this field. Select each field to edit its contents.
Display Name	Specify the name that the Oracle Knowledge Language Workbench uses to display this parameter.
Type	Select the type of field from the drop-down menu of supported field types: • HTML • URL • String • Boolean • Decimal • Integer
Default Value	Specify an optional default value for this field, in the appropriate format for the corresponding data type.
Required...	Specify whether this field is optional or required within a formatted Response.

Intent Response Tools

Before saving an Intent Response, you can verify the function of the Intent Response by selecting an option from the **Tool** menu in the **Intent Response** window. Choose from:

Validate Response on page 49

Reset IM Document Cache on page 50

VALIDATE RESPONSE

After selecting **Validate Response** from the **Tools** menu, Oracle Knowledge Intelligent Search automatically checks the validity of your Intent Response. If a problem is detected, the **Invalid Value** dialog displays explaining the error with the Intent Response

If the Intent Response is valid the following **Response is Valid** dialog displays.

RESET IM DOCUMENT CACHE

Use **Reset IM Document Cache** located under the **Tools** menu to see the latest version of the IM document referenced in the I**M Reference** field. Use this option if the IM document has changed since the Language Workbench last cached the content.

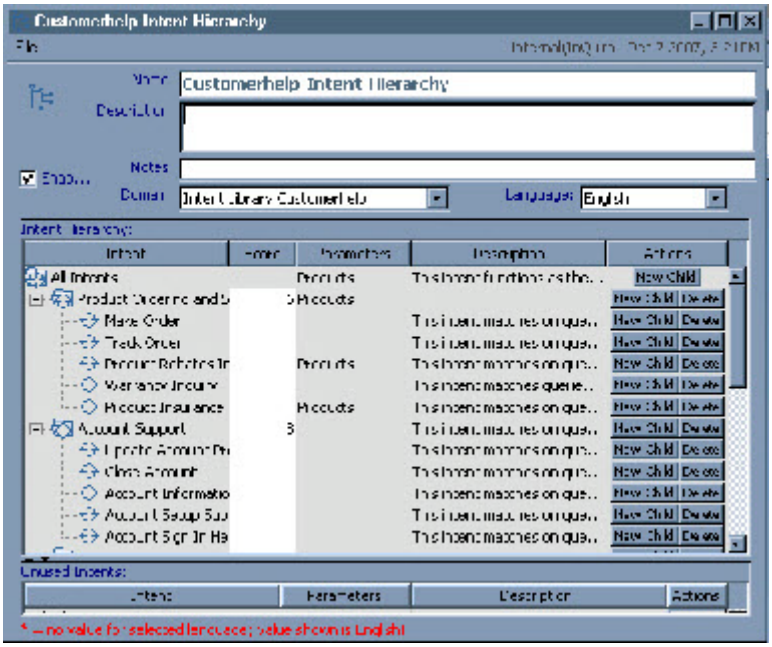
Create a New Intent in the Intent Hierarchy

Oracle Knowledge Intelligent Search provides Intents that account for the majority of user queries. It is rare that you need to create a new Intent. However, if there is a need to do so, you can create a new Intent in order to allow Oracle Knowledge Intelligent Search to analyze common user queries phrased differently as having the same meaning, or intent. You can then associate the Intent to return a specific result, or Intent Response.

In order for an Intent to be used by Oracle Knowledge Intelligent Search, the Intent must be included in the Intent Hierarchy. For this reason we recommend creating a new Intent through the Intent Hierarchy window.

To create a new Intent:

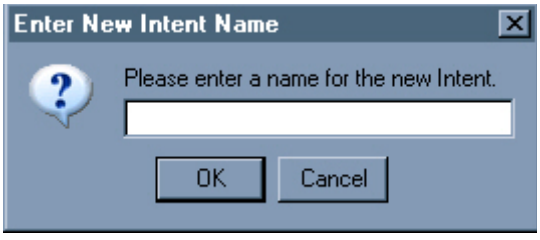
1. In the Dictionary Manager, use the **Dictionary Find Tool** to open the Intent Hierarchy to which the new Intent belongs. For information about the Dictionary Find tool, see *Finding Dictionary Objects* on page 95. The Intent Hierarchy Window displays:



2. From the displayed Intent Hierarchy, select a parent for the new Intent, and select **New Child**.

   Note: When selecting the location, keep in mind that Oracle Knowledge Intelligent Search matches on the higher level Intents as well through degradation. For this reason, choose a location for your new Intent that is closely related yet more specific than the higher-level Intents (parents). The location of the Intent among Intents of the same level (siblings) does not impact search results.

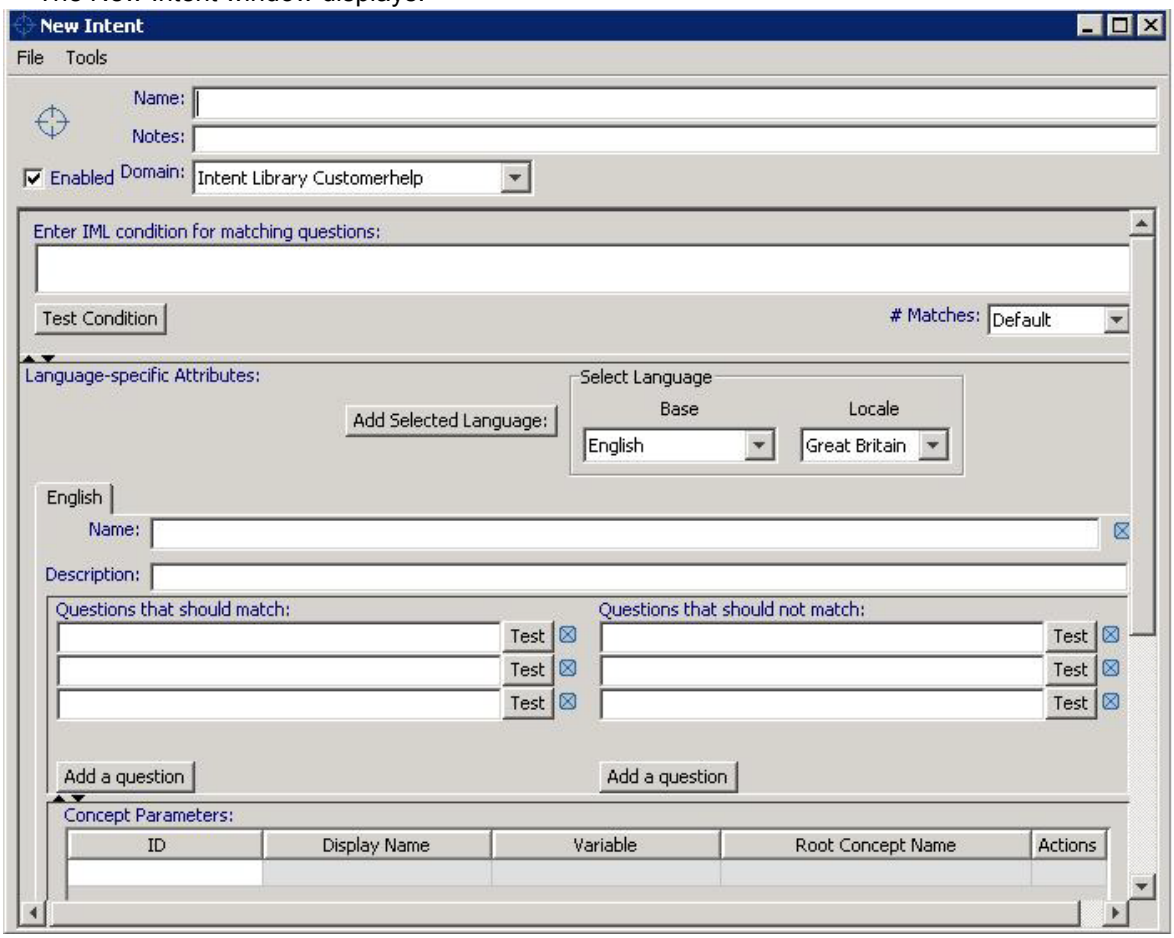
   The Enter New Intent Name window displays.



3. Enter a descriptive name for the new Intent and click **OK.**

Important! Confirm that there is no Rule, Intent, or Intent Response in the Dictionary with the same name.

The New Intent window displays:



4 (Optional.) Enter any information regarding the Intent Response in the **Notes** field.

The Notes field has no impact on Oracle Knowledge Intelligent Search results. It is used to clarify the intended function of the Intent Response among users.

5 Check the **Enabled box** (default setting). If the Enabled box is not checked, the Intent Response is ignored by Oracle Knowledge Intelligent Search.

6 Select the **Domain** from the drop down list. Only Domains you have permission for are displayed. The Domain field is set to the most recently saved Domain by default. For more information see *Non-Concept Domains* on page 83.

7 Complete the **IML Conditions** section.

- In the space provided, enter the IML expression used to match the variety of words and phrases that can be used to express a particular user query for information or assistance. This IML is used to match a question and then trigger the associated Intent Response.

If you plan to create an Intent that uses Concept Parameters, see *Define Concept Parameters* on page 54, then the IML in the Question Pattern must contain a Global variable. *Global variables* are place holders with no intrinsic value. Their value is assigned by the matching process. Global variables are preserved until the complete search process is complete. Intents can refer to a Global variable

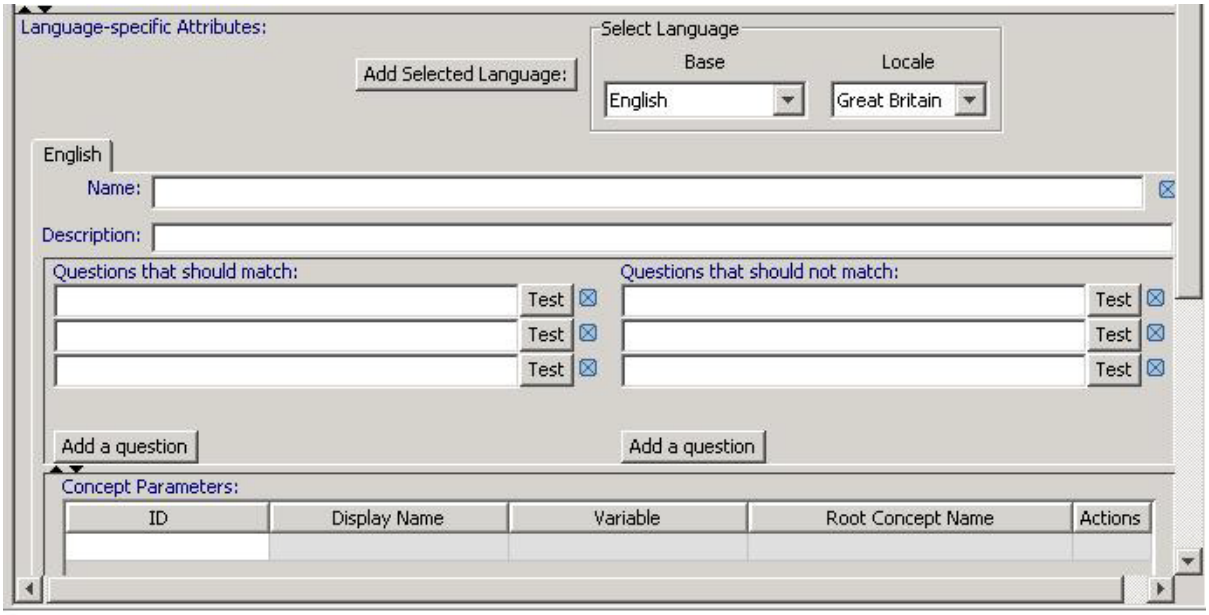
created in the Intent or in a separate Rule or Intent. The value captured by the global variable is what is passed as a concept parameter to Intent Responses. Global variables are indicated with a # appearing before the name of the variable.

8 Select the number of matches for the IML expression to find within a given request from the drop down # **Matches** list.

This value controls how many times a rule may fire if it matches more than one part of a query. This is necessary only for more complex rules; for most purposes this setting is the default.

Default	Specifies to process only one match. (is the default setting.)
One Match	Specifies to process only one match.
Most Matches	Specifies to process most matches.
All Matches	Specifies to process all matches.

9 Complete the **Language-specific Attributes** section.To select an additional language, select from the **Add Selected Language** drop down list. The Language-specific Attributes tab displays:



10 Enter a name in the **Name** field.

The name typed in the Language tab acts as the display name of the Intent. For English, it is best practice to copy the name of the Intent specified above. For all other languages, it is best practice to enter the translation of the name of the Intent specified above.

11 (Optional.) Fill in the **Description** field.

The Description field has no impact on Oracle Knowledge Intelligent Search results. It is used to clarify the intended function of the Intent among users.

12 (Optional.) Enter Test Questions.

Under the appropriate headings enter questions that you expect to match or not match based on the IML question pattern specified above. To test each question individually click the Test button located at the end of the text field. To test all questions at the same time click the **Test Condition** button located

above. After selecting **Test Condition**, the question(s) specified turn blue if it is satisfied correctly (matching or not matching), and red if it is not. The question appearing as red indicates that you need to modify your IML question pattern in order to return the results you expect.

If you want to test additional questions, select **Add a Question** under either heading. An additional text field displays.

Define Concept Parameters

13 (Optional.) Define Concept Parameters.

Concept Parameters enable a single Intent, using the same IML question pattern, to match a range of questions that differ only in a specific detail, such as a product name. In order to make use of Concept Parameters the IML question pattern specified above MUST contain a global variable.

a Enter a name for the Concept Parameter in the **ID** field.

The ID field has no impact on Oracle Knowledge Intelligent Search results. It is used as a place holder during the search processes. The ID cannot be changed once the Intent has been saved.

b Enter a name for the Concept Parameter in the **Display Name** field. (ID name by default.)

The Display Name completes with the ID name by default. You may choose to modify the display name, however keep in mind that the Display Name has no impact on Oracle Knowledge Intelligent Search results. It is used as a place holder during the search processes.

c Select a **Variable** from the drop down list.

In order for Intents to use Concept Parameters the IML Question Pattern specified above must contain a global variable. The global variable(s) defined in the question pattern above appears as a drop down list in the variable field.

d Enter the **Root Concept Name** associated with the variable.

In order for an Intent Response to trigger based on the Concept Parameters you must associate both the Concept Parameter and the Intent Response with a Root Concept. You may type or drag and drop the Root Concept name into the field. .

14 (Optional.) You may add additional languages by repeating Steps 9 10 1112and 13

15 Save the Intent.

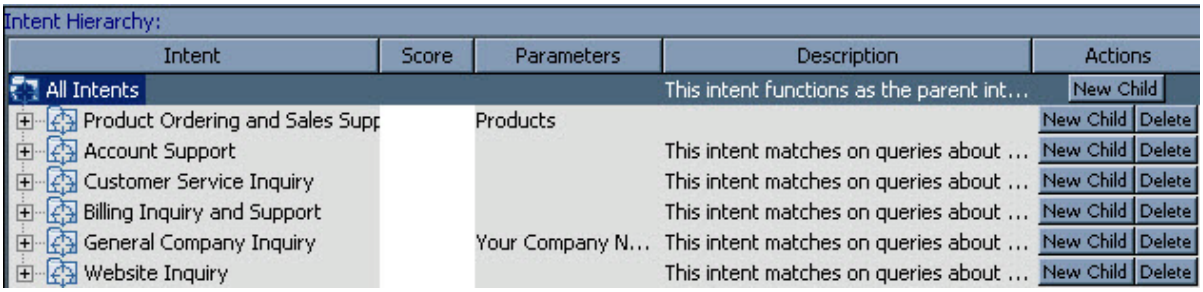
Select **Save** from the **File** menu in the Intent window.

Note: A new Intent does not affect live Intelligent Search results until you *Commit Dictionary Changes to Central Repository (vcroot)*, and the changes have been synchronized to the runtime environment, for more information, see *Oracle Knowledge Intelligent Search Administration Guide*. To see the effects of new or modified Intent without committing, use *Test Drive*.

Intent Scores

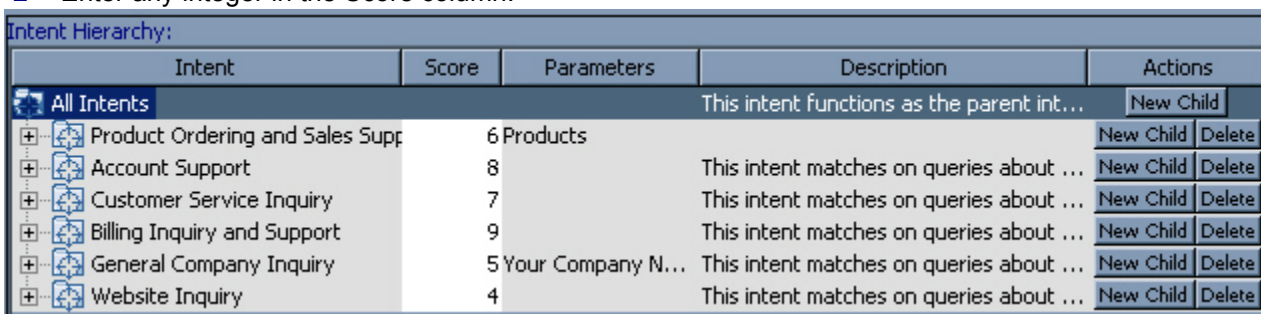
You can assign scores to Intents positioned as the highest-level nodes in the Intent Hierarchy. Each child node inherits the hierarchy score value from its parent. The Rules Engine applies the hierarchy score so that Intents within more important categories are preferred to otherwise equally-matched Intents from less important categories and are displayed first in the search results, above the other equally-matched Intents.

To specify the Hierarchy Score:



1 Select the **Score** column of a top level node from the Intent Hierarchy.

2 Enter any integer in the Score column.



Intent Tools

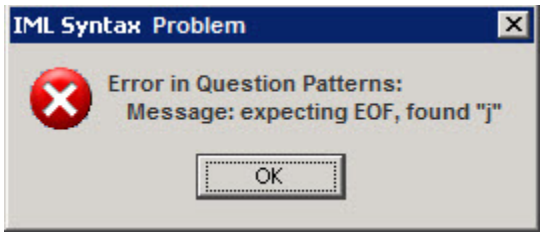
Before saving an Intent you can verify the function of the Intent by selecting an option from the **Tool** menu in the **Intent Response** window. Choose from:

Check IML on page 55

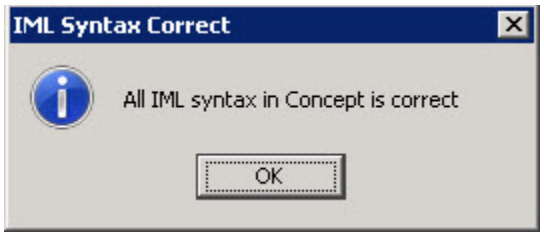
Validate Intent on page 56

CHECK IML

After selecting **Check IML** from the **Tools** menu Oracle Knowledge Intelligent Search automatically checks the validity of your IML. If a problem is detected, the **IML Syntax Problem** dialog displays explaining the IML error:



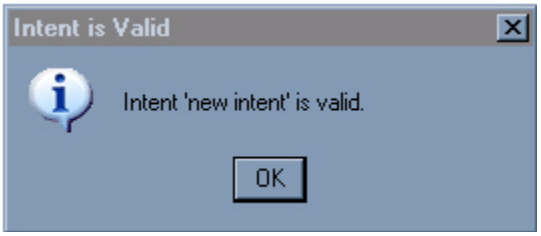
If no problem with the IML is detected, then the **IML Syntax Correct** dialog displays:



VALIDATE INTENT

After selecting **Validate Intent** from the **Tools** menu, Oracle Knowledge Intelligent Search automatically checks the validity of your Intent. If a problem is detected, the **Invalid Value** dialog displays explaining the error with the Intent

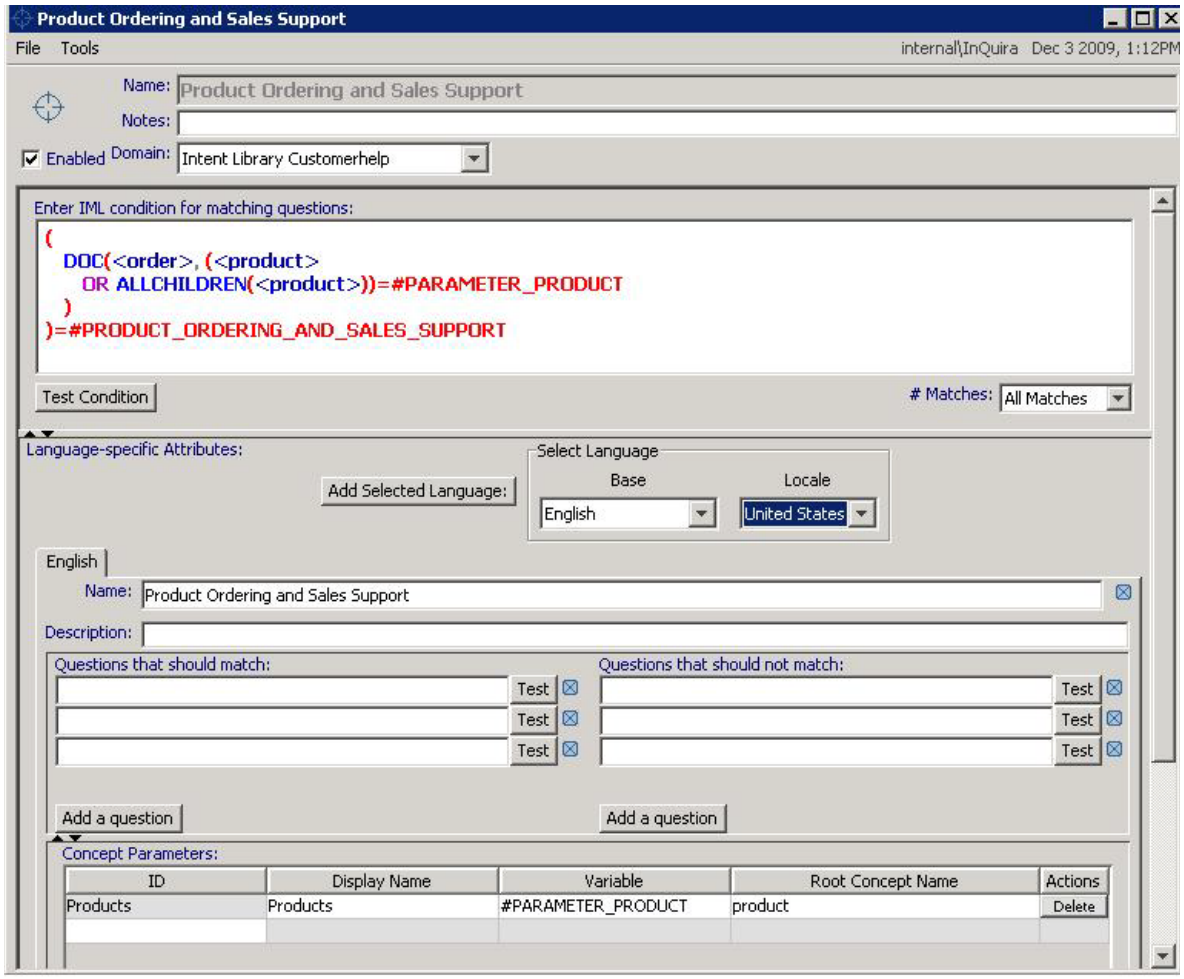
If the Intent is valid the following **Intent is Valid** dialog displays.



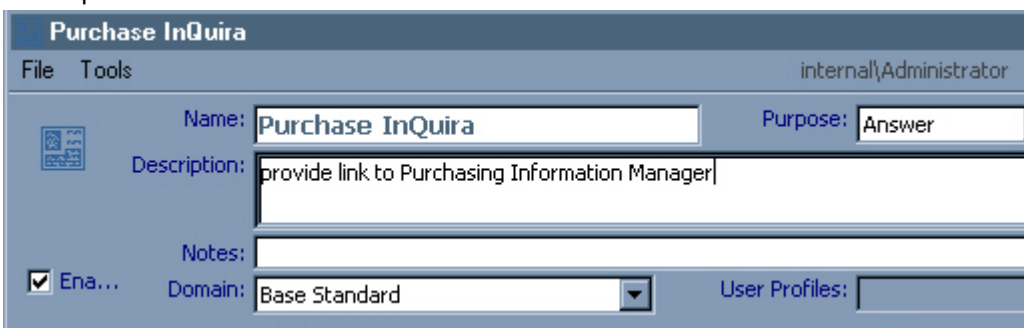
Example of Using an Intent and Creating a New Intent Response

Suppose that after looking at Oracle Knowledge Analytics we found many user queries for information regarding purchasing Oracle Knowledge. To satisfy this query we decide to create a search result that provides the user with quick and direct access to purchasing information. To do this we could create a Managed Answer Rule, however we know that the Intents that were installed with Oracle Knowledge contain an Intent that already captures user queries for purchasing information.

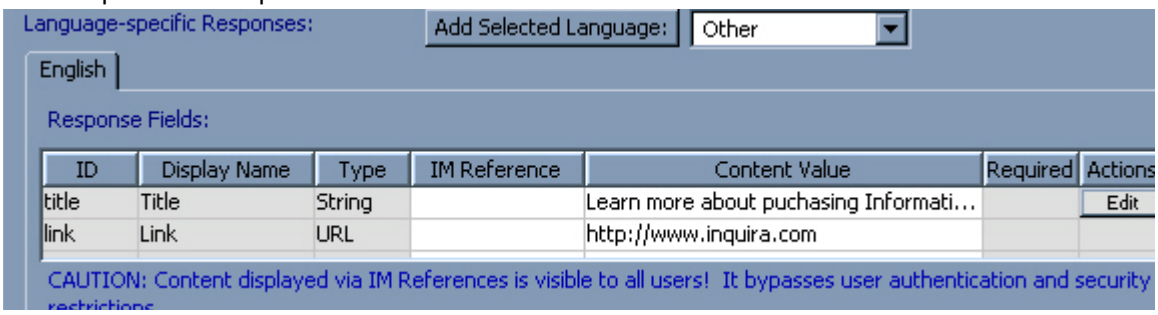
Important! While Dictionary Manager contains question/intent testing functionality, it does not all of the same processes, such as spell checking, that are contained in Test Drive. For the most accurate test of how intents fire in the customer UI, use Test Drive to test. If you use Dictionary Manager, you may get incomplete or different results, depending upon the question.



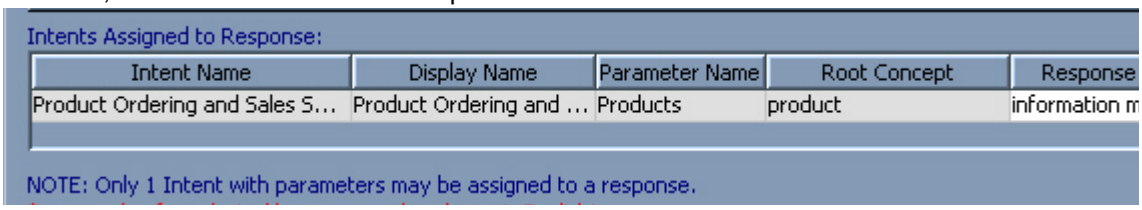
Using this Intent, versus a Rule, minimizes our work as the Condition is already established using IML. Instead, we only need to create the Action. To do this we must create a new Intent Response. From Workbench we select **Create a new Intent Response**. We name the Intent Response and add any additional information as explained above.



We then complete the Response fields as described above.



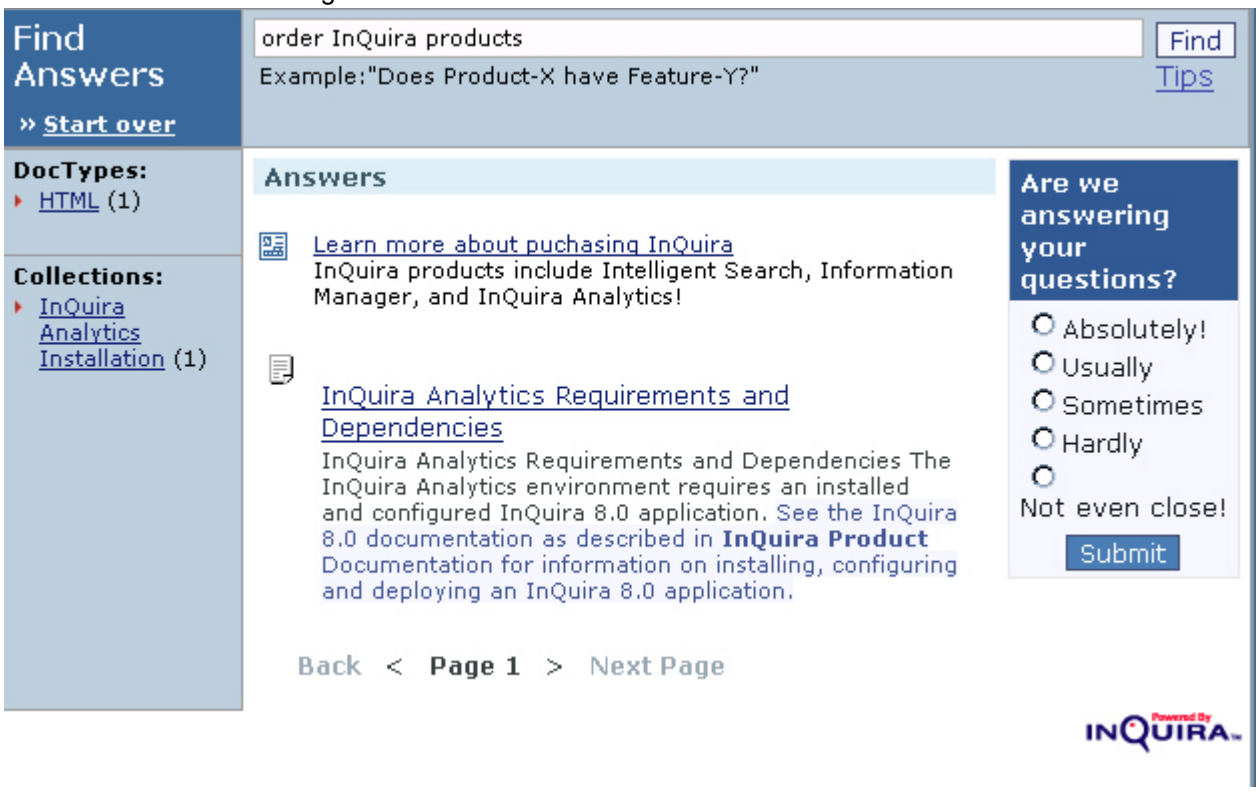
After entering this information, the last step is to assign the Intent to the Intent Response. From the **Intent Name** column, select an Intent from the drop down menu.



In this example, we select the Product Ordering and Sales Support Intent that has a Root Concept associated with it. To utilize this feature we must enter a **Response Concept**, which must be a child of the Root Concept <products>. We select the Concept <information manager> which we have made a child of <products> by establishing a “is a type of” relationship.

After saving the Intent Response, we expect any time a user queries for information regarding ordering information manager the Intent triggers and the Intent Response displays on the search results page.

We check the results using Test Drive.

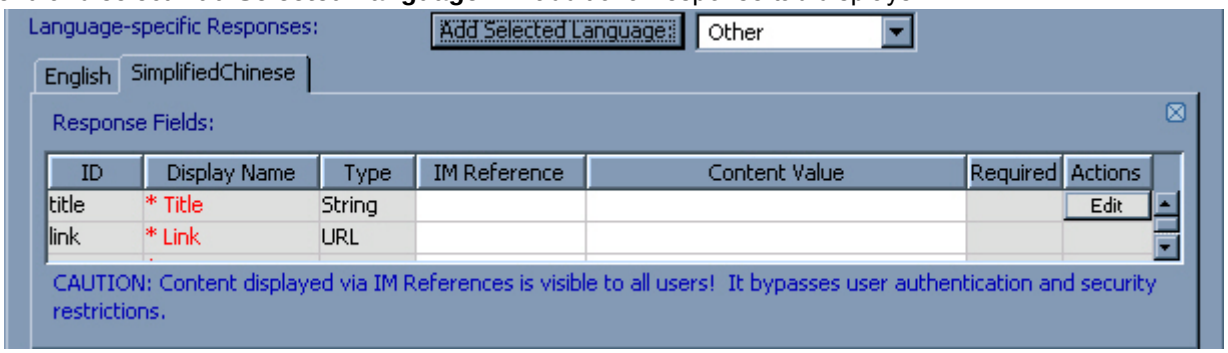


As expected, the first answer returned is the result of the newly created Intent Response.

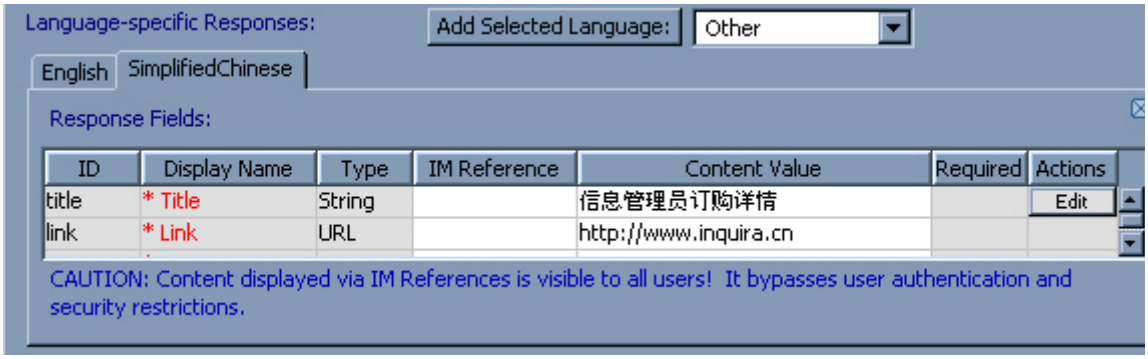
ADDING AN ADDITIONAL LANGUAGE INTENT RESPONSE

Continuing to use the Product Order and Sales Support Intent above we create an Intent Response that displays on a Chinese-based search results page.

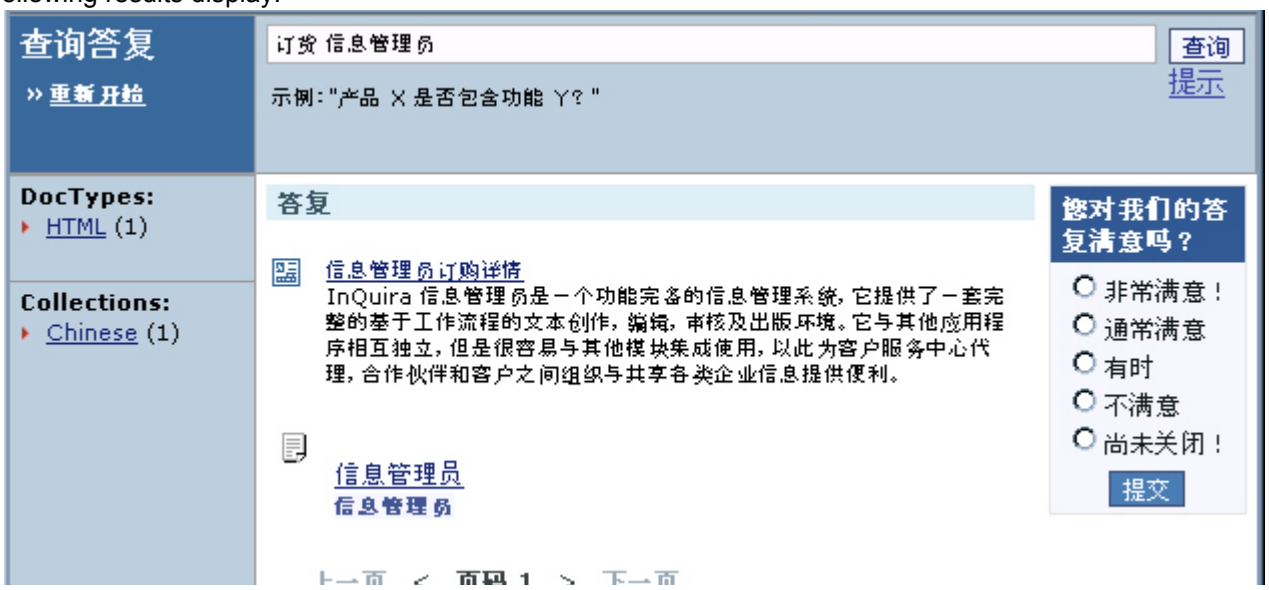
Using the Purchase Oracle Knowledge Intent Response created above, select Chinese from the drop down menu and select **Add Selected Language**. An additional response tab displays:



In the title field we enter the Chinese equivalent to “learn more about purchasing Information Manager” and a link to the associated website.



After saving, we check the results using Test Drive, making sure the Language is set to Chinese. The following results display:



As expected the Oracle Knowledge Intelligent Search results page displays the Intent Response.

Rules

Rules are Dictionary objects that determine how Oracle Knowledge Intelligent Search interprets and responds to user queries. Rules correspond to if/then statements, so that, if the specified Conditions are satisfied, then the specified Action occurs. Rules are used by the Rules Engine to determine the response criteria for a user query and to narrow search results based on user context and your company's content.

Creating Rules in Dictionary Manager allows Oracle Knowledge Intelligent Search to:

- Return the best possible answers by establishing precedence and scoring criteria.
- Return Managed Answers to satisfy users request and your company's needs.
- Restrict returned results based on set conditions and content.

There are various types of Rules that can differ in terms of intricacies. This section focuses on basic Managed Answers and basic Search Component Rules using a specialized language designed by Oracle Knowledge to identify language patterns. The Oracle Knowledge Matching Language, IML, is a proprietary language that is similar to regular expression languages in its logic. It is used capture the variations in written language that can be used to express requests. For more information, see *Oracle Knowledge Intelligent Search Language Reference*.

Another language used in Rules is one that allows the manipulation of strings. The Variable Instantiation Language, or VIL, is designed as a series of replacement operations where a particular string captured in the statement may get, for example, uppercase, lowercase, or replaced by a corresponding string in a list. For more information see *Oracle Knowledge Intelligent Search Language Reference*.

For information on the features of Rules listed on the Advanced menu, please see *Oracle Knowledge Intelligent Search Optimization Guide*.

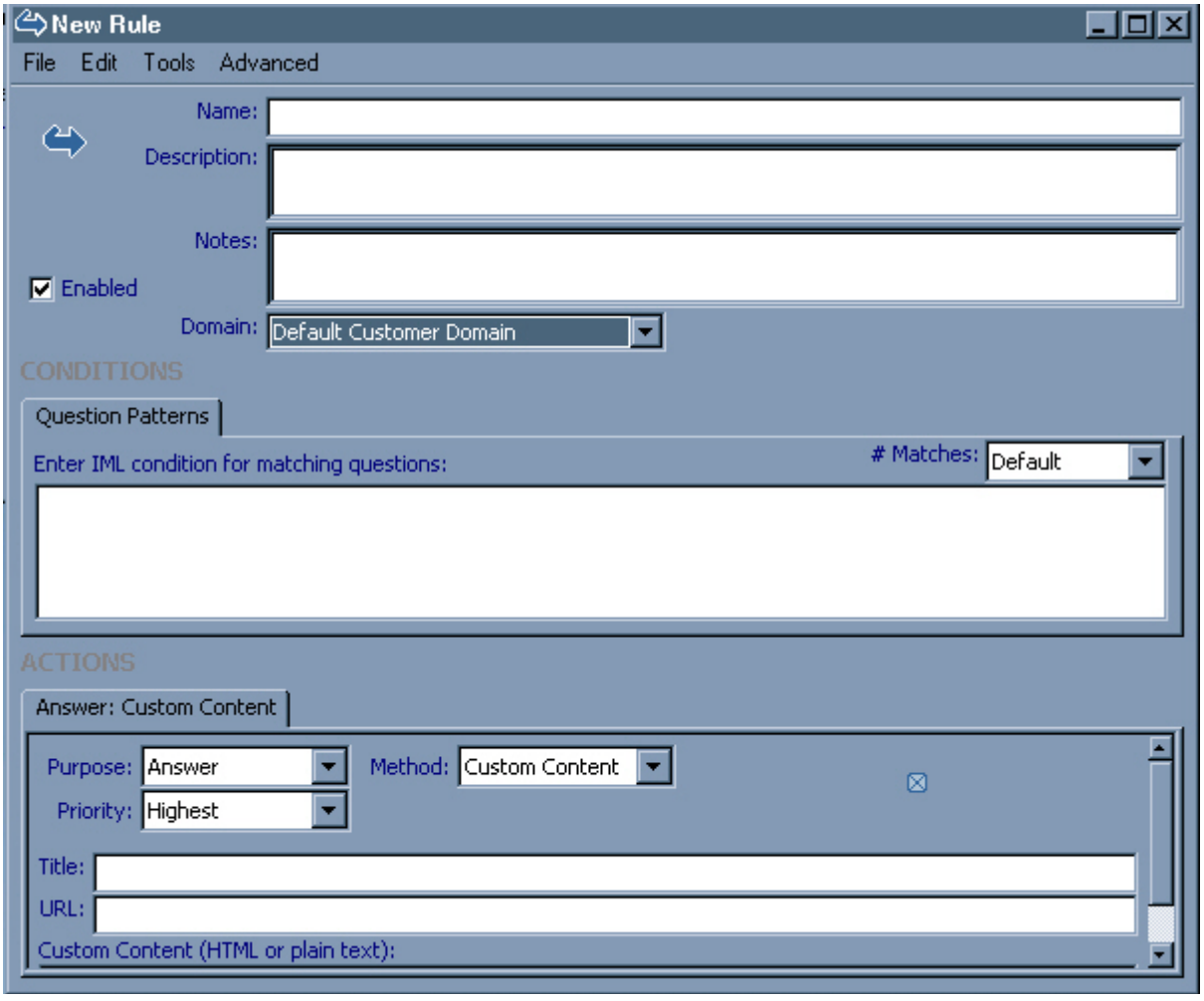
Create a New Rule- Managed Answer Rule (default Custom Content)

By creating new Rules, you allow Oracle Knowledge Intelligent Search to restrict what kind of results are returned based on set conditions and corresponding actions. *Managed Answers*, also referred to as *Best Answers* or *Custom Content Rules*, are used to return a specific answer for a specific set of query types. They are used typically to supplement missing content, for example if your company is lacking documents that accurately answer a type of query, or as a way to ensure business critical content displays in addition to regularly displayed search results. Managed Answers operate independent of regular search, and once triggered, are returned regardless of the other results returned by Oracle Knowledge Intelligent Search. Managed Answers usually appear as the first returned result on the search results page, or they can be designated to appear in a particular portlet.

To create a new Managed Answer Rule:

1 In Dictionary Manager, select the icon or **New Rule** from the File menu.

The Rule window displays with the default Action as *Custom Content*:



2 Name the Rule.

It is best practice to name rules in a consistent manner. We recommend starting all Managed Answers with BA, for best answer, or MA for managed answer.

3 Fill in the **Description** field. (Optional.)

The **Description** field has no impact on Oracle Knowledge Intelligent Search results. It is used to clarify the intended function of the Rule among users.

4 Enter any information regarding the Rule in the **Notes** field. (Optional).

The **Notes** field has no impact on Oracle Knowledge Intelligent Search results. It is used to track comments on Rule modifications among users.

5 Check the **Enabled** box. (This is selected by default.) If the **Enabled** box is not checked, Oracle Knowledge Intelligent Search ignores the Rule.

6 Select the **Domain** from the drop down list. Only Domains you have permissions for display. The **Domain** field is set to the most recently saved Domain by default. Set the Domain to your company Domain or to the "Default Customer Domain." For more information on Domains, see the following section *Domains and Domain Lists* on page 83.

7 Complete the **Conditions** section.

Thinking of Rules as if/then statements, the information provided under the Condition section corresponds with the 'if' portion. By default, the **Question Patterns** Tab displays. Other tabs: **Business Conditions** and **Deferred Conditions** can be added by selecting from the **Advanced** menu. These options are discussed in *Oracle Knowledge Intelligent Search Optimization Guide*.

a Under **Question Patterns**, enter the IML expression used to match the variety of words and phrases that can be used to express a particular user query for information or assistance. This IML is used to match a question and then trigger the **Actions** defined below. Parts of the IML expression entered in the question pattern display as various colors if entered correctly. For more information see *Finding Dictionary Objects* on page 95.

The simplest question pattern you can enter is a string in double quotes. for example, by entering "cat" in the Question Pattern, any time a query contains the word cat (though not cats) the Managed Answer Rule fires, and the custom content appears at the top of the search results page. You can create more complicated Question Patterns using IML, as explained in the *Oracle Knowledge Intelligent Search Language Reference*.

b Select the number of matches for the IML expression to find within a given request from the drop down menu.

This value controls how many times a rule may fire if it matches more than one part of a query. This is necessary only for more complex rules; for most purposes this setting is the default.

Default	Specifies to process only one match. (Default)
One Match	Specifies to process only one match.
Most Matches	Specifies to process most matches, not including overlapping.
All Matches	Specifies to process all matches.

8 Complete the **Actions** section. Thinking of Rules as if/then statements, the information provided under the Action section corresponds with the 'then' portion. By default, the **Answer: Custom Content** tab displays. The Answer tab varies in relation to the method selected from the drop down menu. The simplest Managed Answer to create is using the Custom Content method, and the following steps correspond to this method. For more information on Answer Action Methods, please see *Oracle Knowledge Intelligent Search Optimization Guide*.

a Select a **Purpose** from the drop down menu.

A purpose, also referred to as a portlet, corresponds to display characteristics defined in the User Interface, enabling you to establish consistent, focused and targeted presentation for various types of content. For specific information on purposes, see *Purpose (portlet)* on page 143:

b Select a **Priority** from the drop down menu.

The Priority specifies the ranking of Managed Answers, in particular when multiple Managed Answer rules are returned based on the same criteria. Choose from:

Highest	Displays this Managed Answer first, assuming all criteria for the following Managed Answer are the same.(Default.)
High	Displays this Managed Answer second, assuming all criteria for the following Managed answers are the same.
Medium	Displays this Managed Answer third, assuming all criteria for the following Managed answers are the same.
Low	Displays this Managed Answer fourth, assuming all criteria for the following Managed answers are the same.
Lowest	Displays this Managed Answer fifth, assuming all criteria for the following Managed answers are the same.

c Select a **Method** from the drop down menu.

The Method specifies how Oracle Knowledge Intelligent Search executes a specified action. Choose from:

Custom Content	Creates a response using custom content, including text, title, and link. (Default.)
Specific Excerpt	Creates a response using a specified excerpt.
Search	Launches a search in the indexed unstructured content.
Structured Query	Launches a search against a structured data source.
Glossary	Creates a response from the glossary entry associated with the specified term.
Plug-In	Creates a response using an external method, based on configured plug-in code.
Process Wizard	Creates a response using a specified custom process wizard.

Note: To create the simplest Managed Answer, select the **Custom Content** method and follow the directions below.

d Enter a title in the **Title** field and a URL in the **URL** field. (Optional, or use the Custom Content field described below.)

Type the title as you want it to display as hypertext on the Oracle Knowledge Intelligent Search results page.

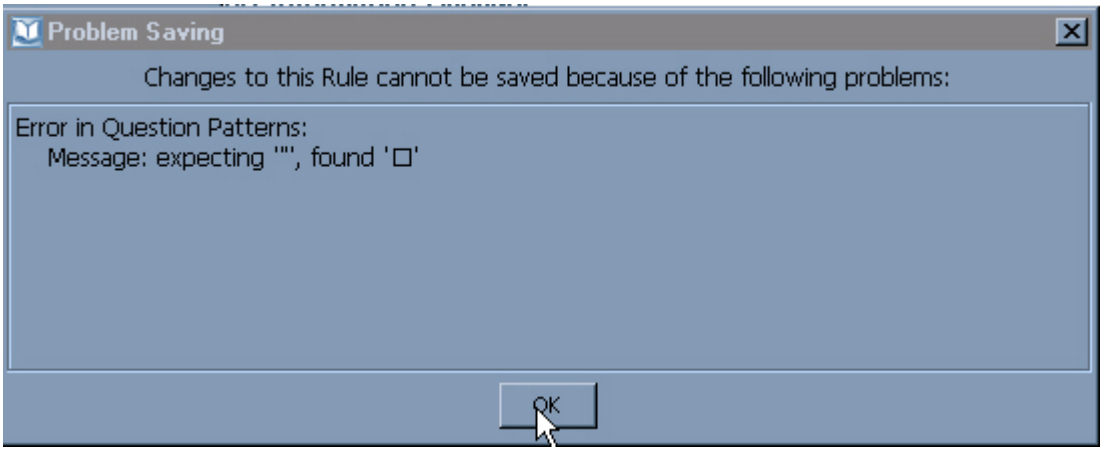
Type the full URL link associated with the hypertext title specified above.

e Enter HTML or plain text in the Custom Content field.

Use this field to create custom text or HTML to display at the top of the Oracle Knowledge Intelligent Search results page.

9 Save the Rule.

Select **Save** from the **File** menu in the Rule window. If your Rule contains an invalid format, the following **Problem Saving** dialog displays:



If this error displays and you are unsure of the problem select a test from the **Tools** menu, see below in *Rule Tools* on page 72.

For information on options under the **Advanced** menu of the Rule window, please see *Oracle Knowledge Intelligent Search Optimization Guide*.

Note: A new Rule does not affect live Intelligent Search results until you *Commit Dictionary Changes to Central Repository (vcroot)* on page 97, and the changes have been synchronized to the production environment, for more information see *Oracle Knowledge Intelligent Search Administration Guide*. To see the effects of new or modified Rules without committing, use *Test Drive*.

Create a New Rule- Search Component Rule

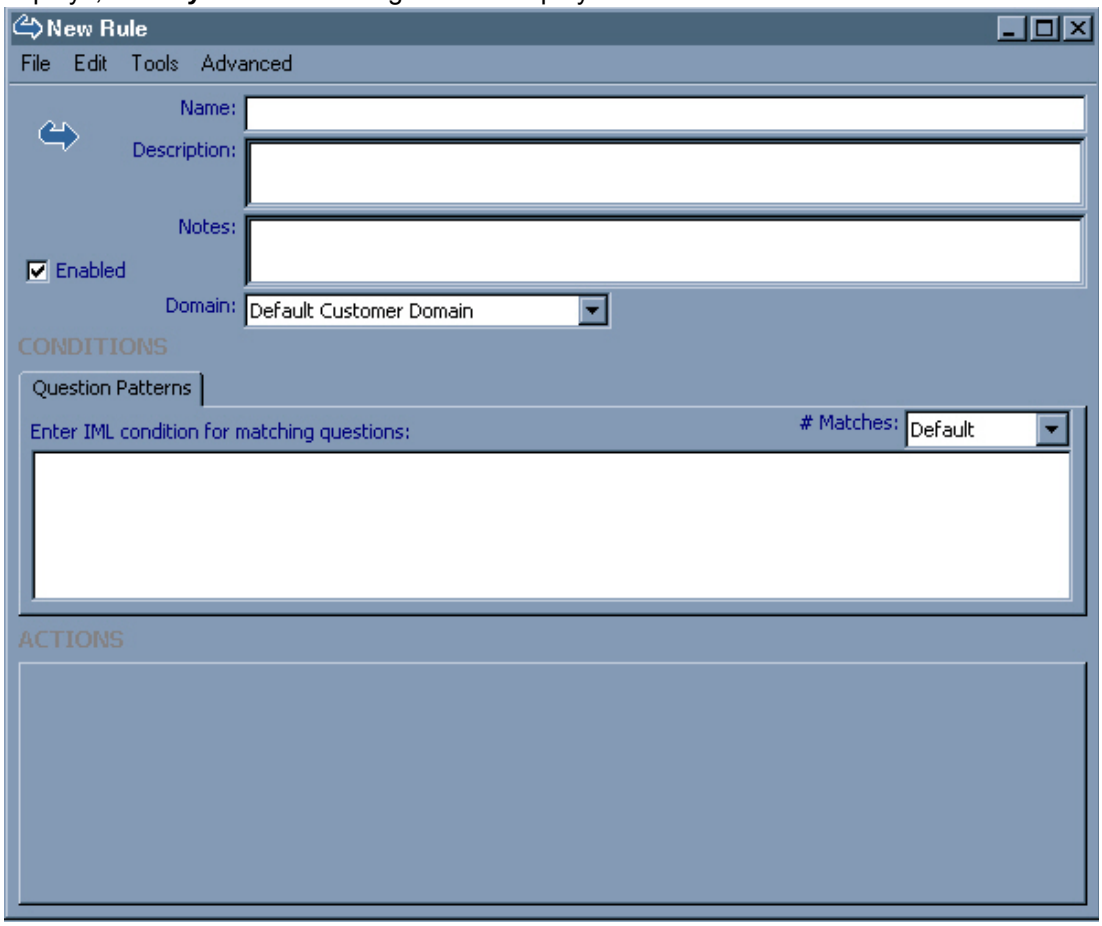
By creating new Rules, you direct Oracle Knowledge Intelligent Search to restrict which results are returned based on set conditions and corresponding actions. Search Component Rules are the primary method of specifying information retrieval for both unstructured and structured data. Search Component Rules are used to define search criteria based on matching some portion, or range, of a user query using IML or SQL expressions.

Search Component Rules affect the search results by adding weight to the matches they specify. Oracle Knowledge determines which documents to provide as the best answers to a user query by assigning a score to a document if it matches the search components that are created from the query. The search components are obtained by, for example, synonyms of the word or phrase that is being searched on, or its clusetermates. Each match between these search components and documents is associated with some numeric score, and each match contributes to the overall score the document receives. Once these scores are tallied, the documents are ranked accordingly, and the highest ranking document is returned as the best match.

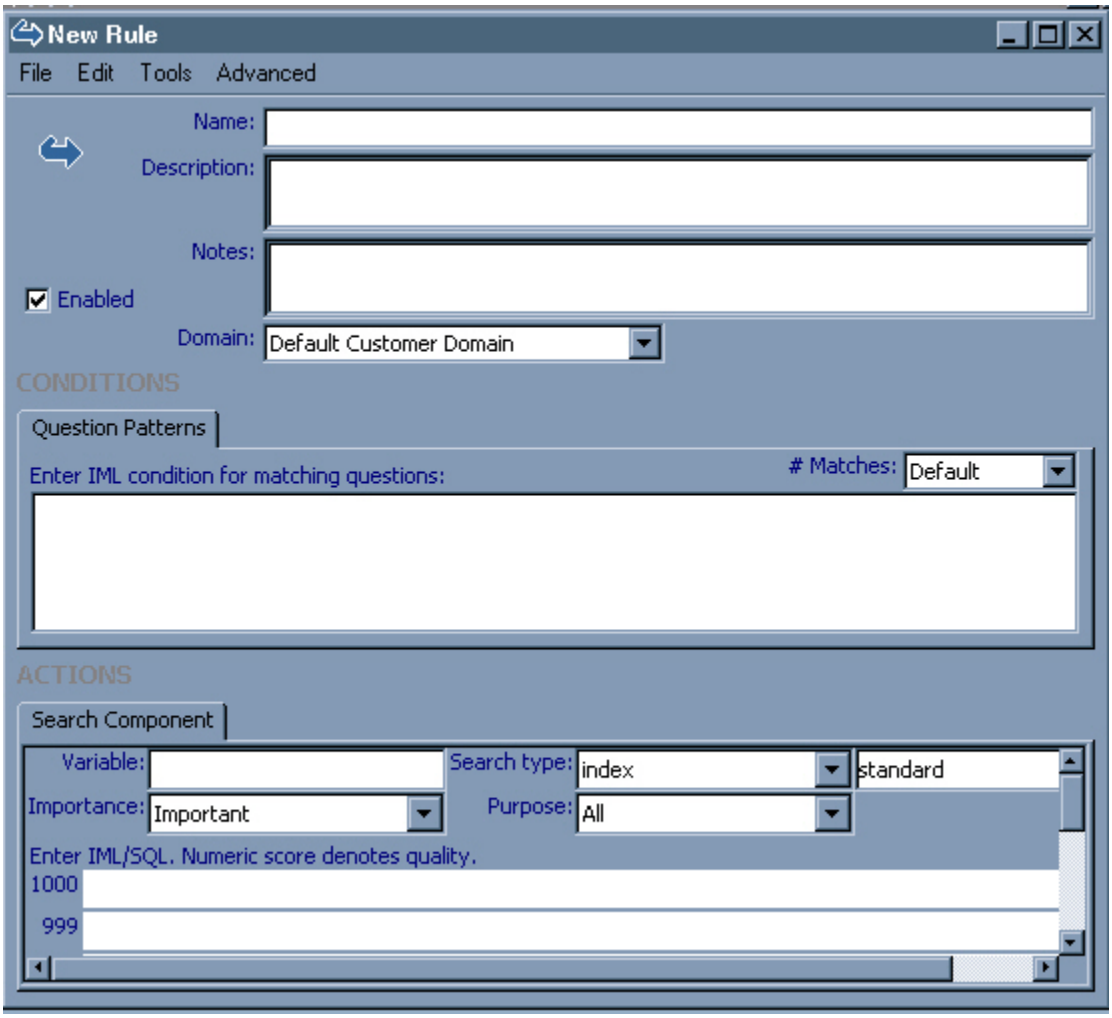
Creating a Search Component Rule allows us to assign additional points to matching documents, thus boosting the overall ranking of the document. For this reason, Search Component Rules, are an effective way to tune your dictionary and maintain the best Oracle Knowledge Intelligent Search results.

1. In Dictionary Manager, select the icon or **New Rule** from the **File** menu.
   The Rule window displays with the default Action as Custom Content.
2. Replace the **Answer Action** tab with a **Search Component** Tab.

a Delete the **Answer: Custom Content** tab by selecting the ⊠ icon. The **really delete?** dialog displays, select **yes**. The following window displays with no **Actions** tab:



b Under the **Advanced** menu, select **Add Search Component**. A Search Component Tab appears in the Rule window:



3 Name the Rule.

It is best practice to name rules in a consistent manner. We recommend starting all Search Component Rules with SC, for search component.

4 (Optional.) Fill in the Description field.

The Description field has no impact on Oracle Knowledge Intelligent Search results. It is used to clarify the intended function of the Rule among users.

5 (Optional.) Enter any information regarding the Rule in the Notes field.

The Notes field has no impact on Oracle Knowledge Intelligent Search results. It is used to track comments on Rule modification among users.

6 Check the Enabled box. (This is selected by default.) If the Enabled box is not checked, Oracle Knowledge Intelligent Search ignores the Rule.

7 Select the Domain from the drop down list. Only Domains you have permissions for display. The Domain field is set to the most recently saved Domain by default. For more information on Domains see the following section *Domains and Domain Lists* on page 83.

8 Complete the Conditions section.

If we think of Rules as if/then statements, the information provided under the Condition section corresponds with the 'if' portion. By default, the Question Patterns Tab displays. The other tabs: Business Conditions and Deferred Conditions can be added by selecting Advanced and are discussed in *Oracle Knowledge Intelligent Search Optimization Guide*.

a Under Question Patterns, in the space provided, enter the IML expression used to match the variety of words and phrases that can be used to express a particular user query for information or assistance. This IML is used to match a question and then trigger the Action defined below. Parts of the IML expression entered in the question pattern display as various colors if entered correctly. For more information on the color code please see *IML Color Code* on page 144.

Search Component Rules must contain a variable. Variables are place holders with no intrinsic value. Their value is assigned by the matching process. For example, if the IML statements assigns the string that matches the concept <information manager> to the variable A (<information manager> =A), if the Concept is matched to the string "information manager: the value assigned to A is "information manager". If the Concept is matched to "IM" then the value assigned to A is "IM." Capturing strings in this was can be very useful. For instance, you may choose to search for the exact string in the documents. Or you may want to tell the user that their search for "IM" or "information manager" retrieved no results. The language that allows us to manipulate the values assigned to a variable is called the Variable Instantiation Language, or VIL. VIL is discussed in *Oracle Knowledge Intelligent Search Language Reference*.

There are two types of variables: Local and Global. Local variables are created within a Rule and disappear as soon as the Rule completes its action. Local variables are any letter or word that has been assigned a value through the *assignment operator*, X. It is common practice to use a set the variable equal to a capital letter. By contrast, Global variables are preserved until the entire search process is complete. Rules can refer to a Global variable created in another Rule. Global variables are indicated with a # appearing before the capital letter used to define the variable.

The simplest question pattern you can create is using a string in double quotes set equal to a variable. for example, *cat*=A, whereas any time a query contains the word *cat* (though not *cats*) the Search Component Rule fires, and the action associated with the variable A receives a value,. Once the condition is satisfied and the rule is fired, the Action portion of the rule is activated. You can create more complicated Question Patterns by understanding IML as explained in the *Oracle Knowledge Intelligent Search Language Reference*.

b Select the number of matches for the IML expression to find within a given request from the drop down menu.

This value controls how many times a rule may fire if the question pattern matches more than one part of a query. This is necessary only for more complex rules; for most purposes this setting is the default.

Default	Specifies to process only one match. (Default.)
One Match	Specifies to process only one match.
Most Matches	Specifies to process most matches.
All Matches	Specifies to process all matches.

9 Complete the **Actions** section. Thinking of Rules as if/then statements, the information provided under the Actions section corresponds with the 'then' portion.Select the **Search Component** tab as described above. The Search Component tab varies in relation to the Importance level selected from the drop down menu.

a Enter a variable.

In the Question Pattern above, you set an IML statement equal to a variable (usually A). In the Variable field enter this variable.

b Select a level of importance from the **Importance** drop down menu.

The **Importance** setting specifies the score value range of the Search Component Rule. The range you select here affects the values assigned to matches in the IML/SQL field below. Choose the Importance level from:

Required	Range descends from 100,000 scoring points.
Very Important	Range descends from 10,000 scoring points.
Important	Range descends from 1,000 scoring points. (Default.)
Minor	Range descends from 100 scoring points.
Trivial	Range descends from 10 scoring points.

Note: It is best practice to set the Importance level to **Trivial**, and then raising the level until the desired effects in the ranking of the search results are obtained.

c Select a search type and search subtype from the drop down menus.

First choose a search type from:

Index	Use this search type to define criteria using IML expressions for searching the unstructured content indexes in the Content Store (Default).
SQL	Use this search type to define criteria using SQL queries for retrieving information form configured structured data sources.

In most cases, you select **Index**, intended for natural language search against indexed documents. After you have selected the search type, choose a search subtype from the drop down menu. Your choices vary depending on which Search type you choose.

If you choose **Index** as your search type, the Rules Engine uses the subtype value to order the assigned Search Components within the action list. Choose the subtype from the following:

ranking	Used by the Rules Engine to order the ranking subtypes after the standard subtypes within the action list.
standard	Used by the Rules Engine to order the standard subtypes first within the action list. (Default.)
unspecified	Used to prevent the Rules Engine from ordering unspecified subtypes.

If you choose SQL as your search type, the Rules Engine uses the subtype value to populate the SQL statement that it generates to query the data source. Choose the subtype from the following:

- condition
- field
- binary operator
- table
- order
- limit
- unspecified

d Select a **Purpose** from the drop down list.This is typically set to **Answer**.

Purposes, or portlets, correspond to display characteristics defined in the User Interface, enabling you to establish consistent, focused and targeted presentation for various types of content. For more information, see *Purpose (portlet)* on page 143.

e Enter IML/SQL statement(s).

Use the I**ML/SQL** field to enter multiple IML, VIL, or SQL statements that Oracle Knowledge Intelligent Search uses to ensure the best possible answers to a user query. Enter the statements in order of priority. Intelligent Search uses this order with the Search Component Rule to specify a set of prioritized search instructions in response to matching the Question Pattern described above. The scoring values assigned to the IML/SQL statements correspond to the Importance level set above. The first level indicates the preferred contents to display on the search results page.

- You can enter as intricate statement as desired by using IML, VIL, and SQL. For more information on creating IML or VIL statements see *Oracle Knowledge Intelligent Search Language Reference*. The IML and/or VIL entered in this section displays as various colors if entered correctly. For more information on the color code please see *IML Color Code* on page 144.

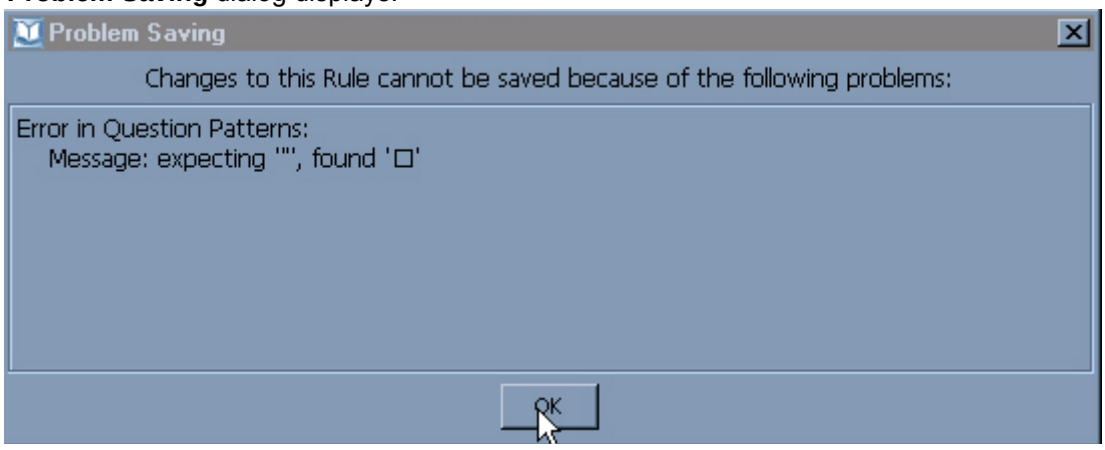
10 (Optional.) Set Supplement and Subsume.

Select **Supplement/subsume** from the **Advanced** menu in the Rule window. The Supplement and Subsume boxes appear in the Search Component tab.

- Check the **Supplement** box if you want the Rule to work in addition to other Search Component Rules triggered by the same portion of the user query. This is selected by default. If the box is unchecked, the Search Component Rule stops other Search Component Rules from matching, making them ineffective in the scoring process.
- Check the **Subsume** box if you want to ignore other Search Component Rules triggered within the same portion of the user query as the current Rule. This is cleared by default. If cleared, multiple Search Components may fire covering the same portions of the user query.

11 Save the Rule.

Select **Save** from the **File** menu in the Rule window. If your rule contains an invalid format the following **Problem Saving** dialog displays:



If this error displays and you are unsure of the problem, select a test from the **Tools** menu, see *Rule Tools* on page 72.

For information on options under the **Advanced** menu of the Rule window, please see *Oracle Knowledge Intelligent Search Optimization Guide*.

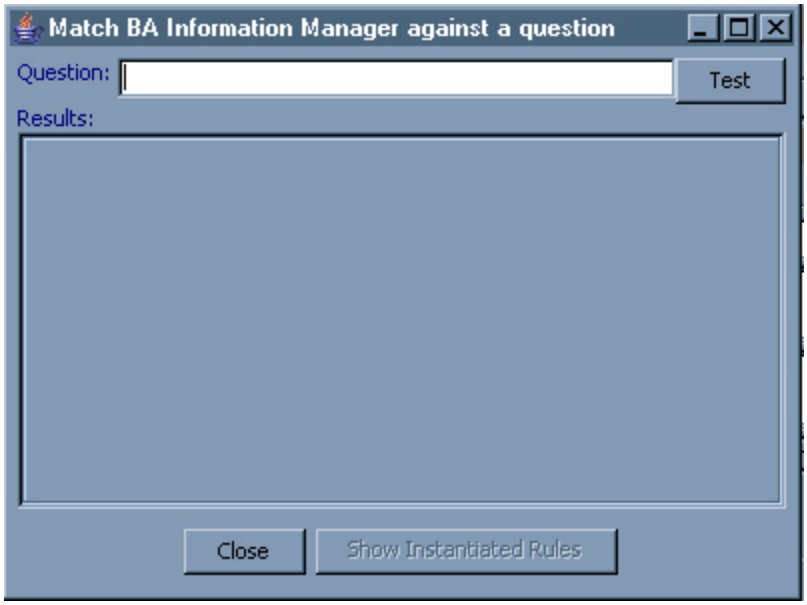
Note: A new Rule does not affect live Intelligent Search results until you *Commit Dictionary Changes to Central Repository (vcroot)*, and the changes have been synchronized to the runtime environment, for more information see *Oracle Knowledge Intelligent Search Administration Guide*. To see the effects of new or modified Rules without committing, use *Test Drive*.

Rule Tools

Before saving a newly created Rule, you can verify the function of the Rule by selecting an option from the **Tool** menu in the **Rule** window. Choose from: *Match Against a Question*, *Check IML*, or *Validate Rule*.

MATCH AGAINST A QUESTION

After selecting **Match Against a Question** from the **Tools** menu the Match Rule Name Against a Question box displays:

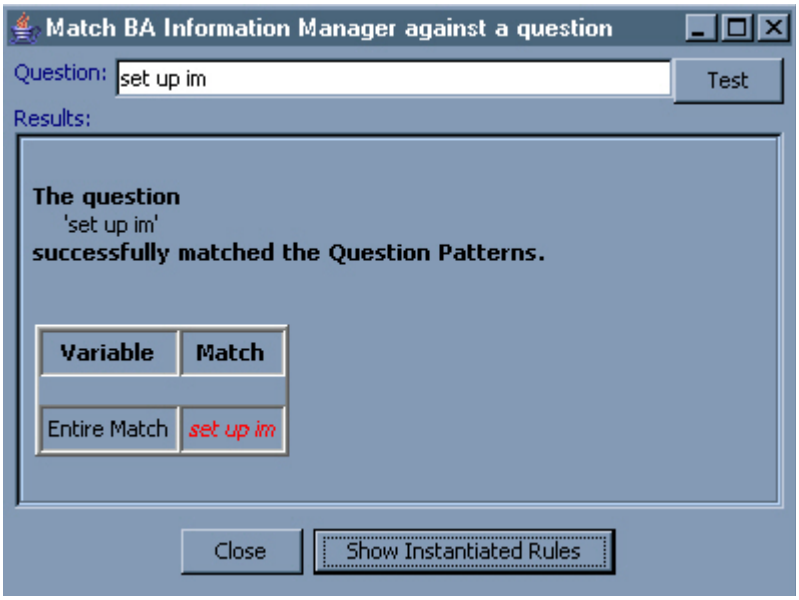


1. Enter a query that you expect your newly created Rule to satisfy in the **Question** field.
2. Select **Test**.

Upon testing the query, the Match Against a Question Results field displays one of the following messages:

- The Question did not match the Question Patterns.
- The Question successfully matched the Question Patterns.

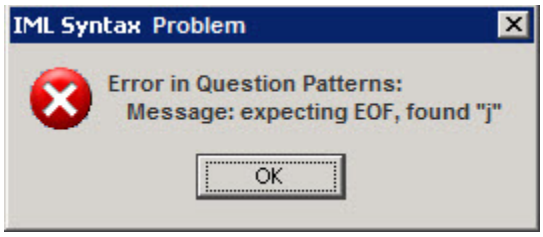
If the query matches successfully the following displays:



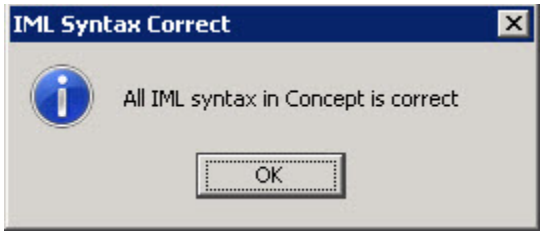
3 Select **Show Instantiated Rules** to see the Rule.

CHECK IML

After selecting **Check IML** from the **Tools** menu Oracle Knowledge Intelligent Search automatically checks the validity of your IML. If a problem is detected, the **IML Syntax Problem** dialog displays explaining the IML error:

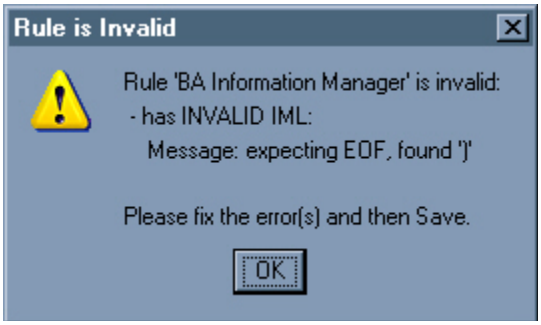


If no problem with the IML is detected, then the **IML Syntax Correct** dialog displays:

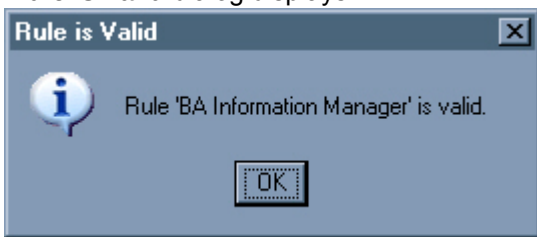


VALIDATE RULE

After selecting **Validate Rule** from the **Tools** menu Oracle Knowledge Intelligent Search automatically checks the validity of your rule. If a problem is detected, the **Rule is Invalid** dialog displays explaining the error with the Rule



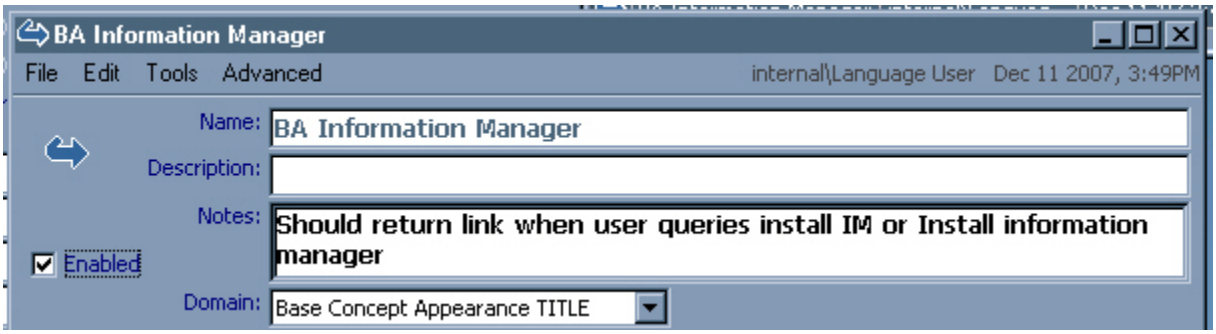
If the Rule is valid the following **Rule is Valid** dialog displays.



Example of Creating a Managed Answer

Based on the Oracle Knowledge business context ,we know that many users query for ‘install IM’. It is for this reason we create a Managed Answer to quickly and easily return the desired information to the user.

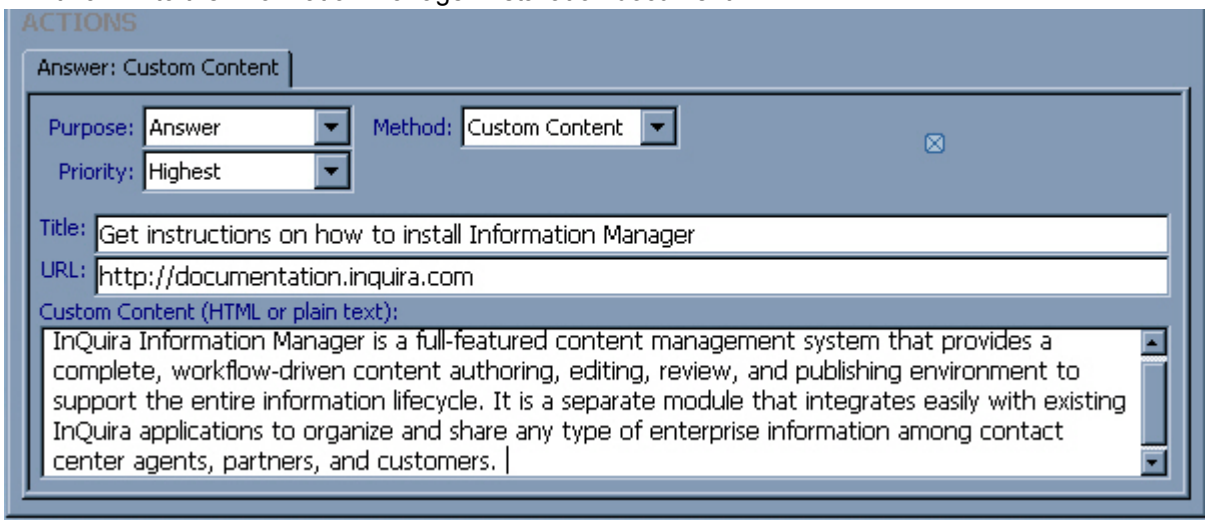
From Workbench, select **create a new Rule**. Name the rule and add any additional information, as explained above.



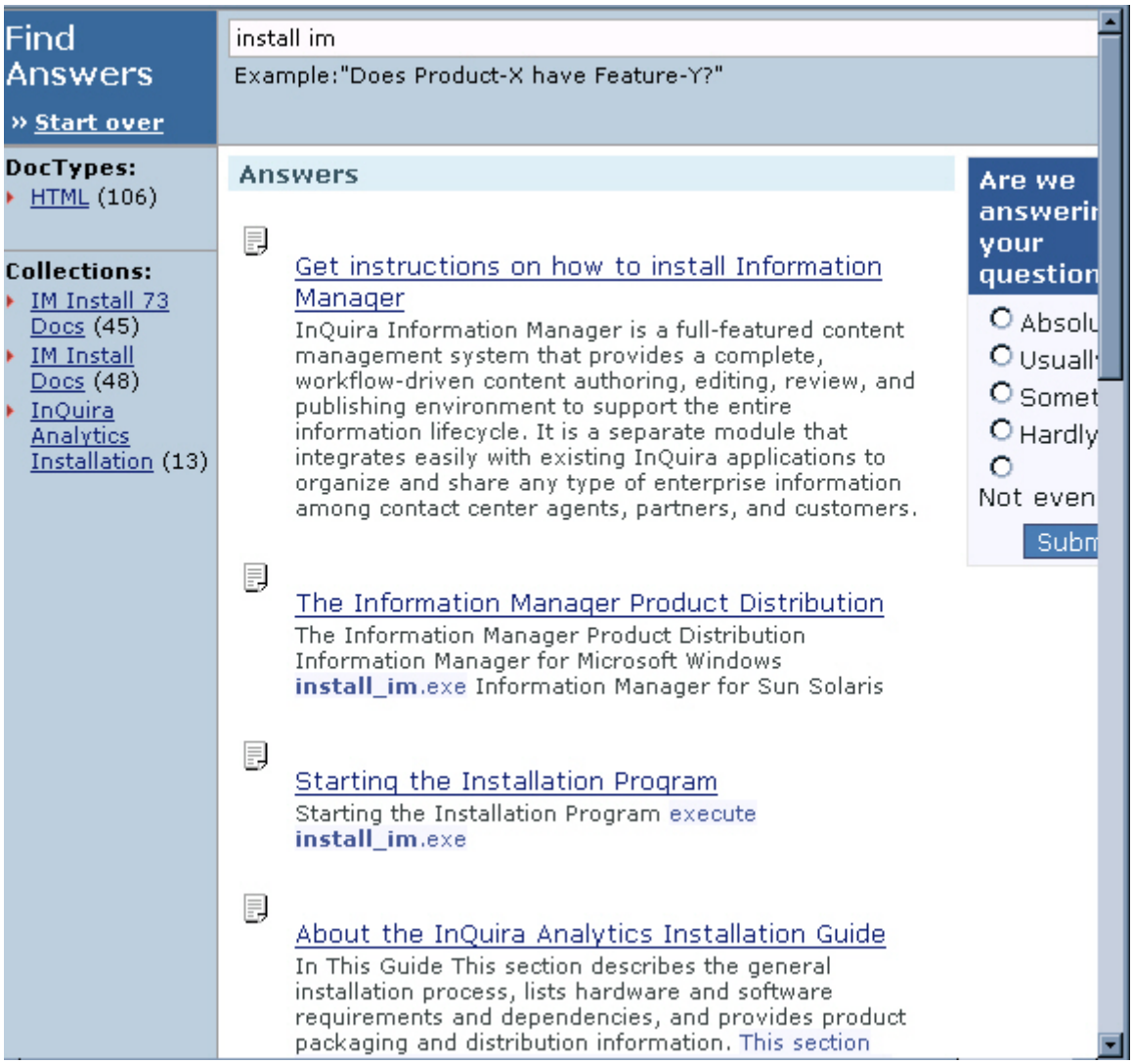
Next define the Question Pattern. We know that we want the Managed Answer to be returned anytime a user queries ‘Install IM’. We could enter a simple string; however, it is best to enter a more general pattern expressed in IML using the Concepts in order to capture the language variants of both install and IM that a user may query, such as ‘Set up Information Manager’. To do this, drag and drop both Concepts <verb.change.set up> and <noun.inquira.information_manager> into the **Question Pattern** field. This IML

requires the user query to contain two words: one a defined synonym to the Concept <set up> and the other a defined synonym to the Concept <information manager>.

After you have established the Question Pattern, define the actions as explained above. For this example, we assume that any time a user queries for information on installing Information Manager we want to provide them with a link to the Information Manager installation document.



Save your Managed Answer and check the effect in Test Drive. The following results are returned:



As expected, the first answer returned is the result of the newly created Managed Answer.

Example of Creating a Search Component Booster Rule

Booster Rules are Search Components or other Search Rules that boost content from Certain facets, documents, or types of documents.

It is best to reserve Booster Rules for very specific situations where, all else being equal, answers from certain parts of the content be returned before answers from other parts of the content. Otherwise, Booster Rules can cause strange responses or no responses to some queries. See *Booster Rules Issue* on page 78.

TYPES OF BOOSTER RULES

The form of a Booster Rule depends on the type of content that needs boosting. The following examples present general guidelines for various types of booster rules. They can all be modified to adapt to specific business needs.

Boosting a Facet

Use or modify the following to boost a facet.

Question Pattern

If there is no restriction on the type of question, Oracle recommends that you consider stating a fairly general restriction such as:

(BEGIN ((*) (#VeryImportantConcept OR #RequiredConcept) (*))=A END)

Action

Search Component. Variable: A. Recommended settings: Try Importance = trivial first, and raise only as needed; Check Supplement, uncheck Subsume. If "Classification.MyFacet" is the facet you'd like to boost, this suffices:

FACET("Classification.MyFacet")

If you need to boost several facets, but not equally, you can write multiple booster rules with varying Importance.

Boosting a Document or Document Type

Use or modify the following to boost a document or document type.

Question Pattern

If there is no restriction on the type of question, Oracle recommends that you consider stating a fairly general restriction such as:

(BEGIN ((*) (#VeryImportantConcept OR #RequiredConcept) (*))=A END)

Action

Search Component. Variable: A. Recommended settings: as above. If you can find a language characteristic X common to the documents of the desired type, you can write:

DOC(X)

where X can be something like

NEAR_6(("tutorial" OR "guide"),"revised")

or

TITLE("tutorial" OR "guide")

Use Service Browser to determine the best way to express X.

Boosting Documents with a Particular Version or Kind of a Product

When a user asks about a product without specifying a version (or other attribute, such as year), the customer wants answers to be returned first that are about a certain version or attribute (such as current year).

Suppose the preferred year is 2005. Assume also you have defined global variables for #modelYear to recognize years and #model to recognize the products or models in question.

Question Pattern

DOC(BEGIN (*)=A #model=B (*)=C END) ISNT DOC(#modelYear)

Action

Plugin=Restart Question. New question:

A->NOVAR(ACTION="")->STRING(NODUPS="false")} {B.EN} 2005 {C->NOVAR(ACTION="")->STRING(NODUPS="false")

BOOSTER RULES ISSUE

Before you create booster rules, you should be aware that Booster Rules may cause questions with no matches to suddenly start returning bad answers. For example, an off-the-wall question such as "Superman?" starts returning answers after the Booster Rule was introduced.

What is happening is that the query satisfies the Booster Rule question pattern, and some content gets enough of a score—thru the boost—to qualify as an answer.

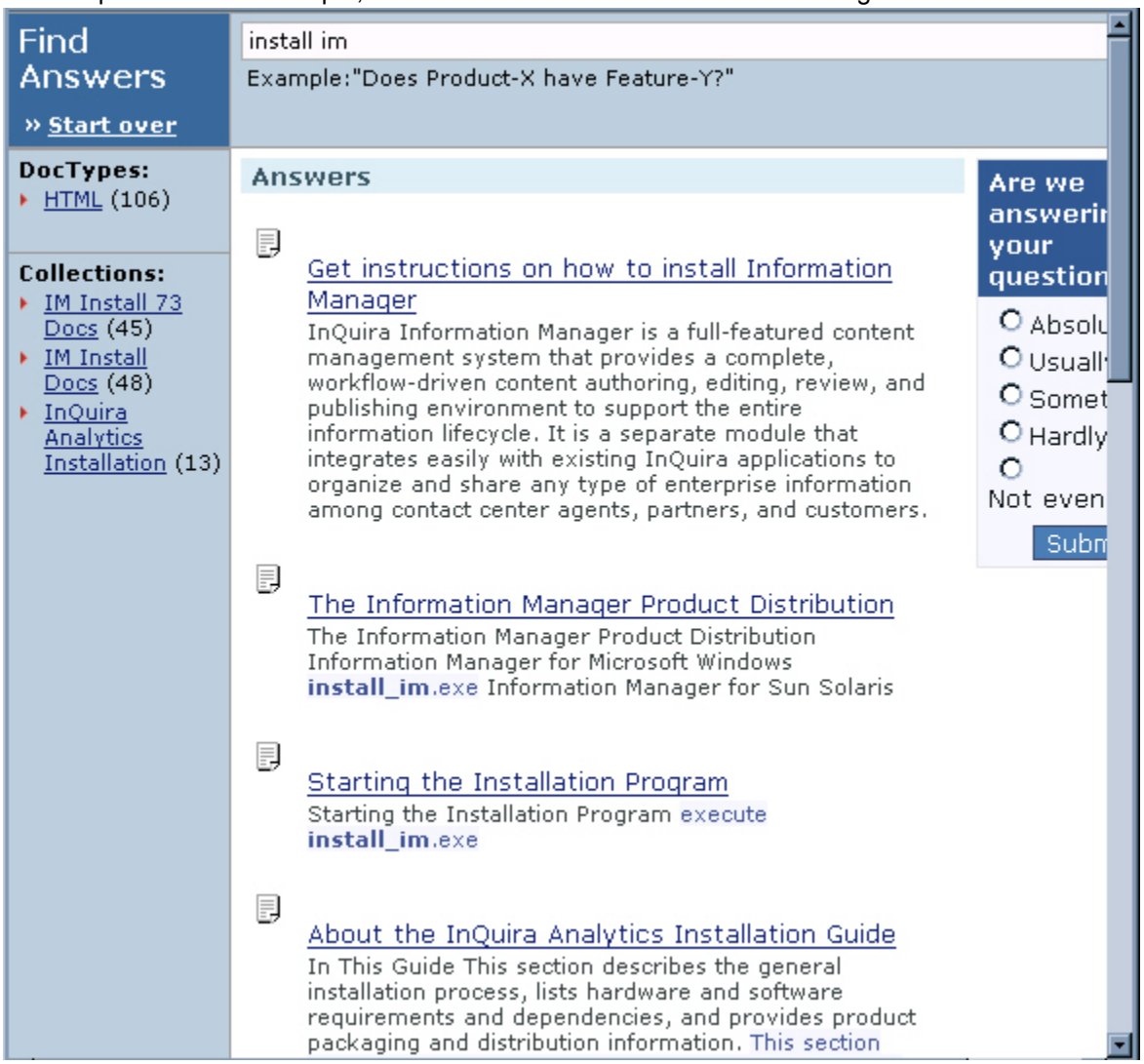
Solutions:

- Restrict the question pattern to avoid matching to questions that don't belong. For example, you can try excluding unknown words or string literals from satisfying the question pattern. Modify the question pattern to:

  (BEGIN ((*) (#VeryImportantConcept OR #RequiredConcept) (*))=A END) ISNT (#UnknownWord(1-) OR ("\"" + "\"")).
- Another cause may be unimportant medium level concepts that still satisfy the question pattern. If possible, change the level of these concepts to "low". If there many such concepts, it is probably faster to export them, use Excel to change their scores to 10, and then reimport them.
- Raise the minimum score for the Portlet, usually Answers. Go to Admin Tool, Dictionary Service > Answer Purpose > Answer Purpose. By default the minimum score is set to 0.0. Raise it just enough to offset the booster rule, probably to 0.2 or 0.3.

EXAMPLE

Oracle Knowledge has three major products: Intelligent Search, Information Manager, and Analytics. As these components are configured to work together, the documentation for each product contains references to the other products. For example, a search on “install im” returns the following results.

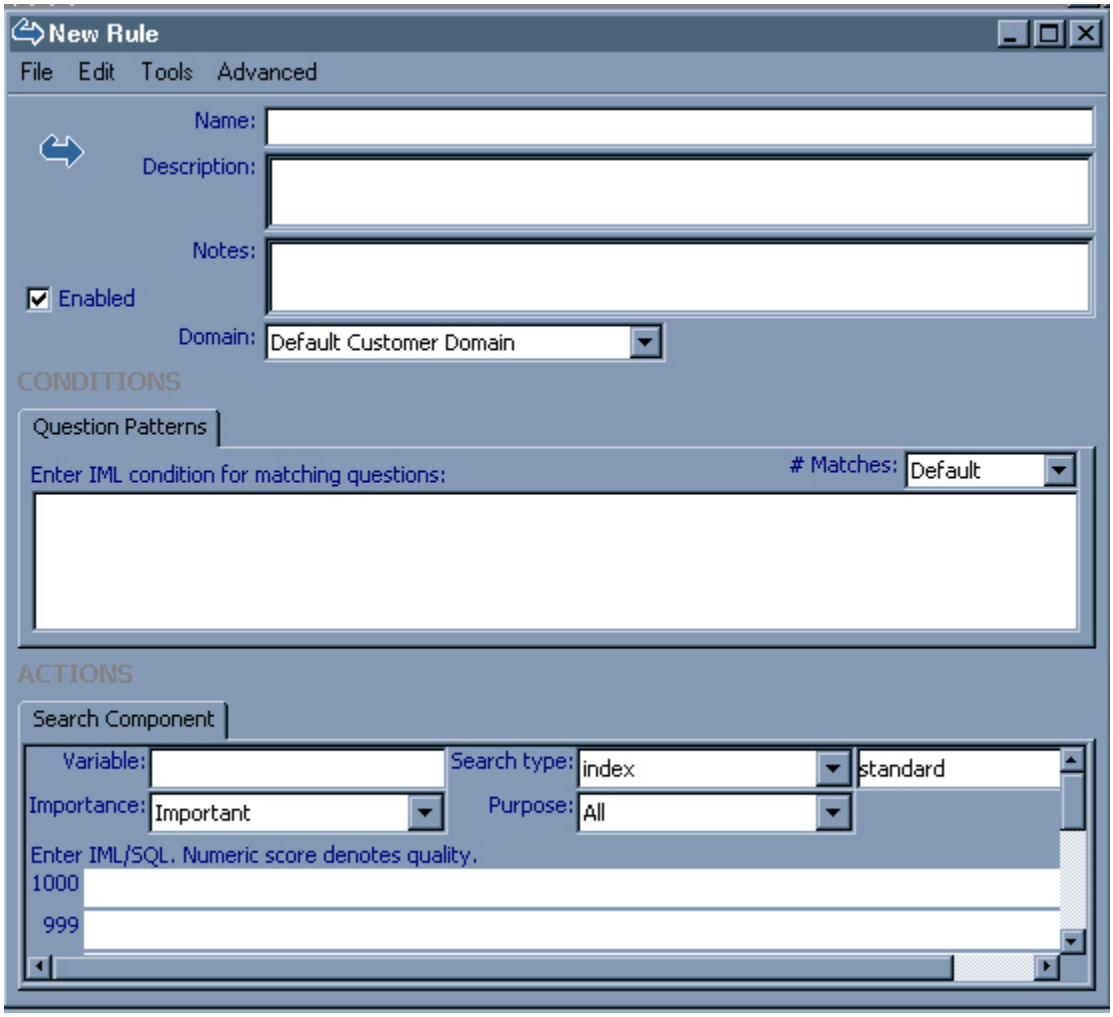


Note: The fourth result is from the Oracle Knowledge Analytics Installation Guide.

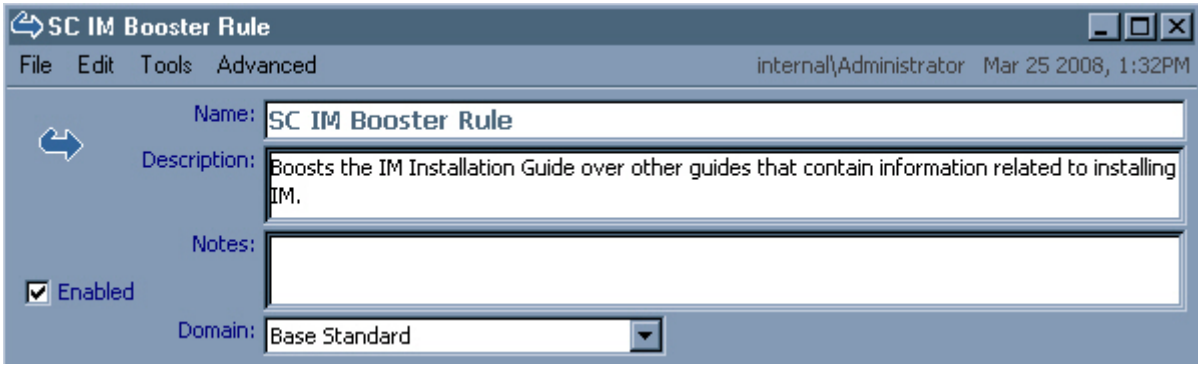
Knowing the Oracle Knowledge business context, we can safely assume that if a user query contains the words “install” and “im”, the user would benefit from the Information Manager documentation set rather than, for example, the Analytics documentation set. In order to prefer the Information Manager documentation set over the other sets we create a Search Component Rule.

To create the new Search Component Rule:

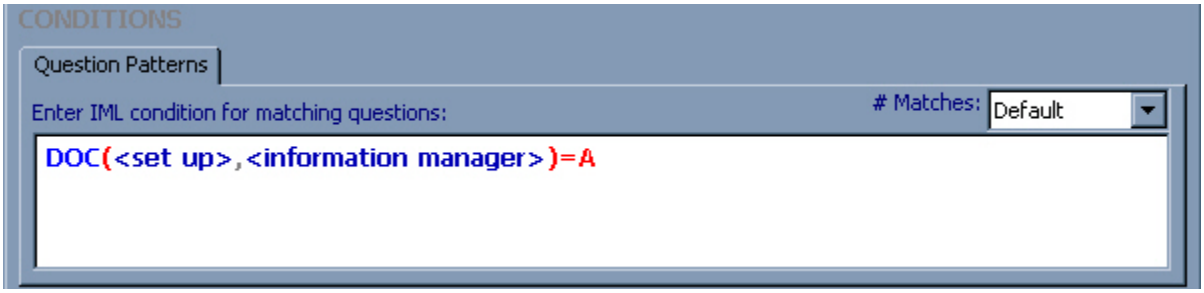
From Workbench, we select create a new Rule. As described above we delete the default action tab. We replace it with a Search Component action by selecting Add Search Component from the Advanced menu of the Rule window. The Rule window now looks like the following:



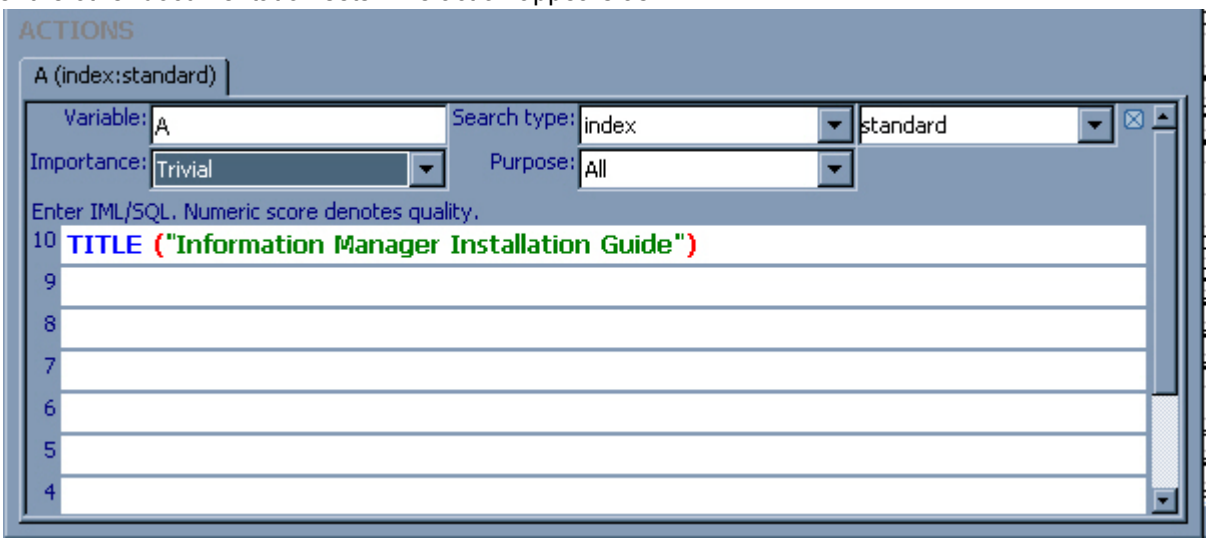
We name the Rule following conventional naming practices.



Using IML we create a question pattern that causes the Rule to fire anytime a user queries about “installing IM”.

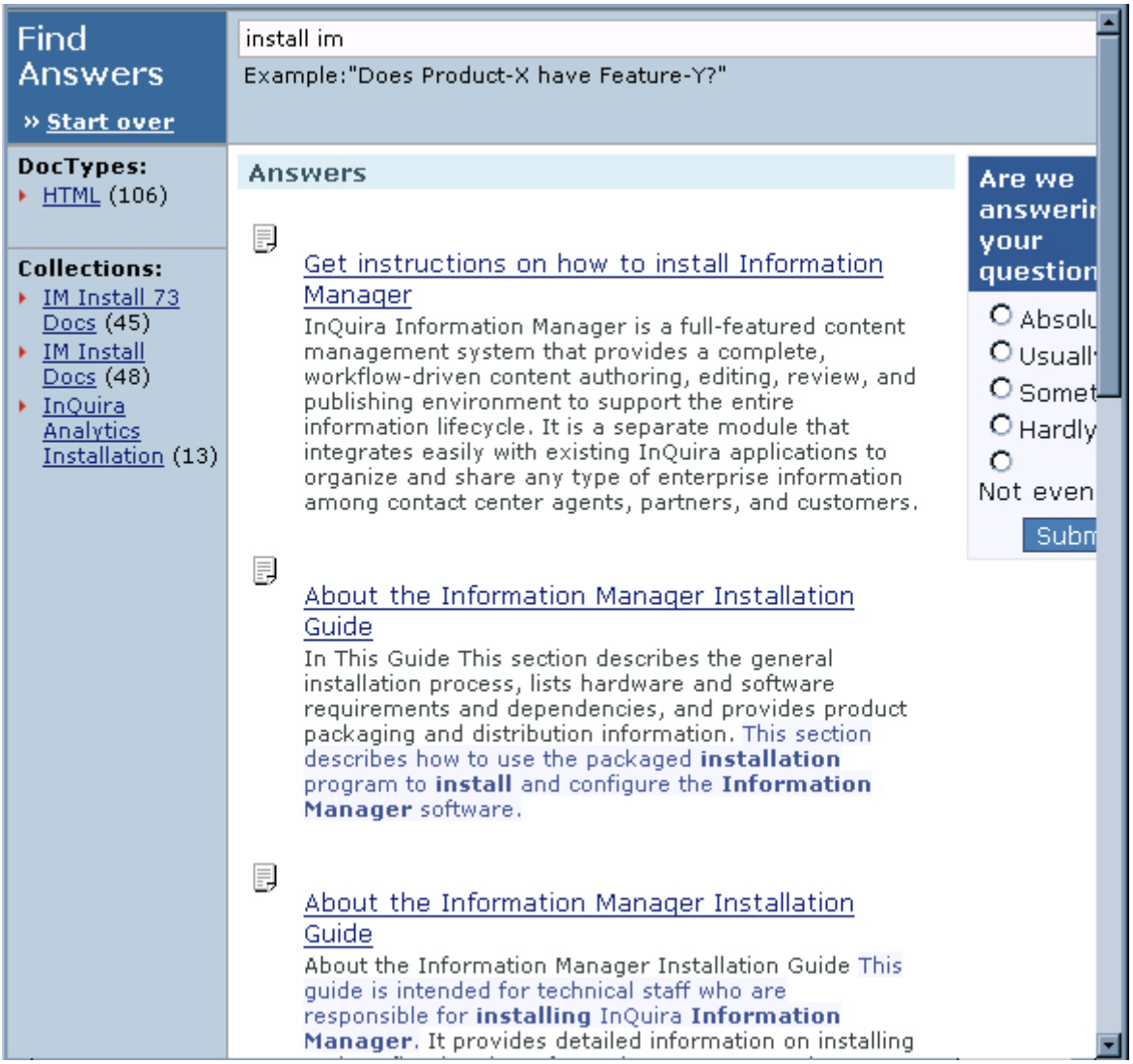


After we have established the Question Pattern we define the actions as explained above. First we set the Importance level to Trivial, keeping in mind that after checking the Rule in Test Drive we may need adjust this setting to obtain the desired results.We use IML in order to prefer the Information Manager Installation Guide over the other documentation sets. The action appears as:



Currently, this action states that when the question pattern above is satisfied, assign an additional 10 points to documents that contain the title Information Manager Installation Guide, which, effectively, is the title for the

complete Information Manager Installation documentation set. We then save our Search Component Rule and check the effect in Test Drive. The following results are returned.



As shown in the Test Drive results, the first response is the Managed Answer created earlier, followed by documents that include the title “Information Manager Installation Guide.” Since the Search Component Rule is effectively altering the search results, there is no need to adjust the Importance level from Trivial.

Domains and Domain Lists

Each Dictionary Object in the Dictionary is associated with a Domain.

There are two types of Domains:

- *Concept Domains* on page 83
- *Non-Concept Domains* on page 83

Each Domain is associated with *Domain Lists* on page 86.

Concept Domains

Concept and Cluster information within the Dictionary is stored in a special repository called the *Ontology*. The Ontology is organized into *Concept Domains* that reflect areas of conceptual knowledge relevant to a discrete area, such as:

- Base language, for example English
- Industry or subject matter area
- Specific company or application

The default Concept Domain is *Base Concepts*.

Non-Concept Domains

Non-Concept Domains are collections of Dictionary Objects other than Concepts and Clusters that address specific areas of knowledge required by Oracle Knowledge Intelligent Search. Domains range in scope from general to specific. The most general Domains contain extensive vocabulary for native languages, these are referred to as *Base Domains*. More specific Domains address various industries and are therefore referred to as *Industry Domains*. These two Domain types are installed with Oracle Knowledge Intelligent Search. The most specific Domains are *Customer Domains*, which address the unique characteristic of your company's business environment and business needs.

The default Domain is set in the Advanced Configuration File, for more information see *Oracle Knowledge Intelligent Search Administration Guide*.

Create a new Domain

To create *Domains:*

1. In Dictionary Manager, select **Edit Domains** from the **Tools** menu.
   The Edit Domains window displays:

2. Enter a descriptive name into the **Non-Concept Domains** field or the **Concept Domains** field of the **Edit Domains** window,

3 Click **Add**. The Save new dialog displays:

4 Click **Yes**. The Domain is now created and may be used immediately.

Your newly created Domain is automatically included in the English Customerhelp Domain List. To modify which Domain List a Domain belongs to, see *Domain Lists* on page 86.

Delete a Domain

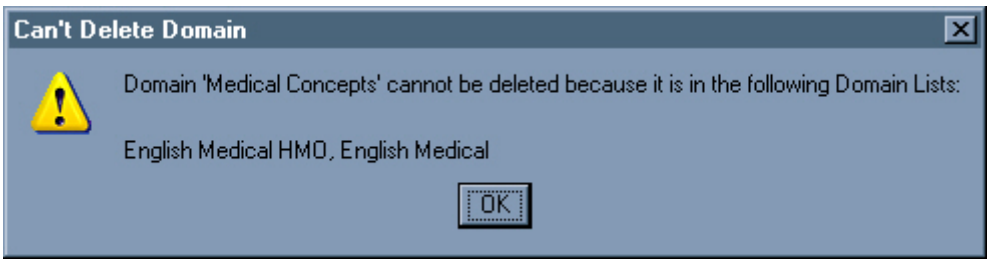
To delete a Domain:

1 In Dictionary Manager, select **Edit Domains** from the **Tools** menu.

The Edit Domains window displays.

2 Select the ☒ box next to the Domain you want to delete from the **Edit Domains** window.

Note: You may not delete a Domain that is included in a Domain List. The Can't Delete Domain error displays:



Domain Lists

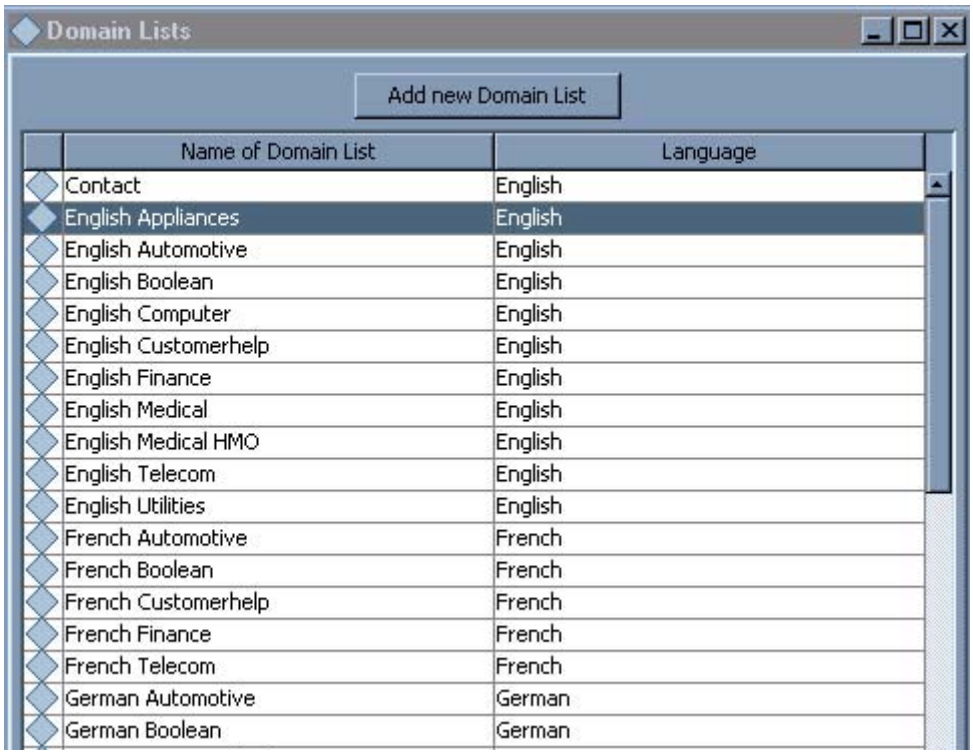
Domains are organized into collections called *Domain Lists*. Each query is submitted with an instruction that identifies the Domain List to which it applies. The default is the English Customerhelp. As a part of the Oracle Knowledge implementation process, Domain Lists have been established for you.

Create a new Domain List

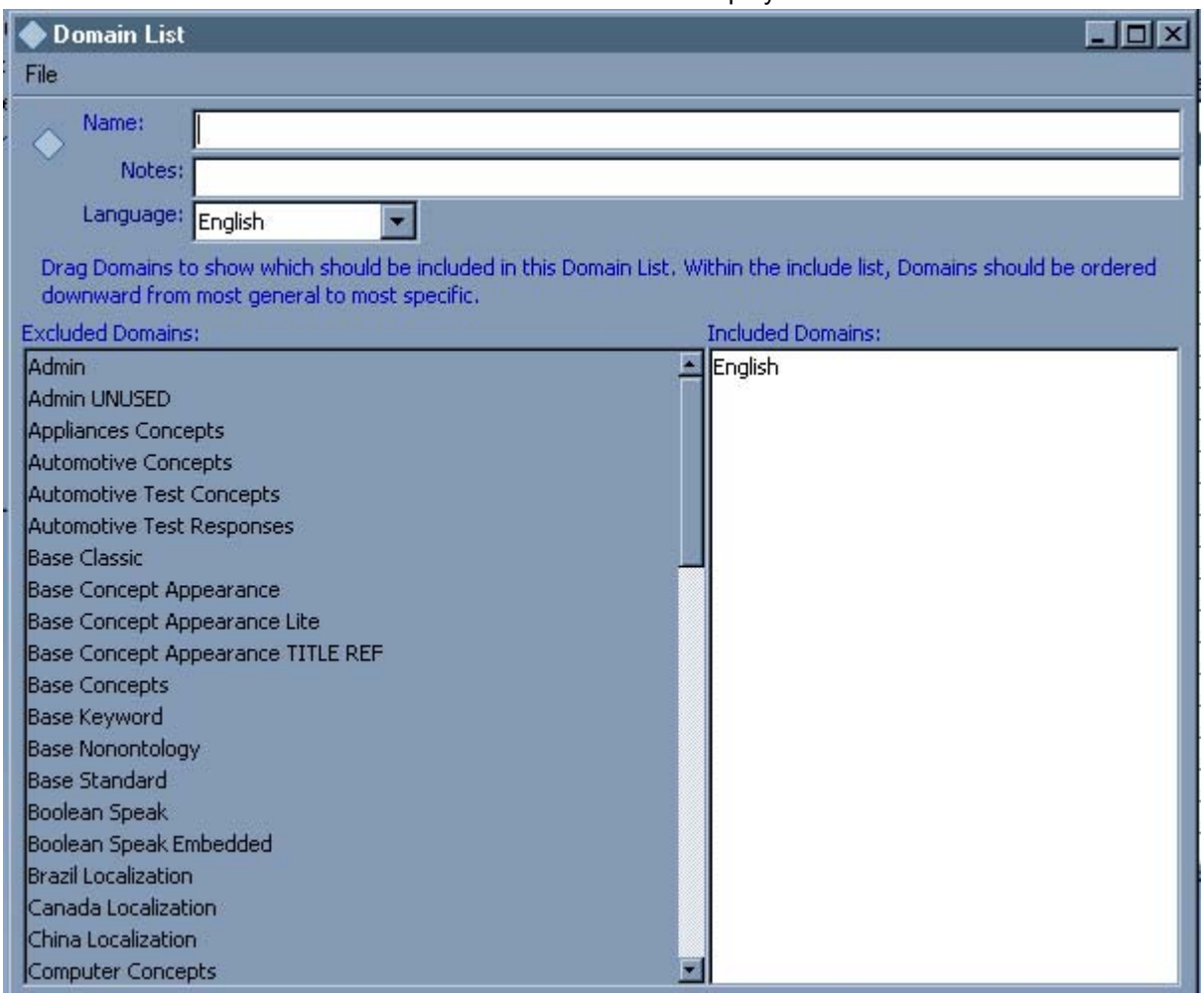
Note: Domain Lists have been created for you as a part your company's custom Oracle Knowledge configuration.

To create a new Domain List:

1 In Dictionary Manager, select **Edit Domain Lists** from the **Tools** menu. The following Domain Lists window displays:



2 Click **Add new Domain List**. The Domain List window displays.



3 Enter a name for your new Domain List in the **Name** field.

4 (Optional.) Enter a description of the Domain List in the **Notes** field.

5 Select a **Language** from the drop down list.

6 Select Dictionary Domains to include in the newly created Domain Lists.

Note: You must order the Domains from most general to most specific

Drag Dictionary Domain names from the **Excluded Domains** column on the left side to the **Included Domains** column on the right side.

After a Domain List has been created, you can modify which Domains it includes by selecting the Domain List name from the Domain Lists window and dragging and dropping the Domain names from the **Excluded** list to the **Included** list and vice-a-versa.

Note: We recommend that you do not modify the English Customerhelp Domain List.

Active Domain List

The *Active Domain List* is used during a context dependent task, for example, match against a question, test intent conditions, validate IML. The Active Domain List restricts which Domain List to use when performing these tasks.

The default Active Domain List is English Customerhelp.

To change the Active Domain List:

- Select the **Active Domain List** from the drop down menu located in the upper right hand corner of Dictionary Manager.

Spell Check List

Intelligent Search provides Spell-Checking capability to return accurate results despite spelling errors in the user queries. It includes a standard spell-correction engine that handles common words, and that can be supplemented with user-defined, custom-corrections. Additionally, the Spell-Checker can learn correct words from both the ontology and the searchable content, so as to best fit customer terminology. Intelligent Search automatically corrects most misspelled words in all supported languages.[1] If you find there is a common misspelling that is not corrected, then you may add it to the Spell Check List.

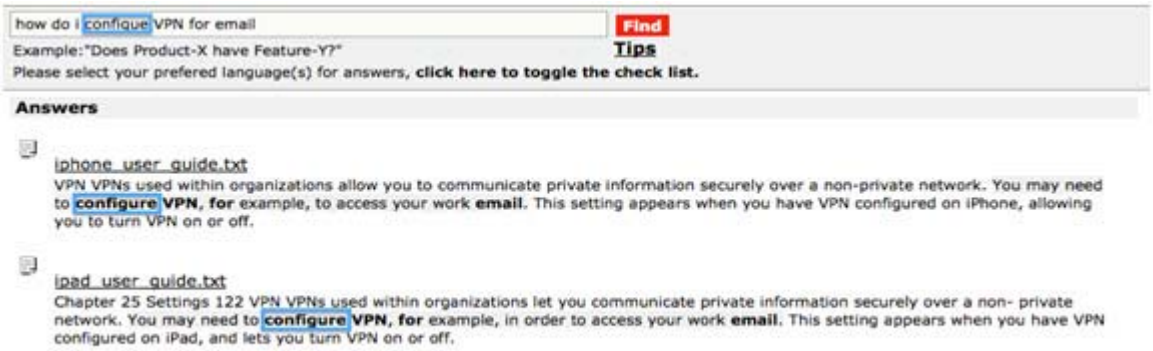
Operating Modes

Intelligent Search Spell-Checker operates in either of two modes: **automatic** and **interactive**.

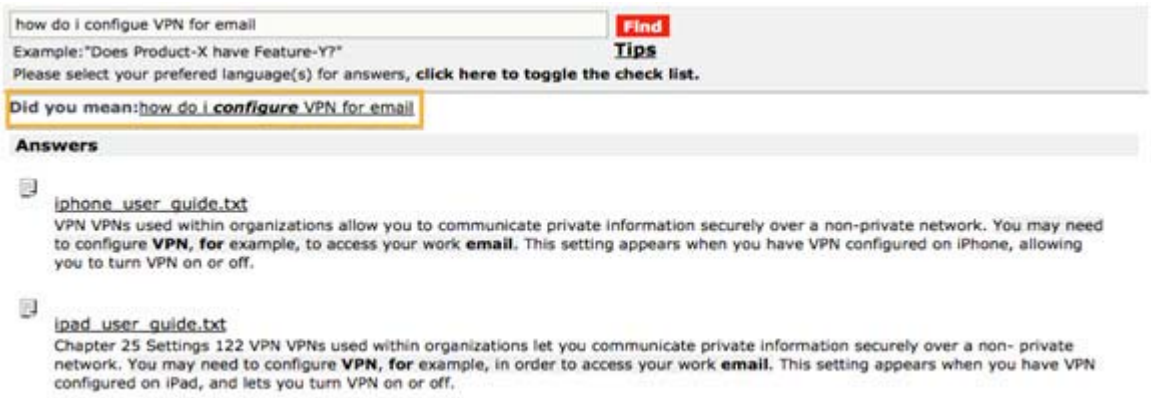
In automatic mode, Intelligent Search Spell-check automatically corrects misspellings that occur in the questions submitted by the users. For example:

Question inputted by the user: *How do I **configue** VPN for email?*

Actual query submitted to the search engine: *How do I **configure** VPN for email?*



In interactive mode, the original, uncorrected query is submitted first. If there are corrections, the user is presented with a link to a new, corrected query. The user can choose to submit the corrected query and obtain new search results.



1. For supported languages, see the *Oracle Knowledge Intelligent Search Administration Guide*.

In addition, you may find that proper names or foreign terms are incorrectly changed to other words. To avoid this you may add the word to the Spell Check List or, create the term as a synonym to a Concept as described in *Concepts* on page 19. Concept synonyms are considered correct spellings and are not spell corrected.

How Spell Checking Works

The Spell-Checker is lexicon-based. This means that to determine if a word in the user query is spelled correctly, it checks whether the word is included in a large list of words, or lexicon, that represent correct spelling usage. If the word is not found, it is considered a misspelling, and the Spell-Checker replaces that word with the most likely correctly spelled word in its lexicon. Hence, as illustrated in the previous example, 'configue' is corrected to 'configure'.

The Spell-Checker lexica cover the vocabulary of common English language. Additional lexica are available for a number of other languages. Intelligent Search uses two sets of criteria to recognize misspellings: a typographical similarity algorithm and/or a phonetic similarity algorithm.

The Spell-Checker provides additional functionalities that are tailored to the operation of Intelligent Search in the enterprise context. These include:

- *Learning from the Ontology*
- *Learning from the Content*
- *User-defined Corrections*

Learning from the Ontology

The Spell-Checker learns the synonyms in the ontology. Ontologies contain industry- and customer-specific vocabulary, such as product names, abbreviations, acronyms, and other customer jargon. Note that only synonyms from concepts that are loaded (as part of the dictionary Domain List) are learned.

CONTEXTUAL SPELL-CHECKING

Because ontology synonyms may include phrases, Oracle Knowledge's Ontology-enabled Spell-Checker can learn certain contextual spellings. For example, if a customer adds to the ontology a fiber optic product with synonyms such as *Internet Fibe*, *Fibe Internet*, these phrases remain uncorrected; however, unless the ontology also contains a synonym *fibe*, this will be corrected to *fibre* when it is not entered next to *internet*.

Learning from the Content

The Spell-Checker also takes into account the words in the content in order to determine if a word in the question should be corrected or not. For instance, while a standard spell-checker would incorrectly change the word *iConnect* in a user query to *connect*, the Oracle Knowledge Spell-check verifies if the word is a legitimate word in the content. If *iConnect* appears repeatedly in the content, it is considered a legitimate word, and the unwanted correction does not take place.

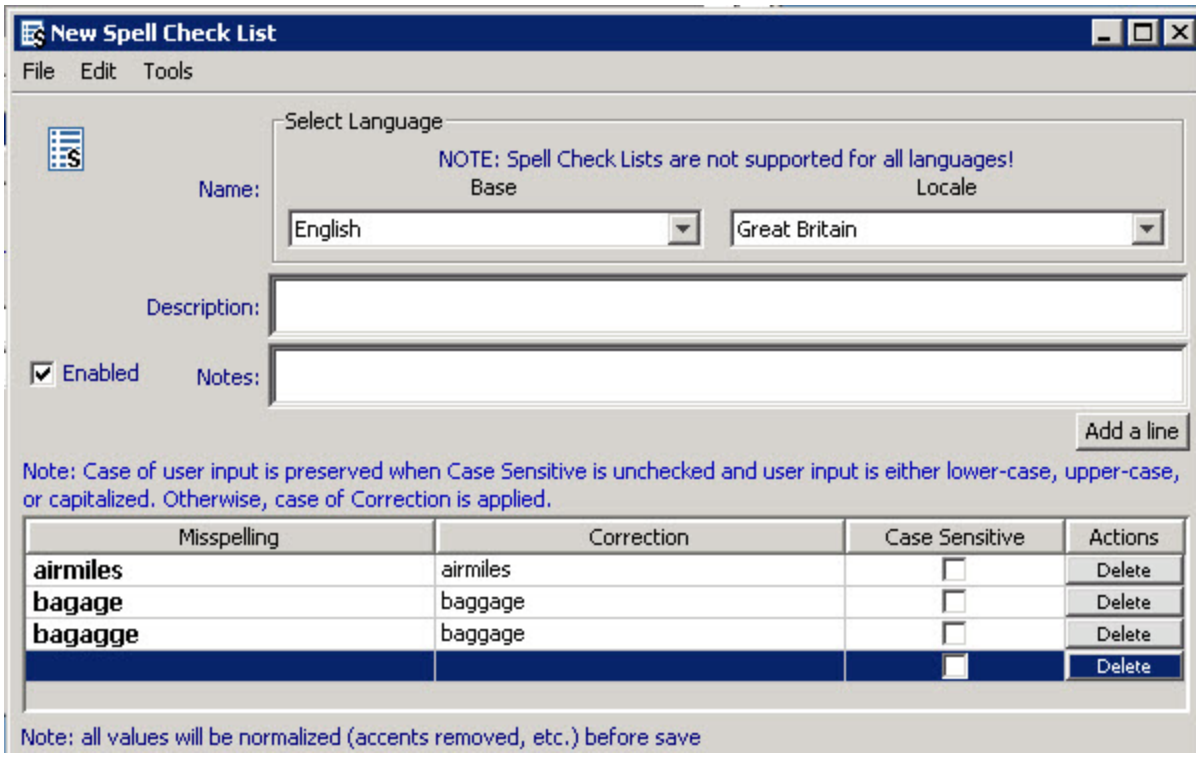
User-defined Corrections

Search administrators can define additional, custom corrections to govern the behavior of the Spell-Checker. To do so, create or modify Spell-Check Lists in the Dictionary Manager. User-defined corrections over-ride corrections from the standard spellchecking lexica.

Dictionary Spell-Check Lists allow you to define the specific form that the correction should take, as well as whether the correction applies case-sensitively or not. Intelligent Search spellchecking case-sensitively

allows you to distinguish between capitalized and lower case words, which may have a different meaning and therefore need to be spell corrected differently. For example, *RLOCATION* and *rlocation* get spell corrected differently to *LOCATION* and, *relocation*, respectively. To have a term spell corrected differently, modify the Spell Check List.

You can also prevent corrections for certain words by making the correction identical to the mis-spelling. The first line in the following Spell-Check List is representative of this usage:



Dictionary Spell-Check Lists can be created for both languages (e.g. English) and localized languages (e.g. British English, or Canadian English). In this case, the lists are merged; if the same misspelling occurs in both lists, the localized language list over-rides.

Additional Features

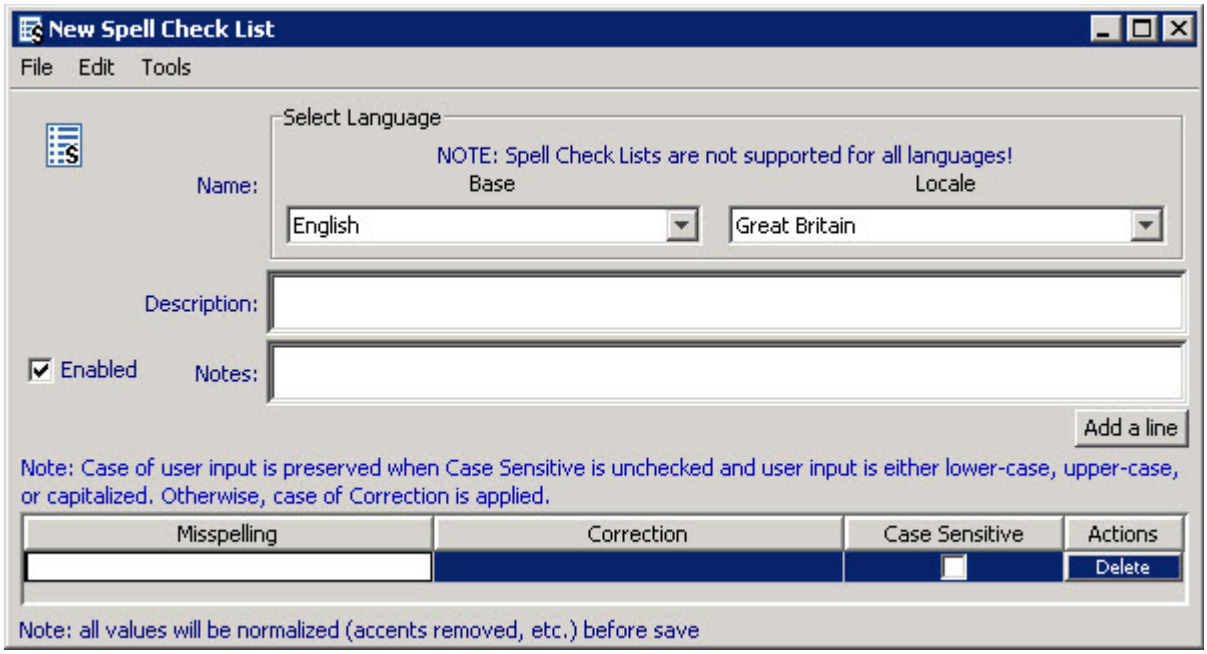
To prevent acronyms from being over-corrected, Intelligent Search Spell-Checker does not correct words that are entered in ALL-CAPS.

Creating a Spell Check List

Note: Spell Check Lists are language specific. They can be created at both a 'base' language level, such as English, French, Japanese, etc. or at a 'locale' level, such as English US, English GB, French Canada, French France, etc. If the language of content being indexed is specified as a 'base' language, then only the base language Spell Check List is used; if the language is specified as a 'locale' language, then the 'locale' Spell Check list, if it exists, is merged with the 'base' language Spell-Check List. Where any conflicts arise, the 'locale' values are used.

To create a Spell Check List:

1. In Dictionary Manager, select the icon or **New Spell Check List**, from the **File** menu.
   The New Spell Check List window displays:



2. Select a Language Base and Locale (if applicable), from the drop down lists in the **Name** field.

   Note: If the language or locale does not display in the drop down menu, then a Spell Check List for the language or locale has not been created, or the language or locale has not been enabled. See *Edit a Spell Check List* on page 25.

3. (Optional) Enter a definition into the **Description** field.

   The **Description** field has no impact on Oracle Knowledge Intelligent Search results. It is used to clarify the meaning among users, and can be used to find dictionary objects.

4. (Optional) Enter any information regarding the Spell Check List in the **Notes** field.

   The **Notes** field has no impact on Oracle Knowledge Intelligent Search results. It is used to track comments on the Spell Check List modification among users.

5. The **Enabled** box is selected by default and should remain selected. If the Enabled box is not selected, the Spell Check List is ignored by Dictionary Manager.

6. In the **Misspellings** column, enter the misspelling that you want to correct.

   Enter only one misspelling of a word per line.

7. Next to the misspelled word, enter the correct spelling in the **Correction** column.

8. Choose whether the user input case is preserved or changed to that of the correction.

   If the **Case Sensitive** box is not selected (default setting), Oracle Knowledge Intelligent Search preserves the user input case, whether lower-case, upper-case, or initial capitalization. If the **Case Sensitive** box is selected, the case of the Correction is applied.

9. To add more spelling corrections, select **Add a line** and repeat Steps 6-8.

10. Select **Save** from the File menu of the Spell Check List window.

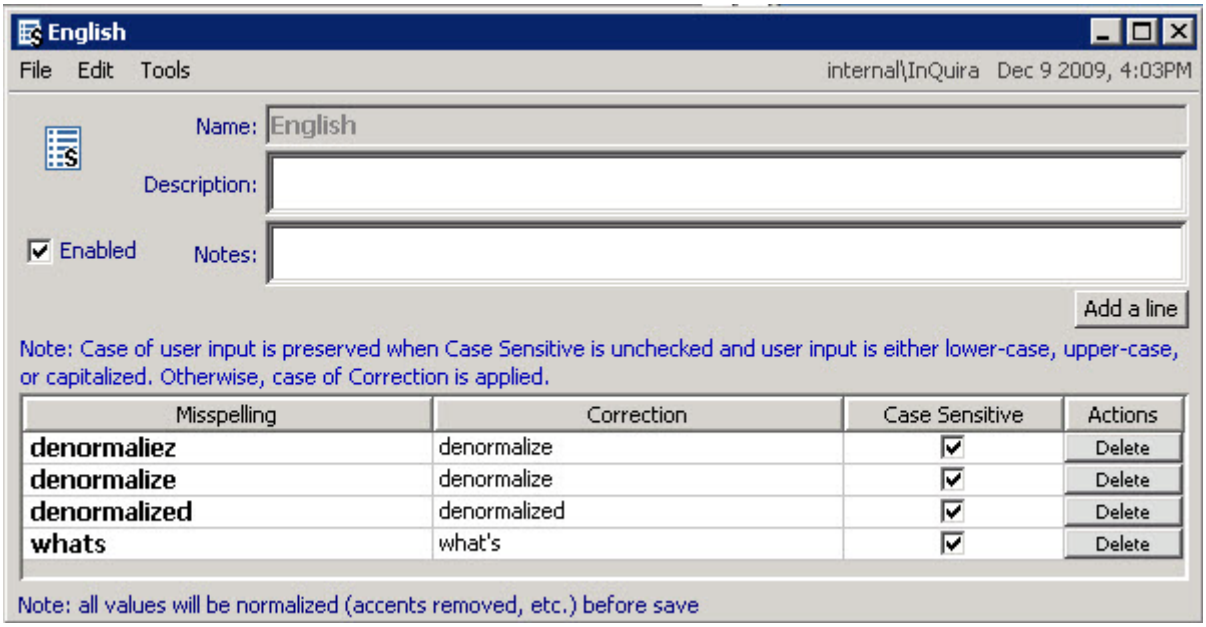
All words in a request are evaluated to see if they are known words, by first checking to see if they can be found in the concept synonyms for the specified request language. At this point, if a word is unknown, the system checks if a correction for it exists in the Spellcheck List Dictionary object for the language, and applies the correction if found.

If the **Use Indexed Content** option has been selected under the Rules Engine configuration in System Manager, then the index is searched for the remaining unknown words, with the idea that if the word is in the edited content, then it is a good word and should not be submitted for spelling correction. Words that cannot be found in the synonyms, content index, and user's Spellcheck List for the language are submitted to the Spellchecker for correction. For more information about the Rules Engine, see *Oracle Knowledge Intelligent Search Administration Guide,* Chapter 9: Advanced Configuration Facility, “System”.

Edit a Spell Check List

To edit a Spell Check List:

1. Using the Dictionary Find Tool, select the language specific Spell Check List.
   The Spell Check List displays:



   Note: If no Spell Check List exists for the language, you can create a new Spell Check List from the File menu, by clicking “New Spell Check List” and selecting the desired Language and possibly, Locale. Please see *Create a New Spell Check List* on page 26.

2. Click **Add a line**. A new line appends to the bottom of the list.
3. In the **Misspelling** column, enter the misspelling that you want to correct.
4. In the **Correction** column, enter the corrected form of the word.
5. Choose whether the user input case is preserved or changed to that of the correction. For additional information, see step 8 above.
6. Select **Save** from the File menu of the Spell Check List window.

Note: You may only correct a single word per line. You cannot enter a phrase in either the **Misspelling** nor the **Correction** columns.

Spell Check List Tools

To import an existing Spell Check List:

1. Select **Import from file** from the **Tools** menu in the Spell Check List window.
2. Select the Spell Check List as a *.csv file, and select **Import**. The Spell Check List imports.

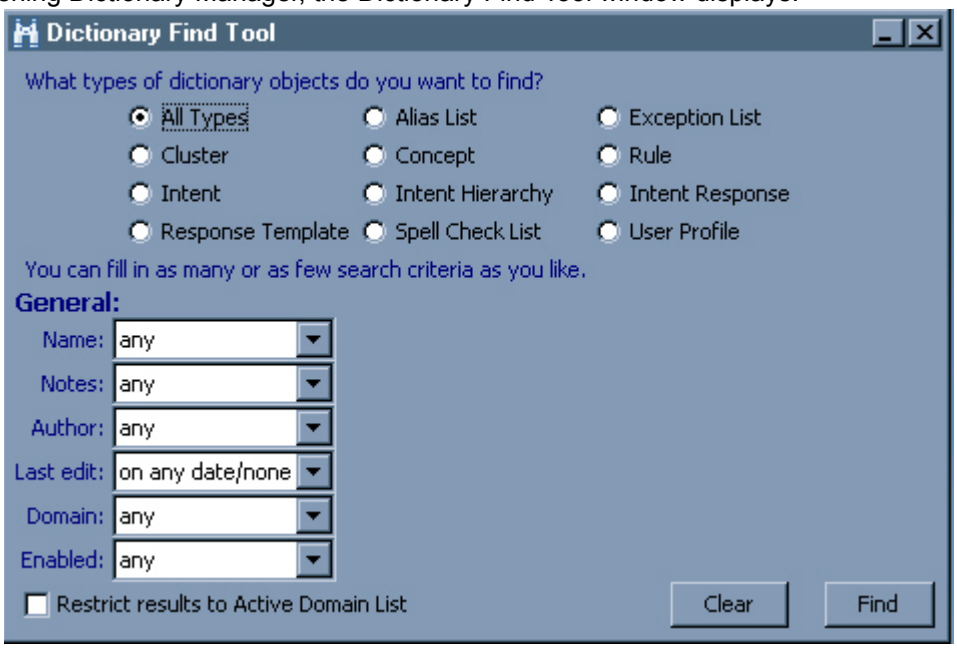
To export the Spell Check List:

1. Select the lines in the Spell Check List that you wish to export. You can use `Ctrl-A` to select all the lines in the Spell Check List.
2. Select **Export to file** from the **Tools** menu in the Spell Check List window.
3. Select the folder, and enter the name of the .csv file that you want the Spell Check List to be exported to.
4. Click **Export**. The selected Spell Check List entries will be exported to the file.

Finding Dictionary Objects

The Dictionary Find Tool allows you to search using set criteria in order to locate a Dictionary Object with ease.

Upon first opening Dictionary Manager, the Dictionary Find Tool window displays:



You can also access the Dictionary Find Tool by selecting Edit > Find Dictionary Objects or by selecting the icon.

In order to limit the amount of Dictionary Objects returned by the Dictionary Find Tool you can specify search criteria. You can use the *Search Criteria Operators* on page 144. You can specify general search criteria to search for all types of objects, or specify definition-specific criteria to search for a single type of object.

To specify search criteria for all types of objects, select the All Types radio button. The General search criteria fields display, as described in *Specifying General Criteria* on page 145.

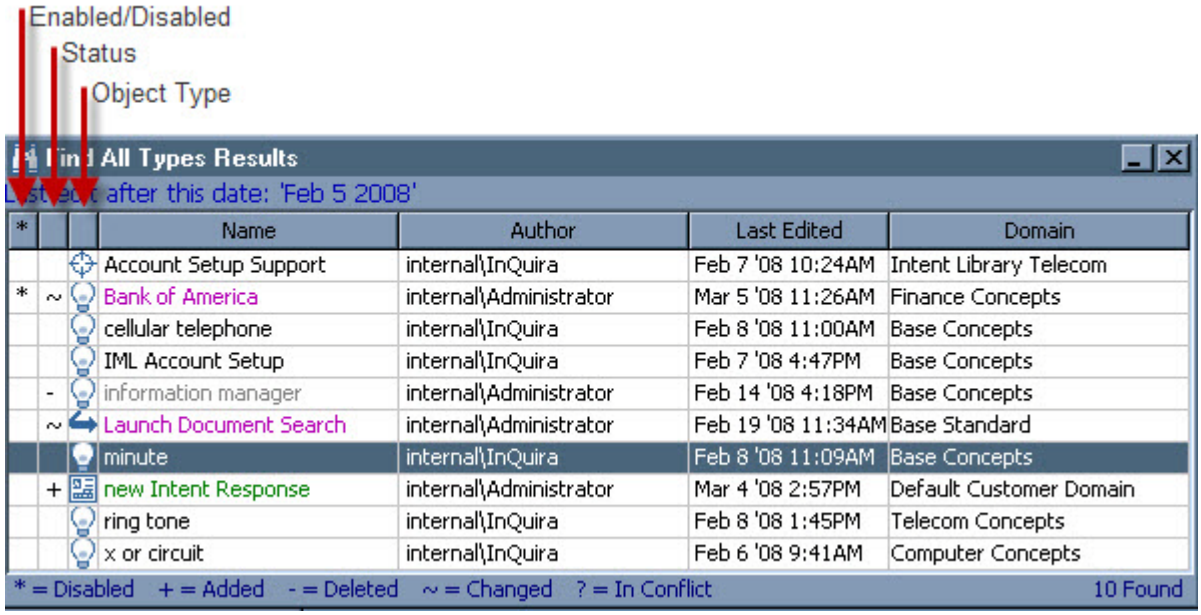
To specify search criteria for a specific type of object select the corresponding radio button. The General search criteria fields display as well as the search fields for the selected type display, as described in:

- *Specifying Concept-Specific Criteria* on page 146
- *Specifying Rule-Specific Criteria* on page 146
- *Specifying Intent or Intent Hierarchy-Specific Criteria* on page 147
- *Specifying Intent Response-Specific Criteria* on page 147

For more information on finding Dictionary Objects not discussed in relation to language tuning please see *Oracle Knowledge Intelligent Search Optimization Guide*.

Displaying Dictionary Search Results

The Find Objects Results window displays the objects that match the search criteria:



The search results are divided into columns displaying the following attributes:

Attribute	Description	
Enabled / Disabled (first column from left)	Displays * when an Object is currently disabled. Column is clear for enabled objects.	
Status (second column from left)	Displays the current status of the Object since the last commit, see *Commit Dictionary Changes to Central Repository (vcroot)* on page 97. The following symbols appear, along with a color code to represent the status of the Object:	
	+ with the name in green	Added
	– with the name in grey	Deleted
	~ with the name in magenta	Changed
	? with the name in red	In Conflict
Object Type (third column from left)	Displays the icon indicating the object type.	
Name	Displays the object name as specified in the section of the object.	
Author	Lists the ID of the user who last edited the object.	
Last Edited	Lists the date and time of the most recent edit.	
Domain / Area	Displays the domain to which the object belongs.	

Adjust Objects from the Dictionary Search Results

From the Find Objects Results window, you may access and modify Objects.

- Select one or multiple objects (using CTRL), and right click. An options menu displays:

Open	Opens the selected object(s) in a separate window.
Disable	Disables selected object(s).
Enable	Enables selected object(s).
Delete	Deletes selected object(s).
Diff	Highlights differences between the local version of the object(s) and the version currently saved in the Central Repository.
Revert	Changes the selected object(s) back to the version saved in the Central Repository.
Rename	Opens an **Input** window to enter a new name for the selected object.
Export	Opens an **Export Selected Objects** window from which you can export selected objects to a .csv file.
Log	Lists all of the log messages for the selected object(s).

Additional options display for Concepts:

Set/Unset Index IML	Sets or Unsets the Index IML option for the selected Concept(s). For more information see, *Oracle Knowledge Intelligent Search Optimization Guide*.
Set/Unset Openclass	Sets or Unsets the Openclass option. for the selected Concept(s). For more information see, *Oracle Knowledge Intelligent Search Optimization Guide*
Set/Unset Subsumes	Set or Unsets the Subsumes option for the selected Concept(s). For more information see, *Oracle Knowledge Intelligent Search Optimization Guide*
Graph	Opens the Ontology viewer for the selected Concept(s). For more information see *Ontology Viewer* on page 30.

Commit Dictionary Changes to Central Repository (vcroot)

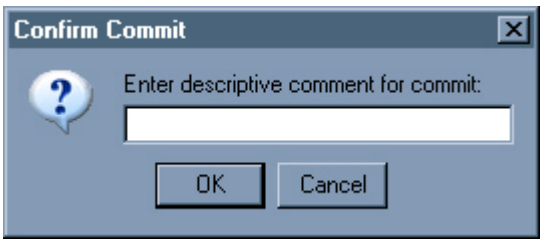
The Oracle Knowledge Dictionary uses a form of version control to allow multiple language users to be able to login to the Language Workbench and work simultaneously. Therefore, Oracle Knowledge creates a copy of the central version of the dictionary for each user, referred to as the *Local Dictionary*. Of these versions, the one that is used by Oracle Knowledge Intelligent Search is the *Central Repository*. As a result, a new Dictionary Object created by a user does not affect Intelligent Search results until it has been committed to the Central Repository, or **vcroot**.

The Central Repository, or vcroot, is stored on the development environment at `<InQuira>\base\development\content\data\vcroot`.

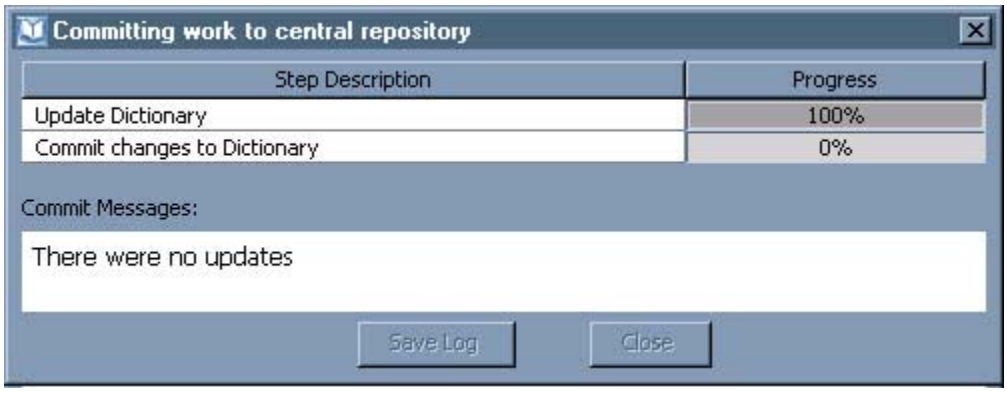
Only commit Dictionary Objects to the Central Repository after checking them fully using Test Drive. For more information see *Chapter 3, Test Drive*. Before committing you may choose to validate the dictionary in order to detect any possible errors. For more information see *Validate Dictionary* on page 103

To commit to the Central Repository:

1. In Dictionary Manager, select **Commit Work to Central Repository** from the **Tools** menu. The Confirm Commit Dialog displays.



2. Enter a brief description of changes or reason for committing.
3. Click **OK**.
   The Commit process begins.

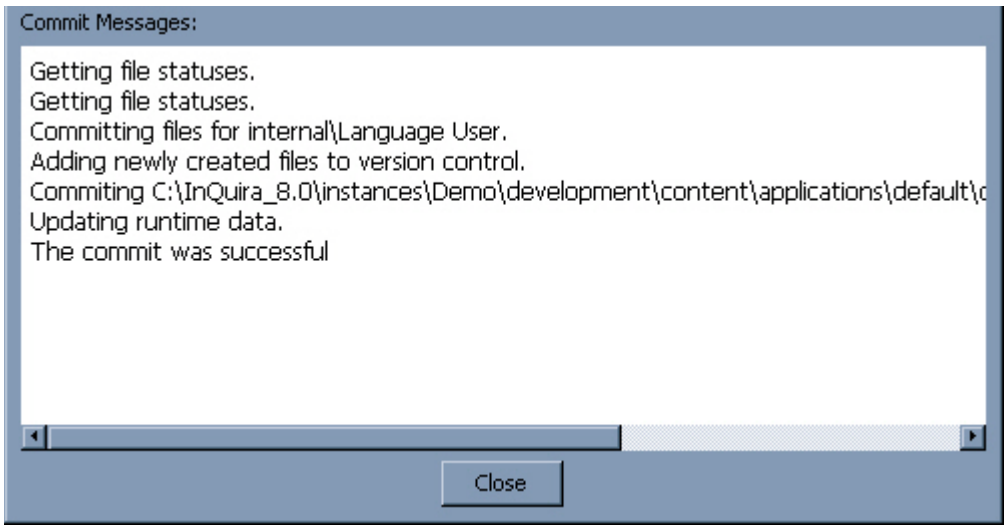


The Dictionary Manager does not perform any other functions during the commit process, since doing so could affect the data being transferred. During the commit process, Dictionary Manager automatically:

- Tests for conflicts between the local data and the Central Repository data
- Checks whether you need to update from the central repository prior to committing changes
- Updates each changed item in your local Dictionary

The Dictionary Manager does not commit changes if it detects a conflict. A conflict is created when two users have checked out the central dictionary, made changes that overlap the other, and one commits the change first. When the second user attempts to commit, the dictionary detects changes have been made and a conflict is created. If conflicts are detected see *Resolving Dictionary Conflicts, Commit Failed* on page 100.

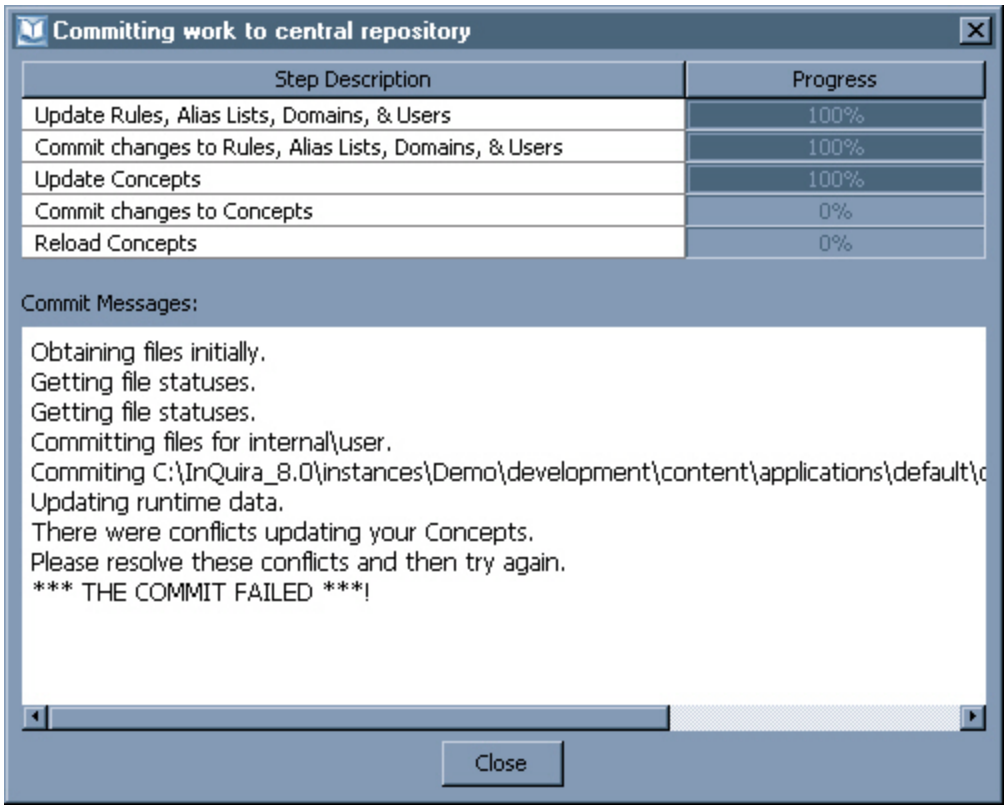
If Dictionary Manager does not detect any conflicts the following message displays:



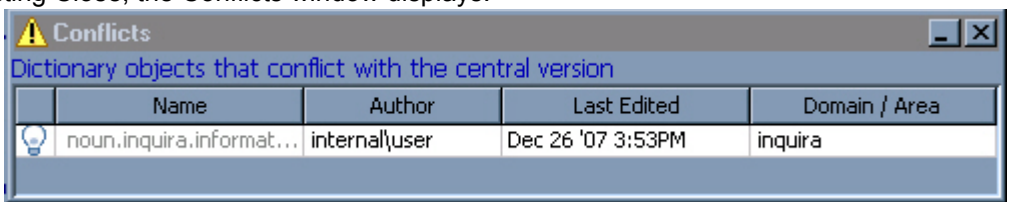
After a successful commit you must run a job in System Manager in order to see the effects on live Intelligent Search results. If you have made changes to Concepts you need to run both a re-index and synch job. If you made changes to Dictionary Objects other than Concepts you must run Mark Revisions for Synchronization and Synchronization jobs. For more information see *Oracle Knowledge Intelligent Search Administration Guide*.

Resolving Dictionary Conflicts, Commit Failed

If Dictionary Manager detects conflicts during the commit process the following message displays:

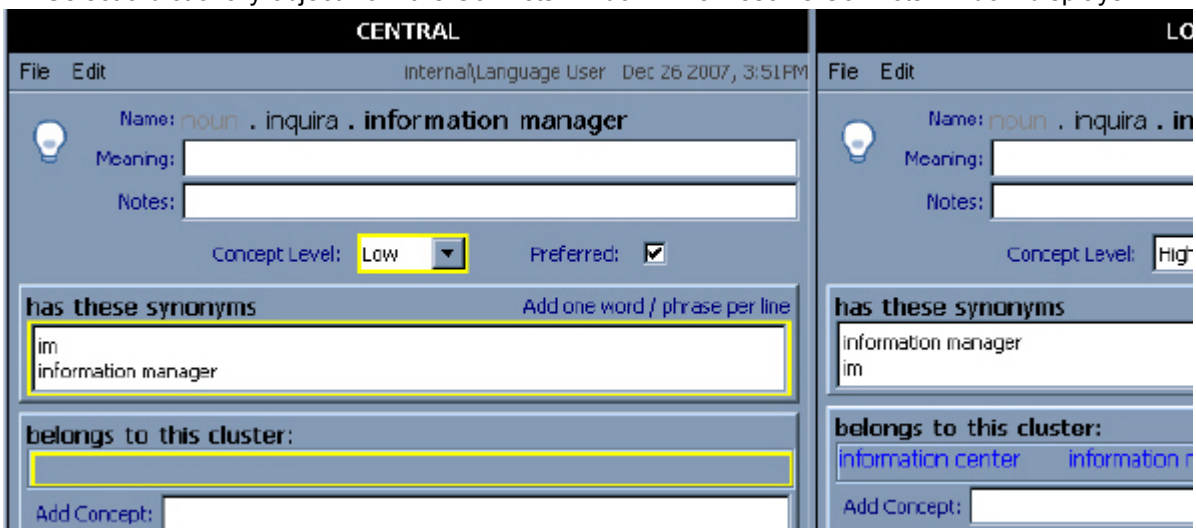


Upon selecting Close, the Conflicts window displays.



To resolve conflicts:

1. Select a dictionary object from the Conflicts window. The Resolve Conflicts window displays:



   The Resolve Conflicts Window compares your local version of the dictionary object with the version already committed in the Central Repository. The differences between the two are highlighted in yellow.

2. After reviewing the two select to save the local version or the central version. You may also combine the changes or make additional changes in the local version and then select Save local version.

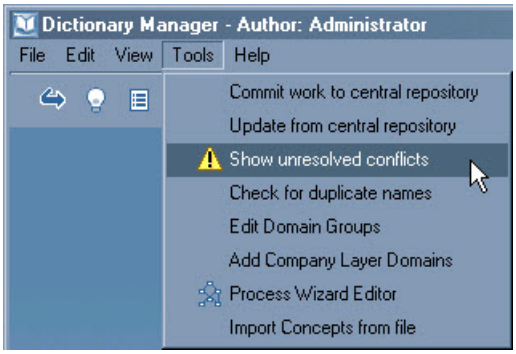
3. Repeat this process until all conflicts have been resolved.

4. After all conflicts have been resolved select **Commit Work to the Central Repository**. The commit process completes successfully.

Reviewing Dictionary Data Conflicts

Use the Conflict List to review conflicts prior to starting the commit process.

To display the Conflict List, select *Show unresolved conflicts* from the Tools menu in the Dictionary Manager main window:



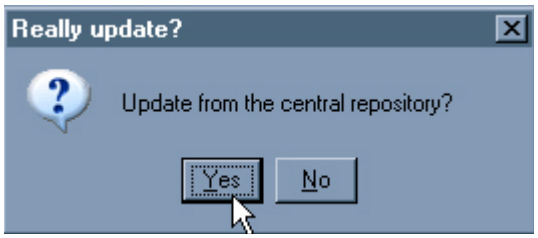
The Conflicts window displays the list of conflicts. To resolve Dictionary data conflicts, follow the steps outlined in *Resolving Dictionary Conflicts, Commit Failed* on page 100.

Update Dictionary from Central Repository

Upon opening the Oracle Knowledge Language Workbench your local version is automatically updated with data committed to the Central Repository. It may be necessary for you to update from the Central Repository during your Workbench session, especially if multiple users are using the Oracle Knowledge Language Workbench. Updating from the Central Repository during your session keeps your local Workbench version consistent with changes that have been committed by other Workbench users.

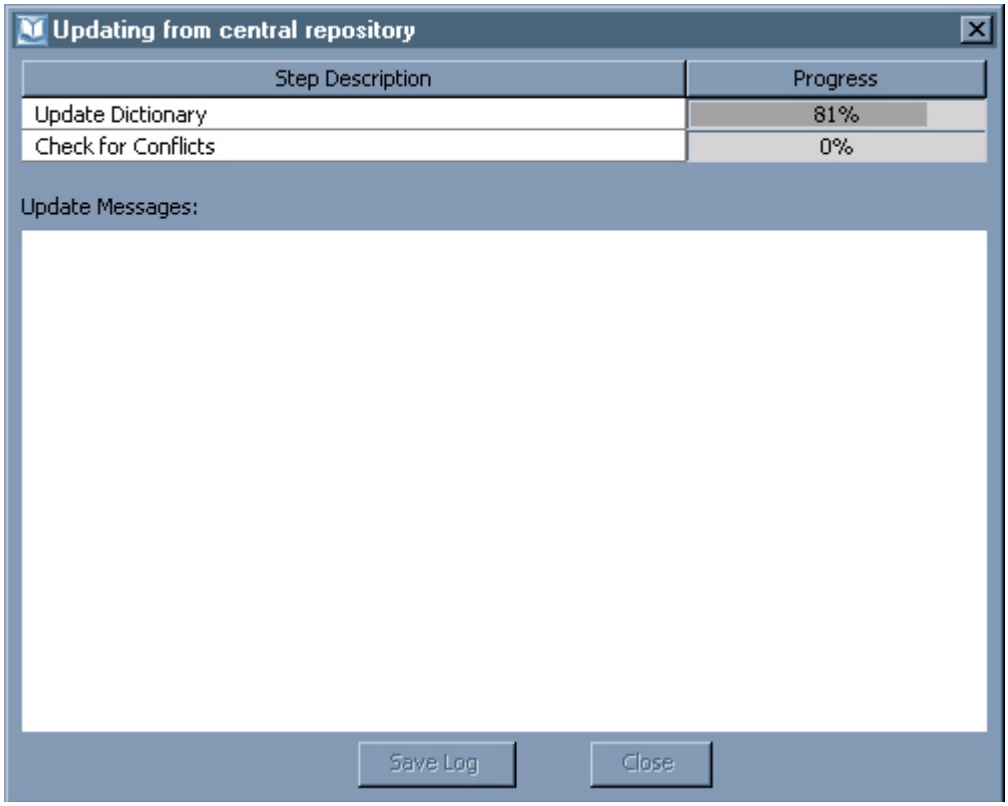
To update your local Oracle Knowledge Language Workbench with data from the Central Repository:

1. In Dictionary Manager, select **Update from central repository** from the **Tools** menu. The Really update dialog displays:



2. Select **Yes.**

   The update process begins:



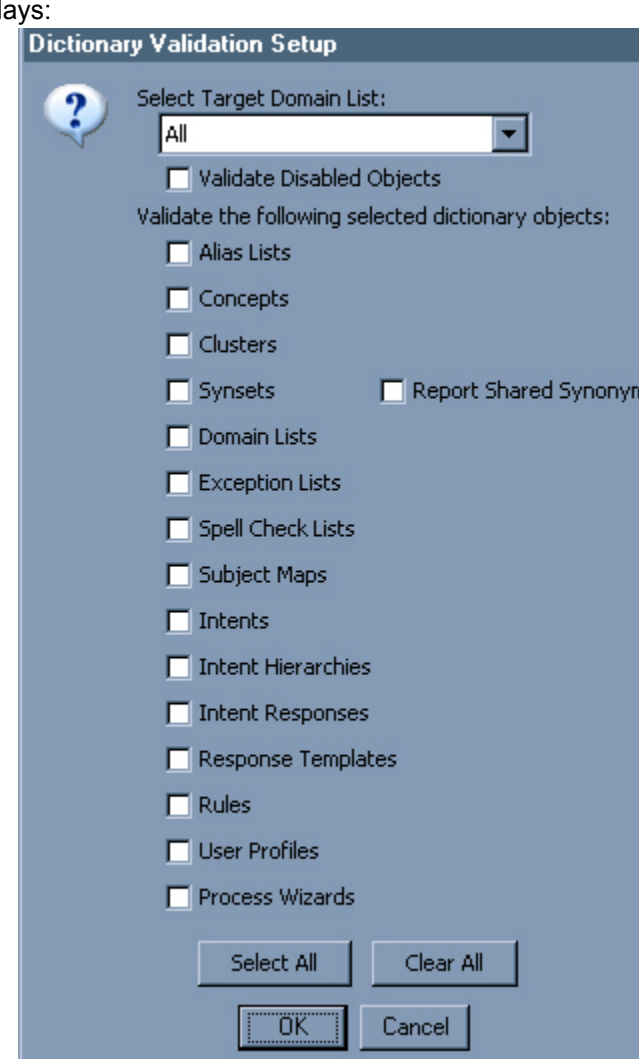
The Dictionary Manager does not perform any other functions during the update process, since doing so could affect the data being transferred. During the commit process, Dictionary Manager automatically tests for conflicts between your local version and the central version. You must resolve all Dictionary Conflicts, see *Resolving Dictionary Conflicts, Commit Failed* on page 100.

Validate Dictionary

The Dictionary Validation tool allows you to check for errors within dictionary objects before committing objects to the Central Repository. The tool finds broken references between objects, missing data, invalid values, and ontology relation loops.

To validate the dictionary:

1. From Dictionary Manager, select **Validate Dictionary** from the Tools menu. The Dictionary Validation Setup window displays:

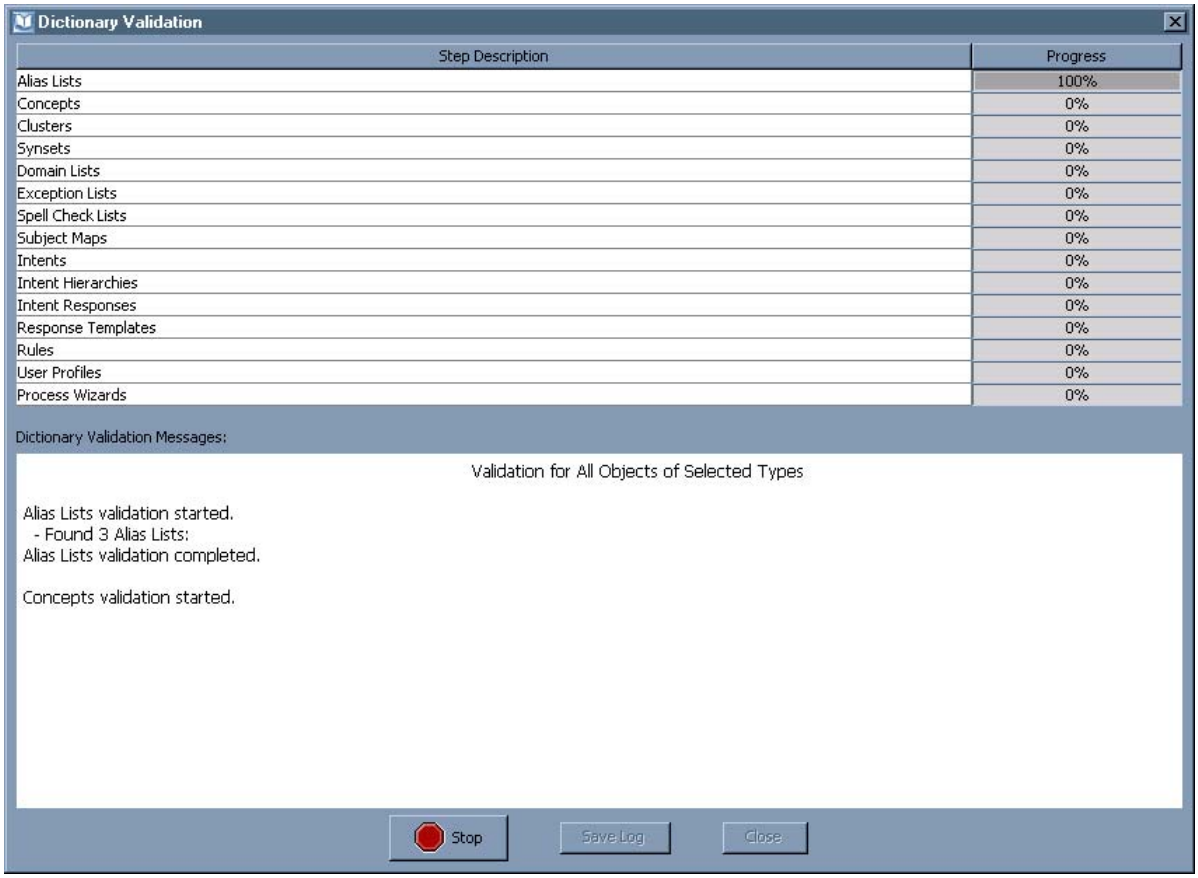


2. Select the Target Domain List you want to validate from the drop down menu.

   If you want to validate objects not included in the selected Domain List, select **Validate Disabled Objects**.

3. Select the dictionary objects you want to validate, or choose **Select All**.

4 Select OK. The Dictionary Validation window displays:



The Dictionary Validation Messages section lists the number of a particular object found and any errors associated with them. For more information regarding what validation tests Dictionary Manager runs, see *Dictionary Validation Tests* on page 148.

Reviewing Processed Workbench Modifications

The following table describes how to process modifications for viewing through Information Manager:

Modification	Steps to Process
Update concepts in the Dictionary Manager tool in Workbench	**1** Save. **2** Commit. **3** Full Reindex of all collections and subsequent content processing steps—Collection Maintenance, Classify (if facets are used), Global Maintenance. **4** Mark Data task. **5** Synch to Runtime (or restart the indexer instance).

Modification (Continued)	Steps to Process (Continued)
Update rules, intents, intent responses, or alias lists in the Dictionary Manager tool in Workbench	1 Save. 2 Commit. 3 Mark Data task. 4 Synch to Runtime (or restart the indexer instance). 5 No reindex is necessary.
Update facet/taxonomy information in the Personalized Navigation tool in Workbench	1 Save (no commit necessary). 2 Run the Classify task (preferably as part of a content processing task set that includes collection maintenance and global maintenance). 3 Mark Data task. 4 Synch to Runtime (or restart the indexer instance).
Update process wizards	1 Save (no commit necessary). 2 Mark Data task. 3 Synch to Runtime (or restart the indexer instance).
Update crawled content (such as on the web or in IM)	Rerun content processing including Status Override, Content Update, Prep/Document Conversion, Index, Collection Maintenance, Classify (if facets are used), Global Maintenance), Mark Data task, Synch to Runtime (or restart the indexer instance). These steps initiate an incremental crawl which generally picks up the change and is faster than a full crawl. If this doesn't pick up the change, try repeating the steps and include a Content Reset task (which clears all the previously crawled data from the database and thus all data is crawled again).
Change a channel's schema in IM	Rebuild the channel's IQXML and do a full recrawl of the collection including Status Override, Content Reset, Content Update, Prep/Document Conversion, Index, Collection Maintenance, Classify (if facets are used), Global Maintenance, Mark Data task, Synch to Runtime (or restart the indexer instance). If you have JMS disabled, you must restart IM to pick up the changes to the channel schema.
Propagation	Run the same steps in production that were recommended in development, as outlined above.

Note: Oracle recommends asking questions to a runtime instance. Oracle **does not** recommend testing changes by asking questions to the indexer instance other than thru Test Drive. If you ask questions against an indexer instance, you should later restart the indexer to clear out classes from memory that are used for question processing but not needed for content processing, to avoid out of memory errors.

Export and Import Dictionary Objects, Ontology Objects, Process Wizards, and Taxonomies

The following sections explain how to import Dictionary Objects, Ontology Objects, and Process Wizards from an existing file into a current version of Oracle Knowledge Intelligent Search.

Important! *Do not* use the following procedures to perform upgrades.
If you are moving from an 8.0.x release to 8.1.2 or higher, you *must* follow the migration procedure for ontology objects, as described in “Migrate Ontology Objects” and “Migrate Non-Ontology Objects” in *Upgrading Oracle Knowledge* version 8.1.2. The procedures discussed in the following sections apply only to release 8.1.2 and later.

Exporting Dictionary Objects

To export Dictionary Objects:

1. Select dictionary objects to be exported.
2. Right-click and choose "Export".

Exported dictionary objects are stored in subdirectories as follows:

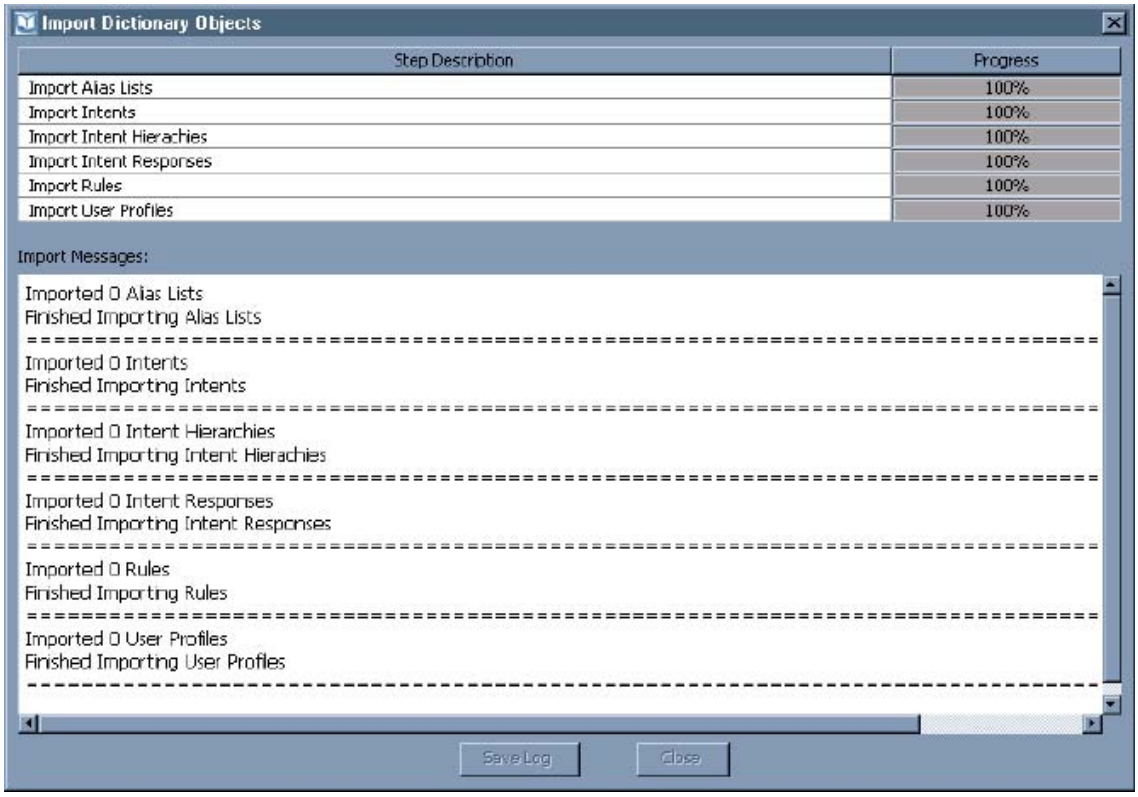
- ALIAS_LIST: Alias Lists
- SPELLCHECK_LIST: Spellcheck Lists
- EXCEPTION_LIST: Stemmer Exception Lists
- ANSWER: Intent Responses
- HIERARCHY: Intent Hierarchies
- INTENT: Intents
- PROFILE: User Profiles
- RULE: Rules
- DOMAIN_LIST: Domain Lists
- SUBJECT_MAP: Subject Maps

Importing Dictionary Objects

To import existing Dictionary Objects:

1. From Dictionary Manager, select **Import Dictionary Objects** from the Tools menu.
2. In the file finder, select the parent directory of the directories containing the Dictionary Objects and select **Import**.

The **Import Dictionary Objects** screen displays:



Importing Ontology Objects

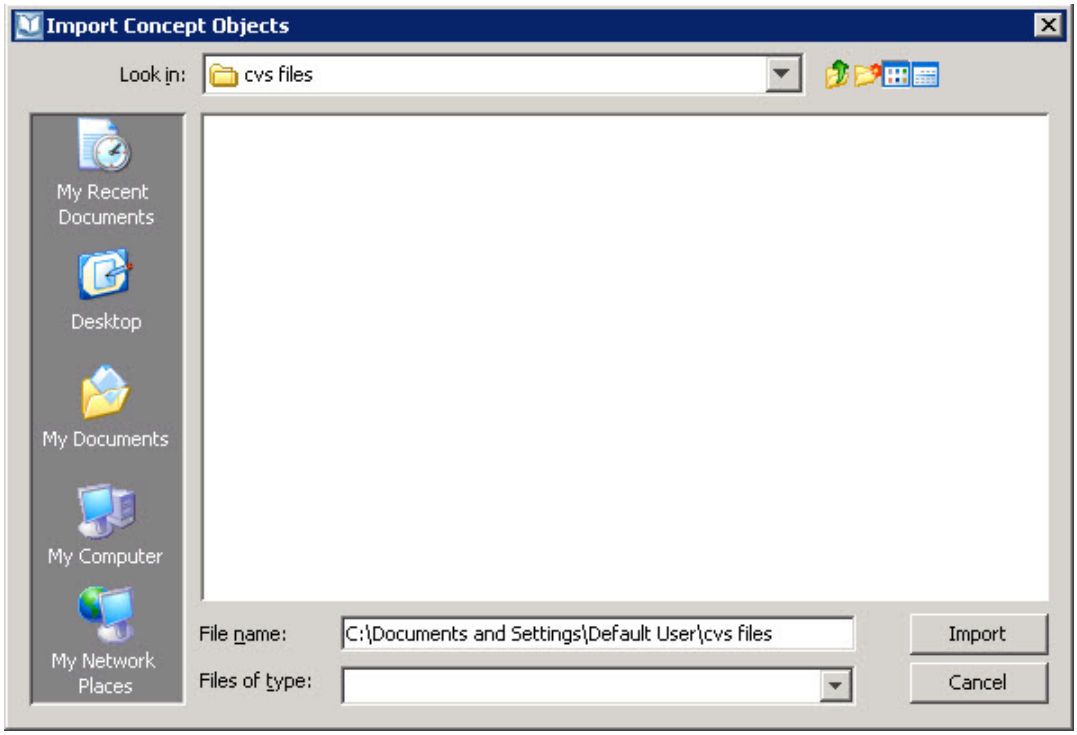
Dictionary Manager provides commands for importing some ontology objects from .csv files. You import all ontology objects together in one process. Before starting the import process, place all of the .csv files together in one folder. The following list shows the ontology objects and the associated .csv files:

- Concepts `concepts.csv`
- Clusters `clusters.csv`
- Synonyms `synsets.csv`

Important! The files must use the name conventions as shown above. Dictionary Manager does not recognize files and will not import files with other names.

To import ontology objects:

1 From the Language Workbench menu, select Tools, Import Ontology Objects.
The Import Concept Objects window displays.

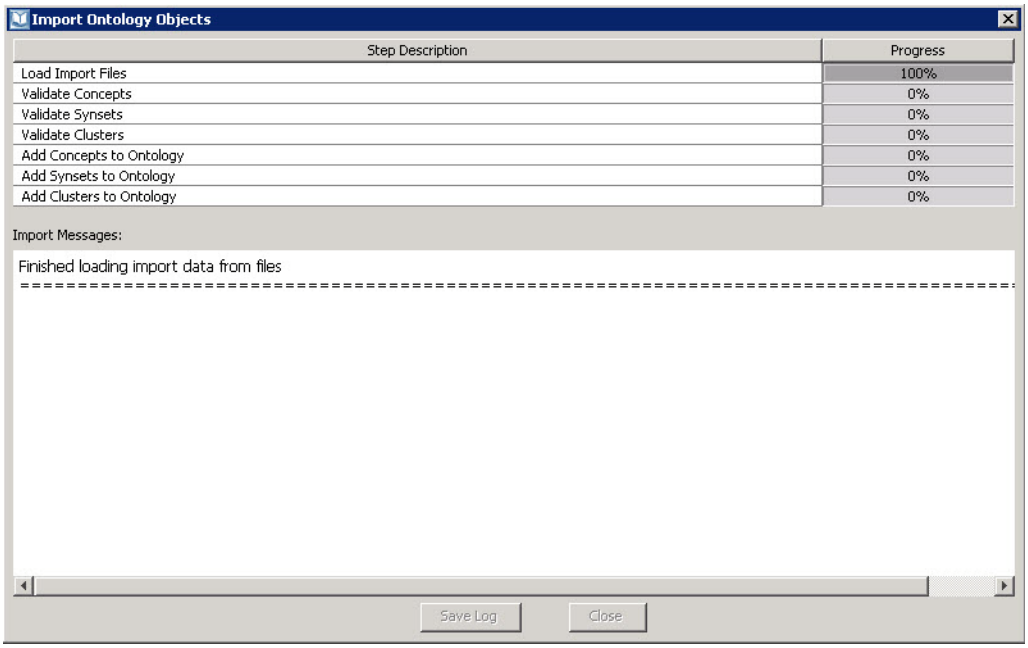


2 Navigate to the folder containing the .csv files for import.

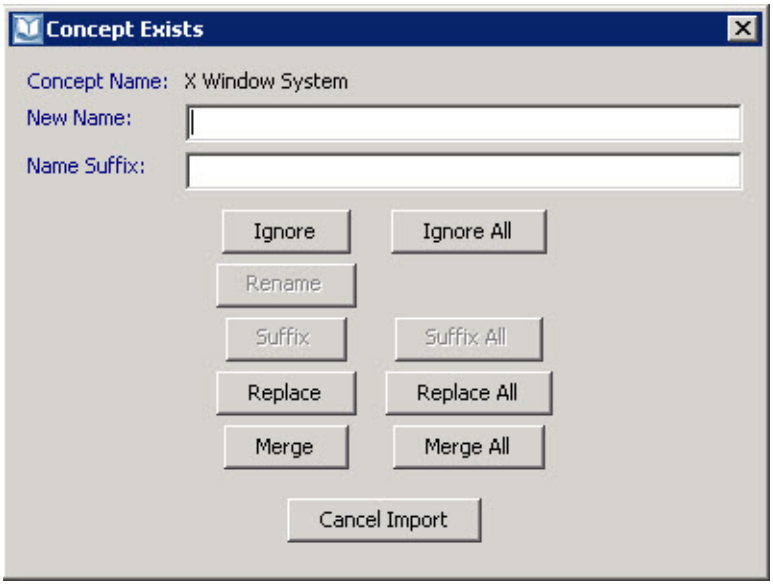
3 Click **Import**.

Note: The files may not appear in the Import Concept Objects window but, Dictionary Manager locates the files when you click **Import**.

The Import Ontology Objects window displays.



Dictionary Manager then loads and validates the files. The Import Ontology Objects window shows the progress of each task as it processes the task. If any exceptions occur during the validation, Dictionary Manager displays the following dialog box:



Exception Options:

Option	Description
New Name	Enter a new name for the object and click **Rename**.
New Suffix	Enter a new suffix for the object.
Ignore / Ignore All	Click **Ignore** to ignore this object. Click **Ignore All** to have Dictionary Manager ignore similar exceptions for objects being imported.

Option	Description (Continued)
Rename	Click to apply the name entered above in **New Name**.
Suffix / Suffix All	Click **Suffix** to apply the suffix entered in **New Suffix** above. Click **Suffix All** to apply the same suffix to all similar objects being imported.
Replace / Replace All	Click **Replace** to replace the existing object with the one being imported. Click **Replace All** to replace all similar exceptions with the objects being imported. the same suffix to all similar objects being imported.
Merge / Merge All	Click **Merge** to merge the existing object data with that of the object being imported. Click **Merge All** to merge the existing data for all similar objects with that of the associated object being imported.
Cancel Import	Click to cancel the import process.

After all objects complete validation, Dictionary Manager adds them to the application, completing the import process.

4 When the import process completes, click either **Save Log** or **Close**.

If you choose **Save Log**, Dictionary Manager prompts you for a file name and location for the log file.

Exporting Ontology Objects

Dictionary Manager provides commands for exporting some ontology objects to .csv files. The following list shows the ontology objects and the associated .csv files:

- Concepts `concepts.csv`
- Clusters `clusters.csv`
- Synonyms `synsets.csv`

To export concepts:

1 Use the Dictionary Find Tool to locate the concepts to export.

2 Select all concepts from the Find Concept Results window that are to be exported.

Select all concepts: Right click on any concept and choose Select All from the contextual menu.

Select multiple concepts: Select a concept and then hold down the control key (Ctrl) and click each additional concept to export.

3 Right-click on a selected concept and choose **Export** from the contextual menu.

Dictionary Manager prompts for the file location and name.

4 Enter file location, accept the default name.

5 Click **Export**.

Note: When you export concepts, Dictionary Manager automatically exports the synonyms (`synsets.csv`) file associated with the concepts that are exported.

Clusters

To export Clusters:

1. Use the Dictionary Find Tool to locate the clusters to export.
2. Select all clusters from the Find Clusters Results window that are to be exported.

   Select all clusters: Right click on an item and choose Select All from the contextual menu.

   Select multiple clusters: Select a concept. Hold down the control key (Ctrl) and click each additional concept to select.
3. Right-click on a selected cluster and choose Export from the contextual menu.

   Dictionary Manager prompts for file location and name.
4. Enter file location, accept the default name.
5. Click **Export.**

Synonyms

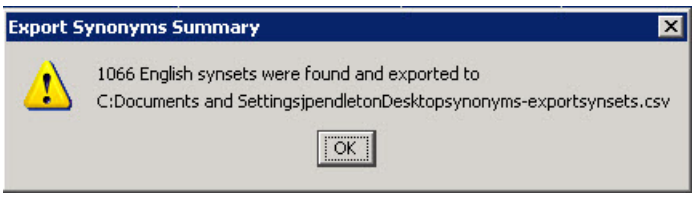
Note: You can export synonyms with the associated concepts using the Concepts procedure above. The following procedure exports only the `synsets.csv` file.

To export synonyms:

1. Use the Dictionary Find Tool to locate concepts containing the synonyms to export.
2. Select all items from the Find Concept Results window that are to be exported.
3. Right-click on a selected concept and choose Export Synonyms from the contextual menu.

   Dictionary Manager prompts you to select the language.
4. Enter the file location, accept the default name.
5. Click **Export.**

   Dictionary manager displays a message saying how many synonyms were found and exported.



6. Click **OK**.

Exporting Process Wizards

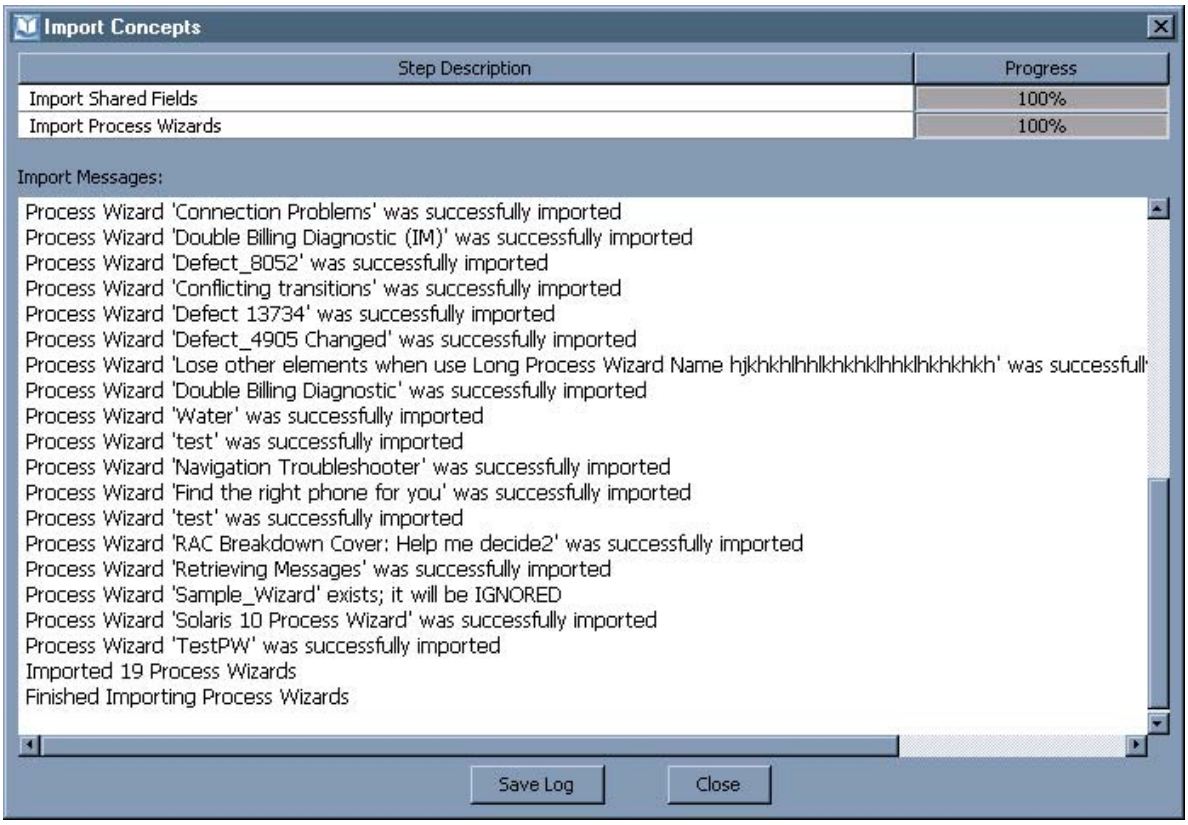
To export Process Wizards:

1. Open Process Wizard Editor.
2. Under Tools, select **Export Wizard** or **Export All Wizards**.
3. Save the files to a directory.

Importing Process Wizards

To import existing Process Wizards to be used by your current version of Oracle Knowledge Intelligent Search:

1. From Dictionary Manager, select **Import Process Wizards** from the Tools menu. The Import Process Wizards file finder displays.
2. Select the directory containing the .csv file containing Process Wizards and select **Import**. The **Import Process Wizards** screen displays:



Exporting Taxonomies

In Personalized Navigation, select "Tools", select "Export Taxonomies", and save the file.

Importing Taxonomies

In Dictionary Manager or Personalized Navigation, select "Tools", select "Import Taxonomies", browse to the taxonomy file to be imported, and import.

CHAPTER 3

Test Drive

Test Drive is a component accessed from within the Language Workbench that allows you to test language processing and the effects of changes made in Dictionary Manager prior to committing to the Central Repository and synchronizing. To access Test Drive see *Test Drive* on page 8.

This chapter explains:

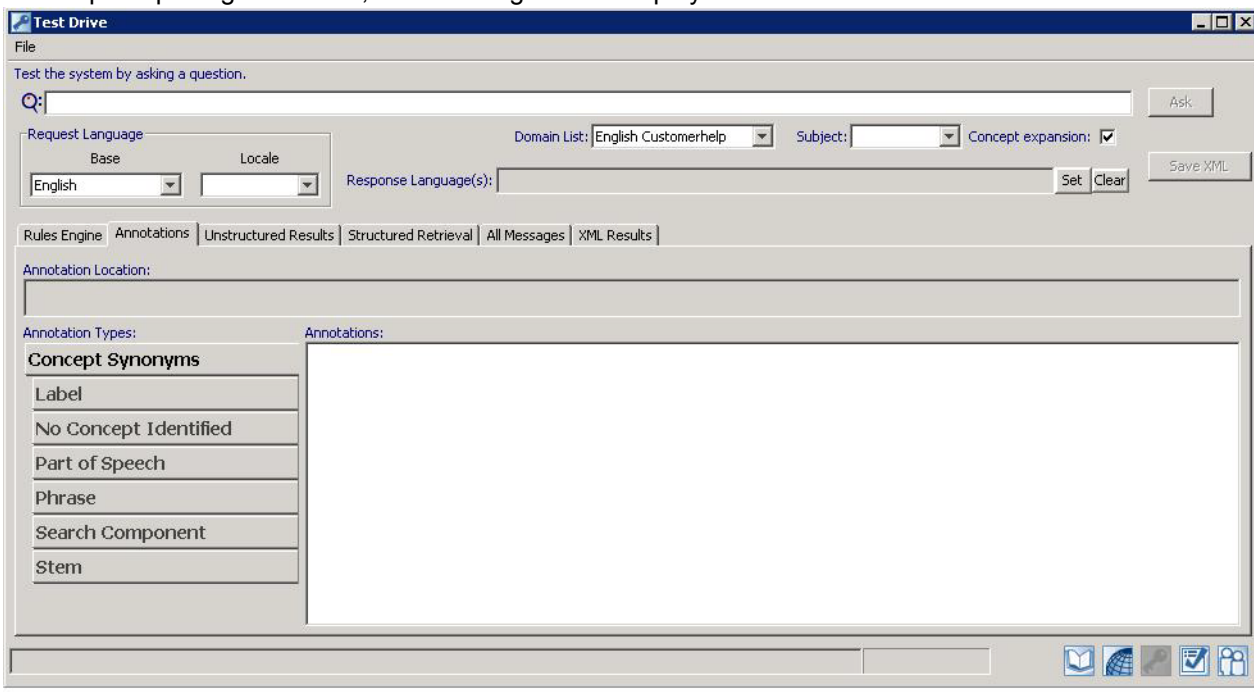
- How to enter a query, *Using Test Drive* on page 113
- How to read the results, focusing on the following tabs used for Language Tuning:
  - *Rules Engine Tab* on page 115
  - *Annotations Tab* on page 117
  - *Unstructured Results Tab* on page 126
  - *Structured Retrieval Tab* on page 127
  - *All Messages* on page 128
  - *XML Results* on page 128

Using Test Drive

To access Test Drive from the Oracle Knowledge Language Workbench Launcher:

1. Enter a valid User name and password
2. Select Test Drive from the Application drop-down menu
   To access Test Drive from other Oracle Knowledge Language Workbench applications:
3. Double-click on the Test Drive icon, displayed in the lower right corner of the main application window
   The Test Drive main window displays.

Upon opening Test Drive, the following screen displays:



4 Enter a query in the **Q:** field.

5 (Optional) Select a **Test Profile** from the drop down menu. The default is Default. For more on Test Profiles see *Oracle Knowledge Intelligent Search Optimization Guide*.

6 Select a **Domain List** from the drop down menu. The options listed reflect the default language.

Note: Upon selecting the **Language**, the **Domain List** drop down menu changes to reflect the selected language.

7 Select a **Language** from the drop down menu.

8 (Optional) Select a **Subject** from the drop down menu.

9 (Optional) Check the **Concept Expansion** box, if you have created new or modified Concepts and want to approximate results without committing to the Central Repository and reindexing. The Test Drive component automatically selects this option when you have made ontology changes in the workbench. Concept Expansion mimics what a fully-indexed system does with ontology changes and provides an approximation so that testing can be faster since a full index across all content does not need to be completed.

Concept Expansion expands into strings all concepts in the retrieval IML expressions generated from a query into their synsets. When the Concept Expansion box is selected, the search matches only concepts in single quoted strings; whereas with live search, concept matching matches retrieval IML concepts that are annotated in the content.

There are two main cases where a concept may not be annotated in the content:

- It corresponds to a range that is within a larger phrase.

The phrase “payment online” is contained within the phrase “bill payment online” so “payment online”is not annotated. However, Concept Expansion finds both since it expands all query concepts into their synsets, and the words “payment” and “online” occur in the synsets.

- It is ambiguous and not preferred.

If a word is in two different base concept synsets (so these concepts are ambiguous and not preferred), then neither of these concepts are annotated, and so search does not find either one. Again, Concept Expansion finds both since it expands all query concepts into their synsets, and the word occurs in the synsets.

Note: Test Drive results using Concept Expansion may differ from actual Oracle Knowledge Intelligent Search results. Concept Expansion simulates the effects of indexing very closely, but not exactly. To see accurate results using Test Drive, it is recommended you commit and reindex after creating a new or modified Concept.
If you are tuning with Test Drive, you must reindex and verify results either in the UI or in Test Drive with Concept Expansion OFF before concluding the real impact of the tuning.

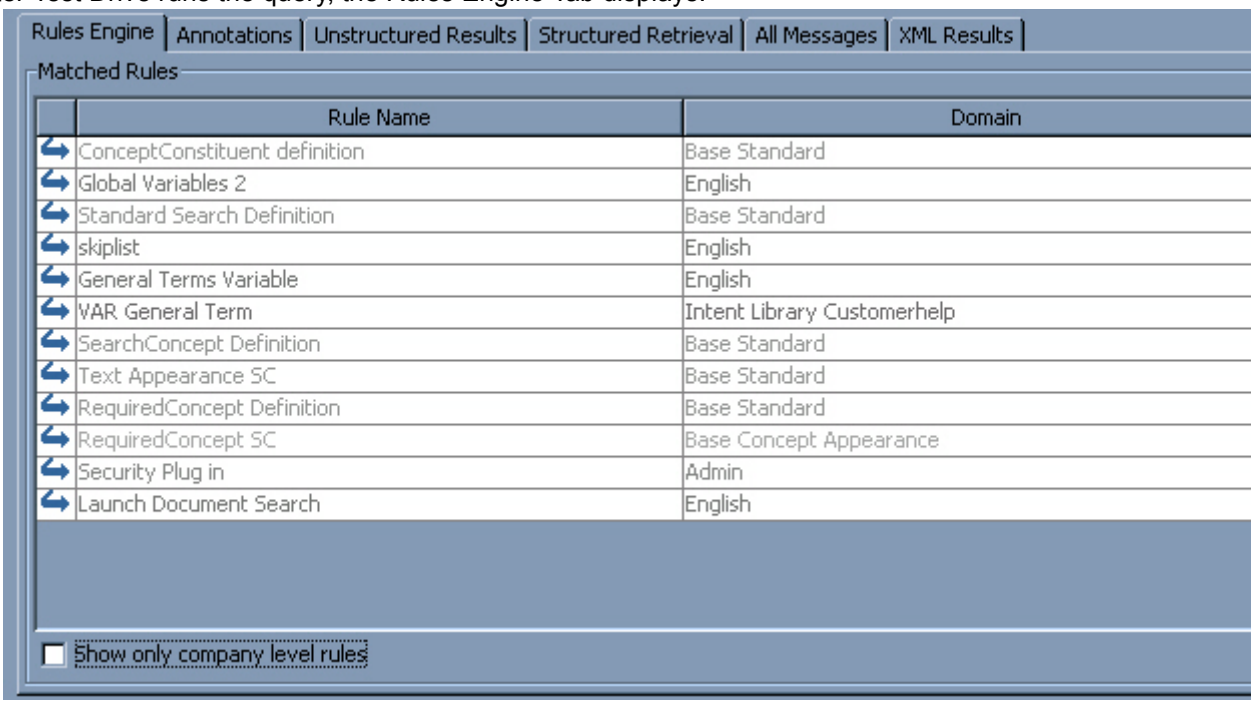
10 Select **Ask**.

Upon selecting **Ask**, Test Drive runs the query using Oracle Knowledge Intelligent Search. The search results are returned in a separate browser. The tabs located in the lower portion of the Test Drive window now contain detailed information regarding your test query. The following sections detail all available results tabs, however the most useful for Oracle Knowledge Language Tuning are:

- *Rules Engine Tab* on page 115
- *Annotations Tab* on page 117
- *Unstructured Results Tab* on page 126

Rules Engine Tab

After Test Drive runs the query, the Rules Engine Tab displays:



The Rules Engine Tab lists all the Rules and Intents that matched the test query. The list is separated into two columns: the Rule Name, and the Domain in which the Rule/Intent belongs. The Rules/Intents are listed in the order in which they were matched by Oracle Knowledge Intelligent Search.

- Check the **Show only company level rules** to see Rules/Intents that are affiliated with Company Level Domains.

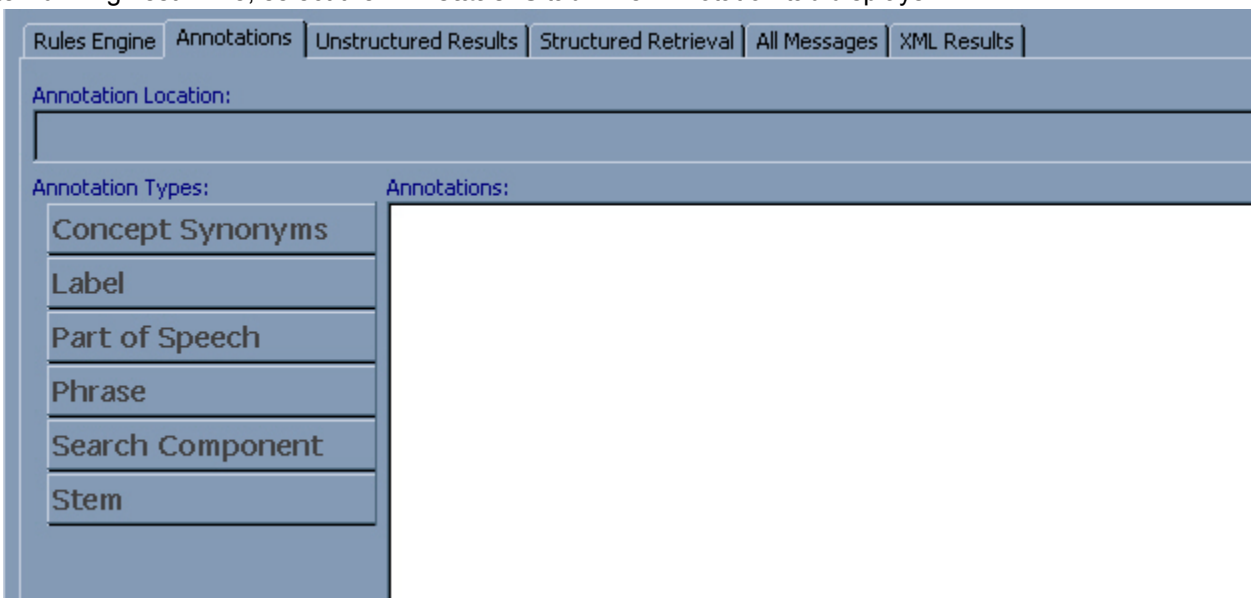
Note: It is recommended that you do not adjust rules other than those listed in Company Level Domains, unless instructed to do so.

- To view or modify a matched Rule/Intent, double-click on the object. The Rule/Intent displays in Dictionary Manager. For more information, please see *Rules* on page 61 or *Intents, Intent Hierarchies, Intent Responses* on page 44.

Annotations Tab

The Annotations Tab displays information regarding the words and phrases that occur in the query and the Concepts associated with them.

After running Test Drive, select the **Annotations** tab. The Annotation tab displays:



- Select the **Annotations** tab to learn more about how Oracle Knowledge Intelligent Search annotated the query.

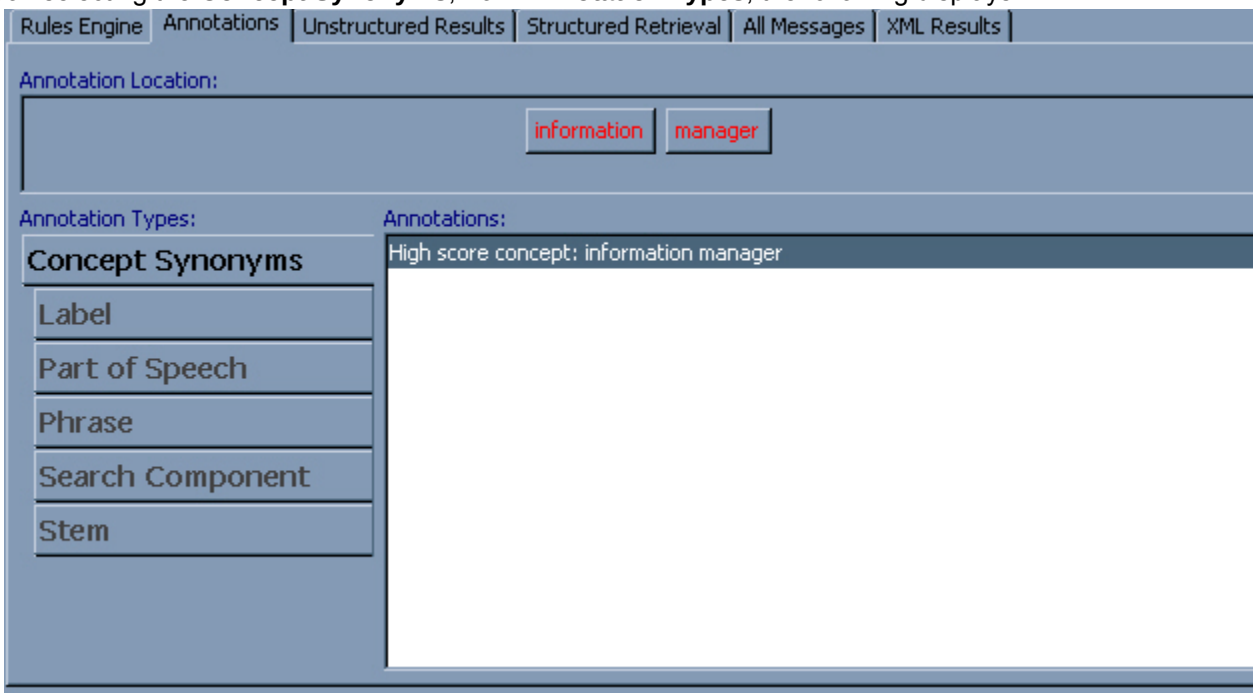
Note: The Annotations tab appears blank until an Annotation Type is selected.

For each Annotation Type, Test Drive displays annotations for the word displayed with a white background in the Annotations Location. If no word is displayed with a white background, the Annotations field contains all data associated with the query.

The annotation types considered the most useful for tuning are: *Concept Synonyms* on page 118 and *Search Component* on page 124.

Concept Synonyms

Upon selecting the **Concept Synonyms**, from **Annotation Types**, the following displays:



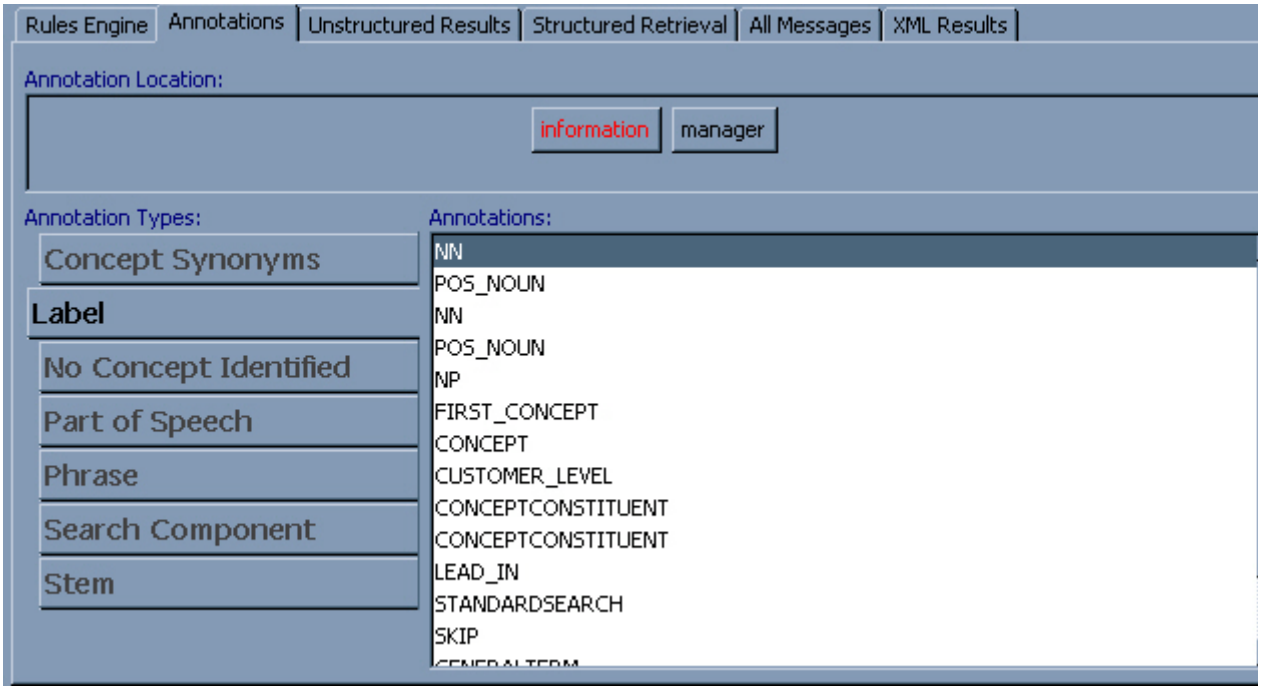
The **Annotation Location** highlights in red those words that have been annotated as Concepts created in Dictionary Manager. Words appearing in black are not yet included in the Ontology.

Under the **Annotations** field, appears the Concept Score for the Concept that appears in the Annotation Location as red text with a white background.

- To view a matched Concept, double-click on the Concept Score listed in the field. The Concept displays in Dictionary Manager. For more information on Concepts and Concept Scores, please see *Concepts* on page 19.

Label

Upon selecting the **Label**, from **Annotation Types**, the following displays:



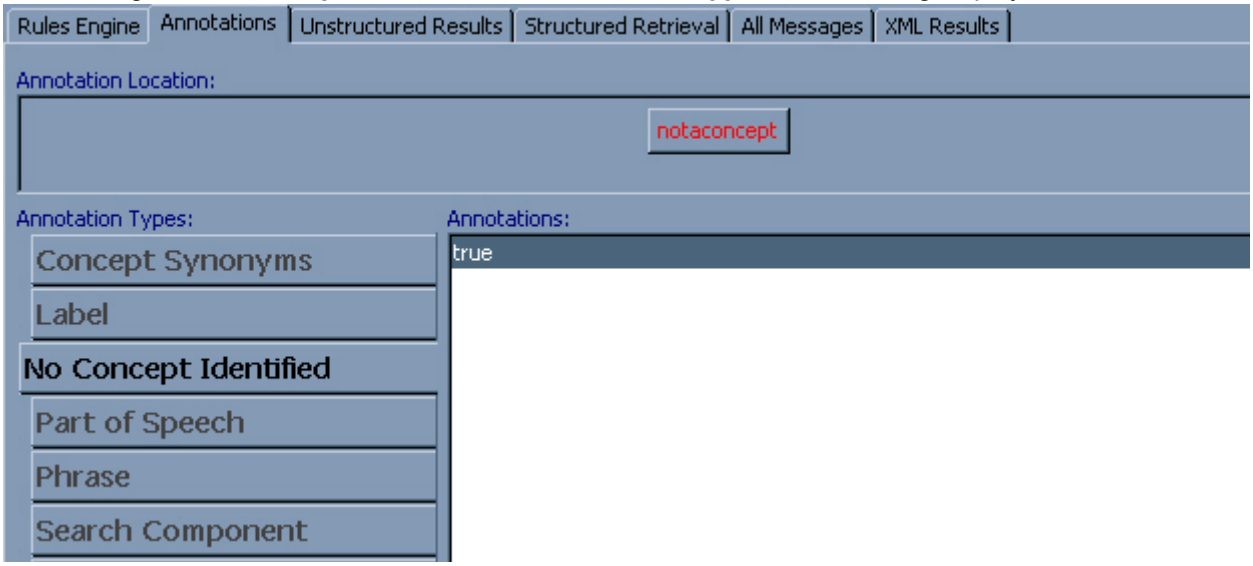
The Annotations field lists all the labels assigned to each word of the query.

To view the labels associated with a particular word, select the word from the Annotation Location. Once selected the word has a white background.

For more information on labels, please see *Language Analysis Labels* on page 151.

No Concept Identified

Upon selecting the **No Concept Identified**, from **Annotation Types**, the following displays:



The No Concept Identified type displays the word TRUE if the word selected from the Annotation Location with the white background is not currently a Concept in Dictionary Manager.

Note: Words that are labeled as skip words receive a value of true. Skip words are words included on *Skip Lists* on page 120.

Skip Lists

Oracle Knowledge Intelligent Search includes preconfigured Rules named **Skiplist** for each fully supported language. The Rule contains a list of common words and phrases, such as definite and indefinite pronouns, that are set to the global variable #SKIP. These words are then “skipped”, or ignored by Intelligent Search during question processing. Words that appear on the Skiplist are not annotated with any concepts, nor are they used in triggering Search Component Rules.

#SKIP_INDEX

#SKIP_INDEX is unrelated to #SKIP. #SKIP_INDEX removes tokens from the index; #SKIP removes tokens from search questions. The installed implementation includes four punctuations (. , | !) as index skipwords.

Adding or Removing Tokens in #SKIP_INDEX

The best practice is to save the ‘VAR SKIP_INDEX’ rule as a rule of the SAME name in your custom domain. This allows the new custom rule to overwrite the installed rule while keeping the installed rule intact for your reference. You can then add or remove #SKIP_INDEX elements in the custom rule as regular IML patterns.

Updating Installed Search Component Rules Affected by #SKIP_INDEX Logic

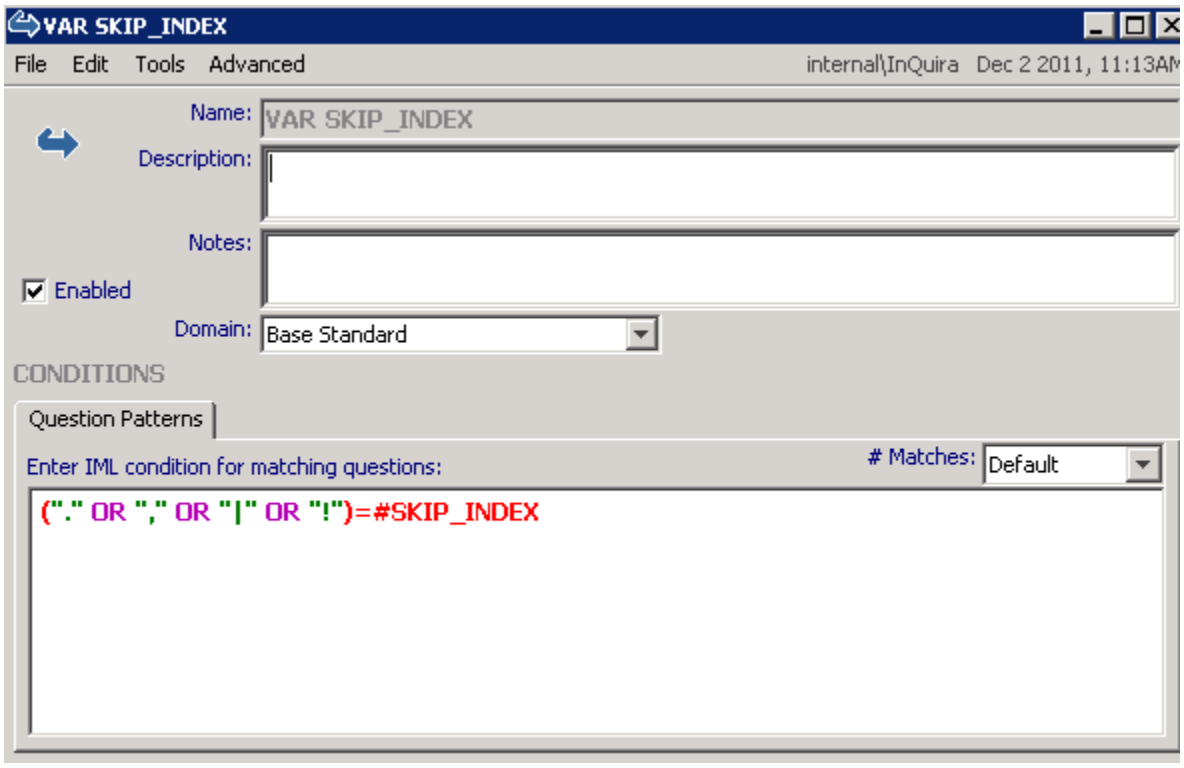
You must manually update all search component rules related to #SKIP_INDEX after you change #SKIP_INDEX. In Dictionary Manager, go to the find tool, Rule option and search for Answer Action Text contains **[** and the installed rules (12 in total) that appear are the ones to modify.

Maintaining Custom Search Component Rules Affected by #SKIP_INDEX Logic

If you create any search components that read a variable and manipulate text, and that variable can contain #SKIP_INDEX tokens, you’ll need to duplicate the logic of the regex statement into your rules, and maintain these rules as well.

Use: `TEXT( OPERATION="replacechar" , ARG1="\"[.,|!]\"", ARG2="WORD")`
for tokens surrounded by double quotes

Use: `TEXT( OPERATION="replacechar" , ARG1="'[.,|!]'” , ARG2="WORD" )`
for tokens surrounded by single quotes



Search Impact

- Regular questions

  Regular questions are not affected. For example, 'install . program' is expected to return the same search results regardless of whether '.' is on the #SKIP_INDEX list.

- Pseudo token search

  Pseudo token search can be affected because being part of a pseudo token does not prevent the token from being removed from the index. For example, when '!' is on the #SKIP_INDEX list, documents containing 'install 1!0' will not be scored higher than documents containing 'install 1x0' for the search question 'install 1!0'. This impact can be mitigated by one of the two following methods:

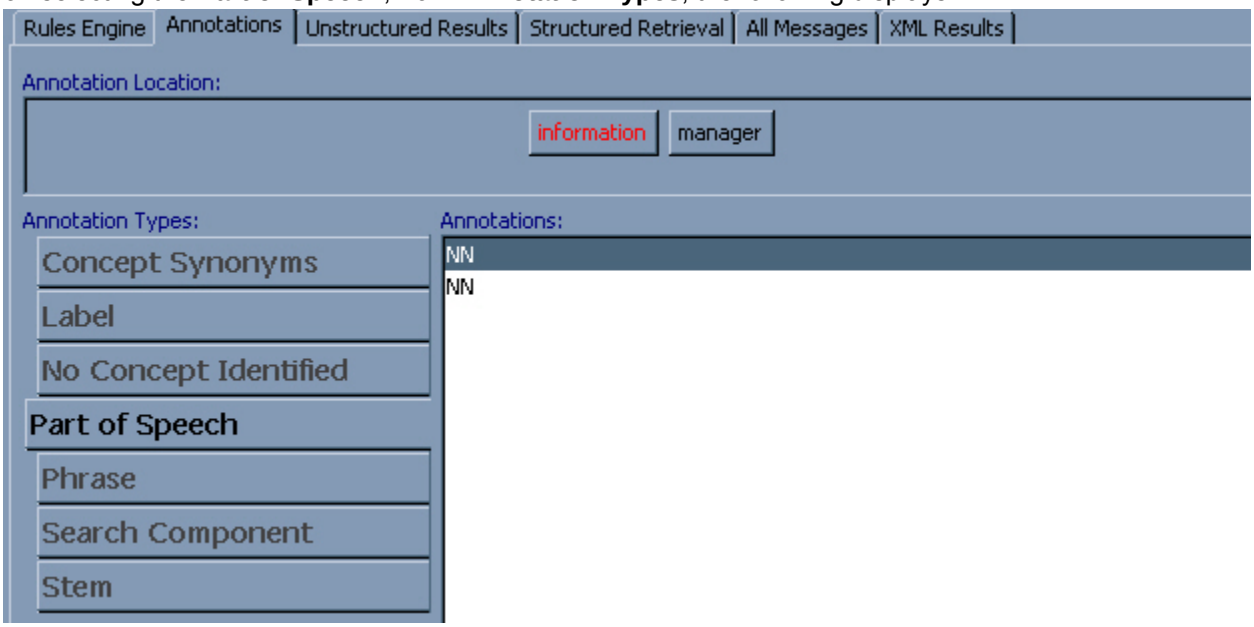
  - Create Openclass concepts: Language patterns for affected pseudo tokens (e.g. 1!0) can be captured in an Openclass concept, which allows these pseudo tokens to be retained in the index as concepts. For example, the installed 'software version Openclass' concept allows pseudo tokens of number and period combinations (e.g. '1.0') to be retained in the index regardless of whether '.' is on the #SKIP_INDEX list.
  - Remove specific tokens from the #SKIP_INDEX list. See the *Adding or Removing Tokens in #SKIP_INDEX* on page 120 for more information.

- Quoted phrase search

  Quoted phrase search can be affected because token removal from index can't predict what tokens may appear in quoted phrase search. The installed implementation matches #SKIP_INDEX tokens inside quoted phrase to any token in the document. For example, "install . program" matches all documents containing one token between 'install' and 'program' (e.g. install . program, install the program, install my program, etc.)

Part of Speech

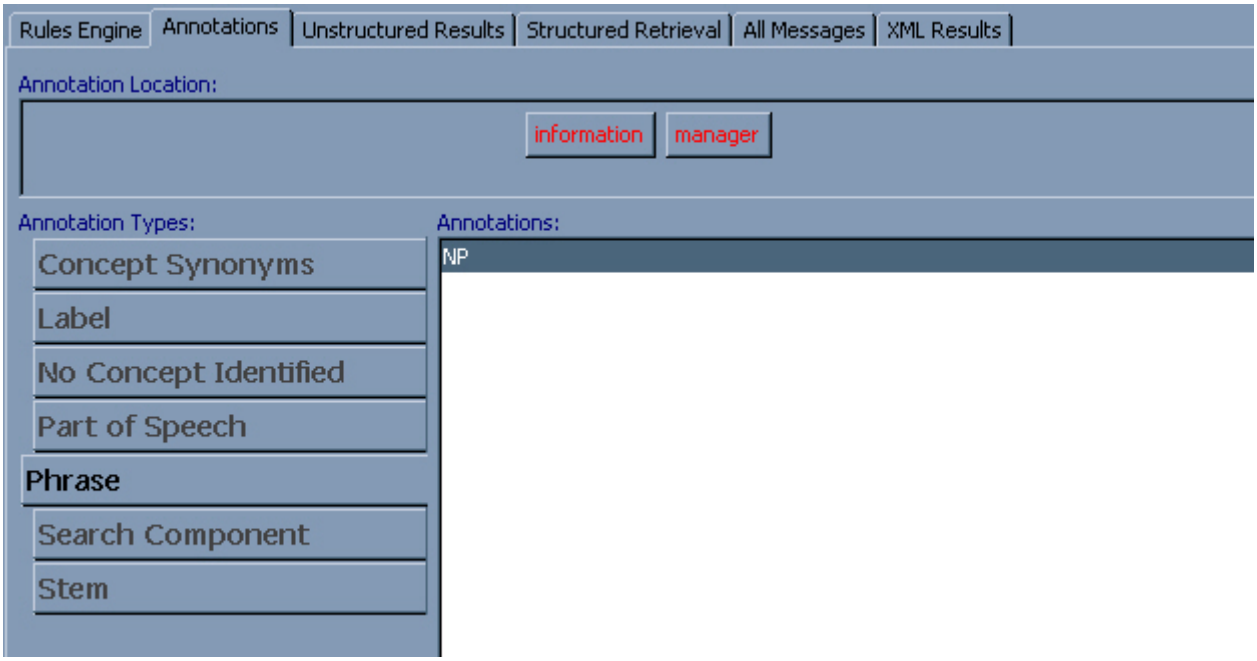
Upon selecting the **Part of Speech**, from **Annotation Types**, the following displays:



The Language Analyzer identifies and annotates the part of speech of words within both queries and indexed content using the tags described in *Part of Speech Annotations* on page 152

Phrase

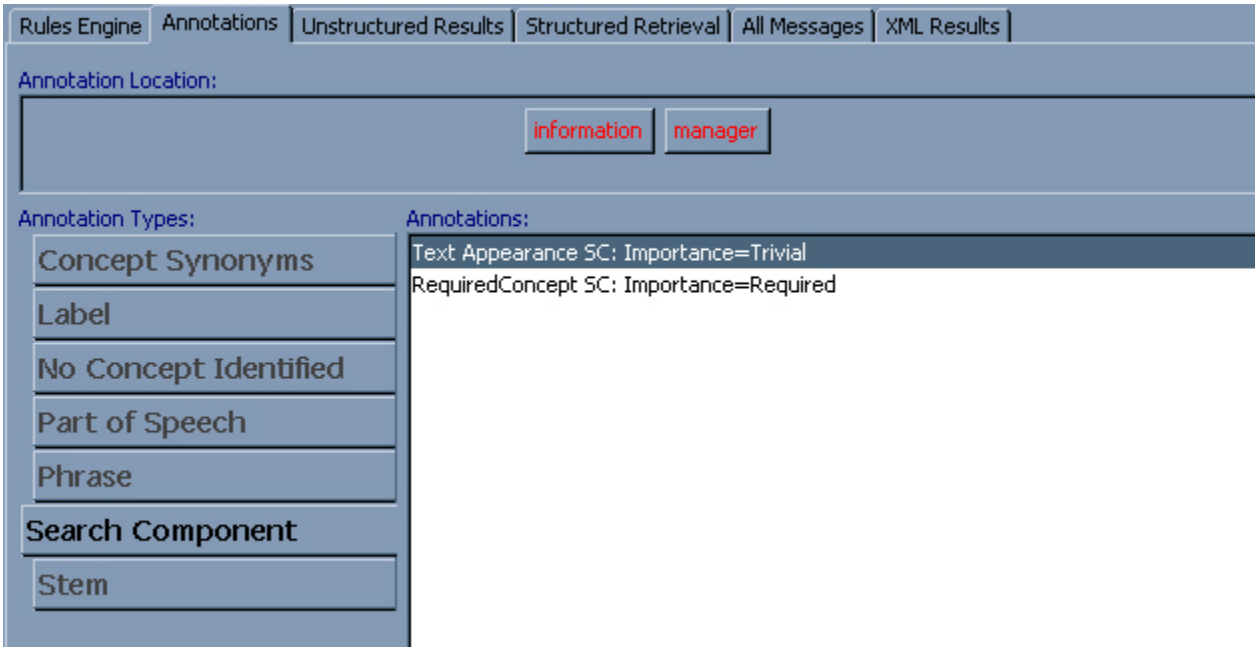
Upon selecting the **Phrase**, from **Annotation Types**, the following displays:



The Language Analyzer identifies and annotates the phrases within both queries and indexed content using the tags described in *Phrase Annotations* on page 153

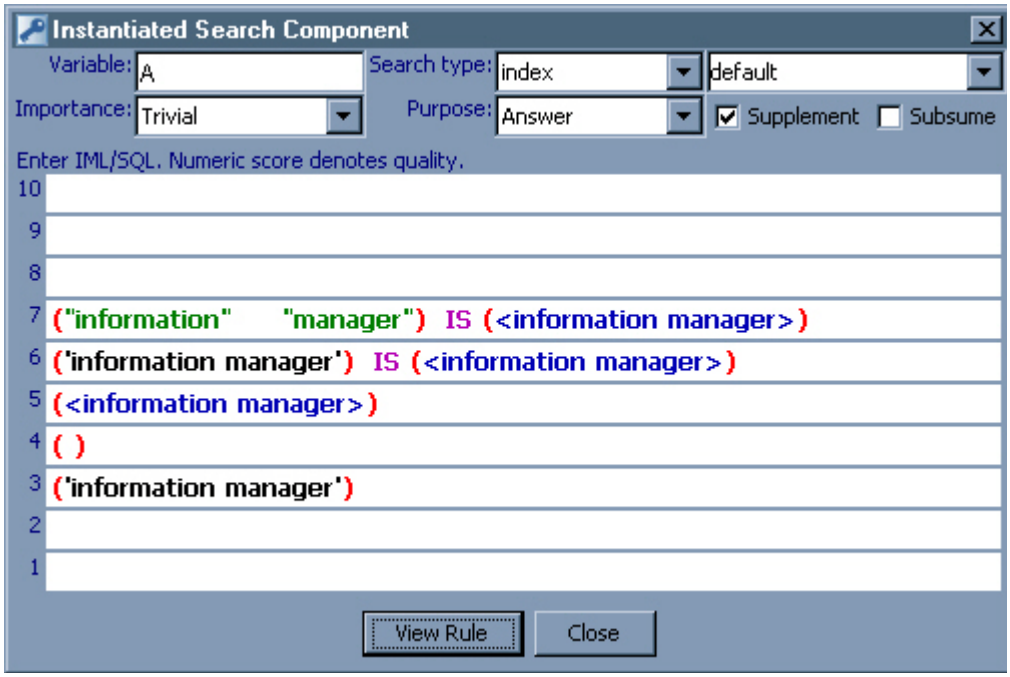
Search Component

Upon selecting the **Search Component**, from **Annotation Types**, the following displays:



Under the **Annotations** field, appears the Search Component Rules for the Concept that appears in the Annotation Location as red text with a white background.

To view the details of how the Concept matched the Search Component Rule, double-click on the Search Component Rule listed in the field. The Instantiated Search Component box displays:



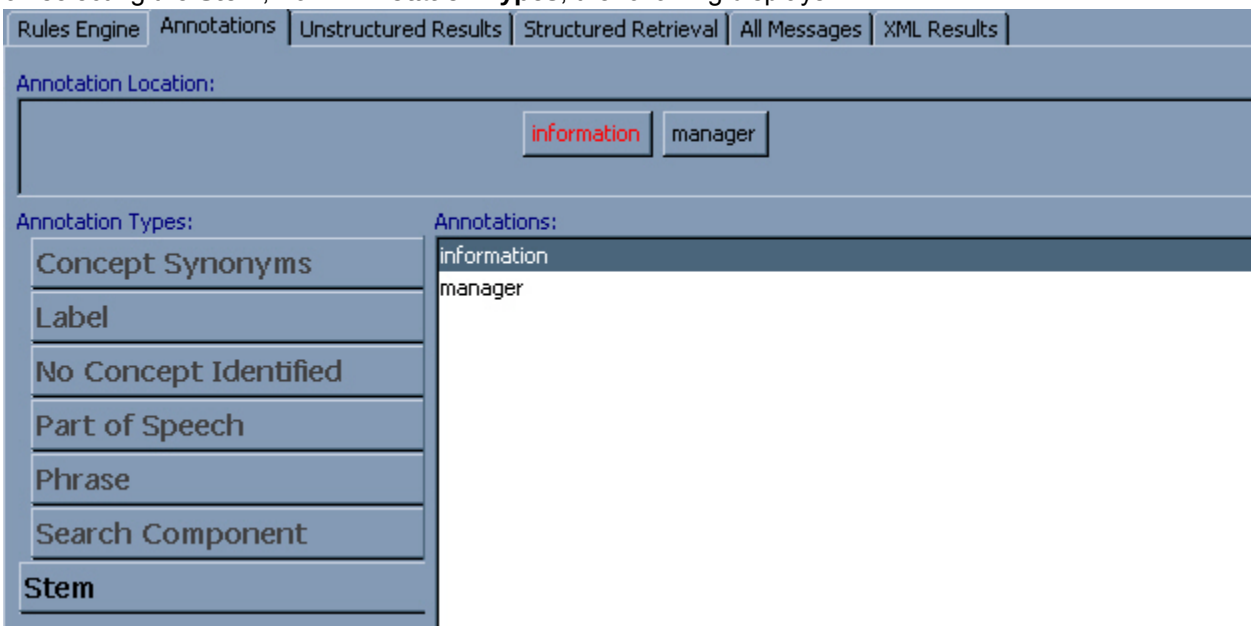
The Instantiated Search Component box details how the highlighted Concept matches the Search Component Rule and the points associated with that match. For example, the above assigns: seven points

when the literal phrase “information manager” appears in the result and that match is annotated with the <information manager> Concept, six points when a canonical match on “information manager,” such as “information managers”, appears in the results and that match is annotated with the <information manager> Concept, five points if any synonyms associated with the Concept <information manager> appears, and three points for the key word match on “information manager”

To view the Search Component Rule, select **View Rule**. The Search Component Rule displays in Dictionary Manager. For more information, please see *Rules* on page 61.

Stem

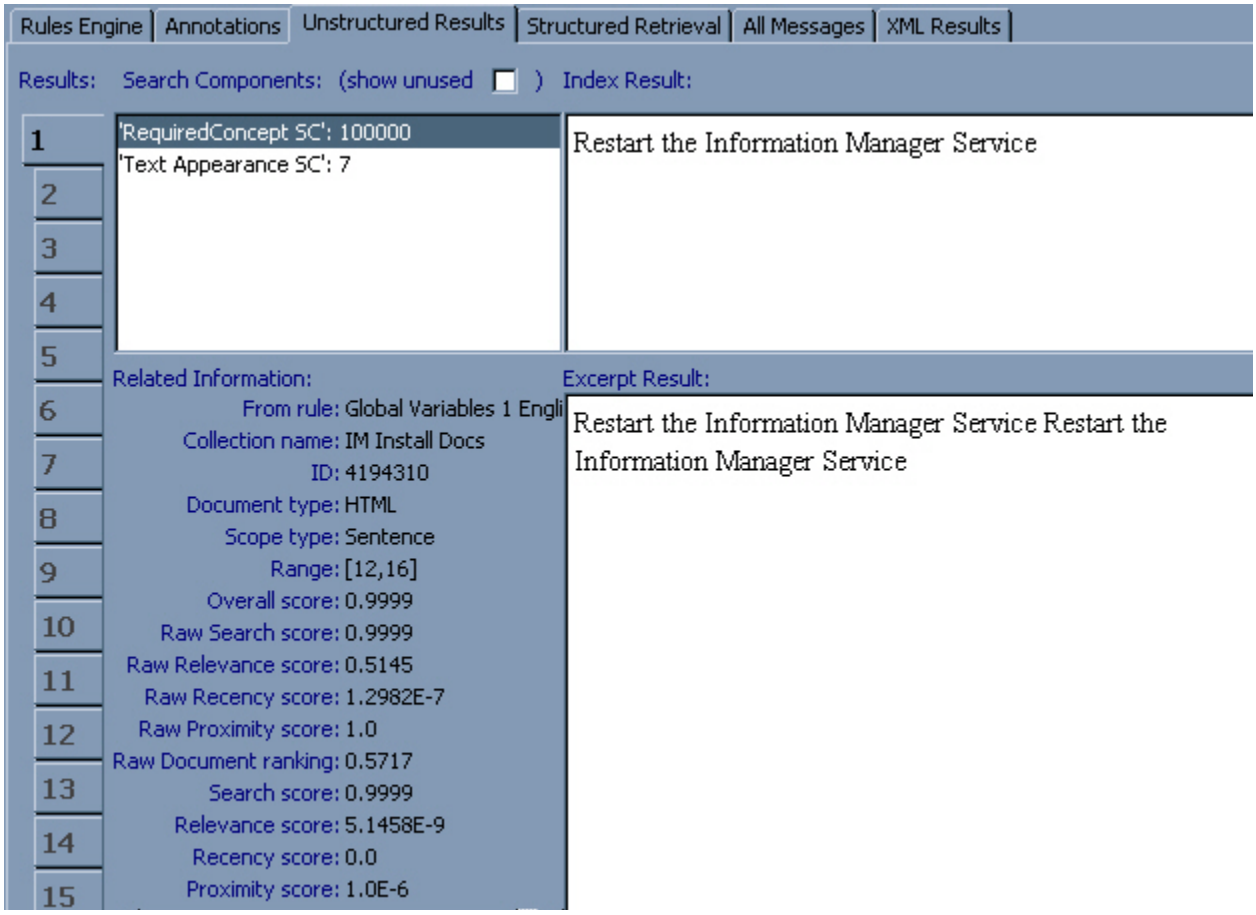
Upon selecting the **Stem**, from **Annotation Types**, the following displays:



Under the **Annotations** field, the stem, or basic unmarked form, of the selected word appears. For more information on stemming, please see *Concepts* on page 19.

Unstructured Results Tab

After running Test Drive, select the **Unstructured Results** tab. Unstructured Results refer to the results that are returned from your company's crawled content. The Unstructured Results tab displays:



On the left-hand side, under **Results,** is a list of numbers.These numbers correspond to the results displayed on the search results page after selecting **Ask** in Test Drive.

Select a **Number tab** to view how many points were assigned to that particular search results returned due to the Search Component Rules. Numbers correspond to the order in which results appear on the search results page.

Note: The Unstructured Results tab appears blank until a number is selected.

For each number, Test Drive displays the Rules and Intents used to match the query with the result in the **Search Component** field.

To see the Search Component Rules that did not contribute to the score of the document, check the **show unused** box.

To view the details of how the query matched the Search Component Rule, double-click on the Search Component Rule listed in the field. The Instantiated Search Component box displays.

For each number, the Unstructured Results tab in Test Drive displays the matched index content associated with the result in the **Index Result** field. As you select a Search Component Rule from the Search

Component field, the text is highlighted red to represent the words or phrases matched by the selected Search Component Rule.

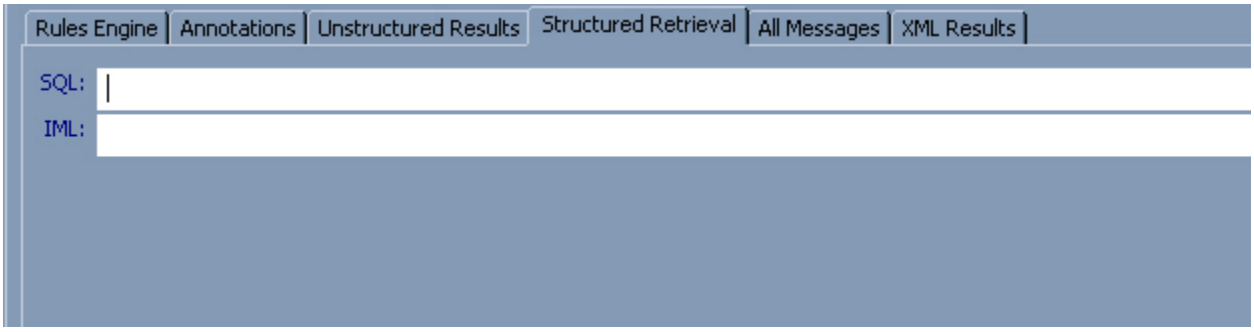
The Unstructured Results tab also displays the excerpt that displays on the Search Results page, for each number, in the **Excerpt Result** field.

The Unstructured Results Tab also includes the following information:

Field	Description
From Rule	Displays the name of the Rule that retrieved the selected answer.
Collection name	Displays the name of the document collection in which the answer is located. Document collections are defined within the content acquisition process.
ID	Displays the ID number assigned to the answer source document during the content acquisition process.
Document type	Displays the file type of the answer document, such as HTML or PDF.
Scope type	Displays the scope type associated with the matched answer. Scope types include `sentence`, `section`, and `document`.
Range	Displays the offsets, which indicate the positions of tokens (words) within the index, associated with the beginning and end of the Search Component scope.
Overall Score	Displays the overall score, which is calculated using weighted averaging and normalization of the component search, recency, and relevance scores.
Search Score	Displays the search score.
Recency Score	Displays the document recency score.
Relevancy Score	Displays the document relevance score.
Document Ranking	Displays a weighted average of links to the answer source document.

Structured Retrieval Tab

After running Test Drive, select the **Structured Retrieval** tab. The Structured Retrieval tab displays:

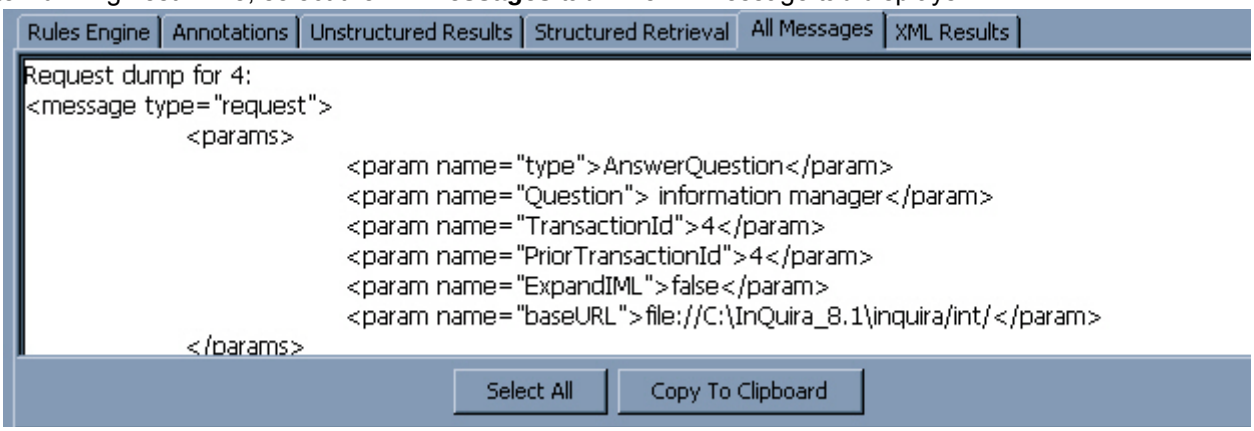


If Test Drive has used a Structured database, then the Structured Retrieval tab displays:

- The SQL statement generated by the SQL Rule matched by the test query.
- The IML Statement used to restrict the SQL query results displayed.

All Messages

After running Test Drive, select the **All Messages** tab. The All Message tab displays:

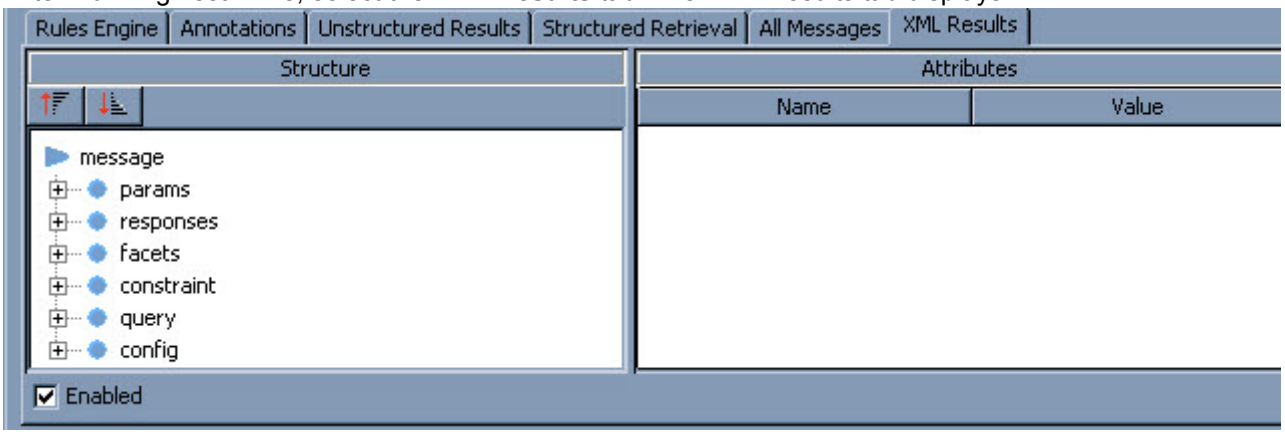


The All Messages tab displays all messages written to the log file as a result of running Test Drive. This information can be useful when troubleshooting Oracle Knowledge Intelligent Search.

- Select **Select All** and then **Copy to Clipboard** to save the messages.

XML Results

After Running Test Drive, select the **XML Results** tab. The XML Results tab displays:



The XML Results tab displays the XML elements that make up the search results, and the attributes associated with the XML elements. The XML elements appear under the **Structure** as a tree structure.

- Select a node from the tree structure under **message**. Choose from:
  - *Parameter Element* on page 129
  - *Responses Element* on page 129
  - *Facets Element* on page 129
  - *Constraint Element* on page 129
  - *Query Element* on page 129
  - *Configuration Element* on page 130

Parameter Element

The XML results tab displays various parameter elements associated with the request and the response. Request-related parameters include:

Name	Value
type	This value is the action completed. Usually “Answer Question.”
Question	This value is the text of the test query.
Transaction ID	This value is the unique ID assigned to the request.
Prior Transaction ID	This value is the unique ID assigned to the previous request within the user session.
Expand IML	This value is TRUE or FALSE.
Base URL	This value is the Base URL of the returned results.
ProcessorVersion	This value is the current version of Oracle Knowledge being used.

Responses Element

The XML Results tab displays the Responses Element in a tree structure. By expanding the tree structure, the Responses Element displays the purpose, title, link, and timestamp for each answer returned on the results page.

Facets Element

The XML Results tab displays the Facets Element associated with the results. The Facet Element uses a tree structure to display the available facets used within Oracle Knowledge Intelligent Search.

Constraint Element

The XML Results tab displays the Constraint Element. The Constraint Element includes any factors used by Oracle Knowledge to restrict results. The most common Constraint is **language** that restricts with Domain Groups to use when returning results.

Query Element

The Query Element contains the test question in both the Original form and a Paraphrase. The Original element is the original request as entered in the Ask field, whereas the Paraphrase element contains the revised request as rendered and processed by the User Interface.

Configuration Element

The Configuration Element includes the Search Within attribute that has a value of TRUE or FALSE.

CHAPTER 4

Quality Monitor

Quality Monitor is a component accessed from within the Language Workbench that allows you to evaluate and monitor the quality of search results. You can use Quality Monitor and the Grading Tool to analyze unsatisfactory search results, identify problems with Dictionary Objects, gaps in your company's content, and tune Oracle Knowledge Intelligent search accordingly. To access Quality Monitor see *Quality Monitor* on page 9.

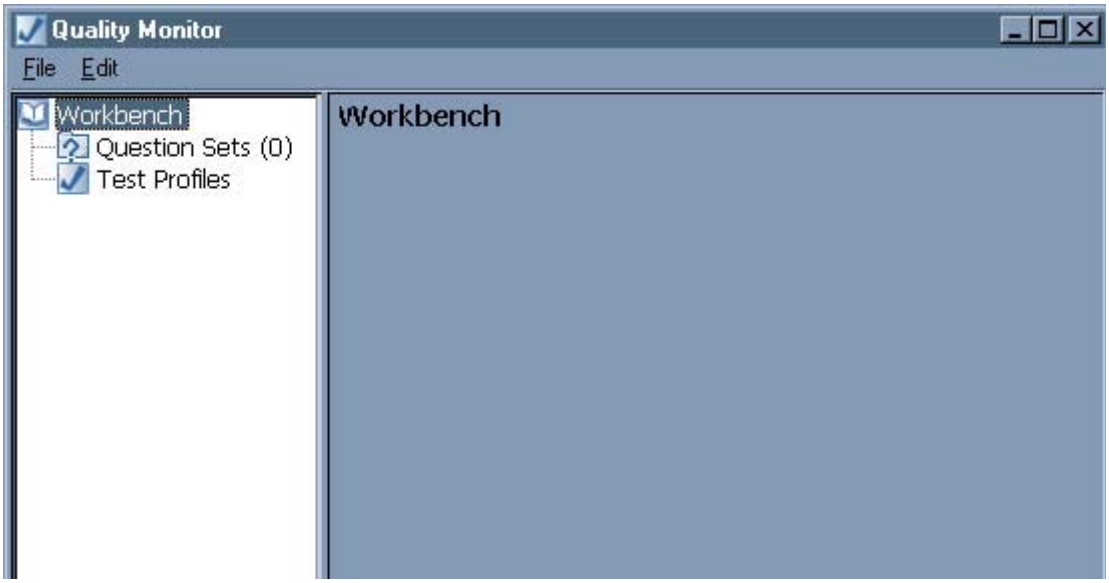
This Chapter explains:

- *Using Quality Monitor* on page 132
- *Test Profiles* on page 136
- *Test a Question Set* on page 138
- *Grade a Question Set* on page 138

Note: Before using Quality Monitor you must configure the Quality Monitor data source by accessing the **Workbench** menu with in the **Advanced Configuration Facility**. For more information, please see *Oracle Knowledge Intelligent Search Installation Guide*.

Using Quality Monitor

Upon opening Quality Monitor, the following screen displays:



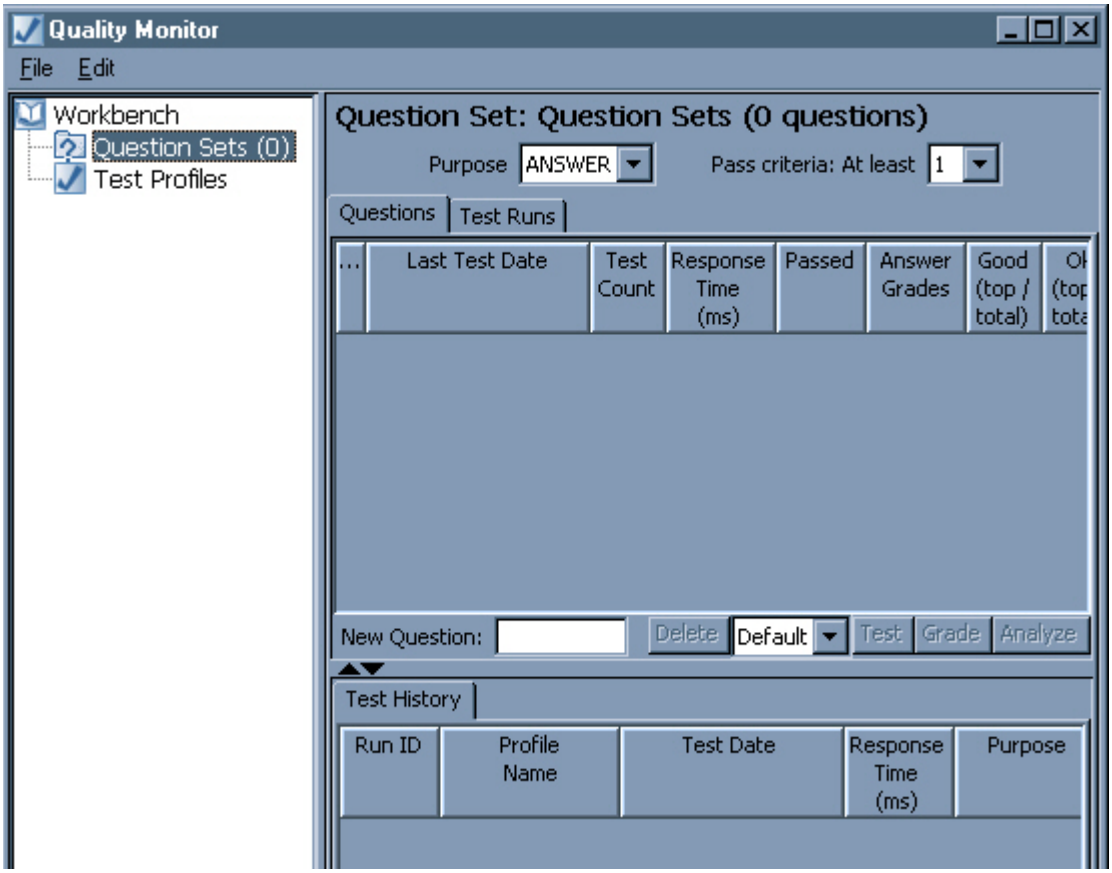
On the left side of the Quality Monitor screen is a tree structure. The tree structure is separated into two parts:

Question Sets on page 133

Test Profiles on page 136

Question Sets

Select **Question Sets**. The Question Sets screen displays:



Add Question Set

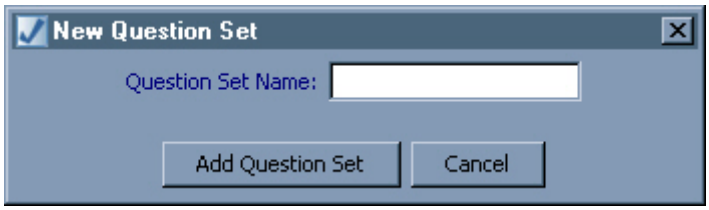
Questions used in Quality Monitor are visible to Language Workbench users with the Quality Monitor permission set. For more information, please see *Oracle Knowledge Intelligent Search Optimization Guide*. In order to keep question sets organized it may be helpful to create various question set folders.

Note: Oracle recommends you create a Question Set hierarchy that is organized by the Tuning Cycle, the date, and the name of the grader, for ease of reference.

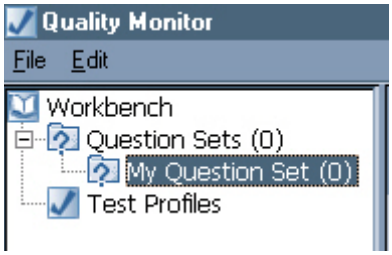
To create a question set folder:

1 Select **Add Question Set** from the **Edit** menu in the Quality Monitor window. The New Question Set dialog displays:

You may also select the **Add Question Set** option by right-clicking on **Question Sets** from the left side of the Quality Monitor screen.



2 Enter a Name in the field, and select **Add Question Set**. The new Question Set name displays in the folder structure.



Import / Export Questions

Quality Monitor is used to grade a predefined list of queries. For more information, please see *Analytics Administration Guide*.

The following sections detail how to load user queries into Quality monitor:

- *Creating a Question List to import* on page 134
- *Import Question Set* on page 135
- *Import Intent Questions* on page 135
- *Import Questions and Grades* on page 135
- *Export Questions and Grades* on page 135
- *Add a New Question Manually* on page 136

Creating a Question List to import

In order to import a Question Set, you must have a list of predefined questions saved as a .txt file. If, for example, you are using a report saved as an .xls file, you need to copy the questions into another program. Each question must be on a separate line. Once this format is complete, save the file as a .txt file.

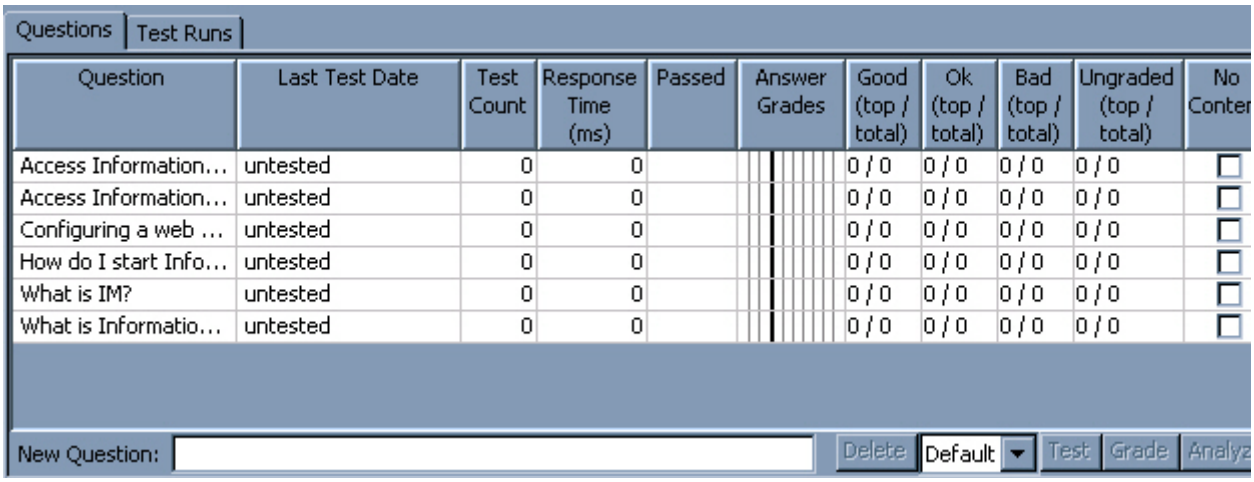
Import Question Set

To import a question set:

1. Select a Question Set folder from the left, and select **Import Question** from the **Edit** menu of the Quality Monitor window. The Import Question List (UTF-8Encoding) file finder displays.

   You may also select the **Import Question** option by right-clicking on a **Question Sets** folder from the left side of the Quality Monitor screen.

2. Select the *.txt file stored on your local machine and select **Import**. The queries appear in the Questions Tab, and the number of questions adds to the Question Set folder display.



Import Intent Questions

By selecting **Import Intent Questions** from the **Edit** menu, Quality Monitor enters all questions listed as Test Question in an Intent into the Questions field. For more information on Test Questions, please see *Create a New Intent in the Intent Hierarchy* on page 51.

Import Questions and Grades

You may import previously graded Quality Monitor questions that have been exported from an existing Oracle Knowledge application. For more information, please see *Export Questions and Grades* on page 135.

1. From the **Edit** menu, select **Import Questions and Grades**. The **Import Question and Grades** file finder displays.
2. Select the folder containing the *.rtml file and select **Import**. The questions and grades append to the Question Set.

EXPORT QUESTIONS AND GRADES

You may export a graded Question Set to review or import into an existing Oracle Knowledge application.

1. From the **Edit** menu, select **Export Questions and Grades**. The **Export Question and Grades** file finder displays.
2. Select a location and enter a name for the *.rtml file. Select **export**. The questions and grades save to the location specified.

Add a New Question Manually

In addition to importing questions as described above, you may also add a question manually to be tested by Quality Monitor.

- Under the Questions tab of Quality Monitor, enter text into the **New Question** field, push enter. The new question adds to the Question Set.

Test Profiles

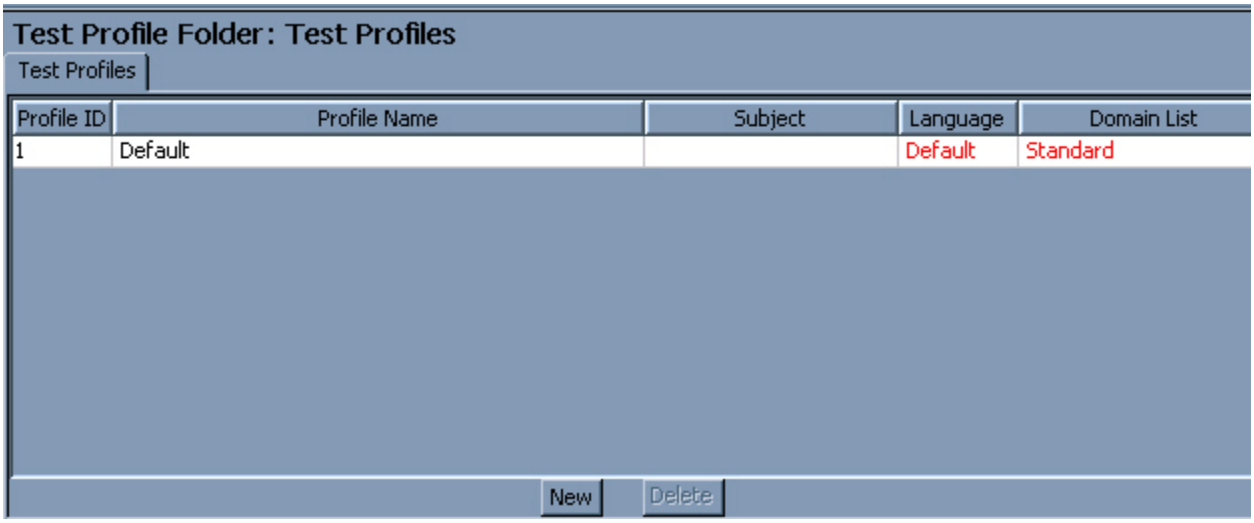
Test Profiles are used to define specific testing environments within the Quality Monitor. You can create and maintain any number of test profiles to simulate a particular type of user context. For example, you can define test profiles with:

- Specific domains of Rule and Concept information.
- Separate supported languages.
- Specific user contexts or business conditions.
- Optional questions required to start a question set.

Using Test Profiles, you can also test response quality using your current local dictionary instead of the Central Repository. For more information on the Central Repository, please see *Commit Dictionary Changes to Central Repository (vcroot)* on page 97.

The Quality Monitor associates each set of test results with the test profile used for that test, and maintains the test profiles as part of the test data. You cannot edit existing profiles, and you can delete profiles only if they are not referenced by any test results.

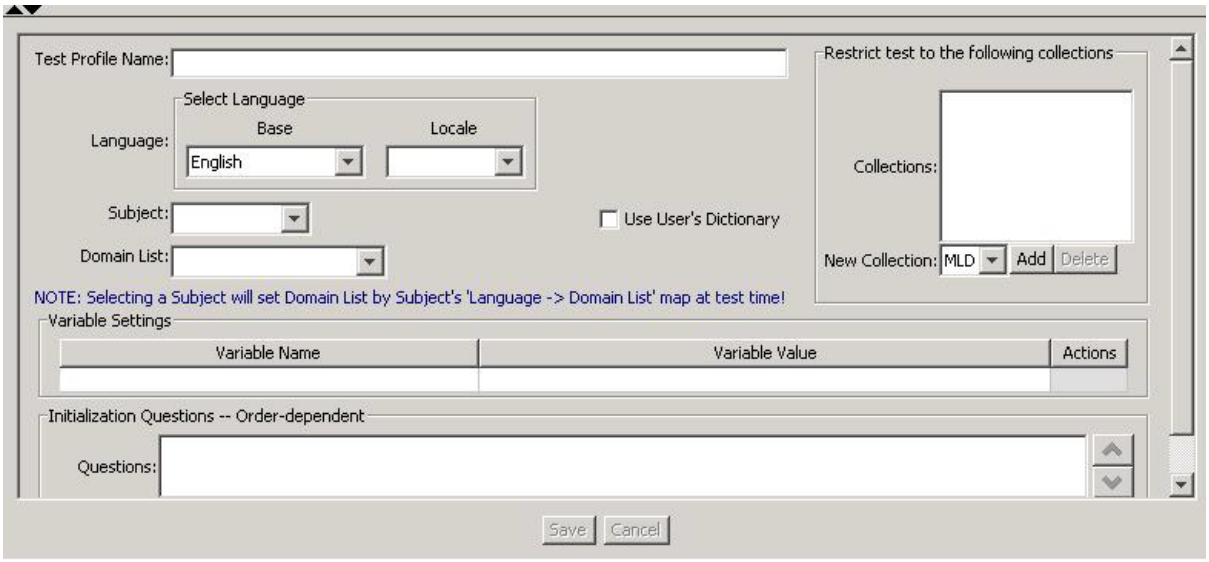
Select **Test Profiles**. The Question Sets screen displays.



Create a New Test Profile

To create a new test profile:

1. Select **New** from the Test Profile screen. The following window displays in the bottom of the Test Profiles window:



2. Enter a descriptive name in the **Test Profile Name**.
3. Select a **Language** from the drop down menu.
4. Select a **Domain List** from the drop down menu.
5. (Optional.) Select a **Subject** from the drop down menu.
6. (Optional.) Check the Use User's Dictionary box, if you want to test the quality of responses using dictionary objects stored in your local dictionary. Unchecked, Quality Monitor tests responses using dictionary objects stored in the Central Repository.
7. (Optional.) Enter **Variable Settings**. Using a context variable created in the Advanced Configuration Facility, see *Oracle Knowledge Intelligent Search Optimization Guide*, enter:
   - The **Variable Name**.
   - The **Variable Value**.
8. (Optional.) Enter an initialization question(s).

   Initialization Questions set parameters to set a user or business context, such as a user profile, location, or product of interest. The Quality Monitor excludes the responses from initialization questions from the test results set.
   - Enter an Initialization Question in the **New Question** field.
   - Select **Add**. The question displays in the Question field.
   - Enter additional questions.

   The Quality Monitor processes the initialization questions in the order in which they occur in the list. You can use the arrows at the right of the initialization Questions field to change the order of the questions as required to set the appropriate user context.
9. Select **Save**. The Quality Monitor displays the new profile in the profile list.

Note: You cannot edit existing test profiles. Once you have used a test profile to generate test results.

Delete a Test Profile

You can delete a test profile only if:

- It has not been used in a test.
- If it has been used in a test, then all results using the profile have been deleted.

To delete an unused profile:

1. Select the profile from the Profile tab.
2. Select the Delete button. The Quality Monitor deletes the selected process.

Note: The Delete process is immediate, there is no confirmation.

Test a Question Set

To test a Question Set after it appears in the Question tab:

1. Select a **Purpose** from the drop down menu. **Answer** is the default selection. For more information on purposes, see *Purpose (portlet)* on page 144
2. Define the **Pass Criteria**.
   - From the drop down menus select the minimum number of **Good** answers and the maximum number of **Bad** answers that may appear in the N **Top** answers.

   Note: Oracle Knowledge recommends setting the passing criteria to **At least 1 Acceptable Answer in the top 5 answers**.
3. (Optional.) Select a **Test Profile**.

   Note: Quality Monitor uses the test profile most recently used by your user ID as the default. Verify the test profile, and select an alternate test profile if desired.
4. Use Ctrl + A to select all question is a Question Set, or highlight the desired questions from a question set, and select **Test** or **Test Question Set** from the **Edit** menu of the Quality Monitor window.Quality Monitor begins the test process. Information regarding the progress of the test run is displayed at the bottom of the Quality Monitor screen.

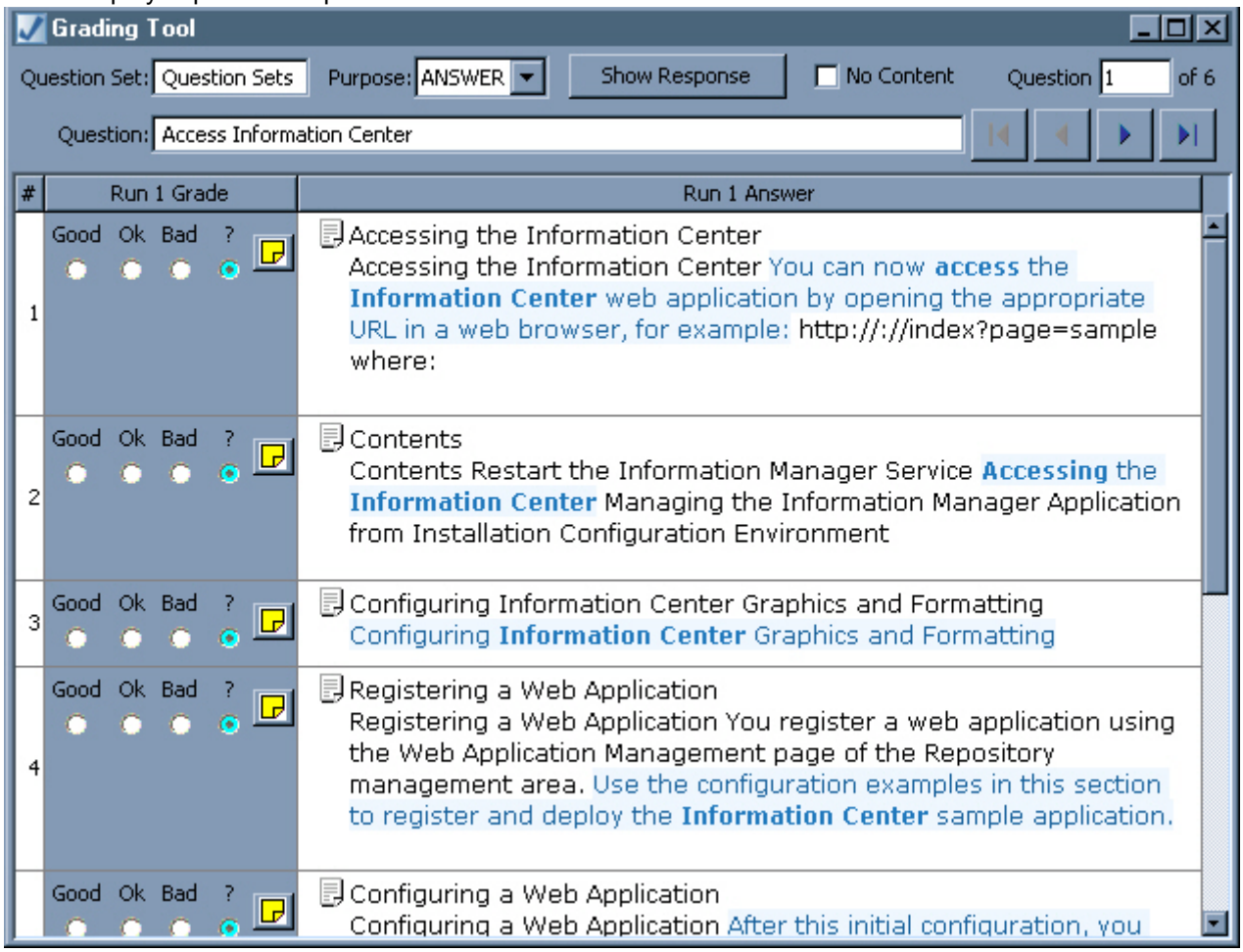
   You may also select the **Test Questions** option by right-clicking on a **Question Sets** folder from the left side of the Quality Monitor screen.

After the testing process is complete, the Last Test Date and Test Count columns of the Questions tab updates.

Grade a Question Set

After a Question Set has been tested, you may grade questions using the Grading Tool. To grade a question set:

1 Highlight the questions to test from a question set, and select **Grade**. The Grading Tool screen displays up to ten responses:



2 Using the set of radio buttons to the left of each answer, select the appropriate grade for each answer.

Note: There are no standard criteria for assigning a grade to an answer; the grade for an answer depends on criteria established by your organization, and the judgment of the grader. For suggested grading guidelines please see the following section *Grading Guidelines* on page 140.

3 If a question appears to have no relevant answer, select the **No Content** box.

Note: Once a question has been marked as having "**No Content**" the question is excluded from the overall test results.

4 Select the forward arrow at the top of the page to continue to the next question.

Quality Monitor immediately saves the assigned grades. You can change answer grades at any time during the grading session, or during subsequent sessions.

Upon closing the Grading Tool screen, the grades for each question display in the **Answer Grades** column of the Questions Tab. Additionally, the **Test Runs** tab displays the percentage of questions that passed the grading criteria.

Note: Oracle Knowledge considers 85% of the questions passing your grading criteria to be a well tuned dictionary.

After the Grading process is complete, it is advised that you tune your dictionary using the techniques discussed above in *Chapter 2, Dictionary Manager* and rerun Quality Monitor to *Compare Test Results* on page 140.

Grading Guidelines

The following criteria are suggested as guidelines for determining the grade of a question using the Grading Tool in Quality Monitor.

A **Good** answer meets one of the following conditions:

- It answers the question directly.
- It answers the question indirectly.
- It has a clear description indicating the linked page answers the question.

An **OK** answer has the following characteristics:

- It has a link that indicates the related page may have relevant information.
- It does not strike the use as a particularly good or bad answer according to the specified guidelines.

A **Bad** answer meets one of the following conditions:

- It is not at all related to the topic of the question.
- It makes the results appear to be a misunderstanding or malfunction of the search.
- It appears to be good, but the linked page does not contain the answer.

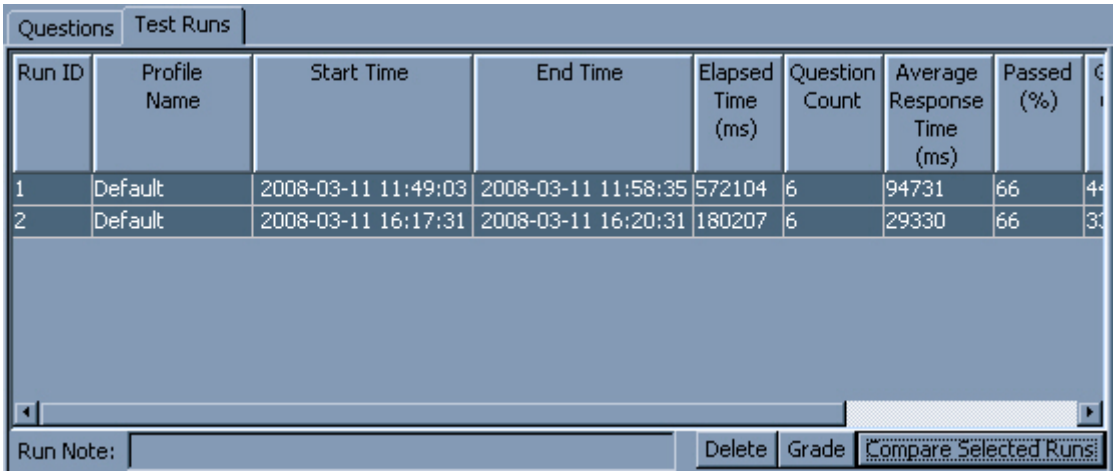
Analyze a Question Set

After a question has been tested you may select **Analyze** from the Questions tab. Upon selecting **Analyze** the selected test question is processed using Test Drive. For more information on Test Drive, please see *Chapter 3, Using Test Drive*.

Compare Test Results

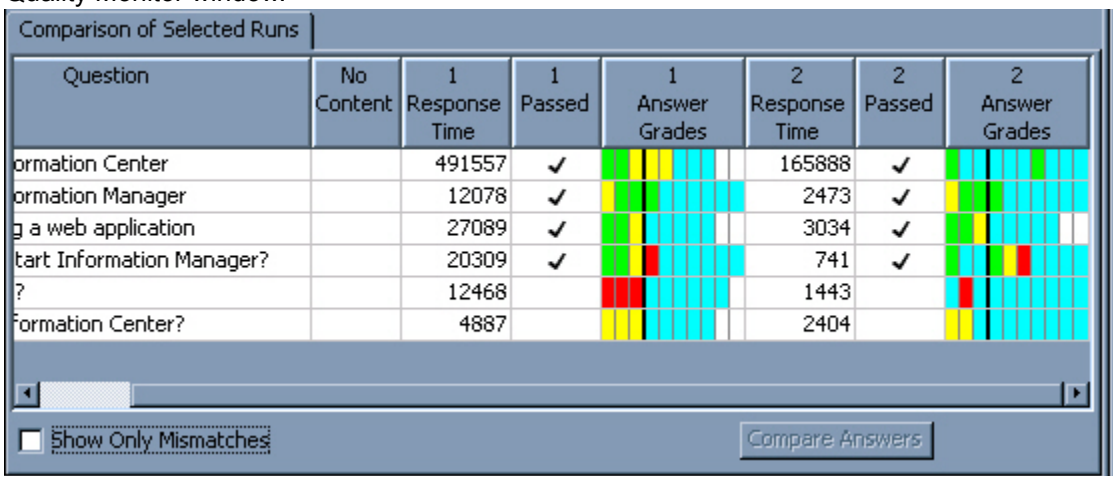
To compare the answers to Question Sets of multiple test runs:

1 Select the **Test Runs** tab. The Test Run tab displays:



2 Using the ctrl, shift function select multiple Test Runs.

3 Select **Compare Answers**. The Comparison of Selected Results tab displays at the bottom of the Quality Monitor window.



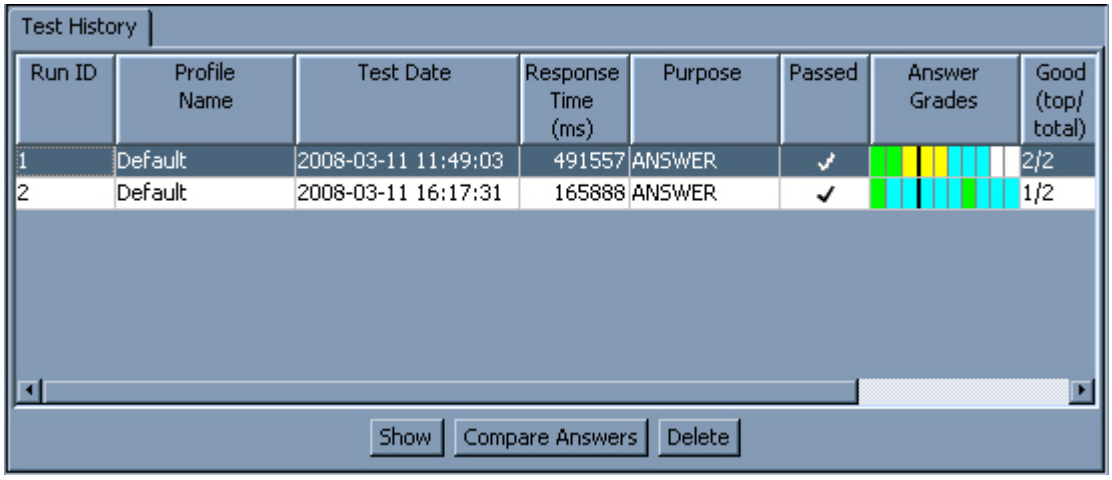
Using the following color code, answer grades for the selected runs display:

Green	Indicates a good answer.
Yellow	Indicates an acceptable (Ok) answer.
Red	Indicates a bad answer.
Cyan	Indicates an ungraded answer.
White	Indicates no answer.

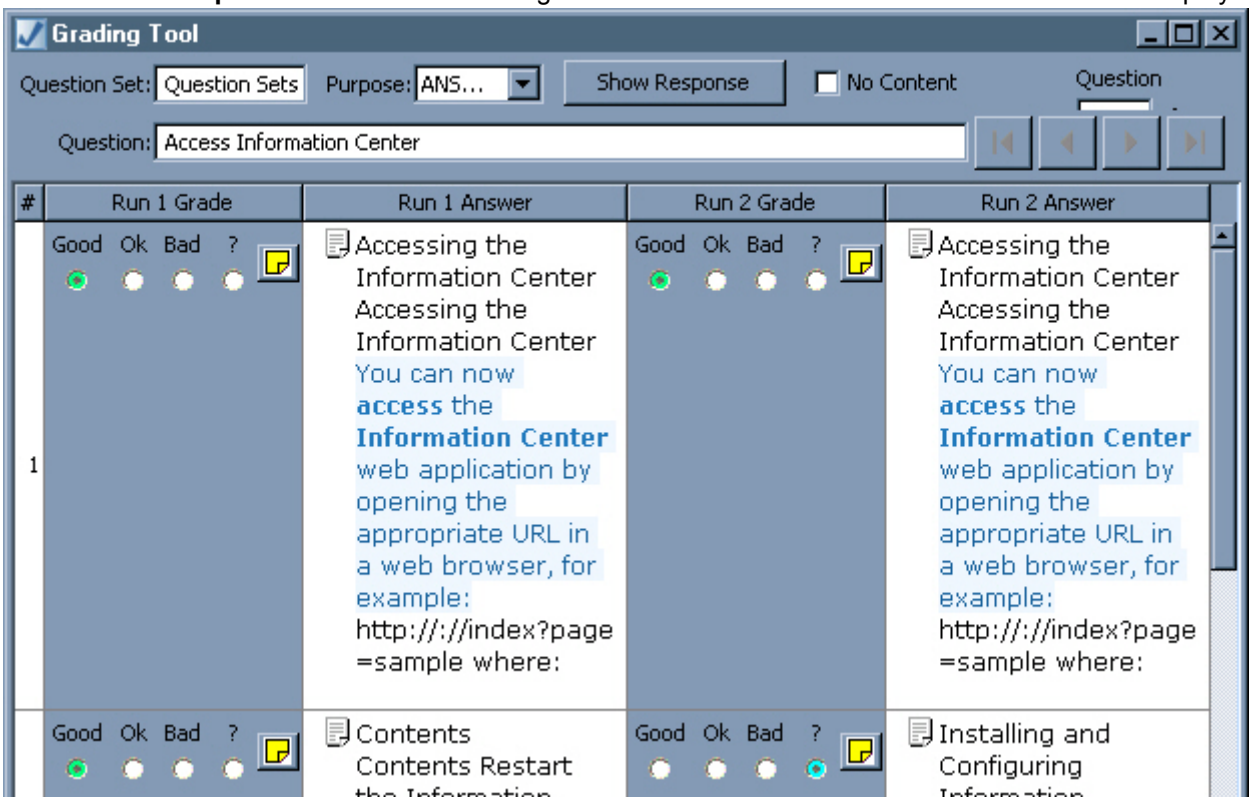
4 Check the **Show Only Mismatches** box to display only questions that have different results are grades across the selected Test Runs.

Test History

The Test History tab displays the answers for the selected question in the Questions tab for each Test Run:



1. Using the Ctrl, shift function, select multiple Test Runs.
2. Select **Compare Answers**. The Grading Tool screen with the answers to the selected runs displays:



Run Batch Jobs

To run testing in batch mode without opening up the Quality Monitor, run this command on the ICE prompt:

```
BatchQualityMonitor <question set name> <test profile name> <pass_min_good>
<pass_max_bad> <pass_top_answers> -p <purpose> -o <run statistics file path>
```

For example:

On Windows:

```
BatchQualityMonitor.bat "My Test Set" "Default" 1 0 3 -p Answer -o output.txt
```

On Linux:

```
BatchQualityMonitor.sh "My Test Set" "Default" 1 0 3 -p Answer -o output.txt
```

The output file records the statistics of the test. For example,

```
RunID=8
QuestionSet=test 1
TestProfile=Default
QuestionCount=1
StartTime=2012-08-15 16:18:43
EndTime=2012-08-15 16:19:06
ElapsedTime=23512
AvgResponseTime=129
ANSWER_PCT_PASSED=100
ANSWER_PCT_GOOD=33
ANSWER_PCT_OK=0
ANSWER_PCT_BAD=0
ANSWER_PCT_UNGRADED=66
```

APPENDIX A

Supplementary Tables

This chapter provides the supplementary tables and information to the previous chapters.

Purpose (portlet)

A purpose, also referred to as a portlet, corresponds to display characteristics defined in the User Interface, enabling you to establish consistent, focused and targeted presentation for various types of content. Choose from:

Answer	Displays responses that directly address the user's question in the main results section.(Default.)
Act	Displays links that provide actions that the user can take on the web site.
Promote	Displays cross-sell or up-sell advertisements for products related to the intent of the question.
Related Topic	Displays links to major topic categories defined for the web site.
Define	Displays links to terms used in the question as well as similar content.
Jump to Page	Displays content configured in the Dictionary for use with the direct page display feature.
Converse	Displays conversational response intended for use with a virtual representative on the response page.
Feature Content	Displays specific featured content from the web site that supplements the answers.
Hidden	HELP
Contact	For use with the Contact Deflection feature.

IML Basics

Define IML List Basic Operators

IML Color Code

The Dictionary object fields that accept IML input use a color code to aid in interpreting IML syntax. The color code applies different colors to the various types of IML elements to aid in identification. The color code may not be modified. The color code is as follows:

IML Element	Color Assignment
Infix	dark magenta
Prefix	blue
Parenthesis	red
Comma	gray
Enumeration	dark orange
Literal	dark green
Assignment	red
Keyword	dark cyan
Label	dark green
Concept	dark blue
Unknown	gold background
VIL expression	bright cyan background

Search Criteria Operators

You can use search criteria operators to match specified attributes of your definition search criteria. You can specify string-related, date-related, and domain-related operators, as described in the following tables:

String-Related Operators	Description
contains	Use this operator to locate items that contain the specified string.
exactly matches	Use this operator to locate items that exactly match the specified string.
starts with	Use this operator to locate items that begin with the specified string.
ends with	Use this operator to locate items that end with the specified string.
Date-related Operators	Description
on any date/none	Use this operator to locate items having any or no specified date.

exactly matches	Use this operator to locate items having dates that exactly match the specified date.
before this date	Use this operator to locate items having any date prior to the specified date.
after this date	Use this operator to locate items having any date later than the specified date.
List Operators	Description
any	Use this operator to locate items having or belonging to any of the parameters in the associated list.
is one of these	Use this operator to locate items having or belonging to any of the parameters selected from the associated list using the Set tool.

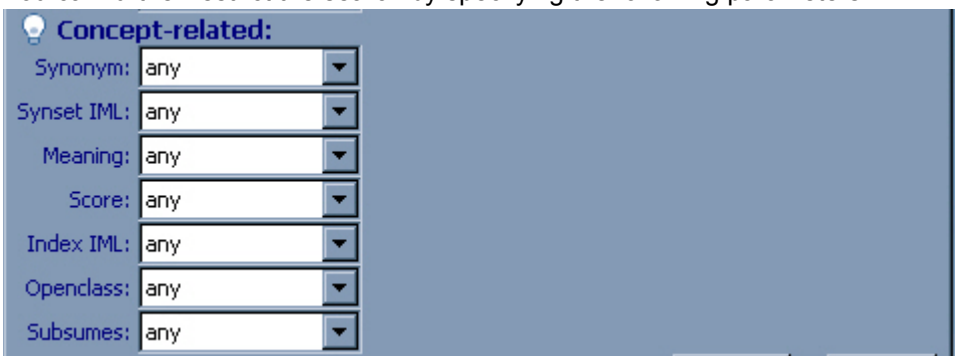
Specifying General Criteria

You can specify the following general search criteria:

Field	Description
Name	Use this field to search for objects by name. **Contains** is the default operator. You can use the following operators to match specific attributes of the object name: • contains • exactly matches • starts with • ends with
Notes	Use this field to search for objects by text within the Notes field. **Contains** is the default operator. You can use the following operators to match specific attributes of the note text: • contains • exactly matches • starts with • ends with
Author	Use this field to search for objects by the ID of the user who last edited the object. Any is the default operator. You can specify: • any • is one of these, using the Set tool to specify one or more user IDs
Last Edited	Use this field to search for objects by the date they were last edited. On any date/none is the default operator. You can use the following operators to match specified attributes of the date: • on any date/none • exactly matches • before this date • after this date
Restrict results to Active Domain List	Check this box to restrict results to the Active Domain List. If this box is not checked, Dictionary Objects from all Domain Lists display.

Specifying Concept-Specific Criteria

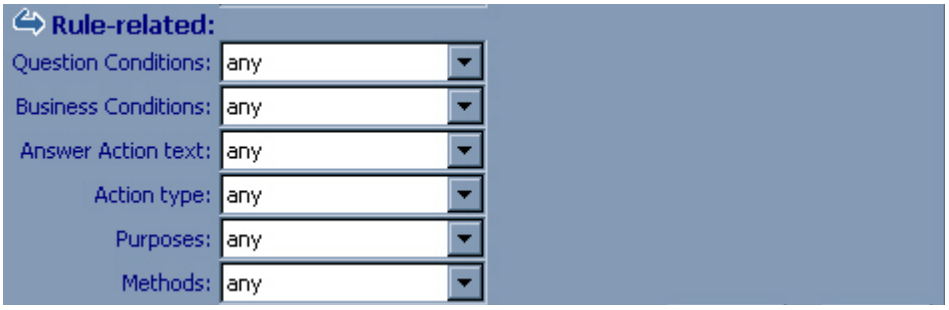
You can specify to search only for Concepts by selecting the Concepts radio button on the Dictionary Find Tool window. You can further restrict the search by specifying the following parameters:



Criteria	Operator
Synonym	Use this field to locate Concepts with synonyms that match the specified string criteria.
Synset IML	Use this field to locate Concepts with synonym sets that match the specified string criteria.
Meaning	Use this field to locate Concepts with Meaning data that match the specified string criteria.
Score	Use this field to locate Concepts at a specified level match the specified string criteria.
Index IML	Use this field to locate concepts with regards to the Index IML setting.
Openclass	Use this field to locate concepts with regards to the Openclass setting.
Subsumes	Use this field to locate concepts with regards to the Subsumes setting.

Specifying Rule-Specific Criteria

You can specify to search only for rules by selecting the Rules radio button on the Dictionary Find Tool window. You can further restrict the search by specifying the following parameters:

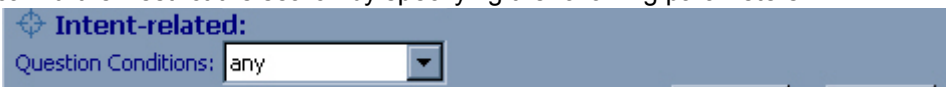


Criteria	Operator
Question Conditions	Use this field to locate rules with question patterns or example questions that contain the specified string.

Business Conditions	Use this field to locate rules with business conditions that contain the specified string.
Answer Action text	Use this field to locate rules with Answer Action text that contains the specified string.
Action type	Use this field to locate rules that contain the specified action type.
Purposes	Use this field to locate rules with assigned purposes that match the specified criteria.
Methods	Use this field to locate rules with assigned methods that match the specified criteria.

Specifying Intent or Intent Hierarchy-Specific Criteria

You can specify to search only for intents by selecting the Intent radio button on the Dictionary Find Tool window. You can further restrict the search by specifying the following parameters:



Criteria	Operator
Question Conditions	Locate rules with question patterns or example questions that contain the specified string.

Specifying Intent Response-Specific Criteria

You can specify to search only for Intent Response by selecting the Intent Response radio button on the Dictionary Find Tool window. You can further restrict the search by specifying the following parameters:

Criteria	Operator
Language	Locate intent responses in a specified language.
Template	Locate intent responses based on specified templates.
Purposes	Locate intent responses associated with specified answer purposes.
Profiles	Locate intent responses associated with a specified profile.

Dictionary Validation Tests

In addition to providing a count for how many of each dictionary object exists, the Dictionary Validation Tool checks each selected dictionary object for errors.

FOR ALIAS LISTS, THE DICTIONARY VALIDATION TOOL TESTS:

- The existence of specified root concepts.
- The existence of specified context variables.
- Whether a key has been duplicated.
- The existence of referenced concepts.
- Whether referenced concepts have been deleted.
- Whether a key concept is a descendent of a root concept.

FOR CONCEPTS, THE DICTIONARY VALIDATION TOOL TESTS:

- Whether a concept has associated synonyms.
- The existence of concept loops.

FOR DOMAIN LISTS, THE DICTIONARY VALIDATION TOOL TESTS:

- The existence of a domain.

FOR INTENTS, THE DICTIONARY VALIDATION TOOL TESTS:

- Whether an intent has valid IML.
- The existence of referenced variables.
- The existence of referenced variables within the target Domain List.
- The existence of referenced concepts.
- Whether referenced concepts have been deleted.
- The existence of referenced alias lists.
- Whether a referenced alias list is not in the target Domain List, but the object is.
- The existence of a concept referenced by VIL.
- The existence of a global/local variable referenced by VIL.
- Whether a parameter has a root concept.
- The existence of referenced root concepts.
- Whether a variable referenced by a parameter in an intent has been defined using IML.

FOR INTENT HIERARCHIES, THE DICTIONARY VALIDATION TOOL TESTS:

- The existence of a root node.
- The existence of a root node intent.
- The existence of referenced intents.

FOR INTENT RESPONSES, THE DICTIONARY VALIDATION TOOL TESTS:

- The existence of a template.
- Whether there is a mismatch between field types.
- The existence of a referenced alias list.
- Whether a referenced alias list is not in the target Domain List, but the response is.
- Whether a referenced alias list is a concept property.
- Whether a referenced alias list is a context property.
- Whether a required field is not set.
- Whether a parameter is not a type of root concept.
- Whether a response with set parameter values is assigned to one or more intent(s).
- The existence of a referenced process wizard.
- The existence of a referenced profile.
- The existence of a referenced intent.

FOR RESPONSE TEMPLATES, THE DICTIONARY VALIDATION TOOL TESTS:

- Whether there are no fields.
- Whether there is a field type mismatch
- Whether a required field is set in the response.

FOR RULES, THE DICTIONARY VALIDATION TOOL TESTS THE SYNTAX:

- Whether an answer action is incomplete.
- Whether the rule has valid IML.
- The existence of a referenced variable.
- The existence of referenced variables within the target Domain List.
- The existence of a referenced concept.
- Whether a referenced alias list is not in the target Domain List but the object is.
- The existence of a concept referenced by VIL.
- The existence of a variable referenced by VIL.
- The existence of a preference that is not defined or configured to be used.
- Whether an action points to an undefined purpose.
- Whether a database action has a SQL query.
- Whether the action has valid IML.
- The existence of a variable for a glossary action.
- The existence of a referenced plug-in.
- Whether an exact answer has both an answer and a title.
- Whether a URL action has a URL.
- The existence of a referenced rule.
- The existence of a referenced process wizard.

- Whether a search component has a variable.
- The existence of a search component variable.
- Whether a search component local variable is defined in a rule condition.
- Whether the search component has valid IML.

FOR PROCESS WIZARDS, THE DICTIONARY VALIDATION TOOL TESTS:

- Whether the process wizard has steps.
- Whether a step has no fields.
- Whether a step has two or more fields named the same.
- Whether a step has more than one search field.
- Whether a step references an unknown shared field.
- Whether a radio field has less than two options.
- Whether a search field sets the same variable two or more times.
- Whether a step has transactions from a search field.
- Whether a step has two or more transition conditions that are ambiguous (meaning they have no fields in common).
- Whether a step has two transitions that include the same condition.
- Whether the transition from one step to the next has the same conditions.
- Whether a transition from a step is missing a value in its condition.
- Whether a transition from a step has duplicate conditions.
- Whether a transition condition from a step references an unknown shared field.
- Whether a transition condition from a step references an unknown shared value.
- Whether a transition condition from a step references the same radio field more than once.
- Whether a transition condition from a step references an unknown field.
- Whether a transition condition from a step references an unknown step.

Language Analysis Labels

The Language Analyzer assigns various types of labels, which are distinct from the variable-related labels assigned by the Rules Engine. Labels assigned by the Language Analyzer are generally notated in all capital letters. The following label types are assigned:

- Part of speech and bracketing labels. These labels use the part of speech tags described in “Part of Speech Annotations” on page 152.
- Orthographic labels, which include PUNCTUATION, CAPITALISED, ALL_CAPS, and NUMBER.
- Synset labels, which include FIRST_CONCEPT, and CONCEPT.
- Scoring labels, which include BASE_LEVEL, INDUSTRY_LEVEL and CUSTOMER_LEVEL. Scoring labels correspond to the synset score assigned in Dictionary Manager.
- The ANAPHORA label, which is added for concepts added during anaphora resolution.

Part of Speech Annotations

NOTE: The Part of Speech Annotations are not configurable.

Annotation	Description
CC	Coordinating conjunction
CD	Cardinal number
DT	Determiner
EX	Existential there
FW	Foreign word
IN	Preposition or subordinating conjunction
JJ	Adjective
JJR	Adjective, comparative
JJS	Adjective, superlative
LS	List item marker
MD	Modal
NN	Noun, singular or mass
NNS	Noun, plural
NNP	Proper noun, singular
NNPS	Proper noun, plural
PDT	Predeterminer
POS	Possessive ending
PRP	Personal pronoun
PRP$	Possessive pronoun (prolog version PRP-S)
RB	Adverb
RBR	Adverb, comparative
RBS	Adverb, superlative
RP	Particle
SYM	Symbol
TO	to
UH	Interjection
VB	Verb, base form
VBD	Verb, past tense
VBG	Verb, gerund or present participle
VBN	Verb, past participle
VBP	Verb, non-third person singular present
VBZ	Verb, third person singular present
WDT	Wh-determiner
WP	Wh-pronoun

Annotation	Description (Continued)
WP$	Possessive wh-pronoun (prolog version WP-S)
WRB	Wh-adverb

Phrase Annotations

NOTE: The Phrase Annotations are not configurable.

Annotation	Description
ADJP	Adjective Phrase
ADVP	Adverb Phrase
CONJP	Conjunction Phrase
FRAG	Fragment
INTJ	Interjection. Corresponds approximately to the part-of-speech tag UH.
LST	List marker. Includes surrounding punctuation.
NAC	Not a Constituent; used to show the scope of certain prenominal modifiers within a noun phrase (NP).
NP	Noun Phrase
NX	Used within certain complex noun phrases to mark the head of the noun phrase.
PP	Prepositional Phrase
PRN	Parenthetical
PRT	Particle. Category for words that should be tagged RP.
QP	Quantifier Phrase (for example, a complex measure/amount phrase); used within noun phrases.
RRC	Reduced Relative Clause
UCP	Unlike Coordinated Phrase
VP	Verb Phrase
WHADJP	Wh-adjective Phrase. Adjectival phrase containing a wh-adverb, such as `how hot`.
WHAVP	Wh-adverb Phrase. Introduces a clause with a NP gap. May be null (containing the 0 complementizer) or lexical, containing a wh-adverb such as `how` or `why`.
WHNP	Wh-noun Phrase. Introduces a clause with a NP gap. May be null (containing the 0 complementizer) or lexical, containing some wh-word, such as `who`, `which book`, `whose daughter`, `none of which`, or `how many leopards`.