Oracle® Ultra Search

Release Notes

Release 2 (9.0.2)

May 2002

Part No. A97355-03

This document summarizes the differences between Oracle Ultra Search in Oracle9*i* Application Server Release 2 (9.0.2) and its documented functionality.

To view the Ultra Search documentation:

- View the documentation with a Web browser from the Oracle documentation CD.

- View the documentation within an Ultra Search installation at $ORACLE_HOME/ultrasearch/doc/help/toc.htm.

- View the documentation through the Oracle Ultra Search Administration Tool by clicking the Help icon.

> **See Also:**   *Oracle9i Application Server Release Notes*

1 Certification and System Requirements

See the Oracle Ultra Search section in the *Oracle9iAS Portal Configuration Guide* for systems requirements.

1.1 Ultra Search Installation

For installation documents within an Ultra Search installation, see $ORACLE_HOME/ultrasearch/doc/help/install.htm.

The Ultra Search middle tier is compliant with Oracle J2EE container (OC4J). Follow the instructions at $ORACLE_HOME/ultrasearch/doc/help/install_midtier.htm to configure the Ultra Search middle tier with OC4J.

Ultra Search requires you to have either JRE or JDK on the database host where you install the Ultra Search server component. By default, JDK 1.3.1 is installed by Oracle Universal Installer (OUI) during database installation under directory `$ORACLE_HOME/jdk`. If you use a different JDK, either create a soft link, or copy the files from the location where you install JDK to `$ORACLE_HOME/jdk`. Ultra Search is certified with JDK 1.3.1 in this release.

2 General Issues and Workarounds

2.1 Installation Issue

Bug 2304242 - Customer database silent mode install hanged at Ultra Search phase

For database installation with Oracle Portal Configuration Assistant (OPCA), Ultra Search only supports silent mode when there is no Ultra Search in the target database. For interactive mode, or if there is already an Ultra Search installed, the "DISPLAY" environment variable must be set correctly.

2.2 Migration Issue from 1.0.3 to 9.0.2

Bug 2310979 - Unable to migrate Ultra Search from 1.0.3 to 9.0.2

Ultra Search provides the SQL script 'wk0upgrade.sql' for migrating user data and database objects from an existing Ultra Search 9.0.1 database to a migrated Ultra Search 9.0.2 database. However, users get errors at the stage 'stop all of the crawler schedules and database jobs'. Also, the script might fail and stop during the stage 're-creating all of the user instances'. You can request for a patch to fix this problem.

2.3 New Features Addendum

In addition to new features covered in the Ultra Search online documentation, the followings should be noted:

- Robots exclusion

  When users create or edit a Web data source in the administration tool, they can specify whether to honor the robots exclusion when crawling the Web data source. If robots exclusion is honored, then the Web crawler traverses the pages based on the robots access policy specified in the Web server robots.txt file. If robots exclusion is ignored, then the Web crawler ignores the robots exclusion policy.

- Web data source model enhancement

There is no default Web data source in the 9.0.2 release. Users can create any number of Web data sources where each one is completely self-contained. In other words, its behavior is the same as any other type of data source

- Improved query performance

  A new Java API is provided for improved query performance. The method signature is public java.util.String compileForCount(). The complete method description is in section 6.1.1.

- Web data source path inclusion/exclusion

  Users can specify a URL path for crawler inclusion or exclusion. This lets the crawler crawl a subset of a particular host.

- Mimetypes as a default search attribute

  In addition to the author, description, subject, and title default search attributes, mimetype is added to let users restrict query results based on document type.

2.4 Deprecated and Desupported Features

The following features have been deprecated or desupported in Oracle Ultra Search release 9.0.2:

- 9.0.1 Query API Sample

  Deprecated.

- Running Ultra Search Middle-Tier Components on OSE (Oracle Servlet Engine)

  OSE has been desupported by Oracle. Use Oracle9*i* Application Server.

- Default Web data source and default schedule

  Desupported due to a Web data source model change. Use the migration script to migrate to Ultra Search 9.0.2. For more information about new Web data source model, see the Web data source model enhancement section.

3 Configuration Issues and Workarounds

This section describes Ultra Search configuration issues and their workarounds for Ultra Search.

3.1 Setting Up Ultra Search Sample Query Applications

In addition to the configuration steps described in `$ORACLE_HOME/ultrasearch/doc/help/install_midtier.htm`, you must also follow these steps in order for Ultra Search sample query application to function correctly.

To configure Ultra Search sample query applications and sample search portlet, edit the OC4J `data-sources.xml` file. For editing `data-sources.xml`, see `$ORACLE_HOME/ultrasearch/doc/help/install_midtier.htm`.

3.1.1 Setting Up 9.0.1 Query API Sample - Deprecated in 9.0.2

To configure `gsearch.jsp`, you must manually edit the file and change the variable setting for 'username' and 'password', and then edit `ultrasearch.properties` to change the connection string. The file `gsearch.jsp` is under the `$ORACLE_HOME/ultrasearch/sample/query/9i` directory.

For editing the `ultrasearch.properties`, see `$ORACLE_HOME/ultrasearch/doc/help/install_midtier.htm`.

For detailed information, see `$ORACLE_HOME/ultrasearch/sample/query/9i/README.html`

3.2 Running the Crawler

The Ultra Search Crawler is a Java process that runs on the server tier when launched. Therefore, Ultra Search requires you to have either JRE or JDK installed on the database host where you install the Ultra Search server component.

> **See Also:** "Certification and System Requirements"

3.2.1 Setting the JOB_QUEUE_PROCESSES Parameter

Oracle Ultra Search schedule launching uses the `DBMS_JOB` package. Therefore, the Oracle Ultra Search DBA must make sure that there is least one SNP process running. In other words, the initialization parameter file for the Oracle Ultra Search database instance should contain a line that specifies the `JOB_QUEUE_PROCESSES` parameter to be at least 2.

3.3 Globalization Support

Oracle Ultra Search, like any other application that uses the database to store content, requires that the character set of the database be able to support the character set used at the application level. For example, if the

application language is in Japanese, and if the database character set is SF7ASCII, then any data that the application attempts to store is corrupt, because the SF7ASCII character set does not support Asian languages.

The *Oracle9i Globalization Support Guide* provides detailed information on database character sets.

> **Note:** Universal character sets, such as UTF8, attempt to support all languages of the world, including, but not limited to, Asian, European, and Middle Eastern languages.

4 Performance Tuning

4.1 Verifying Crawler Progress

After you have configured the Oracle Ultra Search system, you can launch a crawler immediately from the Schedule screen.

View the Oracle Ultra Search crawler status by checking its state in the Oracle Ultra Search Administration Tool page.

Click the "Schedules" tab, and you see a table that lists all schedules and their state.

When a schedule is launching, it is in the "LAUNCHING" state. During the launching state, URLs to be crawled are marked by a SQL update operation. The amount of time this process may take depends on how many URLs are there for update. In cases such as maintenance crawling of the schedule, there can be millions of URLs to update, with the schedule staying in the "LAUNCHING" state for a very long time.

When a schedule has completed launching and the crawler has begun fetching pages, the state changes to "EXECUTING".

4.2 Performance Tuning

4.2.1 Crawler Performance Tuning

For Crawler performance tuning, see Step 5 in Configuring the Oracle Server for Ultra Search at `$ORACLE_HOME/ultrasearch/doc/help/configure_server.htm`.

4.2.2 Query Performance Tuning

This section contains suggestions on how to improve the response time of the Ultra Search query.

4.2.2.1 Tune the DB_CACHE_SIZE Parameter

The database buffer cache keeps frequently accessed data read from datafiles, and efficient usage of the buffer cache can improve Ultra Search query performance. The cache size is controlled by the DB_CACHE_SIZE initialization parameter.

For more information on how to tune this parameter, see *Oracle9i Database Tuning Performance Guide and Reference.*

4.2.2.2 Optimize the Index

Optimize the Ultra Search index after the crawler has made substantial updates. This can be done by scheduling index optimization on a regular basis. Make sure index optimization is scheduled during off-peak hours, because query performance is slow during an index optimization schedule.

For information on index optimization schedules, see the Ultra Search online documentation about the Schedules Page ($ORACLE_HOME/ultrasearch/doc/help/a_schedules.htm).

4.2.2.3 Optimize the Index Based on Tokens

Optimize the Ultra Search index by basing it on frequently searched tokens. Queries can be logged by turning on query statistics collection in the Administration tool. The frequently searched tokens then can be passed to CTX_DDL.OPTIMIZE_INDEX in token mode. The Ultra Search index name is WK$DOC_PATH_IDX.

For more information on OPTIMIZE_INDEX, see *Oracle Text Reference.*

4.2.2.4 Simplify Query Expansion

The search response time is directly influenced by the Text query string used. Although Ultra Search provides a default mechanism to expand user input into a Text query, simpler expansions can greatly reduce search time.

For information on customizing query expansion, see the Ultra Search online documentation about Customizing the Query Syntax Expansion ($ORACLE_HOME/qsyntax.htm) and the Javadoc for the oracle.ultrasearch.query.Query interface.

5 Known Bugs

5.1 Query Bugs

5.1.1 Bug 2114417 - Unwanted/Invalid Syntax Displayed For Query String

When query statistics collection is enabled, the query statistics pages (Daily summary of query statistics, Top 50 queries, Top 50 ineffective queries, Top 50 failed queries) may show text query strings like '(((WKA2X &({abc}))WITHIN S2))*2,({abc}))'. This behavior is due to 9.0.2 Java API expanding the user's query ('abc' in this case) before it is sent to the database.

In most cases, the user's query can be deciphered from the text query.

5.1.2 Bug 2097381 - Portal Users Should Embed Ultra Search Portlets Hosted On Same OC4J Instance

Portal users should embed Ultra Search portlets that are hosted on the same OC4J instance as the Oracle9*i*AS Portal server. For example, if the Oracle Portal OC4J instance is installed on host A / port 7777, then the Ultra Search provider must also be hosted as a Web application on host A / port 7777.

It is possible that the Ultra Search provider running on host A / port 777 could be registered with a second Oracle Portal instance running on a different host / port combination. In such cases, when the Ultra Search portlet is embedded within portal pages, the pop-up list-of-values will not work correctly. This is because of a security bug inherent in Javascript.

5.2 Admin Bugs

5.2.1 Bug 2304942 - Login Not Possible If User Did Not Log Out

If users log on to the Ultra Search administration pages and manually edit the URL to go to the login page again, then the administration login page will not let the user login again.

5.2.2 Bug 2177716 - No Document Found When Click On Link of The Search Result

When using the administration tool to do relevancy boosting or search, if the search result is not from Web data source, then when the user clicks the link of search result, the user will not see the content of the document.

5.2.3 Bug 2185138 - JSP Error Occur When Delete An Entry of URL Authentication Containing MBCS

An Ultra Search JSP error occurs when trying to delete an entry of URL authentication, if hostname or realm contain MBC.

5.3 Crawler Bugs

5.3.1 Bug 2166510 - NLS: Crawler Shutdown Unexpected If Log Directory Name Contains MBCS

The crawler is unable to handle log directory path with multibyte characters.

Avoid specifying log directories that have Chinese, Japanese, or Korean characters.

5.3.2 Bug 2166662 - NLS: Can't Find Files With MBCS Name

File data source crawling cannot crawl directories or files with multibyte characters; for example, Chinese or Japanese.

Avoid naming the file in Chinese, Japanese, or Korean or putting them under such directory.

5.3.3 Bug 2186745 - File Data Source Can Not Crawl Directory Or File Name With HTML Reserved Symbol

The crawler is unable to pick up files or directories whose name contains HTML reserved symbols, like '<' or '>', when doing file data source crawling.

Rename the file or directory that is using such symbol.

5.3.4 Bug 2265100 - Unable To Stop Crawling When Crawler Is Enqueuing URL From A Crawler Agent

Stop crawling does not stop the crawler, even though the schedule status shows that the crawler has stopped. This happens when the crawler agent is used and the crawler is in the process of enqueuing URLs fetched from the agent. The crawler stops only after enqueuing is finished.

Currently, there is no way to stop the enqueuing other than manually killing the crawler process.

5.4 Oracle Portal Dependency

5.4.1 Bug 2244234 - Translations Of Items Have Same Display URL In Portal XML

In Portal, the URLs for translations of an item have the same display URL as that of the base language item. Portal users can view different translations because when users log in to Portal, the language is established as part of the browser session. However, this language negotiation process only works with browsers operated by human users. Therefore, the Ultra Search crawler receives the same display URL for the translated items. This violates the stated requirement that all display URLs presented to Ultra Search be unique. The implication is that Ultra Search cannot crawl translations of an item.

As in bug 2218987, with multiple translations, only one of the translation items or the base item itself is indexed by the crawler. The rest are rejected by the crawler because of the duplicate display URLs.

5.4.2 Bug 2244239 - Some Attributes In Portal XML Are Not Encased In Translations Section

If there are translations for an item or page, then some attributes of that item/page cannot be correctly transmitted to the Ultra Search crawler. As a result, attribute queries may not work correctly for translated items.

5.4.3 Bug 2244254 - Portal XML Should Not Reveal Item Types Of BASETYPE=NONE

Ultra Search crawler can process specific Portal item types. However, Portal item type of "none" does not have display URLs. As a result, they are not revealed to Ultra Search. Because anything that does not have a display URL cannot be represented in the search application search results list in such a way that the user can click on it to view the item.

5.5 Oracle Text Dependency

5.5.1 Bug 2205449 - Oracle Core Dump During Text Indexing in Multibyte Character Set Database, Such as UTF8

This impacts Ultra Search crawling, because the Ultra Search crawler invokes Oracle Text for indexing during the crawling process. The patch is available in 9.0.1.3 ARU 1562387.

6 Documentation Errata

6.1 Oracle Ultra Search Online Documentation

The following are known issues with the Oracle Ultra Search online documentation.

6.1.1 Method compileForCount Not Documented in Ultra Search Java Query API Section

The Oracle Ultra Search online documentation includes a javadoc section for Ultra Search Java Query API. There is one method which is not documented in `oracle.ultrasearch.query.Query` interface.

The method signature is public `java.util.String compileForCount();` The function of this method is similar to the compile() method. It compiles the Query object into some equivalent Text query string. But the string returned by `compileForCount()` is passed to 'CTX_QUERY.COUNT_ HITS', not 'CTX_QUERY.CONTAINS'.

`compileForCount()` is needed because the Text query for `COUNT_HITS` can be much simpler than `CONTAINS` (because scoring is not a concern), and a simpler Text query can improve performance.

Classes implementing the Query interface must implement both the `compile()` and `compileForCount()` methods. A trivial, albeit slow, implementation of `compileForCount()` can return the same string as `compile()`.

6.1.2 Incorrect Jar File Name for Sample Crawler Agent

The "Create a Data Source Type" section in the "Sample Crawler Agent README" page of the Oracle Ultra Search online documentation contains an incorrect value for the sample agent jar file name. This causes you to get the following crawler error when you try to use the sample agent:

WKG-30116: Cannot find agent class "SampleAgent" from the java class path

The correct value is "sampleAgent.jar", not "sampleagent".

6.1.3 Incorrect Statement for Crawl Only Mode

The "Creating Synchronization Schedules" section in the "Schedules" page of the Oracle Ultra Search online documentation contains an incorrect note for examining URLs. The following statement should be removed.

Note: The option to examine URLs only applies to Web data sources. Table, file, and email data sources cannot be assigned to this schedule.

There is no such restriction for examine URLs feature. The option to examine URLs applies to all data sources.