
Rational software

IBM Rational ClearQuest 7.1.1
Full-Text Search Administrator's
Guide

Contents

Introduction.....	2
Getting started.....	2
Using cqtsadmin.pl.....	2
Globalizing cqtsadmin.pl.....	2
cqtsadmin-dbset-userdb.xml.....	3
Understanding cqtsadmin-dbset-userdb.xml.....	3
Examining cqtsadmin-dbset-userdb.xml configurable elements.....	4
cqtsadmin.pl.....	9
Command-line options: An overview.....	10
Required command-line options.....	14
Optional command-line options.....	17
Scenarios.....	38
I want to enable full-text search on an out-of-the-box SAMPL ClearQuest database to learn about its configuration and capability.....	38
I want to enable full-text search on user databases in a Windows production environment.....	41
I want to enable full-text search on user databases in a UNIX or Linux production environment.....	42
I want to change the display field for already-indexed record types.....	42
I want to enable full-text search on additional user databases.....	43
I want to add searchable indexed record types to my deployment.....	43
I want to remove indexed record types so that they are no longer searchable.....	44
I want to add nonindexed fields to indexed record types.....	44
I want to remove indexed fields from indexed record types.....	45
I want to enable full-text search on a user database utilizing load balanced CM Server.....	46
I want to set up a full-text search server at a MultiSite environment.....	46
I want to upgrade or reinstall ClearQuest, which has full-text search WAS profiles for multiple user databases.....	46
I want to enable full-text search on a non-English user database in a production environment.....	47

Introduction

Note that this tool and the command-line options described in this guide replace any and all tools that are documented in the "Configuring and maintaining full-text search" section of the ClearQuest information center. In this document you may be asked to reference the ClearQuest information center or the full-text search whitepaper on developerWorks for additional details, but you should not use any of the old tools unless you are instructed to do so.

IBM® Rational® ClearQuest® full-text search administrator is a tool that helps you set up, configure, and administer IBM Rational ClearQuest V7.1.1's full-text search feature. This guide discusses the command-line options, and provides a list of scenarios on how to administer this tool.

This tool replaces full-text search tools such as **cqtssetup.pl** and **cqtsdbcrawler.pl**, which come with ClearQuest 7.1.x. It hides the complexity and low-level details of setting up and configuring full-text search. It is assumed that you are familiar with the ClearQuest full-text search feature. Two resources that detail ClearQuest full-text search are:

information center	https://publib.boulder.ibm.com/infocenter/cqhelp/v7r1m0/index.jsp?topic=/com.ibm.rational.clearquest.webadmin.doc/topics/c_fts_admin_intro.htm
developerWorks®	http://www.ibm.com/developerworks/views/rational/libraryview.jsp?product_by=P%3ARational+ClearQuest&search_by=full-text+search

At a minimum, it is recommended that you read the first two parts of the article on developerWorks before reading this guide or using this tool.

You must have installed ClearQuest full-text search when installing ClearQuest 7.1.1 or later in order to use the full-text search administrator tool.

Getting started

The full-text search administrator tool consists of the following components:

- **cqtsadmin.pl** – A Perl script to set up and administer the full-text search.
- **cqtsadmin-dbset-userdb.xml** – An XML file generated by **cqtsadmin.pl** that provides and holds data about full-text search deployment.

Using *cqtsadmin.pl*

After you install the full-text search feature, issue the following command from your ClearQuest directory:

```
cqperl cqtsadmin.pl arg1 ... argN
```

To get help, type:

```
cqperl cqtsadmin.pl --help
```

The complete syntax and available command-line options are documented in this guide.

Globalizing *cqtsadmin.pl*

To use **cqtsadmin.pl** with a non-English operating system, set the LANGUAGE system environment variable to one of the following values:

LANGUAGE value	Use for language
en	English (United States) (the default)

de	German (Germany)
fr	French (France)
it	Italian (Italy)
br	Portuguese (Brazil)
es	Spanish (Spain)
cn	Chinese-simplified (China)
hk	Chinese-traditional (Hong Kong S.A.R. of China)
tw	Chinese-traditional (Taiwan) (Same as “hk”)
ja	Japanese (Japan)
ko	Korean (Korea, South)

English is the default. If an unsupported value is used, **cqtsadmin.pl** fails with an error message (in English).

The **cqtsadmin.pl** tool uses this environment variable to determine what language to use. This variable is typically set on UNIX® but not on Microsoft® Windows®. If you don't see output in your native language, it means you need to set this environment variable before you use this tool.

cqtsadmin-dbset-userdb.xml

`cqtsadmin-dbset-userdb.xml` is a full-text search administrator configuration file that holds data about your deployment. This file, used by **cqtsadmin.pl**, is automatically generated based on your operating system and ClearQuest user database. A separate configuration file is generated and maintained for each user database that has full-text search enabled. Although you can customize this file, the default values typically do not need to be modified.

Understanding cqtsadmin-dbset-userdb.xml

The full-text search administrator configuration file holds various default values for your deployment. Some are read only, but most are modifiable to further tune your deployment.

The configuration file is made of XML elements. Each XML element represents a setting in which the following XML tags provide additional information about an element:

Tag Name	Description
<description>	Explains the purpose of this configuration element.
<note>	Provides additional noteworthy information about this tag.
<defaultValue>	Holds the default value for this XML tag, which is used if you do not provide a replacement value via the XML tag <newValue>.
<newValue required="yes">	Holds any overriding value (over the default value of the XML tag <defaultValue>) that you may want to specify. When a value in <defaultValue> is not suitable, a value specific for your deployment in <newValue> is set based on parameters you specify on the command-line option or gathered from your ClearQuest database or your operating system. Required configuration elements are flagged with the XML attribute <newValue required="yes">

Examining cqtsadmin-dbset-userdb.xml configurable elements

The `cqtsadmin-dbset-userdb.xml` file contains brief documentation for each configuration element through the XML tags `<description>` and `<note>`. The following table provides complete documentation for each configuration element.

Tag Name	Description	Notes
<code><dbSet></code>	The ClearQuest database set name.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>The value for this tag is based on the ClearQuest database set specified in the command-line argument to cqtsadmin.pl at initial setup.</p>
<code><userDB></code>	The ClearQuest user database name.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>The value for this tag is based on the ClearQuest user database name specified in the command-line argument to cqtsadmin.pl at initial setup.</p>
<code><siteName></code>	The ClearQuest site name.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>The value for this tag is based on your ClearQuest database type. If your database is not replicated, the value is &lt;local>; (that is, <code><local></code>). Otherwise it is the name of your master replica.</p>
<code><enableAtSites></code>	Comma-separated list of sites that enable ClearQuest full-text search.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>The value for this tag is based on your ClearQuest database type. If your database is not replicated, this value is ignored. If it is replicated, the default value is empty, which means to enable ClearQuest full-text search at all sites.</p> <p>To enable ClearQuest full-text search at selected sites, list the site names in this XML tag using comma as a separator.</p>
<code><entityFile></code>	The ClearQuest full-text search entity file.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>The value for this tag holds the path to the automatically generated entity file for your ClearQuest full-text search deployment. The default file is <code>Entity-dbset-userdb.txt</code>, which is a text file that lists the record types and their fields to be searched.</p> <p>To limit which record types or fields to search, remove record types or fields that you do not want to be searchable.</p>
<code><ftsWASProfileName></code>	The ClearQuest full-text search WebSphere® Application Server profile name.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>If you are deploying ClearQuest full-text search for two or more ClearQuest user databases, use this tool to create a separate WebSphere Application Server</p>



		<p>profile for each deployment. The value for this tag holds the profile name.</p> <p>Profile naming conventions are cqsearchprofile_db-set-name_user-db-name, where <i>db-set-name</i> and <i>user-db-name</i> are determined from the command-line parameters passed to cqtsadmin.pl.</p> <p>If the resulting profile name exceeds 27 characters, then cqsearchprofile is abbreviated as cqfts.</p> <p>If the profile name still exceeds 27 characters, shorten <i>db-set-name</i> or <i>user-db-name</i>. If this is not possible, provide your own profile name by replacing the value in this XML tag with a profile name that is unique on the computer that you are deploying full-text search (for example, <i>MyProfile_1</i>).</p>
<ftsWASProfilePort>	The ClearQuest full-text search WAS profile port number.	<p>The default value for this tag is auto-generated by cqtsadmin.pl tool. You do not need to modify it.</p> <p>The default value is 14080 for the initial cqsearchprofile, but it is overridden to an asterisk (*), which instructs cqtsadmin.pl to determine the value based on the WAS profile name.</p> <p>If you are creating a new WAS profile, the value for this tag is auto-generated to be unique.</p>
<ftsWASProfileMaxHeapSize>	The maximum JVM heap size for the WAS profile in megabytes (MB).	<p>The default value for this tag is empty, which means that the JVM maximum heap size (300 MB) is used.</p> <p>If, during indexing, particularly batch indexing, you experience “out of memory” errors, increasing the heap size should eliminate this issue. The maximum value, on a 32-bit operating system, is 2048 MB.</p>
<ftsServerName>	The host or server name on which this ClearQuest full-text search deployment is being set.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>The value for this tag is set to the server name on which you are deploying full-text search. Change this name only if you rename your computer or move your deployment to a server that has a different name.</p> <p>You must supply a fully qualified hostname if you are doing load balancing or have a shared “cqsearchprofile.” ClearQuest replicated database configuration.</p>
<enableCQFTS>	Set to “true” to enable the Full Text radio button in ClearQuest Web.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>This tag has two values: “true” and anything else (or empty). When set to “true,” the Full Text radio button in ClearQuest Web is enabled during setup. Otherwise this step is skipped.</p> <p>Do not change the value for this tag unless you need finer control over the setup or are trying to resolve an issue.</p>



		<p>WARNING: Do not set this value to “false” (disable the radio button) if the user database is fully or partially indexed but is not replicated. Disabling the radio button disables oplogging for nonreplicated ClearQuest databases. Therefore, full-text search does not track changes to ClearQuest records, and changed records are never reindexed. If you set the value to “false,” and records are changed, you have to completely reindex the database.</p> <p>If it becomes necessary to set the value to “false,” you must prevent ClearQuest record changes until the value is reset to “true” (the Full Text radio button is reenabled).</p>
<runBatchIndex>	Set to “true” to allow batch indexing to start during ClearQuest full-text search setup.	<p>The default value for this tag is auto-generated by cqtsadmin.pl tool. You do not need to modify it.</p> <p>This tag has two values: “true” and anything else (or empty). When it is set to “true,” the batch-mode indexer runs during setup; otherwise this step is skipped.</p> <p>Skipping the batch indexing means that all existing records are not indexed and thus are not searchable. Therefore, your search result will be incomplete.</p>
<startUpdateIndex>	Set to “true” to allow update indexing to start during ClearQuest full-text search setup.	<p>The default value for this tag is auto-generated by cqtsadmin.pl tool. You do not need to modify it.</p> <p>This tag has two values: “true” and anything else (or empty). When it is set to “true,” the update-mode indexer is configured to load on WAS profile startup, otherwise this step is skipped.</p> <p>Skipping the update indexing means that all modified or newly added records are not indexed after the initial batch indexing, and thus are not searchable. Therefore, your search result will be incomplete.</p>
<batchIndexJVMParm>	The JVM parameter setting used by the batch-mode indexer.	<p>The default value for this tag is empty, which means that no JVM parameters are passed to the batch-mode indexer.</p> <p>If you experience “out of memory” errors during batch indexing, increasing the heap size through this value should eliminate this issue. The maximum value, on a 32-bit operating system, is 2048 MB.</p> <p>Because this JVM parameter value is passed as-is to the batch-mode indexer, use correct values per the JVM specification. For example, to increase the JVM heap size to 1.5 GB, use: <code>-Xmx1536m</code>. To increase memory and enable JVM debugging, use: <code>-Xmx1536m -XX:-PrintGCDetails</code>.</p>
<batchIndexBatchSize>	The number of records to index per thread during	The default value for this tag is auto-generated by the cqtsadmin.pl tool. You might want to modify it to speed up batch indexing (increase number of records

	<p>batch indexing. The value is an integer between 1 and 1000.</p> <p>The default value is 250.</p>	<p>to read per thread) or to overcome “out of memory” errors by decreasing the number of records to read per thread).</p> <p>The default value is 250. If your environment has sufficient CPU cycles and RAM resources, and you want to speed up batch indexing, consider increasing the value to 500. This reduces the time it takes to batch index ClearQuest records. However, you run the risk of running into “out of memory” errors. To fix these errors, increase the JVM memory for the batch indexer or the WAS profile.</p>
<batchIndexDelay>	<p>The delay value in seconds for the batch indexer. The value is an integer between 0 and 86400 (1 day).</p> <p>The default value is 30.</p>	<p>The default value for this tag is auto-generated by cqtsadmin.pl tool. You might want to modify it to speed up or slow down batch indexing based on CPU load and performance maintaining resources for other applications on the server.</p> <p>Each time the batch indexer processes the quantity of records specified by <batchIndexBatchSize>, it stops for the duration set in this tag. This allows other processes to utilize the CPU. The default is set to 30 seconds.</p> <p>You can set it to 0, which means that there is no delay. However, doing so might lead to aggressive CPU utilization during batch indexing, both on the ClearQuest full-text search server and the ClearQuest database server, so that other applications do not get their fair share of the CPU.</p>
<batchIndexThreads>	<p>The number of threads to create for batch indexing. The value is an integer between 1 and 10.</p> <p>The default value is 5.</p>	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You might want to modify it to speed up batch indexing (by increasing the value) or slow it down (by decreasing the value) so the CPU is released for other applications, or to relax demand on the database server.</p> <p>When batch indexing, you can specify how many threads are created, where each thread processes its own <batchIndexBatchSize> quantity of records.</p> <p>If you have enough CPU and RAM resources, you can speed up the batch-mode indexer by increasing the number of threads. However, you run the risk of running into “out of memory” errors. If so, increase the JVM memory for the batch or WAS profile (or both profiles), or configure a smaller number of <batchIndexBatchSize> records to be read per thread.</p>
<updateIndexBatchSize>	<p>The number of records to index during a single pass of the update indexer. The value is an integer between 1 and 1000.</p>	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it.</p> <p>The default is 250, and you might need to increase it if the update-mode indexer cannot keep up with the amount of changes made to the ClearQuest database.</p> <p>Note: Unlike the batch-mode indexer, the update-mode indexer is single threaded. Higher size values</p>



	The default value is 250.	per pass, or shorter <updateIndexDelay> settings might be needed to achieve desired update-mode indexing throughput. This depends on average record submissions, and updates and deletions per minute per ClearQuest user database.
<updateIndexDelay>	The delay value in seconds for the update indexer. The value is an integer between 1 and 86400 (1 day). The default value is 600.	The default value for this tag is auto-generated by the cqtsadmin.pl tool. You do not need to modify it. The default is 600 seconds (10 minutes). If you decrease this value to 60 (1 minute), then your search index synchronizes with your ClearQuest database modifications every 1 minute at the cost of more frequent logins (and a temporary additional license consumption), and CPU cycles.
<updateIndexLoginInterval>	When to issue a re-login after a check for changes in the ClearQuest database. The value in seconds is an integer between 1 and 86400 (1 day). The default value is 1.	The default value for this tag is auto-generated by the cqtsadmin.pl tool. To achieve faster update-mode indexing throughput, you might need to modify this value. The most expensive part of update-mode indexing is typically login operation. Use <updateIndexLoginInterval> with <updateIndexDelay> to reduce this cost. For example, to set ClearQuest full-text search to check for new changes every minute but re-login every 10 minutes, increase <updateIndexLoginInterval> to 600 (10 minutes) and reduce <updateIndexDelay> to 60 (1 minute). Now, the full-text search index is updated each minute with ClearQuest record changes, but a login occurs only once every 10 minutes. In effect, the CPU is taxed less often, but at the cost of keeping a license checked out longer. It is suggested that login occurs at least several times daily.
<addRecordType>	A list of record types and fields to be added to the full-text search deployment.	The default value for this tag is empty. You use it when adding new record types or fields to your search index if your ClearQuest schema changes after initial configuration. If you omit a record type during deployment, or you add a record type and want to be able to search on it, you can do so by providing the full record type and associated fields via this tag. For example, use the following format to add the record type <i>Customer</i> and its set of fields: <i>Customer=field1,field2,fieldN</i> To add multiple record types, use a semicolon (;) as a separator, for example: <i>Customer=field1,field2,fieldN;NewRecord=field1,field2,fieldN</i> This option cannot be used to add additional fields to an existing indexed record type. To do so, first remove the record type, then re-add it with this

		<p>option (requires full reindexing of this record type).</p> <p>IBM Rational Client Support can assist you if you wish to add new fields to an existing record type and you do not wish to re-index the entire record type, potentially forgoing values for this field on existing unmodified records.</p>
<removeRecordType>	A list of record types to be removed from the full-text search deployment.	<p>The default value for this tag is empty. Use it to remove indexed record types from your deployment.</p> <p>Enter the names of record types in this XML tag for removal. To remove the record type “Customer”, enter its name in this XML tag. To remove multiple record types, use a semicolon (;) as a separator, for example:</p> <p>Customer;NewRecord</p>
<deleteFtsWASProfiles>	Set to “true” if you want the command-line option prep_upgd_was_profiles to also delete your ClearQuest full-text search WAS profiles.	<p>The default value for this tag is auto-generated by the cqtsadmin.pl tool and is set to “false”. When you upgrade or reinstall ClearQuest, change it to “true” before you run the command-line option prep_upgd_was_profiles.</p> <p>There are only two valid values for this XML tag, “true” and anything else (or empty). When it is set to “true” and you specify the command-line option prep_upgd_was_profiles, in addition to creating backup data of your ClearQuest full-text search WAS profiles, it deletes these WAS profiles.</p> <p>You must delete the profiles when you upgrade or reinstall ClearQuest. Otherwise the installer might fail and you might lose the profile settings. After you upgrade or reinstall ClearQuest, use the command-line option restore_was_profiles to re-create the profiles that were deleted.</p> <p>After completing the command-line option prep_upgd_was_profiles, make sure to reset the value of this XML tag back to “false” to prevent accidental deletion.</p> <p>When the value of this XML tag is set to “false,” issue the command-line option prep_upgd_was_profiles to create a backup of your ClearQuest full-text search profiles.</p>

cqtsadmin.pl

The Perl script **cqtsadmin.pl** is a tool designed to automate the steps required to set up, configure, and administer ClearQuest full-text search. It does this through command-line options that you pass to the tool and data that you provide to it through the full-text search administrator configuration file **cqtsadmin-dbset-userdb.xml**.

The following command-line options are not available on UNIX or Linux:

```
enable_cqweb_fts
disable_cqweb_fts
```

scrub_oplog

These command-line options require the ClearQuest tools **installutil.exe** and **multiutil.exe**, which are not available on UNIX or Linux. When you run these options, you are instructed to follow steps on a Windows computer to complete your operation. Before you deploy the ClearQuest full-text search on UNIX or Linux, verify that:

- IBM Rational ClearQuest 7.1.1 full-text search administrator tool is available on Windows (where ClearQuest full-text search is also installed as part of ClearQuest installation)
- Your Windows computer has a connection profile identical to that on your UNIX or Linux computer.

Command-line options: An overview

The command-line options can be divided into two groups: a required list, which you must provide every time you run **cqtsadmin.pl**, and an optional list, from which you must provide at least one option.

Required command-line options

Command Name	Parameter	Summary
username	<i>user-name</i>	ClearQuest user name with “super-user” privileges.
password	<i>password</i>	ClearQuest user password.
dbset	<i>db-set</i>	ClearQuest database set name.
userdb	<i>user-db</i>	ClearQuest user database name.
ftshome	<i>cq-fts-home</i>	ClearQuest full-text search home directory. Contains all configuration files related to this deployment, as well as Solr files, settings, and the Lucene index.

Optional command-line options

The optional command-line options can be grouped into two categories: commonly used and rarely used options.

Commonly used command-line options:

Command Name	Parameter	Summary
add_record_type		Adds one or more new record types to the index.
backup_fts	<i>destination</i>	Creates a backup copy of your ClearQuest full-text search deployment. Will temporarily disable update-mode indexer while the backup is going on.
prep_upgd_was_profiles	<i>destination</i>	<p>Creates backup data of all your ClearQuest full-text search WAS profiles and optionally deletes (un-deploys) all ClearQuest full-text search WAS profiles (except cqsearchprofiles). The backup data is used with the command-line option restore_was_profiles to restore these deleted WAS profiles.</p> <p>Use these command-line options when you are upgrading or reinstalling ClearQuest. They preserve your ClearQuest full-text search WAS profiles that were created with cqtsadmin.pl and restore them after your upgrade or reinstallation completes.</p>



		<p>This operation is required when you are upgrading or reinstalling ClearQuest, especially if you have created your own ClearQuest full-text search WAS profiles. If you skip this operation, the upgrade or reinstallation of ClearQuest might fail.</p> <p>To actually delete ClearQuest full-text search WAS profiles, you must change the XML tag value for <deleteFTSWASProfiles> from “false” to “true.” If you do not, only a backup of your ClearQuest full-text search WAS profiles is created.</p>
create_fts_was_profile	manual automatic disabled	<p>Creates a ClearQuest full-text search WAS profile.</p> <p>The parameter value is to configure Windows service status for Windows operating system deployments only. This value is ignored on the UNIX system and Linux, for which you must configure the WAS profile to start at boot time as daemon.</p>
delete_fts_was_profile		<p>Deletes the ClearQuest full-text search WAS profile associated with this ClearQuest user database.</p>
disable_cqweb_fts		<p>Disables the Full Text radio button in the ClearQuest Web GUI as well as oplogging in a nonreplicated ClearQuest user database.</p> <p>Be careful when disabling oplogging especially if your ClearQuest database is not replicated. Because oplogs are not generated, records will not be indexed.</p> <p>This option is not available on UNIX or Linux. If you need to run this option or if it was run by another option, you are given instructions to follow on a Windows computer.</p>
enable_cqweb_fts		<p>Enables the Full Text radio button in the ClearQuest Web GUI.</p> <p>If your deployment is not properly configured, users who attempt to use full-text search receive an error message.</p> <p>This option is not available on UNIX or Linux. If you need to run this option or if it was run by another option, you are given instructions to follow on a Windows computer.</p>
gather_diagnostic_data		<p>Collects data to help IBM Rational ClearQuest Client Support diagnose potential ClearQuest full-text search issues.</p>
help		<p>Displays help text and then exits.</p>
init_cq_fts		<p>Copies the ClearQuest full-text search default template and generates the default entity file. In effect, it runs both copy_fts_template and gen_entity_file.</p>
lock_cq_fts		<p>Locks this ClearQuest full-text search deployment so that only nondestructive cqtsadmin.pl commands are executable. All commands, except gather_diagnostic_data and help, are disabled.</p>



		Use the unlock_cq_fts option to reenable the commands.
optimize_idx		Defragments the index. Requires a minimum of 2 ½ times free disk space as your current index size for a successful optimization.
remove_record_type		Removes record types.
restore_was_profiles	<i>directory</i>	Restores your ClearQuest full-text search WAS profiles from backup data created with the command-line option prep_upgd_was_profiles . Use these options when you are upgrading or reinstalling ClearQuest. They preserve your ClearQuest full-text search WAS profiles that were created with cqtsadmin.pl and were deleted with the command-line option prep_upgd_was_profiles .
setup_cq_fts		Performs end-to-end ClearQuest full-text search setup based on your record types and fields setting of your entity file, and the setting specified in your full-text search administrator configuration file.
create	<i>before-date</i>	Scrubs all oplogs created before the given date. If your ClearQuest database is replicated, this command does not scrub oplogs. Instead, it reports an error and instructs you to use your replication oplog scrubbing policy and tool. If your ClearQuest database is not replicated, aggressive oplog scrubbing before the update-mode indexer processes them can lead to incomplete record indexing. This results in index inaccuracy and missed hits. Either ensure update record indexing is up-to-date, scrub only older oplogs (more than 1 month old), or skip oplog scrubbing, as it is not critical. This option is not available on UNIX or Linux. If you need to run this option or if it was run by another option, you are given instructions to follow on a Windows computer.
show_scenarios	[ID all]	Displays a list of scenarios, with examples, of how to use the cqtsadmin.pl tool. If no parameter is used, then a summary of all scenarios with their IDs and headlines is displayed. If an ID is specified, then the full text of that scenario is displayed. If the parameter all is used, then the complete list of scenarios with IDs, headlines, and full text is displayed.
unlock_cq_fts		Unlocks this ClearQuest full-text search deployment so that all cqtsadmin.pl commands are executable.
update_fts_prop_files		Updates all generated ClearQuest full-text search files. When you make changes in your full-text search administrator configuration file, such as the batch size,



or number of threads, update-mode index intervals, those changes are not reflected in the full-text search Properties XML file until you run this command.

Rarely used command-line options:

Command Name	Summary
archive_fts	Archives a ClearQuest full-text search deployment. It disables the active full-text-search deployment but preserves all configuration data and the index.
clear_state	Resets the state in the cqtsadmin.pl tool procedure so that there is no longer a state. In effect, whatever state the tool was in, which might have been an incomplete state, is now gone.
copy_fts_template	Copies and sets the ClearQuest full-text search default template.
cust_solr_files	Customizes your Solr schema file based on your current ClearQuest entity file.
cust_fts_files	Customizes the ClearQuest full-text search Properties XML file based on your user database, entity file, and full-text search administrator configuration file.
fresh_batch_idx	Forces a full reindex in batch mode. Full-text search hits might be inaccurate until batch reindexing is completed.
fresh_update_idx	Forces a full reindex of all oplogged record changes by the update-mode indexer. Use this option with caution. Because the update mode indexer is single threaded, this operation takes a long time, especially if your ClearQuest database contains many oplogs.
gen_entity_file	Generates the entity file, which contains all submittable record types in the ClearQuest database. These record types and associated fields are candidates for full-text search. You can customize the file later to select only those record types and associated fields that you want to index.
gen_fts_files	Generates ClearQuest full-text search setup files based on your user database, entity file, and full-text search administrator configuration file.
run_batch_idx	Executes the batch-mode indexer. If the batch-mode indexer has been stopped, it continues from where it last left off. Once a record type has been indexed, the indexer continues with the next record type even if indexing resumes after new records have been submitted against an already indexed record type.
remove_lucene_idx_lock	Removes the Lucene index lock (if any). Used to troubleshoot indexing errors.
repair_records	Reindexes failed records logged in the ftshome directory.
set_was_max_mem	Sets JVM's MAX memory for the WAS profile based on your setting in the full-text search administrator configuration file.
set_solr_home	Sets the Solr home directory under WAS to your ftshome deployment directory.

start_fts_was_profile	Starts the associated ClearQuest search profile under WAS for this ClearQuest user database.
start_update_idx	Enables and then starts the update-mode indexer under WAS for this ClearQuest user database. The WAS profile is stopped and then started.
stop_fts_was_profile	Stops the ClearQuest search profile under WAS for this ClearQuest user database.
stop_update_idx	Disables and stops the update-mode indexer under WAS for this ClearQuest user database. The WAS profile is stopped and then started

Both the required and optional command-line options are run in the sequence provided and follow these rules:

- 1) The required command-line options can appear only once. For example, **username** cannot appear more than once.
- 2) The optional command-line options run in the order they appear on the command line. That is, if you type the following:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --unlock_cq_fts
--optimize_idx --lock_cq_fts
```

then **unlock_cq_fts** is run first, followed by **optimize_idx**, and finally **lock_cq_fts**.

- 3) An optional command-line option can appear multiple times. When multiple optional command-line options are given, they run in the order that they appear on the command line. If an option fails, subsequent command evaluation and execution ceases, and the tool exits with an error message.
- 4) Some command-line options require arguments. The arguments must be valid before any action is taken, and for any command evaluation and execution to be processed.

Required command-line options

The required command-line options are always necessary in order to use **cqtsadmin.pl**. If any of the parameters to these options are invalid, the tool fails with an error message. The tool authenticates the user against the ClearQuest database before any action is taken.

--username *super-user-name*, **--password** *pass-word*, **--dbset** *db-set*, and **--userdb** *connection-name*

The **username**, **password**, **dbset**, and **userdb** command-line options are required.

Example: **cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --optimize_idx**

This example assumes that you have deployed ClearQuest full-text search for the database set **TextSearch** and the user database **SAMPL** on the D drive in the directory **CQ.Search**. The optional command-line option is **optimize_idx**, which invokes the index optimization command.

--ftshome *cqfts-home*

This command-line option specifies the location of your ClearQuest full-text search deployment.

Example: **cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --init_cq_fts**

This example creates a new deployment of ClearQuest full-text search on the D drive in the directory **CQ.Search**. The directory is created if it does not exist. In this directory, the subdirectory **TextSearch_SAMPL** is created. The subdirectory name is generated based on your ClearQuest database set name and your ClearQuest logical user database name, and contains full-text search data for this

deployment.

To enable another ClearQuest user database for use with ClearQuest full-text search, specify the mandatory command-line options pertaining to that alternate ClearQuest user database when using the **cqtsadmin.pl** tool. Additional ClearQuest user databases will deploy full-text search configurations based on the specified **ftshome** arguments, as well as the generated subdirectory structure.

Example: `cqperl cqtsadmin.pl --username admin --password "" --dbset Marvel --userdb XMEN --ftshome D:\CQ.Search --init_cq_fts`

This example deploys ClearQuest full-text search for a second ClearQuest database. It is on the D drive in the directory `CQ.Search`. A subdirectory named `Marvel_XME` is created. Thus, it is independent of any other deployment.

Note: The **ftshome** command-line option can be in any path. However placing all configurations in one directory helps you find and reference them easily. For optimal performance, it is recommended that the full-text search **ftshome** directory be on a non-system disk, and preferably run at 7200 RM or faster if you are deploying full-text search for several ClearQuest user databases on the same server.

Understanding **ftshome**

When you deploy full-text search or reference a deployment, the required command-line options **ftshome**, **dbset**, and **userdb** play an important role. Together, they define where your deployment data resides. For example, when you issue this command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --init_cq_fts
```

The following checks are done:

1. Does the directory `D:\CQ.Search` exist? If not, create it.
2. Does the directory `D:\CQ.Search\TextSearch_SAMPL` exist? If not, create it.

Once `D:\CQ.Search\TextSearch_SAMPL` is created, it is the “home” of the full-text search deployment for the database set `TextSearch` and the user database `SAMPL`. All data and settings related to this deployment are made in this directory. Also, any subsequent commands that you issue (that use the same value for **ftshome**, **dbset**, **userdb**) are applied to this directory.

If you are deploying full-text search for a second ClearQuest user database, using the same **ftshome** value is recommended. For example:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb PROD --ftshome D:\CQ.Search --init_cq_fts
```

This deploys full-text search for the database set `Marvel`, and user database `XMEN` in the directory `D:\CQ.Search\TextSearch_PROD`. In short, your full-text search deployment location is constructed as follows:

```
ftshome\dbset_userdb
```

Note: The values that you supply for **ftshome**, **dbset**, and **userdb** are case-sensitive within the ClearQuest full-text search administrator tool. This is also true for Windows even though Windows directory names and fields are not case sensitive. If you use a different case, the command fails.

The directory structure of your full-text search deployment is as follows:

Directory or File	Summary
<code>ftshome\dbset_userdb\logs\</code>	This directory holds logs of every command that you issue against this deployment. You might want to reference this directory, to get a history of which commands you have used,

	<p>when you used them, and their status. In addition, IBM Rational Client Support might examine these logs when working with you on an issue.</p> <p>Passwords appear as asterisks (*) in the logs and screen output, and are not captured in plain text.</p>
<code>ftshome\dbset_userdb\Solr\solr\conf\schema.xml</code>	<p>This file is one of the configuration files that the search engine uses to determine what fields to index and search on.</p> <p>When the command cust_solr_files is issued, the fields in this file are customized to match the fields in your record types specified by your entity file.</p> <p>You do not need to edit this file, but IBM Rational Client Support might examine it when working with you on an issue.</p>
<code>ftshome\dbset_userdb\Solr\solr\data\index\</code>	<p>This directory holds the actual index of your deployment.</p> <p>Do not modify the contents of this directory. Doing so can compromise the integrity of your deployment, and might require reindexing or redeployment.</p>
<code>ftshome\dbset_userdb\AboutThisFTS.txt</code>	<p>This file is generated once during your initial deployment of full-text search. It holds information about this deployment that you may need to reference.</p> <p>IBM Rational Client Support might examine this file when working with you on an issue.</p>
<code>ftshome\dbset_userdb\CQ-dbset-userdb.xml</code>	<p>This is your ClearQuest full-text search properties XML file. It holds data about batch and update indexing, the search server, your connection profile, and record types, as well as fields to search on.</p> <p>You need to modify this file to specify which field to set as the display field for each record type that you are indexing. No other changes are required unless you are instructed by IBM Rational Client Support. Also, IBM Rational Client Support examines this file when working with you on an issue.</p>
<code>ftshome\dbset_userdb\cqtsadmin-dbset-userdb.xml</code>	<p>This is your full-text search administrator configuration file. It holds data about your deployment. Most of the data is set during deployment, but you may need to edit this file to customize some settings.</p> <p>IBM Rational Client Support examines this file when working with you on an issue.</p>
<code>ftshome\dbset_userdb\Entity-dbset-userdb.txt</code>	<p>This is your entity file, which holds a list of entity types and their fields for which searching is enabled. During your deployment, you might have</p>

	<p>to edit this file to remove any record types or fields that you do not wish to search on. After completing deployment, you should not modify this file.</p> <p>IBM Rational Client Support examines this file when working with you on an issue.</p>
<code>ftshome\dbset_userdb\Solr-dbset-userdb.xml</code>	<p>This is your intermediate Solr file, which holds a list of fields for which searching is enabled.</p> <p>You do not need to modify this file, but IBM Rational Client Support examines it when working with you on an issue.</p>
<code>ftshome\dbset_userdb\batch_recovery.properties</code>	<p>This file contains information about the last DBID of the last record type indexed by the batch-mode indexer.</p> <p>You do not need to modify this file, but IBM Rational Client Support examines it when working with you on an issue.</p>
<code>ftshome\dbset_userdb\update_recovery.properties</code>	<p>This file contains information about the last indexed oplog 'serial_no' by the update-mode indexer.</p> <p>You do not need to modify this file, but IBM Rational Client Support examines it when working with you on an issue.</p>

Optional command-line options

The optional command-line options perform a specific action on your ClearQuest full-text search deployment. All commands generate informational, progress, warning, error, and instructional (to recover from error) outputs. The output that you see on the screen is also logged in the log directory. This is helpful when trying to debug or trace back your action on a deployment, because you do not need to redirect the screen output to a file.

In general, commands should not fail. If a failure does occur, most commands revert all changes. When a change cannot be reverted, an error message is displayed with instructions on what to do.

--add_record_type

<u>Abstract:</u>	Adds a new record type
<u>Summary:</u>	Use this command to add one or more new record types to your full-text search deployment. The list of new record types, and their associated fields, are provided through the XML tag <code><addRecordType></code> in the full-text search administrator configuration file.
<u>Usage:</u>	<p>If, during your initial full-text search deployment, you left out a record type, or if after your initial deployment, you added a new record type to your ClearQuest schema and you want to be able to search on the new record type, use this command to add the record type.</p> <p>Use this command if you renamed, added, or removed a field in a record type that is already indexed. To reflect such a change in your index, issue the <code>remove_record_type</code> command to remove the record type, and then issue this command to re-add it.</p>
<u>Effect:</u>	This command affects several components in your ClearQuest full-text search deployment. Your index, your ClearQuest full-text search properties XML file, your entity file, and your Solr <code>schema.xml</code> file will change.

Before issuing this command, back up your deployment. Plan this change at off-peak hours because the operation is time consuming and causes brief full-text search downtime.

Is Stateful: Yes. If the command fails during one of its execution points, you should be able to correct the issue and rerun the command. It will continue from where it left off. If a failure occurs, an error message tells you what to do.

Example: You need to add a new record type called *Wolverine*.

- 1) Edit the `cqtsadmin-TextSearch-SAMPL.xml` file and change the XML tag `<addRecordType>` to the following:

```
<newValue
required="no">Wolverine=AlterEgo,Species,Team,Affiliations
</newValue>
```
- 2) Create a backup:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
backup_fts E:\FTSBackup
```
- 3) Add the new record type:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
add_record_type
```
- 4) Edit the `cqtsadmin-TextSearch-SAMPL.xml` file and remove what you added for the XML tag `<addRecordType>`. This step is primarily a clean up task.

--archive_fts

Abstract: Archives your ClearQuest full-text search deployment.

Summary: Use this command to archive a deployment. After a deployment is archived, full-text search for the deployment is disabled. If a new WAS profile was created for this deployment (instead of the default, **cqsearchprofile**), the WAS profile is also removed.

Usage: You need to start a fresh deployment, or you no longer need the full-text search feature of a deployment. This command lets you stop full-text search services and archive all relevant resources, such as services and files. You can reference the deployment after you archive it.

Effect: This command disables full-text search. It also removes and deletes all files, resources, and settings used and set under WAS for this deployment. Your deployment data under **ftshome** remains intact but is renamed to `dbset_userdb.Archived-time-stamp`.

Is Stateful: No. If the command fails during one of its execution points, you might need to complete the archiving manually. A progress report and an error message instruct you how to recover from the error.

Example: You need to archive a deployment.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
archive_fts
```

Your archived full-text search deployment is now called:

```
D:\CQ.Search\TextSearch_SAMPL.Archived-time-stamp
```

--backup_fts

Abstract: Creates a backup copy of your ClearQuest full-text search deployment.

Summary: Use this command to create a backup copy of your full-text search deployment. Once a backup is created, you can recover data from the backup or from the whole deployment. It is recommended that you first create a backup before running commands that considerably

alter your existing deployment.

Before you back up your deployment, make sure that you have enough disk space at the destination. You need the same amount of disk space that your deployment at `ftshome` uses.

Usage: You are about to add or update a record type or your organization policy requires you to maintain periodic backups. To achieve those goals, use this command to create a backup.

Effect: While a backup is in progress, update-mode indexing is disabled. This means that full-text searches might not be up-to-date for the duration of the backup. The duration depends on the size of your index and the speed of your hard drive (and your network, if you are backing up over a LAN or WAN).

Is Stateful: No. If the command fails during one of its execution points, you need to complete the backup manually or start over, depending on the failure type and the error message that you receive. The most likely failure is not enough disk space on the destination device. No deployment data is changed during the backup.

Example: You need to create a backup of your deployment before adding new record types.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
backup_fts E:\FTSBackup
```

This creates a backup of your deployment in:

```
E:\FTSBackup\TextSearch_SAMPL.Backup-time-stamp
```

--prep_upgd_was_profiles

Abstract: Creates backup data of all your ClearQuest full-text search WAS profiles and optionally deletes (un-deploys) these WAS profiles. The command-line option **restore_was_profiles** uses this backup data to restore the deleted WAS profiles.

Summary: Run this command-line option during a scheduled downtime to prepare for a ClearQuest upgrade or reinstallation. Afterwards, use the backup data to restore those profiles back to their original state.

Usage: If you have created ClearQuest full-text search WAS profiles, which is required when deploying full-text search for two or more ClearQuest user databases, these profiles are not preserved when you upgrade or reinstall ClearQuest. This command-line option, when used with **restore_was_profiles**, creates the backup data, deletes the WAS profiles, and then restores the deleted WAS profiles (from the backup data) after your upgrade or reinstallation completes. This is required to avoid a problem during upgrade or reinstallation and to prevent you from losing your ClearQuest full-text search WAS profiles settings.

In order for the ClearQuest WAS profiles to be deleted (un-deployed), you must change the value for the XML tag `<deleteFtsWASProfiles>` from "false" to "true." If you do not, only the ClearQuest full-text search WAS profiles are backed up. Also, the default full-text search WAS profile **cqsearchprofile** is not affected by this command because it is managed by the Installation Manager.

This command-line option can be run against any **dbset**, **userdb**, or **ftshome**. It is not tied to a specific full-text search deployment; it works across all deployments.

Effect: If the XML tag `<deleteFtsWASProfiles>` is set to "true," in addition to creating a backup data of your ClearQuest full-text search WAS profiles, these WAS profiles are deleted (with the exception of **cqsearchprofile**). Until these profiles are restored, any full-text search request fails. This failure is acceptable because the only time you use this command-line option is when you are upgrading or reinstalling ClearQuest.

Is Stateful: No. This command should never fail unless an I/O error occurs.

- Example: You are about to upgrade or reinstall ClearQuest, and you have created full-text search WAS profiles for two or more ClearQuest user databases. Before you start the upgrade or reinstallation, you need to back up your WAS profiles.
- 1) Edit the `cqtsadmin-TextSearch-SAMPL.xml` file and change the XML tag `<deleteFtsWASProfiles>` to the following:
`<newValue required="no">true</newValue>`
 - 2) Issue the command-line option to create a backup data and delete the WAS profiles:
`cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --prep_upgd_was_profiles E:\FTSBackupWASProfiles`
 - 3) Change the XML tag value for `<deleteFtsWASProfiles>` back to “false” or remove the value “true.”
 - 4) Complete your ClearQuest upgrade or reinstallation.
 - 5) Issue the command-line option to restore your deleted WAS profiles from the backup data
`cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --restore_was_profiles E:\FTSBackupWASProfiles`

--clear_state

- Abstract: Clears the current state.
- Summary: Since some commands are stateful, if any of them fail to complete or stop before completion, the state is not cleared. When this happens, no other command can be issued. If you cannot correct this problem, issue this command to clear the state.
- Usage: Use this command to clear the state of a stateful command so that you can issue other commands or reissue the stateful command. When to clear the state depends on which stateful command needs to be cleared, the state the stateful command was last in, and the error message and corrective instructions provided.
- Effect: The side effect of executing this command depends on which stateful command was stopped, and how much of that command ran before stopping. The log, and error message will guide you whether you can reset the state.
- Is Stateful: No. This command should never fail unless there is an I/O error.
- Example: You need to clear the state of a stateful action so that you can recover from a nonrecoverable error, according to the error message instructions.
- ```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
clear_state
```

## --copy\_fts\_template

- Abstract: Copies the ClearQuest full-text search default template.
- Summary: When deploying full-text search on a database that is not yet enabled for full-text search, you must start from a clean default template and copy it to your **ftshome** directory. If you attempt to use this command over an existing deployment, it fails with an error.
- Usage: In general, you do not need to use this command directly because it is called when you issue the command **init\_cq\_fts**. This command is provided in case you need to fine-tune or debug your deployment.
- Effect: This command copies the default data needed for full-text search feature into your specified **ftshome** directory



Is Stateful: No. This command should never fail unless there is an I/O error.

Example: IBM Rational Client Support has instructed you to issue this command to debug a deployment issue, or to customize a deployment.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
copy_fts_template
```

#### **--create\_fts\_was\_profile *startup-type***

Abstract: Creates a ClearQuest full-text search WAS profile.

Summary: The ClearQuest full-text search feature comes with one WAS profile named **cqsearchprofile**. When you deploy full-text search, this default profile is used.

Before you can enable full-text search for additional user databases, you must create a new WAS profile for each user database. This command creates the new WAS profile.

Usage: If you are deploying full-text search for two or more user databases, create a WAS profile for each deployment.

When you create a WAS profile on Windows, you can specify the start-up type of the service. This option is ignored under UNIX and Linux because there is no service. Instead, you must set up a daemon.

The WAS profile name is determined from your database set and user database name. However, you can override it with the XML tag <ftsWASProfileName>.

Effect: This command creates a new WAS profile under WebSphere Application Server, using the next available port. When completed, disk space utilization is about 200 MB. If the profile is set to *Automatic* (under Windows), additional memory and CPU resources are consumed.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You need to deploy full-text search for a second user database. The name of your database set is *Marvel* and the user database name is *XMEN*.

- 1) Issue the following command to set up an initial deployment:  

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
Marvel --userdb XMEN --ftshome D:\CQ.Search --init_cq_fts
```
- 2) Issue the following command to create a WAS profile for this deployment:  

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
Marvel --userdb XMEN --ftshome D:\CQ.Search --
create_fts_was_profile automatic
```
- 3) Modify your entity file to contain only the record types and fields that you are interested in.
- 4) Issue the following command to complete your deployment:  

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
Marvel --userdb XMEN --ftshome D:\CQ.Search --setup_cq_fts
```

Note that you can combined steps 1 and 2:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
Marvel --userdb XMEN --ftshome D:\CQ.Search --init_cq_fts
--create_fts_was_profile automatic
```

#### **--cust\_solr\_files**

Abstract: Customizes the Solr `schema.xml` file based on your user database.

Summary: When deploying ClearQuest full-text search for the first time, you need to customize the

Solr `schema.xml` file based on your entity file.

- Usage:** The default full-text search template is generic and contains default settings which must be customized based on your user database. One files that must be customized is the Solr `schema.xml` file. This file holds all the fields of all the record types that you have set to be searched (via the entity file).
- In general, you do not need to use this command directly because it is called when you issue the command **setup\_cq\_fts**. This command is provided in case you need to fine-tune or debug your deployment.
- Effect:** This commands reads data from your entity file and customizes the Solr intermediate file and more importantly, the Solr `schema.xml` file. Thus, if you use it over an existing deployment, it refactors your `schema.xml` and if you changed your entity files after your initial deployment, your old values are lost.
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** IBM Rational Client Support asks you to debug a full-text search deployment issue, or to customize a deployment.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
cust_solr_files
```

#### --cust\_fts\_files

- Abstract:** Customizes ClearQuest full-text search properties XML file based on your user database, entity type, and full-text search administrator configuration file.
- Summary:** When deploying ClearQuest full-text search for the first time, you need to customize the full-text search properties XML files. The name of this file is `CQ-dbset-userdb.xml`.
- Usage:** The default full-text search template contains generic settings that you customize based on your user database. One file that must be customized is the full-text search properties XML file. This file holds a list of all record types and their fields, based on what is specified in your entity file. In addition, this file holds parameters such as which field to use as the display field, how often to check for changes to your ClearQuest user database, and how to communicate with your ClearQuest database and server.
- In general, you do not need to use this command directly because it is called when you issue the command **setup\_cq\_fts**. This command is provided in case you need to fine-tune or debug your deployment.
- Effect:** If you use this command on a deployment, it overwrites the `CQ-dbset-userdb.xml` file. Any changes made, either manually or by issuing commands, are lost.
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** IBM Rational Client Support asks you to debug a full-text search deployment issue, or to customize a deployment.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
cust_fts_files
```

#### --delete\_fts\_was\_profile

- Abstract:** Deletes a ClearQuest full-text search WAS profile.
- Summary:** If you created a WAS profile that you no longer need, or you want to start over again, you issue this command. When the command completes, the WAS profile under WebSphere is deleted, and resources consumed by this profile are released. However, your deployment data under **ftshome** is unchanged.

|                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <u>Usage:</u>       | In general, you do not need to use this command directly because it is called when you issue the command <b>archive_fts</b> . This command is provided if you need to fine-tune or debug your deployment.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <u>Effect:</u>      | Resources consumed by WebSphere of this WAS profile are all released. If ClearQuest Web full-text search was not disabled (with the command <b>disable_cqweb_fts</b> ), full-text search searches result with an error.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <u>Is Stateful:</u> | No. This command should never fail unless there is an I/O error.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <u>Example:</u>     | <p>You need to rename your WAS profile but do not want to completely redeploy your full-text search solution.</p> <ol style="list-style-type: none"> <li>1) First, delete the WAS profile:<br/> <pre>cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --delete_fts_was_profile</pre> </li> <li>2) Edit <code>cqtsadmin-dbset-userdb.xml</code>, and change the XML tag <code>&lt;ftsWASProfileName&gt;</code> from the automatically generated default to the new name for the WAS profile. The name must be unique, or the command will fail.</li> <li>3) Next, re-create the WAS profile:<br/> <pre>cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --create_fts_was_profile automatic</pre> </li> <li>4) If you are also changing the port number, edit <code>cqtsadmin-dbset-userdb.xml</code>, update the port number in the XML tag <code>&lt;ftsWASProfilePort&gt;</code>, and then issue this command:<br/> <pre>cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --enable_cqweb_fts</pre> </li> </ol> |

## --disable\_cqweb\_fts

|                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <u>Abstract:</u> | Disables ClearQuest full-text search under ClearQuest Web GUI.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <u>Summary:</u>  | <p>Use this command to disable the <b>Full Text</b> radio button in the ClearQuest Web GUI. If your ClearQuest database is replicated, there is no impact on your oplogs or oplogging.</p> <p>If your ClearQuest database is not replicated, oplogging is stopped. If you reenable full-text search (<b>enable_cqweb_fts</b>), any changes made during this period to record types that you have enabled for searching are lost. In effect, your search result may not be complete.</p> <p>Do not issue this command on a full-text search configured ClearQuest user database that has been fully or partially indexed. If it becomes necessary to disable the radio button, prevent ClearQuest record changes until it is enabled again.</p> <p>In nonreplicated ClearQuest databases, disabling the radio button also disables oplogging (the mechanism by which ClearQuest full-text search tracks changes to ClearQuest records). If ClearQuest records are changed when this button (and therefore oplogging) is disabled, the changed records are never reindexed and you will have to perform a complete re-indexing of the user database.</p> <p>NOTE: On a UNIX or Linux operating system, this command is not supported. When you attempt to issued it, you receive instructions on how to issue this command from a Windows system.</p> |
| <u>Usage:</u>    | You do not need to use this command unless you are in a test environment or IBM Rational Client Support instructs you to do so.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <u>Effect:</u>   | Oplog generation stops (if your database is not replicated) and the <b>Full Text</b> radio button,                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

found in the ClearQuest Web GUI, is disabled.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: IBM Rational Client Support instructs you to disable full-text search to help you resolve issues you have been experiencing.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
disable_cqweb_fts
```

#### --enable\_cqweb\_fts

Abstract: Enables ClearQuest full-text search under the ClearQuest Web GUI.

Summary: The **Full Text** radio button in the ClearQuest Web GUI is enabled with this command. Oplog generation is also enabled if your ClearQuest database is not replicated and is feature level 7,. If your ClearQuest user database is replicated, no change is made to oplog generation.

Usage: You do not need to use this command unless you are in a test environment or IBM Rational Client Support instructs you to do so.

On a UNIX or Linux operating system, this command is not supported. When you attempt to issued it, you receive instructions on how to issue this command from a Windows system.

Effect: Oplog generation starts (if your database is not replicated, and is at feature level 7) and the **Full Text** radio button in the ClearQuest Web GUI is enabled.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: IBM Rational Client Support instructs you to enable full-text search to help you resolve issues you have been experiencing.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
enable_cqweb_fts
```

#### --fresh\_batch\_idx

Abstract: Forces a full reindex using the batch-mode indexer.

Summary: Batch-mode indexing is the process of reading all ClearQuest records configured for searching and sending the data of those records to the search engine (Solr) for indexing. Batch-mode indexing is performed as part of running the command **setup\_cq\_fts**. Use this command to force a fresh batch indexing after the initial indexing.

Before you issue this command, issue the **stop\_update\_idx** command to stop the update-mode indexer. If you do not, the batch-mode indexer might replace the data of a more recent record indexed by the update-mode indexer.

When this command completes re-indexing, your index is fragmented. Usually, this has no performance impact on your searches. However, the index size might grow to up to twice the current size. To optimize the index and reduce its size, run the command **optimize\_idx**.

Usage: You do not need to use this command unless you are in a test environment or IBM Rational Client Support instructs you to do so.

Effect: The search index is updated. Its size grows to up to twice the current size. Therefore, confirm that you have enough disk space before using this command. While re-indexing, search results may not be complete because you disable the update-mode indexer,

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: IBM Rational Client Support instructs you to completely reindex your searchable records.



- 1) Stop the update-mode indexer:  

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
stop_update_idx
```
- 2) Force batch mode reindexing:  

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
fresh_batch_idx
```
- 3) Run index optimization:  

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
optimize_idx
```
- 4) Enable update-mode indexer:  

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
start_update_idx
```

**Note:** You can combine these steps into one command. You might want to do this because this operation takes a long time if you reindex many records:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
stop_update_idx --fresh_batch_idx --optimize_idx --
start_update_idx
```

## --fresh\_update\_idx

- Abstract:** Forces update-mode indexer to begin indexing again, starting with the first recorded oplog in your ClearQuest user database.
- Summary:** Update-mode indexing is the process of monitoring the ClearQuest database for changes on record types that are configured for searching. The monitoring is done by checking for oplogs in the ClearQuest database. Update-mode indexing is enabled as part of the command **setup\_cq\_fts**.
- Unlike the **fresh\_batch\_idx**, this command does not require you to stop the update-mode indexer.
- When this command completes against a populated index, your index will be fragmented. Usually, this has no performance impact on your searches. However, the index size might grow to be up to twice the current size. To optimize the index, and reduce its size, run the command **optimize\_idx**.
- Consider carefully whether to start update-mode indexer from the first recorded oplog. Over time, especially if you have not been consistently purging oplogs, you amass an oplog for every action taken in a ClearQuest record and you might not want to index from the first recorded oplog. This command is intended for use with testing environments and to debug full-text search deployment issues working with IBM Rational Client Support.
- Usage:** You do not need to use this command unless you are in a test environment or IBM Rational Client Support instructs you to do so.
- Effect:** The search index is updated. Its size grows to up to twice the current size. Therefore, confirm that you have enough disk space before using this command..
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** IBM Rational Client Support instructs you to force a reindexing of update-mode indexer.
- ```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
fresh_update_idx
```

**--gather_diagnostic_data**

- Abstract:** Used to collect diagnostic data of the IBM Rational ClearQuest full-text search profile for diagnosing issues.
- Summary:** This command gathers relevant data about your ClearQuest full-text search deployment. The data is copied into a directory that you might be instructed to send in to IBM Rational Client Support. Before you send the data, verify that it contains no confidential information. Usually the most sensitive data are the record-type names you enabled for searching and a history of search terms performed by your organization, for the past few days. The history is the log maintained by WebSphere for your WAS profile.
- Passwords are converted to asterisks (*) when outputting to the screen or log files. They are never stored in plain text.
- Usage:** Use this command to gather and send diagnostic data to IBM Rational Client Support to help you diagnose issues with full-text search.
- Effect:** None of your data or configuration settings are changed. A new directory is created, with the name of your deployment and a time stamp. The cumulative size of the diagnostic data varies, depending on the total sizes of logs in your deployment, which is usually in megabytes. The actual index is not part of the diagnostic data.
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** IBM Rational Client Support instructs you to send diagnostic data to help you resolve issues with full-text search.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
gather_diagnostic_data
```

This command creates the diagnostic data and places it in:

```
D:\CQ.Search\TextSearch_SAMPL.Diag-time-stamp
```

--gen_entity_file

- Abstract:** Generates the entity file based on your user database.
- Summary:** When deploying full-text search against a ClearQuest user database for the first time, you need an entity file that holds all record types and their fields that are referenced by your ClearQuest user database schema. This entity file is used as an input to generate the full-text search properties XML file, the Solr Intermediate XML file, and the Solr `schema.xml` file. By default, the entity file contains all record types and all their fields of your ClearQuest schema. You might not want to enable searching on all of these fields. You can customize the entity file to specify which record types and which fields in these types that you want to be enabled for or excluded from searching..
- The contents of the entity file determines the final set of record types and fields to search on.
- Usage:** In general, you do not need to use this command directly because it is called when you issue the command **setup_cq_fts**. This command is provided in case you need to fine-tune or debug your deployment.
- .
- Effect:** If you use this command against an existing deployment, it overwrites your `Entity-dbset-userdb.txt` file and you lose the edits that you have made.
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** IBM Rational Client Support has instructed you to debug a full-text search deployment issue or to customize a deployment.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
gen_entity_file
```

--gen_fts_files

- Abstract:** Generates ClearQuest full-text search properties XML file based on your user database, and entity file.
- Summary:** When deploying full-text search on a user database for the first time, you need to generate the full-text search properties XML file. This file holds information about your deployment settings, such as how often to index, the batch size for indexing, and search server information.
- Usage:** In general, you do not need to use this command directly because it is called when you issue the command **setup_cq_fts**. This command is provided in case you need to fine-tune or debug your deployment.
- Effect:** If you use this command against an existing deployment, it overwrites your *Entity-dbset-userdb.txt* file and you lose the edits that you have made..
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** IBM Rational Client Support has instructed you to debug a full-text search deployment issue or to customize a deployment.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
gen_fts_file
```

--help

- Abstract:** Displays help context and exits.
- Summary:** This command displays available commands, abstracts, and examples.
- Usage:** You are familiar with **cqtsadmin.pl** but you need a quick refresher about the available commands. This command lists all available commands.
- Effect:** None. This command does not change any data and does not require authentication to run.
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** You need to get a list of available commands.

```
cqperl cqtsadmin.pl --help
```

--init_cq_fts

- Abstract:** Copies ClearQuest full-text search default template and generates the default entity file; in effect, this command runs **copy_fts_template** and **gen_entity_file**.
- Summary:** This command is one of the first commands that you run when deploying full-text search. Upon completion, it creates a directory in your **ftshome** directory called *dbset_userdb*. That is, if your database set name is *TextSearch* and your user database name is *SAMPL*, then the directory is *TextSearch_SAMPL*. All default data and settings related to this deployment are placed in this directory.

One of the key files that is created with this command is the entity file. This entity file is called *Entity-dbset-userdb.txt* and is placed in your **ftshome** directory. When customizing your deployment, you decide if you want to leave it as is, or to remove record types and fields that you do not want to be searched.

Another key file that is created is the full-text search administrator configuration file, which is called *cqftsadmin-dbset-userdb.xml*. This file holds additional default settings

specific to your deployment and environment such as the server name, your WAS profile name, the index batch size, and the index frequency.

Refer to part 3 in developerWorks article to learn more about the entity file and the full-text search administrator configuration file.

Usage: Use this command to preconfigure your full-text search deployment. It create your "ftshome" directory, copy the default files to it, and set default values. Before you complete your deployment, you typically customize the entity file.

Effect: This command creates a new **ftshome** directory if it does not exist, and copies the default data and settings for your deployment into it.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You need to start a new full-text search deployment for one of your user databases.

- 1) Run this command to preconfigure your deployment:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
init_cq_fts
```

- 2) Edit this deployment's entity file and remove record types or fields that should not be searchable:

```
D:\CQ.Search\TextSerch_SAMPL\Entity-TextSearch-SAMPL.txt
```

- 3) Complete your deployment by running the command

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
setup_cq_fts
```

- 4) Based on your schema design, decide which field for each record type best represents the display field. Edit your full-text search properties XML file and select the display field for each record type. For example, change:

```
<field name="Headline" disp_order="0" index="true"></
field>
```

to:

```
<field name="Headline" disp_order="1" index="true"></
field>
```

- 5) Run this command to reflect this change for future full-text search hits:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
enable_cqweb_fts
```

This change is visible to users on their next ClearQuest session login.

--lock_cq_fts

Abstract: Locks this ClearQuest full-text search deployment so that the full-text search administrator tool can run only nonmodifiable commands..

Summary: This command prevents accidental modification to a full-text search deployment. Once completed, almost all available commands are blocked with a message that the deployment is locked. Unlock the deployment by issuing the command **unlock_cq_fts**.

Usage: After you deploy full-text search, and want to prevent inadvertent modifications, issue this command to place a lock on the deployment.

This lock is weak because anyone with proper file system access or ClearQuest privileges can unlock a deployment. The goal of this command is to give administrators a way to signal that the deployment is complete and further modification should be communicated and issued with care.

Effect: None. Your full-text search deployment data and settings are not affected.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You have completed a deployment and want to ensure that the deployment is not modified.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
lock_cq_fts
```

--optimize_idx

Abstract: Optimizes (defragments) the search index.

Summary: This command optimizes the search index. Once completed, the index is rewritten and is no longer fragmented.

Usage: As you add or modify records in the search index, it can become fragmented. Fragmented indexes tend to grow larger than unfragmented ones. In addition to growth, there might be a slight performance degradation. To reduce index size and restore performance, optimize your index at least once a year. Optimize it more frequently if your ClearQuest database experiences heavy activity with record modifications, additions, or deletions.

Before issuing this command, make sure that you have enough free disk space. In general, if your index is 2 GB, make sure you have at least 2 GB (preferably 2.5 GB) of free disk space before you start the optimization. Otherwise the optimization fails, but your original index remains intact. Free disk space is required because the original index is rewritten during optimization. The old index is kept until the new index is regenerated.

The time it takes to optimize an index depends on the size of your index and the speed of your hard drive and I/O. Optimization can take a few hours on a 2 GB index. While optimizing, all full-text search services remain available, including update-mode indexer. However, there may be slight performance degradation, so plan index optimization for off-peak hours.

Effect: Your search index is rewritten. If an I/O error occurs during optimization, most likely due to being out of disk space, the original index remains intact. The original index might be larger, but it will return to its original size after its optimization completes.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You need to optimize your search index.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
optimize_idx
```

--remove_lucene_idx_lock

Abstract: Removes the Lucene index lock (if for some reason it gets into a stale state).

Summary: Use this command to remove a Lucene lock on the search index. The search engine, Lucene, uses “locks” to synchronize updating. In very rare cases, if Lucene or the server encounters an error when a lock is obtained, and Lucene cannot recover gracefully, the lock will remain active. While a lock is active, the search index cannot be updated. Therefore, no ClearQuest records can be added or updated. Full-text searches on the index continue to work unless there is an integrity problem with the index.

To recover from such a lock, restart the WAS profile by issuing **stop_fts_was_profile**,

start_fts_was_profile, and this command on a command line.

While this command is running, full-text searches are interrupted for the duration of the service restart.

Usage: You notice that newly added ClearQuest records are not appearing in search results. You examine the WAS profile logs and discover that Lucene is reporting errors that the index is locked. Use this command to clear the lock.

Effect: None. Your full-text search deployment data and settings are not affected.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You need to remove a Lucene index lock. Stop, start, and unlock the index in one step:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
stop_fts_was_profile --start_fts_was_profile --
remove_lucene_idx_lock
```

--remove_record_type

Abstract: Removes a record type.

Summary: This command removes one or more ClearQuest record types from the search index. Any further searches no longer find matches belonging to the removed record types.

This command is used with data that you need to provide via your full-text search administrator configuration file `cqtsadmin-dbset-userdb.xml`. In your configuration file, for the XML tag `<removeRecordType>`, list the names of record types that you want to remove. To remove multiple record types, separate their names with a semicolon (“;”).

This command runs in states. If an error occurs during one of the states, an error message tells you how to correct the issue. When you restart the command, it continues from where it left off.

Create a backup of your deployment before you issue this command.

Removal of record types, removes data related to that record type only from the search index. Your ClearQuest is not affected by this command.

Usage: After deploying full-text search based on your ClearQuest schema, you are asked to no longer allow searches on certain record types. This command removes those record types from the index, and thus the record types are no longer searchable.

Another use of this command is when record types that are indexed have changed in your ClearQuest schema. Perhaps you added new fields or renamed fields. To reflect this change in your search index, use **remove_record_type**, followed by **add_record_type**.

Effect: Your search index is altered so that any reference to the removed record types no longer exists, and thus is not searchable. In addition, references to these record types are removed from your full-text search properties XML file and entity file.

While this command runs, search services are briefly interrupted when the WAS profile is restarted. Users might get an error that the server is down. Also, users who have a ClearQuest session open after the command completes still see the removed record types in their **Search Scope** of the ClearQuest Web GUI. If they attempt to search these record types, they receive no hits. In order for their **Search Scope** to reflect the search index, these users must log in again.

This command causes index fragmentation. Optimize the index after running this command so that both the index size and performance are at peak level.

Is Stateful: Yes. If the command fails during one of its execution points, you should be able to correct

the issue and rerun the command. It will continue from where it left off. If a failure occurs, an error message tells you what to do.

Example:

You need to remove two record types from your search index.

- 1) Edit the `cqtsadmin-TextSearch-SAMPL.xml` file and add the record types that you want to remove to the XML tag `<removeRecordType>`. Use a semicolon (;) to separate record types:
`<newValue required="no">Email_Rule;Customer</newValue>`
- 2) Run the command to remove the two record types:
`cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --remove_record_type`
- 3) Edit the `cqtsadmin-TextSearch-SAMPL.xml` file and remove the two record types that you added to the XML tag `<removeRecordType>`. This housekeeping task prevents accidental removal if you subsequently add these removed record types again.

--repair_records

Abstract:

Indexes records that failed to be indexed by the batch- or update-mode indexer.

Summary:

During batch- and update-mode indexing, if the process runs into issues that prevent reading records or sending them to the server for indexing, a repair file is created that lists the ClearQuest record-IDs of the failed records. This command reads the repair file and reindexes one record at a time to reduce the chance of another failure.

Usage:

As part of your full-text search deployment, when batch indexing is completed, you should run this command to index any records that might not have been indexed. You should also periodically check **ftshome** for any records that are not indexed during update-mode indexing. If records are not indexed, you see files with the naming convention *record-type-nametime-stamp.xml* (for example, *Defect1222923990646.xml*). If you see these files, run this command to index the records.

Effect:

Your search index includes data from the newly indexed records.

Is Stateful:

No. This command should never fail unless there is an I/O error.

Example:

You need to index records that were not indexed during batch or update indexing.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
repair_records
```

--restore_was_profiles

Abstract:

Restores your ClearQuest full-text search WAS profiles from a backup data created with the command-line option **prep_upgd_was_profiles**.

Summary:

Use this command-line option to restore your ClearQuest full-text search WAS profiles from backup data that was created with the command-line option **prep_upgd_was_profiles**.

Usage:

If you have created ClearQuest full-text search WAS profiles, which is required when deploying full-text search for two or more ClearQuest user databases, these profiles are not preserved when you upgrade or reinstall ClearQuest. This command-line option restores all your ClearQuest full-text search WAS profiles after your upgrade or reinstallation of ClearQuest completes.

This command-line option fails if you attempt to restore a ClearQuest full-text search WAS profile that already exists.

You can run this command-line option on any **dbset**, **userdb** or **ftshome**.



Effect: Your full-text search WAS profiles are re-created and restored to their original settings.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You have completed your upgrade or reinstallation of ClearQuest and you need to restore all your ClearQuest full-text search WAS profiles from your backup data.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
restore_was_profiles E:\FTSBackupWASProfiles
```

--run_batch_idx

Abstract: Runs the batch-mode indexer.

Summary: This command starts the batch-mode indexer. It indexes all ClearQuest records that you configured for searching. While it is in progress, you can stop it using Ctrl-C, and then resume running the command from where it stopped. To force a complete reindexing, use **fresh_batch_idx**.

This command may fail if the batch size, or number of threads is set too high, or if your ClearQuest records have a lot of data. The most common failure results from being out of memory. In this case, either reduce the batch size or the number of threads in use. This increases indexing time. Alternatively, you can also increase JVM memory for both the batch-mode indexer, using the XML tag <batchIndexJVMParm>, and the search server's memory, using the XML tag <ftsWASProfileMaxHeapSize>. If you must increase memory, this is a temporary requirement until the batch indexer completes. IBM Rational Client Support can help determine the best action in these circumstances.

Usage: In general, you do not need to use this command directly because it is called when you issue the command **setup_cq_fts**. This command is provided in case you need to fine-tune or debug your deployment.

Effect: This command runs ClearQuest SQL queries against your ClearQuest database. It then extracts resulting records and sends them to the Solr search server for indexing. In effect, while this command runs, your ClearQuest database server is taxed, and your search index is updated.

Is Stateful: No. This command should never fail unless an I/O error (network or disk), an out-of-memory error, or an unexpected configuration error occur.

Example: You need to index all ClearQuest records.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
run_batch_idx
```

--setup_cq_fts

Abstract: Performs end-to-end ClearQuest full-text search setup using current settings from the entity file and the full-text search administrator configuration file.

Summary: This command is used to complete a full-text search deployment. First, it customizes and configures your entity file, your full-text search properties XML file and your Solr `schema.xml` file, based on your full-text search administrator configuration file, ClearQuest database, and operating system. Second, it enables ClearQuest full-text search in ClearQuest Web GUI (the **Full Text** radio button is enabled, and oplog generation begins if your ClearQuest database is feature level 7 and is not replicated). Third, it starts batch-mode indexing (that is, indexing all your ClearQuest records for the record types that you set for searching in your entity file). Finally, it enables update-mode indexing, which completes your deployment.

This command maintains its state. If an error occurs before it completes, the state is set and

you need to correct the error before you can continue. The error message and the logs contain instructions on how to recover from the error. The recovery steps depend on the error and the state in which the error occurred.

Usage: Use this command to complete your full-text search deployment. You typically run this command after customizing your entity file. While this command runs, the **Full Text** radio button is enabled in the ClearQuest Web GUI, and users who log in again are able to search. However, the search results are not complete until the deployment is complete. It is necessary for the radio button to be enabled because oplogs must be generated (especially if your ClearQuest database is not replicated) to capture all record changes that occur during and after batch-mode indexing.

Because batching mode indexing is CPU and I/O bound (and memory if you increase the default JVM memory setting), expect to see high utilization of the computer on which you are deploying. Also, depending on how aggressively you set your batch-mode indexer (increasing the batch size and number of threads), expect to also see high utilization of your ClearQuest database during batch-mode indexing.

Effect: Several files are created in your **ftshome** directory; the index is created, search services under WAS are enabled, and your ClearQuest database is updated to include the full-text search properties XML file (if on a Windows host). If your ClearQuest database is not replicated, oplogging is enabled. ClearQuest-replicated user databases continue their normal oplogging.

Is Stateful: Yes. If the command fails during one of its execution points, you should be able to correct the issue and rerun the command. The command continues from where it left off. If a failure occurs, an error message tells you what to do.

Example: You have run the command **init_cq_fts** and edited your entity file. You now want to complete your deployment.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
setup_cq_fts
```

--set_was_max_mem

Abstract: Sets the JVM MAX memory for the WAS profile.

Summary: This command is used to set the maximum JVM memory that the WAS profile is allowed to use. The default is 300 MB which, may be too small, especially during batch-mode indexing if the batch size has been increased, the number of threads has increased, or the ClearQuest record types are complex (many fields, large data sets).

This command reads the memory setting from your configuration file, under the XML tag **<ftsWASProfileMaxHeapSize>** and sets the JVM memory to this value.

Usage: If indexing fails (for example, during batch-mode indexing an out-of-memory error is reported from the WAS profile server), this the memory setting is probably too low. Do one of the following to recover from an out-of-memory error:

- Reduce the batch size and number of threads. This increases the time it takes to complete indexing.
- Temporarily increase the JVM memory.

The JVM maximum memory should typically be set higher on a temporary basis until batch-mode indexing completes.

Effect: The JVM maximum memory setting for the WAS profile changes to the new value. In effect, more system memory is allocated to the WAS profile.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You are getting out-of-memory errors from the WAS profile while indexing. You need to address this issue before you can continue.

- 1) Edit the `cqtsadmin-TextSearch-SAMPL.xml` file and set the JVM memory in the XML tag `<ftsWASProfileMaxHeapSize>` to 1.5 GB:
`<newValue required="no">1536</newValue>`
- 2) Run this command to set the new JVM memory setting.
`cqperl cqtsadmin.pl --username admin --password "" --dbset TextSearch --userdb SAMPL --ftshome D:\CQ.Search --set_was_max_mem`

--set_solr_home

Abstract: Sets the Solr home directory under the WAS profile for this ClearQuest user database.

Summary: This command is used to set the Solr home directory under WAS. This is because each deployment has its own *schema.xml* configuration file and index. If the Solr home is not set properly, or is set to the wrong location, the WAS full-text search profile might not start. Should this occur, errors are logged to the corresponding full-text search WAS profile logs directory (within the path `%RATIONAL_COMMON%/CM/profiles/cmprofile/logs/server1/logs/`).

Usage: In general, you do not need to use this command directly because it is called when you issue the command **setup_cq_fts**. This command is provided in case you need to fine-tune or debug your deployment.

Effect: The JVM property of your deployed WAS profile is changed so that Solr's home environment variable is set.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You have been instructed by IBM Rational Client Support to use this command to debug a full-text search deployment issue, or to customize a deployment.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
set_solr_home
```

--scrub_oplog

Abstract: Scrubs oplogs created before the given date.

Summary: This command is used to scrub oplogs from a non-replicated ClearQuest user database. ClearQuest generates oplogs to keep track of changes as they are made to your records. ClearQuest full-text search monitors oplogs during update-mode indexing to synchronize the search index with these changes.

Because oplogs are kept in the ClearQuest database and are temporary data, there is no point in keeping them indefinitely. To prevent continuous oplog growth, scrub old oplogs periodically.

If your ClearQuest user database is replicated, use your replication tool and policy for oplog scrubbing. If you attempt to use this command, it fails with an error message instructing you to use your replication tools.

If your ClearQuest user database is not replicated, scrub oplogs as infrequently as possible, based on your rate of oplog creation.

Never scrub all oplogs, especially if some are not yet processed by the update-mode indexer. Doing so makes your search index unsynchronized with your ClearQuest records, and thus searches might not be accurate or complete. This scenario requires batch-mode reindexing.

Aggressive oplog scrubbing where update-mode indexer throughput is insufficient may lead to index inaccuracy and missed hits. Either ensure update-mode indexer is up-to-date, scrub only older oplogs (greater than one month), or skip oplogging scrubbing, because it is not critical.

Usage: To conserve database space and to clean up unused data, you may need to scrub oplogs in a non-replicated ClearQuest user database. If your ClearQuest database is replicated, you should never use this command.

Effect: ClearQuest oplogs in your oplog table that are created before the specified date are deleted.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: As ClearQuest administrator, you need to periodically scrub old oplogs.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
scrub_oplog "31-Oct-2009"
```

For supported date formats, consult ClearQuest User's Guide for oplog scrubbing.

--start_fts_was_profile

Abstract: Starts the ClearQuest search profile under WAS.

Summary: This command is used to start the full-text search WAS profile service. A search WAS profile must be started for search requests to be serviced, as well as for the update-mode indexer to start checking for new or updated records and send them to the search engine for indexing.

Usage: In general, you do not need to use this command directly because it is called when you issue the command **setup_cq_fts**. This command is provided in case you need to fine-tune or debug your deployment.

Effect: If the search WAS profile was stopped, it will start, which causes temporary memory and CPU resource consumption. If search services and the update-mode indexer are enabled, they begin running.

If the WAS profile is already started, no changes is made.

Is Stateful: No. This command should never fail unless there is an I/O error, a Solr server failure, or a WAS profile startup failure, which is typically because of setup issues.

Example: You have been instructed by IBM Rational Client Support to start the search WAS profile.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
start_fts_was_profile
```

--start_update_idx

Abstract: Enables and starts the update-mode indexer under WAS.

Summary: This command is used to enable and then start the update-mode indexer, which runs under the WAS profile. The update-mode indexer synchronizes the search index with changes made to the ClearQuest database. It monitors oplogs for new values to be indexed. You configure how often the index is synchronized with the XML tag `<updateIndexDelay>` of your full-text search administrator configuration file.

Usage: In general, you do not need to use this command directly because it is called when you issue the command **setup_cq_fts**. This command is provided in case you need to fine-tune or debug your deployment.

Effect: The update-mode indexer, which runs under the WAS profile, is enabled and started. Modifications, additions, and removal of ClearQuest records is now indexed, and thus

appears in search results.

Is Stateful: No. This command should never fail unless there is an I/O error, a Solr server failure, or a WAS profile startup failure, which is typically because of setup issues.

Example: You have been instructed by IBM Rational Client Support to enable the update indexer.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
start_update_idx
```

--stop_fts_was_profile

Abstract: Stops the ClearQuest search profile under WAS profile service.

Summary: This command is used to stop the full-text search WAS profile service. If a WAS profile is stopped, then search services are unavailable and the update-mode indexer stops synchronizing the search index with the changes made to your ClearQuest records.

Usage: In general, you do not need to use this command directly because it is called when you issue the command **setup_cq_fts**. This command is provided in case you need to fine-tune or debug your deployment.

Effect: The search WAS profile stops, which means that search services and update indexing also stop. Memory and CPU used by the WAS profile is released. Also, any search results in an error.

Oplogging continues (as long as the **Full Text** radio button remains enabled in the ClearQuest Web GUI), which permits the update-mode indexer to catch up once the WAS profile is running again.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You have been instructed by IBM Rational Client Support to stop the search WAS profile.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
stop_fts_was_profile
```

--stop_update_idx

Abstract: Disables and stops the update-mode indexer under the WAS search profile.

Summary: This command is used to disable and stop the update-mode indexer. When the update-mode indexer stops, search services are available but any changes to ClearQuest records are not reflected in the index until update-mode indexing is reenabled or resumed. Therefore, searches might yield results that are neither up-to-date nor accurate.

Usage: In general, you do not need to use this command directly because it is called when you issue the commands **setup_cq_fts** and **backup_fts**. This command is provided in case you need to fine-tune or debug your deployment.

Effect: The update-mode indexer, which runs under the WAS profile, is disabled. New and modified ClearQuest records are not reflected in search results.

Oplogging continues, and ClearQuest record changes are reflected in full-text searches after the update mode record indexer is reenabled and has time to catch up with the current oplogs.

Is Stateful: No. This command should never fail unless there is an I/O error.

Example: You have been instructed by IBM Rational Client Support to disable the update-mode indexer.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
```

stop_update_idx

--unlock_cq_fts

- Abstract:** Unlocks this ClearQuest full-text search deployment so that all full-text search Administrator commands can run.
- Summary:** This command is used to unlock a locked deployment of full-text search. Once completed, all available full-text search Administrator commands can run.
- Usage:** Use this command to unlock a locked deployment of full-text search so you can run all available commands. This command reverses the lock done by the command **lock_cq_fts**.
- Effect:** None. Your full-text search deployment data and settings are not affected, but all commands can now run.
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** You need to optimize your index, but your deployment is locked. You choose to unlock it, optimize the index, and then relock it.
- ```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
unlock_cq_fts --optimize_idx --lock_cq_fts
```

## --show\_scenarios

- Abstract:** Displays a list of scenarios, with examples, of how to use **cqtsadmin.pl** tool.
- Summary:** This command is used to display a list of scenarios, with examples, of how to use **cqtsadmin.pl** tool. The scenarios are an abbreviated form of the scenarios listed in this guide and might not be the complete list or match one-to-one.
- Usage:** Use this command-line option to display a list of the most commonly used scenarios for **cqtsadmin.pl** tool without having to reference the User's Guide manual. Issue the command without a parameter to see the scenario IDs and headlines. Pass a scenario ID as a parameter to see the full text of a scenario. Pass the **all** parameter to see a complete list of scenarios, each with an ID, headline, and a full text description.
- Effect:** None. Your full-text search deployment data and settings are not affected.
- Is Stateful:** No. This command should never fail unless there is an I/O error.
- Example:** You are about to use **cqtsadmin.pl** to complete a task but cannot remember exactly how to do so. Use this command-line option to list the scenario headlines with their IDs.
- ```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
show_scenarios
```

To see the full text of a scenario, rerun this command but pass the scenario ID that you are interested in.

--update_fts_prop_files

- Abstract:** Updates all generated ClearQuest full-text search files.
- Summary:** After you change any of the following XML tags in your full-text search administrator configuration file, **cqtsadmin-dbset-userdb.xml**, you must propagate the changes for them to take effect:
- ```
<batchIndexBatchSize>, <batchIndexDelay>, <batchIndexThreads>,
<updateIndexBatchSize>, <updateIndexDelay>, <updateIndexLoginInterval>,
<ftsWASProfileName>, <ftsWASProfilePort>, and <ftsServerName>
```

For example, if you change the batch size for the batch-mode indexer, in order for the new

value to be used, you must issue this command before running the command **run\_batch\_idx**.

**Usage:** As you deploy full-text search, you might need to change a default setting. When a change is made to the configuration file, the change has to propagate to the appropriate full-text search components.

**Effect:** The component that is affected depends on which XML tag value you change. See the section covering “cqtsadmin-<dbset>-<userdb>.xml” file for the XML tag documentation for details.

**Is Stateful:** No. This command should never fail unless there is an I/O error.

**Example:** You need to speed up batch-mode indexing by increasing the batch size and number of threads. After changing the configuration file, issue this command.

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome D:\CQ.Search --
update_fts_prop_files --run_batch_idx
```

## Scenarios

IBM Rational ClearQuest 7.1.1.x full-text search administrator tool simplifies the process of setting up and configuring full-text search. This section provides some scenarios to help you better understand how to deploy this tool in your environment.

### I want to enable full-text search on an out-of-the-box SAMPL ClearQuest database to learn about its configuration and capability

This scenario assumes that you have a ClearQuest schema repository with a connection called TextSearch and a user database called SAMPL.

This scenario takes about 15 minutes to complete if used on the SAMPL ClearQuest database, which has few records. The steps should take about 10 minutes, and running commands should take about 5 minutes.

**Note:** In the examples, ellipses (“...”) represent text that is omitted to save space.

#### Windows operating system:

- 1) At the command prompt, issue the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --init_cq_fts
```

After this command completes, configuration data related to your ClearQuest schema is placed in: C:\CQ.Search\TextSearch\_SAMPL\.

- 2) Edit the file C:\CQ.Search\TextSearch\_SAMPL\Entity-TextSearch-SAMPL.txt. For each record type, select a field to use as a display field, and add an ampersand (“&”) in front of that field. For example, change:

```
Customer=Attachment,CallTrackingID,...,Name,Phone,...
Defect=Attachments,Description,Headline,Keywords,...
Email_Rule=Action_Types,Actions,...,Name,Operator_Value,...
Project=Description,Name,dbid,...
```

to:

```
Customer=Attachment,CallTrackingID,...,&Name,Phone,...
Defect=Attachmets,Description,&Headline,Keywords,...
Email_Rule=Action_Types,Actions,...,&Name,Operator_Value,...
```

Project=Description, **Name**, dbid, ...

The ampersand tells ClearQuest Web to display the value of this field, for this record type, in the full-text search results.

You are not required to select a field name for each record type. The field dbid is the default display field. You can always change the display field later without reindexing your data, as explained in the scenario [I want to change the display field for already-indexed record types](#).

**Note:** If you were setting up ClearQuest full-text search on your production ClearQuest database, in addition to selecting the display field, you would also select which record types and associated fields to index. For example, if you do not want to index the record type Email\_Rules, delete it from the list. To index only two fields, Name and Phone, for the record type Customer, then delete the other fields and keep these two.

- 3) Edit the file C:\CQ.Search\TextSearch\_SAMPL\cqtsadmin-TextSearch-SAMPL.xml and change the value for the XML tag <updateIndexDelay> from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue required="no">10</newValue>
```

This changes the default value of update-mode indexing delay from 10 minutes (600 seconds) to 10 seconds. This means it will take up to 10 seconds (not 10 minutes) for a change that you make to a ClearQuest record to appear in a full-text search result.

**Note:** Setting such a low value is not recommended for production use. Doing so will cause excessive logon and logoff calls to the ClearQuest database, consuming CPU, memory, and database resources.

- 4) To complete your deployment, run the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --setup_cq_fts
```

**Note:** ClearQuest Web client users can use full-text search during batch indexing, but results will not be complete until batch indexing is finished.

- 5) Under Windows Services, find the service called IBM WebSphere Application Server V6.1 - cqsearchprofile and change the Startup type from Manual to Automatic. This ensures that both Solr and the record indexer automatically start up on a system reboot.

#### UNIX or Linux operating system:

- 1) At the command prompt, issue the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --init_cq_fts
```

After this command completes, configuration data related to your ClearQuest schema is placed in: C:\CQ.Search\TextSearch\_SAMPL\.

- 2) Edit the file C:\CQ.Search\TextSearch\_SAMPL\Entity-TextSearch-SAMPL.txt. For each record type, select a field to use as a display field, and add an ampersand (“&”) in front of that field. For example, change:

```
Customer=Attachment, CallTrackingID, ..., Name, Phone, ...
```

```
Defect=Attachmets, Description, Headline, Keywords, ...
```

```
Email_Rule=Action_Types, Actions, ..., Name, Operator_Value, ...
```

```
Project=Description, Name, dbid, ...
```

to:

```
Customer=Attachment,CallTrackingID,...,&Name,Phone,...
Defect=Attachmets,Description,&Headline,Keywords,...
Email_Rule=Action_Types,Actions,...,&Name,Operator_Value,...
Project=Description,&Name,dbid,...
```

The ampersand tells ClearQuest Web to display the value of this field, for this record type, in the full-text search results.

You are not required to select a field name for each record type. The field dbid is the default display field. You can always change the display field later without reindexing your data, as explained in the scenario [I want to change the display field for already-indexed record types](#).

**Note:** If you were setting up ClearQuest full-text search on your production ClearQuest database, in addition to selecting display field, you would also select which record types and associated fields to index. For example, if you do not want to index the record type Email\_Rules, delete it from the list. To index only two fields, Name and Phone, for the record type Customer, then delete the other fields and keep these two.

- 3) Edit the file C:\CQ.Search\TextSearch\_SAMPL\cqtsadmin-TextSearch-SAMPL.xml and change the value for the XML tag <updateIndexDelay> from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue required="no">10</newValue>
```

This changes the default value of update-mode indexing delay from 10 minutes (600 seconds) to 10 seconds. This means it will take up to 10 seconds (not 10 minutes) for a change that you make to a ClearQuest record to appear in a full-text search result.

**Note:** Setting such a low value is not recommended for production use. Doing so will causes excessive logon and logoff calls to the ClearQuest database, consuming CPU, memory, and database resources.

- 4) Issue the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --setup_cq_fts
```

- 5) You receive instructions that must be performed on a Windows host computer. The instructions, in the file: /CQ.Search/TextSearch\_SAMPL/EnablingCQWebFTS.txt, are:

- a. On Windows, create the following directory: C:\CQ.Search\TextSearch\_SAMPL\
- b. Copy the following two files to the newly created Windows directory: /CQ.Search/TextSearch\_SAMPL/CQ-TextSearch-SAMPL.xml and /CQ.Search/TextSearch\_SAMPL/cqtsadmin-TextSearch-SAMPL.xml
- c. From your Windows system, run the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome /CQ.Search --
enable_cqweb_fts
```

- 6) Back on the UNIX or Linux computer, rerun step 4 to complete your deployment:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome /CQ.Search --setup_cq_fts
```

**Note:** ClearQuest Web client users can use full-text search during batch indexing, but results will not be complete until batch indexing is finished



- 7) Once the above command completes, based on your UNIX / Linux daemon requirement, make sure the following command is run as part of your daemon startup:

```
/opt/ibm/RationalSDLC/common/CM/profiles/cqsearchprofile/bin/
startServer.sh
```

This will ensure that both Solr and the record indexer will automatically start-up on a system reboot.

**Note:** When you are deploying ClearQuest full-text search for additional user databases, the value for **cqsearchprofile** will be the name of your WAS profile for that user database.

## I want to enable full-text search on user databases in a Windows production environment

This scenario is the same as the first scenario with the addition of pre-deployment planning and post-deployment steps.

Since this scenario involves planning, the planning time is based on the complexity of your ClearQuest schema and your organization policy. When you issue the command **setup\_cq\_fts** to complete the deployment, it may take few hours to a few days to complete, depending on the size and number of records in your ClearQuest database.

### Pre-deployment planning:

To ensure a successful full-text search deployment, plan your deployment by taking the following steps:

- 1) Ensure that you have enough free disk space on the **ftshome** destination. This free disk space is needed for the full-text search index. There is no rule to find out exactly how much free disk space you need. However, here are some guidelines:
  - a) If your ClearQuest database has 1 million records (excluding stateless “history” records), you need 7 GB of free disk space.
  - b) If your ClearQuest database has 8 million records (excluding stateless “history” records) you need 16 GB of free disk space.

The index will not be 7 or 16 GB in size; it will be half of this size. You need to double the index size of free disk space, otherwise index optimization fails.

- 2) Start deployment at off-peak hours, such as over a weekend. The initial batch indexing reads all record types that you set for full-text searching, and this activity might affect ClearQuest performance. Just like with the index size, rule to determine exactly how long initial batch indexing will take. However, here are some guidelines:
  - a) If your ClearQuest database has 1 million records (excluding stateless “history” records) expect the initial batch indexing to take 1 to 1 1/2 days.
  - b) If your ClearQuest database has 8 million records (excluding stateless “history” records) expect the initial batch indexing to take 2 to 4 days.
- 3) After running the command **init\_cq\_fts**, study your entity file, and decide which record types and fields to set for searching and the display field for full-text search hits.
- 4) Start the final step of your deployment by running the command **setup\_cq\_fts**.

### Post-deployment:

After your deployment completes, read the file `C:\CQ.Search\dbset_userdb\AboutThisFTS.txt`. The two important steps in this file that you need to follow are:

- 1) Secure your password key file using appropriate access control lists (ACLs) so that only administrators, local system, and the full-text search WAS service user account can reach the file. The file to secure is: `C:\CQ.Search\TextSearch_SAMPL\key.txt`

- 2) Secure your full-text search WAS profile (Solr search service) using directions in the Information Center under the section [Configuring and maintaining full-text search](#).

All CM servers and ClearQuest Web hosts that access this full-text-search-indexed ClearQuest user database with ClearQuest Web 7.1.1.x must be able to reach the Solr search service hosted by this WAS profile. Any administrative host that needs to access the Solr admin console must also be able to reach this host. If you use ClearQuest replication to share a specific ClearQuest full-text search server host, the remote CM server and ClearQuest Web server hosts must also be permitted access. For best results use IP addresses, optionally host names, or consult WebSphere documentation, the ClearQuest Information Center, or applicable technotes.

## I want to enable full-text search on user databases in a UNIX or Linux production environment

This scenario is similar to the previous scenario. The only difference is that the ClearQuest tool **installutil.exe** is not available on UNIX or Linux. Therefore, when you issue the command **setup\_cq\_fts**, you are given instructions to complete your deployment on a Windows computer. It is required that the same connection profile on UNIX or Linux be available on your Windows computer before you start the deployment.

ClearQuest full-text search must be installed on the Windows computer, and the IBM Rational ClearQuest 7.1.1.x full-text search administrator tool must also be present.

## I want to change the display field for already-indexed record types

This scenario assumes that you are using ClearQuest full-text search in a production environment. You now want to change, for one or more record types, the display field in the full-text search results.

This scenario should take about 5 minutes to complete.

- 1) Edit your entity file `C:\CQ.Search\TextSearch_SAMPL\Entity-TextSearch-SAMPL.txt`. For each record type that you want to change, locate the record. Move the ampersand (“&”) from the old field name to the new field name. A record type can have only one display field.

- 2) At the command prompt, issue the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --
update_fts_prop_files --enable_cqweb_fts
```

This runs the two commands in the order listed. First the ClearQuest Search Properties XML file is updated. Then the ClearQuest Web GUI receives the updated properties XML file and starts serving the new field.

For ClearQuest Web users to see the change, they need to log off ClearQuest Web and then log on again.

**Note:** Because the ClearQuest utility **installutil.exe** is not available on the UNIX or Linux operating systems, the command **enable\_cqweb\_fts** will instruct you to follow steps on a Windows computer. This requires that you have the full-text search component installed on Windows and the same ClearQuest connection profile created on Windows.

## I want to enable full-text search on additional user databases

This scenario assumes that you have a ClearQuest database connection called **Marvel**, with a ClearQuest user database name called **XMEN**, and you want to deploy full-text search on a second ClearQuest database. The process is identical to previous scenarios with one extra step that you need to take during full-text search initialization.

This scenario should take about 10 minutes to complete when used on any ClearQuest database.

- 1) At the command prompt, issue the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset Marvel
--userdb XMEN --ftshome C:\CQ.Search --init_cq_fts --
create_fts_was_profile automatic
```

This runs the standard **init\_cq\_fts** command, followed by the **create\_fts\_was\_profile** command, which creates a new WAS profile called **cqsearch\_Marvel\_XMEN**. Configuration data and files relevant to this deployment are placed in **C:\CQ.Search\Marvel\_XMEN\**.

On Windows systems, the **automatic** parameter causes the Windows service to be automatically started when the system is restarted. On UNIX or Linux systems, this parameter is ignored.

After this command completes, all subsequent commands run on this deployment are consistent, as if run on any user database. Thus, they affect this deployment only as long as the ClearQuest database set **Marvel** and the user database **XMEN** are specified on the command line.

## I want to add searchable indexed record types to my deployment

This scenario assumes that you have deployed ClearQuest full-text search and that you left out one or more record types during deployment or that you added new record types to your ClearQuest schema after deployment and need to be able to search on these new record types.

This scenario requires some planning. For each record type you are adding, you need to decide which fields to index and which indexed field will be the display type. Once planning is done, the steps should take about 5 minutes. However the command to add the record types may take from a few minutes to a few days to complete, depending on the size and number of records that are affected.

- 1) Edit the file **cqtsadmin-TextSearch-SAMPL.xml** and change the value for the XML tag **<addRecordType>** from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue required="no">Wolverine=Affiliations,AlterEgo,
&Name,Species,Team</newValue>
```

Notice that **&amp;** was added to the front of the field name **Name**. This is the XML encoding of the ampersand character (“&”). and is how you select the display field.

**Note:** Use a semicolon (“;”) to add multiple record types. For example:

```
<newValue
required="no">Wolverine=Affiliations,AlterEgo, &Name,Species,T
eam;Storm=Affiliations,AlterEgo, &Name,Species,Team</newValue>
```

- 2) At the command prompt, issue the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --add_record_type
```

The **add\_record\_type** command consists of several commands and it is stateful. If you stop this command or it stops due to an error, when you rerun it, it starts where it left off and continues to completion.

- 3) Edit the file **cqtsadmin-TextSearch-SAMPL.xml** and remove the record types that you added to the XML tag **<addRecordType>** in step 1.

ClearQuest Web users must log off of ClearQuest and then log on again to see the newly added record types.

## I want to remove indexed record types so that they are no longer searchable

This scenario assumes that you have deployed ClearQuest full-text search and you need to remove one or more indexed record types from your search index.

This scenario should take about 10 minutes to complete when used on any ClearQuest database.

Assume you have indexed the record type `Wolverine` and now you need to remove it from the index.

- 1) Edit the file `cqtsadmin-TextSearch-SAMPL.xml` and change the value for the XML tag `<removeRecordType>` from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue required="no">Wolverine</newValue>
```

**Note:** Use a semicolon (“;”) to remove multiple record types. For example:

```
<newValue required="no">Wolverine;Storm</newValue>
```

- 2) At the command prompt, run the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --
remove_record_type
```

The **remove\_record\_type** command consists of several commands and it is stateful. If you stop this command or it stops due to an error, when you rerun it, it starts where it left off and continues to completion.

- 3) Edit the file `cqtsadmin-TextSearch-SAMPL.xml` and remove the record types that you added to the XML tag `<removeRecordType>` in step 1.

ClearQuest Web users must log off of ClearQuest and then log on again to see the newly added record types.

## I want to add nonindexed fields to indexed record types

This scenario assumes that you have deployed ClearQuest full-text search and you need to add one or more fields to record types that were not originally indexed. This might be because of changes made to your ClearQuest schema (new fields were added to record types), or simply because the fields you need to add were not originally indexed when you deployed full-text search.

This scenario should take from a few hours to few days to complete, and it requires some planning. The steps will take about 5 minutes. However command to actually remove the record type may take as little as few minutes to as long as few days, depending on the size and number of records that are affected.

Assume that you have indexed the record type `Wolverine` without the `Abilities` field, and now you need to add it.

- 1) Edit the file `cqtsadmin-TextSearch-SAMPL.xml` and change the value for the XML tag `<removeRecordType>` from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue required="no">Wolverine</newValue>
```

Use a semicolon (“;”) to add multiple record types. For example:

```
<newValue required="no">Wolverine;Storm</newValue>
```

- 2) Edit the file `cqtsadmin-TextSearch-SAMPL.xml` and change the value for the XML tag `<addRecordType>` from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue
```

```
required="no">Wolverine=Abilities,Affiliations,AlterEgo,&Name
,Species,Team</newValue>
```

Use a semicolon (“;”) to separate multiple fields.

- 3) At the command prompt, issue the following command, which first removes the record type and then re-adds it with the new field:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --
remove_record_type --add_record_type
```

The **remove\_record\_type** command consists of several commands and it is stateful. If you stop this command or it stops due to an error, when you rerun it, it starts where it left off and continues to completion.

- 4) Edit the file `cqtsadmin-TextSearch-SAMPL.xml` and remove the record types that you added to the XML tags `<removeRecordType>` and `<addRecordType>`.

## I want to remove indexed fields from indexed record types

This scenario assumes that you have deployed ClearQuest full-text search and you need to remove one or more indexed fields from indexed record types. You might have changed your ClearQuest schema, or you might no longer want to allow searching on these fields per your organization policy.

This scenario should take anywhere from a few hours to a few days to complete, and it requires some planning. The steps will take about 5 minutes. However the command to remove and re-add the record types may take as little as few minutes to as long as few days, depend on the size and number of records that are affected.

Assume that you want to remove the field called `Abilities` from the record type `Wolverine` so that it is no longer searchable.

- 1) Edit the file `cqtsadmin-TextSearch-SAMPL.xml`, and change the value for the XML tag `<removeRecordType>` from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue required="no">Wolverine</newValue>
```

If you are removing multiple record types, use a semicolon (“;”) to separate them.

- 2) Edit the file `cqtsadmin-TextSearch-SAMPL.xml` and change the value for the XML tag `<addRecordType>` from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue
required="no">Wolverine=Affiliations,AlterEgo,&Name,Species,T
eam</newValue>
```

The goal is to list the original fields but to exclude the field `Abilities` so that it is no longer indexed.

- 3) At the command prompt, issue the following command, which first removes the record type and then re-adds it with the new field:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --
remove_record_type --add_record_type
```

The **remove\_record\_type** command consists of several commands and it is stateful. If you stop this

command or it stops due to an error, when you rerun it, it starts where it left off and continues to completion.

- 4) Edit the file `cqtsadmin-TextSearch-SAMPL.xml` and remove the record types that you added to the XML tag `<removeRecordType>` and `<addRecordType>`.

## I want to enable full-text search on a user database utilizing load balanced CM Server

This scenario is for an environment that includes load-balanced change management (CM) servers. When using IBM Rational ClearQuest 7.1.1.x full-text search administrator tool, full-text search works on a load balanced setup of ClearQuest Web without any administrative intervention or special setup. This is enabled by the value in the XML tag `<ftsServerName>`, which is set in the full-text search properties XML file. This XML file instructs all load-balanced CM servers how and where to find and communicate with the full-text search server.

## I want to set up a full-text search server at a MultiSite environment

This scenario is for a ClearQuest MultiSite environment. You want to enable ClearQuest full-text search for all sites by deploying ClearQuest full-text search service process at one site. This scenario is no different from the previous scenarios, except that only one site is used to index all records, as they are updated at that site with MultiSite synchronization. After the full-text search properties XML file is sent to ClearQuest's "master\_properties" table (with the command-line option **enable\_cqweb\_fts**), replication oplogs make this data available to all sites. At each replicated site, once the CM Servers see the data in their local `master_properties`, the **Full Text** radio button is enabled and all full-text search requests are sent to the full-text search server identified by the XML tag `<ftsServerName>` (found in the full-text search properties XML file).

To learn more about how replication, oplogs, and full-text search work in a MultiSite environment, see the ClearQuest Information Center.

NOTE: This scenario is similar to the load balancing scenario, except more than one site is used in a MultiSite environment.

## I want to upgrade or reinstall ClearQuest, which has full-text search WAS profiles for multiple user databases

This scenario assumes that you will upgrade or reinstall ClearQuest at some point, and that you have created two or more full-text search WAS profiles on a single server host. Because the Installation Manager is not aware of the extra full-text search WAS profiles, if you attempt to upgrade or reinstall ClearQuest, you might lose your full-text search WAS profiles (but not your full-text search configuration and index). Before you attempt to upgrade or reinstall ClearQuest, back up and delete all full-text search WAS profiles that you created. You restore these profiles after upgrading or reinstalling.

This scenario backs up and then deletes all custom deployed full-text search WAS profiles on the current server (except **cqsearchprofile**, which is managed by the Installation Manager). This scenario can be run from any full-text search deployment **ftshome**, **dbset** or **userdb**.

The process has two parts. Each part can take from a few minutes to a few hours to complete, depending on how many full-text search WAS profiles that you have. Typically, for each full-text search WAS profile, it takes about 5 minutes to complete each phase.

- 1) Edit the file `cqtsadmin-TextSearch-SAMPL.xml` and change the value for the XML tag `<deleteFtsWASProfiles>` from:

```
<newValue required="no"></newValue>
```

to:

```
<newValue required="no">true</newValue>
```

- 2) At the command prompt, issue the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset Marvel
--userdb XMEN --ftshome C:\CQ.Search --prep_upgd_was_profiles E:
\FTSBackupWASProfiles
```

This command-line option creates a backup of your WAS profiles data and deletes (un-deploys) your full-text search WAS profiles. The backup data is saved in the directory FTSBackupWASProfiles.

- 3) Edit the file cqtsadmin-TextSearch-SAMPL.xml and change the value for the XML tag <deleteFtsWASProfiles> from:

```
<newValue required="no">true</newValue>
```

to:

```
<newValue required="no"></newValue>
```

This prevents accidental re-execution of this destructive command-line option.

- 4) After completing your ClearQuest upgrade or reinstallation, at the command prompt, issue the following command to restore previously deployed and backed-up full-text search WAS profiles:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset Marvel
--userdb XMEN --ftshome C:\CQ.Search --restore_was_profiles E:
\FTSBackupWASProfiles
```

Perform this **restore\_was\_profiles** step if you are rolling back your upgrade of ClearQuest, just as you would with a newly upgraded or installed ClearQuest product (for example, if the upgrade is unsuccessful or undesirable).

## I want to enable full-text search on a non-English user database in a production environment

This scenario is the same as deploying ClearQuest full-text search when the ClearQuest database is in English. The only difference is that after the command-line option **init\_cq\_fts** completes, you edit your *schema.xml* file and set your language as documented in part 4 “Customize to use different languages” of developerWorks.

- 1) At the command prompt, run the following command:

```
cqperl cqtsadmin.pl --username admin --password "" --dbset
TextSearch --userdb SAMPL --ftshome C:\CQ.Search --init_cq_fts
```

- 2) Based on the instructions in developerWorks part 4, edit the file C:\CQ.Search\TextSearch\_SAMPL\Solr\solr\config\schema.xml so that the language setting matches the language of your ClearQuest database. For example, if the native language of your ClearQuest database is in Chinese, edit schema.xml to use the Chinese analyzer. If it is in Spanish, edit schema.xml to use the Spanish analyzer.
- 3) Continue your deployment, and usage of all other command-line options per this documentation.