

Quest® Coexistence Manager™ for
GroupWise 1.7

FBC Configuration Guide



© 2017 Quest Software Inc.

ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software Inc.

The information in this document is provided in connection with Quest Software products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest Software does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.
Attn: LEGAL Dept.
4 Polaris Way
Aliso Viejo, CA 92656

Refer to our website (<https://www.quest.com>) for regional and international office information.

Patents

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at <https://www.quest.com/legal>.

Trademarks

Quest, the Quest logo, and Join the Innovation are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit <https://www.quest.com/legal/trademark-information.aspx>. All other trademarks and registered trademarks are property of their respective owners.

Legend

 **CAUTION:** A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

 **IMPORTANT NOTE, NOTE, TIP, MOBILE, or VIDEO:** An information icon indicates supporting information.

Contents

About the CMG documentation suite	4
About CMG Program Parameters	5
Introduction to CMG parameters	5
Parameter format and structure	5
How to specify or edit program parameters	7
Directory Connector parameters	8
Mail Connector parameters	14
About us	15
We are more than just a name	15
Our brand, our vision. Together.	15
Contacting Quest	15
Technical support resources	15
Index	16

About the CMG documentation suite

The documentation for Quest Quest Coexistence Manager for GroupWise (CMG) includes:

- **Release Notes** (printable PDF): Describes the current CMG release—any new and enhanced features, resolved issues, and known issues. Also documents minimum and recommended installation requirements, and provides Quest contact information.
- **Quick-Start Guide** (printable PDF): An orientation to the product's basic purposes, features and capabilities, with brief case studies showing how its primary components are most commonly used within a typical coexistence scenario. Also summarizes deployment considerations, and explains how to download and install the software.
- **CMG User Guide** (printable PDF): Overview of features and general deployment guidelines for CMG's Directory Connector, Mail Connector and Free/Busy Connector, with screen-by-screen field notes for the Management Console application used to configure all three components. The *User Guide* also provides process instructions for installing and configuring the Directory Connector and Mail Connector, and for configuring the GroupWise and Exchange/AD environments to work with these components. (The same information for the F/B Connector is deferred to a separate *FBC Configuration Guide*.)
- **CMG FBC Configuration Guide** (printable PDF): Process instructions and application notes for installing and configuring CMG's Free/Busy Connector (FBC). An introductory chapter is followed by 12 chapters that each describe the complete process for installing and configuring the FBC for a particular combination of GroupWise environment and Exchange environment, for either a single- or multi-namespace environment.
- **CMG FBC Configuration Guide** (printable PDF): Process instructions and application notes for installing and configuring CMG's Free/Busy Connector (FBC). An introductory chapter is followed by 12 chapters that each describe the complete process for installing and configuring the FBC for a particular combination of GroupWise environment and Exchange environment, for either a single- or multi-namespace environment.
- **CMG Program Parameters Reference** (printable PDF): Listing of all CMG program parameters that can be used to configure CMG component behavior to best suit a particular environment and facilitate local preferences.
- **Management Console Online Help** (three compiled Windows Help files, one for each CMG component): Instructions and application notes for the various screens and features of CMG's Management Console utility.

This table shows where you can find particular types of information about particular CMG components:

	for Dir Connector & Mail Connector	for Free/Busy Connector
Introduction and orientation:	— — CMG Quick-Start Guide and User Guide — —	
Installation instructions:	— — CMG Quick-Start Guide — —	
Configuration instructions:	CMG User Guide	CMG FBC Configuration Guide
Operating instructions:	— — CMG User Guide — —	
Troubleshooting info:	CMG User Guide	CMG FBC Configuration Guide

The CMG application Help files contain the same information as the *User Guide*, but make the information available on-screen at the push of a button (from the CMG Management Console).

All CMG documentation is intended for network administrators, consultants, analysts, and any other IT professionals who will install or use the product components, or who may help plan for their use in a coexistence scenario. All of these documents, including the online Help, are bundled and installed with the product, and all except the Help files are also available separately at Quest's [Support Portal](#).

About CMG Program Parameters

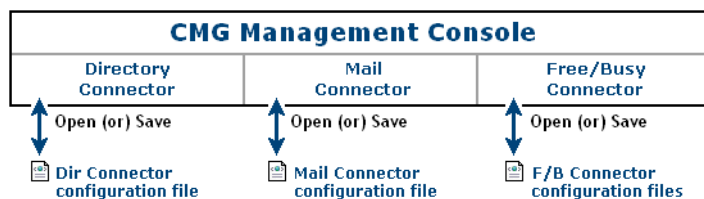
- [Introduction to CMG parameters](#)
- [Parameter format and structure](#)
- [How to specify or edit program parameters](#)

Introduction to CMG parameters

CMG's Directory Connector, Mail Connector and Free/Busy Connector are configurable, and most of their feature options are set in CMG's Management Console. The Management Console is a friendly interface between you and a set of configuration files that CMG components read every time they are started.

Configuration settings for CMG's different components are saved in separate configuration files: a *Configuration.xml* file for each of the Directory Connector and Mail Connector (two files with the same name, but saved in the different CMG component subfolders), and another set of configuration files for the Free/Busy Connector (in the FBC's component subfolder). The Management Console provides GUI screens with labeled fields to simplify entering and editing the configuration settings in those files.

When you start the Management Console, CMG automatically loads the currently active configuration values for all three CMG components. Once the Management Console is running, however, component configuration data is saved and opened separately, for one component at a time, as shown here:



While the Management Console lets you enter and change most CMG feature options, some options can be set and changed only by manually editing the corresponding *Configuration.xml* file. This *FBC Configuration Guide* documents those parameters.

NOTE: All FBC program parameters in the current release are associated with UI fields in the FBC Management Console. As such, these parameters are not documented in this *Reference*, since they should not be edited manually (in the xml files) except at the direction of Quest Support.

The contents of a configuration XML file can be edited by any XML editor, as described separately below.

Parameter format and structure

CMG parameters are specified and stored in XML Unicode text files that conform to XML markup standards and conventions. Each parameter is specified by an XML element, consisting of a start tag, an end tag, and the parameter value between the two tags. For example:

```
<ServicePort>1234</ServicePort>
```

As with all XML markup tags (and as shown in these examples), the start and end tags are both enclosed within angle brackets, and the end tag is denoted by a slash character ("/") inside the angle brackets preceding the parameter name.

Within a CMG parameters XML file, related parameters are grouped into sections, which also are named and marked by pairs of tags. For example:

```
<ConnectorSettingsList>
  <ConnectorSetting>
    <TargetPort>389</TargetPort>
    <TargetUseSsl>>false</TargetUseSsl>
    <SourceOu>DC=mustang,DC=wingra,DC=com</SourceOu>
    <TargetOu>OU=abcdefg,O=wingra</TargetOu>
    <Groups>>true</Groups>
    <Users>>false</Users>
    <Contacts>>false</Contacts>
    <SourceGroupObject>msExchDynamicDistribList</SourceGroupObject>
  </ConnectorSetting>
  <AnotherSection>
    <Parameter241>blue</Parameter241>
    <Parameter242>654321</Parameter242>
    <SomeOtherSection>
      <Parameter243>>true</Parameter243>
      <Parameter244>kangaroo</Parameter244>
      <Parameter245>Casablanca</Parameter245>
    </SomeOtherSection>
  </AnotherSection>
  <Parameter246>>false</Parameter246>
</ConnectorSettingsList>
```

The section tags and parameter elements within an XML file are typically broken into separate lines and indented, as shown here, just for human readability. But the line breaks, indentions and any other white space that occurs between elements are ignored by the programs that read the files.

Spaces that occur within a parameter *value*, however, *are* significant. For example, the spacing within a text string such as:

```
<StreetAddress>123 45th Avenue</StreetAddress>
```

... will be read, interpreted and used by the program in exactly that same form, with spaces intact.

Likewise, upper/lowercase within a parameter value may be significant, but only if case matters in whatever context the program will use the string. For example, if `<CityName>baLtlmOre</CityName>` and the program will copy that parameter value into a program log or report, or into the user interface as a display element, the name will appear there in that same peculiar form: as "baLtlmOre." Section and parameter names (tag contents), however, are case-*ins*sensitive.

Spelling of parameter names and parameter values is always critical. If a program is looking for a parameter name *UserAddress*, it will not recognize a parameter name *UsreAddress* or *UserAdress*.

Parameter defaults

The default value of a parameter is the value that will prevail in the program if the parameter is omitted from its *Configuration.xml* file.

Parameter value data types

For any given parameter, the parameter value will be meaningful only if it conforms to the parameter's particular data type. Every parameter value is one of these five data types:

- **String:** An alphanumeric string.
- **Integer:** A non-decimal numeric value.
- **Boolean:** A value of "true" or "false" to specify whether a particular feature or function is to be enabled or disabled, or whether a particular operating condition is true or false.
- **Date:** A date, formed by eight digits with two slash characters, in the form: mm/dd/yyyy.
- **Keyword:** An alphanumeric string that matches one of a finite set of valid keyword strings defined for the particular parameter.

Parameters associated with UI fields in CMG's Management Console

Many parameters that appear in the CMG xml configuration files simply store values that are entered or selected in the fields of various screens in CMG's Management Console software. These UI-associated parameters are therefore not documented in this *Reference*, since their values should not be manually edited in the xml files except at the direction of Quest Support.

How to specify or edit program parameters

IMPORTANT: The CMG Management Console must be closed before you open a configuration file as described here. When you select **File|Save** in the Management Console, the Console saves all of its open data to the *Configuration.xml* file, and this will overwrite any manual changes to the file that may have been entered and saved while the Console was open.

CAUTION: Be very careful when manually entering or editing values in the *Configuration.xml* file, where incorrect syntax or even a single-character typo could disable a connector or the DC service altogether.

The contents of a configuration XML file can be edited by any XML editor:

- 1 Use an XML editor to open CMG's active XML configuration file. Be sure to open the correct XML file, for the CMG component whose parameter(s) you want to enter or change:

Configuration.XML file for Directory Connector is in: CMG's \Directory Connector\ folder

Configuration.XML file for Mail Connector is in: CMG's \Mail Connector\ folder

The configuration files for the Free/Busy Connector are not listed here because all FBC program parameters in the current release are associated with UI fields in the FBC Management Console. As such, these parameters should not be edited manually (in the xml files) except at the direction of Quest Support.

- 2 Find the parameter within the file. You can text-search for the string of the parameter name, or for the <Section> where the parameter resides.

If the parameter is not already in the file, add it in a new line. In this case, be sure that what you enter matches exactly the parameter name documented in this *Reference*.

- 3 Enter or change the value of the parameter. Make sure the value is valid by value type, range (if appropriate), and syntax as documented in this *Reference*.
- 4 Save and close the *Configuration.XML* file.

Directory Connector parameters

i | **NOTE:** Parameters that appear in the *DC Configuration.xml* file but that are not listed here are associated with UI entries in the DC Management Console, and should not be edited manually (in the xml file) except at the direction of Quest Support.

<AppendAttributes>

Type: string. **Default:** [null] (connector will overwrite target attributes with source attributes if they differ)

Specifies a list of one or more target object attributes whose values will be *appended to*, rather than *replaced by* source attribute data, when the connector is running in *Update* mode and attribute data differs between an existing object in the target and a corresponding source object. Available for either G-to-E or E-to-G connector, specified separately per connector.

This parameter is optional, and is configured separately for each DC connector in the connector's section of the *DC Configuration.xml* file. To designate the attribute(s) that should function this way in *Update* mode for a particular connector, find (or add) the *<AppendAttributes>* tags for the connector, and enter the names of the target attributes separated by a pipe character ("|") between the tags, like this:

```
<AppendAttributes>cn|mail|proxyaddresses</AppendAttributes>
```

Note that a manual change to an attribute value in a CMG target container will persist through subsequent connector runs if the corresponding source object record has not also been changed. A connector compares the source object data to a copy of the source object record from the preceding connector run, and updates the target object only if the source object has changed.

<DoMultipleNovellLogins>

Type: boolean. **Default:** false (off)

Determines whether a G-to-E connector will present to the user a single Novell login question or a page of multiple Novell login credentials. Available for G-to-E connector only, specified separately per connector.

<EnableQcServiceRestart>

Type: boolean. **Default:** false (off)

Determines whether CMG will restart the QcService (the DC "engine") if it encounters an error during a connector synchronization. Applies to the Directory Connector as a whole (not just a particular connector).

<GwUserAttributeList>

Type: string. **Default:** [null] (connector will copy *all* attributes for user objects)

Determines which attributes a GroupWise-to-Exchange connector will capture and copy for user objects. Available for a G-to-E connector only, specified separately per connector. This is one of three DC parameters that do the same thing for the three object types:

```
<GwUserAttributeList>
<GwGroupAttributeList>
<GwResourceAttributeList>
```


These *GwXxxxAttributeList* parameters are unspecified by default, so the connector will copy all attributes. But if particular attributes (one or more) are unnecessary for a particular object type, these parameters let you specify which attributes to include for a particular connector. The attribute(s) *not* listed will be **excluded**.

This feature can improve connector performance, perhaps dramatically, in an environment with slow response times and long distribution lists. The feature can be configured only by these *Configuration.xml* parameters (not in the Connector Creation Wizard), and is available only for existing connectors in the G-to-E direction. The feature is applied per connector, in the *<ConnectorSetting>* section for a particular connector.

For each object type, enter the parameter name with a list of all attributes to be included, all in one continuous line and separated by pipe characters ("|"), with no spaces between. For example:

```
<GwUserAttributeList>UserAttr1|UserAttr2|UserAttr3|UserAttr4</GwUserAttributeList>
```

To create a suitable list of attributes for the parameter value:

- 1 Open the file ...*\Directory Connector\attributes\groupwiseConnector_XXX*, where *XXX* is the object type (user or group or resource). The attributes in this file appear listed one attribute per line.
- 2 Copy all the attributes from the file into a text editor. In the text editor, delete the lines for attributes you want to **exclude** from the connector's operations.
- 3 For the remaining attributes in the list (the ones you want to **include**), replace all the hard returns with pipe characters ("|"), to produce a single-line continuous list (no spaces) of attributes separated by pipe characters.

Then you can insert that list as the parameter value in the DC *Configuration.xml* file.

<GwGroupAttributeList>

Type: string. **Default:** [null] (connector will copy **all** attributes for group objects)

Determines which attributes a GroupWise-to-Exchange connector will capture and copy for group objects. Available for G-to-E connector only, specified separately per connector. This is one of three DC parameters that do the same thing for the three object types:

```
<GwUserAttributeList>  
<GwGroupAttributeList>  
<GwResourceAttributeList>
```

These *GwXxxxAttributeList* parameters are unspecified by default, so the connector will copy all attributes. But if particular attributes (one or more) are unnecessary for a particular object type, these parameters let you specify which attributes to include for a particular connector. The attribute(s) *not* listed will be **excluded**.

This feature can improve connector performance, perhaps dramatically, in an environment with slow response times and long distribution lists. The feature can be configured only by these *Configuration.xml* parameters (not in the Connector Creation Wizard), and is available only for existing connectors in the G-to-E direction. The feature is applied per connector, in the *<ConnectorSetting>* section for a particular connector.

For each object type, enter the parameter name with a list of all attributes to be included, all in one continuous line and separated by pipe characters ("|"), with no spaces between. For example:

```
<GwGroupAttributeList>GrpAttr1|GrpAttr2|GrpAttr3|GrpAttr4</GwGroupAttributeList>
```

To create a suitable list of attributes for the parameter value:

- 1 Open the file ...*\Directory Connector\attributes\groupwiseConnector_XXX*, where *XXX* is the object type (user or group or resource). The attributes in this file appear listed one attribute per line.
- 2 Copy all the attributes from the file into a text editor. In the text editor, delete the lines for attributes you want to **exclude** from the connector's operations.
- 3 For the remaining attributes in the list (the ones you want to **include**), replace all the hard returns with pipe characters ("|"), to produce a single-line continuous list (no spaces) of attributes separated by pipe characters.

Then you can insert that list as the parameter value in the DC *Configuration.xml* file.

<GwResourceAttributeList>

Type: string. **Default:** [none] (connector will copy *all* attributes for resource objects)

Determines which attributes a GroupWise-to-Exchange connector will capture and copy for resource objects. Available for G-to-E connector only, specified separately per connector. This is one of three DC parameters that do the same thing for the three object types:

```
<GwUserAttributeList>
<GwGroupAttributeList>
<GwResourceAttributeList>
```

These *GwXxxxAttributeList* parameters are unspecified by default, so the connector will copy all attributes. But if particular attributes (one or more) are unnecessary for a particular object type, these parameters let you specify which attributes to include for a particular connector. The attribute(s) *not* listed will be **excluded**.

This feature can improve connector performance, perhaps dramatically, in an environment with slow response times and long distribution lists. The feature can be configured only by these *Configuration.xml* parameters (not in the Connector Creation Wizard), and is available only for existing connectors in the G-to-E direction. The feature is applied per connector, in the *<ConnectorSetting>* section for a particular connector.

For each object type, enter the parameter name with a list of all attributes to be included, all in one continuous line and separated by pipe characters ("|"), with no spaces between. For example:

```
<GwResourceAttributeList>RsrcAttr1|RsrcAttr2|RsrcAttr3|RsrcAttr4</GwResourceAttributeList>
```

To create a suitable list of attributes for the parameter value:

- 1 Open the file *...\\Directory Connector\attributes\groupwiseConnector_XXX*, where *XXX* is the object type (user or group or resource). The attributes in this file appear listed one attribute per line.
- 2 Copy all the attributes from the file into a text editor. In the text editor, delete the lines for attributes you want to **exclude** from the connector's operations.
- 3 For the remaining attributes in the list (the ones you want to **include**), replace all the hard returns with pipe characters ("|"), to produce a single-line continuous list (no spaces) of attributes separated by pipe characters.

Then you can insert that list as the parameter value in the DC *Configuration.xml* file.

<IgnoreWizardOuCheck>

Type: boolean. **Default:** true (on)

Determines whether the Connector Creation Wizard will (*false*) or will not (*true*) verify data on pages searching for and displaying OUs (insofar as verifications are possible by other entries and/or environmental inferences). Applies to the Directory Connector as a whole (not just a particular connector). Note that the parameter is defined (named) in terms of *not* verifying, so a *true* setting turns *off* the verifications.

<LogAcceptorAddress>

Type: string. **Default:** 127.0.0.1 (local host)

Specifies the IP address to which CMG listens when allowing connections.

<LogAcceptorBanner>

Type: string. **Default:** QCService

Specifies the banner CMG inserts into CMG logs to identify data coming from the DC engine.

<LogAcceptorLineSplit>

Type: string. **Default:** QCService

Specifies the string CMG will insert into the DC engine logs to mark where CMG splits the line and logs everything to the left of the indicated string.

<LogAcceptorPort>

Type: integer. **Default:** 9091

Specifies the UDP port to which CMG listens when logging activity by the DC engine. Available for either G-to-E or E-to-G connector, specified separately per connector.

<ProcessPartialResults>

Type: bool. **Default:** false (off)

Determines whether a connector will continue operating if it encounters an error in its process. If enabled (true), the connector will continue to the next step.

<QcServiceHost>

Type: string. **Default:** 127.0.0.1 (local host)

Specifies an alternate location (a computer other than the local CMG admin host computer) where the QcService (the DC "engine") resides. Ordinarily the QcService is installed on the local machine, but this parameter permits redirection to a different QC installation. Applies to the Directory Connector as a whole (not just a particular connector).

<ServicePort>

Type: integer. **Default:** 9081

Specifies the communications port for the DC service. Available for the Directory Connector service overall (not specified separately per connector).

Specifies which communications port the Directory Connector should use for data transmissions between the DC service and the DC Management Console, on the server where these applications reside. The port for this is set to 9081 by default, and this will be the correct setting in almost all environments.

But if port 9081 is assigned to some other service or function in your environment, you must change this communications port setting in the Directory Connector configuration file. The <ServicePort> parameter is usually at or near the end of the <DirectoryConnectorSettings> section.

<SourceContactObject>

Type: string. **Default:** Contacts

Specifies an alternate object type (instead of *Contact* objects) in the source whose objects a connector should sync to the target—to a target object type specified by the <TargetContactObject> parameter described below. This feature permits the designation of a particular connector to synchronize a particular object type other than the default *Contact* object. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

<SourceGroupObject>

Type: string. **Default:** Groups

Specifies an alternate object type (instead of *Group* objects) in the source whose objects a connector should sync to the target—to a target object type specified by the <TargetGroupObject> parameter described below. This feature permits the designation of a particular connector to synchronize a particular object type other than the default *Group* object. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

<SourceResourceObject>

Type: string. **Default:** Resources

Specifies an alternate object type (instead of *Resource* objects) in the source whose objects a connector should sync to the target—to a target object type specified by the <TargetResourceObject> parameter described below. This feature permits the designation of a particular connector to synchronize a particular object type other than the default *Resource* object. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

<SourceUserObject>

Type: string. **Default:** Users

Specifies an alternate object type (instead of *User* objects) in the source whose objects a connector should sync to the target—to a target object type specified by the <TargetUserObject> parameter described below. This feature permits the designation of a particular connector to synchronize a particular object type other than the default *User* object. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

<SqlStatus...>

Type: integer. **Defaults:** 0, 0 and 0 (SQL execution history log file is emptied once each day, at midnight)

A set of three parameters that together configure the cleanup of SQL connector-execution logs. Available for DC overall (not specified separately per connector).

CMG deletes the contents of DC's SQL connector-execution histories at regular intervals. This feature is controlled by these parameters in the DC's *Configuration.xml* file:

```
<SqlStatusRetainDays>2</SqlStatusRetainDays>
<SqlStatusCleanupHour>23</SqlStatusCleanupHour>
<SqlStatusCleanupMinute>59</SqlStatusCleanupMinute>
```

The values in this example tell the Directory Connector to empty the history log file every other day (*SqlStatusRetainDays=2*) at 11:59pm local time (*SqlStatusCleanupHour=23*, *SqlStatusCleanupMinute=59*). The defaults (*SqlStatusRetainDays=0*, *SqlStatusCleanupHour=0*, *SqlStatusCleanupMinute=0*) tell the DC to delete the execution logs at midnight every day. These parameters usually appear at or near the end of the *Configuration.xml* file, in the <root> node, typically between <SvcPassword> and <LogAcceptorPort>.

<StichOus>

Type: bool. **Default:** true (on)

Determines whether the Connector Creation Wizard will combine multiple OU trees (in a multi-domain AD forest) into one for presentation in the Wizard's *Source Object Selection* screen. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

<TargetContactObject>

Type: string. **Default:** Contacts

Specifies an alternate object type (instead of *Contact* objects) in the target to be synchronized with objects from the source of the type specified by the <SourceContactObject> parameter described above. This feature permits the designation of a particular connector to synchronize a particular object type other than the default *Contact* object. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

<TargetGroupObject>

Type: string. **Default:** Groups

Specifies an alternate object type (instead of *Group* objects) in the target to be synchronized with objects from the source of the type specified by the <SourceGroupObject> parameter described above. This feature permits the designation of a particular connector to synchronize a particular object type other than the default *Group* object. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

<TargetResourceObject>

Type: string. **Default:** Resources

Specifies an alternate object type (instead of *Resource* objects) in the target to be synchronized with objects from the source of the type specified by the <SourceResourceObject> parameter described above. This feature permits the designation of a particular connector to synchronize a particular object type other than the default *Resource* object. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

<TargetUserObject>

Type: string. **Default:** Users

Specifies an alternate object type (instead of *User* objects) in the target to be synchronized with objects from the source of the type specified by the <SourceUserObject> parameter described above. This feature permits the designation of a particular connector to synchronize a particular object type other than the default *User* object. Available for either a G-to-E or E-to-G connector, specified separately per connector, in the connector's <Section> of the *Configuration.xml* file.

Mail Connector parameters

i | **NOTE:** Only a very few of the parameters that appear in the MC *Configuration.xml* file are documented here, because the others are associated with UI entries in the MC Management Console, and should not be edited manually (in the xml file) except at the direction of Quest Support.

<DisableProcessing>

Type: boolean. **Default:** false

Summary: Optional "pass-through" mode to disengage message processing.

Determines whether CMG's optional "pass-through" mode will disengage CMG's message-processing features while CMG continues to pass mail between the servers. This feature is off (*false*) by default, which tells the MC to **not** enable its pass-through mode—which would *disable* message processing. (A *false* setting here tells the MC to *continue* processing messages in addition to routing them to the other environment.) This feature is available only by manually setting this parameter in MC's *Configuration.xml* (is not offered in the Management Console).

<RouterTimeoutSec>

Type: integer. **Default:** 300 (seconds; = 5 minutes)

Summary: Duration of wait for a socket connection.

Sets the number of seconds CMG will wait for a socket connection before timing out.

<RouterWaitSec>

Type: integer. **Default:** 5

Summary: Configure socket channel usage.

This is a "throttling" feature to enforce a time interval for CMG to conclude its communications protocol with a channel before opening another. The actual delay, in seconds, is a "delay factor" (5 by default) x the ratio of the number of open channels to the maximum allowed number of channels. This parameter, in the <root> node of the MC's *Configuration.xml*, specifies the delay factor. By default, the Mail Connector waits up to 5 seconds for an available socket channel before opening a new one (if none are then available).

We are more than just a name

We are on a quest to make your information technology work harder for you. That is why we build community-driven software solutions that help you spend less time on IT administration and more time on business innovation. We help you modernize your data center, get you to the cloud quicker and provide the expertise, security and accessibility you need to grow your data-driven business. Combined with Quest's invitation to the global community to be a part of its innovation, and our firm commitment to ensuring customer satisfaction, we continue to deliver solutions that have a real impact on our customers today and leave a legacy we are proud of. We are challenging the status quo by transforming into a new software company. And as your partner, we work tirelessly to make sure your information technology is designed for you and by you. This is our mission, and we are in this together. Welcome to a new Quest. You are invited to Join the Innovation™.

Our brand, our vision. Together.

Our logo reflects our story: innovation, community and support. An important part of this story begins with the letter Q. It is a perfect circle, representing our commitment to technological precision and strength. The space in the Q itself symbolizes our need to add the missing piece—you—to the community, to the new Quest.

Contacting Quest

For sales or other inquiries, visit <https://www.quest.com/company/contact-us.aspx> or call +1-949-754-8000.

Technical support resources

Technical support is available to Quest customers with a valid maintenance contract and customers who have trial versions. You can access the Quest Support Portal at <https://support.quest.com>.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request.
- View Knowledge Base articles.
- Sign up for product notifications.
- Download software and technical documentation.
- View how-to-videos.
- Engage in community discussions.
- Chat with support engineers online.
- View services to assist you with your product.

A

AppendAttributes parameter, 8

C

CMG Mail Connector

 Pass-Through Mode, 14

communications port, Directory Connector, 11

configuration XML file, editing, 7

Configuration.xml files, 5

D

data types, 7

Directory Connector

 communications port, 11

DisableProcessing parameter, 14

DoMultipleNovellLogins parameter, 8

E

editing configuration XML files, 7

EnableQcServiceRestart parameter, 8

G

GwGroupAttributeList parameter, 9

GwResourceAttributeList parameter, 10

GwUserAttributeList parameter, 8

I

IgnoreWizardOuCheck parameter, 10

L

LogAcceptorAddress parameter, 10

LogAcceptorBanner parameter, 10

LogAcceptorLineSplit parameter, 10

LogAcceptorPort parameter, 11

M

Mail Connector

 Pass-Through Mode, 14

Management Console fields, parameters for, 5, 7, 8, 14

P

parameter defaults, 6

parameter format and structure, 5

parameter value data types, 7

parameters for Management Console fields, 5, 7, 8, 14

parameters for UI fields, 5, 7, 8, 14

ProcessPartialResults parameter, 11

program parameters, editing, 7

Q

QcServiceHost parameter, 11

R

RouterTimeoutSec parameter, 14

RouterWaitSec parameter, 14

S

ServicePort parameter, 11

SourceContactObject parameter, 11

SourceGroupObject parameter, 11

SourceResourceObject parameter, 12

SourceUserObject parameter, 12

SQL, cleaning up execution histories, 12

SqlStatus... parameters, 12

SqlStatusCleanupHour parameter, 12

SqlStatusCleanupMinute parameter, 12

SqlStatusRetainDays parameter, 12

StichOus parameter, 12

T

TargetContactObject parameter, 12

TargetGroupObject parameter, 13

TargetResourceObject parameter, 13

TargetUserObject parameter, 13

U

UI fields, parameters for, 5, 7, 8, 14

X

XML file, editing, 7