

Kofax Communications Manager

Installation Guide

Version: 5.3.0

Date: 2019-05-31

The logo for Kofax, consisting of the word "KOFAX" in a bold, blue, sans-serif font.

© 2019 Kofax. All rights reserved.

Kofax is a trademark of Kofax, Inc., registered in the U.S. and/or other countries. All other trademarks are the property of their respective owners. No part of this publication may be reproduced, stored, or transmitted in any form without the prior written permission of Kofax.

Table of Contents

Preface.....	5
Related documentation.....	5
Training.....	6
Getting help with Kofax products.....	7
Chapter 1: Before installation.....	8
Ports.....	8
Databases.....	9
Databases for KCM Batch & Output Management.....	9
Manage databases.....	10
License.....	11
Chapter 2: Install Kofax Communications Manager.....	13
Deployment parameters.....	13
Example deployment command.....	17
Additional deployment parameters for KCM B&OM.....	17
Install KCM Repository on an Oracle database.....	18
Oracle user.....	19
Predefined setup options.....	20
Install KCM Designer for Windows.....	20
Install KCM on more than one computer.....	20
Install KCM instances on a computer using remote Contract Manager.....	21
Install dedicated Contract Manager on a computer with no instances.....	21
Install KCM without default contracts.....	21
Install KCM without the Contract Manager.....	22
Install KCM B&OM on multiple computers.....	22
Chapter 3: Upgrade your installation.....	25
Perform the upgrade.....	25
Parameters for the upgrade.....	26
Perform the KCM B&OM upgrade.....	28
Additional upgrade parameters for KCM B&OM.....	29
Switch to an older installation.....	29
Upgrade a customized setup.....	30
Upgrade contract types and interfaces not in use by any contract.....	30
Upgrade KCM when installed on more than one computer.....	31
Chapter 4: Post-installation steps.....	32

Test the installation.....	32
Remove the test web application.....	32
Access product documentation.....	33
Offline documentation.....	33
Add extra instances after installation.....	34
Extra instances parameters.....	35
Add extra instances after installation in combination with Oracle.....	36
Remove specific instances.....	37
Parameters to remove specific instances.....	37
Add custom KCM B&OM service hosts.....	37
Add custom service hosts.....	38
List custom service hosts.....	39
Remove custom service hosts.....	40
Required step to run KCM B&OM processes.....	40
Export and import KCM B&OM content.....	40
Export KCM B&OM content.....	41
Import KCM B&OM content.....	42
Chapter 5: Other settings.....	43
Enable authentication and authorization for Contract Manager.....	43
Enable SSL for connections to the Contract Manager.....	43
Change an account for the WebServer services.....	45
Configure request rate limitation.....	45
Install and configure KCM Toolbox for Word.....	45
Chapter 6: Use of Microsoft Word by KCM Core.....	47
Adjust settings in Microsoft Word.....	47
Chapter 7: Uninstallation.....	48
Uninstall version 4.4.....	48
Chapter 8: Troubleshooting.....	49
Package cannot connect to database.....	49
Services cannot be created.....	49
Cannot start KCM Core for development or test license.....	49
Warning messages.....	49
Deployment log files are empty.....	50
CMRestAPI does not start during system startup.....	50

Preface

This guide contains instructions on how to install Kofax Communications Manager (also known as KCM). You can use the KCM package to deploy a single instance of the software.

Note For information on technical requirements for hardware, server operating systems, web servers, supported word processors, compatible software and more, see the *Technical Specifications* document on the Kofax Communications Manager support page on the Kofax website: www.kofax.com

Related documentation

The documentation set for Kofax Communications Manager is available here:¹

<https://docshield.kofax.com/Portal/Products/CCM/530-1h4cs6680a/CCM.htm>

In addition to this guide, the documentation set includes the following items:

Kofax Communications Manager Release Notes

Contains late-breaking details and other information that is not available in your other Kofax Communications Manager documentation.

Kofax Communications Manager Getting Started Guide

Describes how to use Contract Manager to manage instances of Kofax Communications Manager.

Help for Kofax Communications Manager Designer

Contains general information and instructions on using Kofax Communications Manager Designer, which is an authoring tool and content management system for Kofax Communications Manager.

Kofax Communications Manager Repository Administrator's Guide

Describes administrative and management tasks in Kofax Communications Manager Repository and Kofax Communications Manager Designer for Windows.

Kofax Communications Manager Repository User's Guide

Includes user instructions for Kofax Communications Manager Repository and Kofax Communications Manager Designer for Windows.

¹ You must be connected to the Internet to access the full documentation set online. For access without an Internet connection, see "Offline documentation."

Kofax Communications Manager Repository Developer's Guide

Describes various features and APIs to integrate with Kofax Communications Manager Repository and Kofax Communications Manager Designer for Windows.

Kofax Communications Manager Template Scripting Language Developer's Guide

Describes the KCM Template Script used in Master Templates.

Kofax Communications Manager Core Developer's Guide

Provides a general overview and integration information for Kofax Communications Manager Core.

Kofax Communications Manager Core Scripting Language Developer's Guide

Describes the KCM Core Script.

Kofax Communications Manager API Guide

Describes Contract Manager, which is the main entry point to Kofax Communications Manager.

Kofax Communications Manager ComposerUI for HTML5 JavaScript API Web Developer's Guide

Describes integration of ComposerUI for HTML5 into an application, using its JavaScript API.

Kofax Communications Manager Batch & Output Management Getting Started Guide

Describes how to start working with Batch & Output Management.

Kofax Communications Manager Batch & Output Management Developer's Guide

Describes the Batch & Output Management scripting language used in KCM Studio related scripts.

Kofax Communications Manager DID Developer's Guide

Provides information on the Database Interface Definitions (referred to as DIDs), which is an alternative method retrieve data from a database and send it to Kofax Communications Manager.

Kofax Communications Manager ComposerUI for ASP.NET and J2EE Customization Guide

Describes the customization options for KCM ComposerUI for ASP.NET and J2EE.

Kofax Communications Manager ComposerUI for ASP.NET Developer's Guide

Describes the structure and configuration of KCM ComposerUI for ASP.NET.

Kofax Communications Manager ComposerUI for J2EE Developer's Guide

Describes JSP pages and lists custom tugs defined by KCM ComposerUI for J2EE.

Training

To ensure your best experience in working with KCM Designer, complete our complimentary and interactive e-learning modules located at www.kofax.com.

Getting help with Kofax products

The [Kofax Knowledge Base](#) repository contains articles that are updated on a regular basis to keep you informed about Kofax products. We encourage you to use the Knowledge Base to obtain answers to your product questions.

To access the Kofax Knowledge Base, go to the Kofax [website](#) and select **Support** on the home page.

Note The Kofax Knowledge Base is optimized for use with Google Chrome, Mozilla Firefox or Microsoft Edge.

The Kofax Knowledge Base provides:

- Powerful search capabilities to help you quickly locate the information you need.
Type your search terms or phrase into the **Search** box, and then click the search icon.
- Product information, configuration details and documentation, including release news.
Scroll through the Kofax Knowledge Base home page to locate a product family. Then click a product family name to view a list of related articles. Please note that some product families require a valid Kofax Portal login to view related articles.
- Access to the Kofax Customer Portal (for eligible customers).
Click the **Customer Support** link at the top of the page, and then click **Log in to the Customer Portal**.
- Access to the Kofax Partner Portal (for eligible partners).
Click the **Partner Support** link at the top of the page, and then click **Log in to the Partner Portal**.
- Access to Kofax support commitments, lifecycle policies, electronic fulfillment details, and self-service tools.
Scroll to the **General Support** section, click **Support Details**, and then select the appropriate tab.

Chapter 1

Before installation

Ports

When deployed, Kofax Communications Manager uses the following ports.

Port	Description
80	In use by the KCM Content Management API service
443	In use by Tomcat for the Contract Manager for runtime purposes when using the SSL protocol
2587	In use by KCM Repository Server
3000-3003	In use by KCM Core
8007	In use by Tomcat. Shutdown port for KCM Designer instance
8009	In use by Tomcat for AJP connectivity
8080	In use by Tomcat for KCM ComposerUI for HTML5 and KCM ComposerUI for J2EE
8081	In use by Tomcat for the Contract Manager for runtime purposes when not using the SSL protocol
8180	In use by Tomcat for the LogServer to retrieve logs
8280	In use by Tomcat for KCM Designer

Note

- Inbound TCP/IP Port 8081 should be opened in the firewall and available for use by Kofax Communications Manager. All other ports mentioned above should not be exposed externally.
- KCM additional instances use incremental ports, apart from those specified above, for the KCM Repository Server and KCM Core. A new KCM Repository Server requires one additional port. A new KCM Core requires two additional ports.
- To prevent a Slow HTTP Denial of Service attack, you should set up your firewall to limit the amount of connections per host for port 8081. For example, you can limit the amount to 50 connections per host.

Databases

The deployment process uses two databases. The KCM package configures these databases during installation:

- One database for the Contract Manager (if installed)
- One database for KCM Repository

Note The installer does not create these databases automatically. You need to create empty databases manually. For SQL Server, create the databases with a case-insensitive collation.

The same DBMS and database settings are used for the Contract Manager and KCM Repository.

The names of the databases are partly configurable by using a prefix. The prefix cannot contain the colon and semicolon. The names should appear as shown here:

- Contract Manager: <Prefix>_CM_5.3.0
- KCM Repository: <Prefix>_01_5.3.0

The Contract Manager database is on SQL Server. The Repository database can be either on SQL Server or on Oracle. If both databases are run on SQL Server, they must be accessible by the same SQL Server account. The SQL Server account must have owner rights for these databases and use "dbo" schema. SQL Server must have both SQL Server and Windows Authentication mode enabled. Also, the TCP/IP protocol for SQL Server must be enabled.

Consider the following when adding KCM extra instances:

- KCM additional instances require their own KCM Repository database. The additional instances must follow the same naming conventions as the original instances, but you should increment the number. For example, a second instance requires the database <Prefix>_02_5.3.0.
- The content in the KCM Repository databases is not removed when an instance is uninstalled.

Databases for KCM Batch & Output Management

This section is only relevant if you are installing KCM Batch & Output Management along with the main product.

The Batch & Output Management deployment process requires two empty databases:

- One database for the B&OM Repository
- One database for the runtime information

You should create empty databases manually. They can be either SQL Server or Oracle, but must both be of the same DBMS. Also, you should create the databases with a case-insensitive collation.

Note KCM B&OM requires the Oracle 32-bit client. To perform a complete installation of KCM, if you are deploying KCM Repository on Oracle, you need both the 32-bit and 64-bit Oracle clients.

The names of the databases are partly configurable by using a prefix. The prefix cannot contain a colon or semicolon. The names (or users, in the case of Oracle) must appear as shown here:

- For the B&OM Repository: <Prefix>_OM_<Version>
- For the runtime information: <Prefix>_OM_RT_<Version>

<Prefix> is the value given in the `Database!Prefix` deployment parameter, which can be optionally overridden by the `OMDatabases!Prefix` parameter. <Version> is the version of KCM being deployed.

For example, the final database names might look as follows.

- KCM_OM_5.3.0
- KCM_OM_RT_5.3.0

When installing on SQL Server, the user specified with the `Database!User` deployment parameter requires one following privilege, in addition to the regular requirements:

ALTER_TRACE

When installing on Oracle, the following privileges are required on both database users, in addition to the regular requirements:

- ALTER SESSION
- CREATE VIEW
- CREATE PROCEDURE
- CREATE SEQUENCE

Also, the Oracle user for the B&OM Repository also requires these privileges:

CREATE TYPE

CREATE TRIGGER

EXECUTE ON DBMS_LOCK

EXECUTE ON DBMS_LOB

EXECUTE ON DBMS_OUTPUT

EXECUTE ON CTX_DDL

Manage databases

All databases that you use for KCM contain important production data. Unless following specific instructions from Kofax Technical Support, please do not make database modifications, because they may cause irreversible damage and loss of data. Also, be sure to back up all databases on a regular basis and follow the best practices recommended by your database provider.

License

Use of the Kofax Communications Manager package requires a valid license. The deployment process searches for a license file. The correct structure of the license file is shown here.

```
<?xml version="1.0" encoding="utf-8"?>
<ccm:Licence Version="1" xmlns:ccm="http://www.aia-itp.com/licence">
  <ccm:Product Name="ITP Server">
    <ccm:SubLicence Type="Base">
      <ccm:CompanyName></ccm:CompanyName>
      <ccm:NumberOfDocumentProcessors></ccm:NumberOfDocumentProcessors>
      <ccm:ExpirationDate></ccm:ExpirationDate>
      <ccm:Certificate></ccm:Certificate>
      <ccm:Environment></ccm:Environment>
      <ccm:LicenceCode></ccm:LicenceCode>
    </ccm:SubLicence>
    <ccm:SubLicence Type="OnLine">
      <ccm:ExpirationDate></ccm:ExpirationDate>
      <ccm:Certificate></ccm:Certificate>
      <ccm:NumberOfUsers></ccm:NumberOfUsers>
      <ccm:LicenceCode></ccm:LicenceCode>
    </ccm:SubLicence>
  </ccm:Product>
  <ccm:Product Name="ITP MDK Repository">
    <ccm:SubLicence Type="Base">
      <ccm:CompanyName></ccm:CompanyName>
      <ccm:ExpirationDate></ccm:ExpirationDate>
      <ccm:Certificate></ccm:Certificate>
      <ccm:Environment></ccm:Environment>
      <ccm:LicenceCode></ccm:LicenceCode>
      <ccm:NumberOfUsers></ccm:NumberOfUsers>
    </ccm:SubLicence>
    <ccm:SubLicence Type="Model Developer">
      <ccm:CompanyName></ccm:CompanyName>
      <ccm:ExpirationDate></ccm:ExpirationDate>
      <ccm:Certificate></ccm:Certificate>
      <ccm:Environment></ccm:Environment>
      <ccm:LicenceCode></ccm:LicenceCode>
      <ccm:NumberOfUsers></ccm:NumberOfUsers>
    </ccm:SubLicence>
    <ccm:SubLicence Type="Textblock Editor">
      <ccm:CompanyName></ccm:CompanyName>
      <ccm:ExpirationDate></ccm:ExpirationDate>
      <ccm:Certificate></ccm:Certificate>
      <ccm:Environment></ccm:Environment>
      <ccm:LicenceCode></ccm:LicenceCode>
      <ccm:NumberOfUsers></ccm:NumberOfUsers>
    </ccm:SubLicence>
  </ccm:Product>
</ccm:Licence>
```

For MDK Repository Base license, the value for NumberOfUsers must be set to 1.

For development or test purposes, the value for ITP/Server Base license should be "Development and Test." For all other environment elements, the value should be "Development or Test".

Note The KCM package might not deploy the full number of Document Processors that are permitted by the license. The default configuration does not install more than four Document Processors. Use the `Instance!NumberOfDPs` parameter during deployment to specify a different number of Document Processors. Then, you can use KCM Core Administrator to add or remove Document Processors.

Chapter 2

Install Kofax Communications Manager

The Kofax Communications Manager package is installed and deployed in phases. In the first phase, prerequisites are checked. If the deployment fails in this phase, you can restart it after the cause of the failure is resolved. If it fails again, you can uninstall the software.

1. To start the deployment, copy the package to the server.
2. On the server, extract the package contents to the root directory where a new directory, InstallServer, is automatically created. The unzipped files appear in the new directory, which is referred to as "package root."
3. Set up a PowerShell command to start the installation. This command line should call `Install.ps1` with the list of parameters (see [Deployment parameters](#)).
4. Click **Run as Administrator** to open a PowerShell window (64-bit) with elevated rights.
5. Navigate to the package root.
6. To run the prepared command, execute the **Set-ExecutionPolicy AllSigned** command.
7. If the following prompt appears, select **Always Run (A)**.

Example

```
Do you want to run software from this untrusted publisher?
File ...Install.ps1 is published by CN=Aia Software B.V., O=Aia Software B.V.,
L=Nijmegen,
S=Gelderland, C=NL and is not trusted on your system.
Only run scripts from trusted publishers.
[V] Never run [D] Do not run [R] Run once [A] Always run [?] Help (default is
"D"):
Navigate to the package root.
```

8. Wait for the process to finish.

Note When the installation is finished, you can remove the package root directory.

Deployment parameters

The following is a description of deployment parameters that you need to install Kofax Communications Manager.

Note When a parameter contains spaces and/or special characters, such as \$ or &, enclose the parameter in single quotes to prevent errors.

Parameter	Description
Database!Host	The host name for the database computer.

Parameter	Description
Database!Prefix	<p>The prefix for the database that will be in use.</p> <p>The database for the instance with instance number <nn> appears as <Prefix>_<nn>_<Version>.</p> <p>The database for the Contract Manager appears as <Prefix>_CM_<Version>. The prefix cannot contain a colon or a semicolon.</p>
Database!User	The SQL Server user used to access all databases. At the deployment stage, the user must have owner rights to the database.
Database!Password	The password for the SQL Server user used to access all databases.
Services!User	<p>The user account used for the installed services. This user must be a local administrator and must have the "Logon as a service" rights. When specifying a local user, place a period and backslash (.) at the beginning of the parameter.</p> <p>Example .\myuser</p>
Services!Password	The password for the services user.
License!File	The location of the .xml file with the license information.
Webserver!ServiceUser	<p>The user account used to run the web server services. This user must have the "Logon as a service" rights. When specifying a local user, add a period and backslash (.) at the beginning of the parameter.</p> <p>This parameter is required for the RunTimeOnDemand-NoContractManager setup, even if nothing is installed on a web server. In this case the user does not need a Tomcat installation.</p> <p>Example .\myuserws</p> <p>Note Webserver!ServiceUser parameter is not mandatory for the OutputManagementOnly setup.</p>
Webserver!ServicePassword	<p>The password belonging to the user provided through Webserver!ServiceUser.</p> <p>This parameter is required for the RunTimeOnDemand-NoContractManager setup, even if nothing is installed on a web server. In this case, the user does not need a Tomcat installation.</p> <p>Note Webserver!ServicePassword parameter is not mandatory for the OutputManagementOnly setup.</p>
DesignerAdministratorAccount!Password	<p>The password of the designer Administrator account that is created automatically by the installation. The password must be at least 20 characters long, and may only consist of letters (a-z, A-Z), numbers (0-9), and the "-" and "_" characters.</p> <p>This parameter is required only for default setups (no Deploy!PredefinedSetup).</p>

Parameter	Description
Database!CmdClient	Optional. Location of <code>sqlcmd.exe</code> . The package tries to derive the location of <code>sqlcmd.exe</code> automatically. If it fails, it prompts you to provide the location using this parameter.
Database!ServerName	Optional. The name of the SQL Server instance that should be in use by the installer. Used only if SQL Server is not using the default instance name.
Database!Port	Optional. The port used by SQL Server. Used only if SQL Server is not listening on the default port 1433. Note When using dynamic ports, specify <code>Database!ServerName</code> , not the port number.
RepDB!Type	Optional. Overrides the type of the KCM Repository database. The only value currently allowed is Oracle.
RepDB!ConnectionString	Optional. Overrides the Connection String used by the KCM Repository to connect to its database, if the <code>RepDB!Type</code> parameter is given. This should not contain <code>UserID</code> or <code>Password</code> elements. Required if <code>RepDB!Type</code> is set to Oracle.
RepDB!Password	Optional. Overrides the password for the databases used by the KCM Repository, if the <code>RepDB!Type</code> parameter is given. Default value is the value of the <code>Database!Password</code> parameter.
RepDB!Prefix	Optional. Overrides the prefix for the databases used by the KCM Repository, if the <code>RepDB!Type</code> parameter is given. Default value is the value of the <code>Database!Prefix</code> parameter.
Deploy!RootPath	Optional. The root drive/directory where the software is deployed. If this argument is omitted, the package is deployed to <code>C:\</code> .
Deploy!PredefinedSetup	Optional. The name of a supported predefined setup, where not all components are installed. Supported values are <code>Runtime-Desigtime-NoContractManager</code> , <code>RuntimeOnDemand-NoContractManager</code> , <code>ContractManagerOnly</code> , <code>Runtime-Desigtime-RemoteContractManager</code> , <code>Runtime-Desigtime-NoContract</code> , and <code>OutputManagementOnly</code> . For a complete setup, omit this parameter. For details, see Predefined setup options .
Instance!NumberOfDPs	Optional. Specifies how many Document Processors should be deployed for KCM Core. The default value is 4 or the number of Document Processors allowed by your license.
ContractManager!KeyStoreFile	Optional. The location of the certificate file (<code>.pfx</code> or <code>.p12</code>) for this server. Default connection to KCM is done securely over HTTPS using the SSL protocol. Required if <code>ContractManager!UseSSL</code> is true. This parameter is allowed for setups that do not install Contract Manager.

Parameter	Description
ContractManager!KeyStorePassword	<p>Optional. The KeyStore password. This is the password needed for the KeyStore File.</p> <p>Required if ContractManager!UseSSL is true.</p> <p>This parameter is allowed for setups that do not install Contract Manager.</p>
ContractManager!UseSSL	<p>Optional. Overrides the default behavior on using the SSL protocol. Defaults to true, indicating that any connections to KCM are secure using the SSL protocol and that KCM can be connected over HTTPS.</p> <p>Set the value to false to connect to KCM over HTTP.</p> <p>The ContractManager!UseSSL option configures the connectors for the service Catalina-CM in Tomcat.</p> <p>For security reasons, it disables TLSv1.0, and it only allows ciphers of at least 128 bits.</p> <p>This configuration can be found in the Tomcat server.xml file, which resides in the Tomcat installation folder under instance-CCMRuntime-5.3.0\conf. For more information, see the Tomcat documentation available on the Internet.</p> <p>If you configured SSL for connections to the Contract Manager manually, that is, you did not deploy KCM with the ContractManager!UseSSL parameter, and you want to remove the HTTP connectors, follow these steps:</p> <ol style="list-style-type: none"> 1. Locate the ManageCM.exe .xml file that resides in: <deploy root>\KCM\Programs\5.3.0\Management 2. Open the file and change the ReloadURLs setting as follows: <pre data-bbox="807 1272 1354 1409"><add key="ReloadURLs" value="https://localhost:<CM SSL port>/ccm/Administration/reloaddatabase;https://localhost:<PartnerAdmin SSL port>/ccm/Administration/reloaddatabase"/></pre> <p>This parameter is allowed for setups that do not install Contract Manager.</p>
ContractManager!UseAuthentication	<p>Optional. If set to false, anyone with access to the Contract Manager can execute any SOAP call on its interface.</p> <p>If set to true, when executing SOAP calls on the Contract Manager, the calling application should supply a user name and a password using an HTTP Basic Authentication header.</p> <p>If omitted, default value is true.</p> <p>This parameter is allowed for setups that do not install Contract Manager.</p>

Parameter	Description
ContractManager!SharedResourceLocation	<p>Optional. Specifies the location of a folder where the Contract Manager stores shared objects for import and export, such as project files and reference projects.</p> <p>This parameter is allowed for setups that do not install Contract Manager.</p> <p>By default, the shared objects are stored here: <code><deploy root>\KCM\Work\<version number>\SharedResources.</code></p> <p>Note If you configure the folder in a location outside <code><deploy root>\KCM\Work\<version number></code>, the new location is kept during future upgrades.</p>
ExampleWebApp!Install	<p>Optional. If set to true, an example web application is installed to test the installation and to reference KCM ComposerUI integration resources.</p> <p>If omitted, default value is false.</p> <p>This parameter is allowed for setups that do not install Contract Manager.</p>
OutputManagement!Install	<p>Only relevant if you have the general Kofax Communications Manager package that includes KCM B&OM.</p> <p>Optional. Specifies whether to install B&OM. Default is true.</p> <p>If set to false, this parameter is allowed, but ignored for the OutputManagementOnly setup.</p>

For the KCM B&OM deployment parameters, see [Additional deployment parameters for KCM B&OM](#).

Example deployment command

The following is an example deployment command.

```
.\Install.ps1 Database!Prefix=KCM Database!Host=KCMServer
Database!User=itpuser Database!Password=pwd123 Services!User=. \myuser
Services!Password=pwd456 "License!File=C:\{path}\My License\license.xml"
Webserver!ServiceUser=. \myuserws Webserver!ServicePassword=pwd789
ContractManager!KeyStoreFile=C:\{path}\cert.pfx
ContractManager!KeyStorePassword=pass123
DesignerAdministratorAccount!Password=default-administrator-password
```

Additional deployment parameters for KCM B&OM

The following is a description of deployment parameters that you might need to customize the KCM B&OM installation.

Parameter	Description
<code>OutputManagement!InstallType</code>	<p>Optional. Specifies the type of KCM B&OM installation to deploy:</p> <ul style="list-style-type: none"> • <code>Full</code>. A complete installation. • <code>StudioOnly</code>. Only deploys KCM Studio and its requirements. • <code>ServerOnly</code>. A complete installation, omitting KCM Studio. <p>Default is <code>Full</code>.</p>
<code>OMDatabases!Type</code>	Optional. Overrides the type of the KCM B&OM databases. The only value currently allowed is <code>Oracle</code> .
<code>OMDatabases!ConnectionString</code>	<p>Optional. Overrides the connection string that KCM B&OM uses to connect to its databases if the <code>OMDatabases!Type</code> parameter is specified.</p> <p>Note This connection string must be a valid Oracle connection string. ODBC connection strings or other types are not supported.</p>
<code>OMDatabases!Password</code>	Optional. Overrides the password for the databases used by KCM B&OM if the <code>OMDatabases!Type</code> parameter is specified. Default value is the value of the <code>Database!Password</code> parameter.
<code>OMDatabases!Prefix</code>	Optional. Overrides the prefix for the databases used by KCM B&OM if the <code>OMDatabases!Type</code> parameter is specified. Default value is the value of the <code>Database!Prefix</code> parameter.
<code>OutputManagement!Secondary</code>	Optional. When set to <code>True</code> , and the predefined setup <code>OutputManagementOnly</code> is used, it adds another KCM B&OM installation.

Install KCM Repository on an Oracle database

During installation, you can deploy KCM Repository on a different type of database than the Contract Manager. The only override type currently supported is Oracle.

Note Oracle Client versions 11.02.00.01 and 11.02.00.02 are affected by bug 9877346. These client versions are not supported for use with KCM Repository.

1. To install KCM Repository on an Oracle database, add the parameter `RepDB!Type=Oracle` to the deployment command.
2. To override the Connection String that KCM Repository uses to connect to its database, add the parameter `RepDB!ConnectionString` to the deployment command. Provide a value for the

parameter. For example, `DSN=MyORA`, where MyORA is the name of a data source created using the 64-bit version of the ODBC Data Source Administrator tool.

This parameter should not contain UserID or Password elements.

Note Previous versions of KCM required a 32-bit DSN. Now KCM Repository is a 64-bit program, so a 64-bit DSN is required.

- You can add the optional `RepDB!Password` parameter to override the password used for connecting to the database.

Note When installing more than one instance, use the same password for all users.

- You can add the optional `RepDB!Prefix` parameter to override the prefix for creating the database user needed to connect to Oracle during KCM installation.

Oracle user

When installing the KCM package, the system derives the user to log on to the database using a combination of the prefix, instance number, and KCM version.

`<Prefix>_<Instance number>_<KCM version>`

- `<Prefix>` Value of `RepDB!Prefix` if supplied; otherwise, the value of `Database!Prefix`
- `<Instance number>` 01 when installing a default package. When the command `AddInstance` is used to add new instances, the instance number is the supplied `Instance!Number` in two digits. Numbers smaller than 10 are preceded by 0. For example, `Instance_01`.
- `<KCM version>` KCM version

Example `KCM_01_5.3.0`

Create the user prior to deploying the package. The user should have the privilege to login (`CREATE SESSION`) and the privilege to create tables (`CREATE TABLE`). At least 60 MB of table space is needed for the initial installation.

Note When logging on to Oracle, the user identifier is entered without quotes. This means that Oracle interprets quotes as uppercase. As the predefined user identifier contains a dot, it is likely that you create a user as a quoted identifier. Do not use lowercase characters in the quoted user identifier.

Example

`"KCM_01_5.3.0"` is able to log on.

`"Kcm_01_5.3.0"` fails to log on because `Kcm_01_5.3.0` is automatically interpreted as `KCM_01_5.3.0`, and that user does not exist.

Predefined setup options

The installation package contains predefined setup options. The complete installation consists of KCM Core, KCM Repository, KCM Designer, KCM ComposerUI for HTML5, and the Contract Manager.

- The Runtime-DesignTime-NoContractManager setup performs a complete installation, but omits the Contract Manager.
- The RuntimeOnDemand-NoContractManager setup installs KCM Core and KCM Repository only.
- The ContractManagerOnly setup installs the Contract Manager only.
- The Runtime-DesignTime-RemoteContractManager setup performs a complete installation, omits the Contract Manager, and is configured to have Contract Manager on another machine/server.
- The Runtime-DesignTime-NoContract setup performs a complete installation, but omits creating default contracts.
- The OutputManagementOnly setup installs KCM Batch & Output Management only.

Note Currently, KCM Designer for Windows is still supported to get access to KCM Repository. KCM Designer for Windows is not installed by default so you have to install it manually. For more information, see [Install KCM Designer for Windows](#).

Install KCM Designer for Windows

KCM Designer for Windows is a client application used to perform certain tasks for KCM Designer, such as create projects and users. You can install the application manually after the KCM installation package is deployed. For each instance, an installer and detailed information are provided.

1. Once the installation package is deployed, navigate to `<deploy root>\KCM\Work\<version>\Instance_01\designer\Client` and locate the KCM Designer for Windows installer KCM Designer for Windows setup.exe.

2. Run the installer.

The installer prompts you to enter the installation name that will determine the directory name where the client will be residing. When the installer prompts you to provide the necessary values for server and host, open the ReadMe file located in the same directory and copy the values from the file.

By default, the client is deployed in `C:\Program Files (x86)` for a 64-bit OS.

Install KCM on more than one computer

You can install KCM on multiple computers, with one computer containing the Contract Manager and several computers containing one or more instances. This makes KCM scalable and easy to upgrade.

Install KCM instances on a computer using remote Contract Manager

1. Install the KCM package with no Contract Manager.

To do so, use the predefined setup `Runtime-Designtime-RemoteContractManager`, as shown in this example command.

```
.\install Database!Prefix=KCM Database!Host=localhost Database!User=itpuser
Database!Password=pwd123 Services!User=.myuser Services!Password=pwd456
License!File="c:\{path}\my_licenses\license.xml"
Deploy!PredefinedSetup=Runtime-Designtime-RemoteContractManager
```

2. Now you need to configure KCM instances on the remote Contract Manager. For information on how to perform the action, see [Install dedicated Contract Manager on a computer with no instances](#).

Install dedicated Contract Manager on a computer with no instances

1. Install the KCM package on a machine with the Contract Manager only.

To do so, use the predefined setup `ContractManagerOnly`, as shown in this example command.

```
.\install Database!Prefix=KCM Database!Host=localhost Database!User=itpuser
Database!Password=pwd123 Services!User=.myuser Services!Password=pwd456
License!File="c:\{path}\my_licenses\license.xml" Deploy!
PredefinedSetup=ContractManagerOnly
```

2. To configure the Contract Manager, use the `ManageCM` tool that resides in: `<deploy root>\KCM\Programs\<version>\Management`.
3. Register an instance to your remote Contract Manager with the following command, substituting the example parameters for your actual parameters.

```
ManageCM /RegisterInstance ... /Host=host42 /Instance=5
```

Note In the above command, the optional `/Database=` and `/Prefix` parameters are skipped for brevity. Replace the ellipsis with these parameters. `/Database=` specifies the SQL Server instance while `/Prefix=` is a database prefix chosen during the installation of KCM Core with the `Database!Prefix=` parameter.

4. Now you need to add contracts to the Contract Manager and associate the instance with the contracts. For information on how to do so, see the chapter "Work with the Contract Manager" in the *Kofax Communications Manager Getting Started Guide*.

Install KCM without default contracts

The KCM installation package creates default contracts for each instance. To create your own contracts, you need to install KCM without default contracts and then create contracts to access KCM instances.

1. Install the KCM package on a machine using the predefined setup `Runtime-Designtime-NoContract`, as shown in this example command.

```
.\install Database!Prefix=KCM Database!Host=localhost Database!User=itpuser
Database!Password=pwd123 Services!User=.myuser Services!Password=pwd456
License!File="c:\{path}\my_licenses\license.xml"
Deploy!PredefinedSetup=Runtime-Designtime-NoContract
```

This predefined setup omits default contracts, but creates one KCM instance. To add extra instances, see [Add extra instances after installation](#).

2. As there are no default contracts, you need to create contracts to access KCM instances.

To do so, you need the ManageCM tool that resides in: `<deploy root>\KCM\Programs\<version>\Management`

For information on how to create contracts, see the chapter "Work with the Contract Manager" in the *Kofax Communications Manager Getting Started Guide*.

3. Now you need to create interfaces and contract types to add them to your contract.

For information on how to perform the actions, see the chapter "Work with the Contract Manager" in the *Kofax Communications Manager Getting Started Guide*.

Install KCM without the Contract Manager

The Contract Manager is designed to manage connections to KCM instances, but you can connect to the various KCM products directly. In this case, if more than one instance is installed on a computer, there will be multiple instances of the various KCM components accessible through different ports. This implies that you need to monitor which ports are needed to access a particular KCM component.

Install the KCM package on a computer using the predefined setup `NoContractManager`, as shown in this example command.

```
.\install Database!Prefix=KCM Database!Host=localhost Database!User=itpuser
Database!Password=pwd123 Services!User=.myuser Services!Password=pwd456
License!File="c:\{path}\my_licenses\license.xml"
Deploy!PredefinedSetup=Runtime-Design-time-NoContractManager
```

This predefined setup omits the Contract Manager. Access to the various KCM components will be done directly using the port it is listening on.

- For KCM Core, the port is 3003 for the instance 1; 3005 for the instance 2, and so on. KCM Core offers an integration layer consisting of APIs and a COM library, an AS/400 command, and Java classes used to integrate KCM Core in a business application. For more information on KCM Core, see the *Kofax Communications Manager Core Developer's Guide*.
- KCM Designer is listening on the port 8280 for the instance 1; port 8281 for the instance 2, and so on. You can access KCM Designer with the following example URL:

http://<kcm machine>:<port>/ccm_designer_<instance>

Where `<kcm machine>` is the name of a computer on which KCM is installed, and `<instance>` is the number of an instance in two digits.

Example `http://host24:8280/ccm_designer_01`

Install KCM B&OM on multiple computers

Consider the following prerequisites prior to installing KCM B&OM on a different machine:

- An additional KCM B&OM installation must have access to a shared storage folder as configured for the primary KCM B&OM installation.
- An additional KCM B&OM installation must have access to the KCM Core and KCM Repository available on the primary KCM B&OM installation. Also, KCM Core must be configured to allow external access.
- The database information for an additional KCM B&OM installation must match that of the primary installation as they share the same database.

- By default, the host properties of KCM Core and KCM Repository are set to "localhost." You need to change them to the host name or IP address of the primary KCM B&OM installation server. Additionally, if the connection strings to the Repository and Runtime databases are set to "localhost," you need to change them to the host name or IP address of the database server.
 1. In KCM Studio, on the **Administration** tab, click **System Administration** and then click the system object.
 2. In the **Object Inspector**, locate the **CM/Core** and **CM/Repository** sections and adjust the **Host** properties, if necessary.
 3. Double-click the Repository database.
If the **Connection String** property is set to "localhost," change it to the correct host name or IP address for the server where the primary KCM B&OM installation is deployed.
 4. Click **OK**.
 5. On the **Administration** tab, click **DB Alias Administration**.
 6. Double-click the Runtime (Odin) database.
If the **Connection String** property is set to "localhost," change it to the correct host name or IP address for the server where the primary KCM B&OM installation is deployed.
 7. Click **OK**.
 8. Save the changes.
- Do not configure a shared input request folder for the primary and additional installations as UNC paths are not supported for the registration phase.
- Depending on the type of the DocBridge license you have, you need to use a separate license file for each machine where KCM B&OM is installed or a shared license file accessible from both the primary and additional installations.

To deploy an additional KCM B&OM installation on a new machine, use the predefined setup `OutputManagementOnly` and the `OutputManagement!Secondary` parameter set to true, as shown in the example commands below.

Example for SQL Server:

```
.\install Database!Prefix=KCM Database!Host=KCMServer Database!User=itpuser
Database!Password=pwd123 Services!User=. \myuser Services!Password=pwd456
"License!File=C:\{path}\My License\license.xml"
Deploy!PredefinedSetup=OutputManagementOnly OutputManagement!Secondary=true
```

Example for Oracle:

```
.\install Database!Prefix=KCM Database!Host=KCMServer Database!User=itpuser
Database!Password=pwd123 Services!User=. \myuser Services!Password=pwd456
"License!File=C:\{path}\My License\license.xml"
OMDatabases!Type=oracle OMDatabases!ConnectionString="Data Source=CCMOra"
Deploy!PredefinedSetup=OutputManagementOnly OutputManagement!Secondary=true
```

To complete the post-installation, perform these steps:

1. On the primary KCM B&OM installation, navigate to `<deploy root>\KCM\Programs \<KCM_version>\Output Management` and copy the `Repository.Config` and `UserRepository.Config` files.
2. Paste the files into the new KCM B&OM installation.

3. Start the KCM Output Management services of the new KCM B&OM installation.

Chapter 3

Upgrade your installation

You can upgrade KCM versions 4.4 and later to KCM 5.3.0. The upgrade installs KCM 5.3.0 alongside your existing installation, using the same configuration that the current KCM was installed with.

Note the following when upgrading your KCM installation:

- If you have KCM 4.4 installation, first upgrade it to 4.5. Then, you can upgrade it to 5.3.0.
- You need to manually reinstall any custom additional language packs. We recommend that you execute the script `GetLanguagePack.ps1` prior to upgrading the product, update the custom language packs so they include new translations, and then install the language packs using the script `AddLanguagePack.ps1`. For information on the scripts, see the chapter "Manage language packs" in the *Kofax Communications Manager Getting Started Guide*.

Perform the upgrade

To upgrade a KCM installation, you need the following:

- Create a new database for the Contract Manager called `<Database!Prefix>_CM_5.3.0`
- Create a new database for the KCM instance called `<Database!Prefix>_01_5.3.0`
- Enough disk space to accommodate KCM Repository content

Note The new databases must be owned by the user who owns the existing databases.

In addition to the preceding requirements, when you upgrade an installation that includes KCM Batch & Output Management, the following is required:

- Create a new database for KCM Batch & Output Management called `<Database!Prefix>_OM_5.3.0`
- Create a new database for KCM Batch & Output Management runtime information called `<Database!Prefix>_OM_RT_5.3.0`
- Enough disk space to accommodate KCM Batch & Output Management content

Also, check the lists of the required and optional parameters prior to performing the upgrade (see [Parameters for the upgrade](#) and [Additional upgrade parameters for KCM B&OM](#)).

To upgrade a standard installation, execute the following command, substituting the example parameters values with your actual values.

```
.\Upgrade.ps1 Database!User=itpuser Database!Password=pwd123
Services!User=. \myuser Services!Password=pwd456
"License!File=C:\{path}\MyLicense\license.xml"
Webserver!ServiceUser=. \myuserws Webserver!ServicePassword=pwd789
ContractManager!KeyStoreFile=C:\{path}\cert.pfx
```

```
ContractManager!KeystorePassword=pass123
```

Replace the preceding example parameters with actual values that apply to your currently active KCM setup.

When the upgrade is completed, the 5.3.0 installation is active and available for testing. The previously active installation is disabled.

The KCM Repository content is now copied to the new database. Any changes to the Repository content are not reflected in the older version.

Important An upgrade only creates a basic Tomcat Server configuration for new KCM 5.3.0 Tomcat instances. If any customization has been applied to the Tomcat Server configuration in the previous KCM version, you should reapply them to the instances installed for KCM 5.3.0.

If an upgrade encounters an error while upgrading the instances, but the upgraded software has already been installed, the previously active version of KCM is re-activated. After solving any reported problems and uninstalling the failed upgrade, you can retry the upgrade.

Parameters for the upgrade

The following is a description of the parameters that you need to upgrade the installation.

Parameter	Description
Database!User	The SQL Server user used to access all databases. At the deployment stage, the user must have owner rights to the database.
Database!Password	The password for the SQL Server user needed to access all databases.
Services!User	The user account used for the installed services. This user must be a local administrator and must have the "Logon as a service" rights. When specifying a local user, place a period and backslash (.\) at the beginning of the parameter. Example .\myuser
Services!Password	The password for the services user.
License!File	The parameter is required when upgrading from KCM 5.0.
Webserver!ServiceUser	The user account used to run the web server services. This user must have the "Logon as a service" rights. When specifying a local user, add a period and backslash (.\) at the beginning of the parameter. Example .\myuserws
Webserver!ServicePassword	The password belonging to the user provided through Webserver!ServiceUser.
Upgrade!Version	Optional. The parameter is used to specify the version to upgrade from if no active installation can be found. If an active installation can be found, this parameter is ignored.

Parameter	Description
RepDB!Password	Optional. Overrides the password for the databases used by KCM Repository in case the previous version was deployed with an overridden password. Default value is the value of the Database!Password parameter.
ContractManager!KeyStoreFile	Optional. The location of the certificate file (.pfx or .p12) for this server. Default connection to KCM is done securely over HTTPS using the SSL protocol. Required if ContractManager!UseSSL is true.
ContractManager!KeystorePassword	Optional. The KeyStore password. This is the password needed for the KeyStore File. Required if ContractManager!UseSSL is true.
ContractManager!UseSSL	Optional. Overrides the default behavior on using the SSL protocol. Defaults to true, indicating that any connections to KCM are secure using the SSL protocol and that KCM can be connected over HTTPS. Set the value to false to connect to KCM over HTTP. The ContractManager!UseSSL option configures the connectors for the service Catalina-CM in Tomcat. For security reasons, it disables TLSv1.0, and it only allows ciphers of at least 128 bits. This configuration can be found in the Tomcat server.xml file, which resides in the Tomcat installation folder under instance-CCMRuntime-5.1\conf. For more information, see the Tomcat documentation available on the Internet. If you configured SSL for connections to the Contract Manager manually, that is, you did not deploy KCM with the ContractManager!UseSSL parameter, and you want to remove the HTTP connectors, follow these steps: <ol style="list-style-type: none"> 1. Locate the ManageCM.exe .xml file that resides in: <deploy root>\KCM\Programs\5.3.0\Management 2. Open the file and change the ReloadURLs setting as follows: <pre><add key="ReloadURLs" value="https://localhost:<CM SSL port>/ccm/Administration/reloaddatabase;https://localhost:<PartnerAdmin SSL port>/ccm/Administration/reloaddatabase"/></pre>

For the KCM B&OM upgrade parameters, see [Additional upgrade parameters for KCM B&OM](#) .

Perform the KCM B&OM upgrade

When upgrading a KCM installation that includes KCM Batch & Output Management, it is important that you note the following limitations and consequences:

- **History of the previous systems not transferred**

The contents of the previous B&OM installation, which includes systems and their objects, are transferred to the new installation without their history, and the transfer history details are lost. This information is still present in the previous installation if you reactivate it. Also, changes to the objects in the new version have no effect on the previous version.

- **Paths must be manually updated**

Paths to files or folders on the file system, such as the path to the Compart license file on the OdinConversion component, are not updated automatically. You need to update them manually.

- **Runtime information remains in use**

The runtime information of the previous KCM B&OM installation, including the runtime databases, existing jobs, processes, envelopes and stacks, as well as custom databases, remain in use in the new version. The transferred systems in the new installation continue to use the same runtime databases as before. If strict database separation is needed, you must manually copy the databases and adjust the connection in KCM Studio.

- **New system also installed**

The upgrade also installs a new system into KCM B&OM, which includes standard processes and objects from the new version. It uses the new runtime database that you created before the upgrade.

- **Multiple content databases no longer supported**

Multiple KCM B&OM content databases are no longer supported. When upgrading from a KCM B&OM installation with multiple content databases, you are prompted to select only one with the `OutputManagement!RepositoryAlias` parameter (see [Additional upgrade parameters for KCM B&OM](#)). You can transfer any systems stored in different content databases after the upgrade using the `ExportOutputManagementContent` and `ImportOutputManagementContent` management scripts (see [Export KCM B&OM content](#) and [Import KCM B&OM content](#), respectively). Any system-transfer links between systems residing in different content databases are broken in the new version, and you need to reapply them manually.

- **Upgrade from a machine with no KCM installed has limitations**

Upgrading from a machine that has no other KCM components installed is not fully supported. You can use the `ExportOutputManagementContent` and `ImportOutputManagementContent` management scripts to transfer content from the previous installation to a new installation deployed using the standard installation command.

Note In this case, you must manually deactivate the new version and reactivate the old one, if necessary.

- **Custom service hosts recreated under a new name**

Custom service hosts created in the previous version are recreated in the new version under a different name and run the same processes as before. Services that were running processes from multiple different systems are no longer supported. Such services are not automatically recreated, and you need to split them up manually. Customized configuration, such as log levels or Remote Control, is not transferred, and you need to reapply it manually. To view active services, create or

remove them, use the `ListOutputManagementServices`, `AddOutputManagementService`, and `RemoveOutputManagementService` scripts (see [Add custom KCM B&OM service hosts](#)).

- **Customized Odin parameters saved to a new configuration file**

When upgrading from KCM 5.0 or 5.1, customized Odin parameters in the previous version are taken over, but saved to a new configuration file specific to the previous version to prevent compatibility issues between the different version parameters. To modify the Odin parameters configuration for systems transferred from the previous version, use the new configuration file `odinparameters_<previous version>.config`.

- **Customized configuration for CM Studio not automatically reapplied**

Customized configuration for KCM Studio and the two standard services, Core and Output, is not transferred, and you have to reapply it manually. Also, the system used for assessing the user rights in KCM Studio is now configured to use the newly created system allowing users to immediately access the new objects. The configured user roles and role mappings are transferred as part of the general content transfer, but to continue using them, you have to update them manually to include rights for all new objects in KCM B&OM.

- **Compart configuration files are not automatically transferred to the new version**

Customized DocBridge profiles are transferred to the new version, but other Compart configuration files, such as license files, are not moved.

Additional upgrade parameters for KCM B&OM

The following is a description of the parameters that you might need to upgrade the KCM B&OM installation.

Parameter	Description
<code>OMDatabases!Password</code>	Optional. Overrides the password for the databases used by KCM Batch & Output Management if the previous version was deployed with an overridden password. Default value is the value of the <code>Database!Password</code> parameter.
<code>OutputManagement!RepositoryAlias</code>	Optional. Specifies the content database to use to transfer systems and objects from the previous version. Specify the content database as the name of the database alias. Required if multiple content databases are present in the previous version.

Switch to an older installation

You can switch to an older installation if the new installation does not work properly.

1. Navigate to: `<deploy root>\KCM\Programs\<version>\Management`
2. Run the command `\Activate.ps1 -Version <version number>`

After running this command, the older installation is active and the new installation is disabled. To switch back to 5.3.0, run the preceding command with 5.3.0 as a value for `<version number>`.

Upgrade a customized setup

A customized setup is a setup to which custom configuration has been added after the installation of KCM. To upgrade a customized setup, perform a standard upgrade (see [Upgrade your installation](#)) and customize the new deployment. After performing the upgrade, a standard setup of KCM 5.3.0 is available that might not contain all custom configuration. In this case, you need to manually deploy the missing custom configuration to the newer version.

You need to deploy some custom configuration of the previous version to the newer version:

- Manually created files and folders within the KCM Work directory.

The Work directory for the new deployment resides in: `<deploy root>\kcm\work\5.3.0\<instance_number>`.

The Work directory for the previous KCM version is located at: `<deploy root>\kcm\work\<previous_version>\<instance_number>`.

Note If your customization refers to these folders using paths that contain the KCM version number, verify that you change the version number when you move these folders.

- Changes made through KCM Core Administrator other than constants and Environments settings. For example, changes to the DP Manager, Logging, Monitor, HTTP Monitor, Job Recovery, or Advanced tabs.
- Extra installed KCM components such as KCM ComposerUI for ASP.NET, KCM Core Web Services interface, and KCM Designer for Windows.

Upgrade contract types and interfaces not in use by any contract

An upgrade skips contract types and interfaces (or other configuration) that are not in use by active contracts. Only the configuration needed to duplicate the current setup is moved to a newer version. To duplicate unused contract types and interfaces, follow these steps:

1. Start the ManageCM tool.

For information on the tool, see the chapter "Use the ManageCM tool" in the *Kofax Communications Manager Getting Started Guide*.

2. Export the contract types belonging to the KCM version you are upgrading from.

Execute the following command, substituting the example parameters, such as `/user` and `/password`, for your actual parameters.

```
ManageCM /ExportDatabase /CMVersion=5.0 /File=Export.xml
        /ContractTypes
        /user=dbuser /password=dbpassword
```

This command creates an XML file containing all the contract types and their interfaces.

3. Import the created XML file into the new version of the Contract Manager.

Execute the following command, substituting the example parameters for your actual parameters.

```
ManageCM /ImportDatabase /File=Export.xml
        /user=dbuser /password=dbpassword
```

You can use this procedure for other types of configuration, if necessary.

Upgrade KCM when installed on more than one computer

When KCM is installed on more than one computer, it is not necessary to upgrade all computers in one step. You can upgrade all the computers gradually, taking into consideration the order in which KCM components are upgraded.

Upgrade the Contract Manager before the instances

To upgrade the Contract Manager prior to upgrading the instances, perform the following steps.

1. Upgrade the Contract Manager.

To do so, execute the `.\Upgrade` script on the computer on which the Contract Manager is installed. For more information on the script, see [Perform the upgrade](#). The setup still functions after the upgrade.

2. Upgrade the instances by executing the `.\Upgrade` script on the machine on which the instances are running.
3. Start the ManageCM tool and use the `/Update` flag as shown in this example to complete the upgrade.

```
ManageCM /RegisterInstance /Host=host42 /Instance=1 /Update  
/user=dbuser /password=dbpassword
```

The upgrade is completed. You can now access the new instances.

Upgrade the instances before the Contract Manager

The following topic describes how to upgrade the instances prior to upgrading the Contract Manager.

1. Upgrade the instances.

Execute the `.\Upgrade` script on the computer on which the instances are running. For more information on the script, see [Perform the upgrade](#).

2. Upgrade the Contract Manager.

Execute the `.\Upgrade` script on the computer on which the Contract Manager is installed. Your setup automatically switches to newer versions of the instances, also registering the instances.

Chapter 4

Post-installation steps

Test the installation

When the KCM package is installed, an example web application is available to test the installation and to reference KCM ComposerUI integration resources. The web application that gives you the ability to test the installation is available when the `ExampleWebApp!Install` deployment parameter is set to true. For production and acceptance systems, this web application is not required, and you should not install it. To test the installation, use the following web pages:

- `https://<kcm server>:443/start/home.html`
- `http://<kcm server>:8081/start/home.html` (when `ContractManager!UseSSL` is set to false)

Important Default KCM Contract Manager's Tomcat instances (TCP port 443) are configured to use HTTPS. You should not configure any other Tomcat instances to use HTTPS. Also, configure Tomcat instances to suppress extended error messages.

You can change the default port value 443 to the appropriate port value to avoid possible conflicts in case other third-party software on your machine is using the same ports.

If you want to use alternate ports, ensure that you first consult with your administrator.

The default user is "Administrator". The default password is entered through the `DesignerAdministratorAccount!Password` parameter during installation. After installation you can change the password through KCM Designer, see the section "Change your password" in the *KCM Designer online Help*.

To test an interactive composition of a document, under **ComposerUI HTML5**, click **Test**. Afterwards, fill in the forms and a document is opened by the end of the run. The first time you do this test run, you are prompted to install an ActiveX component. This component is used to open Microsoft Word when the composition run is completed.

KCM additional instances install their own test web page based on the number of the instance. For example, instance 5 receives the following web page: `https://<kcm server>:443/start5/home.html`

To remove the test web application, see the next section.

Remove the test web application

We advise that you remove the test web application in production environments.

To remove the web application, navigate to `<Tomcat installation folder>\instance-CCMRuntime-<version number>\webapps-CM` and remove the `Start` folder.

Access product documentation

You can access the online KCM documentation using the link (see [Related documentation](#)) or directly from the product. When you click the help button in KCM Designer, KCM Designer for Windows, KCM Core, or KCM Studio, online documentation appears in a new browser window.

To access the online documentation, you must have an active Internet connection. If the security policy for your organization restricts Internet access or the Internet connection is not stable, you can access the documentation in offline mode while using the product.

Offline documentation

To make the documentation available for use in offline mode, you need to download it from the [Kofax Fulfillment Site](#) after purchasing the product. From the site, download the following files:

- Download the `<KofaxCommunicationsManagerDocumentation_5.3.0_EN.zip>` file that contains the entire KCM documentation set in English. This file is required for all users.
- If required, download the compressed file corresponding to the language translation that you need, such as `<KofaxCommunicationsManagerDocumentation_5.3.0_ES.zip>` for Spanish.

Note KCM Designer online Help is the only item in the KCM documentation set that is translated at this time.

After you install the KCM product, proceed to the following sections and extract the compressed files for each KCM component as required. When you finish extracting the files, the documentation folder in each location will contain the following two folders:

- `print` with the PDF documentation
- `help` with the KCM Designer Help

You can access the documentation from these locations. Also, when you click a help button in a KCM component and select the required document, the system will open an offline, local copy of that document.

Note The names of the folders where you extract the compressed files are hard-coded and case-sensitive. Do not rename the folders.

Offline documentation for KCM Designer

1. Extract the contents of the English compressed file to:

```
<deploy root>\<Tomcat installation folder>\Tomcat <version>\instance-WebDesigner-5.3\webapps\ccm_designer_<instance number>\doc\en
```

2. If you require other, non-English languages, extract the contents of the compressed file for a particular language to the respective language folder, which resides here:

```
<deploy root>\<Tomcat installation folder>\Tomcat <version>\instance-WebDesigner-5.3\webapps\ccm_designer_<instance number>\doc\<language>
```

The supported languages are represented with the following language codes:

- de: German
- es: Spanish
- fr: French
- it: Italian
- ja: Japanese
- nl: Dutch
- pt_BR: Brazilian Portuguese

For example, to use the KCM Designer Help in Spanish, extract the contents of the Spanish compressed file to the **es** folder.

Offline documentation for KCM Studio and KCM Core Administrator

For KCM Studio and KCM Core Administrator, extract the contents of the English compressed file to:

```
<deploy root>\KCM\Documentation\5.3\en
```

Offline documentation for KCM Designer for Windows

For the KCM Designer for Windows component, which you install separately after the KCM product is installed, extract the contents of the English compressed file to:

```
<deploy root>\KCM Designer Instance <instance number>\Documentation\en
```

For more information on the KCM Designer for Windows installation and default location, see [Install KCM Designer for Windows](#).

Add extra instances after installation

If additional KCM instances are required after the software has been installed, you can add them using the `AddInstance.ps1` script that resides in: `<deploy root>\KCM\Programs\<version>\Management`. Instances added with this script use the same configuration as used by the original installation.

If the original installation deployed the standard KCM contract (with partner KCM and customer local), the additional instances are also deployed with the standard KCM contract. These instances will have partner KCM and customer local<n>, where <n> is the number of the instance. If the original installation did not deploy the standard KCM contract, the additional instances will not have the standard KCM contract deployed.

Note Adding an instance to an active KCM installation restarts the shared services in use by the other instances, making them temporarily unavailable, until they finish restarting.

Check the required parameters prior to adding extra instances (see [Extra instances parameters](#)).

1. To add an instance, navigate to: <deploy root>\KCM\Programs\<<version>\Management
A license file and a database matching the prerequisites for the base KCM installation must be present.

Note When adding an instance to an active KCM installation, ensure that no users are connected to the other instances as the process triggers restart of these instances.

2. Instances are represented by a number, for example, Instance_01. Determine a number to represent the new instance.
3. Execute the script.

The following is an example command.

```
.\AddInstance.ps1 Database!User=itpuser Database!Password=pwd123
Services!User=.\myuser Services!Password=pwd123
"License!File=c:\{path}\My License\License.xml"
Webserver!ServiceUser=.\myuserws Webserver!ServicePassword=pwd789
Instance!Number=5
DesignerAdministratorAccount!Password="password"
```

Extra instances parameters

This table contains a description of the parameters required to add extra instances after the installation.

Parameter	Description
Database!User	The SQL Server user used to access all databases. At the deployment stage, the user must have owner rights to the database.
Database!Password	The password for the SQL Server user used to access all databases.
Services!User	The user account used for the installed services. This user must be a local administrator and must have the "Logon as a service" rights. When specifying a local user, place a period and backslash (.) at the beginning of the parameter. Example .\myuser
Services!Password	The password for the services user.
License!File	The location of the .xml file with the license information.
Webserver!ServiceUser	The user account used to run the web server services. This user must have the "Logon as a service" rights. When specifying a local user, add a period and backslash (.) at the beginning of the parameter. Example .\myuserws
Webserver!ServicePassword	The password belonging to the user provided through Webserver!ServiceUser.

Parameter	Description
<code>DesignerAdministratorAccount!Password</code>	<p>The password of the designer Administrator account that is created automatically for the new instance. The password must be at least 20 characters long, and may only consist of letters (a-z, A-Z), numbers (0-9), and the "-" and "_" characters.</p> <p>This parameter is required only for installations with a default setup (no <code>Deploy!PredefinedSetup</code>) and when <code>Contract!CreateContract</code> is not false.</p> <p>The Administrator account is also created when running in LDAP mode. However, the created account can only be used when LDAP mode is switched off.</p>
<code>Instance!Number</code>	<p>Specifies the number that the instance uses. For example, <code>Instance!Number=5</code>. The maximum value for the parameter is 99. This applies to the <code>AddInstance</code> and the <code>RemoveInstance</code> script specifications.</p>
<code>Instance!NumberOfDPs</code>	<p>Optional. Specifies how many Document Processors should be deployed for KCM Core. The default value is 4 or the number of Document Processors allowed by your license.</p>
<code>Contract!CreateContract</code>	<p>Optional. The parameter is used to indicate whether an additional instance gets the default contract KCM Local<n>. The value can be either true or false.</p>
<code>ExampleWebApp!Install</code>	<p>Optional. If set to true, an example web application is installed to test the installation and to reference KCM ComposerUI integration resources. If omitted, default value is false.</p>

Add extra instances after installation in combination with Oracle

1. To add extra instances after you installed KCM with `RepDB!Type` set to Oracle, create additional users matching the instance numbers you want to create.

2. Execute the command `AddInstance`.

You can add an optional parameter `RepDB!Password` to provide a password for the Oracle database.

For more information on adding extra instances, see [Add extra instances after installation](#). For a description of the command `AddInstance`, see the *Kofax Communications Manager Getting Started Guide*.

Note When installing more than one instance of KCM, use the same password for all users.

Remove specific instances

If a specific KCM instance is no longer needed, you can remove it using the `RemoveInstance.ps1` script that resides in: `<deploy root>\KCM\Programs\<version>\Management`. Removing an instance clears files, services, and settings specific to this instance. The content database is unaffected.

Note Removing an instance from an active KCM installation restarts the shared services also in use by other instances, temporarily making them unavailable.

Check the list of the required parameters prior to removing specific instances (see [Parameters to remove specific instances](#)).

1. To remove an instance, navigate to: `<deploy root>\KCM\Programs\<version>\Management`

Note When removing an instance from an active KCM installation, ensure that no users are connected to the other instances as the process triggers restart of these instances.

2. Execute the script.

The following command removes the instance with the number 2. It also removes the contract registered for this instance from the Contract Manager.

```
.\RemoveInstance.ps1 Instance!Number=2
Database!User=itpuser Database!Password=pwd123
```

Parameters to remove specific instances

This table contains a description of the parameters required to remove specific instances of KCM.

Parameter	Description
<code>Instance!Number</code>	Specifies the number of the instance to remove.
<code>Database!User</code>	The SQL Server user used to access all databases. At the deployment stage, the user must have owner rights to the database.
<code>Database!Password</code>	The password for the SQL Server user used to access all databases. Specify this parameter to remove a registered contract.

Add custom KCM B&OM service hosts

To run KCM B&OM processes without using KCM Studio, you can add custom KCM B&OM service hosts.

Log files for KCM Studio and all KCM Batch & Output Management services, including custom service hosts, are written to the Output Management work folder, which resides in: `<deploy root>\KCM\Work\<version>\Output Management\Logs`.

For more information on KCM B&OM and the processes, see the *Kofax Communications Manager B&OM Getting Started Guide*.

Add custom service hosts

You can add custom KCM B&OM service hosts using the script `AddOutputManagementService.ps1`. New services added using this script execute selected processes in a given system and can be independently managed using Service management tools. They depend on the standard KCM B&OM Core service to run.

Note the following when adding custom service hosts:

- A new service is automatically started and does not require a restart of B&OM or other components of KCM.
- A new executable is created in the KCM `Output Management` programs directory for the new service with the following name: **CCM_OM_SVC_<Identifier>.exe**. A `.config` file is created next to it, which you can customize, if required.
- By default, service events for the new service are logged to the **CCM_OM_<Identifier>_<Version>_wcf.svclog** file, and all other messages are logged to the **CCM_OM_<Identifier>_<Version>_log.txt** file. If starting the service fails, the log files contain information about the failure.

The following table lists the parameters that you need to add custom service hosts.

Parameter	Description
<code>Service!Identifier</code>	Identifier for the new service. The value is used for identifying the service within the KCM B&OM installation. Also, the value is part of the name, default display name, and default description of the Windows service.
<code>Service!DisplayName</code>	Optional. Overrides the default generated display name of the new Windows service. Default is <code>CCM Output Management - <Identifier> - <Version></code> .
<code>Service!Description</code>	Optional. Overrides the description of the new Windows service generated by default. Default is the same value as the <code>Service!DisplayName</code> parameter has.
<code>Service!User</code>	The user account to be used for the new service. This must be the same user as specified in the original KCM deployment, and must consequently have the "Logon as a service" rights. When specifying a local user, place a period and backslash (<code>.\</code>) at the beginning of the parameter value.
<code>Service!Password</code>	The password for the service user.
<code>System!Name</code>	The name of the KCM B&OM system for the new service to use. You can view the names in the System Administration section in KCM Studio.

Parameter	Description
Process!Names	<p>Optional. A colon-separated list of names for KCM B&OM processes for the new service to execute. You can view the names in the Object Viewer pane in KCM Studio.</p> <p>Note The separator is the regular colon character (:), not the semicolon character (;). If this parameter is not given, AliasTable!Names must be specified.</p>
AliasTable!Names	<p>Optional. A colon-separated list of names for KCM B&OM alias tables, which should contain one or more KCM B&OM processes.</p> <p>The service executes each process contained in these alias tables. The names of the keys defined in the alias tables are not taken into account. You can view the names of the alias tables in the Object Viewer pane in KCM Studio.</p> <p>Note The separator is the regular colon character (:), not the semicolon character (;). If this parameter is not given, Process!Names must be specified.</p>

1. Navigate to: <deploy root>\KCM\Programs\<version>\Management
2. Services are managed using an identifier. Select an identifier to represent the new service.
Example Registration
3. Execute the script.

Use the following script substituting the example parameter values with your actual values.

```
.\AddOutputManagementService.ps1 Service!Identifier=Registration
Service!DisplayName=MyService "Service!Description=Runs my
processes" Service!User=.myuser Service!Password=pwd123
System!Name=MySystem Process!Names=MyProcess:MySecondProcess
AliasTable!Names=MyTable:MySecondTable
```

List custom service hosts

You can get an up-to-date list of custom KCM B&OM service hosts in the current KCM installation using the script ListOutputManagementServices.ps1.

This list contains the identifiers, names, display names, and current status of each service. Also, it includes information on the custom services that failed to be added, so you can remove and re-add them.

This script has no parameters.

1. Navigate to: <deploy root>\KCM\Programs\<version>\Management
2. Execute the script.

Remove custom service hosts

You can remove custom KCM B&OM service hosts using the script `RemoveOutputManagementService.ps1`.

Note

- The removed service is automatically stopped, and the removal does not require a restart of B&OM or other parts of KCM.
- Some custom configuration applied to the service executable `.config`, such as log levels, are lost after the service is removed. Log files themselves are kept.

The script has one parameter.

Parameter	Description
<code>Service!Identifier</code>	Identifier of the service to remove. A list of services can be retrieved using the script <code>ListOutputManagementServices.ps1</code> (see the previous section).

1. Navigate to: `<deploy root>\KCM\Programs\<version>\Management`
2. Services are managed using an identifier. Specify the identifier of the service to remove.
Example `Registration`
3. Execute the script.
Use the following script substituting the example parameter values with your actual values.

```
.\RemoveOutputManagementService.ps1
Service!Identifier=Registration
```

Required step to run KCM B&OM processes

To run KCM B&OM processes, you need to apply the Microsoft 2954953 fix, which is available from the Microsoft support web page: <https://support.microsoft.com>.

On the website, search for the Knowledge Base article "2954953" to find "Some APIs do not work when they are called in services in Windows," and follow the instructions.

Export and import KCM B&OM content

This section provides instructions on using scripts to export and import KCM B&OM content.

Export KCM B&OM content

To facilitate one-time transfer between different installations of KCM B&OM and the upgrade to a newer KCM B&OM version, you can use the `ExportOutputManagementContent.ps1` script. The exported content consists of systems and all their objects.

This script is not intended to provide an alternative to the regular Transfer Jobs option accessible from KCM Studio. Optionally, you can filter the export to only include a specific system, or to only export systems from a specific content database.

Note the following when using the script:

- It does not require KCM B&OM to be running at the time of the export.
- It can be used to export content from KCM B&OM 5.0 or 5.1 installed without any other KCM components. In this case, you can execute the script from an extracted package directory, without performing the deployment.

The following table lists the parameters that you need to perform the export.

Parameter	Description
<code>Destination!Path</code>	The path to a folder to store the export. It must be empty if it already exists. If it does not exist, it is automatically created.
<code>System!Name</code>	Optional. The name of a specific system to export. If this parameter is specified, the export will only contain this system. You cannot use it with the <code>Legacy!Export</code> parameter. By default, no filter is applied.
<code>RepositoryAlias!Name</code>	Optional. The name of a specific content database alias to export. If this parameter is specified, the export will only contain systems from this content database. Only useful when exporting from a KCM B&OM 5.0 or 5.1 installation. By default, no filter is applied.
<code>Legacy!Export</code>	Optional. Boolean. Specifies whether to create the export from KCM B&OM 5.0 or 5.1 on the machine. If specified, an installation is located, and the export is created. No active KCM package is required in this case. Default is false.

1. Navigate to: `<deploy root>\KCM\Programs\<version>\Management`
Or, when exporting from a standalone KCM B&OM 5.0 or 5.1 installation, navigate to the extracted KCM package directory.
2. Select a location to save the export. This location must either not yet exist, or be empty.
3. Execute the script.
Use the following script substituting the example parameters values with your actual values.

```
.\ExportOutputManagementContent.ps1 Destination!Path=C:\Export
System!Name=MySystem
```

Import KCM B&OM content

To import the KCM B&OM content, use the `ImportOutputManagementContent.ps1` script. You can import the following content:

- A full-systems export created with the `ExportOutputManagementContent.ps1` script
- An export to a disk created with the Transfer Job option in KCM Studio

This script does not require KCM B&OM to be running at the time of the import.

The following table lists the parameters that you need to perform the import.

Parameter	Description
<code>Source!Path</code>	The path to a folder containing an export.
<code>System!Name</code>	Optional. Specifies the name of the system to contain a Transfer Job export. Required if the source directory contains a Transfer Job export. By default, the directory is assumed to have a full-systems export retrieved with the <code>ExportOutputManagementContent.ps1</code> script.

1. Navigate to: `<deploy root>\KCM\Programs\<version>\Management`
2. Execute the script.

Use the following script substituting the example parameters values with your actual values.

```
.\ImportOutputManagementContent.ps1 Source!Path=C:\Import
```

Chapter 5

Other settings

Enable authentication and authorization for Contract Manager

You can enable authentication and authorization for the Contract Manager through the following PowerShell script:

```
.\SetCMAuthentication.ps1 ContractManager!UseAuthentication=True
```

The `SetCMAuthentication.ps1` script resides in: `<deploy root>\KCM\Programs\5.3.0\Management`

Authentication can be disabled again by calling this script with `ContractManager!UseAuthentication=False` as its argument.

After enabling authentication and authorization for the Contract Manager, it only grants access applications to authorized SOAP calls after they are properly authenticated. For information on how to configure authentication and authorization for a particular application, see the section "Manage calling applications and their access to Contract Manager interfaces" in the *Kofax Communications Manager Getting Started Guide*.

Note This mechanism uses HTTP basic authentication. This means that you should only use authentication and authorization in combination with an encrypted connection. For more details, see information on the `ContractManager!UseSSL` parameter in [Deployment parameters](#).

Enable SSL for connections to the Contract Manager

We advise that you enable SSL for connections to the Contract Manager. To do so, configure the connectors for the service Catalina-CM.

You can find the configuration for these connectors in the Tomcat `server.xml` file, which resides in the Tomcat installation folder under `instance-CCMRuntime-5.3.0\conf`. For more information, see the Tomcat documentation available on the Internet.

You should at least specify the following:

```
<Service name="...">
  <Connector
    port={HTTPS Port for the Contract Manager web service}
    protocol="org.apache.coyote.http11.Http11NioProtocol"
    SSLEnabled="true"
    maxThreads="150"
    scheme="https"
    secure="true"
```

```

    clientAuth="false"
    sslEnabledProtocols="TLSv1.1,TLSv1.2"
    keystoreFile={The path to a key store file}
    keystorePass={The password for the key store file}
    keystoreType="PKCS12"
    connectionTimeout="20000"
    useServerCipherSuitesOrder="true"
    ciphers="TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384, ...
            TLS_EMPTY_RENEGOTIATION_INFO_SCSV"
  />
  ...
</Service>

```

The `KeyStoreFile` parameter specifies the location of a certificate file (.pfx or .p12). The `KeyStore` password is the password needed for the `KeyStore` File.

The `maxTreads` and `connectionTimeout` values are given as examples.

For security reasons, do not include `TLSv1.0` in the `sslEnabledProtocols` and avoid weak ciphers. Only list ciphers of at least 128 bits. Examples are shown here.

```

TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384,
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384,
TLS_DHE_DSS_WITH_AES_256_CBC_SHA256,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA,
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA,
TLS_DHE_DSS_WITH_AES_256_CBC_SHA,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256,
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256,
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA,
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA,
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA,
TLS_DHE_DSS_WITH_AES_128_CBC_SHA,
TLS_ECDHE_ECDSA_WITH_RC4_128_SHA,
TLS_ECDH_ECDSA_WITH_RC4_128_SHA,
TLS_ECDH_RSA_WITH_RC4_128_SHA,
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,
TLS_RSA_WITH_AES_256_GCM_SHA384,
TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384,
TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384,
TLS_DHE_DSS_WITH_AES_256_GCM_SHA384,
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,
TLS_RSA_WITH_AES_128_GCM_SHA256,
TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256,
TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256,
TLS_DHE_DSS_WITH_AES_128_GCM_SHA256,
TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA,
TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA,
TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA,
TLS_EMPTY_RENEGOTIATION_INFO_SCSV

```

After changing this configuration, restart the "Apache Tomcat CCMRuntime instance 5.3.0" Windows service.

Change an account for the WebServer services

The package installer configures the Tomcat services to run under a specified user account. After installation, you can modify the used account with the `ChangeWebserverServiceUser` script, as shown here:

```
.\ChangeWebserverServiceUser.ps1 Webserver!ServiceUser=. \myuser  
Webserver!ServicePassword=myPassword
```

The `ChangeWebserverServiceUser.ps1` script resides in: `<deploy root>\KCM\Programs\5.3.0\Management`

Verify that the specified account has Logon as a Service rights.

Configure request rate limitation

If it is required to shield the server on which KCM is installed from (D)DOS attacks, you should consider enabling a request rate limiting policy. KCM does not provide a mechanism for such a policy, so this should be handled by setting up a firewall or proxy server that is capable of request rate limiting between the KCM installation and the outside network.

All communication with the KCM installation is either HTTP or HTTPS.

Install and configure KCM Toolbox for Word

KCM Toolbox for Word is an add-in to Microsoft Word that gives you the ability to easily add Fields to Rich Text Blocks and Quick Templates in KCM Designer. The functionality is supported on Microsoft Word 2013 and later, and only for the DOCX documents.

Requirements

- KCM Toolbox for Word requires a web server.
 - The `manifest.xml` file for the Toolbox must be stored in a shared location accessible to all users and listed in the Trusted App Catalogs section in Microsoft Word.
1. To install KCM Toolbox for Word, in your KCM installation, navigate to the `<deploy root>\KCM\Programs\5.3.0\ITPMDKRepositoryServer\ccmtoolboxforword\webapp` directory, copy its content, and then deploy it on your web server. Make a note of the URL used to access the application.
 2. Copy the file `ccmforword manifest-5.3.0.xml` from `<deploy root>\KCM\Programs\5.3.0\ITPMDKRepositoryServer\ccmtoolboxforword\manifest` to the shared location listed in the Trusted App Catalogs section in Microsoft Word.

3. In the file `ccmforward manifest-5.3.0.xml`, specify values for the `<AppDomain>` node and the `DefaultValue` attribute of the `<SourceLocation>` node:
 - In the `<AppDomain>` node, specify the URL to access the application.
 - In the `DefaultValue` attribute of `<SourceLocation>`, specify the URL to access the application and add `/app/index.html` to it.
4. Configure a shared location for the `ccmforward manifest-5.3.0.xml` file using the `/Toolboxpath` flag in ManageCM. The location must be configured for each contract where you want to use KCM Toolbox for Word. For more information, see the sections "Create a contract" and "Change properties of a contract" in the *Kofax Communications Manager Getting Started Guide*.

For information on using KCM Toolbox for Word in KCM Designer, see the KCM Designer online Help.

Chapter 6

Use of Microsoft Word by KCM Core

KCM Core uses Microsoft Word for a number of operations such as printing and conversion of documents. To enable these operations, Microsoft Word must be installed locally on the servers that run KCM Document Processors.

Note You do not need Microsoft Word for conversion to PDF if the conversion is done with the Rendition technology. For more information on this technology, see the section "DocToPDF" in the *Kofax Communications Manager Core Scripting Language Developer's Guide*.

You should disable grammar and spelling tools in Microsoft Word. For more information, see [Adjust settings in Microsoft Word](#).

Also, Microsoft Word requires the presence of a Desktop directory. The location of this directory depends on the versions of Microsoft Word and Microsoft Windows and can be one of the following locations:

- In case of the 64-bit versions of Microsoft Word, create the following directory: %WINDIR%\System32\config\systemprofile\Desktop
- In case of the 32-bit versions of Microsoft Word and the 32-bit versions of Microsoft Windows, create the following directory: %WINDIR%\System32\config\systemprofile\Desktop
- In case of the 32-bit versions of Microsoft Word and the 64-bit versions of Microsoft Windows, create the following directory: %WINDIR%\SysWOW64\config\systemprofile\Desktop

KCM Core attempts to create this directory if it does not exist yet. If it fails, KCM Core is unable to open Microsoft Word documents.

Note Microsoft Word must be started interactively at least once under the user account used for KCM Core Document Processors. This initializes Microsoft Word for that account so that it can be used by the Document Processors. If this is omitted, certain Core Script calls, such as DocToPDF, may fail.

Adjust settings in Microsoft Word

Open Microsoft Word and check if the following settings are adjusted correctly.

- Navigate to **File > Options > Display > Printing Options** and select **Update fields before printing**. Click **OK**.
- Navigate to **File > Options > Advanced > Save** and clear **Allow background saves**. Click **OK**.
- Navigate to **File > Options > Proofing > AutoCorrect Options** and clear everything on all tabs. Click **OK**.

Chapter 7

Uninstallation

To uninstall KCM 5.3.0, navigate to <deploy root> and execute the following command. Place the command on a new line.

```
.\KCM\Programs\5.3.0\Uninstall\Uninstall.ps1  
Database!User=itpuser Database!Password=pwd123
```

This command removes all Kofax Communications Manager 5.3.0 related programs, files, services, and registry entries.

Note The content of the KCM Repository and KCM Batch & Output Management databases is not removed when an instance is uninstalled.

Uninstall version 4.4

The KCM package distributes a command to uninstall KCM 4.4. The command is not installed with the package, but it is available as `InstallServer\Uninstall-4.4\Uninstall-4.4.ps1`.

1. To uninstall KCM 4.4, navigate to <deploy root> and execute the following command.

```
.\InstallServer\Uninstall-4.4\Uninstall-4.4.ps1
```

2. To clear the database for further reinstallation, add the switch `-clearContractManagerDatabase` to the uninstallation command.

To clear the database, provide the database password by adding `-databasePassword <password>` to the uninstallation command.

Chapter 8

Troubleshooting

This chapter gives troubleshooting tips that may be useful if you encounter an issue while using the product.

Package cannot connect to database

If the package cannot connect to the database using a SQL Server, verify that SQL Server authentication is enabled, and that SQL Server has been restarted.

Also, the SQL Server user has a password policy that should be set to "never expire."

Services cannot be created

If the package fails when creating a service, it may be unable to find the user. If a local user is used, verify that the value for the parameter `Services!User` starts with a period and backslash (.).

Cannot start KCM Core for development or test license

The license document and the XML file required for the package differ. The environment for KCM Core should be set to "Development and Test." Otherwise, the KCM Core service cannot be started due to an invalid license.

Warning messages

During the deployment, the KCM package may wait for a certain service to stop. This is indicated by warning messages similar to the following:

```
WARNING: Waiting for service '{serviceName}' to finish stopping
```

Deployment log files are empty

The deployment scripts can fail to write any content to the log files on Windows Server 2012 R2 if the Windows update 2919355 is installed. To resolve this issue, the Windows 3014136 Hotfix must be applied, which is available from the Microsoft support web page: <https://support.microsoft.com>

On the web site, search for the Knowledge Base article "3014136" to find "PowerShell transcript file doesn't contain the correct information in Windows Server 2012 R2," and then download the Hotfix.

CMRestAPI does not start during system startup

The CMRestAPI service can fail to start during the Windows startup sequence. In this situation, the Service Control Manager logs an event 7000 with the error "The service did not respond to the start or control request in a timely fashion."

This error occurs if the system is underprovisioned causing the loading of the service to exceed the default 30,000 milliseconds, which results in the Service Control Manager terminating the service. To resolve this, assign more resources to the system or increase the startup timeout for services to an appropriate value.

You can find the instructions to change the service timeout on the Microsoft support web page: <https://support.microsoft.com>.

On the web site, search for the Knowledge Base article "922918" to find "A service does not start, and events 7000 and 7011 are logged in Windows Server...," and follow the instructions.