PAPER SOFTWARE

Contract Tools

Administrator Guide

Paper Software LLC
papersoftware.com

2024-04-24

Contents

Chapter 1

Getting Started

Learn what you need to know before deploying Contract Tools. Find additional resources on the internet.

This guide describes how to configure and deploy Contract Tools. To understand this guide, you should be familiar with entering commands in PowerShell or Command Prompt. If you're not familiar with either topic, search the internet for "Windows PowerShell" or "Windows Command Prompt" to find many sources of information.

This guide also describes how to use Contract Tools License Server. If you're planning to administer Contract Tools License Server using your organization's computing resources (instead of using a hosted license server):

- You should be familiar with reading and editing XML files. If you're not familiar with this topic, search the internet for "XML tutorial" to find many sources of information.

- You should configure and install Contract Tools License Server before deploying Contract Tools; see Chapter 2, "Contract Tools License Server" on the next page to learn how.

If you're using a hosted license server, or an activation key instead of a license server, you can skip to Chapter 3, "Contract Tools" on page 11 to learn how to deploy Contract Tools.

Visit https://papersoftware.com/support to find many resources to help you get started with Contract Tools.

Chapter 2

Contract Tools License Server

Configure and install Contract Tools License Server. See how many licenses are available from the server. Update the server to a new version. Uninstall the server.

To learn about	Go to
The contents of the Contract Tools folder	Section 2.1, "What's in the Contract Tools Folder?" on this page
The contents of the Contract Tools License Server folder; license server prerequisites	Section 2.2, "What's in the Contract Tools License Server Folder?" on the next page
License server configuration settings	Section 2.3, "Configuring Contract Tools License Server" on the next page
How to install the license server	Section 2.4, "Installing Contract Tools License Server" on page 8
Finding information about the license server, including the number of licenses available	Section 2.5, "Monitoring Contract Tools License Server" on page 9
Replacing license server executables with updated versions	Section 2.6, "Updating Contract Tools License Server" on page 9
Removing the license server from a computer	Section 2.7, "Uninstalling Contract Tools License Server" on page 9
What to do if an error message appears when using license server scripts	Section 2.8, "Opening an Administrator PowerShell or Command Prompt" on page 10
How Contract Tools License Server relates to TurboFloat Server	Section 2.9, "Contract Tools License Server Is TurboFloat Server" on page 10

2.1   What's in the Contract Tools Folder?

The Contract Tools folder contains files you can use to configure and install Contract Tools.

PaperSoftwareContractTools.msi is an MSI file you can use to install Contract Tools; see Section 3.5, "Installing Contract Tools" on page 18 to learn more.

PaperSoftwareContractTools.admx and PaperSoftwareContractTools.adml are administrative template files you can use to configure Contract Tools using Group Policy; see Section 3.3, "Configuring Contract Tools" on page 13 to learn more.

The Contract Tools folder also contains the Contract Tools License Server folder; see Section 2.2, "What's in the Contract Tools License Server Folder?" on this page to learn more.

You can move the Contract Tools folder wherever you'd like.

2.2   What's in the Contract Tools License Server Folder?

The Contract Tools License Server folder contains scripts and other files you can use to configure and install Contract Tools License Server.

configuration.xml is an XML file that contains configuration settings for Contract Tools License Server; see Section 2.3, "Configuring Contract Tools License Server" on this page to learn more.

stop.cmd and start.cmd are scripts you can use to stop and restart Contract Tools License Server when updating it; see Section 2.6, "Updating Contract Tools License Server" on page 9 to learn more.

TurboActivate.dat is a file that Contract Tools License Server reads when it runs. You must not edit this file.

The Contract Tools License Server folder also contains folders named x64 and x86. Each folder contains an executable named Contract Tools License Server.exe. Use the executable in the x64 folder on 64-bit operating systems, and the executable in the x86 folder on 32-bit operating systems.

Note: Both executables require Windows Vista or later, or Windows Server 2008 or later.

The x64 and x86 folders also contain identical copies of two scripts, install.cmd and uninstall.cmd.

install.cmd is a script you can use to install Contract Tools License Server; see Section 2.4, "Installing Contract Tools License Server" on page 8 to learn more.

uninstall.cmd is a script you can use to stop Contract Tools License Server from running when your computer starts; see Section 2.7, "Uninstalling Contract Tools License Server" on page 9 to learn more.

You can move the Contract Tools License Server folder wherever you'd like. However, it's not a good idea to move the files in the x64 and x86 folders to other folders. If you do this, scripts like install.cmd may not work.

2.3   Configuring Contract Tools License Server

You configure Contract Tools License Server by editing the XML file *configuration.xml* in the Contract Tools License Server folder. Here are the default contents of configuration.xml:

```
1   <?xml version="1.0" encoding="UTF-8"?>
2   <config>
3   <bind port="4"/>
4   <lease length="1800"/> <!-- seconds -->
5   <log file="Contract Tools License Server.log" level="notification"/>
6   <isgenuine days_between="90" grace="14"/> <!-- days -->
7   <cpu threads="0"/> <!-- 0 means one child thread per CPU core -->
8   <!--
9   <proxy url="http://username:password@127.0.0.1:8080/"/>
10  -->
11  <!--
12  <scgi just_scgi="false">
13    <bind address="127.0.0.1" port="42"/>
14  </scgi>
15  -->
16  </config>
```

You may need to change the `port` attribute of the `bind` element in line 3. You generally don't need to change anything else.

`<bind port="4"/>` sets the port on which Contract Tools License Server communicates. You also use this port to configure Contract Tools; see Section 3.3.3, "Connecting to Contract Tools License Server" on page 15 to learn more. Using default settings, Contract Tools License Server communicates on port 4.

`<lease length="1800"/>` sets the amount of time a license provided by Contract Tools License Server to Contract Tools is valid. Using default settings, licenses are valid for 1800 seconds (30 minutes), and Contract Tools will begin attempting to extend the duration of its license when about 15 minutes remain. You can use a minimum time of 30 seconds, but short times may increase traffic on your network to an unacceptable level.

`<log file="Contract Tools License Server.log" level="notification"/>` sets the path of the file to which Contract Tools License Server logs messages and the verbosity level of these messages. If the value of the `file` attribute is a relative path, the log file is placed relative to Contract Tools License Server.exe. The value of the `level` attribute can be, in order of increasing verbosity, `error`, `warning`, or `notification`. Using default settings, messages are logged in a file named *Contract Tools License Server.log* in the same folder as Contract Tools License Server.exe, and this file contains errors, warnings, and notifications.

`<isgenuine days_between="90" grace="14"/>` sets the number of days between Contract Tools License Server's periodic verifications of your license server activation key. Using default settings, the number of days is at least 90 and at most 104. The value of the `days_between` attribute can be any whole number up to 90, and the value of the `grace` attribute can be any whole number up to 14.

`<cpu threads="0"/>` sets the number of child threads Contract Tools License Server spawns. If this number is 0, Contract Tools License Server spawns one child thread per CPU core. Using default settings, if your computer's CPU has 8 cores, Contract Tools License Server will use a total of 9 threads.

The `proxy` element in line 9 and the `scgi` element in lines 12–14 are in comments because they're optional. Use a `proxy` element and set its `url` attribute to a value appropriate for your

proxy to allow Contract Tools License Server to verify your license server activation key through a proxy. Use an `scgi` element to configure Contract Tools License Server to communicate using HTTPS; visit https://wyday.com/limelm/help/turbofloat-server/#config-https to learn how.

2.4   Installing Contract Tools License Server

After you configure Contract Tools License Server (see Section 2.3, "Configuring Contract Tools License Server" on page 6 to learn how), you can install it by running the script *install.cmd*. To run this script, open an administrator PowerShell or Command Prompt in the x64 folder on a 64-bit operating system or the x86 folder on a 32-bit operating system, and then enter:

```
.\install
```

This script:

1. Activates Contract Tools License Server by sending your license server activation key over the internet

2. Configures Windows Firewall to allow inbound connections to Contract Tools License Server

3. Runs Contract Tools License Server as a Windows service

4. Installs Contract Tools License Server to run as a Windows service if the computer on which it's installed is restarted

If an error message that says "Failed to open the log file for writing" or "Failed to create the registry key in Software\TurboFloatServer" appears, make sure you're using an administrator PowerShell or Command Prompt; see Section 2.8, "Opening an Administrator PowerShell or Command Prompt" on page 10 to learn more.

To verify that Contract Tools License Server is running, enter in PowerShell or Command Prompt:

```
sc.exe qc TurboFloatServer-2942
```

If Contract Tools License Server is running, `sc.exe` will output the configuration of Contract Tools License Server.

The service name `TurboFloatServer-2942` contains `TurboFloatServer` because Contract Tools License Server.exe is a renamed executable from TurboFloat Server; see Section 2.9, "Contract Tools License Server Is TurboFloat Server" on page 10 to learn more. The number 2942 is an identifier for Contract Tools.

If Contract Tools License Server is not running, examine the license server log file to diagnose the problem. See Section 2.3, "Configuring Contract Tools License Server" on page 6 to learn how to configure the log file.

Note: Contract Tools License Server must be installed on real hardware, not on a virtual machine. If you believe you're installing Contract Tools License Server on real hardware, but error messages appear that say the server failed to activate because your computer is a virtual machine, make sure that Windows virtualization features are disabled. Examples of Windows virtualization features include Hyper-V Platform, Memory Integrity, Virtual Machine Platform, Windows Defender Application Guard, and Windows Hypervisor Platform. If disabling Windows virtualization features doesn't resolve the issue, make sure that virtualization is disabled on

your computer's processor; your computer's documentation should have instructions on how to disable virtualization on your computer's processor.

If you cannot install Contract Tools License Server on real hardware, email info@papersoftware.com to learn about obtaining a hosted license server. For detailed information about hosted license servers, see Appendix B, "Hosted License Servers" on page 30.

After you install Contract Tools License Server, you can configure Contract Tools to connect to the server by modifying Registry values; see Section 3.3.3, "Connecting to Contract Tools License Server" on page 15 to learn more.

2.5   Monitoring Contract Tools License Server

To find information about Contract Tools License Server, including how many licenses are available from the server, examine the license server log file. See Section 2.3, "Configuring Contract Tools License Server" on page 6 to learn how to configure the log file.

The license server log file shows information about Contract Tools License Server, but not about a Contract Tools activation key. See Section 3.8, "Monitoring Activations" on page 22 to learn how to find information about a Contract Tools activation key.

Note: The license server log file is not available if you're using a hosted license server.

2.6   Updating Contract Tools License Server

Occasionally, you'll need to update Contract Tools License Server. It's a good idea to do this when usage of Contract Tools is at a minimum.

To update Contract Tools License Server:

1. Open an administrator PowerShell or Command Prompt in the Contract Tools License Server folder.

2. To stop Contract Tools License Server, enter:

   ```
   .\stop
   ```

3. Replace the Contract Tools License Server executables with new versions.

4. To restart Contract Tools License Server, enter:

   ```
   .\start
   ```

   If an error message that says "Access is denied" appears, make sure you're using an administrator PowerShell or Command Prompt; see Section 2.8, "Opening an Administrator PowerShell or Command Prompt" on the next page to learn more.

2.7   Uninstalling Contract Tools License Server

To uninstall Contract Tools License Server, open an administrator PowerShell or Command Prompt in the x64 folder on a 64-bit operating system or the x86 folder on a 32-bit operating system, and then enter:

```
.\uninstall
```

This script:

1. Stops Contract Tools License Server

2. Removes Contract Tools License Server configuration settings from Windows Firewall if possible

3. Removes Contract Tools License Server from the list of Windows services so that it does not run as a background process if the computer on which it is present is restarted

4. Deactivates Contract Tools License Server

If an error message that says "Failed to open the log file for writing" or "Remove failed" appears, make sure you're using an administrator PowerShell or Command Prompt; see Section 2.8, "Opening an Administrator PowerShell or Command Prompt" on this page to learn more.

Uninstalling Contract Tools License Server does not delete the Contract Tools License Server folder from your computer. To delete the Contract Tools License Server folder, move it to the Recycle Bin, and then empty the Recycle Bin.

2.8 Opening an Administrator PowerShell or Command Prompt

The Contract Tools License Server scripts (install.cmd, stop.cmd, start.cmd, and uninstall.cmd) require an administrator PowerShell or Command Prompt. To open an administrator PowerShell from a non-administrator PowerShell, enter:

```
$arguments = "-NoExit -Command `"Set-Location \`"$(Get-Location)\`"`""
Start-Process powershell $arguments -Verb runas
```

To open an administrator Command Prompt from a non-administrator Command Prompt, enter:

```
powershell -Command "Start-Process cmd \"/k cd /d %CD%\" -Verb runas"
```

2.9 Contract Tools License Server Is TurboFloat Server

TurboFloat Server is part of LimeLM by wyDay, LLC. The Contract Tools License Server executables are the executables distributed with TurboFloat Server, renamed to identify them as Contract Tools' license server in case you administer other software that uses TurboFloat Server.

This Contract Tools License Server file	Is this TurboFloat Server file, renamed
Contract Tools License Server.exe	TurboFloatServer.exe
configuration.xml	TurboFloatServer-config.xml

Visit https://wyday.com/limelm/help/turbofloat-server/ to learn more about TurboFloat Server.

Chapter 3

Contract Tools

Learn prerequisites for installing Contract Tools. Configure proxies and firewalls. Configure and install Contract Tools. Share standard text as snippets. Find information about an activation key. Disconnect from a license server. Uninstall Contract Tools.

To learn about	Go to
Contract Tools' requirements	Section 3.1, "Prerequisites for Installing Contract Tools" on the next page
Configuring proxies and firewalls for activating and updating Contract Tools	Section 3.2, "Configuring Proxies and Firewalls" on the next page
Contract Tools configuration settings	Section 3.3, "Configuring Contract Tools" on page 13
Compatibility with iManage Office integrations	Section 3.4, "Configuring iManage Software" on page 17
Installing Contract Tools using an MSI file or an executable	Section 3.5, "Installing Contract Tools" on page 18
Using snippets to distribute standard text to users	Section 3.6, "Sharing Text as Snippets" on page 20
Adding a certificate to lists of trusted publishers	Section 3.7, "Adding Paper Software as a Trusted Publisher" on page 21
Finding the number of activations available on a Contract Tools activation key	Section 3.8, "Monitoring Activations" on page 22
Freeing up activations to use on other computers	Section 3.9, "Deactivating Contract Tools" on page 22
Releasing licenses back to Contact Tools License Server	Section 3.10, "Disconnecting from Contract Tools License Server" on page 23
Removing Contract Tools from computers	Section 3.11, "Uninstalling Contract Tools" on page 23

3.1 Prerequisites for Installing Contract Tools

On all computers, Contract Tools requires Windows 7 or later, Word 2007 or later, and Visual Studio 2010 Tools for Office Runtime.

On computers with Windows 7 and 8.1, Contract Tools requires .NET Framework with a version between 4.5.2 and 4.8. On computers with Word 2007, Contract Tools also requires .NET Framework 3.5.

To download an installer for this	Visit
Visual Studio 2010 Tools for Office Runtime	https://www.microsoft.com/en‐us/download/details.aspx?id=105890
.NET Framework 4.5.2	https://www.microsoft.com/en‐us/download/details.aspx?id=42643
.NET Framework 3.5	https://www.microsoft.com/en‐us/download/details.aspx?id=21

To activate Contract Tools using an activation key (see Section 3.3.2, "Activating Contract Tools" on the next page), Windows, Word, and Contract Tools must be installed on real hardware, not on a virtual machine. Examples of virtual machine software include Citrix Virtual Apps and Desktops, Hyper‐V, Microsoft 365 Defender, Parallels Desktop, Remote Desktop Services, VMware Fusion, Windows 365, Windows Sandbox, and Windows Virtual Desktop. To deploy Contract Tools on virtual machines, you can use Contract Tools License Server; see Chapter 2, "Contract Tools License Server" on page 5 and Section 3.3.3, "Connecting to Contract Tools License Server" on page 15 to learn more.

3.2 Configuring Proxies and Firewalls

You may need to configure proxies and firewalls to activate Contract Tools using an activation key and allow Contract Tools to download updates.

If you're using an activation key instead of a license server to administer Contract Tools (see Section 3.3.2, "Activating Contract Tools" on the next page to learn more), you should configure proxies and firewalls to allow communication from Word and access to wyday.com. (Contract Tools uses LimeLM by wyDay, LLC for license management. LimeLM requires access to wyDay's website, wyday.com.)

If your users will be permitted to install Contract Tools updates individually (see Section 3.3.4, "Updating Contract Tools" on page 16 to learn how to disable updating), you should configure proxies and firewalls to allow communication from PaperSoftwareContractToolsUpdater.exe and access to https://papersoftware.s3.amazonaws.com (Amazon S3). PaperSoftwareContractToolsUpdater.exe is located in Contract Tools' installation folder; see Section 3.5.4, "Installation Folder" on page 20 to learn more. (Contract Tools uses Advanced Updater by Caphyon Ltd. to download and install updates. Contract Tools' Advanced Updater executable is named PaperSoftwareContractToolsUpdater.exe and it downloads updates from Amazon S3.)

3.3   Configuring Contract Tools

Generally, you configure Contract Tools by creating Registry values in the HKEY_CURRENT_USER hive with key path Software\Paper Software\Contract Tools. The only exception is Contract Tools' LoadBehavior Registry value; see Section 3.3.1, "Configuring LoadBehavior" on this page to learn more.

To learn about	Go to
Changing how Word loads Contract Tools	Section 3.3.1, "Configuring LoadBehavior" on this page
Configuring a Contract Tools activation key	Section 3.3.2, "Activating Contract Tools" on this page
Making Contract Tools connect to a license server	Section 3.3.3, "Connecting to Contract Tools License Server" on page 15
Preventing automatic updates	Section 3.3.4, "Updating Contract Tools" on page 16
Disabling saving analysis settings with documents that Contract Tools analyzes	Section 3.3.5, "Saving Analysis Settings" on page 16
How to modify Registry values	Section 3.3.6, "Modifying Registry Values" on page 16
Making Contract Tools turned off by default	Section 3.3.7, "Configuring Contract Tools to Be Turned Off When Documents Open" on page 17

3.3.1   Configuring LoadBehavior

You can change how Word loads Contract Tools by modifying its LoadBehavior Registry value; see *Registry entries for VSTO Add-ins* (https://docs.microsoft.com/en-us/visualstudio/vsto/registry-entries-for-vsto-add-ins) to learn more. Here is where you can find Contract Tools' LoadBehavior Registry value:

- If you install Contract Tools for all users of a computer (a *per-machine* installation), the LoadBehavior Registry value is stored in the HKEY_LOCAL_MACHINE hive.

    - On 64-bit operating systems with a 64-bit version of Office, and 32-bit operating systems with a 32-bit version of Office, the key path is SOFTWARE\Microsoft\Office\Word\Addins\Paper Software.Contract Tools.

    - On 64-bit operating systems with a 32-bit version of Office, the key path is SOFTWARE\WOW6432Node\Microsoft\Office\Word\Addins\Paper Software.Contract Tools.

- If you install Contract Tools for a single user of a computer (a *per-user* installation), the LoadBehavior Registry value is stored in the HKEY_CURRENT_USER hive with key path Software\Microsoft\Office\Word\Addins\Paper Software.Contract Tools.

As an alternative to modifying Contract Tools' LoadBehavior Registry value, you can configure user preferences so that Contract Tools is turned off by default; see Section 3.3.7, "Configuring Contract Tools to Be Turned Off When Documents Open" on page 17 to learn how.

3.3.2   Activating Contract Tools

You can configure an activation key by creating this Registry value:

Name	Type	Data
Activation key	REG_SZ	Contract Tools' activation key

Note: When you configure an activation key using a Registry value, the activation key is hidden from your users.

Because Contract Tools sends and receives information over the internet to become activated using an activation key, you may need to change Contract Tools' proxy settings if your users connect to the internet through a proxy. To do this, create this Registry value:

Name	Type	Data
Proxy URL	REG_SZ	For example, http://username:password@127.0.0.1:8080/

See Section 3.2, "Configuring Proxies and Firewalls" on page 12 for more information about configuring proxies when using an activation key.

As an alternative to using Registry values, your users can activate Contract Tools by using the controls in the Contract Tools pane. Here are instructions you can offer to your users:

To activate Contract Tools:

1. Open a document or create a new document in Word. The Contract Tools pane appears next to your document.

2. Enter Contract Tools' activation key in the Contract Tools pane.

3. Click Proxy Settings, select "Use this proxy", and then enter the proxy URL.

4. Click Activate.

Enter the activation key.

Click Activate.

Click Proxy Settings.

Select "Use this proxy".

Enter the proxy URL.
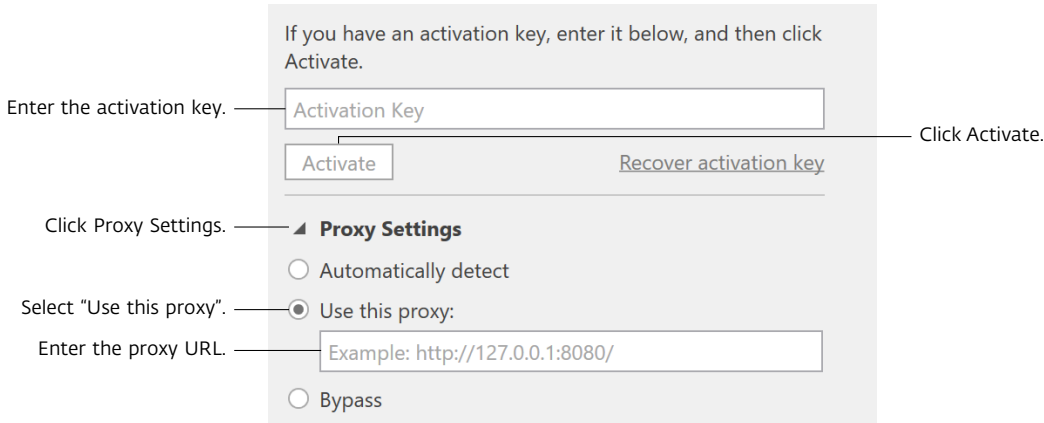
Note: What happens when there are no activations available on an activation key depends on whether the Contract Tools pane or a Registry value is used to specify the activation key. If the Contract Tools pane is used, a user can optionally deactivate Contract Tools on all computers. If a Registry value is used, Contract Tools doesn't offer a way to deactivate on all computers. If you're using an activation key to administer Contract Tools for numerous users, it's a good idea to specify the activation key using a Registry value so that your users can't deactivate Contract Tools on all computers when there are no activations available.

To learn how to find the number of activations available on an activation key, see Section 3.8, "Monitoring Activations" on page 22. To learn how to deactivate Contract Tools, see Section 3.9, "Deactivating Contract Tools" on page 22.

3.3.3   Connecting to Contract Tools License Server

You can configure Contract Tools to connect to Contract Tools License Server by creating these Registry values:

Name	Type	Data
License server IP address	REG_SZ	The address of the computer running Contract Tools License Server
License server port	REG_DWORD	Contract Tools License Server's port

Note: You can set the data of the "License server IP address" Registry value to a domain name instead of an IP address.

For the "License server port" Registry value:

- If you're administering Contract Tools License Server using your organization's computing resources, use the port you set when you configured Contract Tools License Server; see Section 2.3, "Configuring Contract Tools License Server" on page 6 to learn more.

- If you're using a hosted license server, use 443 (communication with a hosted license server uses HTTPS, and port 443 is the standard port for HTTPS communication).

By default, Contract Tools connects to Contract Tools License Server (and starts using a license) when Word loads Contract Tools (see Section 3.3.1, "Configuring LoadBehavior" on page 13 to learn how to configure when Word loads Contract Tools). To make Contract Tools connect to the server only when Contract Tools is turned on in a document window, create this Registry value:

Name	Type	Data
Connect to license server only when enabled	REG_SZ	True

See Section 3.3.7, "Configuring Contract Tools to Be Turned Off When Documents Open" on page 17 to learn how to configure default user preferences so that Contract Tools is turned off when documents open.

As an alternative to using Registry values, your users can connect Contract Tools to Contract Tools License Server by using the controls in the Contract Tools pane. Here are instructions you can offer to your users:

To connect Contract Tools to Contract Tools License Server:

1. Open a document or create a new document in Word. The Contract Tools pane appears next to your document.

2. Choose "Connect to License Server" from the pop-up menu at the top of the Contract Tools pane.

3. Enter Contract Tools License Server's address and port, and then click Connect.

Enter the server's address.

Enter the server's port.

Connect to the server.
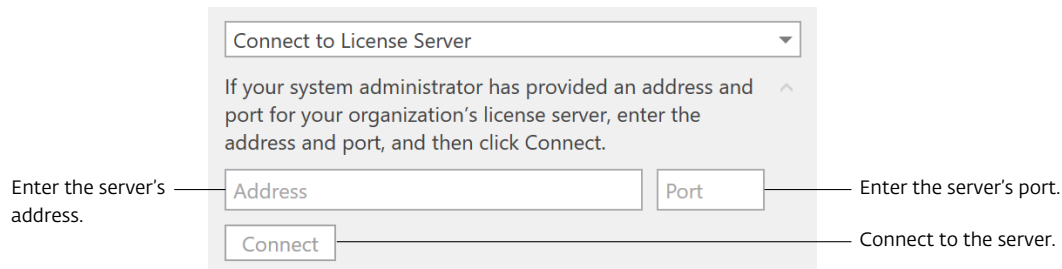
See Section 3.10, "Disconnecting from Contract Tools License Server" on page 23 to learn about disconnecting Contract Tools from Contract Tools License Server.

3.3.4   Updating Contract Tools

By default, Contract Tools checks for updates when it starts, and then gives users the option to install updates. To disable updating, create this Registry value:

Name	Type	Data
Disable updating	REG_SZ	True

If you disable updating, you should sign up to receive email notifications when updates to Contract Tools are released so that you can deploy the latest version of Contract Tools. Visit https://papersoftware.com/contract-tools-admin to sign up.

Note: It's not a good idea to forward email notifications of Contract Tools updates to others. If you do this, another recipient may unsubscribe you from email notifications of Contract Tools updates.

3.3.5   Saving Analysis Settings

In Word 2010 and later, Contract Tools can save analysis settings as custom XML data. To disable saving analysis settings, create this Registry value:

Name	Type	Data
Disable saving analysis settings	REG_SZ	True

3.3.6   Modifying Registry Values

You can use a variety of techniques to modify Contract Tools Registry values using Group Policy.

Here are ways to modify Contract Tools Registry values using Group Policy:

- Use the Group Policy administrative template defined by PaperSoftwareContractTools.admx and PaperSoftwareContractTools.adml (available at https://papersoftware.com/support).

- Use a script. For example, you can disable automatic updating of Contract Tools with this PowerShell script:

```
$keyPath = 'HKCU:\Software\Paper Software\Contract Tools'
if (!(Test-Path $keyPath)) { New-Item $keyPath -Force }
New-ItemProperty $keyPath 'Disable updating' -Value True -Force
```

or this batch script:

16

```
set keyPath=HKCU\Software\Paper Software\Contract Tools
reg add "%keyPath%" /v "Disable updating" /d True /f
```

- Use the Group Policy Management Console; search the internet for "Group Policy Management Console" to learn more.

3.3.7   Configuring Contract Tools to Be Turned Off When Documents Open

By default, Contract Tools turns on automatically when documents open. If your users work with contracts only occasionally, they may prefer to turn on Contract Tools manually. Whether Contract Tools turns on automatically is a user preference, and you can configure default user preferences by modifying PaperSoftwareContractTools.dll.config in Contract Tools' installation folder (see Section 3.5.4, "Installation Folder" on page 20 to learn more) before Word loads Contract Tools for the first time. For example, when Contract Tools is installed for all users on a 64-bit operating system, you can configure Contract Tools to be turned off when documents open with this PowerShell script:

```
$preferenceName = 'EnableAutomatically'
$newDefault = 'False'
$installationFolder = "${Env:ProgramFiles(x86)}\Paper Software\Contract Tools"
$absolutePath = "$installationFolder\PaperSoftwareContractTools.dll.config"
$xml = New-Object Xml.XmlDocument
$xml.Load($absolutePath)
$xml.SelectSingleNode("//setting[@name='$preferenceName']").value = $newDefault
$xml.Save($absolutePath)
```

To learn more about user preferences, see Appendix A, "Contract Tools User Preferences" on page 26.

By default, Contract Tools connects to Contract Tools License Server (and starts using a license) when Word loads Contract Tools, even if Contract Tools is configured to be turned off when documents open (see Section 3.3.1, "Configuring LoadBehavior" on page 13 to learn how to configure when Word loads Contract Tools). You can configure Contract Tools to connect to Contract Tools License Server only when Contract Tools is turned on in a document window; see Section 3.3.3, "Connecting to Contract Tools License Server" on page 15 to learn how.

Your users can turn Contract Tools on or off by clicking the On / Off button in the Contract Tools ribbon tab; see *The Contract Tools Ribbon Tab* (https://contract-tools.helpscoutdocs.com/article/155-the-contract-tools-ribbon-tab) to learn more.

3.4   Configuring iManage Software

An issue with iManage Integration for Office, a Word add-in distributed with some versions of iManage FileSite and DeskSite, can cause Word to temporarily become unresponsive when documents close while using Contract Tools. A solution is available in versions of iManage Integration for Office distributed with FileSite and DeskSite 9.3.5 and later.

When a document closes, iManage Integration for Office sets the property of a *range* (similar to a selection in Word) to the property's current value. In computer programming, this is called a *self-assignment*. A self-assignment typically has no effect, but in this case, it causes Word to perform a time-consuming and unnecessary task with data that Contract Tools uses to sync

analysis results with documents while there are unanalyzed changes. As a result, Word may temporarily become unresponsive.

iManage tracks this issue using reference number NT-72170. For more information, see *Quick Update CSAR-2626 for FileSite and DeskSite 9.3.5 with Office Integration*, available at https://imanage.com.

With versions of iManage Integration for Office distributed with FileSite and DeskSite 9.3.5 and later, you can resolve this issue by creating this Registry value:

Name	Type	Data
RemoveRangeSelfAssignment	REG_SZ	Y

You create this Registry value in the HKEY_LOCAL_MACHINE hive. The key path of this Registry value depends on whether you use 64-bit or 32-bit software.

- If you use the 64-bit version of Windows with 64-bit versions of iManage software, or the 32-bit version of Windows with 32-bit versions of iManage software, the key path is SOFTWARE\Interwoven\WorkSite\8.0\Integration\Options.

- If you use the 64-bit version of Windows with 32-bit versions of iManage software, the key path is SOFTWARE\WOW6432Node\Interwoven\WorkSite\8.0\Integration\Options.

3.5   Installing Contract Tools

You can install Contract Tools with an MSI file (PaperSoftwareContractTools.msi) or an executable (PaperSoftwareContractToolsSetup.exe). Installation resources, including an MSI file and an executable, are available at https://papersoftware.com/support.

To install Contract Tools using Group Policy, you must use an MSI file.

To learn about	Go to
Using an MSI file to install Contract Tools	Section 3.5.1, "Installing with an MSI File" on this page
Using an executable to install Contract Tools	Section 3.5.2, "Installing with an Executable" on the next page
Finding general information about installation settings	Section 3.5.3, "Resources about Installation Parameters" on the next page
Where Contract Tools is installed	Section 3.5.4, "Installation Folder" on page 20

3.5.1   Installing with an MSI File

You can use PaperSoftwareContractTools.msi with Group Policy to install Contract Tools. You can also use PaperSoftwareContractTools.msi with `msiexec` to install Contract Tools. To do this, enter in PowerShell or Command Prompt:

`msiexec /i PaperSoftwareContractTools.msi /q /log setup.log /noRestart`

- `/q` prevents Windows Installer dialogs from appearing.

- `/log setup.log` creates a log file named setup.log in the same folder as PaperSoftwareContractTools.msi.

- /noRestart prevents an automatic computer restart if one is required.

See Section 3.5.3, "Resources about Installation Parameters" on this page for resources that describe additional `msiexec` parameters.

3.5.2   Installing with an Executable

You can use PaperSoftwareContractToolsSetup.exe from PowerShell or Command Prompt to install Contract Tools. Here is a typical installation command:

```
.\PaperSoftwareContractToolsSetup /exeNoUI /q /log setup.log /noRestart
```

The /exeNoUI parameter prevents Contract Tools Setup from appearing. The other parameters are passed to `msiexec`:

- /q prevents Windows Installer dialogs from appearing.

- /log setup.log creates a log file named setup.log in the same folder as PaperSoftwareContractToolsSetup.exe.

- /noRestart prevents an automatic computer restart if one is required.

Letting /• mean any parameter, PaperSoftwareContractToolsSetup.exe commands generally look like:

```
.\PaperSoftwareContractToolsSetup /• /• /• // /• /•
```

PaperSoftwareContractToolsSetup.exe handles these parameters directly.

This is replaced by the path to a temporary MSI file.

These parameters are passed to `msiexec`.

You pass two groups of parameters to PaperSoftwareContractToolsSetup.exe:

1. The first group of parameters is handled directly by PaperSoftwareContractToolsSetup.exe. When installing Contract Tools, this group will typically contain only /exeNoUI.

2. The second group of parameters, beginning with /• //, is passed to `msiexec`. PaperSoftwareContractToolsSetup.exe replaces // with the path to a temporary MSI file.

If you leave out /• //, PaperSoftwareContractToolsSetup.exe uses /i //. For example:

```
.\PaperSoftwareContractToolsSetup /exeNoUI /q
```

is the same as

```
.\PaperSoftwareContractToolsSetup /exeNoUI /i // /q
```

All parameters are optional. Entering `.\PaperSoftwareContractToolsSetup` in PowerShell or Command Prompt starts Contract Tools Setup as if you double-clicked PaperSoftwareContractToolsSetup.exe.

See Section 3.5.3, "Resources about Installation Parameters" on this page for resources that describe additional PaperSoftwareContractToolsSetup.exe parameters.

3.5.3   Resources about Installation Parameters

To learn more about `msiexec` parameters, enter in PowerShell or Command Prompt:

```
msiexec /?
```

To learn more about PaperSoftwareContractToolsSetup.exe parameters, enter:

`.\PaperSoftwareContractToolsSetup /?`

You can also use these resources:

To learn about	See
`msiexec` parameters	*Command-Line Options* (https://docs.microsoft.com/en-us/windows/win32/msi/command-line-options)
	Standard Installer Command-Line Options (https://docs.microsoft.com/en-us/windows/win32/msi/standard-installer-command-line-options)
	Msiexec.exe Command Line (https://www.advancedinstaller.com/user-guide/msiexec.html)
PaperSoftwareContractToolsSetup.exe parameters	*EXE setup file* (https://www.advancedinstaller.com/user-guide/exe-setup-file.html)

3.5.4   Installation Folder

The default location of Contract Tools' installation folder depends on how Contract Tools is installed.

- If Contract Tools is installed for all users of a computer (a *per-machine* installation):

    - On 64-bit operating systems, Contract Tools' installation folder is %ProgramFiles(x86)%\Paper Software\Contract Tools.

    - On 32-bit operating systems, Contract Tools' installation folder is %ProgramFiles%\Paper Software\Contract Tools.

- If Contract Tools is installed for a single user of a computer (a *per-user* installation), Contract Tools' installation folder is %LOCALAPPDATA%\Programs\Paper Software\Contract Tools.

3.6   Sharing Text as Snippets

In Contract Tools, *snippets* are chunks of text that users can quickly add to documents. By default, snippets are limited to those that a user creates. In addition, snippets can optionally include all AutoText that Word loads. If your organization uses standard text in contracts, you can share this text as AutoText in a Word template (DOTX) file, and then your users can work with this text the way they work with snippets.

To share text as snippets:

1. Create a DOTX file that contains the text you want to share as AutoText. Because snippets are AutoText and are saved in a DOTX file, you can use Contract Tools to do this; see *Managing Snippets* (https://contract-tools.helpscoutdocs.com/article/291-managing-snippets) to learn how. You can also use Word's built-in controls; see *Create reusable text snippets* (https://support.microsoft.com/en-us/office/create-reusable-text-snippets-0bc40cab-f49c-4e06-bcb2-cd43c1674d1b) to learn how.

2. Copy the DOTX file to a location that your users' computers can access. If you used Contract Tools to create the DOTX file, you can find the DOTX file using Contract Tools: show the

Contract Tools pane, click the Preferences button, click Snippets File, and then click the "Show in File Explorer" button (looks like a folder).

3. Copy the DOTX file to a folder from which Word loads templates on your users' computers, like Word's Startup folder; see *Load or unload a template or add-in program* (https://support.microsoft.com/en-us/office/load-or-unload-a-template-or-add-in-program-2479fe53-f849-4394-88bb-2a6e2a39479d) to learn more. One way to do this is to deploy a script using Group Policy, for example this PowerShell script:

```
$filePath = '\\Server\Folder\Shared AutoText.dotx'
Copy-Item $filePath "$Env:APPDATA\Microsoft\Word\STARTUP"
```

or this batch script:

```
set filePath=\\Server\Folder\Shared AutoText.dotx
copy /b "%filePath%" "%APPDATA%\Microsoft\Word\STARTUP"
```

4. Communicate to your users that they can use your organization's standard text as snippets by showing the Contract Tools pane, clicking the Preferences button, and then selecting "Include all AutoText and Quick Parts" below Snippets.

Whether snippets include all AutoText and Quick Parts is a user preference, and you can configure default user preferences by modifying PaperSoftwareContractTools.dll.config in Contract Tools' installation folder (see Section 3.5.4, "Installation Folder" on the previous page to learn more) before Word loads Contract Tools for the first time. For example, when Contract Tools is installed for all users on a 64-bit operating system, you can configure snippets to include all AutoText and Quick Parts with this PowerShell script:

```
$preferenceName = 'SnippetsViewShowsAllAutoTextAndQuickParts'
$newDefault = 'True'
$installationFolder = "${Env:ProgramFiles(x86)}\Paper Software\Contract Tools"
$absolutePath = "$installationFolder\PaperSoftwareContractTools.dll.config"
$xml = New-Object Xml.XmlDocument
$xml.Load($absolutePath)
$xml.SelectSingleNode("//setting[@name='$preferenceName']").value = $newDefault
$xml.Save($absolutePath)
```

To learn more about user preferences, see Appendix A, "Contract Tools User Preferences" on page 26.

3.7   Adding Paper Software as a Trusted Publisher

If you require Word add-ins like Contract Tools to be signed by a trusted publisher, you should add Paper Software to lists of trusted publishers.

To add Paper Software to lists of trusted publishers using Group Policy, you can use a certificate that you export as a file; see *Distribute Certificates to Client Computers by Using Group Policy* (https://docs.microsoft.com/en-us/windows-server/identity/ad-fs/deployment/distribute-certificates-to-client-computers-by-using-group-policy) to learn more. To export a certificate from PaperSoftwareContractTools.msi (available at https://papersoftware.com/support) as a file named PaperSoftware.cer, open PowerShell in a folder that contains PaperSoftwareContractTools.msi, and then enter:

```
$installerPath = '.\PaperSoftwareContractTools.msi'
$certificatePath = "$((Get-Item $installerPath).DirectoryName)\PaperSoftware.cer"
$certificate = (Get-AuthenticodeSignature $installerPath).SignerCertificate
[IO.File]::WriteAllBytes($certificatePath, $certificate.Export('Cert'))
```

You can also add Paper Software to a list of trusted publishers by entering in an administrator PowerShell:

```
$computerName = $Env:COMPUTERNAME
$installerPath = '.\PaperSoftwareContractTools.msi'
$store = New-Object Security.Cryptography.X509Certificates.X509Store(
  "$computerName\TrustedPublisher", 'LocalMachine')
$store.Open('ReadWrite, OpenExistingOnly')
$store.Add((Get-AuthenticodeSignature $installerPath).SignerCertificate)
$store.Close()
```

Note: You can change the value of `$computerName` to the name of another computer to add Paper Software to lists of trusted publishers remotely.

3.8   Monitoring Activations

If you're using an activation key to administer Contract Tools, you can use Contract Tools to see how many activations are available.

To see how many activations are available on a Contract Tools activation key:

1. Make sure the activation key is not specified in your Registry (see Section 3.3.2, "Activating Contract Tools" on page 13 to learn more). You cannot use Contract Tools to see how many activations are available on an activation key when it's specified in your Registry. (You *can* use Contract Tools to see how many activations are available on an activation key when it's specified in your users' Registries.)

2. Show the Contract Tools pane in Word.

3. If Contract Tools is deactivated, activate Contract Tools by entering the activation key in the Activation Key field.

4. Click the Subscription button at the top of the Contract Tools pane. Information about the activation key appears at the top of the Subscription view.

   See Section 3.3.2, "Activating Contract Tools" on page 13 to learn about activating Contract Tools using an activation key.

   The Subscription view shows information about a Contract Tools activation key, but not about Contract Tools License Server. See Section 2.5, "Monitoring Contract Tools License Server" on page 9 to learn how to find information about Contract Tools License Server.

3.9   Deactivating Contract Tools

An activation key can be used to activate Contract Tools on a certain number of computers (see Section 3.8, "Monitoring Activations" on this page to learn how to find this number). You can deactivate Contract Tools on some computers to free up activations for use on other computers.

To deactivate Contract Tools:

- Show the Contract Tools pane, click the Subscription button, and then click Deactivate.

  Note: Uninstalling Contract Tools does not deactivate Contract Tools. Before you uninstall Contract Tools, you should deactivate it. See Section 3.11, "Uninstalling Contract Tools" on this page to learn how to uninstall Contract Tools.

  If you need to deactivate Contract Tools on computers that you don't have access to, email support@papersoftware.com.

  See Section 3.3.2, "Activating Contract Tools" on page 13 to learn about activating Contract Tools using an activation key.

3.10   Disconnecting from Contract Tools License Server

Contract Tools automatically disconnects from Contract Tools License Server (and stops using a license) when a user quits Word, locks their session, or signs out of Windows; and when a computer goes to sleep (for example, when the lid of a notebook computer is closed). (Contract Tools attempts to reconnect to the server when a session is unlocked and when a computer wakes.)

Note: Contract Tools does not disconnect from Contract Tools License Server when Word quits abnormally: for example, when Word quits unexpectedly (crashes) or when you quit Word by using Task Manager or `taskkill`. After Word quits abnormally, the license that Contract Tools was using will remain unavailable from the server for, at most, the value of the `length` attribute of the `lease` element in configuration.xml; see Section 2.3, "Configuring Contract Tools License Server" on page 6 to learn more.

In addition to disconnecting automatically, a user can disconnect Contract Tools from Contract Tools License Server manually. Here are instructions you can offer to your users:

To disconnect Contract Tools from Contract Tools License Server:

- Show the Contract Tools pane, click the Subscription button, and then click Disconnect.

  If Contract Tools is configured to connect to Contract Tools License Server only when Contract Tools is turned on in a document window (see Section 3.3.3, "Connecting to Contract Tools License Server" on page 15 to learn more), Contract Tools disconnects from the server when Contract Tools is turned off in all windows. However, due to limitations of Word, Contract Tools cannot disconnect from the server when a user closes the last window in which Contract Tools is turned on if other windows remain open. For example, if Contract Tools is turned on in one window and turned off in all others:

  - If a user turns off Contract Tools, Contract Tools disconnects from the server
  - If a user does not turn off Contract Tools and instead closes the window, Contract Tools does not disconnect from the server

  See Section 3.3.3, "Connecting to Contract Tools License Server" on page 15 to learn about connecting Contract Tools to Contract Tools License Server.

3.11   Uninstalling Contract Tools

You can use PaperSoftwareContractTools.msi or PaperSoftwareContractToolsSetup.exe to uninstall Contract Tools.

To learn about	Go to
Using an MSI file to uninstall Contract Tools	Section 3.11.1, "Uninstalling with an MSI File" on this page
Using an executable to uninstall Contract Tools	Section 3.11.2, "Uninstalling with an Executable" on this page

Note: If you're using an activation key to administer Contract Tools, you should deactivate Contract Tools before you uninstall it; see Section 3.9, "Deactivating Contract Tools" on page 22 to learn how.

It's possible to remove Contract Tools from Word by using Word's COM Add-ins dialog (to show this, choose File > Options > Add-ins to show the Add-ins pane of Word Options, choose COM Add-ins from the Manage pop-up menu at the bottom of the Add-ins pane, and then click Go). Removing Contract Tools from Word is not a good idea. Doing this may permanently prevent Word from loading Contract Tools, even if you reinstall Contract Tools. To undo removing Contract Tools from Word, you may need to perform an online repair of Office; visit *Repair an Office application* (https://support.microsoft.com/en-us/office/repair-an-office-application-7821d4b6-7c1d-4205-aa0e-a6b40c5bb88b) to learn more. You should uninstall Contract Tools instead of removing it from Word.

3.11.1   Uninstalling with an MSI File

If you used PaperSoftwareContractTools.msi to install Contract Tools, you can uninstall Contract Tools by entering in PowerShell or Command Prompt:

```
msiexec /x PaperSoftwareContractTools.msi /q
```

3.11.2   Uninstalling with an Executable

If you used PaperSoftwareContractToolsSetup.exe to install Contract Tools, you can uninstall Contract Tools by entering in PowerShell or Command Prompt:

```
.\PaperSoftwareContractToolsSetup /x // /q
```

While this command will uninstall Contract Tools without prompting a user, Contract Tools Setup will appear. To uninstall Contract Tools without showing Contract Tools Setup, you can pass a Contract Tools product code to `msiexec`. For example, to uninstall Contract Tools v1.37.4.0 without user interaction, enter in PowerShell or Command Prompt:

```
msiexec /x "{59597055-A0A4-490F-A4FA-8A14D8C23285}" /q
```

To find the product code of your version of Contract Tools, enter in PowerShell 3.0 or later:

```
$filter = 'Name="Contract Tools" AND Vendor="Paper Software LLC"'
$instance = Get-CimInstance Win32_Product -Filter $filter
$instance | Format-Table IdentifyingNumber, Version
```

or in PowerShell 2.0 or earlier:

```
$filter = 'Name="Contract Tools" AND Vendor="Paper Software LLC"'
Get-WmiObject Win32_Product -Filter $filter
```

or in Command Prompt:

```
set filter=Name='Contract Tools' AND Vendor='Paper Software LLC'
wmic product where "%filter%" get IdentifyingNumber, Version
```

The IdentifyingNumber in the output is the product code.

For a table of Contract Tools version numbers and corresponding product codes, see Appendix C, "Contract Tools Product Codes" on page 31.

Appendix A

Contract Tools User Preferences

Learn detailed information about Contract Tools user preferences.

You can configure default values of user preferences by modifying PaperSoftwareContractTools.dll.config in Contract Tools' installation folder (see Section 3.5.4, "Installation Folder" on page 20) before Word loads Contract Tools for the first time (see Section 3.3.1, "Configuring LoadBehavior" on page 13). PaperSoftwareContractTools.dll.config is an XML file that contains default values of user preferences in `setting` elements like this:

```
<setting name="PreferenceName" serializeAs="String">
    <value>DefaultValue</value>
</setting>
```

You can use PowerShell to modify `setting` elements. For example, when Contract Tools is installed for all users on a 64-bit operating system, you can modify user preferences named `PreferenceName1` and `PreferenceName2` to have default values of `DefaultValue1` and `DefaultValue2` with this PowerShell script:

```
$defaultValuesForPreferenceNames = @{
  # Change these to actual preference names and default values, for example:
  #   EnableAutomatically = 'False'
  PreferenceName1 = 'DefaultValue1'
  PreferenceName2 = 'DefaultValue2'
}
$installationFolder = "${Env:ProgramFiles(x86)}\Paper Software\Contract Tools"
$absolutePath = "$installationFolder\PaperSoftwareContractTools.dll.config"
$xml = New-Object Xml.XmlDocument
$xml.Load($absolutePath)
foreach ($item in $defaultValuesForPreferenceNames.GetEnumerator()) {
  $xml.SelectSingleNode("//setting[@name='$($item.Name)']").value = $item.Value
}
$xml.Save($absolutePath)
```

To configure default values of user preferences after Word has loaded Contract Tools, delete the folder in which Word saves Contract Tools user preferences, and then configure default values

as you otherwise would. To delete the folder in which Word saves Contract Tools user preferences, enter in PowerShell:

```
$preferencesFolder = "$Env:LOCALAPPDATA\Microsoft_Corporation"
Remove-Item "$preferencesFolder\PaperSoftwareContractTool_*" -Recurse
```

Note: This PowerShell script immediately deletes the folder in which Word saves Contract Tools user preferences. It doesn't move the folder to the Recycle Bin.

Here are names (in **bold**) of Contract Tools user preferences that you can configure.

A.1   Whether Contract Tools Turns On Automatically

EnableAutomatically `True` to turn on Contract Tools when documents open, `False` otherwise. Default is `True`. See Section 3.3.7, "Configuring Contract Tools to Be Turned Off When Documents Open" on page 17 for a PowerShell script that configures this preference.

A.2   Contract Tools Pane

AlwaysShowPane `True` to show the Contract Tools pane when documents open, `False` otherwise. Default is `True`.

PanePosition `Right` to position the Contract Tools pane on the right side of a Word window by default, `Left` to position the pane on the left side. Default is `Right`.

SwitchPaneViewsUsingButtons `True` to use buttons to show Contract Tools pane views, `False` to use a pop-up menu. Default is `True`.

PaneWidth Default width in pixels of the Contract Tools pane, as an integer between 200 and 500. Default is 350.

PaneBackgroundColor Comma-separated RGB color of the Contract Tools pane background. Default is 240, 240, 240.

A.3   Analysis

AnalyzeBeforeSave `True` to analyze when saving a document, `False` otherwise. Default is `True`.

AnalyzeOnSync `True` to analyze when syncing a document to a server, `False` otherwise. Default is `True`.

SyncAnalysisWithDocument `True` to sync analysis results with documents while there are unanalyzed changes, `False` otherwise. Default is `True`.

FindAnalysisBookmark `True` to automatically set an analysis range (saved as a hidden bookmark) when documents open, `False` otherwise. Default is `True`.

UpdateAnalysis `True` to update an analysis after editing text (for example, typing and deleting), `False` otherwise. Default is `True`.

AnalysisUpdateDelay Time in milliseconds between the last edit and an analysis update, as an integer between 0 and 1000. Default is 250.

SaveAnalysisSettings `True` to save analysis settings (including ignored drafting errors) with documents as custom XML data in Word 2010 and later, `False` otherwise. Default is `True`. You can also configure whether Contract Tools saves analysis settings using a Registry value; see Section 3.3.5, "Saving Analysis Settings" on page 16 to learn more.

AnalyzeWhenPaneIsHidden `True` to analyze a document when saving or syncing to a server while the Contract Tools pane is hidden, `False` otherwise. Default is `True`.

A.4   Highlighting and Underlining

HighlightDraftingErrors `True` to highlight drafting errors, `False` otherwise. Default is `True`.

DraftingErrorHighlightColor Comma-separated ARGB color of drafting error highlighting. Default is `30, 180, 0, 0`.

UnderlineDraftingErrors `True` to underline drafting errors, `False` otherwise. Default is `False`.

DraftingErrorUnderlineColor Comma-separated RGB color of drafting error underlining. Default is `180, 0, 0`.

ShowInstancesOfSelectedContent `True` to show instances of selected content, `False` otherwise. Default is `True`.

HighlightInstancesOfSelectedContent `True` to highlight instances of selected content, `False` to underline instances of selected content. Default is `True`.

HighlightMarkersOfSelectedInlineList `True` to highlight markers of the selected inline list or proviso list, `False` otherwise. Default is `True`.

HighlightSelectedMatchedBrackets `True` to highlight matched parentheses and brackets when selected, `False` otherwise. Default is `True`.

HighlightingOpacity Opacity of highlighting of items other than drafting errors, as an integer between 0 and 255. Default is 30.

A.5   Completion Menu

CompletionMenuShowsDefinedTermUses `True` to show defined terms in the Completion menu, `False` otherwise. Default is `True`.

CompletionMenuShowsSnippets `True` to show snippets in the Completion menu, `False` otherwise. Default is `True`.

RightArrowKeyInsertsCompletions `True` to insert completions using the Right Arrow key, `False` otherwise. Default is `True`.

ShowCompletionInstructions `True` to show instructions for choosing a completion in the Completion menu, `False` otherwise. Default is `True`.

A.6   Navigation

DoubleClickNavigates `True` to make double-clicking defined terms and cross-references cause navigation, `False` otherwise. Default is `True`.

ShowLocationWhenScrolling `True` to show location while dragging the vertical scroller, `False` otherwise. Default is `True`.

A.7   Snippets

SnippetInsertionKeepsFormatting `True` to preserve formatting when inserting snippets, `False` otherwise. Default is `False`.

SnippetsViewWarnsBeforeDeletingSnippets `True` to show an alert before deleting snippets, `False` otherwise. Default is `True`.

SnippetsViewShowsCategories `True` to show building block categories in the Snippets view, `False` otherwise. Default is `False`.

SnippetsViewShowsAllAutoTextAndQuickParts `True` to show all AutoText and Quick Parts in the Snippets view, `False` otherwise. Default is `False`. See Section 3.6, "Sharing Text as Snippets" on page 20 for a PowerShell script that configures this preference.

SnippetsTemplatePath Path of a Word template (DOTX) file for snippets. If the value of this preference is empty when Contract Tools starts, Contract Tools attempts to use the path of an already-existing file or the path of a newly-created file. Default is empty.

SnippetsLoadWhenStarting `True` to load snippets when Contract Tools starts, `False` otherwise. Default is `True`.

A.8   Issues Pop-Up Menu

IssuesIncludeUnusedDefinedTerms `True` to include unused defined terms in the Issues pop-up menu, `False` otherwise. Default is `True`.

IssuesIncludeDefinedTermsUsedOnce `True` to include defined terms used only once in the Issues pop-up menu, `False` otherwise. Default is `True`.

IssuesIncludeToDos `True` to include to-dos in the Issues pop-up menu, `False` otherwise. Default is `True`.

IssuesIncludeUnknownCrossReferences `True` to include cross-references to unknown items in the Issues pop-up menu, `False` otherwise. Default is `True`.

A.9   Updating

CheckForUpdates `True` to check for updates when Contract Tools starts, `False` otherwise. Default is `True`. You can also configure whether Contract Tools checks for updates using a Registry value; see Section 3.3.4, "Updating Contract Tools" on page 16 to learn more.

Appendix B

Hosted License Servers

Learn detailed information about hosted license servers.

Hosted license servers are installed on Amazon Elastic Compute Cloud (EC2) (https://aws.amazon.com/ec2/) virtual machines running the Amazon Linux 2 (https://aws.amazon.com/amazon-linux-2/) operating system. A hosted license server is installed on its own EC2 instance, and the EC2 instance is assigned to a security group (https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html) unique to your organization.

HTTPS communication to the EC2 instance is enabled using industry-standard Let's Encrypt (https://letsencrypt.org) Transport Layer Security (TLS) certificates managed using the Electronic Frontier Foundation's Certbot (https://certbot.eff.org).

The log file of a hosted license server is configured to contain only errors (see Section 2.3, "Configuring Contract Tools License Server" on page 6 to learn about configuring Contract Tools License Server); no Contract Tools usage information is logged when you use a hosted license server.

Appendix C

Contract Tools Product Codes

Learn Contract Tools version numbers and corresponding product codes.

Here are Contract Tools version numbers and product codes. If you used PaperSoftwareContractToolsSetup.exe to install Contract Tools, you can pass a Contract Tools product code to `msiexec` to uninstall Contract Tools without showing Contract Tools Setup; see Section 3.11.2, "Uninstalling with an Executable" on page 24 to learn how.

Version number	Product code
1.37.4.0	{59597055-A0A4-490F-A4FA-8A14D8C23285}
1.37.3.0	{0C348C36-9439-4F85-8AE0-4E77CA6CB50B}
1.37.2.0	{7C67F48C-9D0C-45BF-B775-F44B59F9E4EC}
1.37.1.0	{2DAA65D6-F348-46AE-A686-11392785D09E}
1.37.0.0	{921B77DC-5E83-438C-A90D-82766F371AE9}
1.36.3.0	{853B1D02-2CD1-4A1B-9A39-22AFF05165F4}
1.36.2.0	{4581C3B3-436E-46FC-A661-313AD1E49391}
1.36.1.0	{313875EF-2686-428B-8705-1C88E1418032}
1.36.0.0	{871FEDC4-926C-4FBA-BFD6-B1B96996A941}
1.35.11.0	{1FCC5076-CDB3-4F1F-B570-C35806BDBF40}
1.35.10.0	{8C62B169-727E-44A7-B398-FDC6A56A03F8}
1.35.9.0	{D8EF5002-11F0-49FC-9127-B11F48CC436A}
1.35.8.0	{749E6435-6F1C-4EF1-AA8C-EA47F0B443A8}
1.35.7.0	{6846BAEC-7880-4A71-A54C-C57709E513B8}
1.35.6.0	{67D8F305-2E12-442C-8F86-0E0B80976F8A}

Version number	Product code
1.35.5.0	{8D163170-437C-4DB8-BD09-1B118239E144}
1.35.4.0	{FDDFC208-E83D-4FA3-80E0-CAF4D7B189E7}
1.35.3.0	{FAF98289-526C-4381-AB02-325BEC8FE11E}
1.35.2.0	{1B1C0B9E-52B9-4EC1-97C7-C2B5E21BB90C}
1.35.1.0	{119CEA92-43E3-4E70-9CAA-47A266F22CEA}
1.35.0.0	{3E98F3E8-CBAB-4EEF-8EF0-7F49DFDBBE97}
1.34.4.0	{78DBE68D-8EC0-4C81-8D83-93E2AF68448D}
1.34.3.0	{E6CEB4B1-91CA-4105-8E25-FAFCA3827F74}
1.34.2.0	{C4C84816-351D-4215-BABE-8F54873958CA}
1.34.1.0	{45D8EB1A-BC0F-479F-AD00-75A95F0383A8}
1.34.0.0	{BEDCB6BA-F17F-4A33-8616-21E57EB3352A}
1.33.0.0	{83D7A592-37B7-4DA0-A6F6-98D23F905C28}
1.32.3.0	{F161D65E-987A-4090-8985-57535F149A8C}
1.32.2.0	{1D08639C-C788-4732-AF57-BE484D5D32DD}
1.32.1.0	{8808D4C3-D902-4D05-8636-880D23EC29FB}
1.32.0.0	{491E3DF1-7D44-4C22-AFB8-E21A8D99CBDD}
1.31.0.0	{50E6B3FC-561D-44CE-9244-D171136FECA2}
1.30.20.0	{BD743C19-5959-4259-8F36-4A161478C727}
1.30.19.0	{72CB8C4F-293E-4BD9-9B70-6DA45D526798}
1.30.18.0	{66FFE0C1-AFA1-4DEC-9448-BB8799AFC0F7}
1.30.17.0	{9E72432E-5BE4-48BB-836C-BF83C3A3676B}
1.30.16.0	{51A2BF7D-0C40-4F0C-9F7D-79459A94C07E}
1.30.15.0	{7B997C3C-1CE7-46DC-9664-4B057BC37574}
1.30.14.0	{BEF20BAB-41FE-492D-A931-C0921797FA82}
1.30.13.0	{8F6DA81A-BC00-44B9-96B0-3A296D6BE050}
1.30.12.0	{85F9E626-3AEE-4CFE-89D2-C335003C24F7}
1.30.11.0	{E3EC0628-9C8A-490C-A0EF-C95A3F11FE16}
1.30.10.0	{22C54934-5031-4C71-9BF7-71BA656A3EBF}
1.30.9.0	{357C7F71-E3FC-4A42-A438-418438D0CB90}
1.30.8.0	{D1039911-3E11-4D96-B2D4-35A2312781E5}
1.30.7.0	{69FAD4CA-F864-4C47-BC02-9A426B91F57D}
1.30.6.0	{6012D8EA-9380-41AB-9220-483456CE1C6A}

Version number	Product code
1.30.5.0	{E0CD02D3-95B0-41C6-BAC9-B20BEFEBA2D5}
1.30.4.0	{76F353EE-C581-44B8-B4F4-AA46C74EA260}
1.30.3.0	{F0822B17-38D8-4DD5-923C-D09F12B40B4B}
1.30.2.0	{38A0E6D5-9B3C-434D-9ADE-247CD05FAD1D}
1.30.1.0	{E126D4FF-FB8A-4ADA-972A-36EB7ED86BC9}
1.30.0.0	{0FF98323-DB25-473B-B509-265C63969C38}
1.29.15.0	{5FDA9CE7-C1B7-4785-8427-504FD7CAC3E5}
1.29.14.0	{D3C93F5C-7640-4EA2-B132-D3A6243F69ED}
1.29.13.0	{8D3370F0-4837-4C63-AA3B-D74D47B4E5E2}
1.29.12.0	{871E1C54-3038-4A72-8F18-7C09CCB07F98}
1.29.11.0	{67944DFA-EE9B-4B11-9694-654DAFC6DCF7}
1.29.10.0	{9EDB33CB-7EDC-4237-9F83-AC76443E17DF}
1.29.9.0	{1CD694E8-38BF-478B-B5F5-61FDBB174C09}
1.29.8.0	{1BABE020-E51A-4885-90A6-1FC58AB8FAB0}
1.29.7.0	{B69A1CD9-235E-4177-850A-1069B7900923}
1.29.6.0	{88E41A42-79C1-44A5-85FE-ABE690968546}
1.29.5.0	{566929A7-E86D-42D9-B82C-45030BCA8F42}
1.29.4.0	{33BC9D48-2A66-4833-8552-6D2E6A7EDA2F}
1.29.3.0	{44B1E606-CFCB-474D-B67C-EA1A983DAC90}
1.29.2.0	{131688F6-4504-4619-87AD-4C40D22F5D8D}
1.29.1.0	{607297BE-E19A-4D4B-B45B-3AE1B808B8AC}
1.29.0.0	{DC288CF8-F861-4FF4-83CA-0B7C60C59A7C}
1.28.10.0	{8178CE3B-8DDF-4A3C-B501-716D6D8445F5}
1.28.9.0	{7CA82501-6DEA-481F-8E5D-7ADA24018F0F}
1.28.8.0	{D18B9A97-ACFA-4EF7-810E-97AE04A38243}
1.28.7.0	{E61BDD0D-609C-4AFD-9920-BB3FEF085685}
1.28.6.0	{ABC44349-1E04-4A62-A55A-5A3128234BDF}
1.28.5.0	{2F9D4E3C-C34F-4181-97B1-8DF7D6ABBDB3}
1.28.4.0	{09B9607F-6FA1-4C73-9C38-01B1F1609D17}
1.28.3.0	{908460A9-197E-43F4-9258-8179F216B2D6}
1.28.2.0	{7CBCAF4C-FAC9-476F-BA5F-98D567781043}
1.28.1.0	{4F04E8AB-2EC1-4CE1-A363-0F364C0291C4}

Version number	Product code
1.28.0.0	{BF9B896F-1C98-4402-985B-1BA6747E6BFE}
1.27.5.0	{8A708616-11B7-492D-BAEC-DD88163BBA4E}
1.27.4.0	{132ADDB5-A514-4543-986C-53674B06CBD3}
1.27.3.0	{EB41E010-044F-4BD4-B12A-7B524D5B85A0}
1.27.2.0	{65BC795A-FB57-470C-B0EC-8D63E87A9B51}
1.27.1.0	{2F3EC94C-D5FE-42DA-9A7C-FFED65CBAC88}
1.27.0.0	{B1336BDD-85F8-492A-BCC9-57128B213F0F}
1.26.4.0	{43457065-AE19-49CD-BA39-E2F05BBE7D89}
1.26.3.0	{64BA8903-07D2-4E6E-AD4E-1201DF6F0BA0}
1.26.2.0	{35D2B311-6AEC-4767-9922-70F47EEC2310}
1.26.1.0	{30D8B7E9-6F78-45E9-B17E-A3491F789390}
1.26.0.0	{6E44FE77-C823-483C-AE18-8242124C3307}
1.25.3.0	{003E928D-D4AF-4D53-928B-854BB71C65E7}
1.25.2.0	{73F6C0C9-65F0-4587-B223-6879C0176A34}
1.25.1.0	{C52AFC7B-F44E-44BD-9F8D-4706799ACAC0}
1.25.0.0	{D0A55E68-98A9-44D1-8FD8-478305DF0720}
1.24.5.0	{CF0FEEDA-2196-4D3B-8228-B2DE37AEBE81}
1.24.4.0	{E549C859-3C35-4E7B-9D1A-F9AB48C46C68}
1.24.3.0	{825DF421-05D6-4738-8602-29FECF61C503}
1.24.2.0	{94915F56-5671-47A8-82A2-DF99B01E8ABC}
1.24.1.0	{EAD211B0-64A3-4FFF-B513-8B40A1E97F02}
1.24.0.0	{C41BACF6-DEA8-4E20-92FF-D49058F7C11B}
1.23.3.0	{DF73C6B9-C7D7-49B9-8813-C2F7AD914799}
1.23.2.0	{8C920507-5F6E-4DFD-A79C-47B05D40A913}
1.23.1.0	{00CEE861-8A9A-4950-90A4-7442A4658D71}
1.23.0.0	{CCB435FE-FD6C-4854-891A-47804C699894}
1.22.4.0	{199BBBA4-778F-47FA-9FC1-D53052601F06}
1.22.3.0	{7FB9A73F-D1AF-4F93-A295-719FD080D1FC}
1.22.2.0	{4F4CC560-0BD1-4EF5-9EB0-795A12AB4DFD}
1.22.1.0	{BE941628-E16E-4352-8A41-8D4AF49AFDBD}
1.22.0.0	{4B52C4EF-6764-4C18-8FC2-DE1060CDB5FC}
1.21.4.0	{04392077-CDD5-4476-B8E9-2A4C3CDC0293}

Version number	Product code
1.21.3.0	{132F369F-8C3C-469D-B2B0-21D5C7369A9A}
1.21.2.0	{833BBAAB-9E0B-44C6-AA7D-8BE4727A7B20}
1.21.1.0	{E6C4BCCD-FB94-4AD6-A6BA-00F87C7DEDB8}
1.21.0.0	{8D94BC8C-D3C8-45F9-BEBB-639D7AE9BDB0}
1.20.1.0	{B4CE72E7-5C45-4D6A-B7BE-EE8FFA1065B3}
1.20.0.0	{FA9FB2A6-2816-4DFA-9032-8285AD3C2532}
1.19.4.0	{B86D2EC8-C902-4893-A84D-B27F99F44C85}
1.19.3.0	{57C1157D-60C9-4C5E-B199-BB99FB63BFE4}
1.19.2.0	{D99CBFC0-0D2B-4E53-BF4F-A0B349BD1D82}
1.19.1.0	{D53C1295-3F74-4735-875D-E9349C204485}
1.19.0.0	{64CAE0BC-3F97-4938-A3CC-5E907C8C0BF2}
1.18.6.0	{69F2B766-A061-4F9F-8887-85019897D33E}
1.18.5.0	{3493D517-F429-4FD8-9E75-42524180441D}
1.18.4.0	{BB36FCB3-BE0A-4910-BCD2-1A7EEF132415}
1.18.3.0	{7C80F46E-5CC2-4F48-9A2C-CBBB8176F4C6}
1.18.2.0	{1EF788E5-2D3A-4C6E-BFE1-2861A64B391F}
1.18.1.0	{61CF5353-89B8-49D2-9D52-954ABE187F3B}
1.18.0.0	{A8608D32-D590-4596-BB67-D78DFB881E07}
1.17.8.0	{86A21335-6227-42F5-B8ED-E517EFD9EDBF}
1.17.7.0	{2E8A3C7C-CD97-451B-8245-959CC45323B4}
1.17.6.0	{EF862871-258C-45E6-B37A-D2C5B116D03B}
1.17.5.0	{D034D9E8-6854-48B4-9DCF-B34CA1402233}
1.17.4.0	{C6B8EFA4-1A2B-44E9-96B3-0CA6D1A7DE36}
1.17.3.0	{26FD68B8-910F-43FA-943B-BE3C56D85883}
1.17.2.0	{D9586ACB-44E5-4D98-8262-B835538F0736}
1.17.1.0	{216F95BC-7D1D-493B-ACC1-5849466D5AF3}
1.17.0.0	{0CDF4A42-164B-40BC-9171-7CDF3FBB3714}
1.16.5.0	{AC41251D-3340-46E6-81F4-47A034783E4E}
1.16.4.0	{B6209071-4FC5-4039-96F1-AEE939C61DD9}
1.16.3.0	{33D2E91C-430A-4AF1-A0DD-874BAB3CCFB1}
1.16.2.0	{8219C72E-D659-4AF1-B9C7-616F2D889E0C}
1.16.1.0	{B60971B5-D8B2-45D8-8ECF-73C83B1F89BA}

Version number	Product code
1.16.0.0	{8618D9E9-051F-4A04-A889-847BE918CC13}
1.15.12.0	{36F5E732-F411-43B4-9488-E3FEAE197007}
1.15.11.0	{E03F1D66-A306-43B8-A248-A779F0F73C4E}
1.15.10.0	{8A7001E9-F228-4CDD-92F8-9DC5176AD81B}
1.15.9.0	{0B53425D-076C-4D74-9BAD-804FC208C8DD}
1.15.8.0	{91CF89DF-2909-49EB-9DE2-C0CED6714388}
1.15.7.0	{811928AC-BB01-47C1-B15F-FD1A8865DCA9}
1.15.6.0	{E6DC5A97-B0E0-4628-971E-37F812893A9C}
1.15.5.0	{CE38A45E-AB39-4106-A53E-C0F132103C1F}
1.15.4.0	{A639DA03-156C-4D3A-B682-6DB7E9252976}
1.15.3.0	{4AF837D6-5A36-4779-8BF8-4A0542B6E6AA}
1.15.2.0	{DC1F7C51-9F8C-46D5-A706-B9E0A1251A5A}
1.15.1.0	{2DB12D18-9320-4A1C-AB69-88EFA5D9050B}
1.15.0.0	{9D34107E-D752-4491-AE40-4E3452042228}
1.14.2.0	{C0436F1D-1264-4A99-908C-88B5B9314805}
1.14.1.0	{8E28B9EE-1A85-495C-B8F9-BA1DC9F10C4E}
1.14.0.0	{B9974EAB-15F4-493F-B76E-441156E09D8A}
1.13.0.0	{04A52A26-933D-4B63-98A0-C365041A46B5}
1.12.0.0	{14D5A1A7-6377-4C0B-817F-00F5B6A977CC}
1.11.1.0	{4FC989AE-B926-4B25-B1E0-3807B5369509}
1.11.0.0	{A70F3A5B-6586-4D99-87D2-EE856FE11250}
1.10.2.0	{A3308E2A-5F7F-49CB-A210-309D5711CDA0}
1.10.1.0	{873AB223-AA1F-4A6C-B9A2-26FE47FE7000}
1.10.0.0	{110E07B3-B4C2-4119-9CA0-DA192656B554}

Contract Tools v1.9.0.0 and earlier use Microsoft ClickOnce installation and cannot be uninstalled using `msiexec`.

Index

V