





PaintShop[®] Pro X9


Deployment
Guide


Introduction

As you read through this guide, you'll find that the right column of each page contains the main content, while the left column contains the following categories of information:

 **definition** — explains the italicized term or concept

 **tip** — presents a helpful detail, such as a shortcut, variation, or benefit

 **note** — presents an additional detail about the specified topic or task

 **warning** — presents a crucial detail about the specified topic or task

The left column also gives you room to jot down notes.

This guide is intended to help you deploy Corel® PaintShop® Pro X9 (Corporate and Education Edition) to your network as quickly and easily as possible.

Contents

Stage 1: Preparing for deployment	2
Stage 2: Creating the server image	3
Stage 3: Installing the software	5
Stage 4: Maintaining the installations	15

Additional resources

If you are new to software deployment or otherwise require more basic information on deploying Corel® software products, please refer to the *Corel® Beginner's Guide to Network Deployment*. You can request a copy of this guide from your Corel® Support Services representative.

For even more information, see the following Web resources.

Web resource	Description
Corel® website: www.corel.com	Information about Corel Corporation and its portfolio of software products
Corel® Support Services website: www.corel.com/support	Information about product features, specifications, pricing, availability, services, and technical support
Corel® Knowledge Base™: www.corel.com/kb	A searchable repository of articles written by the Corel Support Services team

For help with the Microsoft Windows Installer (MSI) technology that is used to install the software, please refer to the Microsoft website

Stage 1: Preparing for deployment



As used in this guide, the term “network” signifies two or more computers that are connected to each other for the purpose of exchanging information.



Workstations are the computers from which the average user works, and servers are the computers that manage the shared resources of the network.



To more easily manage the access rights of workstation users, you may want to use Group Policy Objects (GPOs, or “system policies”). See “Managing permissions with Group Policy Objects” on page 5.

To deploy the software to your *network* as smoothly as possible, you can prepare by doing the following:

- Check the system requirements for the software.
- Prepare the server.
- Prepare the workstations.

For details, see below.

Checking the software requirements

To begin, make sure that your *server* and *workstations* are eligible for the software. Consult the following:

- Readme file for the software (if available)
- product-information page on the Corel website (www.corel.com)
- any other special instructions for the software


Preparing the server


- Make sure that the server meets the minimum system requirements for the software, and that it has enough free disk space for the installation.
- Make sure that the operating system on the server has been updated with the latest service packs and security patches.
- Make sure that you have the proper permissions for creating a software image on the server. You must be either a local administrator or an administrator for the domain that you are managing, and you must have read/write access to the server location.


Preparing the workstations

- Make sure that the workstations meet the minimum system requirements for the software, and that they have enough free disk space for the installation.
- Make sure that the operating systems on the workstations have been updated with the latest service packs and security patches.
- Make sure that anyone who will be installing the software from the server image has the proper permissions to do so. To install the software on a workstation, you must be either a local administrator or an administrator for the domain that you are managing, and you must have read access to the server location.

Stage 2: Creating the server image

 A server image, also called an “administrator image” or simply an “image,” is a set of uncompressed application files on the server that is created from a set of compressed files on the installation disc.

 A command line is a textual command that lets you specify desired settings.

 To mount an ISO file, you might require third-party software that helps you create a virtual drive. Alternatively, you can burn the ISO file to a CD/DVD and access the files from the disc.

After preparing for deployment, you’re ready to create a *server image* of the software, from which you can install the software to the workstations. (If you want to support multiple installation types, you can even create multiple server images: one for each desired configuration.)

Contents

Creating a server image	3
Finalizing the server image	4

Creating a server image

To create a server image, you run a *command line* that initializes the software setup and specifies your desired installation settings.

To run a command line

- 1 Open the **Run** dialog box (shortcut key: Windows Logo + R).
- 2 Type the command line in the **Open** box, and then click **OK**.

To create a server image

- 1 Do one of the following:
 - If you have an installation disc, insert the disc into the CD/DVD drive.
 - If you downloaded an ISO file, mount the ISO to a directory or extract the files from the ISO image to a directory on your server by using a utility such as WinZip® (available from www.winzip.com).

If the **AutoRun** screen opens, click **Exit**.

- 2 Run the following command line, where **x:** is the directory where the disc, ISO, or extracted files are located.

X:\Setup.exe /a

- 3 Type your user name and serial number (with or without hyphens) in the boxes provided, and then click **Next**.

The customer information that you provide is passed on to the workstations when the software is deployed to the network. By default, users can change the user name but not the serial number.

- 4 Specify a network location for the server image. To change the default location, type a valid server path in the **Network location** box, or click **Browse** to browse to a valid network location.



Proceed with extreme caution when using the /q switch to create a server image.



To change the location of a server image, you must create a new image at the new location. You cannot copy an image from one location to another.

- 5 If you want to allow workstations to detect and download product updates, enable the **Product updates** check box.
- 6 Click **Install** to begin copying the files to the server.
If you click **Cancel**, you are prompted to confirm that you want to cancel creating the server image. Cancelling “rolls back” the setup and undoes most of the changes made; however, some manual clean-up may be required.
- 7 Click **Finish**.

You can also

Create a server image silently (or with limited UI)

Use the following command line (where **X:** is the directory where the disc, ISO, or extracted files are located; **image_location** is the desired location of the server image; and **serial_number** is the assigned serial number for the product):

```
X:\Setup.exe  
TARGETDIR="image_location"  
SERIALNUMBER="serial_number" /q /a
```

The **/q** switch is used to restrict the amount of the user interface that appears during installation. For a list of switch parameters, see page 7.

Create an error log

Use the following command line (where **log_file** is the location and filename of the log file):

```
X:\Setup.exe /l "log_file" /a
```

For a list of the parameters that are available for the **/l** switch, see page 8.

Finalizing the server image

Before deploying from the server image, you may want to take the following steps.


Check for software updates


Check for software updates, and apply them to the image as necessary. This way, you can avoid having to deploy the software twice. For details, see page 16.


Test the image

Test the image with a small subset of workstations before rolling it out to your entire organization.

Stage 3: Installing the software

 Looking for command-line switches and public properties? See “Pushing the software” on page 6.

 Pulling the software involves having the workstation users themselves execute the software installation and perhaps even choose their own installation options.

 Group Policy Objects (sometimes called “system policies”) reside in a central location on a Windows-based network and define how each workstation is configured.

You can install the software on the workstations in two main ways:

- by having the users themselves install (or “pull”) the software from the server image to their workstations
- by using a command line to install (or “push”) the software from the server image to the workstations on the users’ behalf

Contents

Pulling the software	5
Pushing the software	6

Pulling the software

Workstation users themselves can install (or “pull”) the software by using one of the following methods:

- browsing to the location of the server image, double-clicking **Setup.exe**, and following the instructions in the setup. This is the most common method for pulling the software.
- running a command line that installs the software from the setup on the server image. Typically, this method is reserved for push-installation scenarios (see page 6).

Managing permissions with Group Policy Objects


To install the software, workstation users require administrator-level privileges. To assign such privileges, Windows-based networks use *Group Policy Objects* (or “GPOs”): items stored in a central network location and used to automatically update the registry settings on each workstation when its user logs in to the network.


If you want to give users administrator-level access rights (either temporarily or permanently), you may need to configure the GPOs for your network by using a Group Policy Editor.

For general help with GPOs, please refer to the Software Development Kit (SDK) for Group Policy.

To access the Group Policy Editor for Windows

- 1 Open the **Run** dialog box (shortcut key: Windows Logo + R).
- 2 Type **gpedit.msc** in the **Open** box, and then click **OK**.

 *Pushing the software forces installation on the workstations without requiring any user interaction. Typically, the user interface for the setup is suppressed.*

 *Command-line switches typically control the behavior of the software setup, while public properties typically pre-configure the installed software.*

To let workstation users install software

Enable the following system policies for workstation users:

- Computer Configuration\Administrative Templates\Windows Components\Windows Installer\Always install with elevated privileges
- Computer Configuration\Administrative Templates\Windows Components\Windows Installer\Enable user control over installs
- User Configuration\Administrative Templates\Windows Components\Windows Installer\Always install with elevated privileges

To let limited-access users patch software

Enable the following system policy for workstation users:

- Computer Configuration\Administrative Templates\Windows Components\Windows Installer\Enable user to patch elevated products

Pushing the software

If you prefer to install the software on behalf of the workstation users, you can “push” the software from the server image to the workstations. To do this, you use a command line in conjunction with one of the following:


- **a batch file** — a text file that can be scripted to run commands automatically. For help, please visit the Microsoft website.
- **a Group Policy Object** — an item, stored in a central network location, that defines how the workstations are configured. For help, please refer to your Software Development Kit (SDK) for Group Policy.
- **a push technology** — a third-party tool specially designed for installing software automatically. For help, please visit the manufacturer’s website.


As previously stated, all three of these methods involve software-installation command lines, which typically include the following:


- the setup file on the server image
- *switches* that control the setup behavior
- *public properties* that pre-configure the installed software


About command lines

Specifying the setup file.	7
Using switches	7
Using public properties	10

 *Separate switches from other command-line elements, including other switches, with spaces.*

 *Do not type a space between a switch and its parameters or between the parameters in one switch, unless otherwise noted.*

 *For a list of all switches for the Microsoft Windows Installer technology, please see the Microsoft website.*

 *After installation with the /q switch, it is normal for the licensing agreement to appear the first time that users run the program on their workstations.*

Specifying the setup file

The main item to specify in your command line is the executable file that you want to use to install the software.

This file is **Setup.exe**, the executable file for the software setup. **Setup.exe** is located on the server image, at the installation path that you chose when you created the server image.

The basic syntax for a **Setup.exe** command line is as follows:

```
\\server\path\Setup.exe
```

If your server-image path contains spaces, you must place quotation marks around the entire **Setup.exe** component:

```
"\\server\path with spaces\Setup.exe"
```

Using switches

To customize the behavior of the software setup, you can use a variety of command-line switches.

Basic syntax

The syntax for a switch consists of a forward slash (/) immediately followed by a character or string — for example, /q or /quiet.

Be sure to separate switches from other command-line elements, including other switches, with spaces.

Parameters

Some switches have parameters, which let you adjust the settings for the switch. In fact, some switches let you use multiple parameters.

To use a parameter, simply type the parameter immediately after the switch (that is, without a space), unless otherwise noted. If you do not specify any parameters, the switch uses its default settings.

Available switches, by function

Limiting the setup UI with /q	7
Creating a log file with /l	8
Controlling reboots	9

Limiting the setup UI with /q

The /q switch can be used to restrict the amount of the user interface (UI) that appears during installation. You can use the switch to prevent users from entering their own registration



The default parameter for **/q** is **n**.



The **/quiet** switch can be used in place of **/qn**.



The **/passive** switch can be used in place of **/qb**.



The default parameters for the **/l** switch are **iwearmo**.

information, to help enforce specific installation options, or even to perform “silent installations” (in which no user interface is visible during the setup).

Parameters

You can specify one of the following parameters for the **/q** switch.

Parameter	Effect
n	The user does not see the user interface during installation. Errors are recorded in a log file (see page 8). This is the default parameter.
b	The user sees only a progress bar and a Cancel button. If the user pushes the Cancel button, the installation is rolled back.
r	The user sees a progress bar, along with a page containing information about the installation. The user can choose to cancel the installation.
f	The user sees the full user interface.

Syntax

Here’s the command-line syntax:

```
\\server\path\Setup.exe /q
```

Creating a log file with **/l**

Use the **/l** switch if you want to log general information about the installation to a log file with the specified path and filename.

Parameters

You can specify one or more of the following parameters for the **/l** switch.

Parameter	Effect
i	Logs status messages
w	Logs nonfatal warnings
e	Logs all error messages
a	Logs initiated actions
r	Logs action-specific records
u	Logs user requests
c	Logs initial user-interface parameters



The **/log** switch can be used in place of **/l***.

Parameter	Effect
m	Logs error messages about out-of-memory warnings or fatal exits
o	Logs error messages resulting from insufficient hard disk space during installation to a server
p	Logs terminal properties
v	Logs very detailed information
*	Applies all parameters except v , recording all information in a single log file

Location and filename of log file

The **/l** switch can take an additional parameter: the location and filename of the log file. Type **/l** followed by a space, followed by an opening quotation mark, the path to the log file, the filename of the log file, and a closing quotation mark. Here's the command-line syntax (where *log_file* is the location and filename of the log file):

```
\\server\path\Setup.exe /l "log_file"
```

Syntax

The following sample command line uses the default parameters of the **/l** switch to log information in the file **C:\install.txt** during installation:

```
\\server\path\Setup.exe /l "C:\install.txt"
```

Using the /q switch with /l

You can use the **/q** and **/l** switches together. The following sample command line uses **/q** to suppress the user interface during installation and record errors in the specified log file:

```
\\server\path\Setup.exe /q /l "C:\Logs\My_Log.txt"
```

Controlling reboots

You can force a reboot after installation by using the **/forcerestart** switch in your command line. Here's the command-line syntax:

```
\\server\path\Setup.exe /forcerestart
```



Additional information on the use of public properties is available online from the MSDN developer program.



Be sure to use spaces to separate public properties from other command-line elements (including other public properties).



Use quotation marks around any value that contains spaces to ensure it is “read” as a single unit.



For a list of all public properties for the Microsoft Windows Installer technology, please see the Microsoft website.



For best results, enclose **location** in quotation marks.

You can also suppress a reboot after installation by using the **/norestart** switch. Here’s the command-line syntax:

```
\\server\path\Setup.exe /norestart
```

Using public properties

To customize the installed software, you can use a variety of public properties in your command line.

Basic syntax

Public properties are case-sensitive; they must be typed in capital letters, and they cannot contain spaces.

To use a public property in a command line, you must type the name of the public property in capital letters, followed directly by an equals sign (=), followed directly by the desired value.

PROPERTY=value

Values are also case-sensitive, but they can contain both uppercase and lowercase letters. A value can be a text string (such as a feature name) or a number. If a value contains spaces, you must enclose it in quotation marks to ensure that it is “read” as a single unit.

PROPERTY="value containing spaces"

Available public properties, by function

Installing the software at a specified location	10
Installing the specified application module	11
Installing language modules for the software	11
Controlling network-related features of the software	12
Controlling license-agreement prompts for the software	12
Controlling the installation of desktop shortcuts	12
Controlling file associations for the software	12
Specifying the locations of user resources	13
Controlling the database import process	13
Controlling when user data is created	14

Installing the software at a specified location

You can use the **INSTALLDIR** public property to install the software at a specific location on the workstations. The value for this property is the desired installation **location**. Here’s the command-line syntax:

```
\\server\path\Setup.exe INSTALLDIR="location"
```



*Do not set the properties of both **INSTALLX86** and **INSTALLX64** to 0. Otherwise, neither version will be installed.*



*English (**EN**) is always installed, so it does not need to be specified.*



When installing multiple languages, do not leave spaces between the comma and the language codes.



*If **FORCELANG** specifies a language module not also specified by **INSTLANG**, the software defaults to English.*

Installing the specified application module

Both 32-bit and 64-bit application modules of Corel PaintShop Pro X9 are installed on a 64-bit OS by default. You can use the **INSTALLX86** and **INSTALLX64** public properties to specify your preferred application module.

Here's the syntax for installing the 32-bit version only:

```
\\server\path\Setup.exe INSTALLX86=1 INSTALLX64=0
```

Here's the syntax for installing the 64-bit version only:

```
\\server\path\Setup.exe INSTALLX86=0 INSTALLX64=1
```

Installing language modules for the software

You can use the **INSTLANG** public property to install language modules for the software. A language module typically includes both the user-interface and the offline Help for that language.

The accepted value for this property is the two-letter *code* for any of the following supported languages:

- Chinese Traditional: **CT**
- Dutch: **NL**
- English: **EN**
- French: **FR**
- German: **DE**
- Italian: **IT**
- Japanese: **JP**
- Spanish: **ES**
- Russian: **RU**

Here's the command-line syntax:

```
\\server\path\Setup.exe INSTLANG=code
```

Installing more than one language module

You can install multiple languages by using comma separators, as in the following example:

```
\\server\path\Setup.exe INSTLANG=EN,FR,DE
```

When installing multiple language modules, you can use the **FORCELANG** public property to specify the default module. As with **INSTLANG**, the accepted value for this property is a two-letter language code.



*Disabling network-related features with **IOFF=1** also disables in-product messaging, along with automatic updates.*

In the following example, the English, French, and German language modules are installed, but French is specified as the default:

```
\\server\path\Setup.exe INSTLANG=EN,FR,DE FORCELANG=FR
```

Controlling network-related features of the software

You can disable ALL network-related features of the installed software by specifying a value of **1** for the **IOFF** public property. Here's the command-line syntax:

```
\\server\path\Setup.exe IOFF=1
```

Controlling license-agreement prompts for the software

If the software is installed silently, the workstation user will be prompted to accept the End-User License Agreement (EULA) at first start-up. You can suppress the EULA prompt on the workstations by specifying a value of **1** for the **FORCENOSHOWLIC** public property. Here's the command-line syntax:

```
\\server\path\Setup.exe FORCENOSHOWLIC=1
```

IMPORTANT: Please note that if you choose to suppress the EULA prompt on the workstations, you are accepting the terms of the EULA on behalf of all users on your network when you create the server image.

Controlling the installation of desktop shortcuts

By default, every workstation receives desktop shortcuts for the software. You can disable the installation of these shortcuts by specifying a value of **NONE** for the **DESKTOP** public property. Here's the command-line syntax:

```
\\server\path\Setup.exe DESKTOP=NONE
```

Controlling file associations for the software

Depending on the version of the OS, some file formats are automatically associated with Corel PaintShop Pro X9. You can disable these file associations by specifying a value of **0** for the **CDS_ASS_IMAGE** public property. Here's the command-line syntax:

```
\\server\path\Setup.exe CDS_ASS_IMAGE=0
```



*For best results, enclose **location** in quotation marks.*



*For best results, enclose **location** in quotation marks.*

Specifying the locations of user resources

Corel PaintShop Pro X9 supports a variety of user resources — files (such as workspaces or images) that can be shared among workstation users over the network, or stored locally for personal use. When deploying to the workstations, you can specify the locations of the shared and local resource folders.

Shared resource folder

If you want to provide workstation users with a shared repository of resources, use the **C_GLOBALREAD** public property. The value for this property is the desired **location** of the shared network folder; all workstation users can access, but not modify, this folder.

```
\\server\path\Setup.exe C_GLOBALREAD="location"
```

Local resource folder

When a workstation user starts Corel PaintShop Pro X9 for the first time, a local resource folder (**Corel PaintShop Pro\19.0**) is created in that user's **Documents** (on Windows 10, Windows 8, and Windows 7) folder. The workstation user requires full (read/write) access to this folder — and is prompted to provide an emergency folder when it is unavailable.

If you want to customize the location of the resource folder on the workstations, you can use the **MY_PSP_FILES** public property. The value for this property is the desired **location** to the local folder; the resources stored in this user folder can be accessed only by that user. Here's the command-line syntax:

```
\\server\path\Setup.exe MY_PSP_FILES="location"
```

Creating user resources

For details on creating user resources for Corel PaintShop Pro X9, please see the Help.

Controlling the database import process

By default, when you install a new version of Corel PaintShop Pro X9, it imports the database from the previous version. If you want to start with a clean database, you can turn off the importing action by using the following command:

```
IGNORE.IMPORTDB=1
```

Here's the command-line syntax:

```
\\server\path\Setup.exe IGNORE.IMPORTDB=1
```

Controlling when user data is created

By default, after you install the application, it launches and initializes the user data setup. The following command lets you suppress the initialization so that it occurs when the users first launch the application:

SKIPSILENTINIT=1

Here's the command-line syntax:

\\server\path\Setup.exe SKIPSILENTINIT=1

The initialization must occur, it's just a matter of timing. In the event that you have many seats and might not have sufficient time to deploy the application, you can use this command option.

Stage 4: Maintaining the installations

You can maintain the installed software in the following ways:

- **Modify** — to change which program features and languages are installed
- **Repair** — to resolve technical issues
- **Remove** (or “uninstall”) — to prepare to upgrade to the newest version of the software

You can maintain a single installation of the software by using the Windows Control Panel, or you can maintain multiple installations of the software by using command lines.

Contents

Maintaining a single installation	15
Maintaining multiple installations	15

Maintaining a single installation

You can use the Windows Control Panel to remove a single installation of the software.

To remove a single installation of the software

- 1 Log on to the workstation.
- 2 Open the **Control Panel**.
- 3 On Windows 10, Windows 8, Windows 7, click **Programs**, and click the **Uninstall a program** link.
- 4 Choose **Corel PaintShop Pro X9** from the list.
- 5 Click **Uninstall/Change**.

Maintaining multiple installations

You can use a command line to repair, update, or remove the software.

By using a push-installation method to deploy your command line (see page 6), you can maintain multiple installations of the software.

Command-line functions

Repairing the software	16
Updating the software	16
Removing the software	17



Help updates for locally-stored Help files (all languages) are also available from the Corel support website. Follow the "Updating the software" instructions to update the server image and deploy the Help to workstations.



Applying a patch may require you to re-enter your customer information and serial number. Just in case, keep these details handy.

Repairing the software

You can use the command-line switch `/reinstall` to repair the software by reinstalling it from the specified server image.

Here's the command-line syntax:

```
\\server\path\Setup.exe /reinstall
```

To silently repair the software, include the `/qn` switch:

```
\\server\path\Setup.exe /qn /reinstall
```

Updating the software

Corel periodically releases Microsoft patch (MSP) files, or "patches", for its products. Installing patches helps to keep the software up-to-date.

By default, the software is configured to use an automatic-update feature to detect when patches are available.

However, you yourself must deploy updates to the workstations if you choose to disable the automatic-update feature (see "Controlling network-related features of the software" on page 12 and "To create a server image" on page 3). In this scenario, you must monitor the website for Corel Support Services (www.corel.com/support); when a patch is made available, you can then download it to the server.

Patching the server image

To apply a patch to the server image, use the following command-line syntax (where `Patch.exe` is the filename of the patch):

```
Patch.exe /a
```

If desired, you can include the location of the server image:

```
Patch.exe /a "\\server\path"
```

Patching the workstations

To deploy the update from the patched server image to the workstations, use the following command-line syntax:

```
\\server\path\Setup.exe /reinstall
```

To silently deploy from the patched server image, include the `/qn` switch:

```
\\server\path\Setup.exe /qn /reinstall
```



The **/uninstall** switch can be used in place of **/x**.



For a list of parameters for **/q**, see "Limiting the setup UI with /q" on page 7.

Removing the software

You can use the command-line switch **/x** to remove the software.

Here's the command-line syntax:

```
\\server\path\Setup.exe /x
```

Silent removal

You can include the **/q** switch in your command line if you want to silently remove the software:

```
\\server\path\Setup.exe /x /q
```

Copyright © 2016 Corel Corporation. All rights reserved.

Corel® PaintShop® Pro X9 Deployment Guide

Product specifications, pricing, packaging, technical support and information ("specifications") refer to the retail English version only. The specifications for all other versions (including other language versions) may vary.

INFORMATION IS PROVIDED BY COREL ON AN "AS IS" BASIS, WITHOUT ANY OTHER WARRANTIES OR CONDITIONS, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, OR THOSE ARISING BY LAW, STATUTE, USAGE OF TRADE, COURSE OF DEALING OR OTHERWISE. THE ENTIRE RISK AS TO THE RESULTS OF THE INFORMATION PROVIDED OR ITS USE IS ASSUMED BY YOU. COREL SHALL HAVE NO LIABILITY TO YOU OR ANY OTHER PERSON OR ENTITY FOR ANY INDIRECT, INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES WHATSOEVER, INCLUDING, BUT NOT LIMITED TO, LOSS OF REVENUE OR PROFIT, LOST OR DAMAGED DATA OR OTHER COMMERCIAL OR ECONOMIC LOSS, EVEN IF COREL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR THEY ARE FORESEEABLE. COREL IS ALSO NOT LIABLE FOR ANY CLAIMS MADE BY ANY THIRD PARTY. COREL'S MAXIMUM AGGREGATE LIABILITY TO YOU SHALL NOT EXCEED THE COSTS PAID BY YOU TO PURCHASE THE MATERIALS. SOME STATES/COUNTRIES DO NOT ALLOW EXCLUSIONS OR LIMITATIONS OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, SO THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.

Corel, the Corel logo, Knowledge Base, PaintShop, and WinZip are trademarks or registered trademarks of Corel Corporation and/or its subsidiaries in Canada, the U.S. and/or other countries. All other product names and any registered and unregistered trademarks mentioned are used for identification purposes only and remain the exclusive property of their respective owners.