COREL™

CorelDRAW®
GRAPHICS SUITE X6

Deployment Guide

Introduction

Welcome! This guide is intended to support you, the administrator, in deploying CorelDRAW® Graphics Suite X6 to your network.

Are you new to network deployment? For a nontechnical introduction to the concepts and processes discussed in this guide, please see the companion *Corel® Beginner's Guide to Network Deployment*.

Contents

Conventions Please note the following conventions used in this guide.

Wherever you see this	You'll find
italicized text	A placeholder for user-specified information, such as a path or filename
`bold monospace text`	A reference to programming syntax

For more information The following resources offer additional support.

For information on	See the following resource
deploying Corel® software: concepts & processes, general advice & best practices	*Corel Beginner's Guide to Network Deployment*
Corel Corporation, or a specific Corel software product	Corel® website: www.corel.com
support for Corel software	Corel® Support Services website: www.corel.com/support
the Microsoft® Windows® Installer (MSI) technology, which is used to install Corel software	Microsoft® website

Stage 1: Preparing for deployment

Before beginning the deployment, take the time to familiarize yourself with your network and the software you want to deploy to it. Doing so can help your administrative duties run as smoothly as possible.

In this stage

Getting to know your network

Make sure to consider your network requirements for the deployment.

Administrative factors Does your organization (or do you, as administrator) have any particular requirements for deploying and maintaining software?

Server factors Does your server have any special characteristics or constraints?

Workstation factors Do your workstations have any special characteristics or constraints?

User factors Do your workstation users have any special workflows?

For more information For general guidance on assessing your deployment needs, please refer to "Processes: Stage 1" in the *Corel Beginner's Guide to Network Deployment*.

Getting to know the software

Make sure that your server and workstations are eligible for the software.

Server requirements You can create an image of CorelDRAW Graphics Suite X6 on a server that meets the following requirements:

- Operating system with the latest service pack and critical updates: Windows Server® 2008 or Windows Server® 2008 R2
- 1 GB of server space (for a typical installation without extra content)
- Microsoft® .NET Framework 3.5 and 4.0
- Microsoft Windows Installer 4.5

Windows® Terminal Server (2008 or 2008 R2) is supported, but please note the following:

- You must use the CorelDRAW Graphics Suite X6 DVD to install the software on the server.
- Windows Media® Player is required if you want workstation users to have access to the Hints docker and the Video Browser.
- Corel Corporation does not provide technical support for installing third-party software included with the product.

For details on server permissions, see "Server setup" on page 4.

Workstation requirements

You can install CorelDRAW Graphics Suite X6 on workstations that meet the following requirements:

- Operating system with the latest service pack and critical updates: 32-bit or 64-bit version of Windows® 7 or Windows Vista®, or 32-bit version of Windows XP
- Pentium® 4, AMD Athlon™ 64, or AMD Opteron™
- 1 GB of RAM
- 1 GB of hard-disk space (for a typical installation without extra content)
- Mouse or tablet
- 1024 × 768 (or greater) screen resolution; 768 × 1024 on a tablet PC
- Microsoft® Internet Explorer® 7.0 or later
- Microsoft® .NET Framework 3.5 and 4.0
- Microsoft Windows Installer 4.5
- Windows Media Player 10 or later (for Hints docker and Video Browser)

For details on workstation permissions, see "Workstation setup" on page 4.

For more information

For important notes on installing and using the software, it's a good idea to review the Readme file for the software.

To access the Readme file on the installation disc

- Run the following command line (where $X:$ is the disc drive):

  `X:\English\Readme.html`

Readme files for other languages can be accessed from their respective language folders on the DVD. Examples:
- `X:\French\Lisez-moi.html`
- `X:\German\Infodatei.html`

Setting up your network accordingly

After familiarizing yourself with your network and the software you want to deploy to it, you can set up your network accordingly.

Server setup Do the following:
- Make sure that you are either a local administrator or an administrator for the domains and workgroups that you are managing.
- Create a shared network location for the server image, and make sure that you have read/write access to that location.

If you want to make extra content (such as clipart and videos) available to workstation users, do the following:
- Set up a shared network location, and make sure that users have access to that location.
- Use the program DVD to install or copy the content to the shared location.
- Download the zipped video files from the following FTP site, and using WinZip®, unzip them to the shared location that stores the other extra content.
  - **FTP site:** ftp://cfs2.corel.com/
  - **User name:** CDGSX6_EXT
  - **Password:** Uyg65fW7

You must use WinZip to unzip the video files. Otherwise, one of the videos will not be displayed correctly.

The video files are extracted to a folder (called **Languages**) that must reside in the shared location that contains the other content folders such as **Clipart**, **Fonts**, and **Photos**.

Workstation setup Do the following:
- Make sure that you are — and anyone else who will be installing the software is — either a local administrator or an administrator for the domains and workgroups that you are managing.
- Make sure that you have — and anyone else who will be installing the software has — read access to the server-image location.
- Make sure that the workstations have Microsoft .NET Framework 3.5 and 4.0 installed.
- Make sure that the workstations have Windows Installer 4.5 installed.

To run Bitstream® Font Navigator®, workstation users require either administrator-level privileges (on Windows 7 and Windows Vista) or "Power User" privileges (on Windows XP).

GPOs

To more easily manage the access rights of workstation users — especially for pull-installation scenarios — you may want to consider using Group Policy Objects (GPOs). The *Corel Beginner's Guide to Network Deployment* offers basic information on GPOs. For more advanced information, please refer to the Software Development Kit (SDK) for Group Policy.

Stage 2: Creating the server image

Highlights:

After preparing for deployment, you're ready to create the server image.

To create a basic image, you run a standard command line. To create a customized image, you can modify this command line — or you can modify certain aspects of the server image or its setup. If you want to support multiple installation types, you can create multiple server images.

In this stage

Creating a basic image

To create a basic server image, you must initialize the setup. After setting up the image and tweaking it as necessary, you can use it for deployment.

In this section

Initializing the setup

To initialize the setup for creating a server image, you use a command line. You can create either a 32-bit server image or a 64-bit image.

The following command line lets you create a basic 32-bit server image from the installation disc (where **X:** is the disc drive):

```
X:\x86\Setup.exe /a
```

The following command line lets you create a basic 64-bit server image:

```
X:\x64\Setup.exe /a
```

If your deployment scenario requires using **msiexec.exe** instead of **Setup.exe**, see page 38.

Event logging

If you want to create a log file of installation events, include the **/l** switch in your command line. For details on this switch, see page 12.

Limited setup UI

If you want to limit the amount of user interface (UI) encountered when creating a server image, include the **/q** switch in your command line. For details on this process, see "Limited setup UI" on page 7.

Setting up the server image

If your **/a** command line succeeds, the setup initializes in one of two ways:

- with full UI — for the standard command line
- with limited (or no) UI — for a command line customized to include **/q**

Full setup UI The setup UI takes you step-by-step through the process of setting up the server image. A few notes on this process will suffice.

UI component	Notes
End-User License Agreement (EULA)	To create the server image, you must accept — on behalf of your organization — the terms in the license agreement. Installations deployed from the image will not prompt users to review the license agreement.
Image Location	Type the full (UNC) path in the box provided, or click the **Change** button to browse to that location. For best results, limit the path to 90 characters.
Product Updates	By default, automatic updates are disabled. However, you can choose to have workstations detect and download product updates.

At the end of setting up the server image, you will be asked if you want to install extra content. Click **No** to the prompt to prevent potentially confusing messages. For information about making extra content available, see "Making extra content available" on page 33.

Limited setup UI If you want to limit the amount of user interface (UI) encountered when creating a server image, include the **/q** switch in your command line. For details on this switch, see page 13.

Be careful to avoid suppressing (or "silencing") a required user-interface field when introducing a **/q** switch into your command line — otherwise, you may fail to create a valid server image. You may need to include the following public properties in your command line:

- **TARGETDIR="*path*"** — to specify the desired location of the server image. For best results, do not end the path with a backslash (****).
- **ALLOW_PRODUCTUPDATES=0** — to disable automatic updates (and in-product messaging) on the workstations. By default, automatic updates are enabled for workstations installed from a server image.

- additional installation properties supported by the setup. For details, see "Using public properties" on page 14.

Finalizing the server image

This section describes recommended steps to take before deploying from the server image.

Optionally, you can finalize the image in the following ways:
- Customize the registry settings you want to deploy (see page 33).
- Customize the setup fileset (see page 35).

Patching the image　To avoid deploying the software twice, you may want to check for software updates and apply them to the image as necessary. For details, see page 19.

Testing the image　You may want to test your finalized image with a small subset of workstations before rolling it out to your entire organization.

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

Creating a customized image

If you require a more customized server image than offered by command-line switches (see page 12) and public properties (see page 14), you may want to consider the specialized scenarios discussed in the Appendix.

MSI files　You can modify the Microsoft Windows Installer (MSI) files used to install the software. See page 30.

MST files　You can apply Microsoft® transformation (MST) files to the setup. See page 32.

Registry keys　You can edit the registry settings for the server image. See page 33.

Setup fileset　You can customize the setup fileset. See page page 35.

Installation settings　You can customize the settings used to install the software on the workstations. See page 37.

Creating multiple images

If your workstations require different configurations of the software, you can create one server image for each installation type.

If you want to support both 32-bit and 64-bit installations of the software, you must create two server images: one for the 32-bit installations, and one for the 64-bit installations.

Product updates When a product update becomes available, you will need to apply it to each server image. For details on image patching, see page 20.

Naming convention Use a naming convention that makes it easy to distinguish the server images from one other.

Stage 3: Installing the software

After creating the server image, you can use it to install the software on the workstations in one of two ways:

- manually, by having users "pull" the software to their own workstations
- automatically, by "pushing" the software to the workstations on their users' behalf

Pulling the software

If users have both access to the server and administrator-level rights to their workstations, they can install (or "pull") the software themselves. To do this, they run the setup from the server image.

Preparing for pull installation

To ensure that the pull-installation process runs as smoothly as possible, take the time to prepare for it.

Requirements
- Confirm that each workstation meets the minimum requirements for the software (see "Workstation requirements" on page 3).
- Confirm that each workstation user has read-only access to the server image and administrator-level access to the workstation (see also "Workstation setup" on page 4).

Recommendations
- Map the workstations to the server image, and set the workstations to log in to that server location at startup. Doing so ensures that users are always connected to the server image.
- Confirm that all workstations are running the latest version of Microsoft Windows Installer. Doing so simplifies the installation process.

Carrying out a pull installation

Users pull the software to their workstations by running the setup from the server image.

The degree of user interaction required by the setup depends on how you've customized the server image and the deployment process.

If you prefer to restrict user customization, you may want to offer multiple server images — one per set of installation options. For details, see page 9.

To pull the software to a workstation

1 Browse to the location of the server image, and double-click **Setup.exe**.

2 Carry out the installation, specifying any desired options, by following the on-screen instructions.

For best results, limit the installation path to 90 characters.

If you experience a scripting error, cancel the setup and try again. Doing so will resolve the error by installing required components of the Microsoft .NET Framework.

Pushing the software

To "push" the software from the server image to the workstations, you must write a command line that includes the following:

- the filename of the setup you want to use
- any desired switches, to control how the setup runs
- any desired public properties, to specify installation settings

Using this command line, you can install the software on the workstations by using any supported push method.

In this section

Specifying the setup file

The most important item to specify in your command line is the executable file for the setup: the **Setup.exe** file on the server image you want to use:

- ***server**path*\\x86\\Setup.exe** — to push the 32-bit version of the software to 32-bit or 64-bit operating systems
- ***server**path*\\x64\\Setup.exe** — to push the 64-bit version of the software to 64-bit operating systems.

You cannot use the 64-bit version of the software on 32-bit operating systems.

If your path contains spaces, you must place quotation marks around it:

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

If your network requires deployment via **msiexec.exe** instead of **Setup.exe**, see page 38.

Using switches

By using switches in your command line, you can control the setup process. Listed in this section are the switches most commonly used when installing the software.

For general information on switches, along with details on switch syntax, please "Switches" in the *Corel Beginner's Guide to Network Deployment*.

For a list of all command-line switches for Microsoft Windows Installer, please see the Microsoft website.

/l Event logging

Use **/l** to log installation events.

Parameter	What it logs
i	Status messages
w	Nonfatal warnings
e	All error messages
a	Initiated actions
r	Action-specific records
u	User requests
m	Error messages for out-of-memory warnings or fatal exits
o	Error messages for insufficient hard-disk space during server installation
p	Terminal properties
*	All of the above, in a single log file NOTE: Alternatively to **/l***, you can use **/log**.
v	Very detailed information

The default parameters for **/l** are **iwearmo** (/liwearmo "c:\install Logs.txt").

If you want to specify a path for the log file, use the following syntax:

```
/l "path"
```

If you do not specify a path and filename, the log file is created in the current user's temporary (**Temp**) folder.

In the following example, the **/l** switch (with default parameters) is used to log installation events in the specified file:

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

/q   Limited setup UI

Use **/q** to limit the amount of setup UI encountered during installation.

You can use **/q** to prevent users from entering their own registration information, to help enforce specific installation options, or even to perform silent installations (in which no setup UI is visible whatsoever).

Parameter	What the user sees during installation
n	No UI whatsoever. Errors are logged in a file that you can control by using **/l** (see page 12). This is the default parameter. NOTE: Alternatively to **/qn**, you can use **/quiet**.
b	Basic UI: a progress bar and a **Cancel** button. If the user pushes the **Cancel** button, the installation is rolled back upon confirmation. NOTE: Alternatively to **/qb**, you can use **/passive**.
b!	Basic UI: only a progress bar. The user cannot cancel the installation.
b+	Basic UI: a progress bar and a **Cancel** button. If the user pushes the **Cancel** button, the installation is rolled back (without first prompting for confirmation).
r	Reduced UI: A progress bar, along with a page containing information about the installation. The user can cancel the installation.
f	Full UI

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

In the following example, all setup UI is suppressed, and any errors are recorded in the specified log file:

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

/?   Help

The **/?** switch (or **/help**) displays Help for the setup.

Using public properties

By using public properties in your command line, you can specify installation settings. Listed alphabetically in this section are the properties most commonly used during software installation.

For general information on public properties, along with details on property syntax, please see the *Corel Beginner's Guide to Network Deployment*.

For a list of all command-line properties for Microsoft Windows Installer, please see the Microsoft website.

ADDLOCAL Installing specific components

For details on this scenario, see the following topics in the appendix:
- "Deploying specific features" on page 23
- "Deploying specific languages" on page 26

CONTENTDVDINSTALLDIR Pointing installations to the extra content

You can give users access to the extra content installed from the program DVD, along with the videos downloaded from the FTP site, by placing it at a shared location on the server. See "Server setup" on page 4.

To point the installations to this location, use the following property:

    CONTENTDVDINSTALLDIR="path"

For best results, do not end **path** with a backslash (\ ).

DESKTOPSHORTCUTS Opting out of desktop shortcuts

You can prevent the installation of desktop shortcuts by using the following property:

    DESKTOPSHORTCUTS=0

IGNORE_LAUNCH_CONDITIONS Suppressing the software-requirements check

By default, the setup checks whether the computer meets the minimum software requirements (see page 3) before proceeding with the installation. To suppress this check, use the following property:

    IGNORE_LAUNCH_CONDITIONS=1

INSTALLDIR Customizing the installation path

When deploying from the server image, use the following public property to customize the installation path of the software:

    INSTALLDIR="path"

For best results, do not end the path with a backslash (\ ).

REBOOT **Handling reboots**

The **REBOOT** property is used with one of the following values to specify how the installation process handles any required reboots.

Value	What it does
Force (or **F**)	Prompts for reboot after installation. If the setup UI is suppressed, the computer is automatically rebooted after installation. NOTE: Alternatively to **REBOOT=Force**, you can use the **/forcerestart** switch.
Suppress (or **S**)	Prompts for reboot during installation (if required), but reboots automatically after installation. If the setup UI is suppressed, all required reboots are automatic. NOTE: Alternatively to **REBOOT=Suppress**, you can use the **/promptrestart** switch.
ReallySuppress (or **R**)	Suppresses all reboots and all reboot prompts, both during and after installation. NOTE: Alternatively to **REBOOT=ReallySuppress**, you can use the **/norestart** switch.

USERNAME **Inputting the user name**

To input the user name for the installation, use the following property:

    USERNAME="*user name*"

Running your command line

Your command line can be used to push the software to the workstations through any of the following:

- **a batch file** — see the Microsoft website for details
- **a GPO** — see the Group Policy SDK for details
- **a third-party push technology** — see the manufacturer's documentation for details

For many administrators, using a third-party push technology is the preferred push method. This section offers general guidelines on using a few such technologies.

SCCM At the time of this writing, Corel has verified support for Microsoft® System Center Configuration Manager (SCCM) only insofar as its Microsoft Systems Management Server (SMS) components.

For complete information on using SCCM, please see the Microsoft website.

SMS   For deployment to the workstations on a "per-system" basis using package definition files (or "packages"), CorelDRAW Graphics Suite X6 supports the use of Microsoft Systems Management Server (SMS).

You can create a package from scratch — or you can create one from the existing SMS files included with CorelDRAW Graphics Suite X6 (in the **Administrator** folder on the installation disc or the server image).

For complete information on using SMS files, please refer to your resource kit for Microsoft Systems Management Server.

⚠️   The recommendations posted on the Microsoft website have not been tested with CorelDRAW Graphics Suite X6 and must therefore be used at your discretion.

IntelliMirror   For deployment to the workstations, CorelDRAW Graphics Suite X6 supports the use of the IntelliMirror technology, which is part of the Windows Server 2003, Windows Server 2008, and Windows Server 2008 R2 operating systems.

⚠️   "Per-user" installations are not supported.

To make deployment with IntelliMirror as easy as possible, CorelDRAW Graphics Suite X6 includes a ZAP file (in the **Administrator** folder on the installation disc or the server image).

For complete information on IntelliMirror, please see the Microsoft website.

Stage 4: Maintaining the installations

An important part of administering a network is maintaining the software installed on its workstations.

Repairing the software

Repairing the software lets you install (or reinstall) missing (or corrupted) components. You can repair a single installation of the software by using the Windows Control Panel on that workstation, or you can repair multiple workstation installations by using a command line.

Repairing a single installation

You can use the Windows Control Panel to repair a single installation.

To repair an installation with the Windows Control Panel

1   In the Windows Control Panel, display the list of currently installed programs, and choose the program from the list.

2   Click **Uninstall/Change** (for Windows 7 or Windows Vista) or **Change/Remove** (for Windows XP).

3   Enable the **Repair** option, and then click **Repair**.

4   Follow the on-screen instructions.

Repairing multiple installations

You can use a command line to repair multiple installations at a time.

/f   Basic repair

Use the **/f** switch with the **Setup.exe** file on the server image:

        *server**path***Setup.exe /f**

You cannot use the **/f** switch with public properties. For greater control over the repair process, use the **REINSTALL** and **REINSTALLMODE** properties instead of **/f**.

The **/f** switch has the following parameters.

Parameter	What it does
p	Reinstalls missing files
o	Reinstalls missing files and files older than current version
e	Reinstalls missing files and files equal to or older than current version
d	Reinstalls missing files and files different from current version
a	Reinstalls all files
u	Reinstalls all required user-specific registry keys: **HKEY_CURRENT_USER**, **HKEY_USERS**
m	Reinstalls all required computer-specific registry keys: **HKEY_LOCAL_MACHINE**, **HKEY_CLASSES_ROOT**
s	Reinstalls all existing shortcuts
v	Runs from the source files and re-caches the local package

The default parameters for **/f** are **oums**.

REINSTALL and REINSTALLMODE

Advanced repair

For greater control over how the software is repaired, use the **REINSTALL** and **REINSTALLMODE** public properties together.

Property	What it does
REINSTALL=ALL	Reinstalls all program features. If you prefer to reinstall only specific features (see page 23 for a list), use the following syntax: **REINSTALL=***Feature*
REINSTALLMODE=*type*	Specifies the type of reinstallation to perform. The values for this property are the same as the parameters for the **/f** switch, so its default values are **oums**.

Modifying the software

Modifying the software lets you change which program features are installed. You can modify a single installation of the software by using the Windows Control Panel on that workstation, or you can modify multiple workstation installations by using a command line.

Modifying a single installation

You can use the Windows Control Panel to modify a single installation.

To modify an installation with the Windows Control Panel

1 In the Windows Control Panel, display the list of currently installed programs, and choose the program from the list.

2 Click **Uninstall/Change** (for Windows 7 or Windows Vista) or **Change/Remove** (for Windows XP).

3 Enable the **Modify** option, and then click **Next**.

4 Follow the instructions that appear.

Modifying multiple installations

You can use a command line to modify multiple installations at a time.

REINSTALL and REINSTALLMODE

Adding features

You can use the **REINSTALL** and **REINSTALLMODE** properties to add program features. For details on these properties, see page 18.

REMOVE

Removing features

You can use the **REMOVE** public property to remove a program feature:

    REMOVE=*Feature*

For more than one feature, use a comma-separated list:

    REMOVE=*Feature1,Feature2*

For a list of available features, see page 23.

Updating the software

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

In this guide, **Patch.exe** is a placeholder for the filename of the patch. The actual filename varies with each patch.

Locating patches

Many network administrators keep their software up-to-date by monitoring the Corel Support Services website (www.corel.com/support). When a patch is made available, these administrators download it and carry out the patching process themselves.

In this guide, **\\server\path\Patch.exe** is a placeholder for the location and filename of the downloaded patch.

If you prefer to have workstation users patch the software for themselves, you can set up the server image to enable the automatic detection of available patches. For details, see "Setting up the server image" on page 7.

Applying patches

After locating and downloading a patch, you can apply it to the server image and then use the updated image to patch the installed software.

Image patching You can apply a patch to the server image. Optionally, you can begin by extracting the patch files to a specified location.

To extract the patch files to a specified location

To extract the patch files to a specified location (instead of the default **Temp** folder), use the following command-line syntax:

```
\\server\path\Patch.exe /extract_all "location"
```

Be sure to apply all extracted patch files to the server image. Applying only some of the patch files may cause the installations to function incorrectly.

To apply the patch files to the server image

Use a command line that includes the executable file for the patch and the **/a** switch:

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

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

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

Workstation patching After patching the server image, you can deploy the update to the workstations.

To patch the workstations

Use the following command-line syntax:

```
\\server\path\Setup.exe REINSTALL=ALL
```

By default, the **REINSTALLMODE=oums** property is applied. For best results, you may want to specify **REINSTALLMODE=vdm**:

```
\\server\path\Setup.exe REINSTALL=ALL
REINSTALLMODE=vdm
```

For details on **REINSTALL** (and **REINSTALLMODE**), see page 18.

To silently patch the workstations

Use the following command line:

```
\\server\path\Setup.exe REINSTALL=ALL
REINSTALLMODE=vdm /qn
```

Removing the software

You can remove the software from your network. Uninstalling the software cleanly is crucial when the time comes to upgrade to a newer version.

Image removal You cannot automate the removal of a server image. Instead, you must manually delete the image from the server.

For best results with deleting an image, begin by doing the following:

* Make sure that the programs to be removed — and their associated files — are not currently in use on the workstations.
* If you are using Windows Terminal Server, make sure that all users are logged off.

The only way to re-cover a deleted server image is to re-create it by re-running the setup.

Workstation removal You can remove a single installation of the software by using the Windows Control Panel on that workstation, or you can remove multiple workstation installations by using a command line.

Removing a single installation

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

To remove an installation with the Windows Control Panel

1 In the Windows Control Panel, display the list of currently installed programs, and choose the program from the list.

2 Click **Uninstall/Change** (for Windows 7 or Windows Vista) or **Change/ Remove** (for Windows XP).

3 Enable the **Remove** option.

4 If you want to remove user files (such as presets, user-created fills, and customized files), enable the **Remove user files** check box.

5 Click **Remove**.

Removing multiple installations

You can use a command line to remove multiple installations at a time.

/x ### Basic removal

You can use the **/x** switch (or the **/uninstall** switch) to silently remove the software:

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

REMOVE and REMOVEUSERFILES ### Advanced removal

If you want more control over the removal process, use the following two public properties.

Property	What it does
REMOVE=ALL	Removes all features
REMOVEUSERFILES=value	Specifies whether to remove user files: • **0** — no • **1** — yes

Troubleshooting the software

For help troubleshooting any issues with the software, please visit the Corel® Knowledge Base™ (www.corel.com/knowledgebase), an online repository of FAQs and articles.

Appendix: Specialized deployment scenarios

If you require a more specialized deployment scenario, the supplementary topics in this appendix may be of interest to you.

Deploying specific features

When performing a basic or silent installation with the **/q** switch (see page 13), you may want to use the **ADDLOCAL** property to specify which software features to install. You can specify a single feature:

> **ADDLOCAL=*Feature***

Or you can specify a comma-separated list of components:

> **ADDLOCAL=*Feature1,Feature2***

You can even specify all features:

> **ADDLOCAL=ALL**

If you want to install nearly all features, naming them individually would be quite tedious. In this scenario, you can use the **REMOVE** public property to specify the features to subtract from **ADDLOCAL=ALL**:

> **ADDLOCAL=ALL REMOVE=*Feature1,Feature2***

Feature names are case-sensitive.

Specifying software features by feature name

CorelDRAW Graphics Suite X6 components are organized into the following child features of **MainApp**.

Software component	"MainApp" child feature
CorelDRAW	**Draw**

Software component	"MainApp" child feature
Corel® PHOTO-PAINT™	`PP`
Corel CAPTURE™	`Capture`
Corel® CONNECT	`CONNECT_App`
Utilities	`Utility`
Import/export filters	`Filters`
Guidebook (PDF file)	`UseGuide`

Some editions of the software do not include certain features listed in this section.

When you specify a parent feature (such as **MainApp**), only program files are included. To add other child features, you must specify them individually. And remember: Feature names are case-sensitive.

CorelDRAW

The CorelDRAW component (feature name **Draw**) contains the following child features.

CorelDRAW component	"Draw" child feature
Program files	`DrPFiles`
Help files	`DrHFiles`
Industry-standard labels	`DrLabels`
Plug-in filters	`DrFilt`
Predesigned fills and effects	`DrPres_App`
Predesigned graphics	`DrTiFra`
Short videos	`DrVideoTips`

Specifying **Draw** includes only the CorelDRAW program files (DrPFiles). To include other child features, you must specify them individually.

Corel PHOTO-PAINT

The Corel PHOTO-PAINT component (feature name **PP**) contains the following child features.

Corel PHOTO-PAINT component	"PP" child feature
Program files	`PPPFiles`
Help files	`PPHFiles`
Short videos	`PPVideoTips`
Filters	`PPFilt`

Corel PHOTO-PAINT component	"PP" child feature
Predesigned graphics	`PPTiFra`
Photo-enlarger plug-in	`Photozoom`

Specifying `PP` includes only the Corel PHOTO-PAINT program files (PPPFiles). To include other child features, you must specify them individually.

Corel CAPTURE The Corel CAPTURE component (feature name `Capture`) contains the following child features.

Corel CAPTURE component	"Capture" child feature
Program files	`CaPFiles`
Help files	`CaHFiles`

Specifying `Capture` includes only the Corel CAPTURE program files. To include the Help files, you must specify the child feature (CaHFiles).

Corel CONNECT The `CONNECT` feature contains the files required to run Corel CONNECT.

Utilities The utilities (feature name `Utility`) include the following child features.

Utility	"Utility" child feature
Duplex wizard (for two-sided print jobs)	`DrDupWiz`
Bitstream Font Navigator	`FontNav`
Windows Shell Extension	`ShellExt`
Visual Basic® for Applications (VBA)	`VBAFiles`
Visual Studio® Tools for Applications(VSTA)	`VSTA`

Specifying `Utility` does not include its child features. You must individually specify each desired utility.

Filters The `Filters` feature is a parent to `FiltersAdditional`, which contains both secondary-level import/export filters (CUR, EXE, FMV, ICO, PCD, PCX, SCT, VSD, XCF, and XPM) and tertiary-level import/export filters (GEM, HTM, IMG, MET, MOV, NAP, PIC, QTM, and SHW) for the software.

Specifying `Filters` does not include its child features. You must instead specify its child feature, `FiltersAdditional`.

User guide The `UseGuide` feature contains, as a PDF file, the Guidebook for the software.

Deploying specific languages

When performing a basic or silent installation with the **/q** switch (see page 13), you may want to use the **ADDLOCAL** property to specify which software languages to support. You can specify a single language:

> **ADDLOCAL=*LanguageCode***

Or you can specify a comma-separated list of languages:

> **ADDLOCAL=*FEATURE1,LanguageCode1,LanguageCode2***

Using language codes, you can specify language modules or writing tools.

Some editions of the software do not include certain languages listed in this section.

Language codes are case-sensitive.

Specifying UI modules by language code

A language module includes both the software UI and the writing tools for that language. By installing multiple language modules, you can enable *dynamic language switching* — the ability to change the language of the software UI.

The following table lists the available language modules for the software, along with their corresponding language codes.

Language module	Language code
Brazilian Portuguese	**BR**
Chinese (Simplified)	**CS**
Chinese (Traditional)	**CT**
Czech	**CZ**
Dutch	**NL**
English	**EN**
Finnish	**SU**
French	**FR**
German	**DE**
Hungarian	**MA**
Italian	**IT**

Language module	Language code
Japanese	JP
Korean	KR
Polish	PL
Russian	RU
Spanish	ES
Swedish	SV
Turkish	TR

Specifying writing tools by language code

Each set of writing tools includes Spell Checker and may include Thesaurus or Grammatik (or both). The available languages are as follows.

Language	Code	Spell Checker	Thesaurus	Grammatik
Afrikaans	Afrika	✓		
Catalan	Catalan	✓		
Czech	Czech	✓		
Danish	Danish	✓	✓	
Dutch	Dutch	✓	✓	✓
English	English	✓	✓	✓
Finnish	Finnish	✓		
French	French	✓	✓	✓
Galician	Galician	✓		
German	German	✓	✓	✓
Greek	Greek	✓		
Icelandic	Iceland	✓		
Italian	Italian	✓	✓	
Norwegian	Norway	✓	✓	
Polish	Polish	✓		
Portuguese	Portugue	✓		

Language	Code	Spell Checker	Thesaurus	Grammatik
Russian	**Russian**	✓		
Slovak	**Slovak**	✓		
Sotho	**Sotho**	✓		
Spanish	**Spanish**	✓	✓	✓
Swedish	**Swedish**	✓	✓	
Tswana	**Tswana**	✓		
Turkish	**Turkish**	✓		
Xhosa	**Xhosa**	✓		
Zulu	**Zulu**	✓		

In most cases, writing tools are automatically installed with their corresponding language module—and to match the installed keyboard languages of the operating system.

The parent feature for the writing tools is named **WTools**. However, specifying **WTools** does not include its child features. You must individually specify each desired set of writing tools.

Deploying customized workspaces

The software installs a selection of workspaces for use in CorelDRAW, and Corel PHOTO-PAINT. However, the software also lets you create and deploy customized workspaces.

Understanding workspaces

Information about the layout of a workspace is saved to XML files.

One way to customize a workspace is to manually edit its XML files. However, the easiest way is to design the workspace from within the program; in doing so, updated XML files are automatically generated.

Image location The server image stores the default XML files for each workspace at the following locations:

- **cgs16\Program Files\Corel\CorelDRAW Graphics Suite X6\ *ProgramName*\Workspace —** location for the 32-bit version of the software

- **cgs16\Program Files 64\Corel\CorelDRAW Graphics Suite X6** *ProgramName***\Workspace —** location for the 64-bit version of the software

Installed location   When the software is deployed from the server image to the workstations, the workspace XML files are copied to one of the following locations (where *X:* is the drive where the software is installed):

- *X:***Program Files\Corel\CorelDRAW Graphics Suite X6** *ProgramName***\Workspace**
- *X:***Program Files (x86)\Corel\CorelDRAW Graphics Suite X6** *ProgramName***\Workspace**

If a workstation user customizes a program workspace, the XML files for that workspace are copied to the user's profile, updated to reflect the workspace changes, and used to re-create the customized workspace whenever that user runs the application. The default workspaces deployed from the server image to the workstations remain untouched at their installed locations.

Understanding the workspace XML files

The workspace XML files contain the following main tags.

XML tag	What it does
`applicationInfo`	Identifies the program
`commandBars`	Defines the content of all menus, all toolbars, and the status bar
`frame`	Defines the layout of the program window
`items`	Defines the items that appear in the menus and toolbars
`customizationList`	Defines all customizable items on the command bars. NOTE: Changes must be made within the XML file, not the program.
`shortcutKeyTables`	Defines all shortcut keys and associated items

The XML entries contain Globally Unique Identifiers (GUIDs), which are used to identify various user-interface objects in the system. For example, a command bar contains a list of items, each with its own GUID:

- If the item order is changed, the entries are simply reordered.
- If an item is added, a new item entry with a GUID is added.

CorelDRAW default workspaces   The default workspaces for CorelDRAW are installed in the following subfolders of **Draw\Workspace**:

- **_BootDefault** — standard CorelDRAW workspace
- **Adobe Illustrator** — Adobe® Illustrator® workspace

Each subfolder contains an editable **CorelDRAW.ini** file, which specifies a few basic program settings. Subfolders may also contain the following workspace XML files (which can be edited — or added, if missing):

- **DRAWUIConfig.xml** — settings for the main program
- **PreviewUIConfig.xml** — settings for the **Print preview** window
- **FiltMan.ini** — settings for the Filter Manager

Corel PHOTO-PAINT default workspaces

The default workspaces for Corel PHOTO-PAINT are installed in the following subfolders of **PHOTO-PAINT\Workspace**:

- **_BootDefault** — standard Corel PHOTO-PAINT workspace
- **Adobe Photoshop** — Adobe® Photoshop® workspace

Each subfolder contains an editable **CorelPP.ini** file, which specifies a few basic program settings. Subfolders may also contain the following workspace XML files (which can be edited — or added, if missing):

- **PPUIConfig.xml** — settings for the main program
- **PreviewUIConfig.xml** — settings for the **Print preview** window

Working with MSI files

Stored in the **Setup** folder on the installation disc, the main Microsoft Windows Installer (MSI) file for the setup — **\x86\CGS16\Setup.msi** (for 32-bit installations) or **\x64\CGS16\Setup_x64.msi** (for 64-bit installations) — contains a database of all features, registry keys, folders, and shortcuts for the software.

The server image contains a copy of the main setup MSI file — the MSI tables within which you can modify for deployment to the workstations.

Various components of the software are stored in additional MSI files. Some of these MSI files are required by the setup, while others represent optional features of the software.

You can exclude optional MSI files from the setup by creating a customized server image. Excluding an MSI file prevents the corresponding feature from being installed. For details, see page 30.

Setup files Besides **Setup.msi**, the setup requires additional MSI files to carry out the installation. These required MSI files include the following:

- **x86\ICA.msi** or **x64\ICA_64.msi**
- **x86\CGS16\ShellExt.msi** or **x64\CGS16\ShellExt.msi**

Language modules The setup requires at least one language module, MSI files for which are stored at **x86\CGS16** (32-bit) or **x64\CGS16** (64-bit) on the installation disc. The language modules that are available vary with each edition of the software.

Language module	MSI file
Brazilian Portuguese	**BR.msi** (32-bit) or **BR_x64.msi** (64-bit)
Chinese (Simplified)	**CS.msi** (32-bit) or **CS_x64.msi** (64-bit)
Chinese (Traditional)	**CT.msi** (32-bit) or **CT_x64.msi** (64-bit)
Czech	**CZ.msi** (32-bit) or **CZ_x64.msi** (64-bit)
Dutch	**NL.msi** (32-bit) or **NL_x64.msi** (64-bit)
English	**EN.msi** (32-bit) or **EN_x64.msi** (64-bit)
Finnish	**SU.msi** (32-bit) or **SU_x64.msi** (64-bit)
French	**FR.msi** (32-bit) or **FR_x64.msi** (64-bit)
German	**DE.msi** (32-bit) or **DE_x64.msi** (64-bit)
Hungarian	**MA.msi** (32-bit) or **MA_x64.msi** (64-bit)
Italian	**IT.msi** (32-bit) or **IT_x64.msi** (64-bit)
Japanese	**JP.msi** (32-bit) or **JP_x64.msi** (64-bit)
Korean	**KR.msi** (32-bit) or **KR_x64.msi** (64-bit)
Polish	**PL.msi** (32-bit) or **PL_x64.msi** (64-bit)
Russian	**RU.msi** (32-bit) or **RU_x64.msi** (64-bit)
Spanish	**ES.msi** (32-bit) or **ES_x64.msi** (64-bit)
Swedish	**SV.msi** (32-bit) or **SV_x64.msi** (64-bit)
Turkish	**TR.msi** (32-bit) or **TR_x64.msi** (64-bit)

Programs To install the programs included with the software, the setup uses the following MSI files at **x86\CGS16** (32-bit) or **x64\CGS16** (64-bit) on the installation disc.

Program	MSI file
Bitstream Font Navigator	**FontNav.msi** (32-bit), or **FontNav_x64.msi** (64-bit)
Corel CAPTURE	**Capture.msi** (32-bit), or **Capture_x64.msi** (64-bit)

Program	MSI file
Corel CONNECT	**Connect.msi** (32-bit), or **Connect_x64.msi** (64-bit)
Corel PHOTO-PAINT	**PHOTO-PAINT.msi** (32-bit), or **PHOTO-PAINT_x64.msi** (64-bit)
CorelDRAW	**Draw.msi** (32-bit), or **Draw_x64.msi** (64-bit)

Features   To install the various components of the software, the setup uses additional MSI files stored at **Setup\x86\CGS16** (32-bit) or **Setup\x64\CGS16** (64-bit). These files include the following.

Feature	MSI file
Ghostscript general public license, for EPS and PostScript® import	**Ghostscript.msi**
Microsoft Visual Basic for Applications (VBA)	**VBA.msi** (32-bit), or **VBA_x64.msi** (64-bit)
Microsoft Visual Studio Tools for Applications (VSTA)	**CGS_VSTA.msi** (32-bit), or **CGS_VSTA_x64.msi** (64-bit)

Working with MST files

If you want to apply a Microsoft transformation (MST) file to the setup, simply apply it to your command line.

By default, MST files are applied to the **Setup.msi** file for the setup.

To apply a transformation to the setup

1 Store the MST file on the server image.

2 Use the **TRANSFORMS** property to specify that MST file, as shown here (where *MST* is the filename of the MST file):

```
Setup.exe TRANSFORMS="MST"
```

If the MST file is not in the same folder as **Setup.exe**, *MST* must specify the full path and filename of the MST file.

Applying a transformation to a specific MSI file

To apply an MST file to a specific MSI file, use the following syntax (where *MSI* is the filename of the MSI file, not including its extension; and where *MST* is the filename of the MST file, including its extension):

```
Setup.exe TRANSFORMS_MSI="MST"
```

If the MST file is not in the same folder as the MSI file, *MST* must specify the full path and filename of the MST file.

Example 1 To apply **my_draw.mst** to **Draw.msi** (in the same folder), use the following syntax:

```
Setup.exe TRANSFORMS_Draw="my_draw.mst"
```

Example 2 To apply **MyTransform.mst** to **PHOTO-PAINT.msi** (in the same folder), use the following syntax:

```
Setup.exe TRANSFORMS_PHOTO-PAINT="MyTransform.mst"
```

Troubleshooting MST files

If you have trouble applying an MST file, do the following:
- Check the syntax of your command line — in particular, the path to the MST file.
- Check your permissions.
- Make sure that the MST file was created correctly.
- Check the associated MSI files.

For additional help with MST files, contact Corel Support Services (www.corel.com/support). Please note, however, that charges will apply.

Working with registry keys

After creating the server image, you may want to edit its registry keys for deployment to the workstations. By doing so, you can avoid having to manually configure the registry keys of each individual workstation installation of the software.

Making extra content available

As previously discussed (see page 4), you can make extra content available to workstation users by copying it to the server share.

You can set the path to the server content by using the **CONTENTDVDINSTALLDIR** public property (see page 14) — or, by editing the appropriate section in the following **HKEY_LOCAL_MACHINE\Software** registry key:

- 32-bit OS — **Corel\Media\CorelDRAW Graphics Suite X6**
- 64-bit OS with (default) 64-bit build — **Corel\Media\CorelDRAW Graphics Suite X6**
- 64-bit OS with 32-bit build — **Wow6432Node\Corel\Media\ CorelDRAW Graphics Suite X6**

Setting up dynamic language switching

Dynamic language switching allows the language of the user interface to be changed, both during and after installation. This feature requires the installation of multiple language modules for the software — one for each desired user-interface language (see page 26). The selection of languages you can implement depends on the license you have purchased.

Two sets of registry settings are required for dynamic language switching:

- workstation settings
- current user settings

Workstation settings　　Workstation settings affect all users. The registry setting for each workstation is as follows:

> **HKEY_LOCAL_MACHINE\SOFTWARE\Corel\CorelDRAW\16.0\ Languages**

Inside this key are language keys — three-letter codes that represent the available languages. These language keys are put in place by the setup, so no intervention is required. Shown here (for reference only) are the three-letter codes for the available languages.

Language	Code
Brazilian Portuguese	**PTB**
Chinese (Simplified)	**CHS**
Chinese (Traditional)	**CHT**
Czech	**CSY**
Dutch	**NLD**
English	**ENU**
Finnish	**FIN**
French	**FRA**

Language	Code
German	**DEU**
Hungarian	**HUN**
Italian	**ITA**
Japanese	**JPN**
Korean	**KOR**
Polish	**PLK**
Russian	**RUS**
Spanish	**ESN**
Swedish	**SVE**
Turkish	**TRK**

Under each language key are the following registry settings:

• **DirName** — specifies the name of the folder for the language-specific files, relative to *InstallationPath***\Languages**

• **UIName** — specifies the name to show in the startup dialog box and on the **Tools ▸ Options ▸ Global** page

Current user settings   The registry setting for each current user is as follows:

> **HKEY_CURRENT_USER\Software\Corel\CorelDRAW\16.0**

The **UILang** setting is not initialized by the setup because the setup cannot determine which user will run the applications. After the user has run the application once, this setting changes to the three-letter code that corresponds to the user-specified language key at the following location:

> **HKEY_LOCAL_MACHINE\SOFTWARE\Corel\16.0\Languages**

Customizing the setup fileset

The main configuration file **Setup.xml**, located in the **x86** or **x64** folder, lists all features included in the setup.

When you create a server image, a copy of the **Setup.xml** file is included. If you want to customize the setup fileset on the server image, you must edit the **Setup.xml** file to reflect your desired changes.

Understanding Setup.xml

The **Setup.xml** file contains four main types of nodes. With an understanding of these four nodes, you can modify the **Setup.xml** file to create a customized server image.

<Msi/> nodes

Each **<Msi/>** node specifies a main setup file, such an executable (EXE) file, a self-extracting ZIP file, or an MSI file that bundles (or "chains") other MSI files.

<Dbm/> nodes

Each **<Dbm/>** node specifies a component of a **<Msi/>** node — typically, an MSI file. A **<Dbm/>** node has the following structure, where *GUID* is the associated GUID, *MSI* is the path and filename of the MSI file, and *name* is a descriptive name for the MSI file:

```
<Dbm productcode="{GUID}" file="MSI"
progresstext="Str.ProgressText.name" />
```

Some **<Dbm/>** nodes also require a condition.

If you want to apply an MST file to the specified MSI file, you can use the following syntax (where *MST* is the path and filename of the MST file):

```
<Dbm productcode="{GUID}" file="MSI"
cmdline="TRANSFORMS=MST"
progresstext="Str.ProgressText.name" />
```

For a list of the available MSI files, see page 30.

<Feature/> nodes

Each **<Feature/>** node specifies a feature that is associated with an MSI file. A **<Feature/>** node has the following format, where *name* is the name of the MSI file and *feature* is the name of the associated feature:

```
<Feature name="Str.Feature.name"
desc="Str.Feature.name.Desc" property="feature">
```

Some **<Feature/>** nodes have subnodes.

For a list of the available features, see page 23.

<Property/> nodes

Each **<Property/>** node specifies a public property. A **<Property/>** node has the following structure, where *property* is the name of the public property and *value* is its value:

```
<Property name="property" value="value" />
```

You can install the software faster by disabling log-file creation: Simply change the value of the **ICA.LogOptions** property to an empty string.

For a list of available public properties, see page 14.

Creating a server image with a customized fileset

You can use **Setup.xml** to create a server image with a customized fileset.

To create a server image with a customized fileset

1 Copy the server image to a new location.

2 Add any desired language-module MSI files to the server image.

3 Edit **Setup.xml** to reflect the additions you've made to the fileset:

- Add an **<Msi/>** node for each new setup file.
- Add a **<Dbm/>** node for each new language-module MSI file.
- Add the following **<Property/>** node for each new language-module MSI file, where *$$* is the language:
  <Property name="Include.$$" value="1" />

4 Remove any unwanted MSI files from the server image.

Some MSI files cannot be removed from the setup fileset. For details, see page 30.

5 Edit **Setup.xml** to reflect the removals you've made from the fileset:

- Remove the corresponding **<Msi/>** node for each removed setup file.
- Remove the corresponding **<Dbm/>** node for each removed MSI file.
- Remove the corresponding **<Feature/>** node and subnode for each removed MSI file.
- Adjust any corresponding **<Property/>** nodes by changing their value.

For example, let's say that you're creating a single-language setup from a multi-language setup. To record the fileset changes in the **Setup.xml** file, you must remove the corresponding **<Dbm/>** and **<Feature/>** nodes for each removed language module. Next, you must set each **<Property/>** node for **Include.$$** to a value of **0**, where *$$* is a removed language module. The **<Property/>** node for **ShowApplicationLanguageSelector** is automatically set to a value of **0**.

Deploying customized installation settings

To modify the settings used to install the software on the workstations, you can use the **Persist.xml** file stored at the root of the server image.

Each entry in **Persist.xml** corresponds to a public property for the software (see page 14). The syntax of each entry is as follows:

<Property name="*property*" value="*value*" />

By modifying the value of an existing entry in **Persist.xml** — or by adding a new entry — you can customize the installation settings deployed from that server image. This deployment method offers an excellent alternative to creating a different server image or customizing a command line.

Example 1 The following **Persist.xml** entry specifies that automatic software updates are enabled:

```
<Property name="ALLOW_PRODUCTUPDATES" value="1" />
```

By changing the value of this entry from **"1"** to **"0"** (and saving this change to **Persist.xml**), you can disable automatic updates for the workstation installations.

Example 2 The following **Persist.xml** entry specifies the serial number for the installation:

```
<Property name="SERIALNUMBER" value="SN" />
```

By changing the **"SN"** value, you can specify a different serial number for the workstation installations.

Deploying the software with msiexec.exe

Alternatively to using **Setup.exe**, you can deploy the software by using **msiexec.exe** — a Microsoft Windows Installer file provided by the Windows operating system. In fact, you must use **msiexec.exe** if you want to interact directly with the MSI files, such as in the following scenario:

• if you want to use the **Setup.msi** file, rather than a start-up script, to install the software by using a Group Policy Object

Command-line syntax As with **Setup.exe**, you use **msiexec.exe** to create command lines that perform a setup-related function. A **msiexec.exe** command line requires the following items:

• a switch that signals the desired action: creating a server image (**/a**); or installing (**/i**), repairing (**/f**), or removing (**/x**) the software
• the location and name of the desired MSI file: typically, **Setup.msi** (located on the installation disc or on the server image).
• any desired public properties
• any desired switches, as specified by the **CHAINER_CMD="switches"** public property (multiple switches are separated by a space)

For example, the following **msiexec.exe** command line uses the file **\\server\CGS16\Setup.msi** to install the software while displaying the full user interface:

```
msiexec.exe /i "\\server\CGS16\Setup.msi"
ADDLOCAL="ALL" CHAINER_CMD="/qf"
```

MST files   As previously explained (see page 32), you can use the **TRANSFORMS** public property to apply an MST file to the setup. By default, all MST files are applied to the **Setup.msi** file. If you want to apply an MST file to a different MSI file, you must use the following syntax (where *MSI* is the filename of the MSI file, not including its extension; and where *MST* is the filename of the MST file, including its extension):

```
TRANSFORMS_MSI="MST"
```

If the MST file is not in the same folder as the MSI file, *MST* must specify the full path and filename of the MST file.

For example, the following **msiexec.exe** command line uses the file **\\server\CGS16\Setup.msi** to install the software silently, applies the transformation **my_draw.mst** to **Draw.msi** (in the same folder), and applies the transformation **my_pp.mst** to **PP.msi** (in the same folder):

```
msiexec.exe /i "\\server\CGS16\Setup.msi"
ADDLOCAL="ALL" CHAINER_CMD="/qn"
TRANSFORMS_Draw="my_draw.mst"
TRANSFORMS_PP="my_pp.mst"
```

Index

CorelDRAW® Graphics Suite X6 Deployment Guide