

System Planning, Deployment, and Best Practices Guide

ZENworks® Application Virtualization

Legal Notices

Novell, Inc., makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc., makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the [Novell International Trade Services Web page \(http://www.novell.com/info/exports/\)](http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 2012 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.
www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see the [Novell Documentation Web page \(http://www.novell.com/documentation/\)](http://www.novell.com/documentation/).

Novell Trademarks

For Novell trademarks, see [the Novell Trademark and Service Mark list \(http://www.novell.com/company/legal/trademarks/tmlist.html\)](http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Guide	5
1 Creating a ZENworks Application Virtualization Solution	7
1.1 ZENworks Application Virtualization Architecture	7
1.2 Packaging-Snapshot Method	8
1.2.1 Machine Configuration	8
1.2.2 Snapshot Process	9
1.3 Packaging-Desktop Scan Method	10
2 Setting Up and Managing a Build Lab	13
2.1 Setting Up a Build Lab	13
2.2 Build Process	14
2.3 Build Testing Process	14
2.4 Profiling the Application	15
2.5 Managing Builds	15
3 Virtualizing Microsoft Internet Explorer	17
4 Troubleshooting	19
4.1 Accessing Internet Resources	19
4.2 Generating Diagnostic Mode Virtual Applications	20
4.3 Roaming User Profiles	20
4.4 FAQs	20

About This Guide

The purpose of this *System Planning, Deployment, and Best Practices Guide* is to help administrators to design a Novell ZENworks Application Virtualization solution and deploy it across a small or large enterprise.

The information in this guide is organized as follows:

- ♦ Chapter 1, “Creating a ZENworks Application Virtualization Solution,” on page 7
- ♦ Chapter 2, “Setting Up and Managing a Build Lab,” on page 13
- ♦ Chapter 3, “Virtualizing Microsoft Internet Explorer,” on page 17
- ♦ Chapter 4, “Troubleshooting,” on page 19

Audience

This guide is intended for ZENworks Application Virtualization administrators.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation.

Additional Documentation

ZENworks Application Virtualization is supported by other documentation (in PDF format) that you can use to learn about and implement the product. For additional documentation, see the [ZENworks Application Virtualization Web site \(http://www.novell.com/documentation/zav10/index.html\)](http://www.novell.com/documentation/zav10/index.html). You can also click *Help > Novell ZENworks Application Virtualization* from within the program to access this information.

1 Creating a ZENworks Application Virtualization Solution

Novell ZENworks Application Virtualization lets you convert most applications that run on Microsoft Windows into self-contained virtual applications. After being virtualized, an application becomes a single, isolated file that runs instantly from anywhere, including a thumb drive or other removable media. Unlike traditional installation methods, the single virtual application file does not require a separate setup process, and does not rely on external components and runtimes, reboots, or administrative privileges. The application is now isolated from other system applications, preventing DLL conflicts and other deployment nightmares, yet the experience for the application's user is unchanged.

For additional documentation, see the [ZENworks Application Virtualization Web site \(http://www.novell.com/documentation/zav10/index.html\)](http://www.novell.com/documentation/zav10/index.html).

The following sections contain more information:

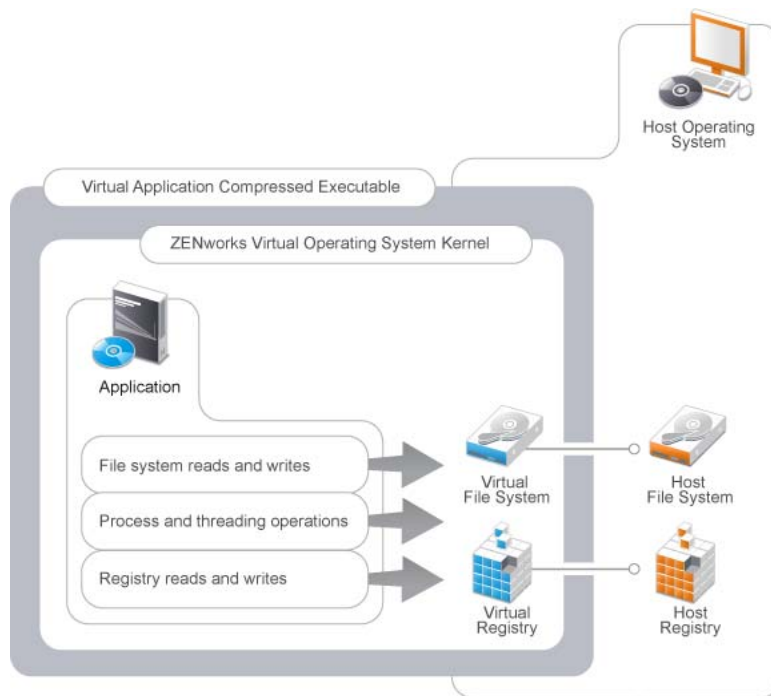
- ♦ [Section 1.1, “ZENworks Application Virtualization Architecture,” on page 7](#)
- ♦ [Section 1.2, “Packaging-Snapshot Method,” on page 8](#)
- ♦ [Section 1.3, “Packaging-Desktop Scan Method,” on page 10](#)

1.1 ZENworks Application Virtualization Architecture

A virtual application is an application executable that contains a virtual runtime environment. It includes all files, registry data, settings, components, runtimes, and other dependencies required for the application to execute immediately. Virtual applications allow application publishers and IT administrators to significantly reduce the costs and complexity associated with development, setup, configuration, deployment, and maintenance of software applications.

ZENworks Application Virtualization uses a sandbox to contain all of a virtual application's data and settings, which are determined by the virtual application's isolation configuration settings. Applications registered to the same sandbox can view and modify each other's virtualized data and settings. Applications in a sandbox work through a virtual machine environment, which enables them to interact with local machine resources, such as, printers and shares.

All applications are registered into a single default sandbox, named `Default`. You can group related applications into a sandbox that is treated as a single management unit. When a sandbox is reset, all application content and data stored in that sandbox is purged and it reverts to the default state.



1.2 Packaging-Snapshot Method

Commercial off-the-shelf applications require combinations of file system and registry entries. To facilitate virtualization of these applications, ZENworks Application Virtualization Studio can take a snapshot of the application installations and automatically configure them based on modifications made to the host system during setup.

- ♦ [Section 1.2.1, “Machine Configuration,” on page 8](#)
- ♦ [Section 1.2.2, “Snapshot Process,” on page 9](#)

1.2.1 Machine Configuration

Make sure the machine you are using is configured to provide the best environment for taking the snapshot.

- ♦ [“Perform the Snapshot on a Clean Machine” on page 8](#)
- ♦ [“Use the Snapshot with Whole-Machine Virtualization” on page 9](#)
- ♦ [“Take the Snapshot on the Earliest Variant of the Target Operating System” on page 9](#)

Perform the Snapshot on a Clean Machine

Performing the snapshot on a clean machine ensures that all dependencies are installed by the application setup. Installing on a machine with existing components can inadvertently include dependencies in the before snapshot and exclude them from the final output.

Use the Snapshot with Whole-Machine Virtualization

If you configure a clean machine by using whole-machine virtualization tools, such as VMware Workstation or Microsoft Virtual PC and then save a before snapshot based on this image, you can snapshot many distinct virtual applications in rapid succession by reverting the machine to the whole-machine virtual state and using the same before snapshot.

Take the Snapshot on the Earliest Variant of the Target Operating System

Most applications can be configured by performing the snapshot on the earliest (least common denominator) target operating system. For example, if an application functions on both Windows XP and Windows 7, perform the first snapshot on Windows XP. If specific configuration is required for later versions of operating systems, such as Windows 7, these can be captured separately and merged into the application by using the platform merge functionality.

1.2.2 Snapshot Process

Use the following best practices as you capture the snapshot and prepare it for virtualization.

- ♦ [“Use the Proper Build Procedure” on page 9](#)
- ♦ [“Saving the Capture and Diff Snapshot” on page 9](#)
- ♦ [“Saving Before Pruning” on page 9](#)
- ♦ [“Running the Native Application to Determine Isolation Characteristics” on page 10](#)
- ♦ [“Setting a Folder as Fully Isolated or Merged” on page 10](#)
- ♦ [“Removing Uninstall Shortcuts” on page 10](#)
- ♦ [“Removing Logs” on page 10](#)
- ♦ [“Testing Isolation” on page 10](#)

Use the Proper Build Procedure

Capture the before snapshot on a clean machine. Install the application and any necessary dependencies/components. If necessary, perform a custom installation to include or exclude items from the build. When the installation is complete, perform the *Capture and Diff* snapshot. Edit the configuration file to remove unnecessary registry keys, files, and folders. Set the desired isolation settings. Finally, build and test the application.

Saving the Capture and Diff Snapshot

When you select a folder to save the *Capture and Diff* snapshot, select *Make New Folder* in ZENworks Application Virtualization Studio. This folder is not included in the snapshot. If the folder is included in the *Capture and Diff* snapshot, remove it by using the *Filesystem* tab.

Saving Before Pruning

Before beginning the pruning process, save a backup of the *Capture and Diff* snapshot (snapshot.xappl) in the same directory as the *Files* folder. Revert to the original *Capture and Diff* snapshot if there is an error.

Running the Native Application to Determine Isolation Characteristics

Run and use the native application to understand what registry keys and folders it updates at runtime. There are many free tools available that can help you track registry changes. This process lets you determine the proper isolation settings for folders and registry keys.

Setting a Folder as Fully Isolated or Merged

To determine if a folder should be set to full isolation or merge isolation, decide if the user would want access to files created within that folder, outside the virtual application environment. Set the folder to merge to allow users to access files, or set it to `full` to keep the folder isolated.

Removing Uninstall Shortcuts

There are two steps to removing uninstall shortcuts during the pruning process:

- 1 If uninstall files have been created by the installation, these need to be removed from the file section and also from the multiple section, where triggers are defined.
- 2 Remove the shortcut from the list of startup files; if a shortcut exists, select *Multiple*, then delete any uninstall files.

Removing Logs

When the application is ready for the final build, make sure to deselect the *Generate Diagnostic-Mode* executable check box. Otherwise logs are created every time the virtual application runs.

Testing Isolation

Complete the following steps to verify that a virtual application does not create unnecessary files outside of the sandbox:

- 1 Take a before snapshot on a clean machine and run the virtual application.
- 2 Use the virtual application as an actual user.
- 3 Close the virtual application and take the *Capture and Diff* snapshot. Verify that there are no unintentional files or folders created outside the sandbox, within the *Filesystem and Registry* tab. If such files or folders exist, edit `xapp1` to set the files and folders to Full Isolation and then retest for fixing isolation issues.

1.3 Packaging-Desktop Scan Method

When you run the Desktop Scan Wizard, there are several best practices to follow for optimal results:

- ♦ Run the desktop scan on the oldest operating system version available. For example, if users use both Windows XP and Windows 7, you should run the desktop scan on the Windows XP machine.

However, for Microsoft Office. For the Microsoft Office suite, you should run the scan from the operating system where it is intended to be used.

- ♦ 32-bit (x86) applications do not show up in the Recipes list when they are installed on a 64-bit (x64) operating system.

- ♦ When you scan LibreOffice or OpenOffice, you should scan the version that includes Java. This allows the packaged application to function on machines that do not have Java installed as well as on machines that do. It is also possible that the desktop scan cannot find an installation of LibreOffice or OpenOffice that does not include Java.
- ♦ Applications that are packaged with the desktop scan might retain settings from the local machine used during the packaging process.
- ♦ Adobe Creative Suite (all versions) do not scan correctly if you scan it from Windows Vista. Use Windows XP or Windows 7 for a desktop scan of Adobe Creative Suite.
- ♦ When you use the desktop scan with browsers, ensure that any plug-ins that are installed outside the browser, such as Windows Media Player, are already installed when the scan is run. This is important because some plug-ins are installed through the `msiexec.exe` process, and are not recognized within the virtualized browser. Plug-ins that are installed from within the browser, such as Adobe Flash, do not have this problem.

2 Setting Up and Managing a Build Lab

If you intend to build virtualized applications on a regular basis, set up a build lab to ensure quality application builds to decrease build time. Perform certain steps on multiple operating systems to ensure maximum compatibility and performance. Because many steps are time-consuming, the process can move more efficiently if steps are performed on multiple machines. This process is referred to as pipeline building, and is useful when virtualizing a high volume of applications.

- ♦ [Section 2.1, “Setting Up a Build Lab,” on page 13](#)
- ♦ [Section 2.2, “Build Process,” on page 14](#)
- ♦ [Section 2.3, “Build Testing Process,” on page 14](#)
- ♦ [Section 2.4, “Profiling the Application,” on page 15](#)
- ♦ [Section 2.5, “Managing Builds,” on page 15](#)

2.1 Setting Up a Build Lab

Set up a build lab with the following six machine configurations:

- ♦ Windows XP, Service Pack 3, x86 architecture
- ♦ Windows XP, Service Pack 2, x64 architecture
- ♦ Windows Vista, Service Pack 2, x86 architecture
- ♦ Windows Vista, Service Pack 2, x64 architecture
- ♦ Windows 7, Service Pack 1, x86 architecture
- ♦ Windows 7, Service Pack 1, x64 architecture

Regularly re-image each machine to an initial configuration, or clean state. Clean machines are required to complete build and testing processes. Save an image of the initial configuration on a local partition, then use third-party software to create an external drive to boot the machine from that image.

Complete the following steps to setup the initial configuration for each machine:

- 1 Set up two partitions; one for the operating system and the other for the clean image.
- 2 Install the operating system.
- 3 Install all current patches for the OS, except for Microsoft Internet Explorer upgrades.
- 4 Create accounts for an administrator and a standard user.
- 5 Disable automatic updates.

The following steps are optional, but recommended:

- ♦ Run Microsoft Internet Explorer 8 once, to prevent repeating the first-run wizard each time you re-image the machine to a clean state.
- ♦ Enable remote desktop protocol access.

- Configure the power options so the machine never enters the sleep state.
- Disable the Microsoft Windows Firewall.

After the operating system is set up, create a system image (using software such as ZENworks Configuration Management) on a USB drive. Use this image to restore the machine to a clean state.

NOTE: It is not necessary to set up a build lab using physical machines. A build lab can be set up by using a collection of virtual machine images. For this approach, Novell recommends creating virtual machine images for each of the six machine configurations. The virtual machines should be configured by following the steps in this section, excluding the creation of a new partition. It is not necessary to create a partition to save the clean image because you can save the current state of a virtual machine.

2.2 Build Process

The build process is the process of converting a Windows desktop application into a standalone executable (EXE) or Spoon Virtual Machine (SVM).

- Use the snapshot technique to capture the application on a clean Windows XP 32-bit machine. The snapshot process is covered in detail in “Snapshots” in the *ZENworks Application Virtualization User Guide* (<http://www.novell.com/documentation/zav10/>).
- Use ZENworks Application Virtualization Studio to generate a standalone executable.
- Test this executable according to the recommendations in “Section 2.3, “Build Testing Process,” on page 14”.
- If the virtual application fails to run on Windows Vista or Windows 7, it might be necessary to perform the snapshot on multiple platforms, and then perform a platform merge. For details on how to perform a platform merge, see “Platform Merge” in the *ZENworks Application Virtualization User Guide* (<http://www.novell.com/documentation/zav10/>).
- After the application has been verified to work on all six platforms, use ZENworks Application Virtualization Studio to generate a component (SVM file) if necessary.

If the virtual application fails to build or function correctly, see “Chapter 4, “Troubleshooting,” on page 19” or contact Novell Technical Support.

2.3 Build Testing Process

The build testing process ensures that the build process successfully captures all aspects of the target application necessary for the virtual application to run correctly.

Novell recommends adhering to the following strategies while testing virtual applications:

- Conduct testing on each of the six machine platforms identified in “Section 2.1, “Setting Up a Build Lab,” on page 13”.
- Under ideal circumstances, each machine in your build lab should be cleaned prior to testing the virtual application.
- It is recommended not to test virtual applications on a machine where the application is also natively installed. However, in some instances, testing the virtual application with older or newer versions of the similar application natively installed is good.

2.4 Profiling the Application

If the virtual application is to be delivered via the Web as a streaming application, you can choose to profile the virtual application in order to dramatically reduce the buffering time needed to launch the application. Novell recommends profiling any application greater than 10 MB in size. For more information on configuring applications for streaming, see “[Streaming Virtual Applications](#)” in the *Novell ZENworks Application Virtualization 10.0 Integration and Streaming Guide*.

During the profiling process, you use the application as a typical user would, while Studio observes this behavior and stores it in a transcript. These transcripts are then subsequently used to construct a streaming model of the application. There is no limit to the number of transcripts that can be used to construct a streaming model. Novell recommends gathering at least one transcript from each of the six machine platforms identified in the “[Setting Up a Build Lab](#)” section. The profiling process is covered in detail in “Create Application Streaming Models” in the *ZENworks Application Virtualization User Guide* (<http://www.novell.com/documentation/zav10/>).

2.5 Managing Builds

Novell recommends allocating a folder on a shared network drive to store and organize the SVMs, EXEs, and models generated as a result of the virtualization process.

During the build process, Novell recommends storing the snapshot files, the EXEs, and the SVMs in the following folder structure:

Snapshot - \\<Share>\Builds\<Application>\<Version>

During the profiling process, Novell recommends storing the transcripts and model files in the following folder structure:

- ♦ **Transcripts:** *\\<Share>\Builds\<Application>\<Version>\Transcripts*
- ♦ **Models:** \\<Share>\Builds\<Application>\<Version>\XStream\<Model Number>

3 Virtualizing Microsoft Internet Explorer

Because of the complexity of packaging Internet Explorer, Novell provides several versions as part of the Template Wizard. If using the Template Wizard is not an option, there are other methods you can use to package Internet Explorer:

- ♦ Use Microsoft Internet Explorer snapshot compatibility mode under the *Options* menu when snapshotting. This ensures a complete capture of the Microsoft Internet Explorer portions of the registry and file system.
- ♦ Internet Explorer 9 should be virtualized from Windows Vista x32 to support the largest number of Windows platforms.
- ♦ Internet Explorer 9 is not supported on Windows XP.

4 Troubleshooting

This section describes the most common configuration errors that occur when using ZENworks Application Virtualization Studio.

If you encounter a problem with a virtual application, read this section carefully or query the online [Novell Support Knowledgebase \(http://www.novell.com/support/\)](http://www.novell.com/support/) before using other support options. It is very likely that the issue you have encountered is addressed in one of these places.

- ♦ [Section 4.1, “Accessing Internet Resources,” on page 19](#)
- ♦ [Section 4.2, “Generating Diagnostic Mode Virtual Applications,” on page 20](#)
- ♦ [Section 4.3, “Roaming User Profiles,” on page 20](#)
- ♦ [Section 4.4, “FAQs,” on page 20](#)

4.1 Accessing Internet Resources

Several ZENworks Application Virtualization Studio features require access to Internet resources. These features might be unavailable if ZENworks Application Virtualization Studio is unable to connect to the Internet.

If the machine on which you are creating a virtual application is itself behind a firewall, then to configure the proxy settings in the ZENworks Application Virtualization application:

- 1 In the ZENworks Application Virtualization application window, select *Options*.
- 2 Click *Proxy Settings*.
This displays the Proxy Settings dialog box.
- 3 Specify the appropriate proxy server settings in the dialog box.

In many corporate environments, access to the Internet is filtered through a firewall or proxy server. In these cases, ZENworks Application Virtualization Studio attempts to automatically configure for Internet access. It might be necessary to manually configure the proxy server settings.

- 1 Open a Web browser, from the Tools menu navigate to the Proxy Settings dialog box.
For Internet Explorer, select *Internet Options > Connections > LAN Settings*.
For Mozilla Firefox, select *Options > Advanced > Network > Settings*.
This displays the Proxy Settings dialog box.
- 2 Specify the appropriate proxy server settings in the dialog box. Consult your system administrator to obtain your proxy server settings.

Users not on a corporate network might need to ensure that Windows Firewall is configured. To allow an exception for ZENworks Application Virtualization Studio, so that the program can access Novell's Library Server.

To add an exception to Windows Firewall (Windows Vista and Windows 7):

- 1 Open the Control Panel, then select *Windows Firewall*.
- 2 Click the *Allow a program or feature through Windows Firewall* link.
- 3 Find the Spoon Studio and ensure that the proper check boxes are selected.
- 4 Click *OK* and close the window.

NOTE: If users are running third-party antivirus or security software, exceptions might also need to be added for those programs. The exception process varies for each program.

4.2 Generating Diagnostic Mode Virtual Applications

Sometimes errors from virtual application configuration result in an executable that fails to run properly. These typically happen when there is an error in the virtualization configuration, such as a missing file or registry entry.

ZENworks Application Virtualization Studio enables you to create diagnostic-mode executables. Diagnostic-mode executables generate logging data that can assist in virtualization problem diagnosis.

To generate a diagnostic-mode executable:

- 1 In the *Output* section of the virtual application, select *Generate Diagnostic Mode Executable*.
- 2 Select *Build* to generate the executable.

This generates an `xclog_<id>.txt` file in the application startup directory. The file contains detailed diagnostic data gathered during execution. Use a tool such as `DebugView` to examine the file for entries labeled `WARNING` or `ERROR`, that might help you diagnose virtualization errors. If you require assistance from Novell Technical Support to resolve issues, submit this information along with your support request to facilitate a faster resolution.

NOTE: Because diagnostic-mode executables run significantly slower than standard executables, and generate very large log files, diagnostic-mode executables should not be distributed to your end users except for diagnosing an issue.

4.3 Roaming User Profiles

If you are using the default sandbox location to create multiple virtualized applications or to create a large number of virtualized applications and if you have Roaming User Profiles enabled, you might have performance issues on logging in or logging out.

The default location for the sandbox is:

@APPDATALOCAL@\\Novell\\Sandbox\\@TITLE@\\@VERSION@

4.4 FAQs

- ♦ [“Why do I get a “File not found” error when I try to load my configuration?” on page 21](#)
- ♦ [“Why does my application not start after building?” on page 21](#)
- ♦ [“I receive an “Unable to contact server” error when I try to build my application” on page 21](#)

- ♦ “Why do I receive a connection error when I start my application?” on page 21
- ♦ “Why is my application unable to find a child process?” on page 22
- ♦ “I have Excel 2007 and Excel 2010. I tried adding both my Microsoft Office 2007 and Microsoft Office 2010 applications to my desktop in the same directory and they do not work. Why is that?” on page 22
- ♦ “I virtualized Outlook 2010 64-bit and added it to my desktop for use with Novell GroupWise, and now Microsoft Outlook 2010 does not start” on page 22

Why do I get a “File not found” error when I try to load my configuration?

Explanation: This error appears when ZENworks Application Virtualization Studio cannot find one of the files referenced in your configuration. If your configuration file (XAPPL) has moved, ensure that the Files folder is in the same relative location to the configuration as it was when it was created.

Why does my application not start after building?

Possible Cause: There are several possible reasons why an application might fail to start. Use the following list to check for problems and correct them:

- ♦ Verify that you have the correct startup file selected.
- ♦ If your application does not open on one operating system, verify that the application failure is the same on all available operating systems.

Action: To resolve the problem:

- ♦ Create a diagnostic version of the application and look in the logs for specific error messages. This can often provide a quick fix by letting you know exactly where the issue is.
- ♦ If the virtual application does not run, but the native version does run, try building from the *Capture and Diff* snapshot without any changes. Verify if the application is still not starting. If not, it is possible that the application is not compatible and cannot be virtualized. If the application does launch, start removing the registry and file system items again and see if it still executes. Perhaps something necessary was unintentionally removed.

I receive an “Unable to contact server” error when I try to build my application

Action: If you are including a runtime in your build, ensure that you can access Internet resources, so ZENworks Application Virtualization Studio can download the needed runtime information in order to complete the build.

Why do I receive a connection error when I start my application?

Explanation: If the hard drive is compressed and the SQL 2005 Express Runtime is included in the virtual application, a known issue in SQL 2005 causes a connection error to display at startup.

Action: The only current workaround is to decompress the hard drive.

Why is my application unable to find a child process?

Action: If your application cannot find child processes, check to see if *Enable startup executable optimization* is enabled. This option is found on the *Settings* menu under *Process Configuration*. *Enable startup executable optimization* is disabled by default. This means that there is a bootstrap process between your application and the host services. For example, you might have a code statement that looks like this:

```
ProcessServiceProcess = _GetProcessByNameAndParent ("sqlservr",  
Process.GetCurrentProcess ());
```

This statement does not return any results because of the bootstrap process.

Editing the statement as follows allows you to look first at the parent process of the bootstrapped application, and then find the child process you want to modify.

```
ProcessServiceProcess = _GetProcessByNameAndParent ("sqlservr",  
GetParentProcessOfCurrentProcess ());
```

I have Excel 2007 and Excel 2010. I tried adding both my Microsoft Office 2007 and Microsoft Office 2010 applications to my desktop in the same directory and they do not work. Why is that?

Explanation: Microsoft Office 2007 and 2010 have some shared registry keys. Because of this, the applications cannot be run in the same sandbox. It is the same as if you had both versions installed locally. In order to have both versions, you need different applications and need to ensure that the sandbox path for each is unique. This applies to all Microsoft Office applications.

I virtualized Outlook 2010 64-bit and added it to my desktop for use with Novell GroupWise, and now Microsoft Outlook 2010 does not start

Explanation: A known issue in GroupWise causes it to not work with 64-bit versions of Outlook 2010. The error also occurs when both applications are installed locally.