

Treasure Island

A Deep Dive into ZENworks Application Virtualization 7.1

This article first appeared in the October 2009 issue of *Novell Connection* magazine.

Dealing with software conflicts, incompatibility between software and OS versions, and testing applications in new environments can create a significant drain on IT resources as well as cut into user productivity. One of the major causes of application failure is from application conflicts created by DLL files getting overwritten and registry settings being changed as a result of new software installations. By isolating the application from its underlying operating system through application virtualization, these problems can be drastically reduced and, in many cases, completely eliminated. In fact, industry analysts indicate that by taking advantage of virtual applications you can realize a 60 percent reduction in costs associated with testing, packaging and supporting an application.

As a mature, full-featured application virtualization solution, Novell ZENworks Application Virtualization 7.1 enables you to quickly and easily encapsulate into a single executable package all the components that an application needs to run. (See [Packaging Best Practices](#).)

It does this by separating these components from the underlying Windows operating system. Since the resulting executable is self-contained and does not change the host operating environment, the virtual application runs consistently on every system. Not only does this significantly reduce application testing times, but it also eliminates common software problems such as DLL conflicts. In fact, it even lets you run some types of applications on platforms where they typically would not be able to run, such as Internet Explorer 6 on Windows Vista or Windows 7. (See [Key Features](#).)

As executable files, virtual applications do away with the need to install and configure applications separately on individual systems. They also simplify the process of rolling back, removing or updating the application. The virtual applications you create using ZENworks Application Virtualization can also be run securely on a wide variety of media, including USB devices, the network or a local drive.

In addition to its fast and easy methods for creating virtual applications, ZENworks Application Virtualization also has advanced configuration features that provide you a higher level of customization.

> Virtual Apps Made Easy

ZENworks Application Virtualization does not require the deployment of any client-side or agent-side software. Instead, it uses an easy-to-use packaging utility that lets you quickly repackage and customize your application into a standard Windows executable that you can deploy using any software delivery mechanism you have.

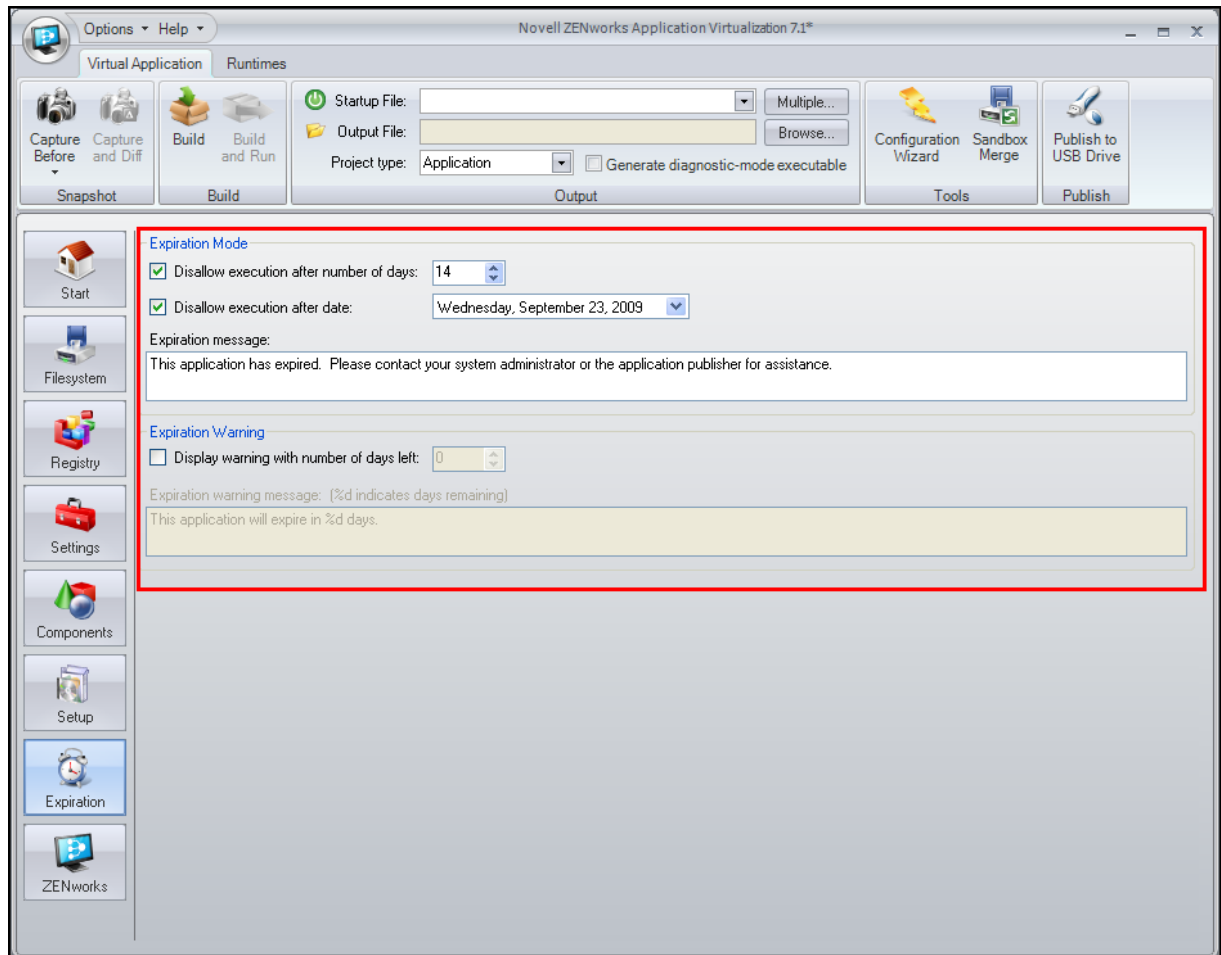
The packaging utility enables you to build applications in a variety of ways. The first method is by using the auto-configuration wizard for popular applications. The wizards guide you through a step-by-step process to build and customize applications such as Internet Explorer, Firefox and GroupWise, or common components including the .NET Framework, Java and Adobe Flash Player. Using this method you can virtualize a popular application in as little as five minutes.

For applications not covered by the auto-configuration wizard, ZENworks Application Virtualization provides a snapshot method that captures a system's state before and after an application is installed. It then automatically configures the virtual application settings based on the system changes it detects. As a third method, if you use Novell ZENworks to package and distribute applications, you can also use ZENworks Application Virtualization to quickly convert your legacy ZENworks AXT-based applications into virtual applications.

> Advanced Configuration

In addition to its fast and easy methods for creating virtual applications, ZENworks Application Virtualization also has advanced configuration features that provide you a higher level of customization. For example, you can restrict how long a virtual application can be used. This application expiration feature is useful if you need to provide applications on a short-term basis, such as to contractors, temporary employees or students. Configuring application expiration is as simple as clicking the Expiration tab in the build interface and setting the number of days or the date when the application will no longer be allowed to execute. (See Figure 1.) The

Figure 1: *When distributing applications on a temporary basis, you can set an expiration date on a virtual application to restrict how long it can be used.*



expiration feature also gives you the ability to warn the user that the application is about to expire or has already expired.

Another advanced configuration feature is the ability to impose security restrictions on your virtual applications. Since virtual applications simplify deployment by allowing you to easily copy the executable file from one machine to another, they also make it easy for unauthorized personnel to access the software. You can prevent unauthorized use by leveraging the integration between ZENworks Application Virtualization and Novell ZENworks Configuration Management.

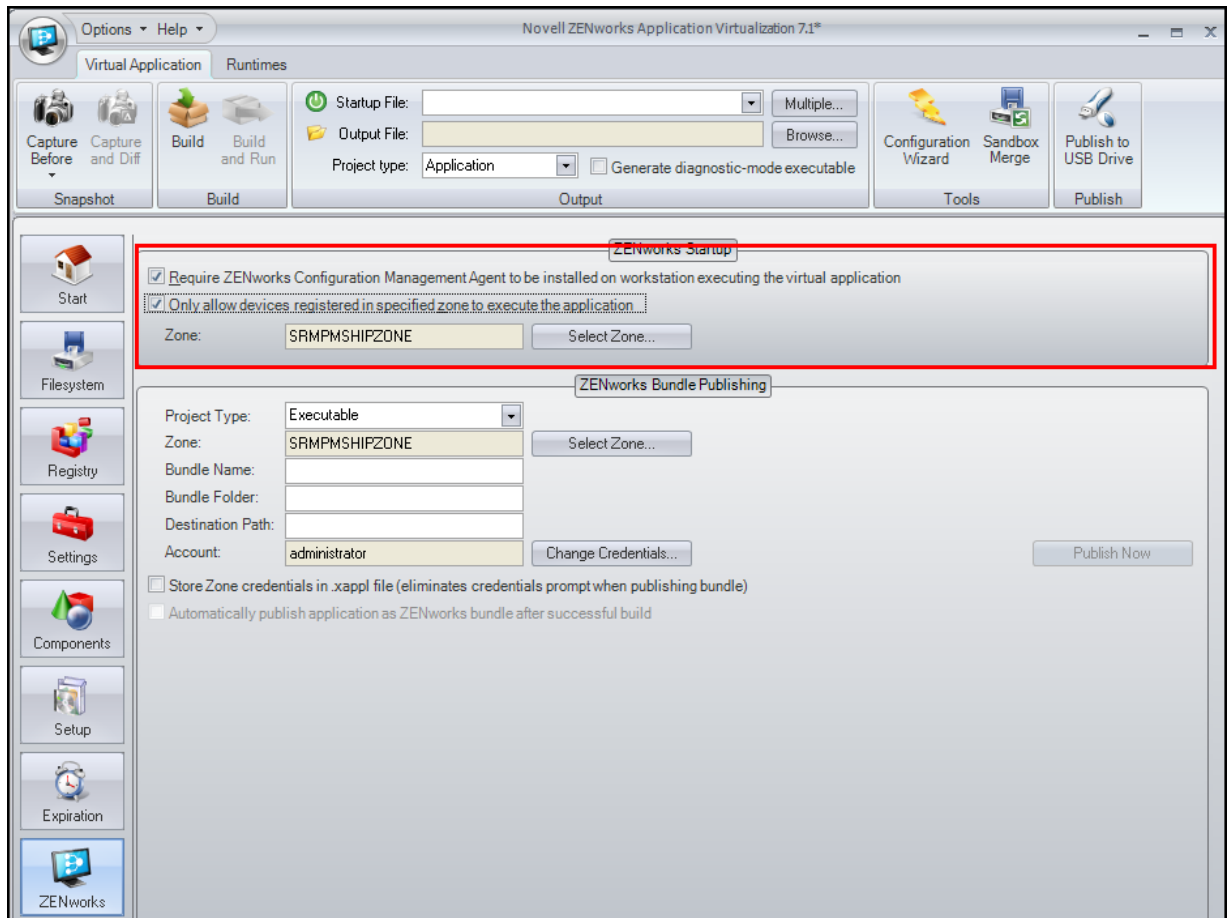
With this feature, you can prevent your virtual applications from running on machines that don't have the ZENworks Configuration Management agent. You can also further restrict operation to workstations registered to a given ZENworks Configuration Management zone. Implementing this feature ensures the application will only run on devices you manage. (See Figure 2.) Additionally, since ZENworks Configuration Management provides

asset inventory and asset management, you can successfully track the dissemination of the virtual application among the devices in your environment.

One of the most powerful advanced features that ZENworks Application Virtualization provides administrators is the ability to customize how the virtual environment will interact with the host environment through the isolation of a given directory or registry key. This allows you to control what application related files and registry entries can be modified on the host machine versus the virtual environment. The solution provides three modes of control for this isolation feature: Merge, Full and Write Copy. (See Figure 3.)

The *Merge* mode allows the virtual application to read and write files to the selected directory. This is the typical behavior of common folders such as *My Documents*. By default, this ensures that documents saved to *My Documents* end up in the user's actual *My Documents* folder.

Figure 2: *By leveraging the integration with Novell ZENworks Configuration Management you can configure a virtual application to only be allowed to run on certain workstations*



Key Features

ZENworks Application Virtualization provides the following key features:

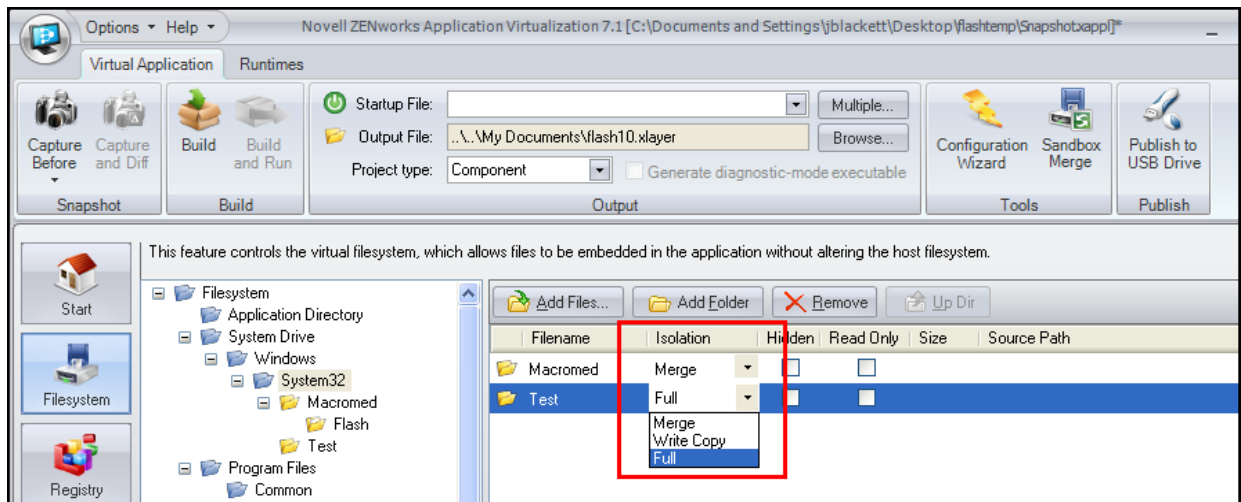
- **Creation of a single executable file** – Packages all application files, settings, runtimes and other components into a single compact executable that runs anywhere.
- **Eliminates problems with Windows Vista/7** – Since it eliminates the need to access privileged system resources, it reduces Windows Vista/7 UAC prompts and eliminates most compatibility errors.
- **Integrates with ZENworks Configuration Management** – With its ability to leverage ZENworks Configuration Management, ZENworks Application Virtualization lets you quickly and easily secure virtual applications and publish them to ZENworks Configuration

Management zones as a Windows Bundle that can then be deployed in the traditional ZENworks way.

- **Easy application registration** – Employs an easy-to-use tool to register the virtual application with the host OS to ensure file associations and shortcuts perform properly.
- **Intuitive user interface** – Simplifies the creation and customization of virtual applications, including a one-click component configuration that quickly adds popular runtimes, components and viewers to virtual application executables.

The *Full* mode only allows the application to read or write files to the virtual file system. If you set *My Documents* to *Full*, then the user would only see the contents of the *My Documents* folder in the virtual application's sandbox. The sandbox is an isolated area in the virtual environment for storing user-defined settings and files. Additionally, the application would write all of the user's saved files to the sandbox. The *Full* mode is especially useful if you want

Figure 3: ZENworks Application Virtualization lets you control how the virtual environment will interact with its host environment.



the virtual application to run on a USB thumb drive and you want to make sure that the user's documents are only saved on the thumb drive.

The *Write Copy* mode allows the application to read from the host, but it always writes any changes to the sandbox. This mode is useful when a file or files are needed from the host machine, but you want to make sure the virtual application cannot impact the host.

The *Merge* and *Full* options can also be used for isolating registry settings. In other words, you can control whether the application has access to registry keys not included in the sandbox.

Whether you use its basic or advanced configuration features, ZENworks Application Virtualization gives you the ability to create custom and secure virtual applications that eliminate the traditional overhead of managing Windows applications.

> The XLayer

In addition to letting you create virtual applications, ZENworks Application Virtualization also allows you to build a virtual set of self-contained components called an XLayer. An XLayer would be comprised of various files, registry entries or other components that a certain application might require, such as your browser needing a certain version of Java. When you create an XLayer it

can be dynamically used by a virtual application or can be embedded into an application. Building an XLayer file simply requires you to change the *Project Type* in the solution's interface to *Component* instead of *Application*.

A typical usage for an XLayer would be to create a reusable set of self-contained virtual components that can be leveraged by multiple virtual applications. For example, the prebuilt run times available in ZENworks Application Virtualization (i.e., .NET Framework, Flash Player and the Java Runtime Environment) can be injected into a virtual application at build time. To do this you simply click the button in the toolbar that represents the component you want to inject and it will embed the component when it builds the virtual application. By embedding these required components into the virtual application you only have to distribute a single executable file, rather than all the other applications or files it depends on for proper execution.

Another common use case for XLayers is patching your virtual applications. Since virtual applications run in an isolated environment, they generally can't be patched using traditional self-updating methods. Rather than re-creating an entirely new patched version of the virtual application, you can create an XLayer that contains the patched information. This file can then be distributed to your user machines and placed in the same directory as the parent virtual application. By doing this, the next time the application launches, the application will dynamically load the contents of the XLayer file, patching the application. To create an XLayer patch file, you install the baseline application on your build machine and then use the snapshot mode to capture the patch information as you apply the patch to the application.

XLayer files can also be used for common application plug-ins, such as Microsoft Silverlight or Adobe Reader for your Web browser. Without the ability to use XLayers

Packaging Best Practices

To ensure the virtual applications you build with ZENworks Application Virtualization run properly in their isolated virtual environments on any Windows machine, it is important that you adhere to the following best practices.

- **Use a clean machine when packaging** – When packaging applications with the ZENworks Application Packager you should ensure that only the operating system and baseline patches for all devices in your environment are present. If you use a machine with other software, your virtual application may not contain all of the files and registry settings it needs to run.
- **For applications that exist as pre-builds, start with the pre-build** – When using applications that have been prepackaged, such as Internet Explorer, it is recommended that you use the prepackaged application as a starting point. You can then perform additional customization as needed.
- **Set the sandbox path according to your needs** – Depending on the delivery and means the user will use to run the application, make sure you select the proper sandbox location. For instance, if you are building an application that runs on a USB stick, make sure the sandbox is on the USB stick as well. This ensures that any data the user saves is written to the stick and not to the local machine.
- **Clean up the application** – As with most software packages, depending on what background operations occur, the application packaging process may identify registry settings and file data that are not application related. This can cause these unassociated settings and files to be included in the virtual application. To

prevent this, prior to building the application, review the files and registry information and clean up items that may not be related.

- **Save a copy of the XAPPL** – The XAPPL is the instruction file that is used to build the application. If you save the XAPPL and files that accompany it, you can always make more changes over time.
- **Consider using whole machine virtualization for the packaging machine** – Using whole machine virtualization such as VMware, Virtual Box or Virtual PC allows you to have a base environment for packaging that you can rapidly revert to when you need to create new applications.
- **Don't try to package applications that install drivers or other system level components** – ZENworks Application Virtualization can only package applications that run in user mode.

for your plug-in components, you would have to create different versions of the same virtual application, each containing the different plug-ins your different users need. Instead, by building your plug-ins as virtual components, you can create a single baseline version of the virtual application and place its associated plug-in XLayer files as needed on user machines in the directory where the application resides. This allows the plug-ins to be automatically loaded the next time the application launches.

Whether you use its basic or advanced configuration features, ZENworks Application Virtualization gives you the ability to create custom and secure virtual applications that eliminate the traditional overhead of managing Windows applications. ZENworks Application Virtualization also enables you to simplify and accelerate application rollouts, while reducing downtime and helpdesk calls, and providing you greater application portability, flexibility and interoperability. **N**