

Inside Windows Installer (.MSI) Migration

Take full advantage of Windows 2000 and XP by migrating applications to Windows Installer format

How Does Windows 2000 and XP Migration Affect Deployment Teams?

Migration to Windows® 2000 and XP has become an important issue for system administrators. Chances are your organization is embarking on this migration, or planning one in the future. Many reasons exist to undertake this enterprise-wide project, including increased reliability and security, simplified support, and reduced installation, operating and help desk costs. Migrating to this platform, according to Microsoft®, reduces the cost of IT support services 29.9 percent and the cost of desktop management services 24.4 percent.

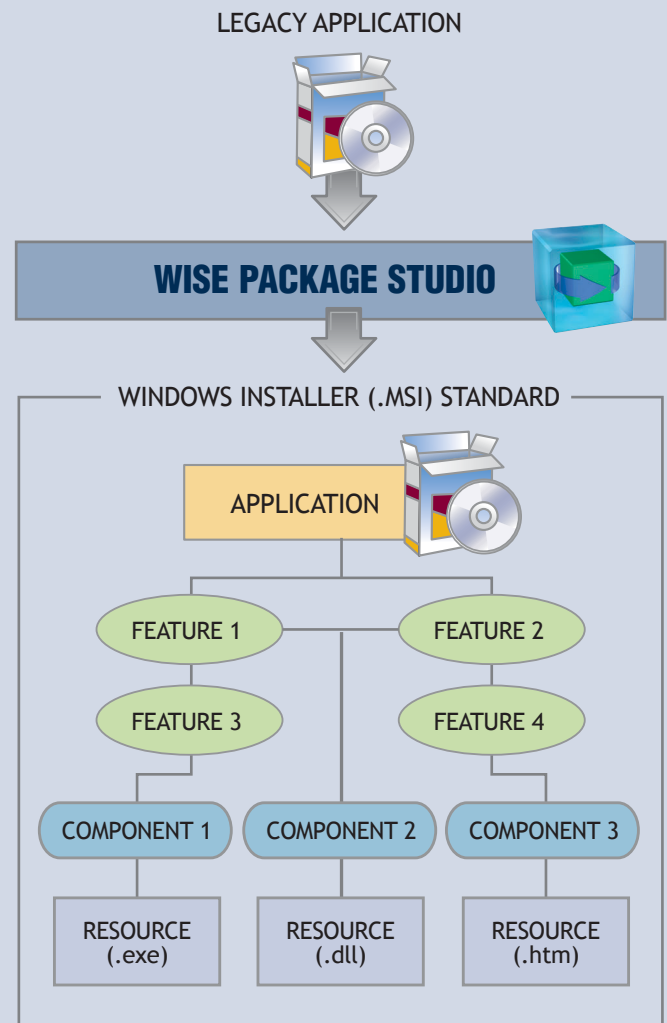
Much of the benefit of Windows 2000/XP migration is gained by also migrating applications to the Windows Installer (.MSI) standard. Windows Installer is an engine built into Windows 2000/XP for managing the state of an application. It is a key enabler of remote installation, group policies, IntelliMirror support and on-demand application installation, all of which contribute to the cost and resource savings associated with Windows 2000/XP migration.

Though there is much to gain, many organizations undertaking Windows 2000/XP migration find the premise of .MSI migration intimidating. Most applications in the enterprise environment are not currently packaged in .MSI format. Even independent software vendors (ISVs) developing new applications have been slow to adopt this format. As a result, most applications need to be repackaged into .MSI format to realize the benefits offered by Windows Installer. And, applications already in .MSI format need to be customized for integration into the enterprise environment.

Migrating to Windows 2000/XP, according to Microsoft®, reduces the cost of IT support services 29.9 percent and the cost of desktop management services 24.4 percent.

Most current repackaging tools do not give full support to this migration, despite the limited resources and urgent needs of system administrators trying to accomplish this task. The technology is complex and most system administrators are still learning about it.

That's where Wise Package Studio® from Wise Solutions can help. Wise Package Studio is the only solution that provides everything you need for .MSI migration. For years Wise has worked closely with Microsoft on installation technology developments. Our position as the repackaging experts allows us to share our considerable knowledge with Wise Package Studio users, and incorporate this knowledge and experience into Wise Package Studio.



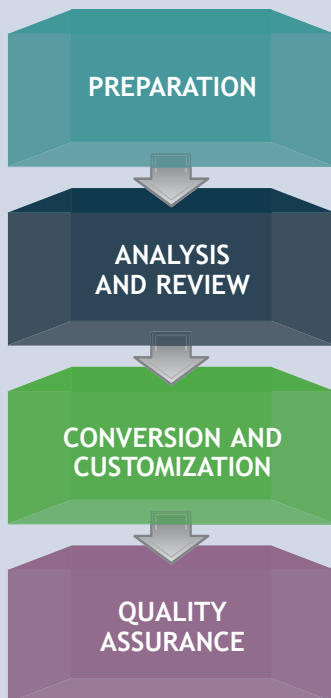
Wise Package Studio is the only comprehensive solution for .MSI migration.

Inside Windows Installer (.MSI) Migration

.MSI Migration Process

The best way to handle .MSI migration is to break it down into specific phases. Wise Package Studio helps you accomplish each phase of .MSI migration. By understanding each phase, the steps within the phase, and the related challenges, your migration project will be more efficient and cost-effective. In this white paper, we'll discuss each phase of .MSI migration and what to "watch out for" in each phase. The phases include:

- Preparation
- Analysis and Review
- Conversion and Customization
- Quality Assurance



Wise Package Studio helps you accomplish each phase of .MSI migration.

PREPARATION

Preparation is the first phase of .MSI migration. During this phase you define the objectives and standards of the migration project, which help ensure efficiency and consistency as the project progresses. The steps involved in preparation are:

- Identifying applications that will be migrated
- Defining target dates for completing migration
- Defining and documenting corporate standards the repackaged applications will follow, such as user interaction with the installation, destination directories, common templates, etc.
- Choosing a migration and repackaging solution, such as Wise Package Studio
- Assessing training needs of the migration team
- Developing process templates that give the migration team a standardized, common process to follow as applications are migrated and deployed
- Prioritizing the order applications will be migrated
- Assigning team resources to the migration project

Preparation Checklist

Do you have the right capturing resources?

One of the hidden costs in the preparation phase is lining up the right resources for repackaging. Traditionally, capturing each application installation and repackaging it into .MSI format requires at least one clean machine dedicated to capturing installations, and the machine must be re-imaged between each capture.

How does Wise Package Studio help?

You can make repeated captures on a single "non-clean" machine, such as your working PC, without re-imaging between captures with Virtual Capture™. This helps you reduce the resources and time required to make installation captures.

ANALYSIS AND REVIEW

Once the .MSI migration project parameters, goals and resources are identified, the next step is assessing the type of work to be done on each application and how to accomplish the work required. This includes:

- Identifying who will receive repackaged installations
- Determining the operating system(s) the package will be installed on
- Defining installation requirements
- Analyzing the vendor package
- Determining appropriate options for modifying the installation
 - Converting legacy installations where applicable
 - Repackaging non-.MSI installations
 - Creating transforms (.MSTs) for packages already in .MSI format
(.MSIs should not be directly modified; they should be customized through transforms)
- Performing operating system snapshot for Virtual Capture

Analysis and Review Checklist

Do you have the right source media for the application?

Make sure you have the correct version of the application and all the related components.

Do you understand the requirements of each application installation after it has been repackaged into .MSI format?

Understanding all of the package requirements is essential. For example, you should know the minimum system requirements (hardware and software), which components should be installed, and other applications that coexist on destination desktops.

How does Wise Package Studio help?

The Wise Software Repository™ and the Application Gateway help you store, manage and share all of your application resources, such as files and components, so you can easily know at a glance the location of resources. SetupCapture also helps you analyze the requirements of each package as you capture the installation.

PACKAGE CREATION AND CUSTOMIZATION

Package creation is at the heart of .MSI migration. During this phase, you take the legacy applications, convert or repackage them into Windows Installer format and customize the repackaged installations to meet your organizational standards. Package creation and customization includes:

- Converting the legacy installation into .MSI format, or repackaging the application using snapshots or application monitoring
- Customizing the application to meet organizational standards by editing the repackaged installation, or creating a transform

Conversion Checklist

Are you using exclusion lists?

Defining a list of resources that should be automatically excluded by the installation capture process is essential to creating a reliable package.

How do you capture changes to the hardware registry?

In most cases, you probably do not want to capture registry changes involving hardware settings. However, you may want to consider capturing registry changes involving hardware settings in situations where a package will only be deployed to machines that have identical hardware configurations.

How does Wise Package Studio help?

SetupCapture offers many options for making the most accurate captures possible, including building global exclusion lists, capturing hardware registry changes and more.

Once you've converted or repackaged your applications into Windows Installer format, you should customize them to meet corporate standards so they integrate seamlessly into your environment.

Customization Checklist

Can your packages be easily updated?

In order to create a patch or update that can be properly applied, you should ensure the proper values are used for the ProductCode, PackageCode, and ProductVersion properties. It is also essential to ensure the components in the newer version of the package are properly aligned with those in the older version of the package.

Are you properly creating and applying transforms (.MST)?

In cases involving installations that are received from the vendor in Windows Installer format, you should create a transform (.MST) file to customize the application. You should not directly edit the vendor installation. Direct editing of the vendor installation may be in violation of the vendor's license agreement.

How does Wise Package Studio help?

Wise Package Studio offers the most options for editing .MSI installations. You can choose from a step-by-step approach, direct editing of Windows Installer tables, or edit .MSI installations in script format. You can also easily create and apply transforms with InstallTailor.

QUALITY ASSURANCE

Once you've migrated your packages to .MSI format and customized them, you should undertake Quality Assurance activities to ensure the application will install and work properly the first time. This includes validating the package, resolving conflicts between the application and other applications, and performing testing in your environment. There are several issues to consider as you perform Quality Assurance on repackaged applications:

Quality Assurance Checklist

As you validate applications, should you try to meet Windows 2000 Logo requirements?

Prior to deployment, you should validate how well the package conforms to Microsoft's Windows 2000 Logo requirements. These requirements contain several criteria related to internal consistency, component design, file duplication, advertising support, uninstall support, etc. While many of these criteria identify important considerations that should be addressed prior to deploying a package, some of the other criteria should be regarded as more informational. Very few commercially available applications fully comply with the Windows 2000 Logo requirements.

Most current repackaging tools do not give full support to this migration, despite the limited resources and urgent needs of system administrators trying to accomplish this task. The technology is complex and most system administrators are still learning about it.

Wise Package Studio is the only solution that provides everything you need for .MSI migration.

Inside Windows Installer (.MSI) Migration

What else should be validated in the package?

In addition to validating the package's compliance with recognized industry standards, you should also validate that your package complies with your own organization's standards. For example, your organization may have a standard that packages should not contain any hard-coded paths.

Does your application isolation work properly?

Although application isolation offers great potential for eliminating some file conflicts, it is not optimal for every situation. Application isolation will not work in situations in which the application code uses hard-coded paths to call .DLL and .OCX files. After isolating .DLL or .OCX files during the repackaging process, it is important to verify that the repackaged application is actually calling such files from the application directory, rather than a shared directory.

When checking for conflicts, are you looking for conflicts involving device drivers and group policy objects?

As part of the conflict management process, you should identify and resolve conflicts between applications that are likely to be deployed to the same desktop, and conflicts between applications and the operating systems on which they will be deployed. In addition, you identify and resolve conflicts related to the configuration information associated with device drivers and group policy objects.

Are you testing the package in your own environment?

Software developers spend a lot of time verifying that an application works properly. However, such testing usually verifies that an application functions properly when run in isolation (i.e. on a clean machine). As part of the application integration process, it is essential for you to validate that each application will properly function in your environment. In other words, it is important to verify that the application can peacefully co-exist with other applications that are likely to be deployed to the same desktops.

Can the package be installed with less than administrative rights?

In cases where a repackaged application will be used by users with less than administrative rights, it is important to install and run the package using an ID that has such rights, to ensure that users have access to all of the resources required to exercise the full functionality of the application.

Are you encountering errors by manually passing packages to distribution systems?

The process of manually passing a repackaged application to your software distribution system can be a time-consuming endeavor that is prone to error. You can significantly reduce the amount of time and error involved by using an automated process to seamlessly pass packages to your software distribution system.

How does Wise Package Studio help?

All of your Quality Assurance activities can be accomplished with Wise Package Studio Quality Assurance. Preflight Deployment helps you test the installation in the production environment. Test Expert generates a master test plan with multiple tests cases and guides you through the steps of testing in your organization's environment.

.MSI Migration Success

Following these phases of .MSI migration, taking the tips into consideration, and choosing a comprehensive migration solution such as Wise Package Studio will help you implement an efficient, smooth migration. With a successful .MSI migration, your organization can realize the full benefit of Windows 2000/XP migration and deploy more reliable applications.



Global Headquarters

47911 Halyard Drive • Plymouth, MI 48170 USA
Phone: 734 456 2100 • 800 554 8565
E-mail: info@wise.com
www.wise.com

European Headquarters

President Kennedylaan 19
2517 JK The Hague • The Netherlands
Phone: +31 (0)70 392 72 20
E-mail: europe@wise.com

