Peregrine

Desktop Inventory

Application Encyclopedia User's Guide

For use with Desktop Inventory 8.0

Peregrine
SYSTEMS®

Contents

About This Guide

This User's Guide serves as an introduction to the Desktop Inventory Application Encyclopedia (ApE) and as such is recommended reading for anyone working with the product.

It contains details of the application, a description of the features and details of how you would use the application.

Knowledge Requirements

As an Application Encyclopedia user, you need a basic knowledge of the environment in which you are working (Windows 95/98/2000/NT/XP or UNIX).

Who Uses Application Encyclopedia

Typically, you will have someone in your IT department who is responsible for managing in-house applications and keeping your application library up to date.

Organization of This Guide

This guide is organized around the main functions associated with the Application Encyclopedia. The following table shows you which part of the manual you need to reference to find the information you require.

To find this...	Look here...
Application Encyclopedia overview; get to know Application Encyclopedia's components.	*Introduction*
Application Encyclopedia's relationship with the Desktop Inventory products.	*Introduction*
How to update the data in your Application Encyclopedia database and how to plan this project	*ApE Database Maintenance*
How to use the Application Encyclopedia Explorer menus, toolbars and buttons.	*ApE Explorer*
Logging on to and logging out from the Application Encyclopedia Explorer.	*ApE Explorer*
How to view information in the Application Encyclopedia Explorer	*ApE Explorer*
How to use the MSI Importer to teach the contents of a set of MSI files.	*The MSI Importer*
How to view the structure and contents of the MSI files using the MSI browser	*The MSI Importer*
How to use MSI Scanner command line utility to create an XML file containing a summary of the MSI	*The MSI Importer*

Contacting Peregrine Systems

For technical support on this or any other product from Peregrine Systems, Inc., refer to the Customer Support Web site at:

http://support.peregrine.com

1 Introduction

CHAPTER

This chapter provides an overview of the Desktop Inventory Application Encyclopedia (also referred to as ApE in this document).

You will find information on the following topics:

- *Application Encyclopedia Overview* on page 10
- *The ApE Components* on page 11

Application Encyclopedia Overview

Application Encyclopedia (ApE) is a database-style recognition library which provides the application recognition technology foundation for Peregrine Systems's Desktop Inventory products.

After you have configured a Scanner and scanned the computers in your population, a scan file is produced which contains inventory data about those computers.

The scan file does not initially contain a list of recognized software applications. Instead the Scanner collects information about executable files. This information is then analyzed to determine the applications the files are part of.

This analysis makes use of a Software Application Index (SAI) file (which resides in the following directory by default:

C:\Program Files\Peregrine\Desktop Inventory\8.0.0\Common

A SAI file contains encoded knowledge of what software applications can be recognized. Once applications have been recognized, the scan file is rewritten, or 'enriched', to include the application list.

There are three ways to update the SAI file with knowledge of new applications.

- You can download a new SAI file from the Peregrine support web site. This file is updated on a quarterly basis to reflect the on-going work of our software librarians.
- You can export (generate) a SAI file with knowledge of your own applications using the Application Encyclopedia (ApE) Explorer. The ApE Explorer is also used to maintain your database.
- A customized SAI file (user.sai) can be created using Analysis Workbench or the SAI Editor. This file contains applications that you have taught to it.

Further Information

You can find further information about SAI files and how they are used for recognition in the Desktop Inventory User's Guide.

The ApE Components

The Application Encyclopedia recognition engine consists of three components:

- **Application Encyclopedia Database**

  This database contains information about software applications. You can export this information to a SAI file used when application recognition is required for Analysis Workbench and Viewer.

- **Application Encyclopedia Server**

  The ApE Server is a software component that manages updates to the ApE database. It ensures that user customizations are maintained and integrated with Peregrine data from new SAIs.

- **Application Encyclopedia Explorer**

  The ApE Explorer is a Windows-based tool that allows multiple users to manage, view and update the information in the ApE database. It is possible to get several views of the data in the database, add new applications to the recognition library, export the contents of the database to the SAI format, import SAI files, ApE explorer connects to an ApE Server via IP sockets.

  The MSI Importer is an ApE Explorer module that uses the MSI standard to add applications packaged in MSI-based installers directly to the ApE database without having to install then first.

These components work together to create a recognition data maintenance system.

The following diagram shows the communication between the three components of the Application Encyclopedia.
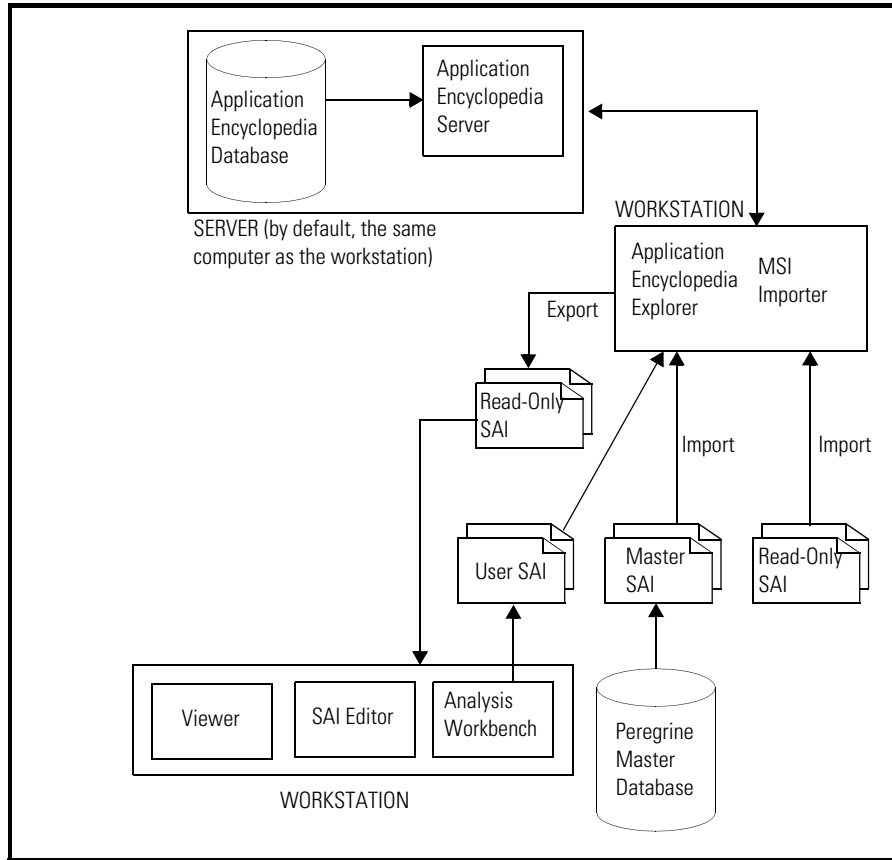
Figure 1-1: The Ape components

2 ApE Database Maintenance

CHAPTER

You will find information on the following topics in this chapter:

Keeping Your ApE Database Up-To-Date

You will want to keep your ApE database up to date, so that it includes in-house applications and any applications that you may have taught to User SAIs in Analysis Workbench.

When you first install Desktop Inventory, the ApE database is unpopulated and you populate it with data from Peregrine's Master database (Master SAI). The ApE database can also be populated with data from National SAIs (currently French and German SAI) if you have national editions of the applications installed across your software population.

Note: If no such applications are installed then we recommend that you import only the Master SAI.

If you are upgrading from a previous version of InfraTools Desktop Discovery, you may have a customized database that you wish to use. You may do this by exporting an SAI from your old installation and importing that into the new database. Once you have your existing data in the new database you may import the latest Master SAI to ensure that the Peregrine portions of your software library are up to date.

Further information
- Further information about the SAIs and how to maintain them is covered in detail in the Desktop Inventory User's Guide.
- The procedures used to manually update the ApE database are covered in *ApE Explorer* on page 25.

Techniques Used to Maintain the ApE Database

The Application Encyclopedia Explorer is used to maintain and update the data in the Application Encyclopedia database by:

- Importing file data from a scan file. This should be the main way in which you update files in the ApE database.
- Importing a User SAI file that may have applications in it that you have manually taught via Analysis Workbench.
- Manually adding publishers, applications, releases, files and release relations.

Importing SAIs Into Your ApE Database

In this section...

Types of SAI Files

Entries can be added to an SAI file. These SAI files can be imported into the Application Encyclopedia database. Three types of SAI files can be imported.

- **SAI updates** (**Master.sai**)

  Peregrine Systems hold a master copy of the database. This Master database is being continually updated and new information is made available in the form of SAI Updates. You can obtain these updates by visiting the Peregrine Customer Support website. These SAI updates can then be imported into the customer Application Encyclopedia database.

- **User SAIs** (**User.sai**)

  User SAIs are custom created using the Analysis Workbench.

- **Read Only SAIs**

  A Read Only SAI is an export from your ApE database. It can be imported back, although doing so will wipe out the existing entries in your database. If you want to have your own entries as well as the new entries from the Master Library when setting up the database, **ReadOnly.sai** must be imported first.

The mechanics of importing SAI files are covered in *Importing Data from an SAI* on page 56 of this guide.

Master Application Libraries and National Libraries

The central Peregrine ApE database is called the Master Database and is distributed to customers as a set of **Master SAI** files:

- Master.sai
- French.sai
- German.sai

It is exported from the Peregrine Master database by Peregrine Librarians and posted as a quarterly update on the Customer Support website.

The **Master.sai** contains English (English US, English UK, international English and others) versions of software and language-independent application versions or neutral versions.

The **French.sai** and **German.sai** files are exports from the Master database. Although **Master.sai** contains only English and neutral applications, the Master database contains all data, that is, including French and German data.

These SAI files can be used to keep your ApE database up to date (quarterly) in addition to adding entries of your own to the ApE database.

The ApE database can then be used to produce a single SAI file to be used for your recognition purposes in applications such as Analysis Workbench, XML Enricher and Viewer.

Note: When importing National SAI files, you must import the Master SAI first.

Manually Updating Your ApE Database (Teaching)

The process of manually updating your database with new entries is also known as teaching entries to the ApE database.

Manual updates to the Application Encyclopedia database are used to update publishers, applications and versions.

Important: This is not the preferred way of maintaining the files in the database, since numerous properties must be manually entered into the database for each file to be described.

ApE database maintenance can be broken down into four stages:

- *Stage 1 - Obtaining the Software* on page 17
- *Stage 2 Application Structure Analysis and Installation of All the Components* on page 17
- *Stage 3 - Adding Obtained Information Into the Library* on page 18
- *Stage 4 - Creating a New SAI File For Recognition Purposes* on page 22

Further Information

The actual teaching procedure is covered in detail in *Teaching New Applications to the ApE Database* on page 62.

Stage 1 - Obtaining the Software

This stage involves deciding which software is to be added to the ApE database and where it can be obtained from. Generally, this will be in-house software that your company has produced or any new commercial applications acquired by your company.

Stage 2 Application Structure Analysis and Installation of All the Components

This stage involves the installation and analysis of the software. However, before the installation of the software takes place a certain amount of preparation work should be done.

Preparation Work

We recommend that you have at least one test machine for the purpose of application installations. For installations of Client/Server applications access to an additional machine should be considered.

The best way to collect all the necessary information about files belonging to an application is to make a 'clean' installation of that application. That is, install the application on a machine which has only an operating system installed.

Since it can be very time consuming to install an operating system on a machine every time the application installation is required, we recommend that you install it once and then use some kind of imaging software. By using this software the images of all base operating systems can be created. When a certain operating system is required, the image can be quickly restored.

Application Analysis

In order to add application data into the database you will have to establish:

- The hardware and software requirements for an application installation and whether they can be met.
- The platforms an application is designed for

- What third party applications must be installed for the application to function correctly (for example, Interbase Server is required for Delphi).
- What the structure of the application is. That is:
  - What the components of the application are (whether they are standalone)
  - What the **Main** files of the application are and each of the components
  - What the **Associated** and **Third Party** files are

For definitions of these file types, refer to *Recognition File Types* on page 20.

A clean installation of the application and each of the standalone components should be made on every platform the application is designed for. Data should be scanned and added into an application library.

Warning: The default Scanner might be changed to include other executables if required.

Stage 3 - Adding Obtained Information Into the Library

During this stage, application data is added into the ApE database, taking into account release relations between different application components (provided there are any) and naming conventions for the Publisher/Application/Version names.

Naming Conventions

In order to provide and maintain data consistency in the ApE database, applications should be described in a standardized way. This can potentially guarantee that if different people work with the database, the standard is not reduced.

Publisher

We define a publisher of a software application as a company that holds the full Copyright for the application. It is not necessarily the company that developed the product. Sometimes, the developers of the product do not own the copyright or have only partial copyright. If there is more than one company listed in the Copyright list, the one with the full Copyright should be chosen.

All publisher names should be spelled the same way in the ApE database. Before adding a new publisher name to the database it is a good idea to check whether the entry for the publisher already exists. Generally, publisher names should be entered into the database the way it was spelled in the Copyright for the product. However, there are a few things that should be kept in mind.

- Publisher Names with **The** in the title. For example, **The Wollongong Group.** The article (The) should be omitted, since most people will look for the name 'Wollongong' not 'The Wollongong'.

- If it is a publisher group (for example, Matrox Electronic Systems consists of Matrox Graphics, Matrox Imaging, Matrox Networks, Matrox Video), the branch of the company responsible for a particular development should be considered the publisher. You can enter a note in the Publisher details that the company is a member of a certain Publisher group.

- If one publisher acquires another, then use the following convention to rename the publisher that has been acquired:

  - The acquiring company name should be written first followed by the name of the publisher that acquired it in brackets. For example, Peregrine (fPrint).

Application

We define an application as a group of files that performs certain tasks for users.

If during an application installation you have to agree to a licence agreement, then we would say that the application requires a licence. The licence can be free of charge (for example, Netscape Communicator and Internet Explorer).

All applications should be spelled in the same way. Before adding an application into the database it is a good idea to check whether this application already exists in the database.

If the application is not already in the database, Application Name should be entered into the database the way it is spelled in the following sources (whatever is available):

- **Help About... information box** in the installation of the product
- **CD-ROM cover** for the product CD
- **Properties... Product Name** in Windows Explorer (not always consistent)
- **Product Name field** of the Windows Version resource data in the scan file.

Version

The version of an application is a release in which application functionality has been changed significantly compared to the previous release. This is reflected in the file data changes and/or structure of the application.

Version Name consists of the version number, operating system details and the language description. It is relatively difficult to establish how far you should go with the version number level (2, 4, 6, digits or more....).

The general rule is that it should contain enough differences so the recognition engine can distinguish it from the other releases. By the Operating Systems in the version description we mean the operating system(s) that the application version was designed for, as opposed to the operating systems it would be able to run on.

The version information should be taken from the following sources:

- **Help About... information box** (provided it is possible to launch the product after the installation)
- **Properties...Product Version** in Window Explorer, although that information is not always consistent
- **Product Version field** of the Windows Version resource Data in the FSF

Recognition File Types

Each file in the ApE database is categorized as Main, Associated or 3rd Party.

- **Main file**

  This is a key application file, that is, it is a file without which the application could not be running. It should always be present whenever the application is installed regardless of whether it is a minimum or complete installation. An application can have more than one Main file. If at least one of them is found on the machine, the application will be reported.

- **Associated file**

  This is a non-key file that is part of the application and is written by the publisher that developed the product.

- **3rd Party**

This is a file written by another Publisher (3rd Party) and is used in the current application. Examples of 3rd party files are some DLL files that are technically used by the application, but are also distributed freely and can be used by other applications.

Releases

When a publisher releases updates for an existing version of one of their products, they typically don't change the name of that version. Any particular version of a product could have several updates to it. ApE refers to the umbrella version as the Release, and to each of the updates as a Version.

Release Relations

The release relation mechanism has been implemented in order to report the correct number of licences required for an application installation.

If an application is a suite of products (for example, Microsoft Office or Lotus SmartSuite) the suite products (for example, Access, Excel, PowerPoint) need to be covered by one suite licence when they are installed as part of the suite and by individual licences when they are bought separately.

Release relations connect the main components that represents the suite (Office Tools in the case of Microsoft Office or Lotus SmartCenter in the case of Lotus SmartSuite) with other products belonging to the suite.

In the case when the main suite component is not installed on the machine, the installed suite components will be regarded as individual products.

Release relations are also useful when an application has a complex component structure. If application components can be installed on their own, they should be treated as independent applications, but the connection with the application should be reflected in release relations of that application.

Further Information
For further in formation about release relations please see *What is a Release Relation?* on page 72.

Stage 4 - Creating a New SAI File For Recognition Purposes

When you have made additions to the ApE database, export an SAI file from it that can be used for the recognition purposes when using Analysis Workbench and Viewer.

Further Information

For further in formation about creating a new SAI file, see *Exporting Data to an SAI* on page 60.

Using the MSI Importer to Add Applications to the ApE Database

The MSI Importer imports applications into the ApE Database without having to install them or perform and scan machines.

For further information about the MSI Importer and how to use it, see the *MSI Importer* Chapter.

Database Backups

To protect the Application Encyclopedia database from potential data loss, the database should be backed up regularly. In addition, it is recommended to store the backup medium in a different physical location from the database server.

The Application Encyclopedia database can be backed up by making a copy of the following directory and zipping it

<mysql_install_dir>\data\apeserver

Typically this will be:

c:\program files\peregrine\Desktop Inventory\8.0.0\mysql\data

Frequently Asked Questions

In This Section...

- *How Often Are Recognition Libraries Released?*
- *For Which Languages Are Recognition Libraries Available?*

- *Can I Use a New Recognition Library With An Old Version of the Software?*
- *I See That SAI Updates Are Available Regularly, What About Updates to the Server Based APE Database?*
- *Can I Get Peregrine to Add an Application Into the Master SAI?*
- *Does Peregrine Have a Unix Master SAI?*

How Often Are Recognition Libraries Released?

Recognition libraries for all languages are released once each quarter via the secure customer support site. Library updates are available without additional charge for all customers under maintenance.

For Which Languages Are Recognition Libraries Available?

There are currently three languages of libraries available for Desktop Inventory, in all cases the master (English) library should be used. This can be supplemented currently by French and German language libraries.

Can I Use a New Recognition Library With An Old Version of the Software?

As technology is always improving, it is necessary for the recognition libraries to evolve in conjunction with the application recognition technology within product releases.

The latest library can be only be safely assumed to be fully compatible with the current release. However, when attempting to use a new recognition library with an old version of the software, the software will warn you if the SAI version is not supported.

I See That SAI Updates Are Available Regularly, What About Updates to the Server Based APE Database?

Quarterly updates to the SAI and ApE databases are distributed. For both of these kinds of updates, an SAI file is used as the transfer medium.

For users of ApE, the updated file is used to identify and apply the changes since the last update, including additions, changes and deletions. The update file is automatically applied through the ApE Explorer's Import SAI function.

Can I Get Peregrine to Add an Application Into the Master SAI?

If the application is a "mainstream" application, this is possible. Contact support@peregrine.com for details.

If the application is the result of in-house development or is very specialized in nature, Peregrine will not add it to the standard application library.

Does Peregrine Have a Unix Master SAI?

Yes, Desktop Inventory comes with a UNIX Master SAI. The file is called **unix.sai** and can be found in the following default location:

C:\Program Files\Peregrine\Desktop Inventory\8.0.0\Common

We currently have support for Solaris, AIX and HP-UX.

3 ApE Explorer

CHAPTER

This chapter describes the Application Encyclopedia (ApE) Explorer, which is used to maintain the data in the ApE database and for importing and exporting data to different formats.

You will find information on the following topics:

- *Overview of ApE Explorer* on page 26
- *Getting Started* on page 26
- *The Application Encyclopedia Explorer Interface* on page 32
- *Using the ApE Explorer* on page 39
- *Release Relations* on page 72
- *Verifying Recognition* on page 75

Overview of ApE Explorer

The ApE Explorer is used to do the following:

- View information in your ApE database
- Add a new software publisher/application/release/version/file to the ApE database
- Delete an existing software publisher/application/release/version/file from the ApE database
- Edit the properties for an existing software publisher/application/release/version/file in the ApE database
- Import data from a Software Application Index (SAI) file
- Export a data to a SAI file
- Manually teach new applications to the ApE database

Getting Started

In This Section...

If Upgrading From a Previous IDD Release

Important: If you are exporting a Read Only SAI from InfraTools Desktop Discovery version 6.00 after the date of the latest Master SAI you will not be able to import the latest Master.SAI after importing the Read only SAI.

There are two workarounds:

1 Change the date on your PC before exporting a Version 6.00 Read Only SAI to a date prior to the latest Master.sai. Change the date back before importing the Read Only SAI and the latest Master.sai into ApE

2 Import the latest Version 6.00 Master.sai into the IDD Version 6.00 ApE database before exporting the Read Only SAI

If upgrading from an InfraTools Desktop Discovery release, do the following:

1 Create a Read-Only SAI by exporting it from your old InfraTools Desktop Discovery ApE database.

2 When you have installed the new Desktop Inventory software, import this SAI into the new APE database.

The new ApE Server will convert the data and place it in the ApE database.

3 Once you have imported the old SAI, import the new Master SAI into the new database.

Warning: This action will wipe the database. We recommend that you perform this procedure on an empty database.

For further information on the upgrade procedure, refer to the Desktop Inventory Installation Guide.

Installing ApE Explorer

Refer to the Desktop Inventory Installation Guide for instructions on how to install the Application Encyclopedia client.

Elements Installed
On installation the following components are installed on the workstation:

- ApE Server
- ApE Explorer

  This component communicates with the ApE server for customization of the database.

Manually Creating the ApE Database

> **Important:** The following procedure will delete all data in an existing ApE
> database should one exist already. Caution should therefore be
> used.

If Desktop Inventory is installed on a computer that already has a version of
MySQL installed then the installer will not attempt to install an additional
copy. The ApE Server can use the existing MySQL installation.

However, the installer will not be able to automatically create the ApE
database - this would require the use of a MySQL administrator login and
password, unknown to the installer if it didn't install the MySQL installation
in question.

The ApE Database needs to be created by a user who can provide a MySQL
login and password with sufficient rights to create databases and tables, and
to insert rows.

To manually create the ApE Database:

1 Using the Windows Explorer, copy the files **apeserv.sql** and **userids.sql** from
   the ape server directory (**C:\Program Files\Peregrine\Desktop
   Inventory\8.0.0\Ape Server** by default) to the MySQL bin directory
   (**C:\Program Files\Peregrine\Desktop Inventory\8.0.0\mysql\bin** by
   default)

   This is necessary due to a bug in the MySQL client which prevents the client
   from accessing files that reside in a directory with spaces in the path name.

2 Launch the MySQL command line client. This is done by running the
   following:

   c:\mysql\bin\mysql -u <username> -p

   from a command prompt. If you installed MySQL to a non-default location,
   the path to mysql.exe will be different.

   <**username**> refers to the MySQL login that you intend to use to create the
   database. The MySQL client will prompt you for a password.

3 Once successfully logged in, execute the following command:

   \. apeserv.sql

   This will execute the SQL script that creates the database.

4 Execute the following command:

\. userids.sql

This inserts the initial ID records into the ApE database.

5   Execute the following command:

\. userprefs.sql

This creates the ApE logins **ApeUser** and **ApeAdmin**.

6   Type **exit** to quit the MySQL client.

7   You may now delete the copies of the SQL scripts from **C:\mysql\bin**.

Your ApE database will now be created, and you can use the ApE software as described in this User's Guide.

Required Operating Environment

The Application Encyclopedia Explorer works on any Windows-32 bit Operating System.

These are the platforms on which the Ape Server component will run:

Supported OS	Notes
Windows NT 4.0 with SP6	
Windows 2000	Preferred platform
Windows XP	

Starting and Stopping the ApE Server Service

The ApE Server is an NT service. By default, this service is **started** when the program is installed.

If you should ever need to stop or start this service manually, there are two ways in which you can start and stop an NT service:

- Using the Services facility in Control panel.
- Using Net Start and Net Stop commands in a DOS window.

Refer to the documentation supplied with your Windows installation for more information on how to do this.

Important: You can configure various settings for the ApE Server by changing entries in the **apeserv.ini** file. Further information on how to do this is given in *Viewing Log Files* on page 31.

Launching ApE Explorer

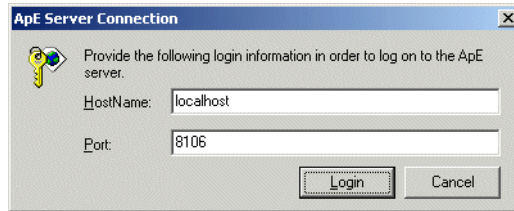**To launch the Application Encyclopedia Explorer:**

▶ Select the **Application Encyclopedia Explorer** entry in the **Peregrine|Desktop Inventory 8.0.0** sub menu of the **Start** menu.

Note: Multiple ApE Explorer clients can run concurrently on different workstations, connecting to the same ApE Database.

Logging in to ApE Explorer

Once you have launched the program, a splash screen is displayed showing the version of the software.

After a few seconds the Ape Explorer login dialog is displayed.

To log in to the Application Encyclopedia Explorer:

1 Enter the **Hostname** and **Port** for the Ape Server.

Double check that you have entered the correct details. A failed connection is usually due to incorrect information being entered here.

Note: The Hostname is not case sensitive.

2 Click the **Login** button.

You are now logged into the ApE Explorer.

Initial Activity

When ApE is first installed, all of the tables in the database are empty and need to be populated. The method used to initially populate a database is to import a **Master.sai** (French.sai, German.sai, if necessary).

Important: If you are planning to migrate your old ApE database to this version do not import the **Master.sai** file. Refer to the Installation Guide for Upgrade instructions.

Refer to *Importing Data from an SAI* on page 56 for further information on how to do this.

Viewing Log Files

You can view the **Apeserv.log** file using any text editor. By default this log file can be found in the following location:

Peregrine\Desktop Inventory\8.0.0\ApE Server

# The Application Encyclopedia Explorer Interface

In This Section...

- *Application Encyclopedia Explorer Workspace* on page 32
- *Icons* on page 33
- *Menus* on page 33
- *Toolbars* on page 37
- *Shortcut Keys* on page 37

Application Encyclopedia Explorer Workspace

Once login has been successful and you have imported data from the Master SAI, you will see data in the Application Encyclopedia Explorer.

ApE Explorer displays a hierarchical structure of software publishers, application names, releases, application versions and files contained in the ApE database.
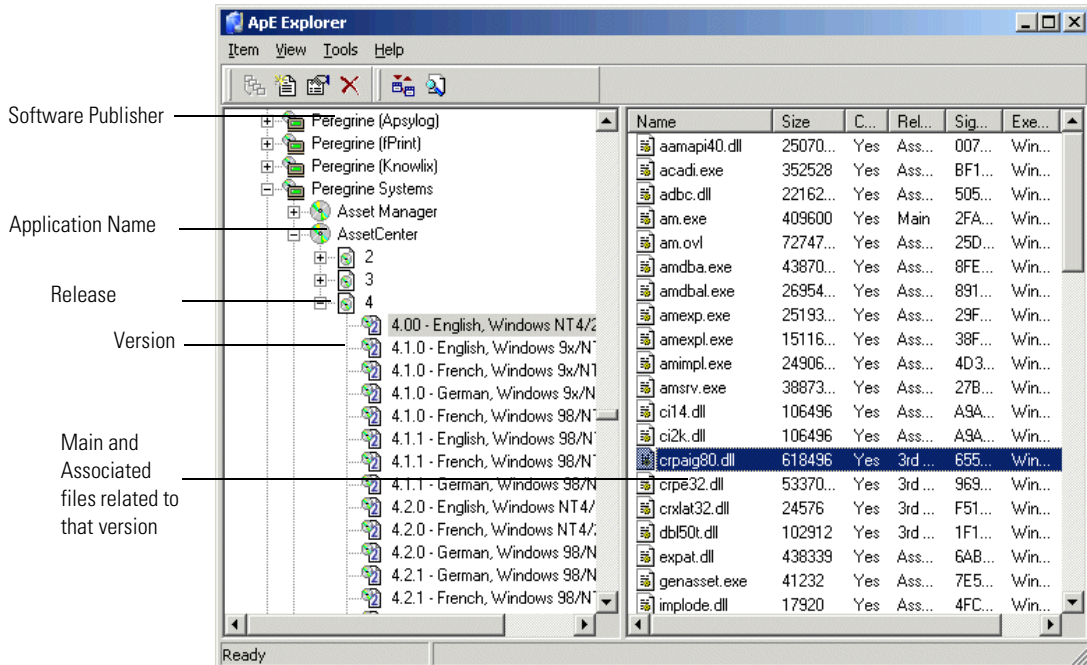
Icons

Icons are used to represent the various elements of the database.

Icon	Element
	Software Publisher
	Application Name
	Release
	Application Version
	File

Menus

Four menus are available:

- *The Item Menu*
- *The View Menu*
- *The Tools Menu*
- *Help Menu*

The Item Menu

Menu item	Description
New...	Depending on what has been selected in the main window, this option will display a dialog enabling you to create a new item in that category.
	For example, if have selected a version and you select **New**, then the **New File** dialog is displayed.
Expand	Expand the tree under the point you have selected.
Delete	Deletes the currently selected item. A confirmation message is displayed.
Properties...	Displays a properties dialog that shows the characteristics and parameters for the selected item. You can edit the entries in this dialog.
Edit Release Relations	Displays a dialog which allows you to establish or edit a release relation between two or more applications.
	Only displayed if an Application Release is selected.
Get Install String From Scan File...	Allows you to add an install string from a scan file that you know contains this detail. An install string is the line that is shown in Windows Add/Remove Programs.
	Only displayed if an Application Version is selected.
Import Version Data From A Scan File	Displays the **Scanfile Import Wizard**.
	This dialog is used when teaching new applications to the ApE database.
	Only displayed if an Application Version is selected.
Connect To Server	Establishes a connection with the Ape Server.
Disconnect From Server	Disconnect from the Ape Server.
Exit	Exit the ApE Explorer.

The View Menu

This menu contains only one option:

Refresh F5

Updates the display with the current information from the database.

The Tools Menu

Menu item	Description
Edit Application Types...	Allows you to add, edit or remove application types in the database.
Edit Operating Systems...	Allows you to add, edit or remove Operating systems in the database.
Import/Export SAI...	This option has two purposes: ■ Allows the import of data from the SAI files into the Application Encyclopedia database. ■ Allows the export of data from the Application Encyclopedia database to a customer read-only SAI file that can be used for recognition in Analysis Workbench and Viewer.
Verify Recognition...	This feature allows you to load in a scan file and check what the recognition results will be. You can also use this feature to add versions from the scan file directly to the ApE database.
Search for Files...	Allows you to search the database for a named file.
Import MSI Installer...	The **MSI Importer** can be used to teach any application that was packaged using the MSI Installer. This module allows software teaching based on an MSI file, which removes the need to install and scan software in order to add it to the ApE database. See the *MSI Importer* Chapter for more information.

Help Menu

The Help menu contains options for displaying the About box and the Online Help.

Toolbars

Toolbars allow you to access various commands without using the menu.

▶ Click on the toolbar icon to activate the function associated with it.

Buttons in the ApE Explorer toolbar offer the following functionality:

Icon	Name	Function
	Expand	Expands the tree under the point you have selected
	New	Depending on what has been selected in the panes, this option will display a dialog that enables you to create a new item in that category.
	Properties	Displays a properties dialog that shows the characteristics and parameters for the selected item. You can edit the entries in this dialog.
	Delete	Deletes the currently selected item. A confirmation message is displayed.
	Import/Export SAI	Displays a wizard that allows the import of data from the SAI files into the Application Encyclopedia database and the export of data from the Application Encyclopedia database to a customer read-only SAI file.
	Search for files in the ApE database	Allows you to search the database for a named file.

Note: When the mouse pointer passes over an icon, a tooltip appears indicating the command corresponding to the icon.

Shortcut Keys

You can quickly accomplish tasks you perform frequently by using shortcut keys.

Keyboard Shortcut	Function
Ctrl-A	Selects all of the items visible in the right-hand pane (the list view).
Ctrl-F	Allows you to search the database for a named file.

Keyboard Shortcut	Function
F5	Updates the display with the current information from the database.
Del	Deletes the currently selected entry.

Using the ApE Explorer

In This Section...

Viewing Information in the APE Explorer

ApE Explorer displays a hierarchical structure of software publishers, application names, releases, application versions and files contained in the ApE database.

Software Publisher
This is the name of the company or corporation that released the software application. For example, Peregrine Systems, Inc. is a software publisher.

Application Name
This is the name of the software program. For example, Desktop Inventory is an application name.

Release Application Version
When a publisher releases updates for an existing version of one of their products, they typically don't change the name of that version. Any particular version of a product could have several updates to it. In this scenario, ApE refers to the umbrella version as the Release.

Application Version
Each of the updates to an umbrella release is referred to as a Version.

Files
Each file in the ApE database is categorized as Main, Associated or 3rd Party. For an explanation of these file categories please refer to *Recognition File Types* on page 20.

Displaying Information About Software Applications

To display information about software applications:

1 Click on the + sign next to a **Publisher** in the tree. You can also use the **Expand** option in the **Item** menu.

   All the **Applications** by that software publisher are shown. A detailed view of the software applications is shown in the right pane.

2 Expand an application to display the **Releases** for it. A detailed view of the Releases is shown in the right pane.

3 Expand the tree further to display the **Versions** of that particular software application release that are in the database. Details for the software versions are shown in the right pane.

4 Click on a version of the software to display the **main**, **associated** and **3rd Party Files** for that version in the right pane.

In summary

- When a software publisher is selected, the applications associated with that publisher are displayed in the right pane.
- When a software application is selected, the releases associated with that application are displayed in the right pane.
- When a software release is selected, the versions associated with that release are displayed in the right pane.
- When a software version is selected, the files associated with that version are displayed in the right pane.
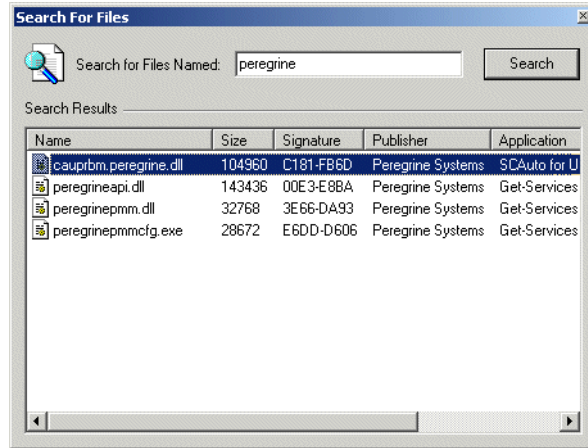
Searching For Files

To search for a file in the ApE Database:

1 Click the ⌕ icon or select the **Search for Files...** option from the **Tools** menu. The following dialog is displayed:
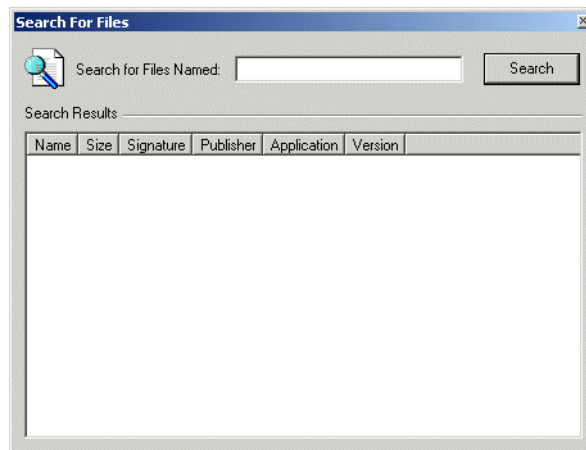
2 In the **Search for Files Named** field, type all or part of the file name or folder you want to find.

3 Click the **Search** button. The results will be displayed in the **Search Results** pane.

4 Double click on an entry to go directly to that file in the ApE Explorer window.

Using Wildcard Characters

You can use the asterisk as a substitute for one or more characters.

For any search string entered, even one with no wildcards, there is an implicit wildcard at the beginning and at the end of the search string. So **myfile** is the same as ***myfile***, and **my*le** is the same as ***my*le***.

Adding a New Software Publisher

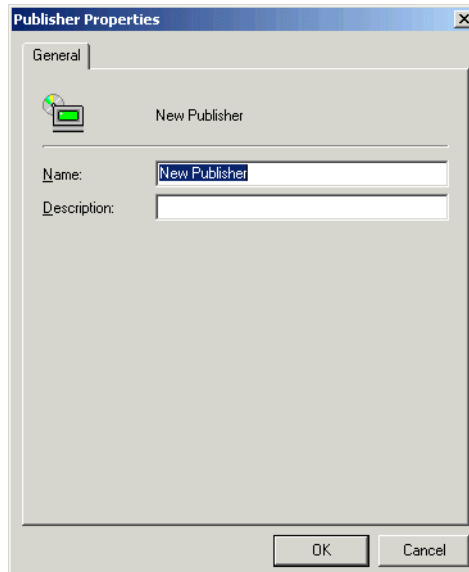This procedure is used to add a new software publisher to the ApE database.

To add a new software publisher:

1 Right-click on the Ape Database entry in the tree. It is the first entry (the root) of the tree.
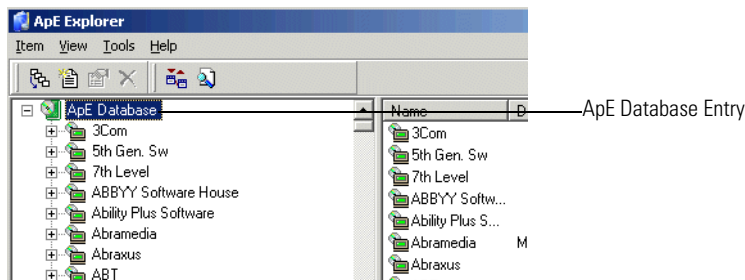
2 Select the **New...** option. You can also select the **New** option from the **Item** menu or click the ▣ toolbar icon.

The **Publisher Properties** box appears:

Enter the **Name** and **Description** for the new software publisher.

Warning: Ensure that the Publisher Name is spelled correctly. Before adding a publisher name to the database, check whether the entry for the publisher already exists in the database.

---

3 Click **OK**.

Once the new publisher is added, its details are displayed in the tree.

Deleting an Existing Software Publisher

This procedure is used to delete an existing software publisher from the ApE database.

To delete an existing publisher:

1 In the tree, right-click on the software publisher to be deleted.

2 Select the **Delete** option. You can also select the **Delete** option from the **Item** menu or click the ✕ toolbar icon.

A confirmation message is displayed.
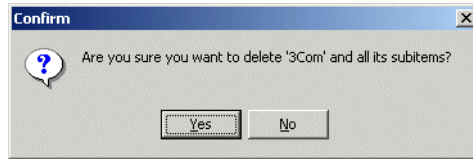
3 Click **Yes** to delete the software publisher and all its associated sub-items.

Editing the Properties for a Software Publisher

This procedure is used to edit the contact details and other properties for a software publisher.

1 In the tree, right-click on the software publisher to be edited.

2 Select the **Properties** option. You can also select the **Properties...** option from the **Item** menu or click the 🖺 toolbar icon.

The **Publisher Properties** box appears.

3 Edit the details as required.

4 Click the **OK** button to complete the edit.

Once the publisher details have been edited, the amended details are displayed in the tree.

Adding a New Software Application

This procedure is used to add a new software application to the ApE database.

To add a new software application:

1 In the tree, right-click on the software publisher the application belongs to.

2 Select the **New...** option. You can also select the **New** option from the **Item** menu or click the 🖺 toolbar icon.

The **Application Properties** box appears:

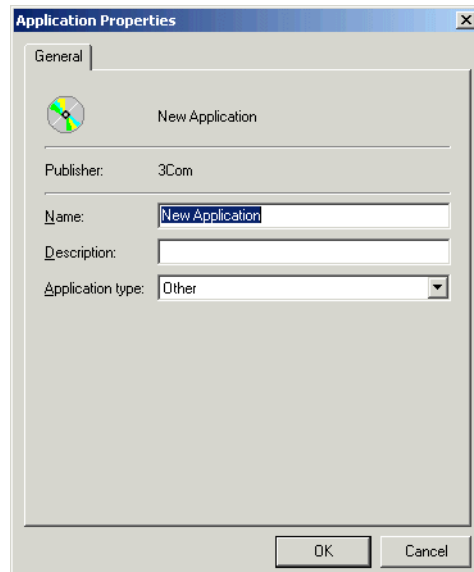Enter the following information:

- **Name**

  The name of the new application (must be entered)

- **Description**

  A text description indicating the application activity (optional).

- **Application Type**

  From the pull-down list select the software category that the application falls under (for example, word processor and spreadsheet).

3 Click **OK** to add the new application to the database.

Warning: Ensure that the application is spelled correctly. Before adding the application, check whether this application already exists in the database.
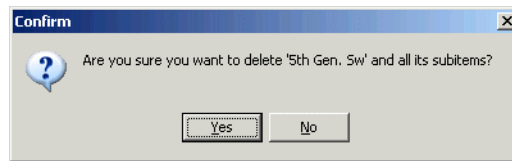
Deleting a Software Application

This procedure is used to delete an existing software application from the ApE database.

To delete an existing application:

1 In the tree, right-click on the software application to be deleted.

2 Select the **Delete** option. You can also select the **Delete** option from the Item menu or click the ✖ toolbar icon.

A confirmation message is displayed.

3 Click **Yes** to delete the software application and its associated sub-items.

Moving a Software Application to Another Publisher

You can move an application so that is contained under another software publisher.

To move a software application:

1 Drag and drop the application onto the software publisher entry in the tree.

A confirmation message is displayed.
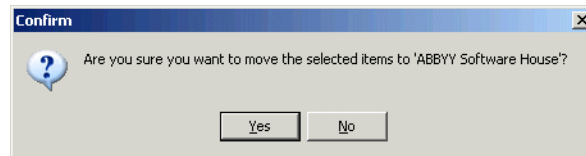
2 Click **Yes** to move the application.

Adding a New Release

This procedure is used to add a new software release to the ApE database.

To add a new software release:

1 In the tree, right-click on the software application the release belongs to.

2 Select the **New...** option. You can also select the **New** option from the **Item** menu or click the 🗐 toolbar icon.

The **Release Label Properties** box appears:
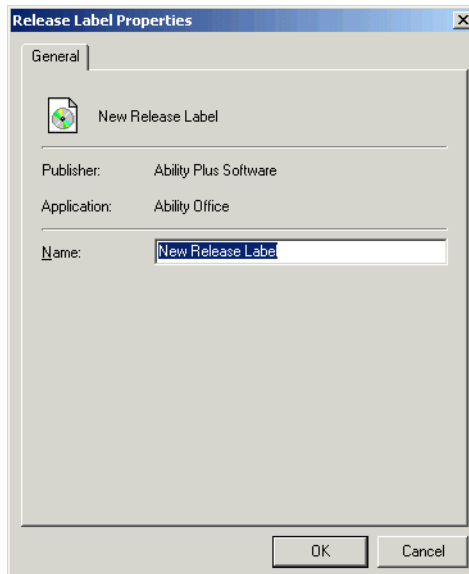
3 Enter the name of the new release (must be entered)

4 Click **OK** to add the new application to the database.

> **Warning:** Ensure that the release is spelled correctly. Before adding the release, check whether it already exists in the database for this application.

Deleting an Existing Release

This procedure is used to delete an existing software release from the ApE database.

To delete an existing software release:

1 In the tree, right-click on the software release to be deleted.

2 Select the **Delete** option. You can also select the **Delete** option from the Item menu or click the ✕ toolbar icon.

A confirmation message is displayed.
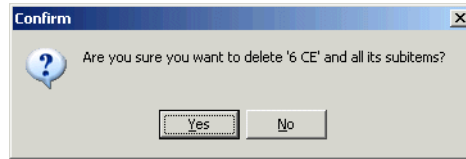
3 Click **Yes** to delete the software release and its associated sub-items.

Editing the Properties for a Release

This procedure is used to edit the properties for a software release.

To edit the properties for a release:

1 In the tree, right-click on the release to be edited.

2 Select the **Properties** option. You can also select the **Properties...** option from the **Item** menu or click the 🖼 toolbar icon.

The **Release Label Properties** box appears.

3 Edit the name as required (this is the only property you can edit for a release).

4 Click the **OK** button to complete the edit.

Once the release details have been edited, the amended details are displayed in the tree.

Adding a New Software Version

This procedure is used to add a new software version to the ApE database.

To add a new software version:

1 In the tree, right-click on the software release the version belongs to.

2 Select the **New...** option. You can also select the **New** option from the **Item** menu or click the ⬚ toolbar icon.

The **Version Properties** box appears:

3 Enter the following details:

- **Name**

  Name of the new version (must be entered)

- **Description**

  This is a user defined field that can contain any text.

- **Install String**

  This is the line that the version writes to the add/remove programs list when it is installed.

- **Operating System**

  The Base Operating System of the application version (for example, Windows NT). Select the Base OS from the drop-down list.

- **Language**

Select the language of the application version from the drop-down list.

4 Click **OK** to add the new application to the database.

# Deleting an Existing Software Version

This procedure is used to delete an existing software version from the ApE database.

To delete an existing version:

1 In the tree, right-click on the software version to be deleted.

2 Select the **Delete** option. You can also select the **Delete** option from the **Item** menu or click the ✖ toolbar icon. You can also select multiple versions for deletion by holding down the **Shift** key while selecting them.

A confirmation message is displayed.

3 Click **Yes** to delete the software version and its associated sub-items.

Editing the Properties for a Software Version

This procedure is used to edit the properties for a software version.

To edit the properties for a software version:

1 In the tree, right-click on the version to be edited.

2 Select the **Properties** option. You can also select the **Properties...** option from the **Item** menu or click the 🖼 toolbar icon.

The **Version Properties** box appears.

3 Edit the details as required:

- **Name**

  Name of the new version (must be entered)

- **Description**

  This is a user defined field that can contain any text.

- **Install String**

  This is the line that the version writes to the add/remove programs list when it is installed.

- **Operating System**

  The Base Operating System of the application version (for example, Windows NT). Select the Base OS from the drop-down list.

- **Language**

  Select the language of the application version from the drop-down list.

4 Click the **OK** button to complete the edit.

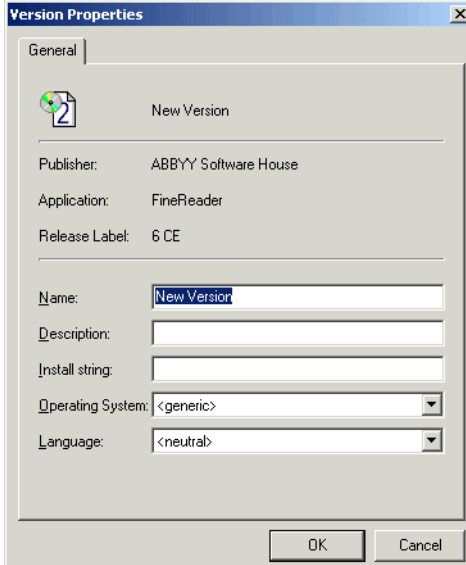Once the version details have been edited, the amended details are displayed in the tree.

Moving a Software Version to Another Release

You can move a software version and its contents so that is contained under another release.

To move a software version:

1 Drag and drop the version onto the software release entry in the tree.

A confirmation message is displayed.

2 Click **Yes** to move the version.

Adding the Install String From a Scan File

You can add an install string from a scan file that you know contains this detail. An install string is the line that is shown in Windows Add/Remove Programs.

To add an install string from a scan file:

1 Right click on a version in the ApE Explorer tree and select the **Get Install String From Scan File...** option.

2 Navigate to the scan file which contains the install string.

The following dialog is displayed showing a list of all install string contained in the scan file.
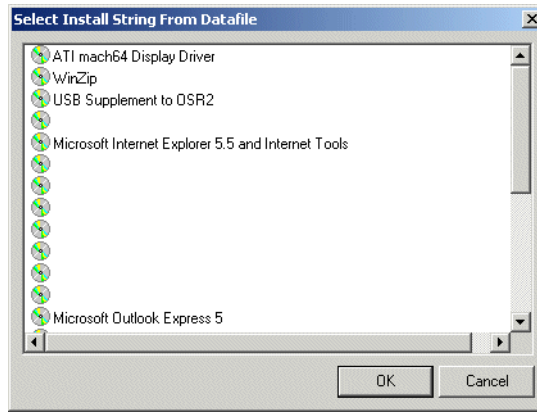
3 Select the appropriate application install string and click **OK**.

Now when you right click on the version and select **Properties...** you will see the install string has been populated automatically with the value you selected previously.

Adding a New File

> **Warning:** This method is not recommended for teaching files, because it is very prone to errors. Many file characteristics have to be added manually. The recommended method is to import files from data scan files. See *Teaching New Applications to the ApE Database* on page 62

This procedure is used to add a new file to the ApE database.

To add a new file:

1 In the tree, right-click on the software version that the file belongs to.

2 Select the **New...** option. You can also select the **New** option from the Item menu or click the 📄 toolbar icon.

The **File Properties** box appears:
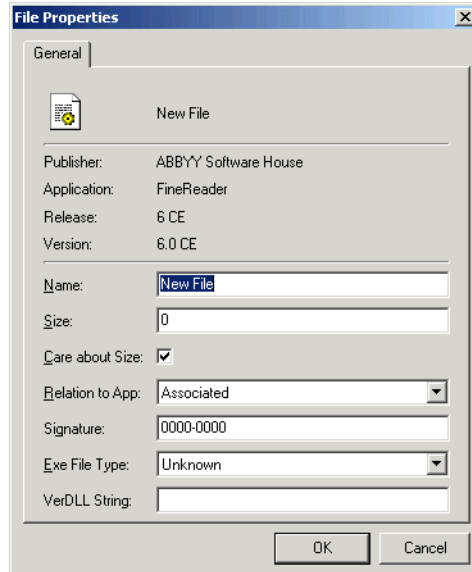
3 Enter the following details:

- **Name**

  Name of the new file (must be entered)

- **Size**

  The size of the file

- **Care about size**

  When this option is enabled, the size of the file is used as a recognition criteria. When disabled, file size will not be used in recognition and **N/A** will be displayed in the size column.

- **Relation to App:**

  Whether the file is a Main, Associated or 3rd Party file. See *Recognition File Types* on page 20 for definitions of these terms.

- **Signature**

  The signature is a checksum of the first 8 Kilobyte of a file.

- **Exe File Type**

  The type of executable file (for example, DOS com or DOS exe).

- **VerDLL string**

  The **VerDLL** is a concatenation of a set group of version data fields. Do not change the entry in this field unless absolutely necessary.

4 Click **OK** to add the new file to the database.

Warning: Ensure that the file name is spelled correctly. Before adding the file, check whether it already exists in the database for this version.

Deleting an Existing File

This procedure is used to delete an existing file from the ApE database.

To delete an existing file:

1 In the tree, right-click on the file to be deleted.

2 Select the **Delete** option. You can also select the **Delete** option from the **Item** menu or click the ✖ toolbar icon. You can also select multiple files for deletion by holding down the **Shift** key while selecting them.

   A confirmation message is displayed.

3 Click **Yes** to delete the software file.

Editing the Properties for a File

This procedure is used to edit the properties for a file.

To edit the properties for a file:

1 In the right pane navigate to and right-click on the file to be edited.

2 Select the **Properties** option. You can also select the **Properties...** option from the **Item** menu or click the 🖼 toolbar icon.

   The **File Properties** box appears.

3 Edit the details as required:

   ■ **Name**

   Name of the new file (must be entered)

   ■ **Size**

   The size of the file

   ■ **Relation to App:**

   Whether the file is a Main, Associated or 3rd Party file. See *Recognition File Types* on page 20 for definitions of these terms.

   ■ **Signature**

The signature is a checksum of the first 8 Kilobytes of a file.

- **Exe File Type**

  The type of executable file (for example, DOS com or DOS exe).

- **VerDLL string**

  The **VerDLL** is a concatenation of a set group of version data fields. Do not change the entry in this field unless absolutely necessary.

4 Click the **OK** button to complete the edit.

Once the file details have been edited, the amended details are displayed in the tree.

Importing Data from an SAI

This procedure takes you through the steps required to import the data from a SAI file into the database.

This procedure can be used either to perform an initial database load or to allow User SAI files or updated Master SAI files to be added to the existing ApE database.

The following types of SAI files can be imported into the Application Encyclopedia database:

- The Master database is supplied to customers in the form of three SAIs:
  - Master SAI
  - French SAI
  - German SAI

  The French and German SAI files are also exports from the Master database.

- User editable SAIs (**User.sai**) which are created by the Analysis Workbench. You may have taught new applications to your User SAI in Analysis Workbench. You can import this file into your own ApE database to make sure that these changes are included.

- Read Only SAIs are exports from a customer ApE database. They contain everything that was in the database. They can be imported into the ApE database, for example, when you are upgrading from InfraTools Desktop Discovery 6.xx to Desktop Inventory 8.0.0.

> **Warning:** When upgrading, the Read Only SAI has to be imported before the latest Master.SAI. Importing the SAI on top of existing data will wipe out the entire database.

To import data from an SAI:

1 Click the 🔳 toolbar button or select the **Import/Export SAI...** option from the **Tools** menu. The **SAI Import/Export Wizard** is displayed:
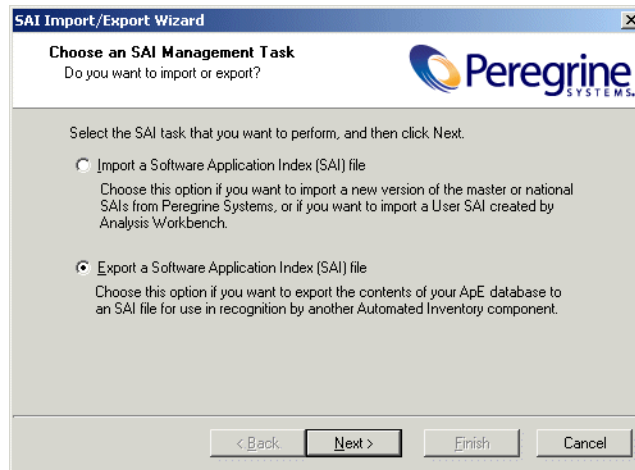
2 Select the **Import a Software Application Index (SAI) file** option.

3 Click **Next** to continue. The second page of the wizard appears.

4 Navigate to the SAI file that the data is to be imported from. The default location for the Master SAI is

Program Files\Peregrine\Desktop Inventory\8.0.0\Common\Master.sai

5 Click **Next** to continue. The third page of the Wizard appears.
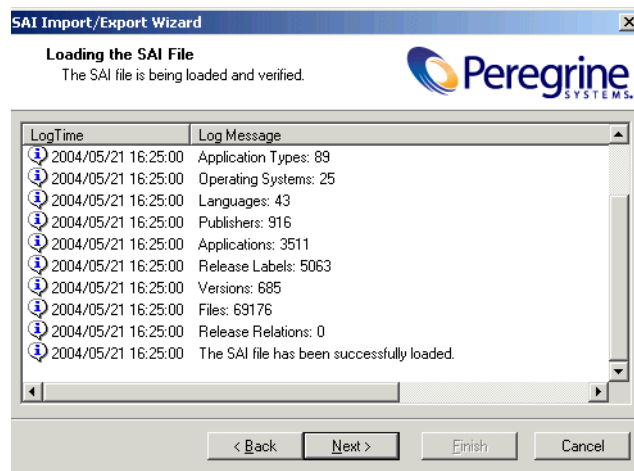
This page shows the SAI being loaded and verified. It shows the numbers of application types, operating systems, languages, publishers, applications, releases, versions and files that were loaded from the SAI.

6 Once it has finished this process, click the **Next** button to continue. The fourth page of the Wizard appears.
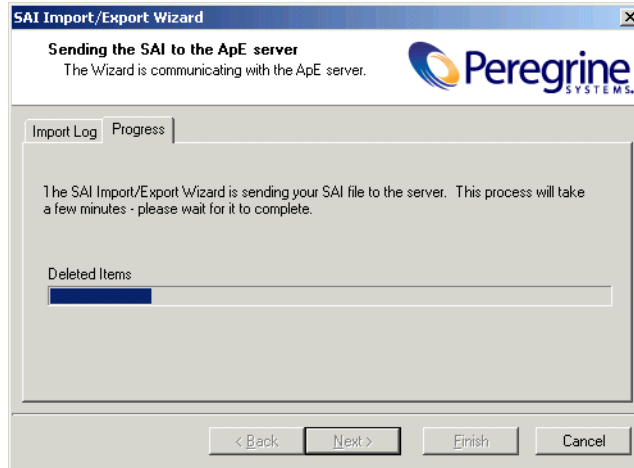
The **Progress** tab page shows the progress of the actual import. The contents of the SAI file are sent to the ApE Server.

7 Once the import process is complete, you can view a log of the events that took place by clicking the **Import Log** tab.

Note: The log is also available during the import procedure.
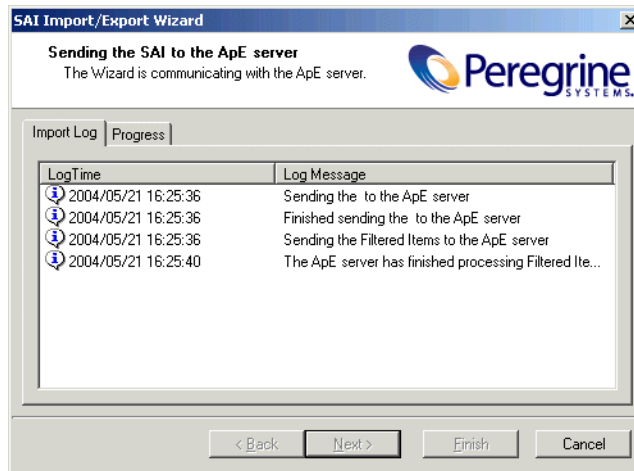
8 Click the **Finish** button. The import is now complete and the results can be seen in the ApE Explorer workspace.

Exporting Data to an SAI

Allows you to export data directly from the Application Encyclopedia database to a Read Only SAI that can be used for recognition in Analysis Workbench or Viewer.

To export data from an SAI:

1 Click the ▨ toolbar button or select the **Import/Export SAI** option from the **Tools** menu. The **SAI Import/Export** Wizard is displayed:
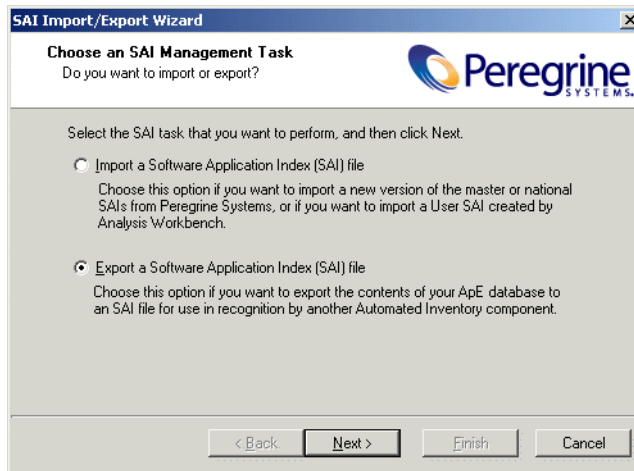

2 Select the **Export a Software Application Index (SAI) file** option.

3 Click **Next** to continue. The second page of the wizard appears.
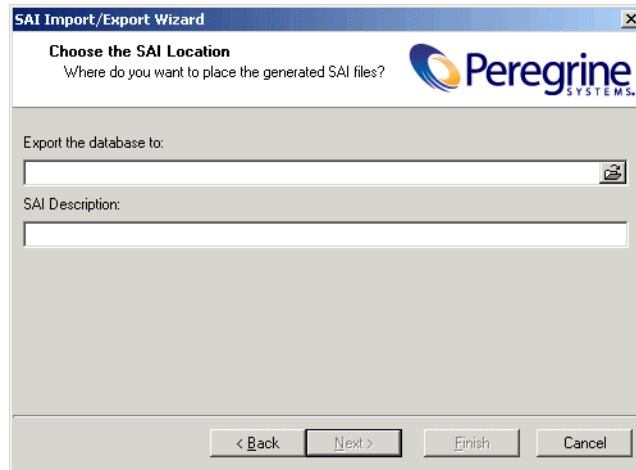
4 Navigate to the location that the data is to be exported to.

5 Enter a description for the SAI file.

6 Click **Next** to continue. The following page is displayed:
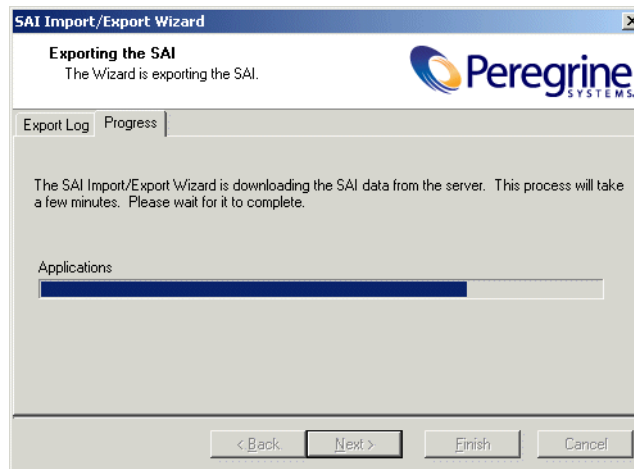
This page shows the progress of the export to the SAI. You can click between the **Progress** and **Export Log** tab pages.

If you should encounter any errors, these will be shown on the **Export Log** page.

7 Once the export is complete, click the **Finish** button to end the export process. A new **Read Only SAI** file will have been created in the location you specified. This file can now be used for recognition in Analysis Workbench or Viewer. It will contain Master SAI entries as well as entries added by you.

Teaching New Applications to the ApE Database

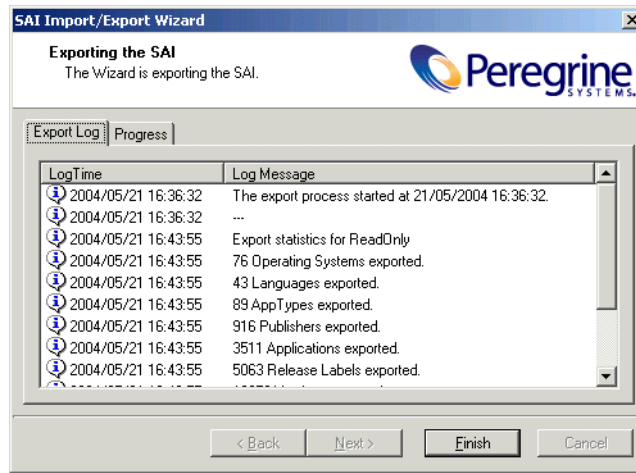See *Manually Updating Your ApE Database (Teaching)* on page 16 for detailed information about the planning of a teaching project.

Before You Start...

Before you start this procedure, ensure that you have done the following:

- Started with a scan of a machine before the application is installed.
- Installed the application
- Taken a scan of the machine after the application has been installed.

To teach new applications to the ApE database:

1 Create a new version for the application in the ApE Explorer, unless you want to add files to an existing version.

2 Click on the version in the tree view and select the **Import version data from a Scan File** option from the **File** menu. The **Scanfile Import Wizard** appears.

Depending on what entry you clicked in the tree, the information on the pages of the wizard will reflect this. In the following example, we have created a new version of MSN Messenger Service. In the tree and wizard it looks like the following:
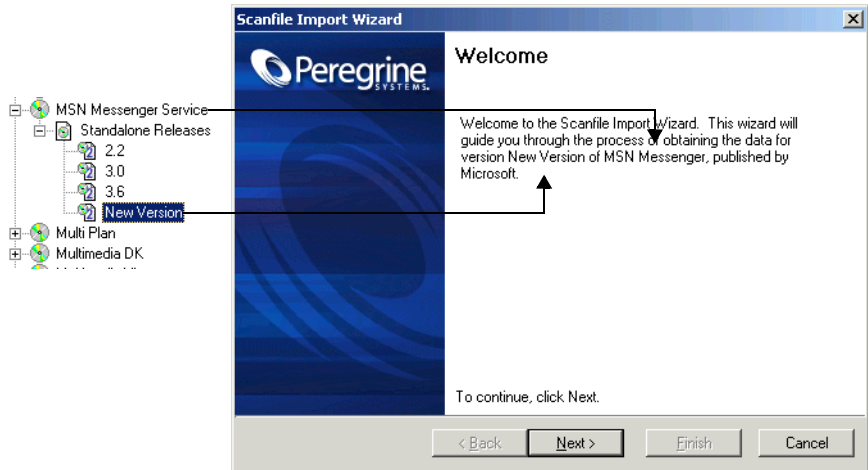
3 Click **Next>** to continue. A **Filtering** dialog is displayed. Set options in this dialog to specify which files should be loaded and optionally passed to the recognition engine.
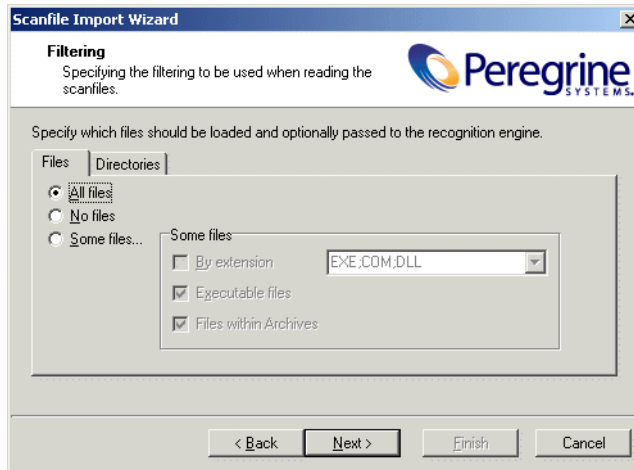
4   In the **Files** tab page, select the Load filter, select one from the following
    options:

  ■ **All files** Loads all files.

  ■ **No files** No files are loaded. For example, for hardware only analysis.

  ■ **Some files** Enables the **Some files** options:

  By extension… This allows you to specify the extensions of any files that
 you want to be included. Select the extensions from the drop-down list
  (this list maintains a history of previously entered extensions) or type the
  extension directly in to the box. Multiple extension must be separated by
  semi colons.

  Executable files This will allow all executable files to be processed against
  the recognition engine. Executable files include *.exe, *.com files, DLLs
  and other files containing executable code.

  Files within Archives This will process any files in archive files (for
  example, Zip files, CAB, LZH). When this option is checked, any files
  matching the other criteria (extension and/or executable) are loaded, even
  if they are inside an archive. If the option is cleared, no files within archives
  are loaded.

  If this is the only option enabled, nothing is loaded - it is not possible to
  load only files within archives.

5   In the **Directories** tab page, specify the directories to exclude.
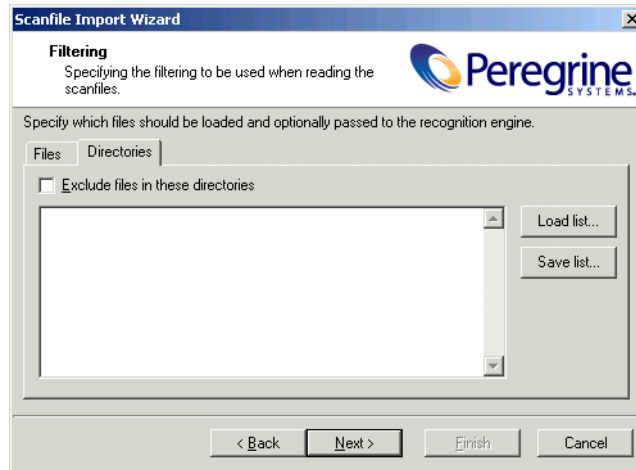
6 Select the **Exclude files in these directories** check box.

7 Place the cursor in the list box and type in the path (do not include the drive letter) of the directories to be excluded. An example of the format is as follows:

\WinNT\$NTServicePackUninstall$

8 You can save the current list displayed in the list box. To do this, click the **Save list...** button and specify the filename and path of the text file to save to. The default filename is **dirlist.txt**.

9 You can load a previously saved text list file by clicking the **Load list...** button to display a dialog box. Enter the filename and path or navigate to the file to be loaded.

10 Click **Next** to continue. The **Scanfile Selection** dialog is displayed.

Scanfile Import Wizard

Scanfile Selection
Selecting the scanfiles that the new version will be taught from

Peregrine SYSTEMS.

This wizard requires one scan file from prior to MSN Messenger's installation, and one or more of the same machine after MSN Messenger was installed. More than one scanfile may be specified if patches needed to be installed.

Pre-installation scan file:

Post-installation scan files:

Filename	Asset Tag	Description

Add...

Remove

< Back   Next >   Finish   Cancel

11 The wizard requires two scan files. The first scan file (pre-installation scan file) is the result of a scan performed on a machine prior to the installation of the program (in this case MSN Messenger Service) and the second (Post-installation scan file) is the result of the scan of the same machine after the new version of MSN Messenger Service was installed.

   a Enter the path to the pre-installation scan file.

   b Enter the post-installation scan files. Click the **Add...** button and navigate to the file. You can specify more than one file for cases where patches need to be installed too.

12 Click the **Next** button. If the Asset Tags of the two scan files are not identical, the following message will be displayed.

Confirm

⚠ The asset tags of these two scanfiles are not identical. This would indicate that the two selected scanfiles result from scans of different machines. The Scanfile Import Wizard will be unable to meaningfully analyse these scanfiles.
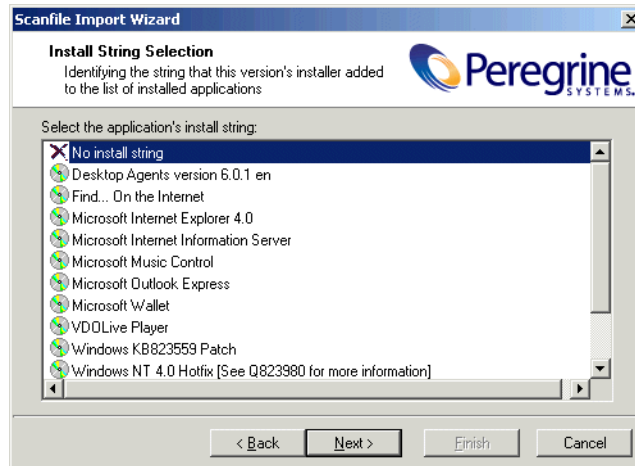Continue?

☐ Don't ask again   Yes   No
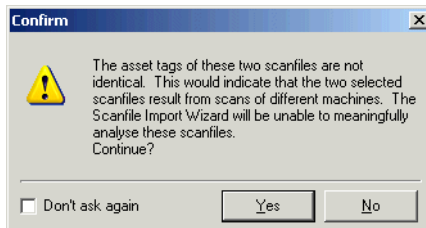
13 Click Yes if you want to continue.
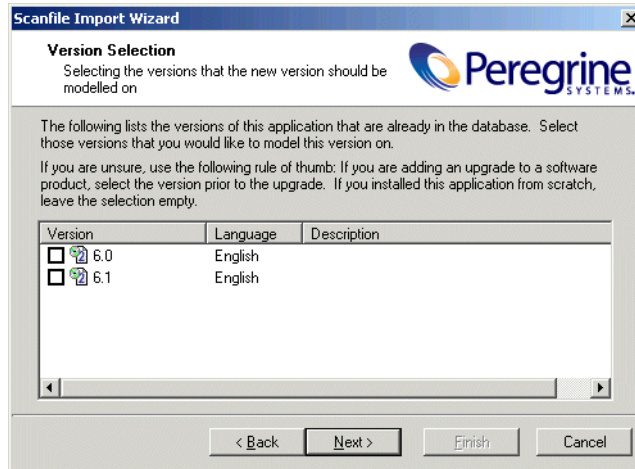
14 The wizard displays any versions of the software already contained in the ApE database. You can select versions you want to model this installation on. If you are unsure, use the following rule of thumb.
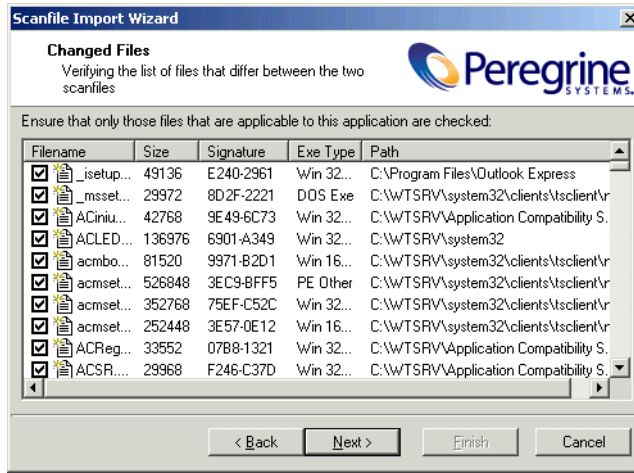
If you are adding an upgrade to a software product, select the version prior to the upgrade. If you installed this application from scratch, leave the selection empty.

15 Click **Next** to continue.

16 Highlight and select the string that this version's installer added to the list of installed applications.
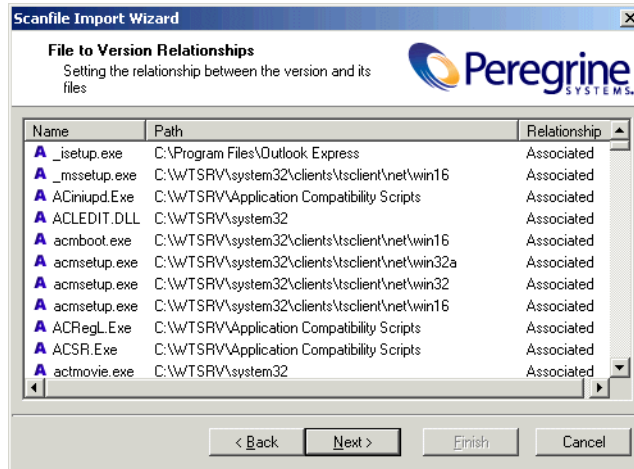
17 Click **Next** to continue.

18 This page shows files the teacher has picked up that differ between the two scan files. Check the files applicable to this application only. Different icons are used to denote the following:

| Icon	Meaning
🗎	New file added by the installation.
🗎	File changed by the installation.

19 Click **Next** to continue.

20 Set the relationship between the version and its files. Right click on an entry and you will see a menu which allows you to categorize the type of the file. You can also use the space bar or click on an icon to cycle through the various possibilities.

Icon	File type	Meaning
🅜	Main	This is a key application file, that is, it is a file without which the application could not be running. It should always be present whenever the application is installed regardless of whether it is a minimum or complete installation.
A	Associated	This is a non-key file that is part of the application and is written by the publisher that developed the product.
3	3rd Party	This is a file written by another Publisher (3rd Party) and is used in the current application. An example of a 3rd party files is some DLL files that are technically used by the application, but are also distributed freely and can be used by other applications.

21 Check the list of files and the status they have been given. If necessary, change them as required.

22 Click **Next** to continue.
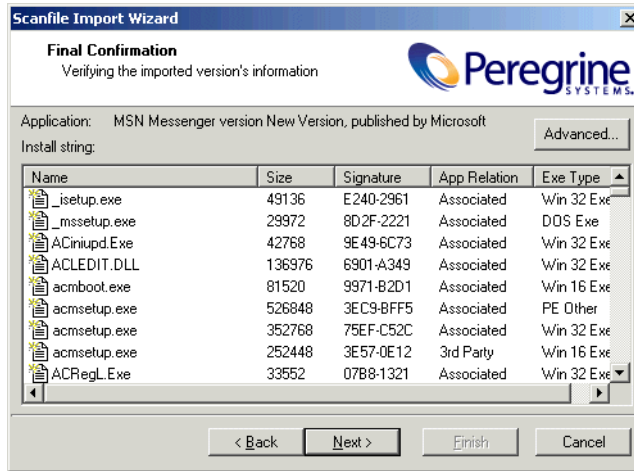
23 This page is the final confirmation page. It asks you to verify the imported version's information. It shows the application name and publisher, the install string as shown in the installed applications list and a list of files. Different icons are used to denote the following:

.

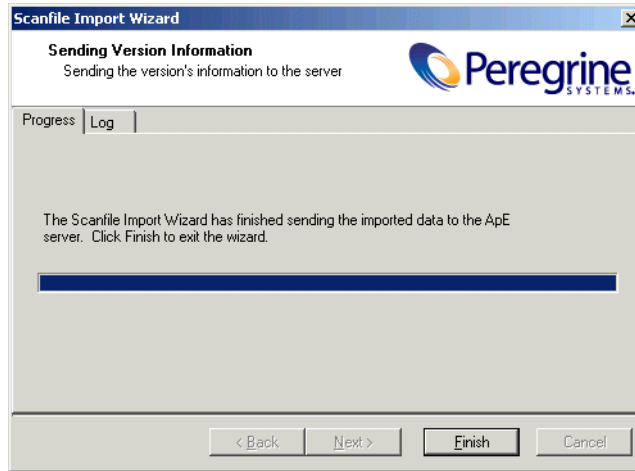| Icon	Meaning
📄	New file added by the installation.
📄	File that was changed by the installation.
🌐	File was selected in the **Advanced** selection, not in the standard wizard pages.

Advanced configuration

Click the Advanced... button. Now another window is displayed.
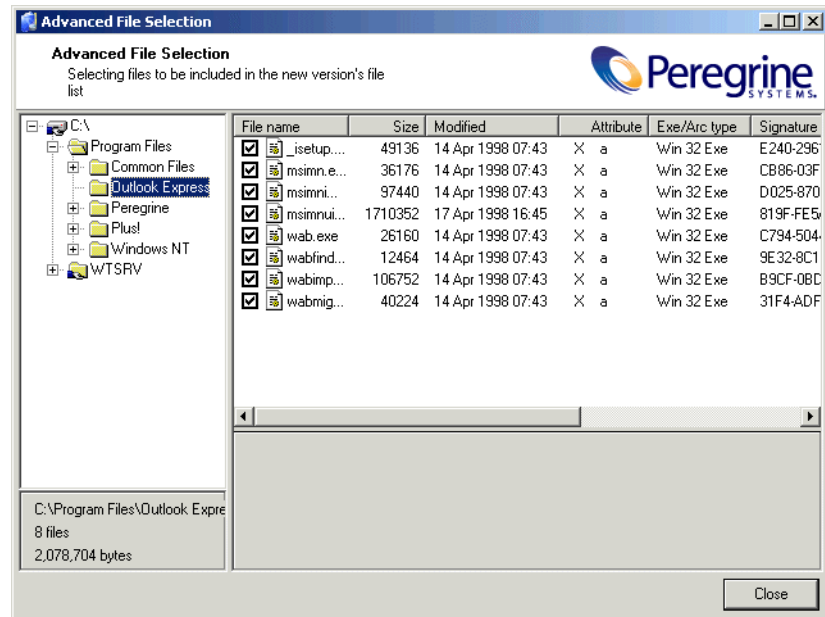
This is an explorer style window which allows you to select files that were found relating to the version in the scan file loaded. It provides you with a view of the whole scan file, so you can add whatever you want. If you browse to the files already included by the wizard, you will see that they are already checked.

1 Click the **Next** button to continue.

Scanfile Import Wizard

Sending Version Information
Sending the version's information to the server

Peregrine SYSTEMS.

Progress | Log

The Scanfile Import Wizard has finished sending the imported data to the ApE
server. Click Finish to exit the wizard.

< Back | Next > | **Finish** | Cancel

2 The wizard sends the details you entered to the ApE server and the data is
imported into the ApE database.

3 Click **Finish** to exit the wizard.

You have now taught an application to the ApE database.

Release Relations

In This Section...

- *What is a Release Relation?* on page 72
- *Establishing a Release Relation Between Two or More Applications* on
  page 73

What is a Release Relation?

Each identified file is categorized as either Main, Associated, or 3rd Party.
The recognition library can contain any number of files in a particular
version as belonging to either category.

When at least one Main file is identified on a machine, a licence is required
for the application, unless another application on the machine assumes
licence responsibility for the application.

Release relations are part of the Application Library; when two interdependent applications are identified on a machine, one application can assume licence ownership on that particular machine. The chain of release relations can be any number of levels deep.

Release relations solve the problem of Suite licensing simply and effectively, and is applicable to licence relationships outside of what would normally be considered a software suite.

For example, MS Office 2000 licenses MS Access, Word, Excel, PowerPoint and Outlook and is a well-known software suite. In addition Outlook licenses MS Data Access Components (ADO) 2.1 sp1 - if Outlook or another licensing application is not present on the machine, a licence is required for the software, but if Outlook is installed on the machine, Outlook assumes licence ownership.

Further Information

For further information about release relations, see *Release Relations* on page 21.

Establishing a Release Relation Between Two or More Applications

To establish a release relation between two or more applications:

1   Click on a release in the explorer tree view.

2   Select the **Edit Release Relations** option from the **Files** menu. The **Release Relations Editor** appears. This dialog is a non modal dialog. This means you can have it displayed while working in the ApE main window. Position the two dialogs on your screen so that you can work with both dialogs at the same time.
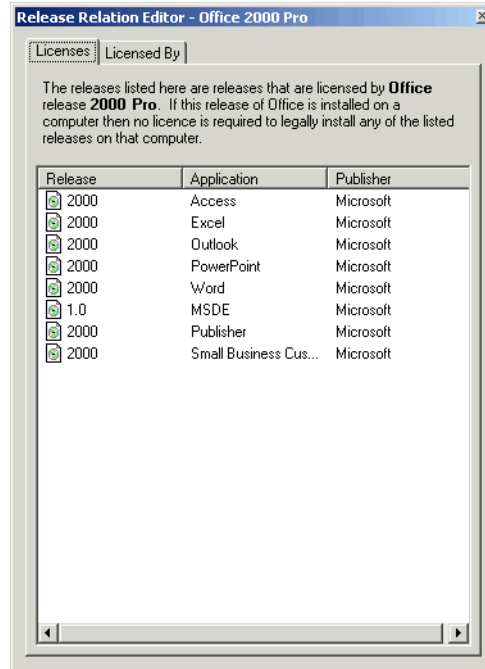
There are two tab pages in this dialog.

- **Licenses**

  The versions listed in this tab page are versions that are licensed by the selected application.

- **Licensed By**

  The versions listed in this tab page are versions that license the selected application.

3 To add an entry to either of these tab pages, drag and drop the entry from the explorer tree view.

4 If you make a mistake, you can remove the entry from the tab page by right--clicking on it and selecting the **Delete** option.

5 Right click on an entry and select the **GoTo** option to display the application version in the main ApE window,

Verifying Recognition

The Application Encyclopedia Explorer has a feature that allows you to load in a scan file and check what the recognition results will be. You can also use this feature to add files from the scan file directly to the ApE database. Each of these functions are covered in this section.

In This Section...

- *Checking Scan File Recognition* on page 75
- *Adding Versions from a Scan File Directly Into the ApE Database* on page 77

Checking Scan File Recognition

This feature allows you to view recognition results when a scan file is loaded into ApE Explorer.

To check scan file recognition:

1   Select the **Verify Recognition** option from the **Tools** menu. The following dialog is displayed:
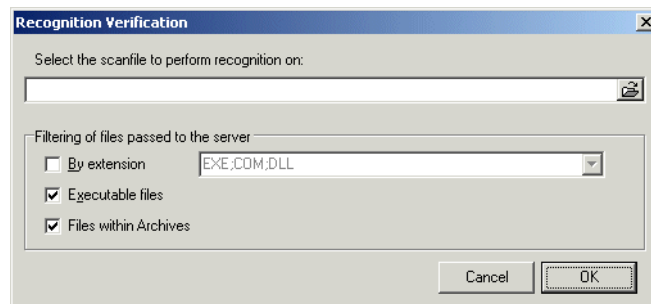
2   Navigate to the scan file to perform recognition on. You can also apply filters on files to be passed to the ApE server.

- By extension
- Executable files
- Files within archives

3   Click the **OK** button. The following dialog is displayed with the **Log** page showing.
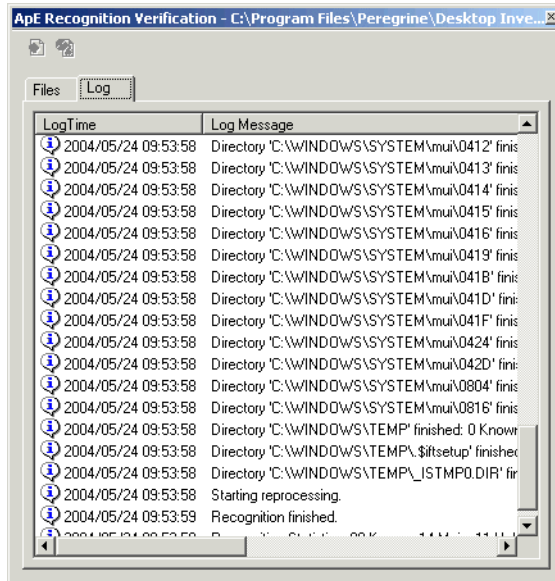
4 Click the **Files** tab.
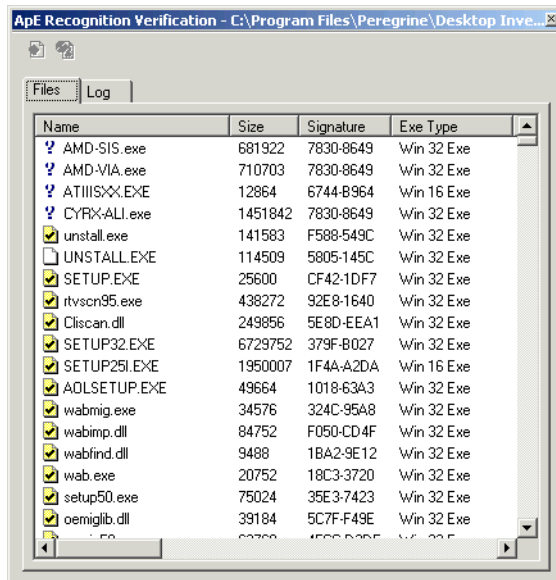
This tab page shows how the Ape Server recognized files in the scan file. Icons are used to indicate the following:

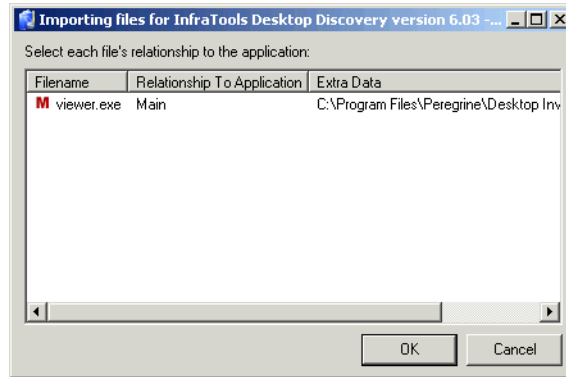| Icon	Meaning
☑	Has been recognized with a high degree of confidence.
▯	Has been recognized, but the recognition cannot be guaranteed 100%.
?	Does not exist in the ApE database and has not been recognized.

Adding Versions from a Scan File Directly Into the ApE Database

To add versions from a scan file directly into the ApE database:

1 The **ApE Recognition Verification** dialog is a non modal dialog. This means you can have it displayed while working in the ApE Server. Position the two dialogs on your screen so that you can work with both dialogs at the same time.

2 In the Ape Explorer, navigate to the Application version you want to add the file(s) to. In the following example we will be adding a file to the Peregrine InfraTools Desktop Discovery 6.03 version.

3 From the **Files** tab in the **ApE Recognition Verification** dialog, select a file(s) to be added to that Application version. Multiple files can be selected by holding down the Shift key while clicking on the files.

4 Drag and drop the file(s) directly onto the version.

Drag and drop the file(s) to the version
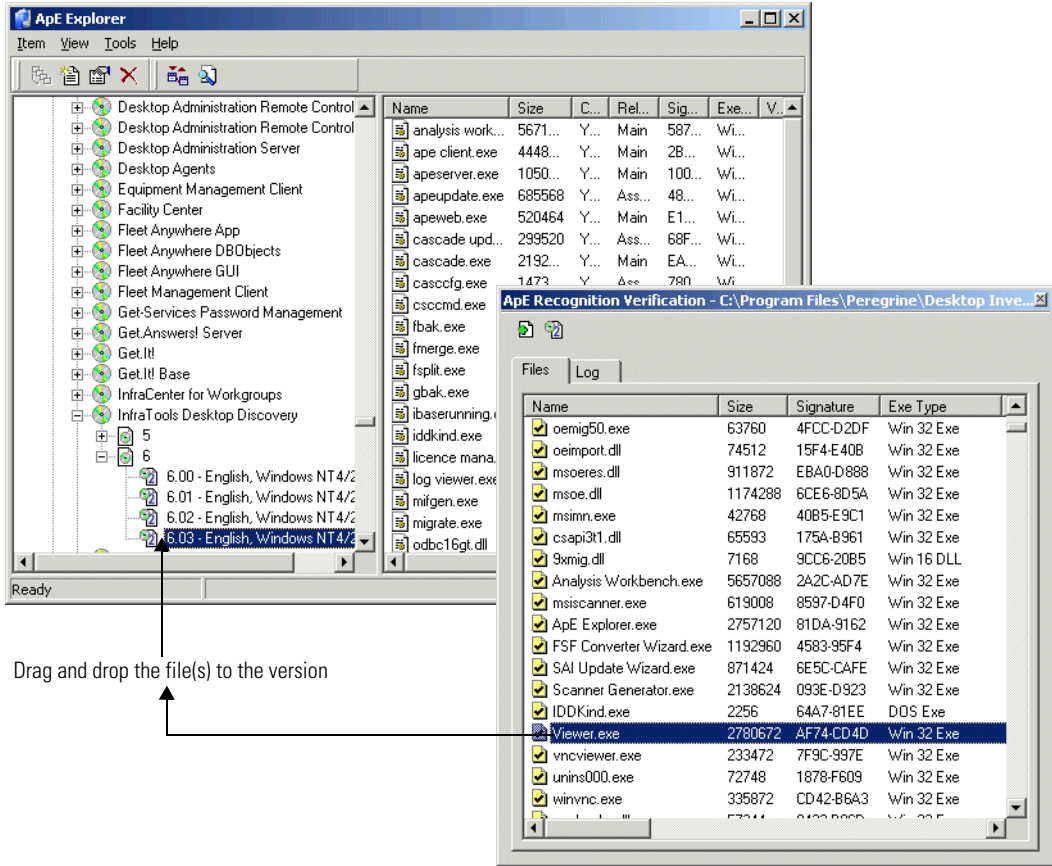
5 The following dialog is displayed.

6 Right-click on the file and from the menu, select the file's relationship to the Application. You have the following choices:

Icon	Meaning
M	This is a key application file, that is, it is a file without which the application could not be running. It should always be present whenever the application is installed regardless of whether it is a minimum or complete installation.
A	This is a non-key file that is part of the application and is written by the publisher who developed the product.
3	This is a file written by another Publisher (3rd Party) and is used in the current application. Examples of 3rd party files are some DLL files that are technically used by the application, but are also distributed freely and can be used by other applications.

7 Once you have done this, click **OK**. The file(s) is added to the application version.

4 The MSI Importer

CHAPTER

This chapter describes the Application Encyclopedia (ApE) MSI Importer, which is used to import applications into the ApE Database without having to install them or perform and scans of machines.

You will find information on the following topics:

Overview of the MSI Importer

In This Section...

What is the MSI Importer?

The **MSI Importer** can be used to teach any application that was packaged using the MSI Installer. This module allows software teaching based on an MSI file, which removes the need to install and scan software in order to add it to the ApE database.

MSI is short for Microsoft Installer and was introduced with Windows 2000, although Microsoft provides MSI as a separately deployable piece of software that can be added to older Windows operating systems.

To distribute an installer for a piece of software, a publisher no longer has to rely on proprietary installer packages for managing creation of directories, registry entries, copying of files or uninstallation. If instead the installer is distributed as one or more MSI files, the MSI process in Windows takes care of the details.

The MSI Importer is an ApE module that uses this standard to add applications packaged in MSI-based installers directly to the ApE database without having to install them first.

To teach the contents of an MSI files in ApE Explorer

1 Use the **MSI Importer** to view the structure and contents of the file. See *Components of a Software Installation* on page 85.

2 Once you have become familiar with the structure of the MSI file, teach the contents of the MSI file to the ApE database. See *An Example of Teaching an MSI Based Installer to the ApE Database* on page 106.

Important: The MSI Importer supports Microsoft's self extracting installers and installers that were generated using Wise for Windows Installer versions 3.5 and 4.0.l

---

Further Information

You can find comprehensive information about the MSI standard at:

▶ http://msdn.microsoft.com

The Advantages of Using the MSI Importer

- **Speed**

  The overhead of having to install the software and scan the machine is removed.

- **Convenience**

  The teaching is accomplished using only one computer, as the need for a separate 'clean' computer to install the software is removed.

- **Completeness**

  An installer often contains files that are only installed under certain conditions. For example, an application might require certain extra libraries under older operating systems to make up for missing functionary available under newer versions of Windows. Using the traditional method of installing software before teaching it, these extra files would only be found if the correct combination of software and operating system was used during installation. The MSI Importer, on the other hand, retrieves all of the files available in the installer.

The Challenges of Using the MSI Importer

- **Understanding MSI structure**

  You will need to understand how MSI files are structured. With MSI, every installation is different. It is necessary to browse the MSI Installer (and its associated directories) to understand how the installer has been structured before teaching the application.

  It is also necessary to understand what components the application has. For example, the same installer might contain application Server and Client components. While browsing, special attention should be paid to shared components included in the MSI installer. Some of them may already exist in the Microsoft library applications, such as MSDE (Microsoft SQL Server Desktop Engine) or DAO components.

- **Lack of MSI installations**

More and more software companies are adopting MSI as their installer platform. However, it will be some time before all software companies ship their products in MSI Installers. For these products, the traditional method of scan-install-scan-teach will still have to be used. See Chapters 2 and 3 for more information about this method.

- **Hybrid MSI Installers**

  Some MSI installers invoke legacy proprietary installers to install some components of the product. The files in the invoked legacy installers will not be available for teaching from the MSI Importer.

Components of a Software Installation

In This Section...

Software Components Provided in an MSI Packaged Installation

A software publisher must provide three components when publishing an application:

- The MSI System installer
- The **.msi** file
- The **Setup.exe** file

MSI was introduced with Windows 2000. Consequently, some computers will not be able to install MSI-based installers if they are provided only as **.msi** files.

It is for this reason that most MSI based installers are still invoked by running an executable file usually named **Setup.exe**.

The Setup.exe file:

1 Checks to see if the MSI system is installed

2 Installs it, if necessary

3 Once it has done the previous two steps, invokes the installation of the **.msi** file.

Note: The MSI system is available as a downloadable module from Microsoft. This module can be installed for the purpose of MSI installation.

Locating the MSI and Setup Files for Different Application Installations

There are two ways in which a software application can be made available:

- *Applications Shipped on a CD*
- *Applications Downloaded from the Internet*

Applications Shipped on a CD

Important: When loading such an installer into the MSI Importer, load the **.msi** file.

If the application is shipped on a CD, all three of the components will normally be placed separately on the CD. An installer packaged in this way may be identified by the following:

- A **.msi** file will be found in the same directory as the **Setup.exe** file.
- The **Setup.exe** file will normally be no larger than 1 MB
- An installer for the MSI system will be located either in the same directory or in a subdirectory.

An example of this type of installer is Microsoft Project 2000, as shown in the figure below.

## Applications Downloaded from the Internet

Important: When importing a self-extracting installer to the MSI Importer, load the **Setup.exe**. The MSI Importer will display an error message if you attempt to load an unsupported installer package.

If the application is downloaded from the Internet, a publisher will typically provide a single-file download. The **Setup.exe** file will be a sophisticated self-extracting archive containing the other two components.

You can identify this type of installer by the following:

- There is no associated MSI file
- The **Setup.exe** file will be several MB in size.

An example of this type of installer is MSN, as shown in the figure below.
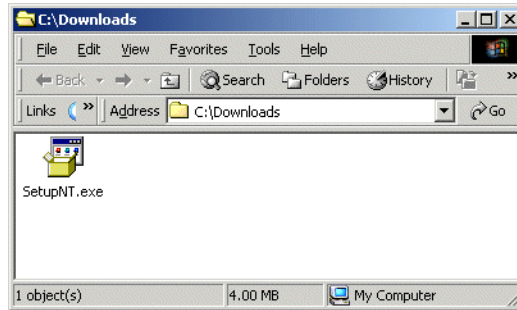
Other MSI Files on the Install Media

Often a search of the install media (the CD) will reveal a number of additional **.msi** files in subdirectories. These **.msi** files are usually installers for additional components the publisher provides with the application. Examples of these are required system updates and third-party database engines.

You will need to examine these **.msi** files to determine whether their contents should be added to the application in the ApE database.

- If the MSI appears in the nested installers tab, it should be taught - but only once.

- If the **.msi** file was provided by another publisher then its contents should either be added to the application as **third party files** or, in case when a component exists as a separate application, the MSI file data should be taught to the respective application. The connection between the two applications should be established through Release Relations (see *Release Relations* on page 72).

- If the software publisher has provided one of the additional **.msi** files themselves, then its contents should either be added to the application as **associated** files or, in the case when the supplied component exists as a separate application, the data should be taught to the respective application. The connection between the two applications should be established through Release Relations (see *Release Relations* on page 72).

MSI Installers and External Files

Some of the files referenced by an MSI file might be outside the MSI Installer package itself. This may take the form of a relative path directory structure or an Cab file outside of the MSI.

If an MSI file is moved, for example, it might cause a problem if external files are referenced. Always check the **MSI Reading Log** page to see if a **CAB Missing Error** has been displayed. If this is the case, ensure that all the files are in the correct locations. If you have copied a file from a CD, ensure that all the external files have been copied too.

Identifying Main Files

The **Main** file will probably be pointed to by a Windows shortcut installed in the Start menu. Use the list of shortcuts to identify and add the Main files. Bear in mind that an application may have more than one shortcut, and that not all of those shortcuts will point to Main files. You will need to examine the file and its internal version information to determine whether it is a Main file.

You can view more information about the list of shortcuts on the Summary page of the MSI Importer. See *Summary Tab Page* on page 95 for further information.

Identifying Third Party Files

Third Party files can be identified by checking the name of the publisher in the version information contained in most executable files. If the publisher name specified differs from the publisher of the product, the file is a third party file.

You can view the version information in the MSI Database page of the MSI Importer. See *MSI Database Tab Page* on page 97 for further information.

The Structure of an Application Packaged in an MSI Installer

An application packaged in an MSI-based installer is divided up by the software publisher into:

- *Features*
- *Components*

A Feature can contain any number of Components. Essentially, when you decide to install a feature, you are deciding to install all of the Components that comprise that Feature.

Features

A Feature is a part of an application's total functionality that is user recognizable and can be installed independently.

For example, a Feature could be a spell-checker, a thesaurus, or a group of clip art. Hierarchical relationships of parent and child Features commonly exist such that if a child Feature is installed, the parent Feature is automatically included as well.

Every, MSI-based installer contains a tree-like structure of Features. Each Feature could potentially contain children. If you decide to install a Feature then you could further decide whether or not to install its children.

An example of Features is shown in the figure below, which shows the Custom Setup page of the Desktop Inventory 7.3.0 Installation program.
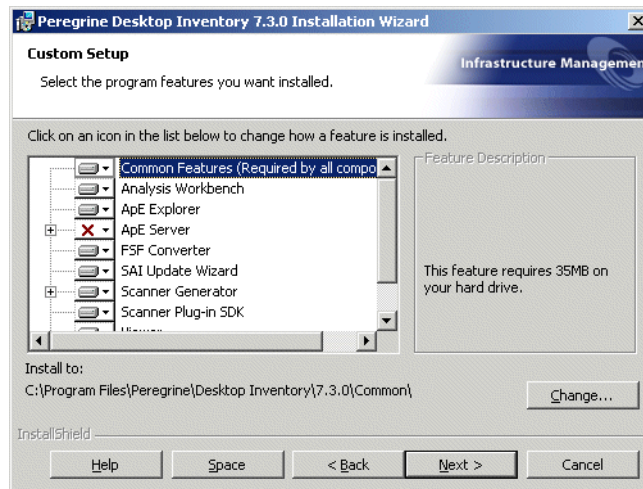
Components

A Component is a piece of the application or product to be installed. Example of Components included single files, a group of related files, COM objects, registration, registry keys, shortcuts, resources, libraries grouped into a directory, or shared pieces of code such as MFC or DAO.

Components can appear in more than one Feature.

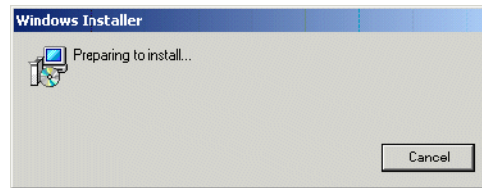# How to Open an Installshield MSI Installer in the MSI Browser

InstallShield is a popular installer authoring tool, used by many people worldwide. While ApE Explorer is unable to extract MSI files directly from InstallShield installers, it is possible to use the installer itself to extract the required files.

How Can You Tell if an Installer is Built Using Installshield?

When installing the application, the words **...is preparing the InstallShield wizard...** appear somewhere in the text of the opening page.

How Can You Tell if an Installer is MSI Based?

Very shortly after starting the installer, the following screen will be shown. The graphic on the left hand side is characteristic of MSI based installers.

Importing the InstallShield MSI Based Installer

The process to import an InstallShield based MSI installer takes advantage of the fact that InstallShield extracts the MSI in its entirety to the TEMP directory when it runs.

Start the installer and leave it open while you import the MSI and cancel it afterwards. When the installer closes it will delete the MSI and associated files that it extracted to the TEMP directory.

1   Start the installer.

2   Wait until the installer stops and waits for user interaction. This could be the Welcome page of the installer, or even an error message indicating that the installer cannot continue, like the following:
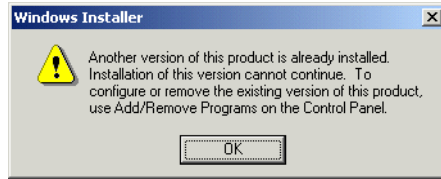
3 Leave the installer open and waiting for interaction until you have finished this process. If you close the installer it will delete the MSI file that you are attempting to import.

4 Determine where your temp directory is on the system. You can do this by opening a command prompt, and typing the following:

echo %temp%.

For example:

5 Open the ApE explorer.

6 Select **Import MSI Installer...** from the **Tools** menu

7 Enter (copy and paste or type in) the TEMP location revealed above.

8 Look for a directory named **_is*** where * is any combination of alphanumerical characters.

9 You will find the extracted MSI file in this directory - select it.

10 When you have finished importing the appropriate files, the installer can be closed. You can also cancel the installation (you do not need to install the application).
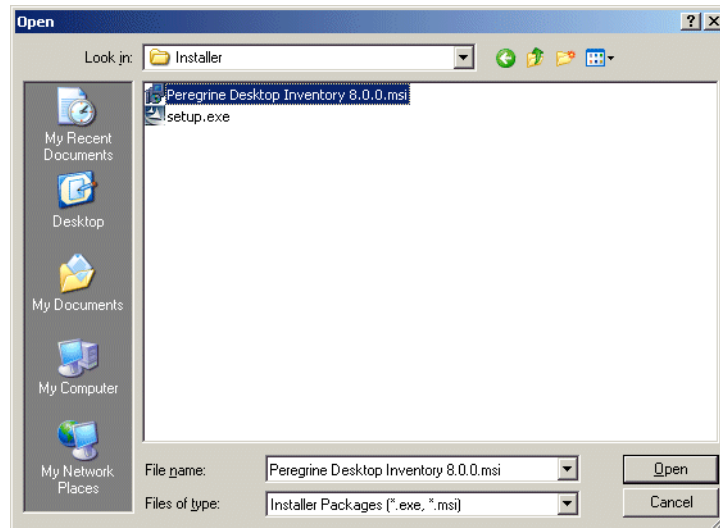
The MSI Importer Workspace

In This Section...

## Launching the MSI Importer

To launch the MSI Importer:

1 In the ApE Explorer, select the **Import MSI Installer...** from the **Tools** menu.

A **File Open** dialog is displayed.

2 Navigate to and select the installer file (this is either the **setup.exe** or the **.msi file** depending on how the publisher published the application).

For further information on which file to choose, see *Locating the MSI and Setup Files for Different Application Installations* on page 86.

The **Import MSI Installer** workspace is displayed.
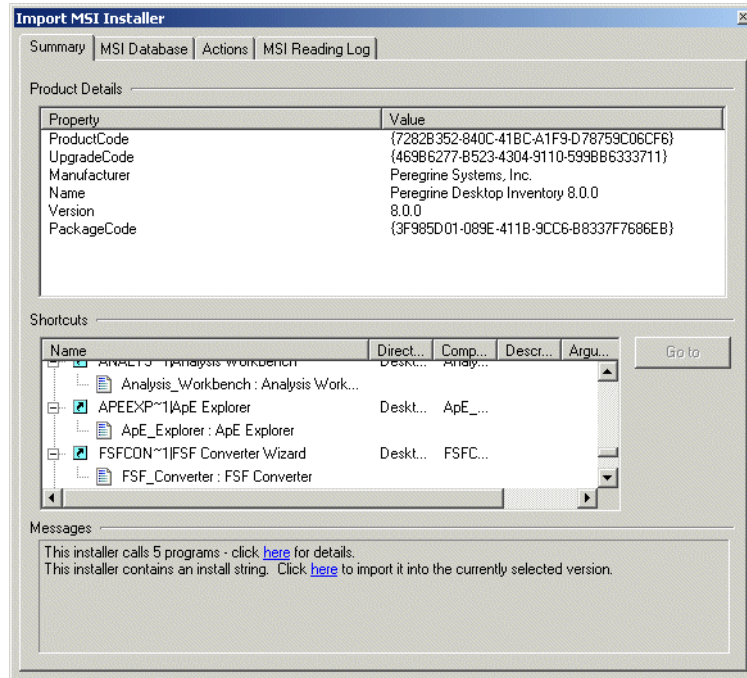
The **Import MSI Installer** window is non modal. This means you can have it displayed while working in the ApE database. Position the two windows on your screen so that you can work with both at the same time.

The **MSI Importer** window consists of four tab pages.

- Summary
- MSI Database
- Actions
- MSI Reading Log

Summary Tab Page

The **Summary** page shows information extracted from the MSI file.
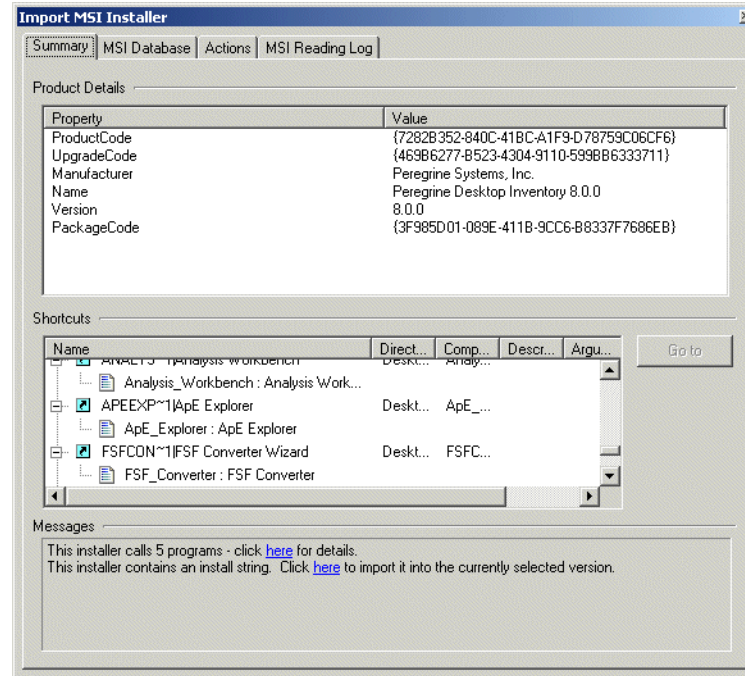
Information includes:

- **Product Codes**

  - **Product Code**

    The product code is an ID that is the main identifier for an application or product.

  - **Upgrade Code**

    This is an identifier used by the MSI system to determine which installed products are affected by a given upgrade.

  - **Manufacturer**

    The software application publisher

  - **Name**

    The name of the application. This is the application's install string.

  - **Version**

    The version of the application

- **Package Code**

  The package code is an ID identifying a particular Windows Installer package. It associates an .msi file with an application or product.

■ **Shortcut information**

Lists the shortcuts this installer created and shows the targets under each shortcut.

The following information about the shortcut is given:

- Name of the shortcut
- The Directory the shortcut points to
- The file pointed to
- A description of the shortcut command
- Any arguments applied if relevant
- The Working directory

The **Go to** button is only enabled when you select a Feature or if a shortcut's target exists in only one feature and you select the shortcut. If you select a shortcut in this pane and click the **Go to** button you will be automatically redirected to the MSI Database tab page and the file will be highlighted.

■ **Warnings**

Warning are provided whenever actions are found that require further examination. You will need to examine the entries associated with the action warnings and decide whether they are files that need to be taught to the application and what relationship they have with the application. Examine the warnings by clicking on the links in the bottom pane. You will be automatically directed to the appropriate place in the **Actions** tab page.
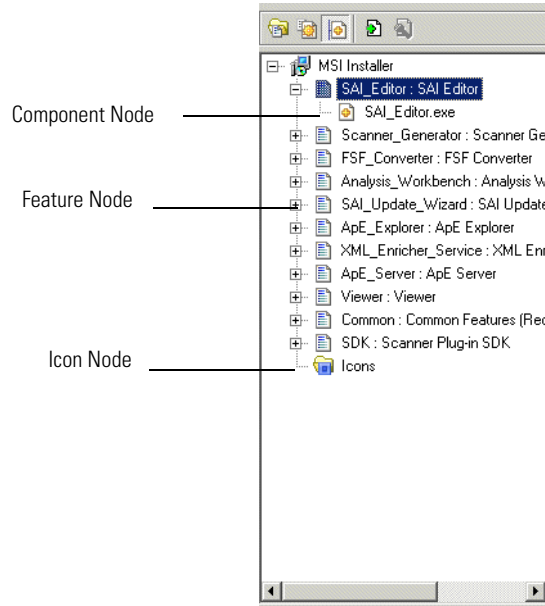
MSI Database Tab Page

The **MSI Database** tab page has several components:

- The Features and Components tree
- The Files pane
- The Version Information pane
- Toolbar
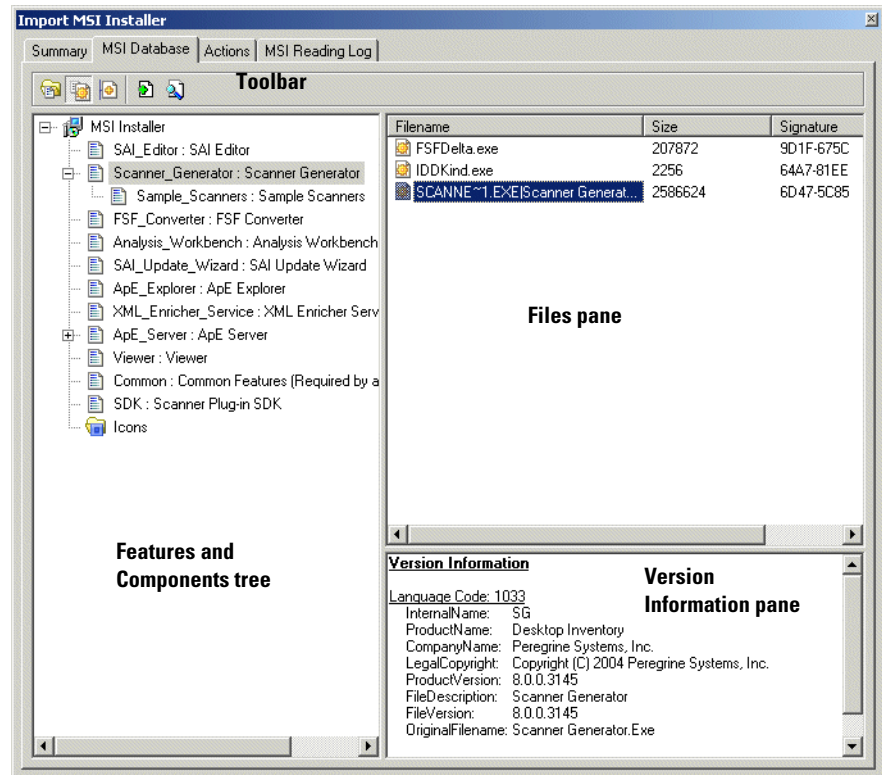
The Features and Components Tree

The **Features and Components** tree displays a tree view of the features and (optionally) their child components.

This displays a tree structure of the features and (optionally) components contained in the installer. The following icons are used to differentiate between the three types of node:

- Feature Node 📄
- Component Node 🔷
- Icon 🔲 Icons are stored in different locations. They are extracted and placed into the Icon node.

Pop-up Menu

A pop-up menu item (**Import Files**) is displayed when you right-click on the selected Feature Node. This option allows you to import the files contained in the Feature Node into the version selected in the ApE Explorer.

The Files Pane

The **Files** pane lists the files contained in the Feature or Component currently selected in the **Features and Components** tree.

Filename	Size	Signature	ExeType
scanW32.exe	598167	9C95-7A95	Win 32 Exe
scanOs2.exe	646443	BD0F-F511	OS/2 32 Exe
scanDos.exe	431595	5350-ECB4	DOS Exe
scanhpx	3223199	2AC8-C765	Unknown
scanlnx	2841665	F73C-887F	Unknown
scanaix	3556925	5DDF-37DC	Unknown
scanR32.exe	503074	7DEA-C94C	Win 32 Exe
scansp2	2287768	6325-D290	Unknown
scanW16.exe	738747	38D3-A697	Win 16 Exe

The **Files** pane displays a list of the files belonging to the currently selected node.

If the node is a Feature and the **Show Component Node** toolbar option ⊕ is deselected, the Feature will contain the files that belong to all of its (now invisible) Component nodes. You can find further information about the toolbar options on page 100.

Due to the legacy of MS DOS, the file name is stored in the MSI in two versions.

- The old-style MSDOS 8.3 format (with eight characters, a full stop, and then a three character extension)
- The more recent mixed-case long version.

Both of these names are displayed in the **Files** pane separated by a '|'. For example:

IMAGES~1.EXE|ImageSlicer.exe

If the MSI Importer was unable to extract a file from the installer in order to scan it then that file's icon will have a red 'No Entry' Symbol superimposed on it to indicate this.

You can find more information about why the file could not be extracted in the **Messages** tab page.

Pop-up Menu

Two pop-up menu items (**Find Files** and **Import Files**) are displayed when you right-click on a file.

- **Find File**

  The ApE Explorer will search its database for other occurrences of the selected file. The ApE Explorer's **Search for Files** window is displayed.

- **Import Files**

  This option allows you to import selected files into the version selected in the ApE Explorer.

The Version Information Pane

The **Version Information** pane displays the version information from the currently selected file.
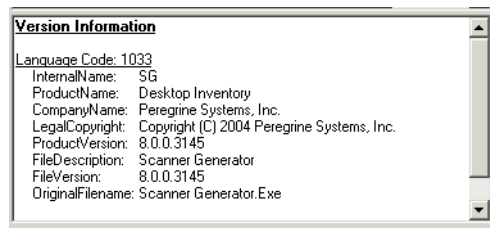
The **Version Information** pane displays data that the MSI Importer has extracted from the file currently selected in the **Files** pane. If no information was extracted from the file, this pane will be blank.

The information in this pane is exactly the same as the version information that the Scanners extract from a file during a normal scan of a computer. It is commonly used to determine the original publisher of the file so that the file relationship to the application (main, associated or third party) can be determined.

Toolbar

There are two types of buttons available from the toolbar:

- *Functional Buttons*
- *Option Buttons*

Functional Buttons

There are two functional buttons in the MSI Importer toolbar:

Find File 🔍

Clicking the Find File button 🔍 while a file is highlighted will cause the ApE Explorer to search its database for other occurrences of the selected file. The ApE Explorer's **Search for Files** window is displayed.
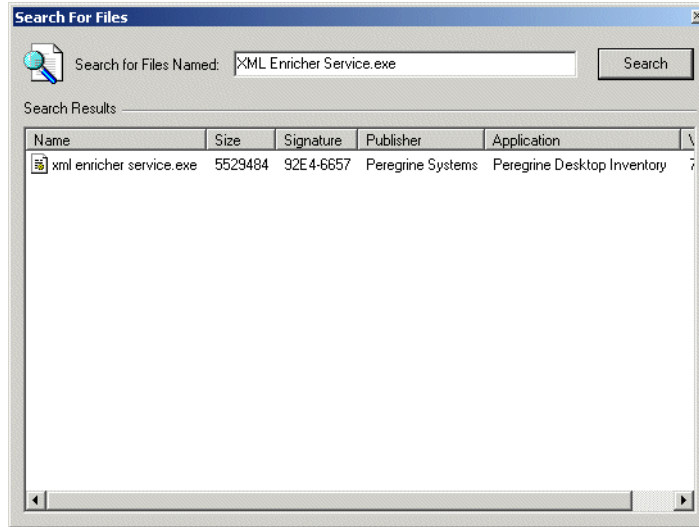
See *Searching For Files* on page 41 for further information on how to use the this dialog.

Add Selected Files to a Version 📄

Clicking this button 📄 imports the selected files into the version selected in the ApE Explorer.

It has the same effect as dragging the currently selected item from the **Features and Components** tree or the **Files** pane (whichever was last clicked) to the currently selected version node in the ApE Explorer. If no version node is selected in the ApE Explorer, this button will appear greyed out.

Note: When the mouse pointer passes over an icon, a tooltip appears indicating the command corresponding to the icon.

Option Buttons

Three option buttons are also available from the toolbar. All three of the option buttons are sticky buttons.

- If the option is selected the button appears depressed

- To de-select the option, click the button again and the button returns to

  its normal state

Hide Features With no Files

Enabling this option causes the ApE Explorer to only show those Features that contain files. Features without files in an installer are typically containers for registry entries, short cuts and services, and as such are not used to teach to the ApE database. They therefore appear empty in the tree and can safely be ignored. This option makes them invisible.

Note: When enabled, the MSI importer queries the information on all Features and Components before it displays the tree. This could take some time on larger installers.

Note: The option to remove empty Features is carried out before the option to **Show only executable** files. This means that if a node has only non-executable files, it will still appear in the tree even though the **Show Only Executable Files** option has caused it to be empty.

Show Only Executable Files

When this option is enabled only those files that were identified as executable are visible. Further, only the files displayed will be added to the ApE database during imports. This is the default setting, as non-executable files should not be added to the ApE database.

Show Component Nodes

When this option is enabled, the **Features and Components** tree will contain a node for every Component. This is a true reflection of how the MSI-based Installer is structured internally. That is, a Feature does not contain files directly when this option is applied. All of the Features contain Components, and those Components in turn contain the files. This is useful for determining the origin of a particular file.
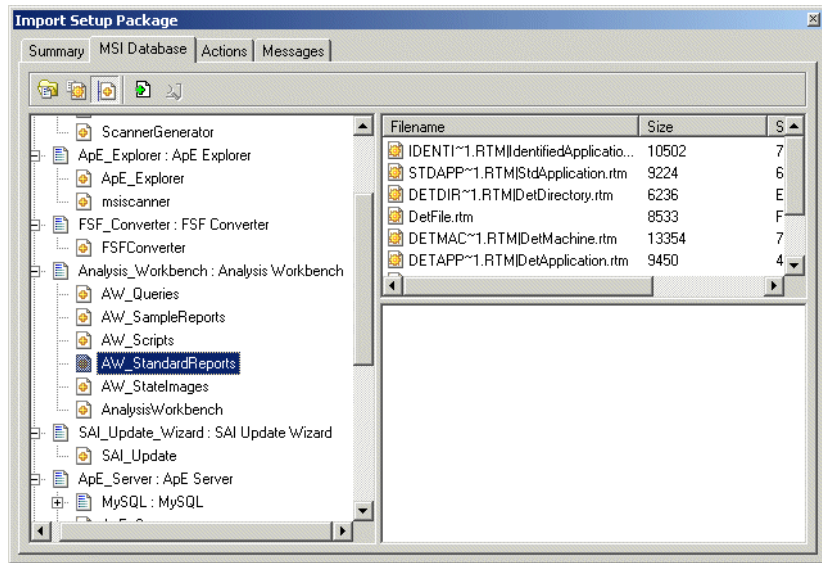
When this option is disabled, the Component nodes are not shown. All of the files contained in the Component nodes are placed directly under the Feature node. While this is a less accurate view of the installer, it is more user-readable. You do not have to click on the hundreds of Component nodes to view the files a Feature would install.
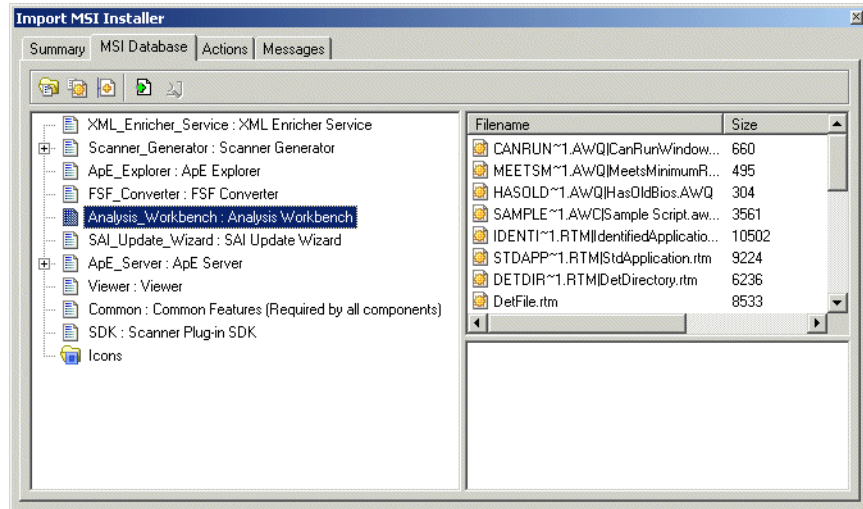
In the example above, note how the files that were shown for each individual component now appear with those Components' parent feature.

Actions Tab Page

This page lists the actions that the MSI Installer invokes during the installation process. It is necessary to examine these actions in order to determine whether any of them will install files not contained in the MSI.
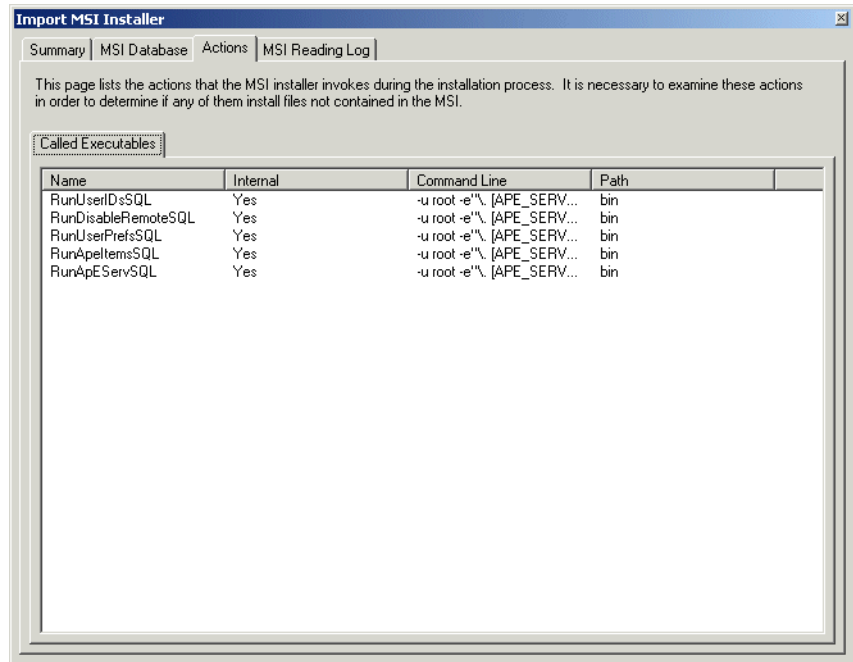
- **Nested Installers**

  A nested installation action installs another Windows Installer package during a currently running installation. A nested installation action can only reinstall or remove an application that has been installed by the current application's installation package.

- **Called Executables**

  This page shows any actions that may install some other executable files.
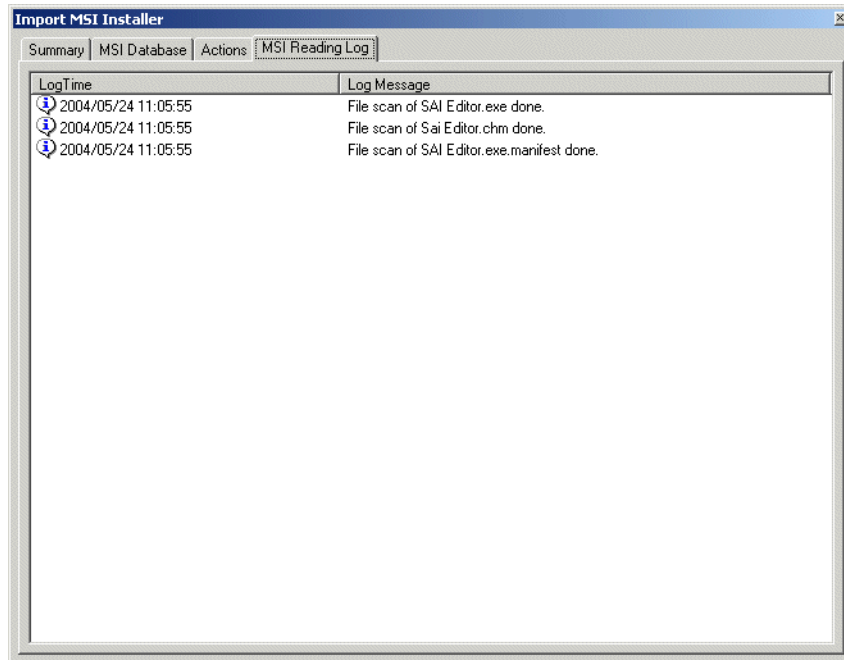
- **DLL Calls**

  This page shows any dll calls that may have been made.

- **Invoked Scripts**

  This page shows any scripts that may have been invoked

MSI Reading Log Tab Page

The **MSI Reading Log** tab page is used to view the log. It displays messages generated by the MSI Importer when reading the MSI-based Installer.

An Example of Teaching an MSI Based Installer to the ApE Database

In This Section...

Step 1 Importing the MSI Installer Package

1 In ApE Explorer, select **Tools|Import MSI Installer package**.

2 From the **Open** dialog navigate to and select the MSI file to use. The file will have an extension of **.msi** or **.exe**.

In this example we have used **Peregrine Desktop Inventory 8.0.0 msi**. This is the msi file supplied for the installation of this product.
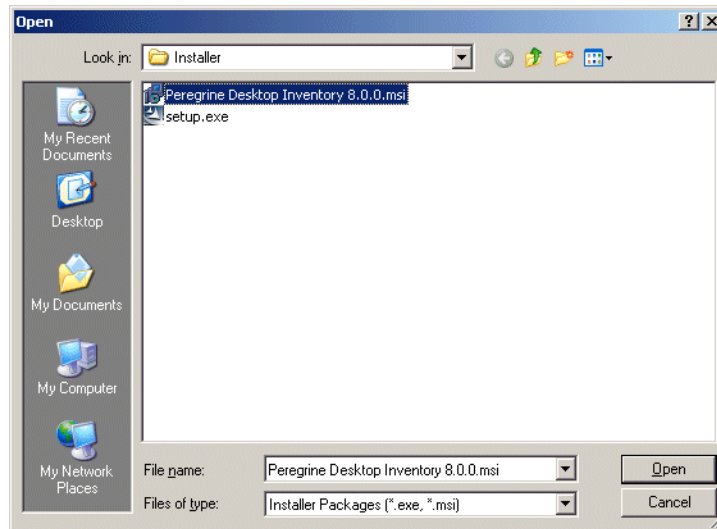
3 Click on the **MSI Database** tab to examine the contents of the msi file.
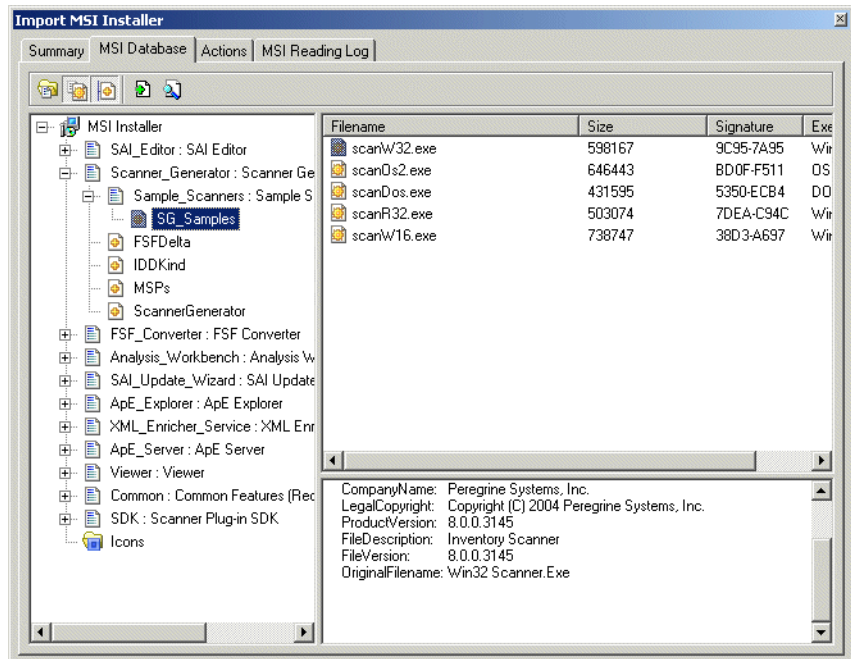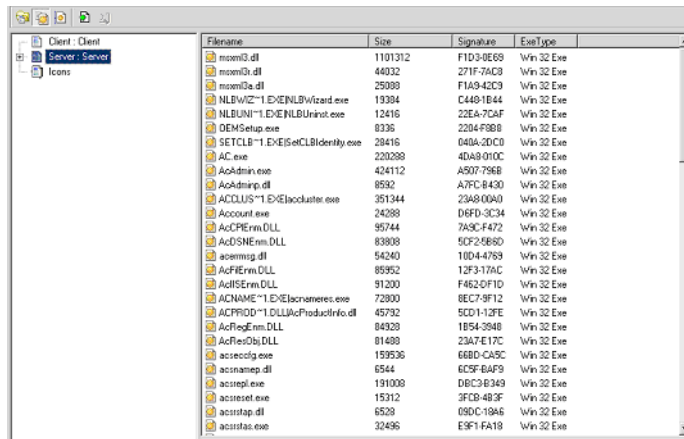
4 In this example, there are several features. Features are displayed in the following way and are indicated by the 📄 icon.

Unique Name Key: Display name

If we look at the Scanner Generator Feature entry we can see that the Unique Name Key is **Scanner Generator**, while the display name has no underscore.

Looking through the MSI structure, try to look for the components of the application that can be installed separately and therefore should be taught as individual applications. For example Client/Server parts (see the example below) or some shared applications that are also packaged in the installer, for example MSDE or DAO components.

This example displays the contents of the Microsoft Applications Center 2000 installer. The same installer contains both Server and Client components of the application.

5 Click on the **Show Only Executable Files** button 🔘 on the toolbar and make sure that all other toolbar buttons have been deselected. Now you will see only executable files.
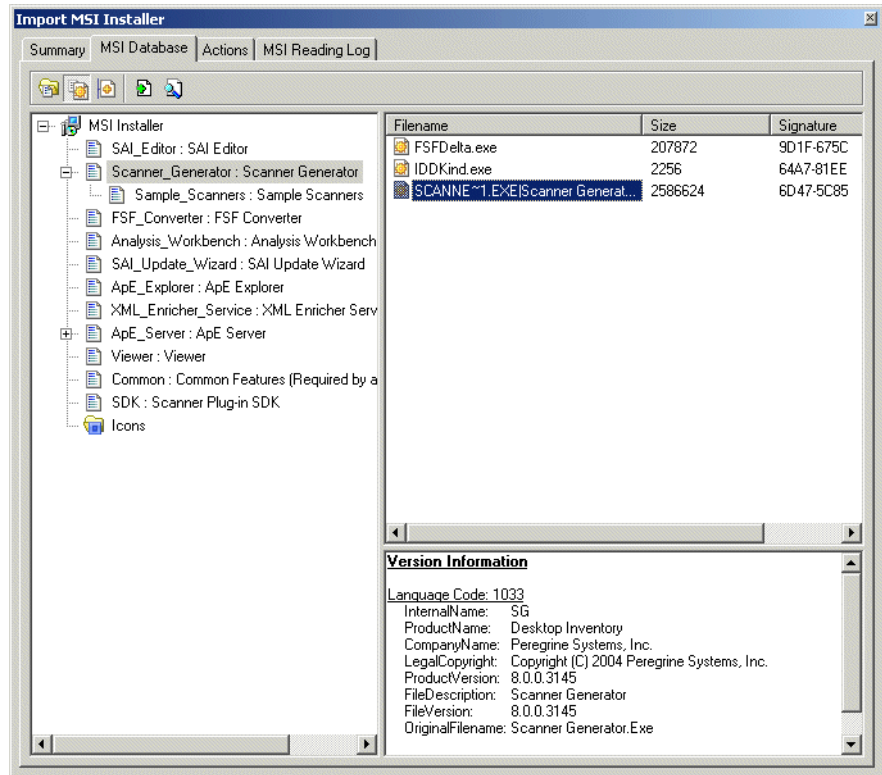
6 Check the **Company Name** and see if it is the same as the **Manufacturer** entry in the **Summary** tab page.

7 You can do this by clicking on the executable file name in the **Files** pane and looking at the **Version Information** displayed in the pane below. In this case the **Company Name** is the same as the **Manufacturer** entry in the **Summary** tab page.
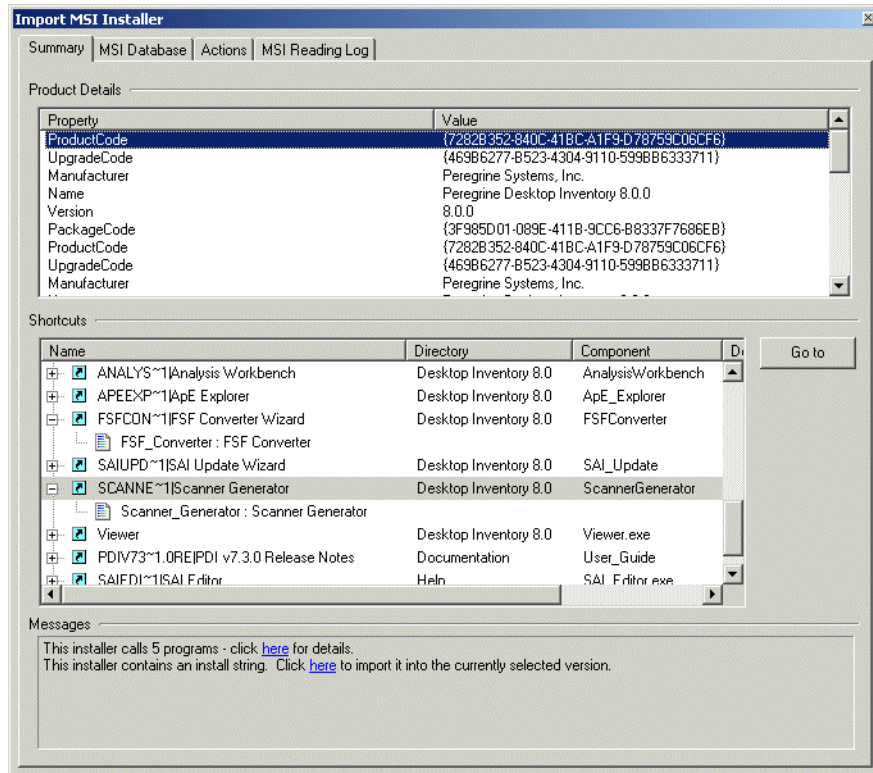
8 However, had it not been the same, you would then add the file as a third
party file. An example is shown below:
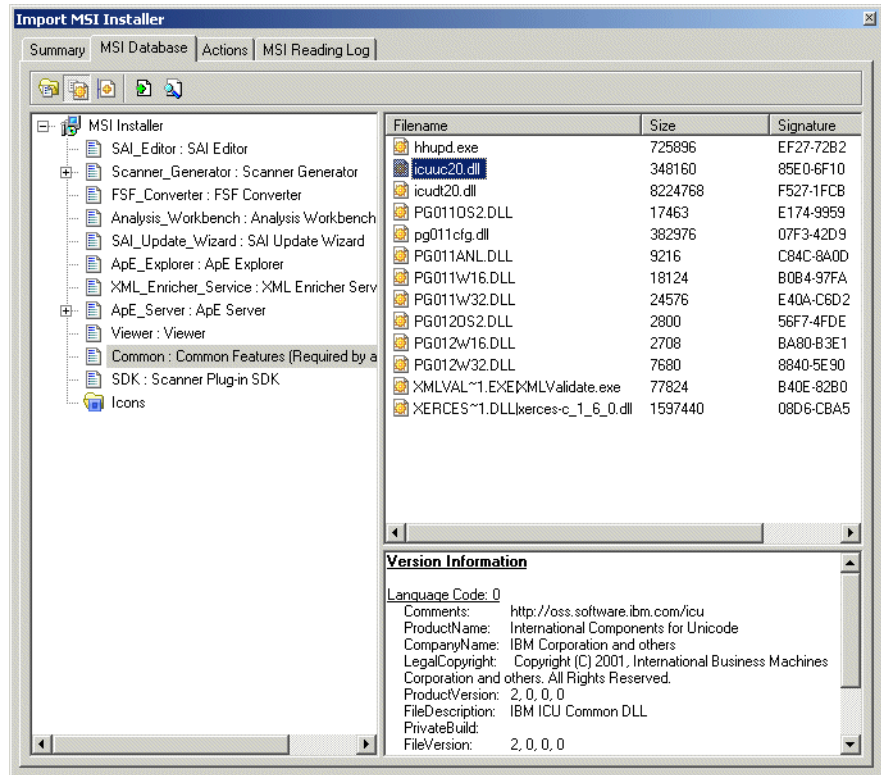
Here we can see that the **Company Name** is IBM Corporation. Hence it is not a file created by Peregrine Systems, Inc. We will need to add this file as a Third Party file in the Peregrine Systems Inc., Desktop Discovery 8.0.0 entry of the ApE Database.

Step 2 Teaching Files to a Version of the Software

To teach files to a version of the software:

1 In the **ApE Explorer**, navigate to the Manufacturer (Peregrine Systems, Inc.).

2 Examine the **Version** entry in the **Summary** tab page of the MSI Importer.

3 Create a new version in the ApE Database if necessary. In this example, we will create an **8** version in the **Peregrine Desktop Inventory** release label.

4 Select the release **8** folder you just created and click the 📑 icon. A **Version Properties** dialog is displayed. Fill in the details and leave the **Install string** blank because it gets imported later.
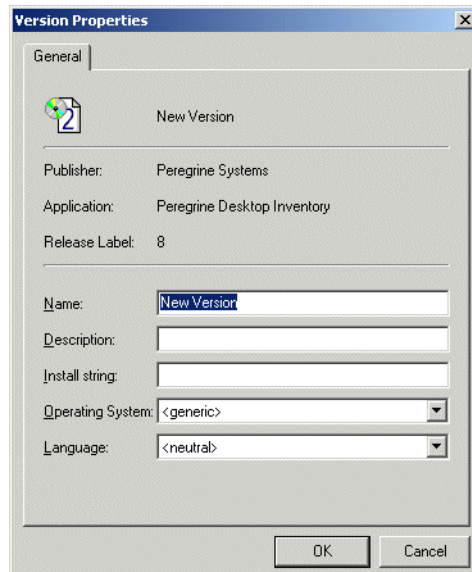
5 Click **OK**. The entry now appears in the ApE Explorer.

6 Go back to the **Summary** tab of the MSI Importer.

Note: Ensure that the version you created in step 3 on page 111 is highlighted in the ApE Explorer.
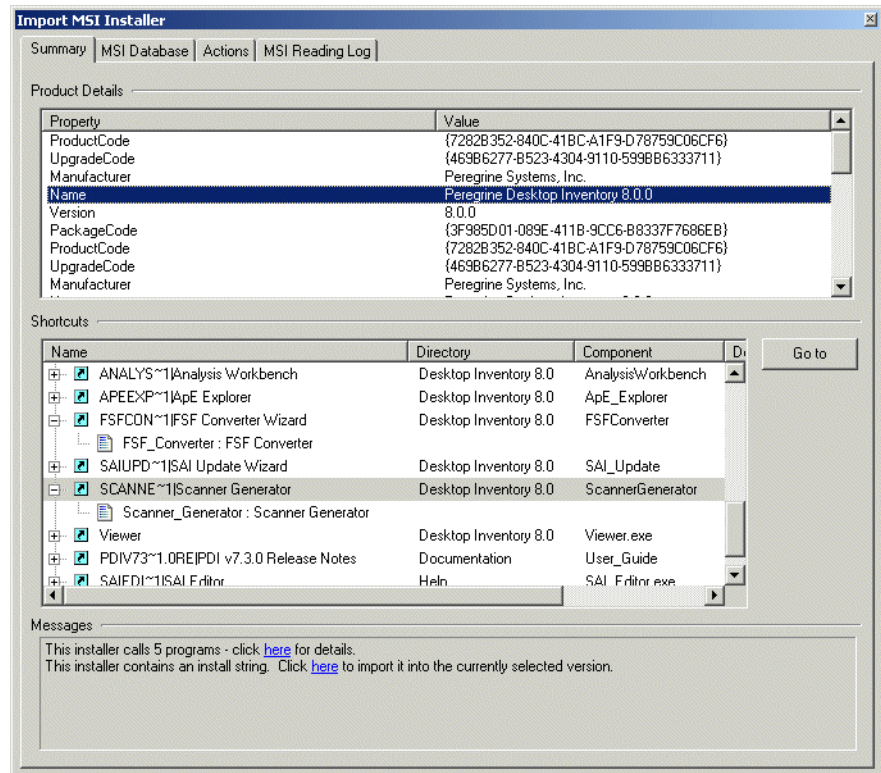
7 Click the **Import...** link at the bottom of the tab page.

The **Version Properties** dialog is displayed and the entry is added to the
Install String field automatically.
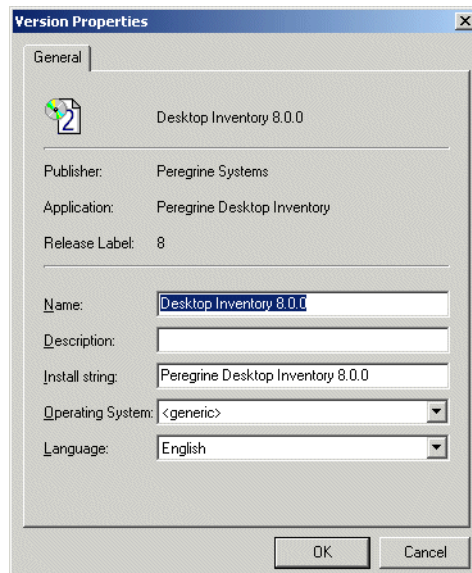
8 Click the **OK** button.

9 Now go back to the **MSI Database** tab page of the MSI Installer.

10 Either:

a Drag the whole feature onto the version node in **ApE Explorer** or drag
individual files onto the version node. You can select multiple files by
using the Shift Key while clicking on the files.

b Right-click on the Feature/Component and choose **Import Files** from the
pop-up menu to add files from that Feature/Component.

c Right-click on the selected files in the **Files** pane and choose **Import Files**
from the pop-up menu.

Note: It is necessary to expand the tree in the ApE Explorer to expose the
version node that you want to add the files to.

If you drag a Component node on to a Version, the files in that node will be
imported. If you drag a Feature node onto a version, the files in that Feature
and all of that feature's descendants will be imported.

The **Files' relationship to Application** dialog is displayed.
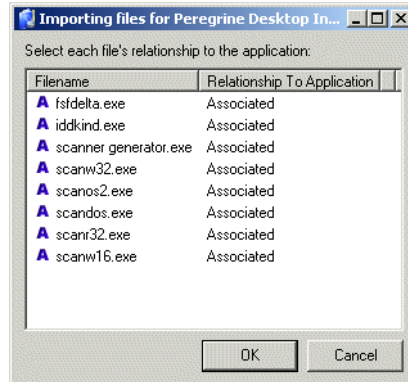
This allows you to select the relationship for each file. There are three ways to change the file's relationship.

- Click the icon to the left of the file to cycle through the relationships.
- With the file highlighted, press the space bar on your keyboard to cycle through the relationships.
- Right-click on a file to use a popup menu to change the relationship.

Note: Main files are generally pointed to by a shortcut. Any files pointed to by shortcuts will be marked as 'Main'. Double check that these files should really be given the 'Main' rating.
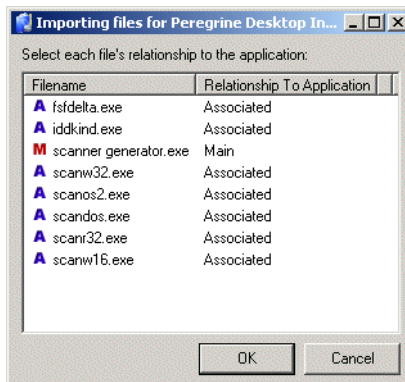
11 To continue the import click **OK.** The files are imported into the ApE Database.

Step 3 Examining Warnings and Messages

1 Examine the warnings and messages by clicking on the links in the bottom pane of the **Summary** tab page.

Messages

This installer calls 5 programs - click here for details.
This installer contains an install string. Click here to import it into the currently selected version.

2 You will be automatically directed to the appropriate place in the **Actions** tab page. You will need to examine the entries associated with the action warnings and decide whether they are files that need to be taught to the application and what relationship they have with the application.

Nested Installer tab

A nested installation is where another MSI resides inside the MSI. You will need to decide several things:

- Is it another separate application with a licence relationship for example?

- Do the files need teaching to the same application as associated files? For example, Live updates would be associated files. An installer for a Live Update component would have added its files taught as 'Associated'.

Called Executables

You will have to examine whether the called executable is going to install some other executable files. For example, look at the command-line column - if it says **install.exe** or **setup.exe** this would indicate that an executable file is being installed. You could get clues from the Name of the Action. A Called Executable Actions could be for example,

Install <name of the executable>.

In the following example, the **Called Exectuables** are not installing other files, they are merely sending commands to run sql scripts.
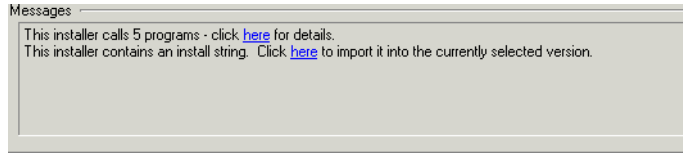
Name	Internal	Command Line	Path
RunApEServSQL	Yes	-u root -e"\. [APE_SERVER]apeserv.sql"	bin
RunUserIDsSQL	Yes	-u root -e"\. [APE_SERVER]userids.sql"	bin
RunDisableRemoteSQL	Yes	-u root -e"\. [APE_SERVER]disableremote.sql"	bin
RunUserPrefsSQL	Yes	-u root -e"\. [APE_SERVER]userprefs.sql"	bin

Nested Installers | Called Executables | DLL Calls | Invoked Scripts |

DLL Calls

This tab lists all Dll calls made.

Invoked scripts

This tab shows any scripts that would be invoked.

3   Once you have examined the files, return to the **MSI Database** tab page.

The MSI Scanner Command Line Utility

In This Section...
- *Overview of the MSI Scanner* on page 117
- *Starting the MSI Scanner* on page 117
- *Opening the MSI Scanner Output File in the MSI Importer* on page 118
- *MSI Scanner Error Level Codes* on page 118

Overview of the MSI Scanner

For situations where the installation media is not accessible, the MSI Scanner can be used. This is a command line utility that takes as input an MSI file. This is then analyzed. The result of the analysis is an XML file containing a summary of the MSI.

In a large enterprise it may not be feasible to have remote offices send their software install media to central locations to be added to the corporation's ApE database. In this case, the MSI Scanner may be of some use.

The MSI Scanner is a command line utility used to scan an MSI based installer, extract all required file information and write an XML file describing the installer and its contents. This XML file can then be sent to the central office where the person maintaining the ApE database can load it into the ApE Explorer exactly as if it was the original MSI based Installer.

Starting the MSI Scanner

Important: The MSI Scanner (msiscanner.exe) is not generated by the Scanner Generator. It is supplied with the software in the following location by default: **C:\Program Files\Peregrine\Desktop Inventory\8.0.0\ApE Explorer**

To start the MSI Scanner:

▶ From the command prompt, type the following:

msiscanner<setup_packge><output_file>

Where:

- <**setup_package**> is the path and filename of the MSI-based installer.
- <**output_file**> is the path and filename of the output XML file. Note that if the specified filename does not end in .xml, the MSI Scanner will append a .xml extension to it. The ApE Explorer does not recognize the output file from the MSI Scanner unless it has an .xml extension.

Opening the MSI Scanner Output File in the MSI Importer

The output from the MSI Scanner is usable in the MSI Importer so that you can browse the MSI and teach from using on the XML file.

To open the MSI Scanner output file:

1 In the ApE Explorer, select the **Import MSI Installer** option from the **Tools** menu.

The **File Open** dialog box is displayed.

2 In the **Files of Type** dropdown box, select the MSI Scanner output file.

3 Navigate to the file to be opened.

4 Click **OK**

The file is opened in the same way as any MSI-based installer. You can import files from it in the same way as described on page 106.

MSI Scanner Error Level Codes

The MSI Scanner produces error level codes which can be used to handle situations if the Scanner terminates without producing a scan file.

These error codes can, for example, be used in a batch file so that specified actions can be carried out in the event that particular error codes are returned.

These can be used to control re-scan activities when a scan has not completed successfully.

Error Level	Description
6	Unexpected error
5	Unable to open the output file
4	Insufficient space available in the **Temp** directory.

Error Level	Description
3	Unable to open input MSI
2	Unrecognized package
1	Incorrect parameters
0	Success

The MSI Scanner Deposit Program

This section provides information about how to send information from the MSI Scanner output to Peregrine Systems so it can be added to the Master Database.

In This Section...

The Scanner Deposit Program Overview

The MSI Scanner is a command line utility that helps Librarians to add applications to the Library.

By the using the MSI Scanner, a remote office can easily send information about a given application to a central location as long as the application was built on the Windows installer. This information can then be added to the Library.

The MSI Scanner can also be used to send application information to Peregrine System's own Librarians. These Librarians maintain the Master database.

Peregrine Systems subscribes to many publishers to obtain application software from them. These subscription services are used to build the Peregrine Win32 and UNIX Libraries.

In some cases, when an application cannot be obtained, customers are asked to send the software media for the application. If this is not possible the MSI Scanner can be used provided the application is built on the Microsoft standard.

The MSI Scanner is executed from the command line and produces an XML base file.

Naming Convention

The application must have been packaged using the Microsoft Standard. Microsoft DameWare is used as an example here. The following naming convention should be used:

<PublisherName>-<ApplicationName>-<Version>.xml

Example:
Microsoft-Dameware-3.51.xml

This ensures the Peregrine Librarian can identify the application quickly before adding it to the Library.

Using the Scanner to Produce the XML File

The MSI Scanner is located in the following default path:

C:\Program Files\Peregrine\Desktop Inventory\<pdi version>\Ape Explorer

If Desktop Inventory was installed to a different path, search for the **MSIScanner.exe** file.

The MSI Scanner syntax is very simple to launch and can be run in a DOS window.

The MSI package file can be located in a different directory to the MSI Scanner but the user running the MSI must have access rights to the file.

Examples of Running the MSI Scanner

MSI File Located in the Same Directory as the MSI Scanner:
msiscanner dntuw.msi Microsoft-Dameware-3.51.xml

MSI File Located in a Different Directory to the MSI Scanner:
msiscanner e:\dameware\dntuw.msi mircosoft-dameware-3.51.xml

MSI File Located on a Network Share:
msiscanner \\<Machinename>\dameware\dntuw.msi
mircosoft-dameware-3.51.xml

In some cases the source path for the MSI file may contain spaces or be named using the long file naming convention. If this is the case it may be necessary to put quotations around the source information.

Syntax for Running the MSI Scanner with Source Information Formatted Using the Long File Name Convention:

Msiscanner "e:\DameWare Installer.MSI" mircosoft-dameware-3.51.xml

Sending the File to Peregrine Systems Librarians

To send a file to the Peregrine Librarians:

1 Create your account using the form on the Peregrine Systems Customer Support Web page.

http://support.peregrine.com/support/nnPRGN/index.cfm?fa=login

You will need to know your company account number.

An account on the Support FTP Server will be automatically created for you. After your account has been created, you will have access to the FTP Server where you will be able to send the XML file.

2 Using the account you created, log into the Peregrine FTP Server using an FTP Client or DOS Window.

Note: This is a Unix System and the creation of all directories is case sensitive.

3 After you have logged in to the FTP Server, create a directory called xmlfile.

mkdir xmlfile

Note: The directory naming must be exactly as written above. Unix is case sensitive.

4 Change to the directory that was just created.

cd xmlfile

5 Type the following commands.

bin
prompt
hash

6 Now you are ready to upload the XML file you created with the MSI Scanner.

mput mircosoft-dameware-3.51.xml

After the file has been placed into the xmlfile directory, a process is launched which monitors the directories created in customer accounts. The Peregrine Librarian is notified when an XML file appears and will add this file to the Peregrine Master Library.

Summary

By sending an application to the Peregrine Librarians in this manner you are achieving three things:

- You are improving the application recognition level within your company's software environment.
- You are helping to keep the Peregrine Master Library up to date.
- You are providing application information to Peregrine without sending the original media.

After the file has been added to the Library, it can be found in the next quarterly release of the Master.sai.