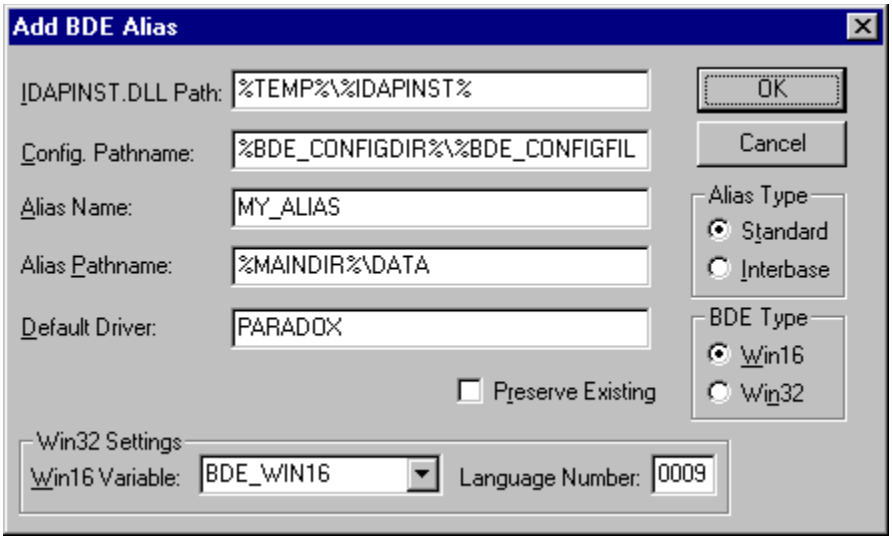
Add Button

This button will add another entry to the list of items.

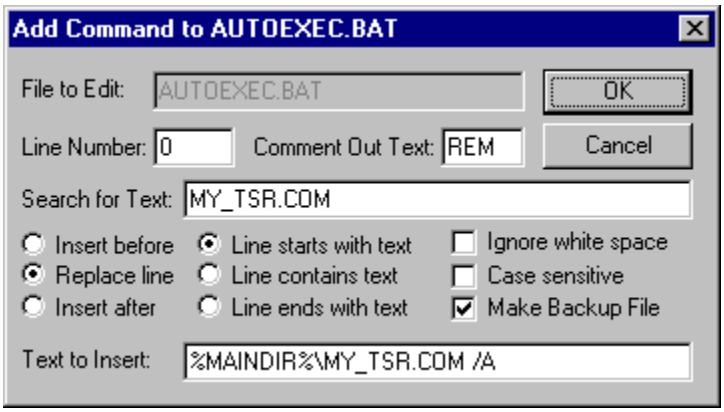
Add BDE Alias

This script item will add an alias to the Borland Database Engine (BDE) configuration. The Borland Database Engine must be installed in the IDAPINST.DLL file and must be installed before this script item is used. Please refer to the sample installation file SAMPLE\BDEALIAS.WSE for an example of adding BDE aliases.



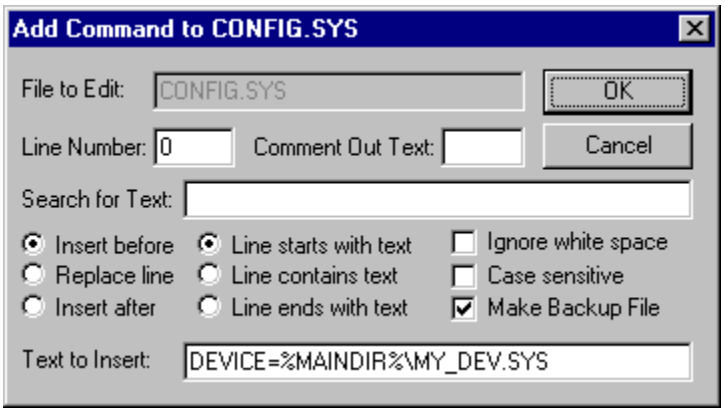
Add Command to Autoexec.bat

This script item will add a command to the AUTOEXEC.BAT file. If the command already exists (it must match the added command exactly), it will not be re-added. A backup file AUTOEXEC.001 is created.



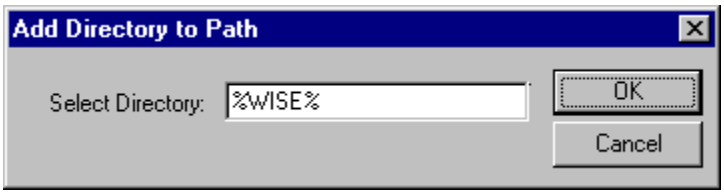
Add Command to Config.sys

You can add commands to the CONFIG.SYS file using this script item. A backup file (CONFIG.001) will be created. If the command already exists in the CONFIG.SYS, it will not be re-added.



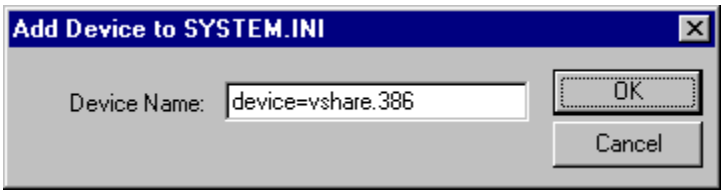
Add Directory to Path

You can add a directory to the PATH environment variable with this script item. A backup copy of the AUTOEXEC.BAT will be made. (AUTOEXEC.001) If the directory is already in the path, it will not be re-added.



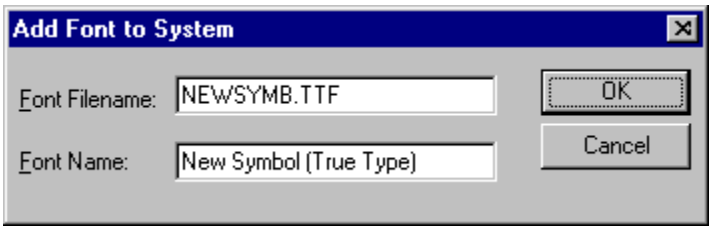
Add Entry to System.ini

This script item will add an entry to the [386Enh] section of the SYSTEM.INI file. A backup file (SYSTEM.001) will be created if changes are made. You should enter the full name of the line to add (i.e. device=vshare.386). If the line already exists, it will not be re-added.



Add Font to System

This script item will add a font to your system. You must not specify a full pathname. The font file must have already been copied into the Windows System directory.



Add Text to INSTALL.LOG

This script item will add a line of text to the installation log file. You can use this script item to add comments to your installation log or to add the names of files that should be removed during the uninstall.

AddFile

This button will add the file selected on your machine to your installation. The file will be placed in the currently highlighted component (or directory).

Add Folder

This button will add the directory selected on your machine to your installation. The file will be placed in the currently highlighted component (or directory).

Installation Log Text

This field holds the text that will be added to the INSTALL.LOG file.

Add Command To End of File

Normally the command will be placed at the start of the file. Checking this box will cause the command to be appended to the end of the file.

Adding Support for SHARE

File sharing capabilities are required by many programs. There are two ways of adding support for SHARE to Windows: 1) via the DOS SHARE.EXE program and 2) via the Windows VSHARE.386 file. It is recommended that the Window VSHARE.386 method be used to add support for SHARE. A copy of VSHARE.386 is included with Wise in the ADVANCED sub-directory. The following steps are required to add support for SHARE:

1 Check if SHARE is already running (use the Check System Capabilities script item), if it is, skip the following steps.
2 Check if the VSHARE.386 file is already in the %SYS% directory. If it is not, install the VSHARE.386 file to %SYS%. (use the Check If File Exists script item and the Install File(s) script item).
3 Add the line device=vshare.386 to the SYSTEM.INI file. (use the Add Device to SYSTEM.INI file)

The Installation Expert has a checkbox (Support File Sharing via SHARE) that will add the appropriate script items to your installation script.

Adding Uninstall Support

The Wise Uninstall program (UnWise.EXE) is located in the same directory as the Wise program. You can freely distribute this program in your installation executables. This program reads the INSTALL.LOG file and uninstalls the program/data files that were installed. To add an uninstall option to your installations:

1 Copy the UnWise.EXE file from the Wise directory to the Applications directory on the destination system.
2 Create an icon with the command name of %MAINDIR%\UNWISE.EXE and a parameter of your installation directory and the INSTALL.LOG name as in: %MAINDIR%\INSTALL.LOG
3 You can add an optional second command line parameter to the UNWISE command line, the title of the window displayed during the installation. You should place the title in double quotes.

You may rename the UNWISE.EXE program to anything you like. There are two optional switches that you may place at the beginning of the UNWISE command line: /A or /S. The /A option will run the uninstall in automatic mode without prompting the user. The /S command line switch will run the uninstall in silent mode, nothing will be dispayed to the user as the software is being removed. This can be useful for a network administrator to silently remove packages from networked computers. Only one of these command line switches may be used at a time and they must be placed at the start of the command line.

Abort Installation

If this button is checked, the installation will be aborted after the message is displayed.

Append

Check this button to append the value to any existing value of the variable.

Start Block

If this button is checked, a new block will be started if the given check is true.

Description

This fields holds the text of the message that will be displayed if the check is true.

System Check

This list holds the possible items that can be checked on the destination system.

Command

This field holds the command that will be added to the file. You should use full pathnames when referencing files.

Component List

This field holds the list of components, one per line, that will be installed. For each component you list, you should add an If script item that checks for a letter (A-J) to be contained in the destination variable.

Destination Variable

This variable will receive a list of letters that correspond to the selections made by the user. If the user selects the first component, the letter 'A' will be added to this variable. You should use the If-Contains script item to create a block that copies all of the files to each of the components.

Default Value

This field holds the default value of the dialog box.

Description

This text will be displayed in the dialog box. It should explain the actions that the user is to perform to fill out the dialog box.

Device Name

This field holds the command that will be added to the [386Enh] section of the SYSTEM.INI file. This command normally starts with the device= prefix.

Directory Name

This field holds the name of the directory to add to the PATH environment variable in the AUTOEXEC.BAT file.

Disk Variable

You should place the name of the variable that holds the pathname where your software will be installed. The free space on the drive that holds that directory will be used in the display of the free space left.

Environment Variable Name

This field holds the name of the environment variable to read.

File Offset

This is the number of bytes (starting at 0) from the beginning of the file to write the data to.

Font Name

This field holds the full name of the font. This name will be added to the Fonts section of the WIN.INI file.

Font Filename

You should enter the file name of the font file (TTF or FON) that you want to add to Windows. The file must have been already copied to the System directory.

Compare Type

This field selects the type of comparison that is done. You should select whether the system should or should not have the selected feature for the action to be performed.

INI Item Name

This holds the INI entry name to be read from.

INI Pathname

This field holds the pathname of the INI file to read.

INI Section Name

This field holds the name of the INI file section (without the []'s) to read.

Advanced Script Items

This list contains a list of advanced script items that you can add to your installation script. Select an item from this list and press the OK button.

Display Message Only

If this button is selected, only the message will be displayed if the system meets the selected criteria.

Append Null Terminator

If this box is checked, a null byte (a zero byte) will be appended to the string that is written to the data file.

Variable Operation

As the variable is being set, the following operations may be performed on it:

- Increment: add one to a numeric variable
- Decrement: subtract one from a numeric variable
- Remove trailing blackslashes: format a variable to be a valid directory name by removing any trailing backslashes
- Convert to Long Filename: converts an **existing** pathname to a long filename. If you are not running under Windows NT or Windows 95, this operation keeps the pathname unchanged. If a filename is specified, the current directory will be used.
- Convert to Short Filename: converts an existing long pathname to it short alias pathname. If you are not running under Windows NT or Windows 95 the pathname will remain unchanged. If a filename (and not a full pathname) is specified, the current directory will be used.
- Convert to Uppercase: converts the value to all uppercase letters.
- Convert to Lowercase: all of the characters in the value will be changed to lowercase letters.

Prompt Name

This is a short description of the information that the user should enter. This will be displayed directly below the window title.

Radio Button List

This list holds the labels for the radio buttons, one per line, that will be displayed. The destination variable will contain a single letter (A thru J) representing the radio button that was selected.

Read Me Pathname

This field holds the pathname on the destination system that holds the text to be displayed in the readme dialog box.

Read or Write File Data

These radio buttons select whether the data is read into the variable from the binary data file or written into the data file from the variable.

Win32 Root

If you are reading the Win32 registry, you should select the proper registry root to read the key from using these radio buttons. If you are reading the Win16 registry, you should leave the HKEY_CLASSES_ROOT selected.

Value Name

If you are reading a named value from the Win32 registry, you should enter the value name into this field. If you are reading the Win16 registry, this field should be left blank.

Registration Database Key

This field holds the registration database key that will be read.

Remove Filename

By checking this box, any filename that is at the end of the pathname will be removed. For example, if the variable has the value C:\WISE\WISE.EXE the new pathname would be C:\WISE. NOTE: The default value will not have any file names removed.

Description

This text will be displayed on the screen while the search is taking place.

Search Filename

You enter the name of the file that you want to search for into this field. No wildcard characters are allowed (*, ?).

Search Directories

This field holds a list of directories to search for the specified file. The directories in this list must be separated by semi-colons. If this field is left blank, the value of the PATH environment variable is used.

Company Name

This field will be displayed next to the edit box that will hold the company name entered by the user.

Name Prompt

This field holds a short description that will appear next to the field where the name will be entered.

Serial Number

This field holds a short description of the serial number field.

New Variable Value

This field holds the new value of the variable.

Variable Name

This field holds the name of the variable that will receive the results.

Window Title

This field holds the title of the dialog box that will be displayed. This field must be filled in for each language that your script uses.

Read/Write File Pathname

This field holds the pathname for the file that you will reading from or writing to. The file must already exist and must contain space for the variable data.

Read/Write Variable

The value in this variable will be written to the binary file or read from the binary file into the variable.

Allow Floppy Disk Change

You should use this script item if you need to insert a floppy disk other than the current installation floppy. This script item will temporarily close the installation EXE. When the installation EXE is required next (i.e. for an Install File or Display Graphic) the user will be prompted to re-insert the installation floppy, if it has not already been inserted.

Append New Items

Normally, new script items are inserted before the currently selected script item. You can check this box to have the new script items inserted after the currently selected script item.

Make Floppy

Press this button to create your installation and copy it to floppy disks. You can then place these disks into another computer to install your software.

Run Installation

Press this button to run your new installation from your local hard disk. The installation will be run and files will be installed to your computer.

Test Installation

Press this button to test your new installation. The installation will be run, but no files will be installed, and no other changes will be made to your computer.

Archive

This checkbox will set the archive bit for the selected files. The archive bit is used by many backup programs to determine which files are backed up in an incremental backup.

Hidden

Checking this box will make all of the selected files hidden.

File Pathname

This field contains the filename or wildcard pathname of the files to change.

Read-Only

Checking this box will make all of the selected files read-only.

Scan Sub-directories

You should check this box to search for all of the matching files in the directories below the search directory.

System

Checking this box will make all of the selected files system files.

BDE Add Alias

Some Borland Databases require an alias to be added before the database can be referenced. The IDAPI.CNF file that ships with WISE supports both Paradox and DBase alias'. If you require another type of alias, you may need to contact Borland for the proper CNF file.

Alias Name

This field holds the name of the BDE alias to add/update.

Alias Pathname

This field contains the pathname to the database directory for the alias.

Alias Type

These radio buttons select which type of BDE alias to add, standard (DBASE/PARADOX) or Interbase (SQL).

Configuration Directory Variable

When the IDAPINST.DLL field is empty, this variable receives the default directory for the IDAPI.CFG file. When the IDAPINST.DLL field is filled, the IDAPI.CFG file will be created in the specified directory.

Configuration Filename

If an existing IDAPI configuration file exists, its name will be placed into this variable. The filename in this field will be used to create a new IDAPI configuration file.

Configuration Pathname

This field holds the BDE configuration file pathname. The pathname is located in the IDAPI section of the WIN.INI file as the entry CONFIGFILE01.

Confirm IDAPI Directory

Check this box if you want to prompt the user for the location of the Borland Database Engine files. Otherwise, the files will be installed into the default location, C:\IDAPI or the location of any existing installation.

Default Driver

When adding a standard alias, this field holds the default driver for the databases, either PARADOX or DBASE. If an Interbase alias is being added, then you must specify the Server Name and User Name required to access the databases.

BDE Directory

This must point to the directory where DELPHI (both 16bit and 32bit) reside. Wise will pull the standard setup files from that tree.

BDE DLL Directory

This variable receives the default location for the BDE DLLs. This is normally C:\IDAPI or the existing location of the BDE DLLs.

Borland Database Engine Type

These radio buttons select whether BDE for Win16 or BDE for Win32 is being installed.

Error String

If an error occurs during the configuration of BDE, a string describing the error will be placed in the specified variable. This variable may be displayed to the user with the Display Message script item.

Existing Configuration Pathname

The variable in this field will receive the full pathname of the existing IDAPI.CFG file (or ODAPI.CFG). If no existing IDAPI configuration file exists, the value of variable will be blank.

IDAPINST.DLL Pathname

This field holds the pathname of the IDAPINST.DLL file. This file must be installed for BDE to be configured. This file is normally placed into the %TEMP% directory and removed at the end of the installation. If this field is blank, only the default values for the installation directories will be retrieved only.

IDAPINST.DLL Pathname

This field holds the full pathname for the IDAPINST.DLL file. The file IDAPINST.DLL must be installed to add an alias to BDE. This file is normally placed in the %TEMP% directory prior to adding aliases. The file is normally deleted at the end of the installation.

BDE Installation Type

This section controls which Borland Database Engine files will be installed. It is strongly recommended that a full installation is performed. If a partial installation is done, a proper installation may not be performed on computers that contain a complete installation of an older version of the Borland Database Engine.

BDE (Borland Database Engine)

This dialog controls the installation of the Borland Database Engine (BDE) for Win16 and Win32. It is strongly recommended that you perform a full installation of BDE 32. A partial installation will require that your users only run a single BDE application at a time. You should select the directory on your computer where the Borland Databaes Engine files are currently located (normally C:\IDAPI for BDE 16). You can also add a BDE alias by entering it's name and the pathname of the directory (starting with a variable reference) that the alias should point to. For more information about installing BDE please refer to your Borland user manual.

Language Driver Directory

This field holds the directory where the BDE Language Drivers are installed on your computer.

Language Driver Directory

This field holds the variable that will receive the pathname where all of the BDE language driver files will be placed.

BDE Language Drivers

This section controls which Borland Database Engine language files will be installed.

Preserve Existing

Normally, if an alias already exists with the specified name, it will be overwritten. You can check this box to not update an alias if it already exists.

BDE Source Directory

This field holds the pathname of the Borland Database Engine files on your computer.

IDAPI Configuration Template

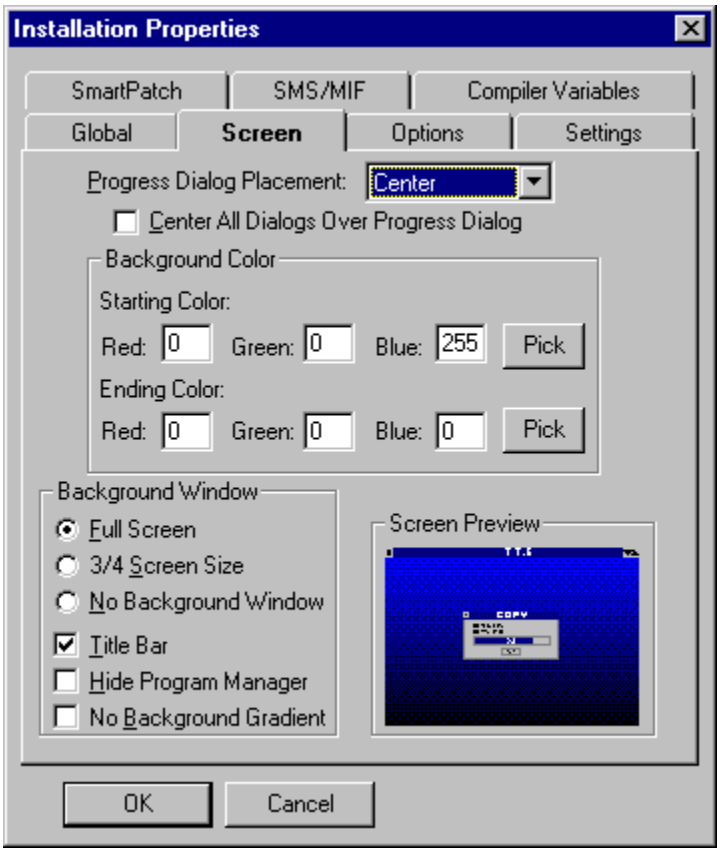
The IDAPI.CNF file is used during the installation to provide default values for BDE. This file must be installed prior to BDE configuration. This file is normally placed in the %TEMP% directory and removed at the end of the installation.

Background Processing

If this box is checked, your system will be able to process background tasks during the compile process. This will slow down the compile process by 50%. If this box is not checked Windows will not respond to user input during the compile.

Screen Settings

This dialog lets you configure the background window that is displayed during the installation. You can set the color, gradient, and size of the background window.



Background Uses 256 Colors

The background gradient colors normally use the default 16 colors Windows provides. You can check this box for the background to use 96 palette entries to display the background gradient on computers that support this. You should not use this option when placing 256 color bitmaps on the background. The background gradient will appear distorted if your 256 color graphic has more than 120 unique colors in it.

Language Number

This field holds the language number of the installed language resource. This field should be set to 0009 for English.

Partial BDE Installation

This box if you are installing a subset of the full BDE installation.

Win16 Variable

This variable is blank if the BDE configuration file supports only Win32 applications. If you will be running both Win16 and Win32 BDE applications, you should set the variable to A.

Beep When Prompting for New Disk

If this box is checked, the computer will beep when a new disk is needed.

Font Weight

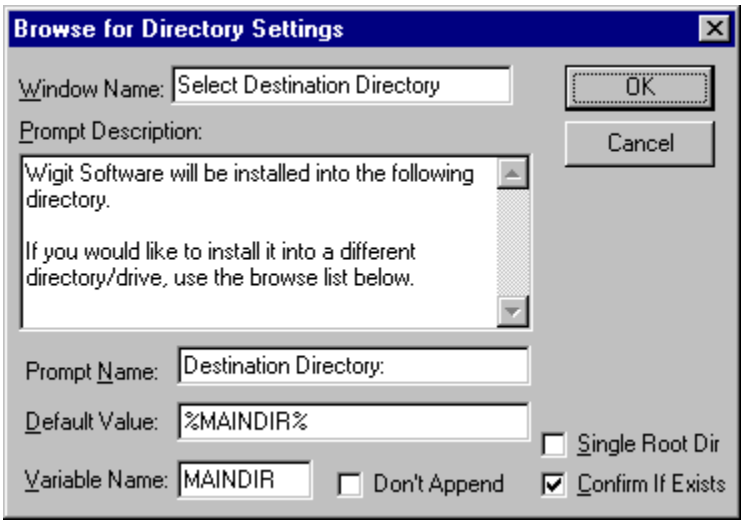
Normally, all messages and dialog boxes use a bold font to display text. Windows 95 now uses non-bold fonts as the default. You can use these radio buttons to select whether the fonts displayed will be bold or light in message and dialog boxes.

Browse Button

Pressing this button will allow you to browse the files and directories on your disks to select the proper source files and directories. If you are not sure what the file and/or directory names are, press this button.

Browse Directory Tree Prompt

This script item will get a pathname into a variable. The dialog will display the directory tree and allow the user to select the drive and pathname to install to. If you place a full treename into the default, it will be the default directory. If you do not enter a full pathname, the root will be the current directory.



Browse Program Manager Groups

This script item will display a dialog that lists the existing Program Manager groups, and allows the user to select one.

Single Root Dir

If your default directory includes sub-directories (i.e. C:\PROGRAM\MYAPP), you can check this button to only include the final directory name (i.e. MYAPP) when the user is browsing directories.

Cancel Button

This button cancels the changes you are about to make. Any changes that you have made to THIS dialog box will be gone when you press this button.

Center Dialogs Over Progress Dialog

If you check this box, all dialogs and message boxes will be centered over the same part of the screen as the progress dialog. This is useful if you do not want any graphics located on one portion of the background to be covered during the installation.

Center Horizontal

Checking this button will cause the graphic image to be centered on the screen left to right.

Center Vertical

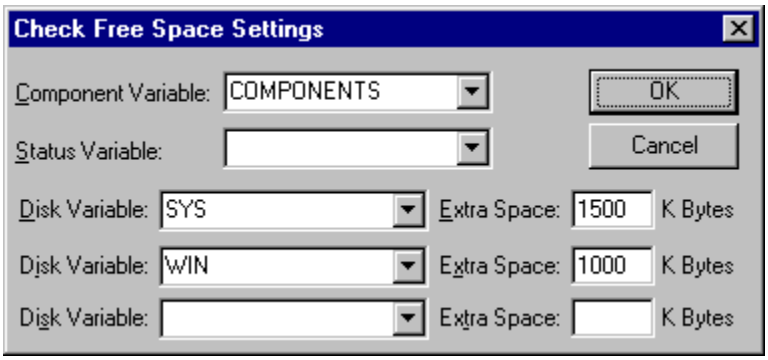
This button, when checked, will cause the graphic to be displayed centered on the screen vertically.

Changing Source Directories

If the directories that contain the files in your installation change, you can use this dialog to modify the source directories for your installation script. To change a source directory, select the source directory from the list, and modify its value in the editbox. If the Change Sub-Directories box is checked, all of the directories below that directory will also be changed. You can access this command from the Source Directory item in the Edit menu.

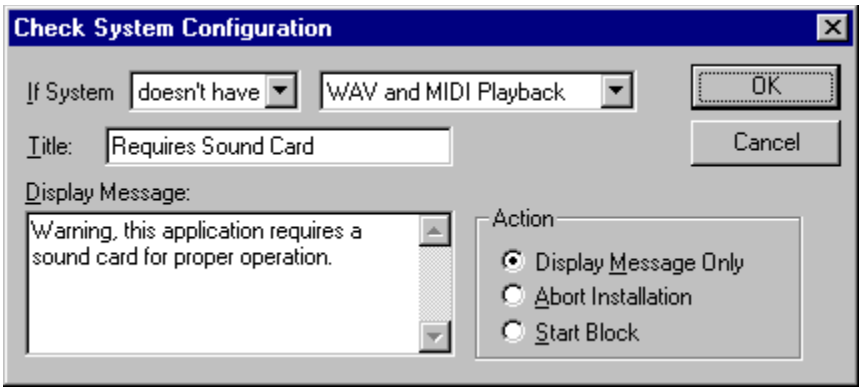
Check Free Disk Space

This script item should be used to check that enough free space is available on the destination computer for the installation to succeed. You can also indicate how much free space should be on the disks after the installation finishes.



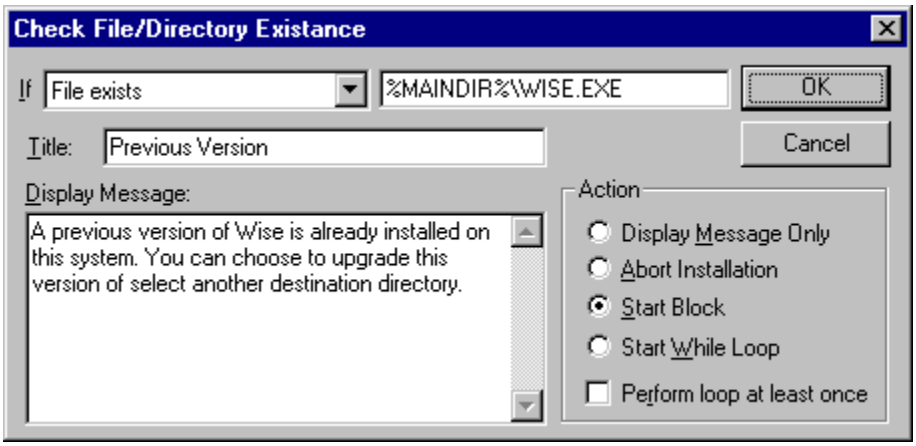
Check System Configuration

You can add this script item to check the capabilities of the system your software is being installed on.



Check If File/Dir Exists

This advanced script item can be used to determine if a certain file exists on the destination computer system. This can be useful in network installations or when detecting previous versions of software.

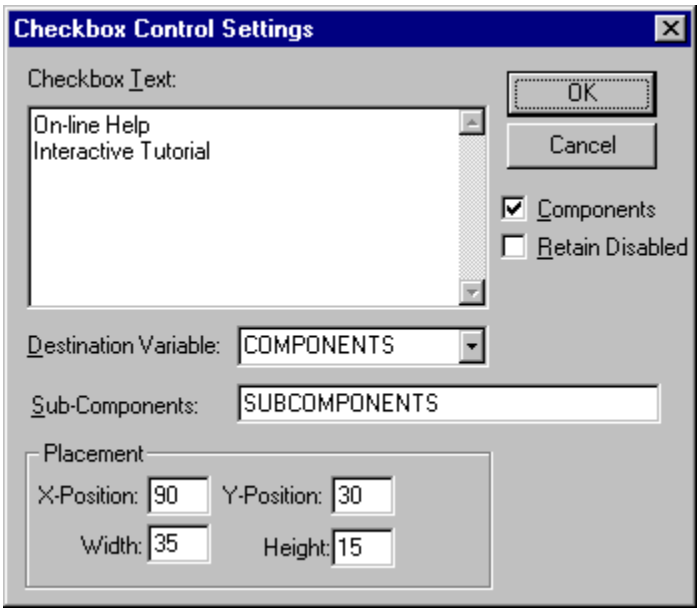


Check Disk Free Space

This checkbox controls when the amount of free space on the disk drive selected in the prompt will be checked. If the total size of the files on the installation is larger than the amount of free space on the destination drive, an error will be reported. This disk space check does not take components or mulitple destination directories into account. You should use the Check Disk Space advanced script item for advanced disk space checking.

Checkbox Control

The checkbox control should be used to present the user with yes/no options. For each option that they check, the corresponding letter (A for the first checkbox, B for the second, etc.) will be appended to the variable.



Component Variable

If you are performing a component based installation, place the component variable into this field. The disk space check will only check the files that will be installed based on the components selected by the user. The component script item must appear before the check disk space item.

Exists

This list selects whether the action is performed if the file exists or does not exist on the destination computer.

NOTE: The "Module Loaded in Memory" option will only find Win16 DLLs that are loaded in memory. The Win32 DLLs that are loaded by other applications are not visible to the installation application.

File Pathname to Check

This field holds the pathname on the destination computer to search for. The pathname should begin with a variable. No wildcard characters are allowed in the file pathname.

Extra Space

You should enter the amount of K bytes that should be free on the given disk after the installation of your software.

Status Variable

Normally, the Check Disk Space script item will display an error when there is not enough disk space. When this error is displayed, the user can choose to abort the installation, ignore the error, or retry the disk space check. If this field is left blank, the retry operation will be performed in this script item. If you set the field to the name of a variable, the variable will be set to R if the user selected to retry the disk space check. You can use the value of the variable to loop back to a Browse for Directory script item to allow the user to select another destination drive.

Extra Space Variables

You can add up to three disks variables that must have extra space available at the end of the installation. This will ensure that space is available for printing, temporary files, or save files.

Clear Button

This button will delete all script items. Use this button with care!

Starting and Ending Colors

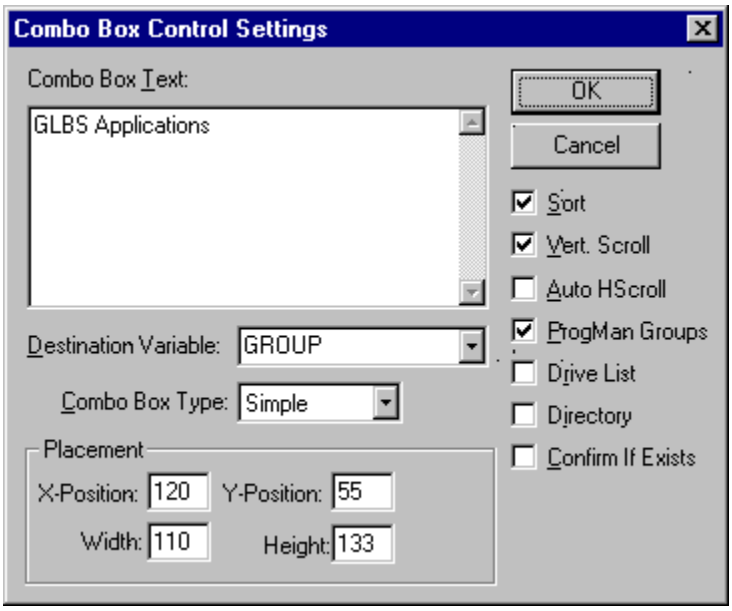
You can set the starting color (at the top of the screen) and ending color (at the bottom of the screen) on the background screen. The numbers you place in the boxes indicate the amount of red, green, or blue in the background. The numbers in each box can range from -255 to 512. The smaller the number the less of that color will appear, the larger the color, the more will appear.

Pick Color

This button will display a dialog that will allow you to graphically choose a color for the background starting or ending color.

Combobox Control

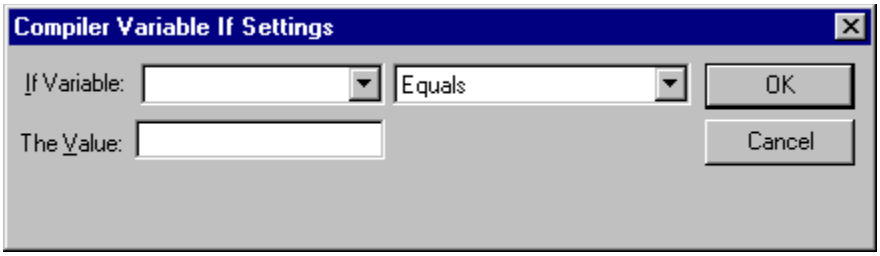
The combobox control combines an edit text control with a listbox control. This control can be used to present the user with a list of possible choices and also allow them to edit the returned text as well.



Compiler Variables IfBlock

This script item works exactly the same as a standard IF Block, however, it will only accept compiler variables.

You need to define the compiler variables in the Installation Properties/Compiler Variables property page. For example, if you created a single installation that installs both the 16bit and 32bit versions of your program, you could use a compiler variable to determine what type of installation you would be creating, and choose only to include the required files for that platform.



Prompting for Compiler Variables

In some cases (such as testing) you may not want to prompt for compiler variables. You can also select in exactly what situation you would like to prompt for a value for a compiler variable.

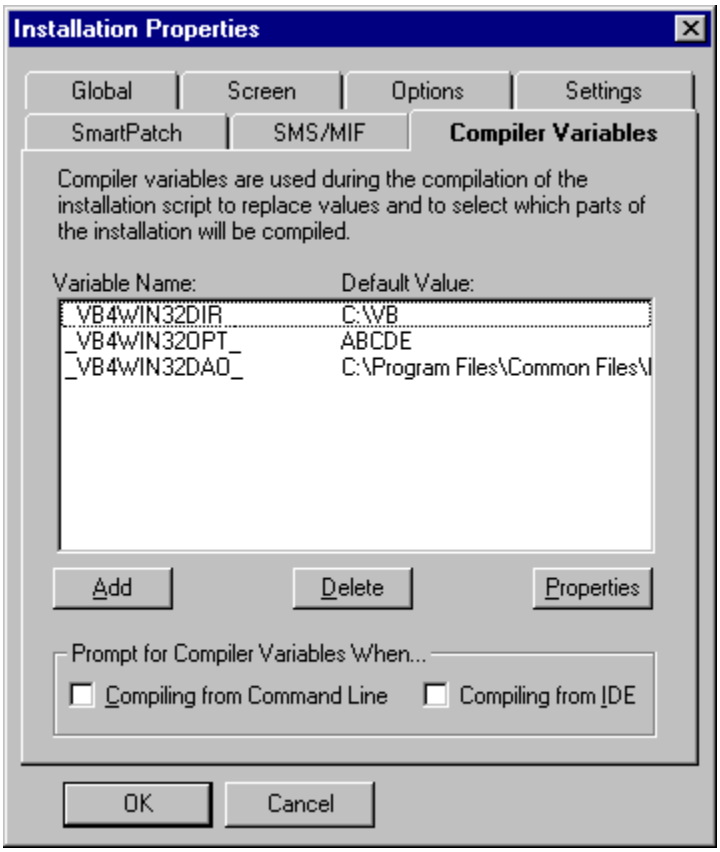
Prompting for Compiler Variables

In some cases (such as testing) you may not want to prompt for compiler variables. You can also select in exactly what situation you would like to prompt for a value for a compiler variable.

Compiler Variables

You may use compiler variables within your script to have conditional compilations or to replace information that may change from one compile to another.

Compiler variables are used exactly like an #ifdef in a C/C++ type programming language. These variables will look exactly like standard WISE variables, except they need to be defined in this dialog. These variable will be replaced at compile time and not at runtime. Compiler variables can be used to create conditional compiles of an installation for deployment or testing.



Compiler Variables

WISE uses a standard of an underscore ("_") as the first and last character of a compiler variable to make it simpler to determine which type of variable it is. Once a variable has been defined you may set a default value to be used. You may also specify when you want to prompt for the value. For example, during testing (compiling from the IDE) you may not want to be prompted every time for each compiler variable. However, when you are creating a final installation (either from the IDE or from the Command Line) you may want to prompt the user for the values.

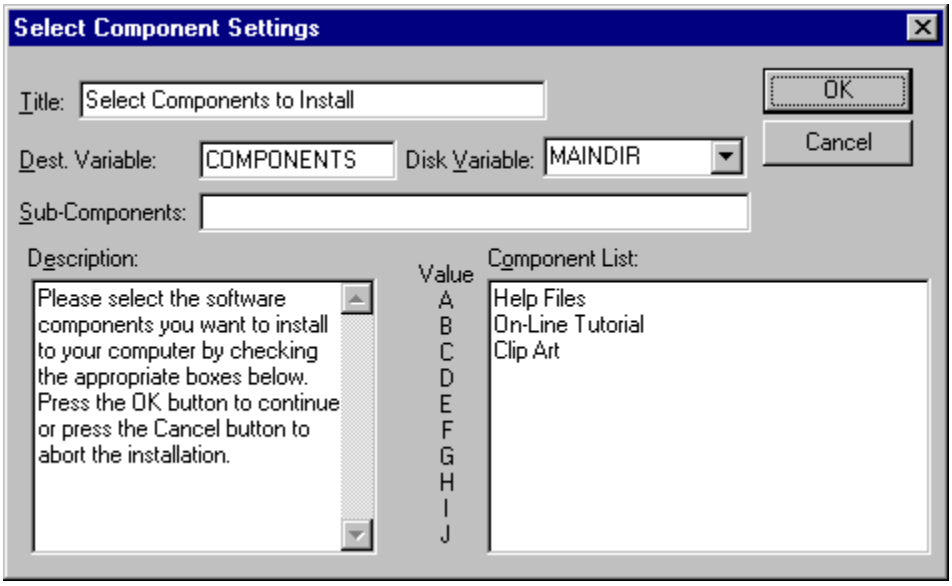
Select Components

This script item will allow you to prompt the user for the components of your software they would like to install. The destination variable will be filled with a list of the letters of the components that the user chooses. You should use the If variable Contains script item to copy the proper files.

To have some of the checkboxes initially checked when the dialog is displayed, use the Set Variable script item to set the value of the components variable to the letters that correspond to the boxes you want checked. For example, if you want the first and third checkboxes initially selected, you use the script item:

Set Variable COMPONENTS to AC

You must use uppercase letters in the set variable script item and the script item must preceed the components dialog script item.



Sub-Components

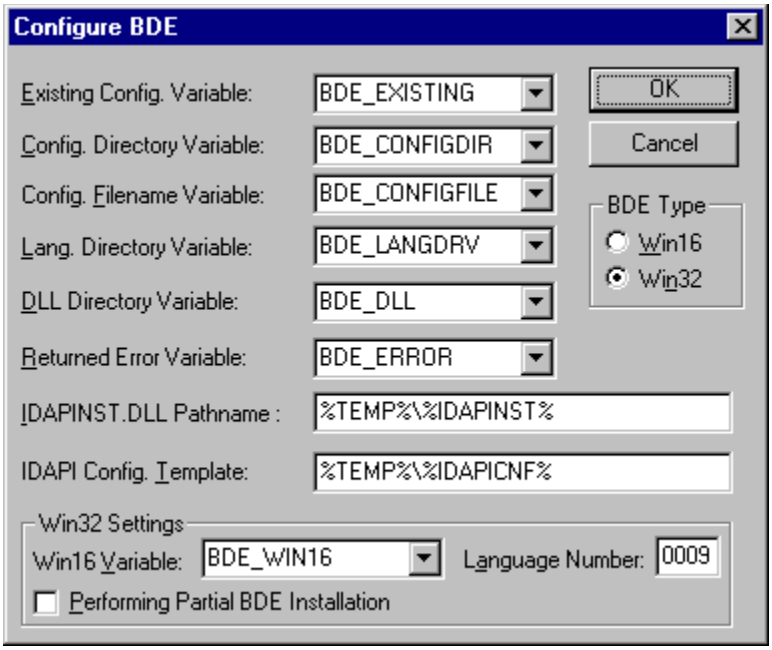
This field holds a list of sub-component variables that are used by the main component variable. You should list these variables separated by commas.

Maximum Compression

This selects a higher compression ratio for files that are being placed into the installation executable. If you check this box, it will take longer to create the installation executable. The resulting installation executable will be slightly smaller.

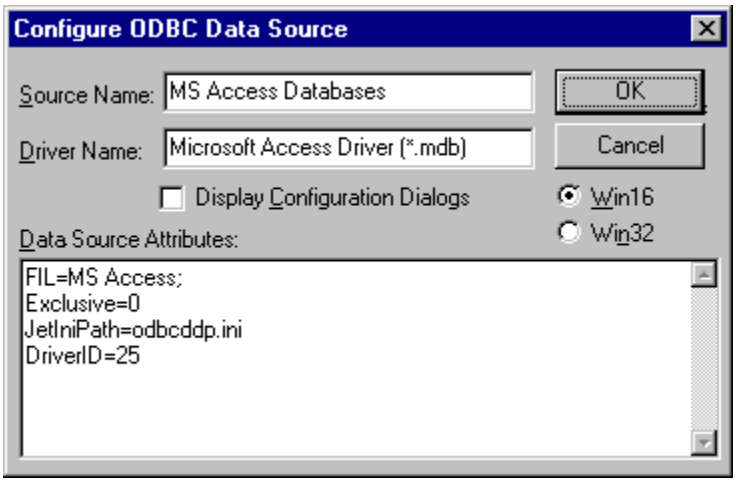
Configure BDE

The script item configures the Borland Database Engine during the installation. If the IDAPINST.DLL Pathname and IDAPI Config. Template are blank then this script item will retrieve the default values from the installation files and directories. The script item will start a block if the configuration succeeds. Please refer to the sample installation script SAMPLE\BDEINSTL.WSE for an example of a BDE installation.



Configure ODBC Data Source

You can configure ODBC data sources with this script item. The ODBC driver must already be installed and all driver files must have been installed.



Confirm If Directory Already Exists

If you check this checkbox, and the installer specifies a directory that already exists, they will be prompted if they want to continue. If they answer no, they will be returned to the prompt dialog.

Confirmation Text

This field contains the text that will be displayed to confirm the users choices. If this field is blank, no confirmation is done.

Wise Installation System

Introduction

The Wise Installation System creates and edits Windows Self-Installing Executables. It uses an installation script to determine which files to install and where the files are copied to. The resulting executables can be directly uploaded to a BBS or other On-Line service. The end-user does not need any software (other than Windows itself) to install the Wise executable. When they run the program under Windows, the program extracts the files from itself and installs them onto the client system.

Wise Installation System

Introduction

The Wise Installation System creates and edits Windows Self-Installing Executables. It uses an installation script to determine which files to install and where the files are copied to. The resulting executables can be directly uploaded to a BBS or other On-Line service. The end-user does not need any software (other than Windows itself) to install the Wise executable. When they run the program under Windows, the program extracts the files from itself and installs them onto the client system.

Wise Installation System

Introduction

The Wise Installation System creates and edits Windows Self-Installing Executables. It uses an installation script to determine which files to install and where the files are copied to. The resulting executables can be directly uploaded to a BBS or other On-Line service. The end-user does not need any software (other than Windows itself) to install the Wise executable. When they run the program under Windows, the program extracts the files from itself and installs them onto the client system.

Wise Installation System

Introduction

The Wise Installation System creates and edits Windows Self-Installing Executables. It uses an installation script to determine which files to install and where the files are copied to. The resulting executables can be directly uploaded to a BBS or other On-Line service. The end-user does not need any software (other than Windows itself) to install the Wise executable. When they run the program under Windows, the program extracts the files from itself and installs them onto the client system.

Wise Installation System

Introduction

The Wise Installation System creates and edits Windows Self-Installing Executables. It uses an installation script to determine which files to install and where the files are copied to. The resulting executables can be directly uploaded to a BBS or other On-Line service. The end-user does not need any software (other than Windows itself) to install the Wise executable. When they run the program under Windows, the program extracts the files from itself and installs them onto the client system.

Wise Installation System

Introduction

The Wise Installation System creates and edits Windows Self-Installing Executables. It uses an installation script to determine which files to install and where the files are copied to. The resulting executables can be directly uploaded to a BBS or other On-Line service. The end-user does not need any software (other than Windows itself) to install the Wise executable. When they run the program under Windows, the program extracts the files from itself and installs them onto the client system.

Controlling the Install via DDE

The programs that are executed via the Execute script item can establish a DDE link to the installation program using the program name WiseInst. The programs can then do one of the following:

- Send a DDE request with a topic name of the name of a variable to read. The value returned is the variable value. The item name is not used.
- Execute a DDE command with a topic name of a variable to set/create. The command string should contain the value of the variable. The item name is not used.
- Execute a DDE command with a topic name of Cancel. This will cause the installation to be aborted.

The example program VBDDE in the DDE subdirectory demonstrates using a DDE link to read and set run time installation variables.

Convert CD-ROM to Floppy Install

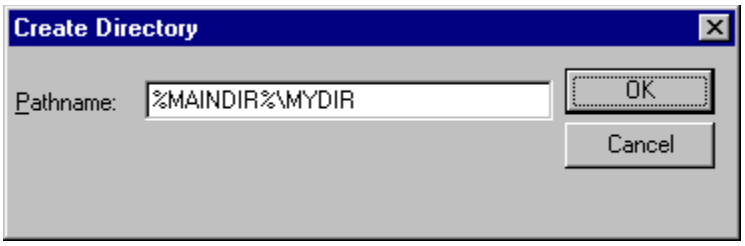
If you have created an installation that uses Copy Local File(s) to install files from a CD-ROM, you can use this checkbox to create a floppy based installation. When this box is checked, all Copy Local File(s) without the No Floppy Install Convert checked will be converted to an Install File(s) script item.

Compile Executable

When you press this button, a Windows Self-installating Executable will be generated from the script that you have entered. It will have the same name as your script file, but have the EXE extension.

Create Directory

This installation script will create an empty directory on the destination computer. If the directory already exists, no action is taken.



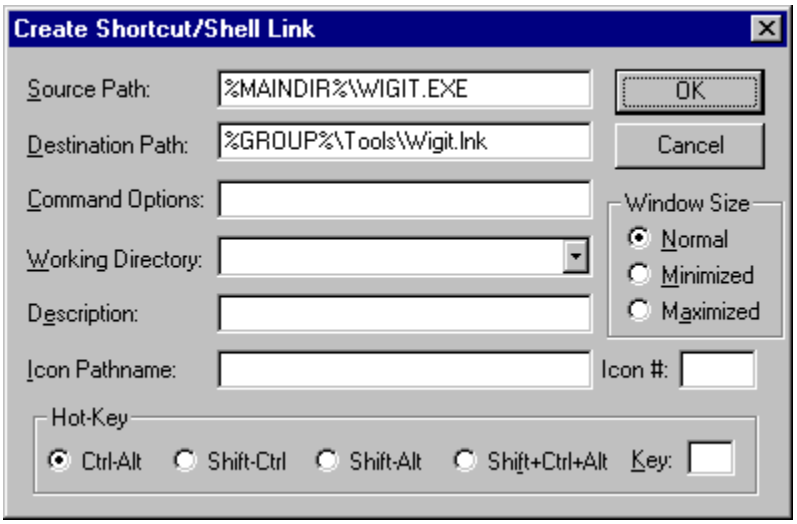
Create Shortcut/Shell Link

The Windows 95 operating system supports the concept of Shortcuts. A Shortcut is a reference to a file without actually copying the file to a new location. You can use this script item to create a new Shortcut. You must be running under Windows 95 (or Windows NT 4.0) to use this script item.

The Create Shortcut can be used to add items to the Start menu under Windows 95. If you create a directory under %WIN%\Start Menu\Programs and fill it with Shortcuts, those items will be added to the Start menu.

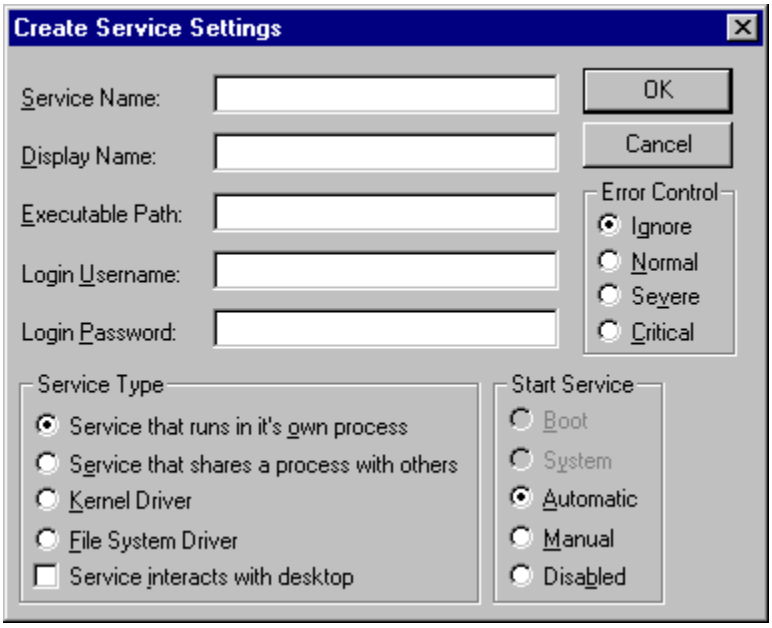
The %GROUP% variable can be retrieved from the: HKEY_CURRENT_USER/Software/Microsoft/Windows/CurrentVersion/Explore/ShellFolders registery key.

If you would like to add Shortcuts to the COMMON group (all desktops) under Windows NT 4.0 you should use the: HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Explore/ShellFolders registery key.



Create Service

This script item gives you the ability to add a Service (background process) to a system. You should consult Microsoft's Windows NT documentation for more information about creating a service. The service will be removed during uninstallation with UnWise.



Current Language

This field holds the current language that you are working on. When you change this field, all of the script items will change to reflect that language. If you are creating a multi-lingual script, modify the installer messages by selecting the Languages item from the View menu.

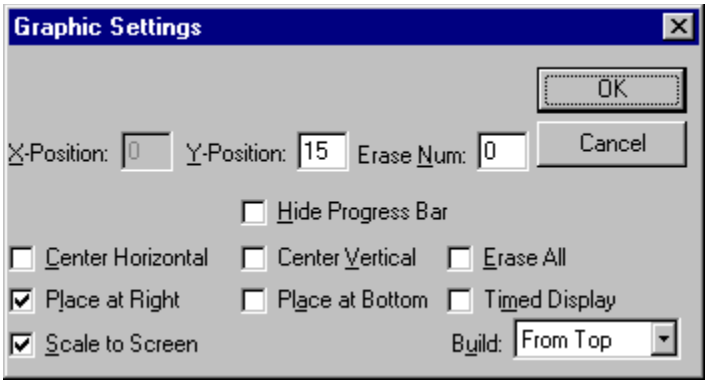
Custom Dialog

The custom dialog editor creates and edits dialog sets. These dialog sets display and gather information from the user during the installation. Each dialog box contains one or more controls that display information, gather information, or perform an action.

Controls

Custom Graphic Editor

The custom graphic editor will create and edit graphics that can be displayed during the installation. You select the type of graphic element to add from the toolbar and click-drag the mouse in the window to define where the new graphic will be displayed. If you double click on the background outside any controls, you can set the properties of the custom graphic.

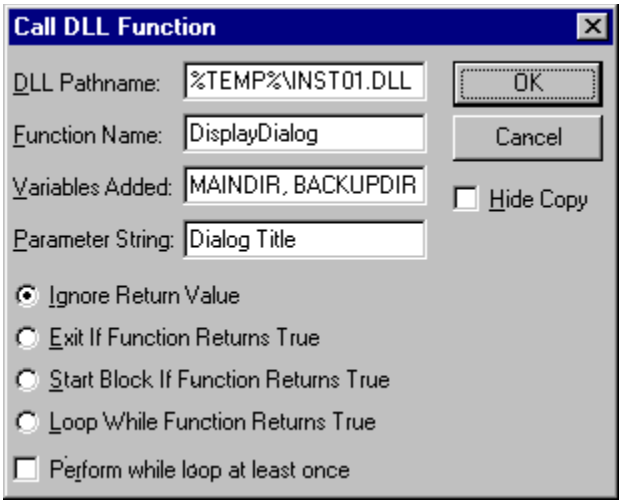


DDE Support

The Wise Installation System allows you to control the installation process from a program (such as a Visual Basic program). This can be useful if you do not want to write DLL functions to customize the installation. The DDESAMP installation script demonstrates the use of a Visual Basic program that reads the values of variables and creates new variables during the installation. The source code to the Visual Basic program is in the DDE subdirectory below the WISE main directory.

Call DLL Function

You use this script item to call a function in a DLL library. The DLL function must be declared as receiving a single parameter as specified in the WISEDLL.H file in the DLL sub-directory. You should copy the DLL to the TEMP directory before you call the DLL function. If you copy the DLL file to any directory other than the TEMP directory, you will not be able to run the installation script in test mode.



Exit If Function Returns True

The DLL function returns a boolean value (true or false). If you select this button, the installation program will exit if the DLL function returns a TRUE.

DLL Function Name

This is the name of the function within the DLL to call. The function should be exported to be called. It must also match the parameters and return value specified exactly.

Ignore Return Value

The DLL function returns a boolean (true or false) value. If you select this button, the value will be ignored by the installation program.

Parameter String

You can pass a string to your function with this field. For example, you may want to pass a pathname of the file you want the DLL function to operate on.

DLL Pathname

You should enter the pathname of the DLL to call. This pathname should be specified for the destination system. The DLL should have already been copied to the destination system.

Variables Added

Since the DLL function has access to the variable list of the running installation program, you may decide to have it add new variables. List the names of the variables you will add separated by commas.

Default Directory

This is the directory that the program will be placed in when it is first executed. If this field is left blank, the default directory will be the directory where the program is located.

Delete File Pathname

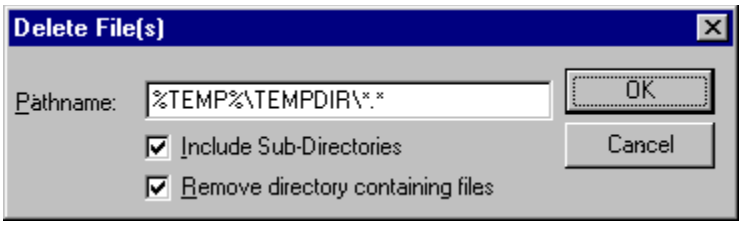
This field holds the pathname on the destination system of the file to delete. You may want to delete installation DLLs after you have used them.

Delete Button

This button will delete the currently selected script item. You should select a script item by clicking on it in the list to the left.

Delete File

This script item will delete a file on the destination system.



Delete File

This button will delete the currently selected file within your installation.

Delete Folder

This button will delete the currently selected folder within your installation.

Delete Group

This button can be used to delete an entire Program Manager group. You should fill in the group name to the right.

Delete Item

You can select this button to delete an icon from a Windows Program Manager group. You should fill in the group name and the icon name to the right.

Delete Registry Key

This will delete the currently selected registry key and all values.

Delete Registry Value

This will delete the currently selected registry value.

Destination Disk Drive

This list selects which floppy disk drive the installation executable will be copied to.

Detail Button

This button will give you additional information about the item selected.

Dialog Directory

You may specify the directory where the dialogs for the installation expert are stored.

Filename on Floppy Disk

This field holds the name that you want your installation executable to be called on the floppy disk. If this field is blank, the file will have the same name as the installation script. Common names for this field are INSTALL and SETUP.

Disk Label

This field holds the volume label for the disks that you are creating. The label should be up to 11 characters long and contain valid file name characters. You can include a %d in your label to include the disk number. If you want a two digit disk number you should include %02d as in the example:

INSTALL #%02d

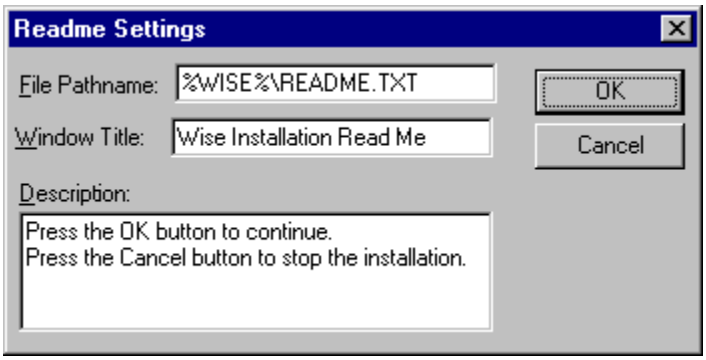
This would label the first disk "INSTALL #01", the second "INSTALL #02", etc.

Reserve Space on First Floppy

If you will be placing files on the first floppy in your installation, you will need to make the installation EXE smaller on the first floppy. You can set this field to the amount of space in K bytes to reduce the first part of the installation EXE.

Display Read Me Dialog

You can display a text file into a dialog box with this script item. The file must contain only text characters and be 30K or smaller in size.



Push Button Action

This list selects the action to be performed when the button is pushed. The valid actions are:

- Return to Previous Dialog: the last dialog displayed before this dialog will be displayed. If this is the first dialog, the dialog set will be exited.
- Return to Script: the dialog set is exited.
- Display Dialog: another dialog is displayed. Select the dialog to display from the list to the right.
- Abort Installation: the user is asked if they want to abort the installation. If they answer yes, the installation is aborted.
- Help Context: The help context number is displayed. The variable HELPFILE must be set to the pathname of a valid Windows help file.

Align

This field determines how the text in aligned in the edit field. The text can be left justified, centered, or right justified.

Auto HScroll

Scroll the text horizontally if it extends past the end of the control.

Auto VScroll

Scroll the text vertically if it extends past the bottom of the control.

Bevel

This field determines how frame and rectangles will be displayed.

Border

Display a border around the edit control. This should normally be set.

Display Component Sizes

If this box is checked, the sizes of the components are displayed to the right of the checkboxes. The variable specified in the Destination Variable field is used to check for the component sizes.

Checkbox Text

Each line of text in this field will be displayed next to a checkbox. References to variables are allowed in this field.

Destination Variable

The variable that will receive the value that represents the checkboxes selected by the user. If the user selects the first checkbox, the variable is set to A. If the user selects the second checkbox, the variable is set to B, etc. If more than one checkbox is selected the values are concatenated (i.e. AC for first and third checkbox).

Sub-Components

This field holds the list of sub-components for the main component variable. List all of the sub-component variables in this field separated by commas.

Combobox Text

The text in this field will be used to fill the listbox portion of the combobox. Each line of text will be displayed as another line in the combobox.

Combobox Type

This field selects the type of combobox to display: Simple (edit control with a listbox), Dropdown (edit control with a drop-down listbox), Droplist (no edit control, listbox only).

Destination Variable

This variable will receive the value the user enters/selects into the edit field of the combobox.

Component Variable

If this field is filled with the name of previously set component variable, the sum of the sizes of the files in the currently selected components will be displayed in the static control.

Confirm If Exists

If the directory exists, the user will be prompted to continue.

Default

This value will be filled in as the default value of the control. This field may contain variable references.

Default Button

Check this box to select this button to be the default button for the dialog box. Only one button should be the default button per dialog box. The default button will be selected if the user presses the enter key.

Directory Tree

This checkbox selects a directory browse listbox. An edit control, a listbox with the directory tree listing, and a combobox with the valid drives will be displayed.

Directory

If this box is checked, the text returned will have any trailing backslashes removed. If the user does not enter a disk drive letter, the default one will be pre-pended.

Disable No Scroll

Normally, if there are not more items than will fit into the listbox, the vertical scrollbars are not displayed. You should check this box if you want to suppress this behavior.

Disk Variable

If this field is set, the variables value is used to display the remaining disk space on the drive referenced in the variable. If the component field is also filled, the total disk space minus the amount used by the selected components is displayed.

Don't Append

If you select a directory tree, this checkbox will set whether the default value will be appended to the default directory as the user changes directories. This is normally set when installing into a new directory. If you are installing files into an existing directory or upgrading software, this box should not be checked.

Display Drive List

The box will cause the combobox to display the current list of valid disk drives. The returned value will be the drive letter and a colon (i.e. D:).

Graphic Pathname

This field holds the pathname of a BMP file (either 16 or 256 color) that will be displayed in a graphic static control. If you display more than two 256 color BMP files, they must use the same palettes.

Height

The height of the control is set with this field.

Horz. Scroll

The control will contain a horizontal scroll bar.

Label

This field holds the text that will be displayed in the push button.

Listbox Text

Each line of this text will be displayed as a line in the listbox. References to variables are allowed in this field.

Destination Variable

The variable will receive the value from the selections made by the user. If Return Letters is selected the letters (A, B, etc.) representing the lines selected will be returned. Otherwise, the actual text in the listbox line will be returned. If this variable has already been set, it's value is used to select the default listbox line.

Lowercase

This checkbox will cause all characters that are entered to be converted to lower case.

Maximum Length

This field defines the maximum length of text the user may enter. If the user enters more than the number of characters specified, they will not be able to leave the dialog box.

Minimum Length

This field defines the minimum length of text the user may enter. If the user enters less than the number of characters specified, they will not be able to leave the dialog box.

MultiLine

Check this box to support multiple lines of text in the text control.

Multi-Select

Check this box if you want the user to be able to select multiple items from the listbox. The values will be concatenated together in the destination variable.

No Check Directory

If checkbox is checked, no directories will be checked in edit fields as the dialog exited. This can be used if another dialog that edits the directory will be called.

No Hide Selection

If the text control loses focus, any selected text will remain selected. Normally the selected text is not displayed as inverse if the control loses focus.

No Prefix

Normally, the amperstand character "&" is used to indicate that the next character will be underlined. If you check this box, the "&" will be displayed and no underlining performed.

No Wrap

This will cause the static text to not be wrapped if it is too long to be displayed on a single line.

Password

Echo all of the characters entered as an asterisk '*'. This is useful for passwords or other secure information.

ProgMan Groups

Check this box to display the current list of Program Manager groups in the control.

Set Variable to Value

If this field is set, the variable will be set to the value specified if the button is pushed.

Radio Button Text

Each line of text in this field will be displayed next to a radio button. References to variables are allowed in this field.

Destination Variable

The variable that will receive the value that represents the radio button selected by the user. If the user selects the first radio button, the variable is set to A. If the user selects the second radio button, the variable is set to B, etc.

Read Only

The user will not be able to modify the text in the edit control. This is useful for read-me displays.

Return Letters

Normally, the actual text of the listbox lines selected is returned into the destination variable. If you check this box, the letter A will be returned if the first line is selected, B for the second line, etc.

Sort

The values in the list are sorted before they are displayed.

Text Font

Use this button to change the font that will be displayed in the dialog. The font that is used on the destination computer may be different if the font selected is not available.

Tab Stop

If this box is checked the edit control can be reached with the tab key. You should un-check this box for read-me text displays.

Text

This text will be displayed as part of the static dialog box. This field is ignored for a graphic static control. You may use variable references in this field. The maximum number of characters in this field is 511.

Text File

If you would like the contents of a text file displayed in the text field, fill in the pathname of a text file on the destination computer into this field. This field should begin with a variable reference.

Type

This field sets the type of the static control. The valid types are:

- Text: Static text will be displayed.
- Group Box: A rectangle will be draw as well as the text. This static control is useful for grouping common controls together.
- Graphic: A graphic will be displayed. The pathname must be set in the Graphic Pathname field.
- Frame/Rectangle: Draws a box on the dialog.

Uppercase

This checkbox will cause all characters that are entered to be converted to upper case.

Variable

This variable will receive the value from the control.

Vert. Scroll

A vertical scroll bar will be part of the control.

Want Return

Checking this box will allow the user to press the enter or return key to add blank lines to the edit control. Normally, you must press the Ctrl-Entry key sequence to add a blank line to the edit control.

Width

The width of the control in dialog units (a device independent measurement) is set with this field.

X-Position

The field holds the left-right position of the control in the dialog box.

Y-Position

The field holds the top-bottom position of the control in the dialog box.

Hide Copy Dialog During DLL Install

Checking this box will cause the "Installing" dialog box to not be displayed while the DLL is being installed onto the system. This checkbox may be used on DLLs that are copied at the beginning of the installation.

Do Loop At Least Once

Normally, the While Loop checks the condition first before the loop is executed. If you check this box, the first check will not be performed. This will guarantee that the script items in the block will be executed at least once.

Drag and Drop Support

You can use Drag and Drop to quickly add many files and directories to your installation script. To use Drag and Drop, do the following:

- Minimize the Wise Installation System so that it is an icon at the bottom of the screen.
- Select the files and/or directories that you would like to add to your installation script in the File Manager. You should use the Shift-Click and Ctrl-Click combinations to select as many files/directories as desired.
- Click and hold one of the selected files and drag it until the mouse is over the icon of the Wise Installation System. Release the mouse button.
- Fill in the Drag and Drop dialog box. The destination field should contain the directory name (without a file name or trailing backslash) where the files will be installed on the destination system.

Duplicate Button

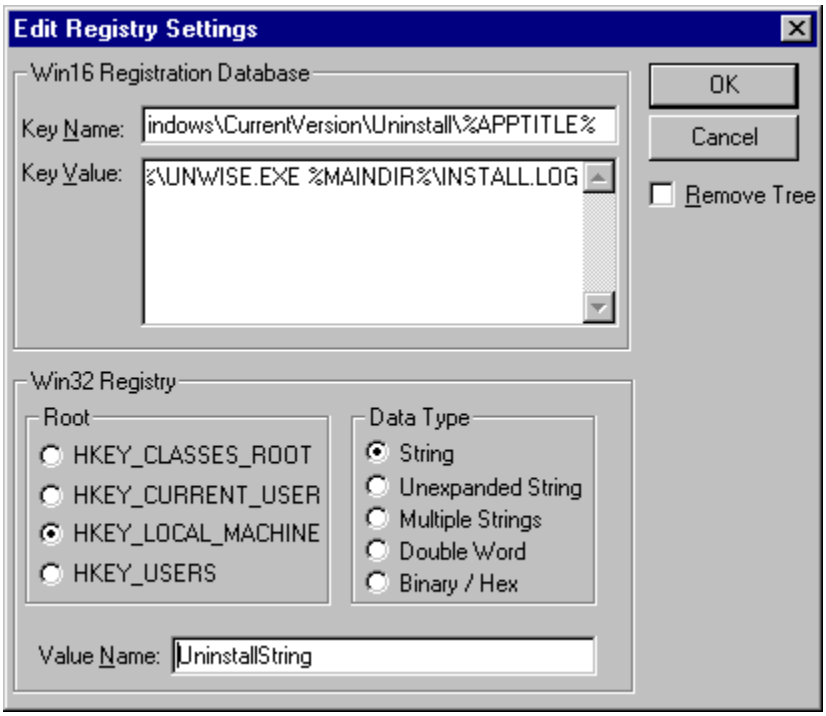
This button will duplicate the currently selected script item. The new item will be added directly below the selected item.

Edit Button

Press this button to edit the values in an existing script item. Select the script item from the list to the left first. You can also double-click the script item to edit it.

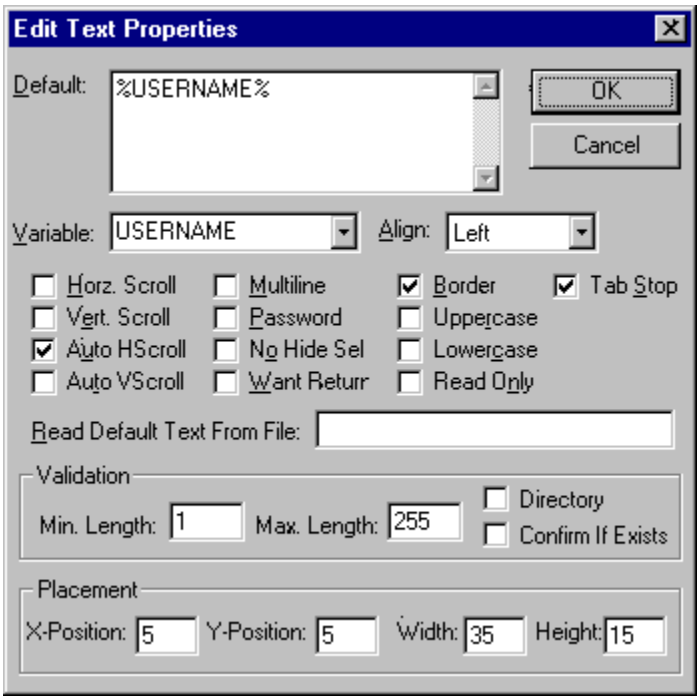
Edit Registry

You can add or update keys in the registration database using this script item. This can be used to register document types and DDE/OLE with the shell.



Edit Text Control

The edit text control should be used to display a large amount of textual information (such as a read-me file) or to gather textual information from the user. The information that is gathered can be up to 255 characters long.



Edit Scipt

This button will switch the view from the Installation Expert to the Script Editor. The Script Editor is an advanced way to customize your installation script.

Create Backup File

If you check this box. a backup copy of the original file will be made. This file will be in the same directory and have the same filename except for the extension. The new extension will be generated as a number that is incremented each time a backup file is created.

Case Insensitive Search

Check this box to ignore case while searching for the text.

Comment

If the search text is found in the file, the value of this field will be placed at the beginning of the original line. This way, you can add a comment out string (i.e. REM, /*, etc.) to the beginning of the old line of text that is replaced.

Insert Options

These options select how the new line of text should be inserted if the search text is found in the text file.

Line Number

This field holds the line number to insert the new text into. If this field is 1 the new line will be added at the beginning of the file. If this value is 0 the new value will be appended to the end of the file. This field is used if you do not search for text or that search text is not found in the file.

File to Edit

This field holds the pathname of the file to edit. This field should begin with a variable reference.

Search Text

This field holds the text to search for in the text file. If the text is found the new line will be placed next to that line.

Ignore Spaces

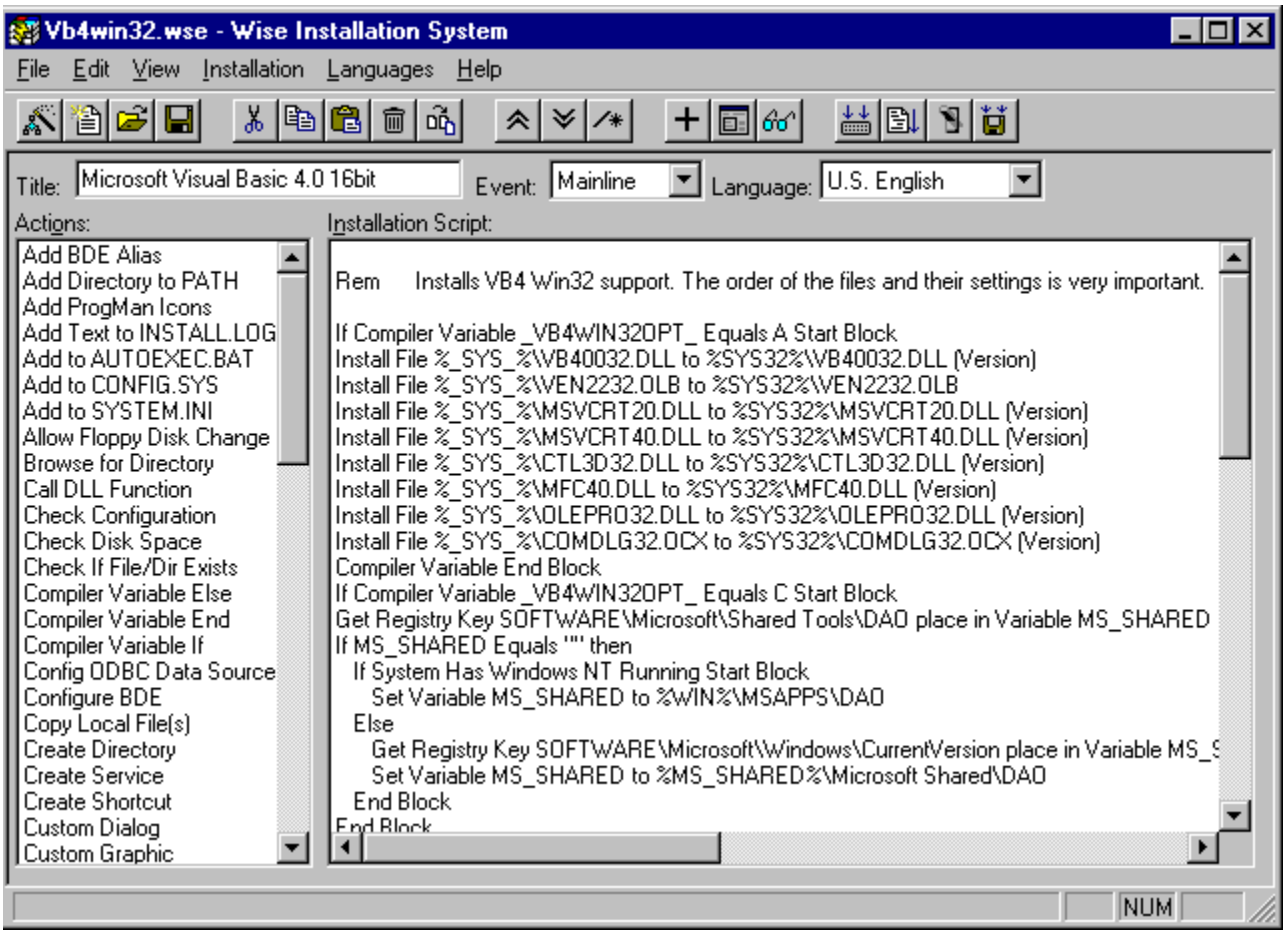
Check this box if you want to ignore white space when searching for the text in the text file.

Search Options

These options determine how the text is searched for in the text file.

Editing the Installation Script

When you first run the Wise Installation System, you will see the following screen. This screen shows you everything about the installation executable that you will create. You can click on parts of the screen below for a description of their function:



Editing PIF Files

NOTE: The internal format of PIF files has not been formally documented by Microsoft and is subject to change in future operating systems.

You can use the Write Variable to File script item to modify a Windows 3.1 PIF file. The following table lists the locations of the fields within the PIF file:

Field Description	Offset	Max. Length	Null Terminated
Window Title	2	30	No
EXE Pathname	36	63	Yes
Command Options	481	63	Yes
Start-Up Directory	101	63	Yes

Else Block

This script item ends the current block and starts an else block. The script items in this block will be executed if the matching start block was false.

3D Effects

This checkbox controls whether the installation will embed the CTL3D.DLL into the installation executable. This will add about 11K bytes to the size of the installation. Checking this box will guarantee that the dialog boxes will have a 3D effect. It also allows you to update the CTL3D.DLL on the destination computer without getting a sharing error.

End Message Block

This script item ends a message or a DLL block. These blocks are used to determine which items in the script will be executed. You must have one End Message Block for each Message Block or DLL Block.

Erase All Graphics

If this button is checked all previous graphics that were displayed will be erased from the screen.

Erase Destination Disk

If this checkbox is selected, all files that are on the floppy will be deleted before the installation file is copied to the disk.

Erase Number

This field holds the number of previous graphics that are being displayed to erase.

Erase Previous

This button will cause the last graphic that was drawn to be erased and replaced with this graphic.

Exclude List

This list contains the DLLs and VBXs that you do not want to be included in the dependency checking. For example, since VBRUN300.DLL is commonly downloaded separately, you would include that file in this list. All DLL files should have the .DLL extension (ex: VBRUN300.DLL) and all VBX files should have the .VBX extension (ex: THREED.VBX).

Executable Name

You may specify a new pathname and filename for the installation exectuable that is being created.

Command Line

Enter the command line for the program you will execute. This field is optional.

Default Directory

You may place the default directory for the program into this field. This field may contain variables.

Maximize Window

If this box is checked, the window for the executed program will take up the entire screen. This is useful if you want to execute Notepad or Write at the end of an installation.

Program Path

Enter the pathname of the program on the destination system that you want to execute. If you leave this field blank, the file extension on the command line will be used to determine the program to execute on the destination system.

Variables Added By Program

A program can query and modify variables via DDE during the installation. You should add all of the variables that this program will add to this field.

Wait for Program to Finish

This checkbox controls whether the installation waits for the program to complete before continuing. You can use this checkbox to allow external programs (via a DDE link) to control the installation process.

Installation Executable Options

The executable files that are created have command line options that can be used to test the installation script and to install software manually.

Test Mode

When you run the Wise executable with the /T command line option, the installation script will be tested. Although the installation will look the same, no changes to your system will be made. No files will be copied and no INI or Program Manager changes will be made. This mode is useful when testing an installation script before it is distributed.

Manual Mode

The /M option runs the installation in manual mode. You will be prompted for the locations of your Windows, System, and Temp directories. You can specify any directories, if they do not exist they will be created. Any changes to INI files will be saved into the Windows directory you specify. With this option you can have the installation run without installing any files into your real Windows and System directory. You can then manually copy the files to their proper destination.

Extract Mode

The /X command line option allows you to extract single files from the installation executable. The files that are contained in the Wise executable will be listed along with the file dates and sizes. You can then select a file and choose the directory to place the file into. If you follow the /X option with a pathname, all of the files in the installation executable will be placed into that directory. If the directory does not exist, it will be created. The /Z command line option acts exactly like to /X option except Windows will be exited after the files are extracted. This can be useful for BBS operators that want to perform virus scanning on all of the files in the Wise executable.

Silent Mode

The /S command line option must be placed at the beginning of the command line. This option can be used to create a silent installation, no background or progress dialogs are displayed during the installation when this option is specified. Normally this option is followed by the pathname of an INI file that contains all of the answers to the normal prompts displayed during the installation. Your installation script can use these values instead of prompting the user (using the Read INI Value script item). This option can be useful for network administrators that wish to install software remotely over a network.

Execute Program

This script item can be used to execute a Windows or DOS program. You should enter the pathname on the destination system and the command line.

NOTE: If you do not check the Wait for Program to Exit checkbox, you should only use this at the end of your installation script.

Exit Installation

This installation script item will cause the installation to be exited.

Installation Expert: Catagories

The Installation Expert has is broken down into five major catagories to ease the creation of your installation. They are: Installation Interface, Application Files, Runtime Support, User Configuration, and System Configuration.

Repackage Installation

You may use the repackage installation function to encapsulate a specific installation for redistribution. An installation script will be created that will duplicate the exact installation procedure (ie file installations, registry modifications, ini changes, etc.)

Installation Expert: Summary

Summary information is displayed for each component of an installation catagory.

Build Effect

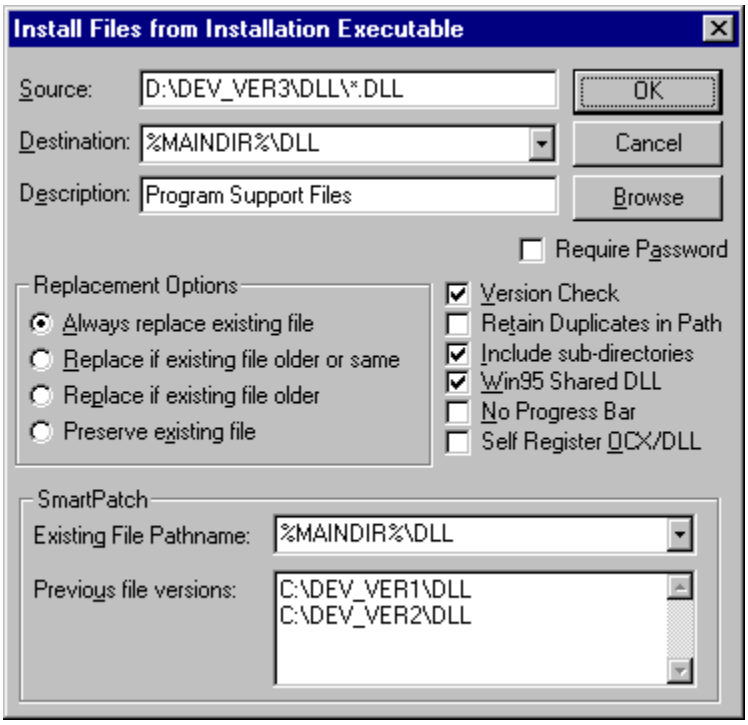
Check this box to have the graphic slowly fade in or move across the screen as it is being first displayed. This option will work with transparent, opaque, and custom graphics.

Fast Create

This option speeds up the installation creation process by copying the compressed version of a file from the previous installation executable to the new one. If the size or date of a file has changed, it will be re-compressed.

Install File(s)

This item will copy a single file into the installation executable. During installation, the file will be copied into the destination directory. The destination directory must begin with a variable name.



File In Use Errors

During the installation you may receive File In Use with the options to Abort, Retry, or Ignore. This indicates that the file you attempted to install is currently being used by another program and cannot be updated. Normally the user should exit all other applications are press the Retry button to continue the installation.

There are some files that are almost always in use by Windows and should not normally be placed into your installation. These files are included in the standard Windows 3.1 installation and do not need to be re-installed: COMMDLG.DLL, DDEML.DLL, and VER.DLL

In addition, the CTL3D.DLL or CTL3DV2.DLL files may be used by the installation, if your install updates these files you must check the Use Internal 3D Effects checkbox in the Global Properties dialog.

File Description

This field holds the description that will be displayed while the file is being copied.

Destination

Enter the pathname of the file on the destination system. This pathname must begin with a variable name. This allows the installer to determine where all of the files will be installed on their system. The variable names must be enclosed in percent signs (example: %MYDIR%). There are three pre-defined variables, WIN (the Windows directory), SYS (the Windows System directory), and TEMP (the Windows temporary directory).

For example: if you have prompted for the destination directory and given the prompt the variable name DESTDIR a valid pathname would be %DESTDIR%\MYFILE.EXT

Source

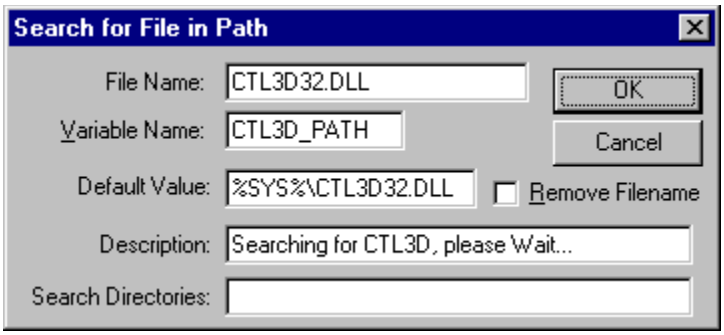
Enter the full pathname of the source file into this field. You can use the browse button to locate a file.

Properties

Press this button to edit the properties of the selected files.

Find File in Path

This script item will search all of the directories in the PATH environment variable for the file specified.

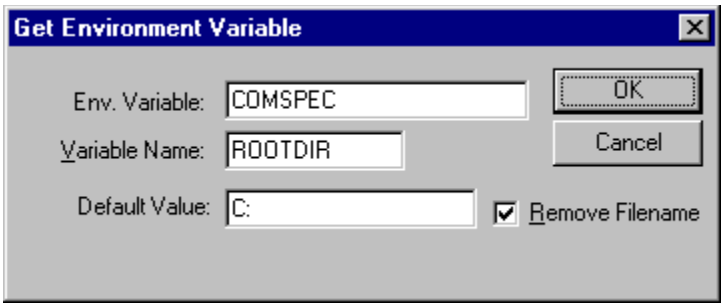


Find Text in Installation Script

This dialog can be used to locate script items that contain a string of text. The search is not case sensitive. Press the Find Next button to locate the next occurance of the text. The search is started at the currently selected script item.

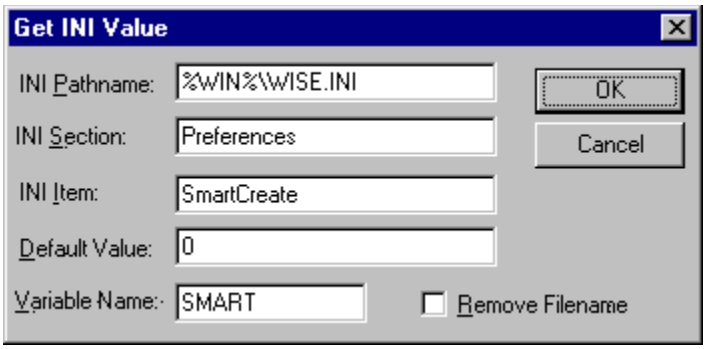
Get Environment Variable

You can read the value of an environment variable into a installation variable with this script item.



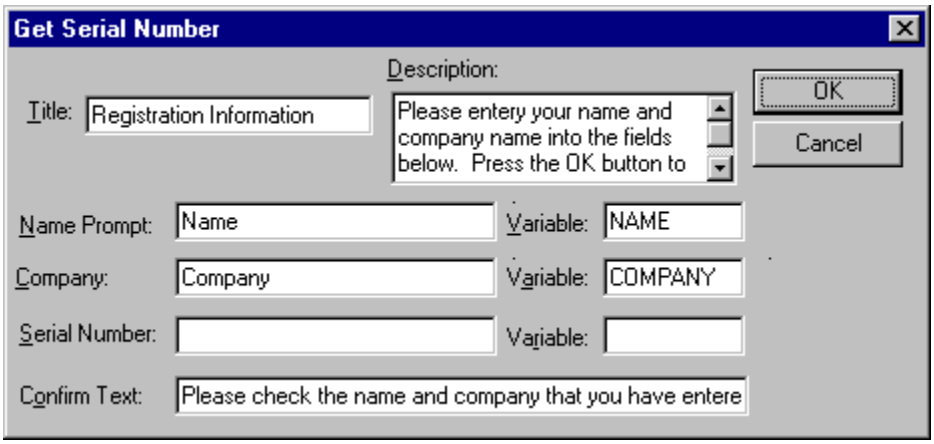
Get INI Entry Value

This script item reads the value from a INI file into a variable.



Get Name/Serial Number

A dialog will be displayed that prompts for the users name, company, and the serial number of the product that is being installed. The values that are entered are returned into variables.



Get System Information

This script item will retrieve information into a variable. The following information can be gathered.

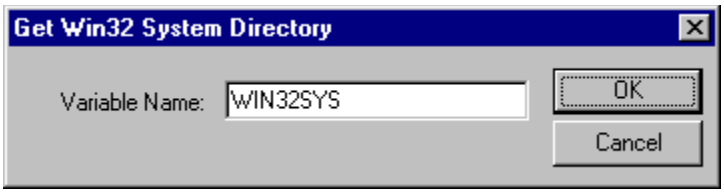
- Current Date/Time
- Windows Version
- DOS Version
- K Bytes of Available Memory: this is the amount of free space when the installation is running, not the amount of physical memory in the computer.
- File Date Time Modified: The Pathname field holds the name of the file to return the information on.
- File Version Number: This return the version number of a EXE/DLL that contains a version resource. If the file specified in the Pathname field does not contain a version resource the returned value will be empty.
- Registered Owner Name: Returns the value entered by the user in the user name field when Windows was first installed. This is a useful default value for the Get Name/Serial Number dialog.
- Registered Company Name: Returns the value entered by the user in the company name field when Windows was first installed. This is a useful default value for the Get Name/Serial Number dialog
- Drive Type for Pathname: This returns the type of drive the given pathname is on. The following values may be returned: F (floppy or removable drive), H (hard or local disk), N (network or remote disk), C (CD-ROM drive), or R (ram disk). If the pathname does not point to a valid disk drive, the variable is set to nothing (blank).
- First Network Drive: Returns the first drive in the system that is a remote or network drive. For example, if the first network drive is F, the returned value is F:. If there are no network drives, the variable will be empty.
- First CD-ROM Drive: The first CD-ROM drive on the system is returned into the variable. For example, if the CD-ROM is on drive D, the returned value is D:.

Get Temporary Filename

This script item creates a temporary file in the %TEMP% directory and places the filename into the variable. All temporary files are automatically removed when the installation is completed. This script item is useful for creating temporary files (DLLs, help files) that are used only during the installation. Since this script item returns only a filename, you **must** reference the temporary file with the %TEMP%\ prefix. For example, if you retrieve a temporary filename into the variable HELPFILE the full pathname of the file is %TEMP%\%HELPFILE%.

Get Win32 System Directory

The system directory pathname will be placed into the variable.



Getting Started

This section describes the basic steps you must take to create an installation script using the Wise Installation System. You may press the F1 button at any time to receive context sensitive help on the current operation.

To create a basic installation script perform the following steps:

- Select the Installation Expert item from the View Menu. (You may have to save the previous script if you have changed it). The Installation Expert groups the functions of your installation into 5 Catagories.

- The Installation Interface defines the the media type, application name, which dialogs and graphics you would like to display.

- The Application Files defines the components and the files that your applcation requires. You can add files to the application
directory and the Windows/System directories

- The RunTime Support defines any sub-systems that you may need (i.e. uninstall, ODBC, VB4, BDE, etc)

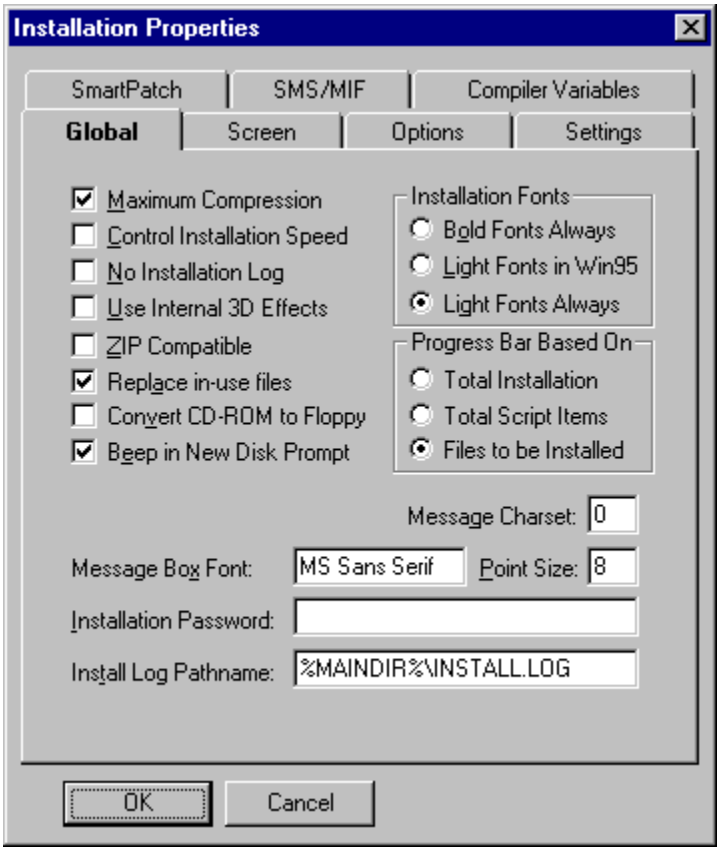
- The User Configuration defines the Icons, File Associations, INI file creation/editing, and registry changes.

- The System Configuration defines Devices, Services, and Autoexec.bat/Config.sys changes.

Once you have defined all of the requirements for your installation you may either Test, Run, or Distribute your installation. If you require additional customization click on the Edit Script button.

Global Properties

These options effect the way the installation script runs. You can click to on the dialog box below to get more detailed information on the global properties.

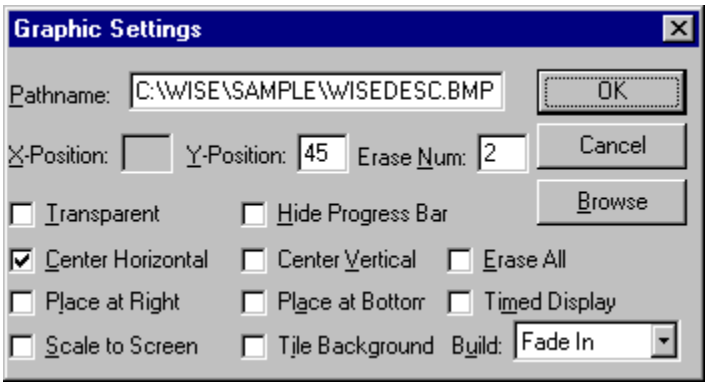


Graphic Pathname

This is the full pathname of the graphic file on your machine. This file must be a Windows Paintbrush BMP file. It should be 16 colors or less.

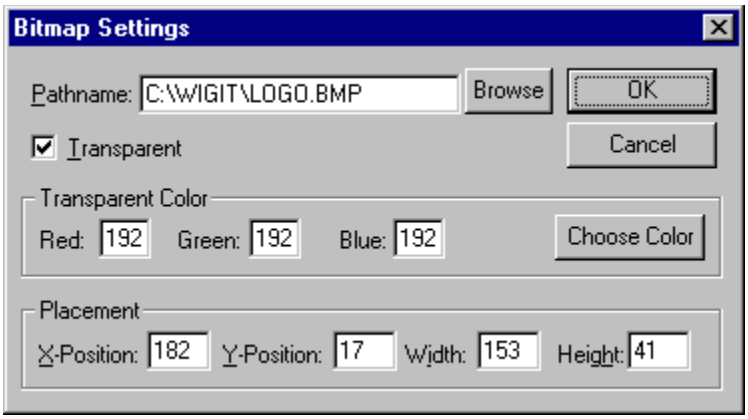
Graphic

You can add bitmap graphics to the installation screen that are displayed during the installation process. The file type of the graphic files you use should be BMP. A transparent graphic will make any solid blue color (Red=0, Green=0, Blue=255) in the graphic transparent as it is being displayed.



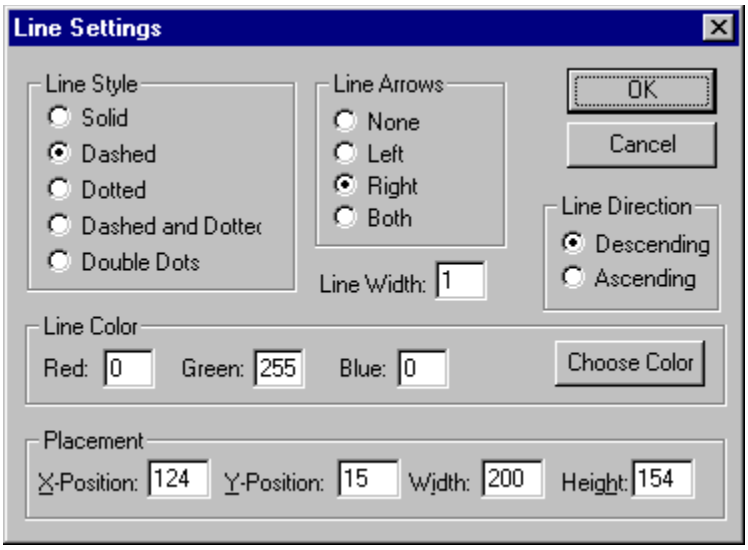
Graphic Bitmap Properties

This dialog controls the properties of bitmaps that are part of custom graphics.



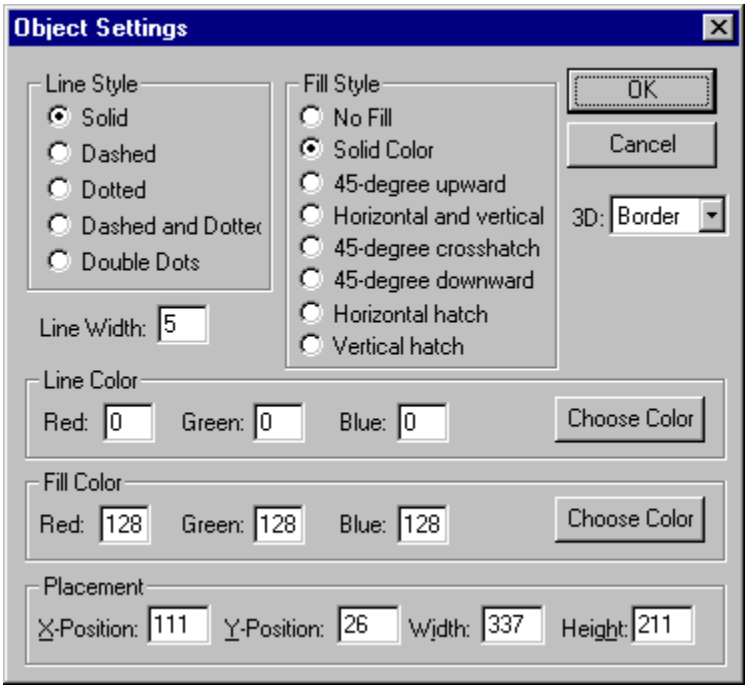
Graphic Line Properties

This dialog controls the settings of lines in a custom graphic.



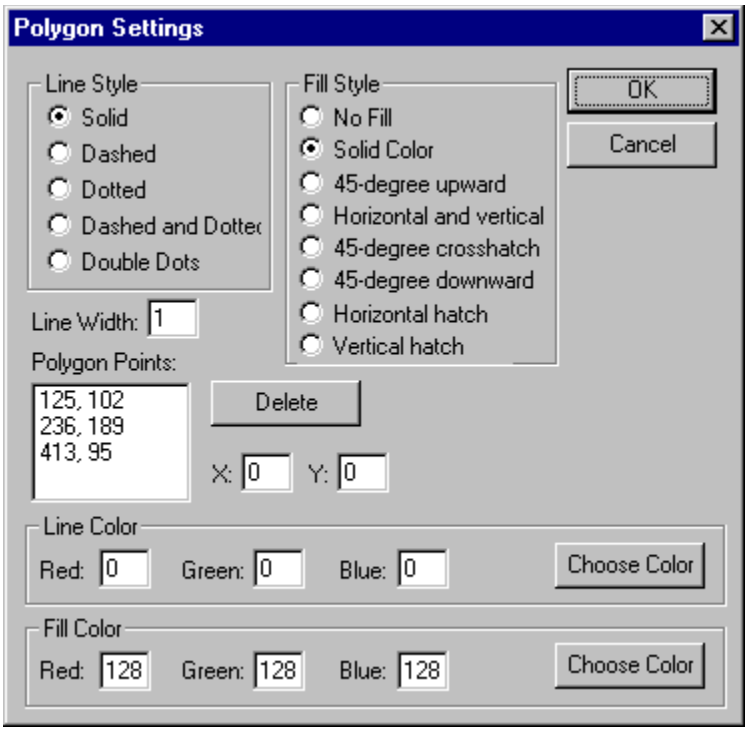
Graphic Object Properties

This dialog controls the properties of a rectangle, rounded rectangle, or ellipse.



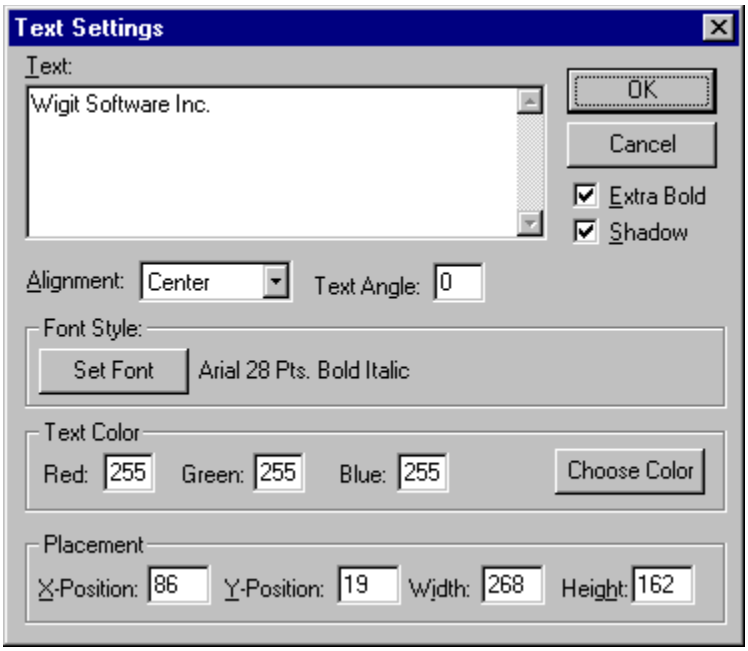
Graphic Polygon Properties

This dialog controls the properties of polygons.



Graphic Text Properties

This dialog controls the setting of text that is to be displayed in a graphic.



Hide Progress Bar

By checking this box the Progress dialog box will be hidden before the graphic is displayed.

3D Border

This list selects which type of border to display around a rectangle. This border has been optimized for gray boxes.

Extra Bold

Check this box to have the text displayed extra bold.

Fill Color

These fields control the color that is painted within the graphics object.

Fill Style

These radio buttons select the type of fill pattern.

Line Arrows

You can choose to place arrow heads at one or both of the ends of the line using these radio buttons.

Line Color

These fields hold the red, green, and blue colors of the line to draw. You can enter the values directly into these fields (by entering numbers from 0 to 255) or use the Choose Color button.

Line Direction

These buttons select whether the line is drawn as either ascending or decending.

Line Style

Select the look of the line to draw using these radio buttons.

Line Width

This field holds the width of the line in pixels, from 1 to 99.

Bitmap Pathname

This field holds the pathname of the bitmap to display on the local computer system. You cannot use variables in this field. You can use the Browse button to locate the file to display.

Placement

These fields control the size and placement of the graphic in the drawing surface. You should normally place your graphics in the upper left corner of the drawing surface and use the graphic settings dialog to set the location of the entire graphic on the screen.

Polygon Points

These fields display and edit the endpoints of the polygon. To change an endpoint select on of the points from the list and change it's value in the X and Y edit fields. You can remove a point from the list with the Delete button.

Shadow

This will display a shadow below the text that is displayed.

Alignment

This field sets the alignment of the text that is displayed.

Text Angle

This field holds the angle to display the text from 0 to 359 degrees. If you display the text at any angle other than 0 you can only display a single line of text.

Text Color

These fields control the color of the text that is displayed. Enter values into the Red, Green, and Blue (from 0 to 255) or use the Choose Color button to select from a dialog box.

Text Font

Press Set Font button to change the font of the text that will be displayed. Since the graphic is displayed using the fonts on the destination computer you cannot be guaranteed that the font you choose will show up exactly the same during the installation. It is best to use fonts common to all Windows installations (i.e. Arial, Times New Roman, New Courier, etc.)

Text

This field holds the text that will be displayed. You may not place variable references into this field.

Transparent

Check this box if you would like your bitmap to be displayed with one of the colors being transparent (i.e. the background shows through that color).

Transparent Color

If you are displaying a transparent bitmap, set the color that should be transparent with these fields.

Group Name

This is the name of the Program Manager group that you want to effect.

Help

Press this button to receive help on this dialog box.

Hide Program Manager

If this box is checked, the Program Manager will not appear as icons are added or deleted.

No Progress Bar

Checking this box will cause the progress dialog box to be hidden as the file is being installed to the system. This can be useful for small temporary files that will be installed (install DLLs, temporary readme files, etc.)

Add Command to Autoexec.bat

You can add a path to the AUTOEXEC.BAT by filling in the Add to Path editbox. This field should start with a variable reference. (i.e. %MAINDIR% for the main application directory). You can also add commands to the AUTOEXEC.BAT file by clicking the Add Button.

–

Add Command to Config.Sys

This script item will add a line of text to the CONFIG.SYS file. If the CONFIG.SYS file is changed, the computer will be rebooted at the end of the installation.

Add File

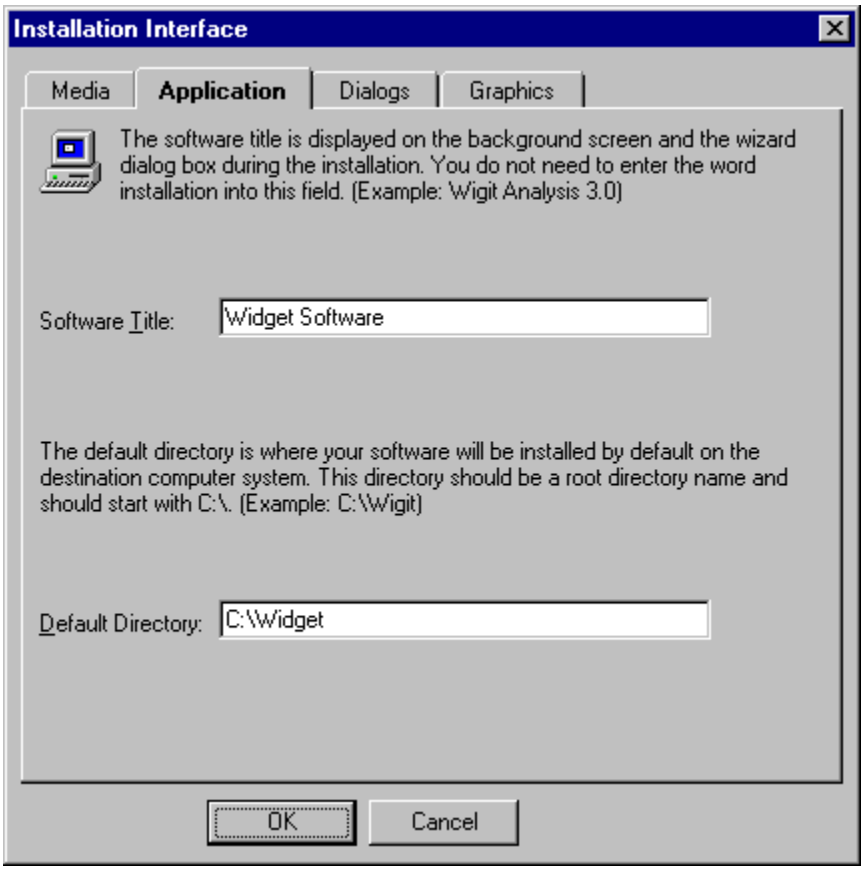
This dialog shows a list of files that you may select that are aleady included in your installation. If a file that you require does not appear on the list you may add the file in the Files page of the Installation Expert.

Add Path to System

This can be used to add a directory to the PATH environment variable. The AUTOEXEC.BAT file will be edited and the directory will be appended to the PATH statement, if it is not already there. If a change is made to the AUTOEXEC.BAT file, the system will be rebooted at the end of the installation. All occurrences of the SET PATH command will be edited. This is done to ensure that the proper SET PATH command is executed during bootup.

Installation Application

This dialog defines the software title and the default directory. The software title is used for the for the wizard dialog titles, in the welcome dialog, as the primary icon name, and as the title in Add/Remove Software dialog under Windows 95. The default directory will appear exactly how you type it under Windows 3.1 and Windows NT 3.51, however it will be changed to be under "Program Files" for Windows NT 4.0 and Windows 95.



Installation Expert Default Directory

This directory will be shown as the default directory for the user to install into. This value may be modified for different operating systems. Under Windows 3.1x and Windows NT 3.5x the value will not be changed. Under Windows 95 the default value for "Program Files" will be obtained from the registry and will be prepended to the beginning of the directory

Installation Expert Software Title

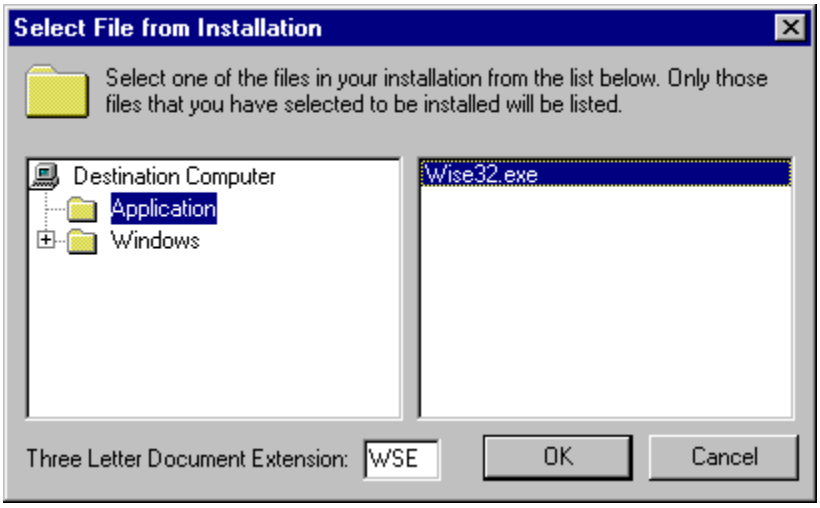
The software title is used for the for the wizard dialog titles, in the welcome dialog, as the primary icon name, and as the title in Add/Remove Software dialog under Windows 95. You should not include the word installation within your title, as it will be added automatically when needed

Installation Expert File Association

When you press the Add Button you will be prompted with the thee letter document extension (ie .WSE) and the executable to run. A file association is used to run a specific program when a file with specific extension is double-clicked within the Explorer or File Manager. For example, Microsoft would associate the .DOC extension with the WINWORD.EXE application.

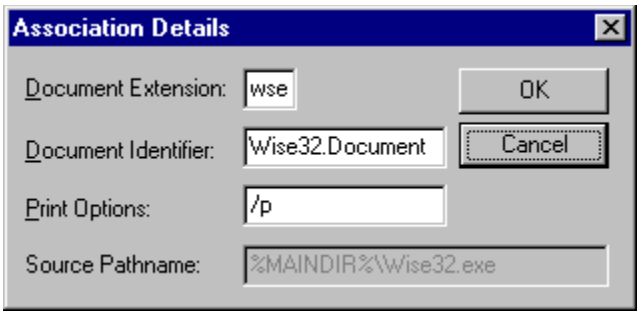
Installation Expert File Association

A file association is used to run a specific program when a file with specific extension is double-clicked within the Explorer or File Manager. For example, Microsoft would associate the .DOC extension with the WINWORD.EXE application.



Installation Expert File Association

A file association is used to run a specific program when a file with specific extension is double-clicked within the Explorer or File Manager. For example, Microsoft would associate the .DOC extension with the WINWORD.EXE application.



Installation Expert File Association

This is the three letter extension that you would like to be associated with your program. When a user double clicks (or opens) the file with this extension your program will be executed with the filename as the command line parameter.

Files Associations Document Identifier

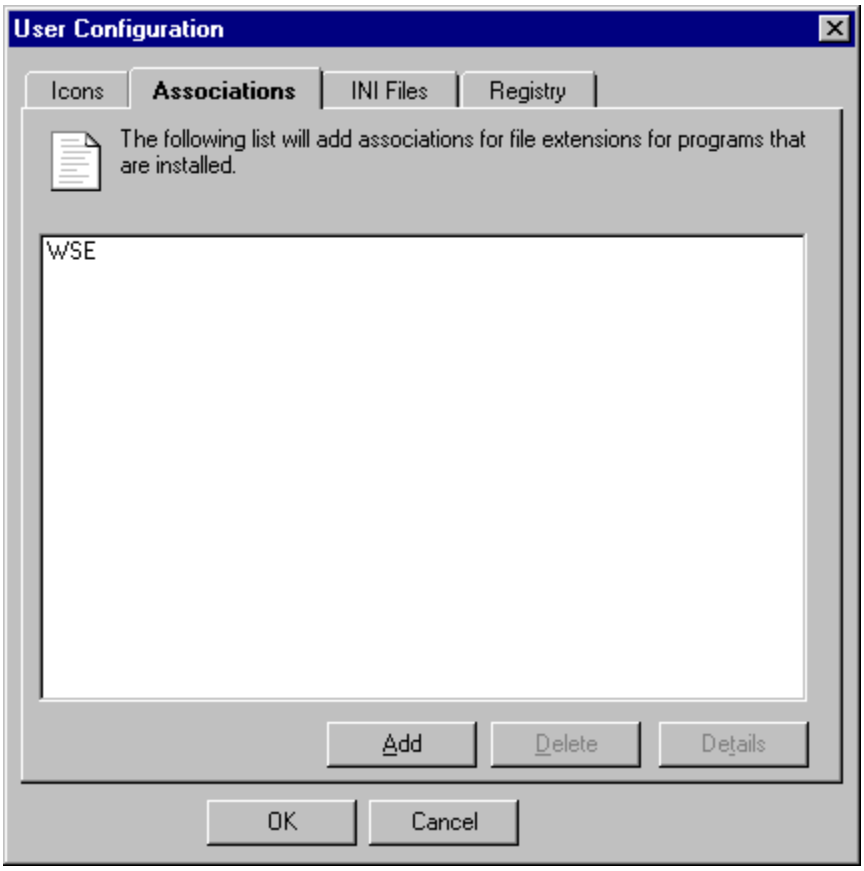
This is an internal name used within the registry. In most cases this will not need to be edited.

Files Associations Print Options

When a user selects a file that is associated with your progam he can either open or print the document. You should include the command line to allow your program to print your document (if supported).

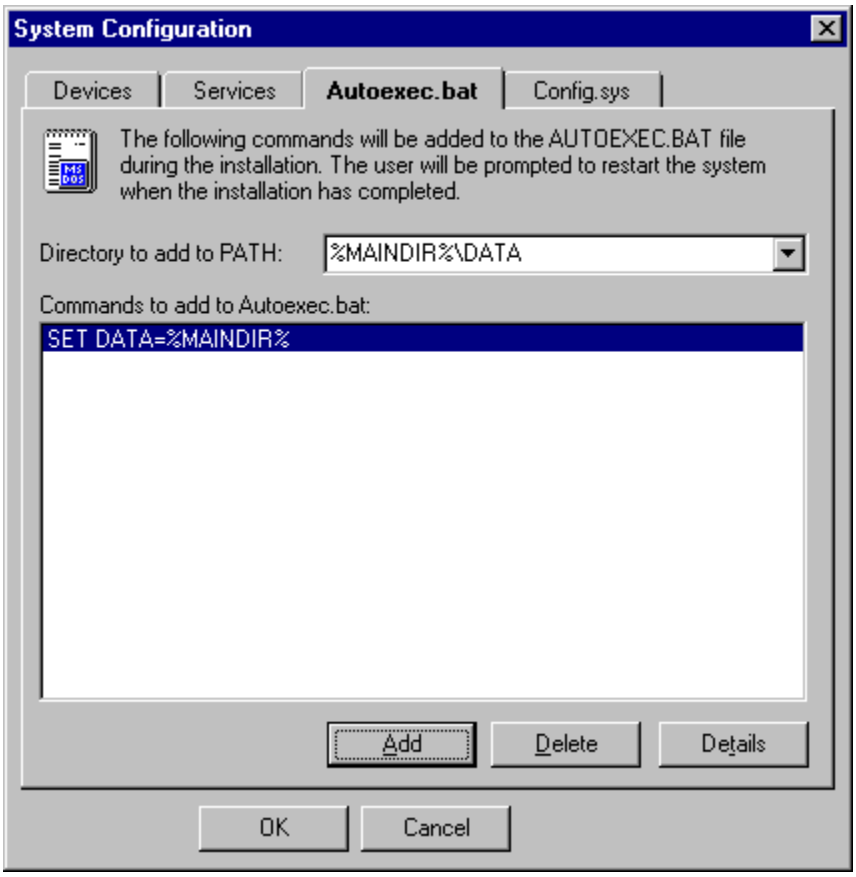
Files Associations

To open an application when you double-click a related document file, the file's type must be defined in the Registry. If the file type is defined in the Registry, it appears in a list of file types that you can associate with an application. If a file type has been associated with an application, you can reassociate the file type to a different application.



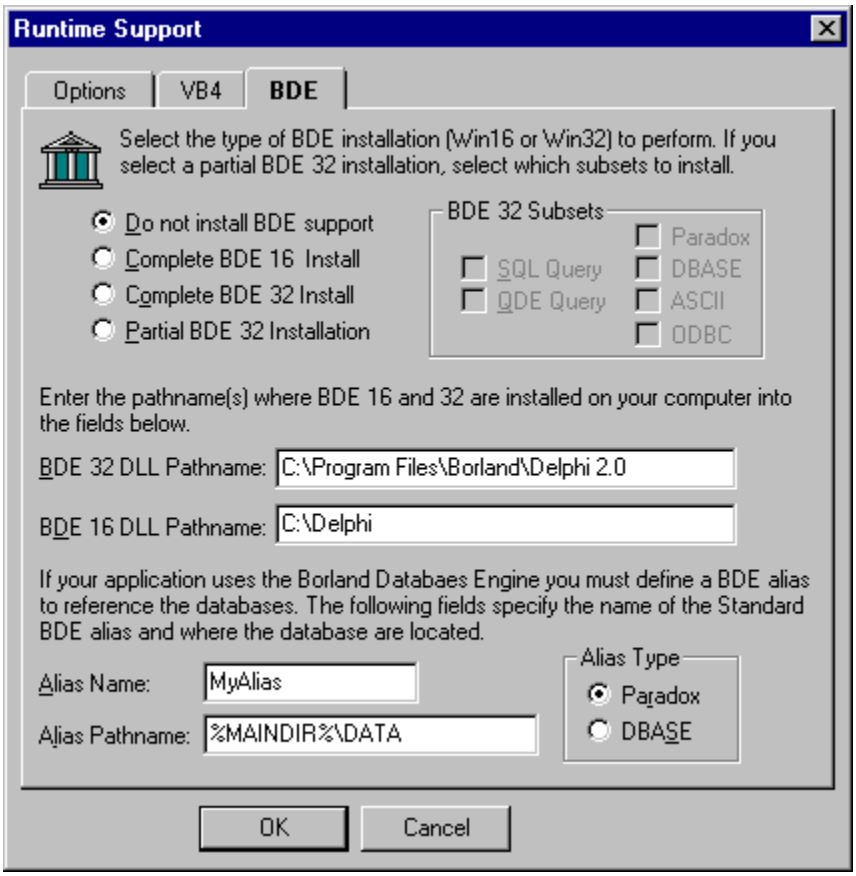
Editing the Autoexec.bat file

This script item can be used to add a directory to the PATH environment variable. The AUTOEXEC.BAT file will be edited and the directory will be appended to the PATH statement, if it is not already there. If a change is made to the AUTOEXEC.BAT file, the system will be rebooted at the end of the installation.



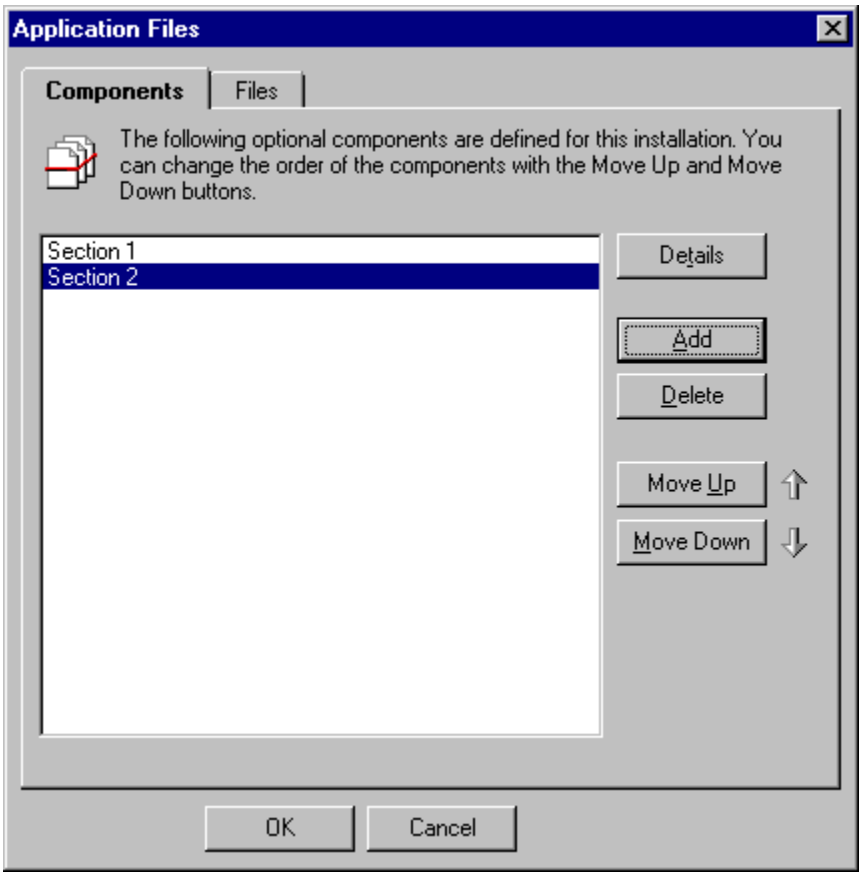
BDE Runtime Support

WISE supports both BDE-16 and BDE-32 installations. You need to specify what platform, and for the 32bit, you need to specify whether you will be creating a full or partial installation. You may optionally specify an Alias for your database.



Components

You may create components of your installation that the user may choose to install or not install. Each component that you create in this dialog will be listed in the same fashion as a directory in the Files dialog. To create sub-components you will need to use the Script Editor.

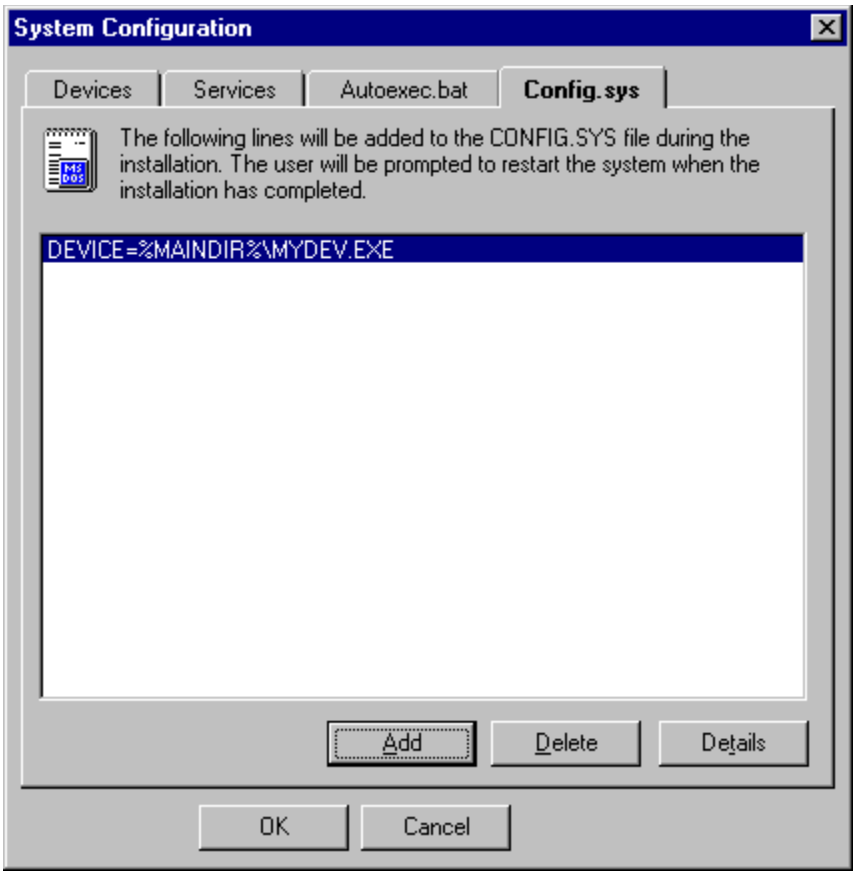


Installation Expert Component Name

The name of the component will be displayed in the component dialog for the user to pick. The components created here will be at the same level within the installation. To create sub-components you will need to use the Script Editor.

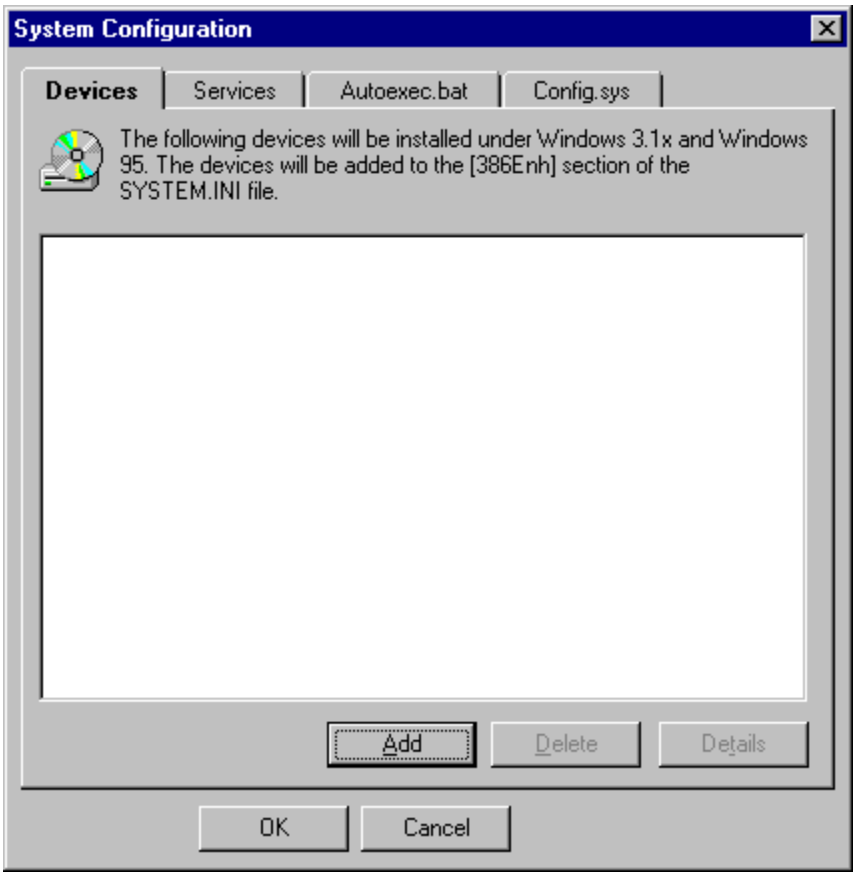
Editing the Config.sys file

This script item will add a line of text to the CONFIG.SYS file. If the CONFIG.SYS file is changed, the computer will be rebooted at the end of the installation.



Adding Devices

The [386enh] section contains information specific to running Windows in 386 enhanced mode, including information used for virtual-memory page swapping.
For entries in this section that specify virtual devices, the value can appear in two ways: either the filename of a specific virtual device driver (with path if necessary), or as an asterisk (*) followed immediately by the device name to refer to a virtual device built into the WIN386.EXE file. Press the Add button to add a new device to this section.



Installation Expert Adding Devices

.When you press the Add button you will be prompted for which device file to add the the [386Enh] section of the SYSTEM.INI file. This file must already have been selected to be installed in the Files page.

Installation Expert Standard Dialogs

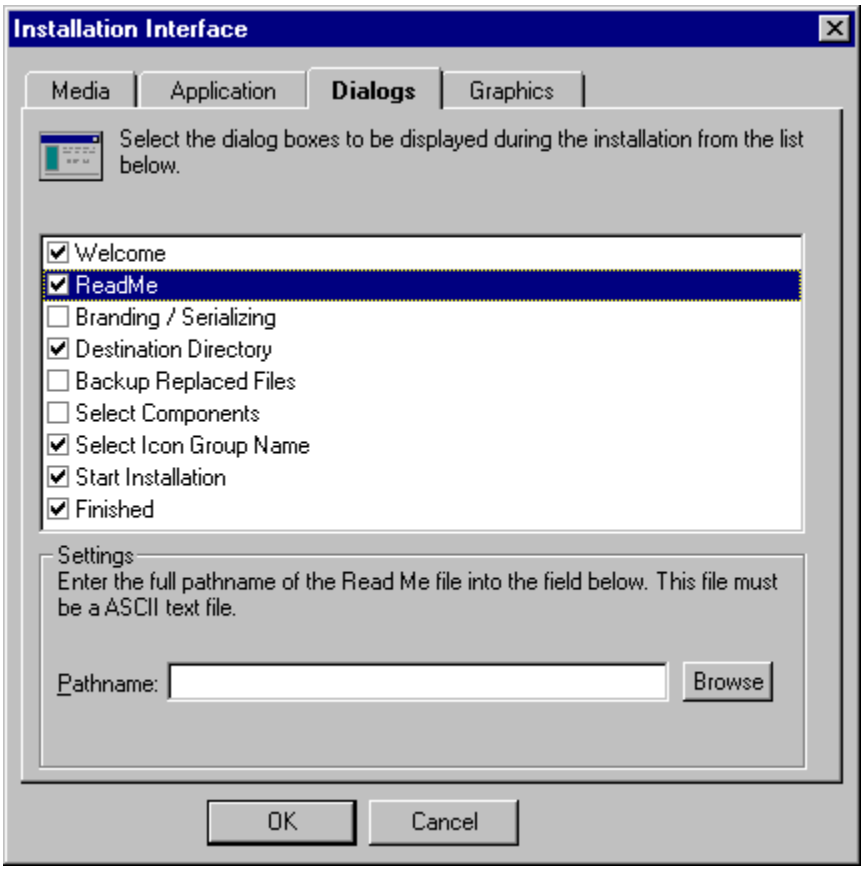
There are nine standard dialog templates that may be included in your installation. The Branding/Serializing and Backup Replaced files will display 2 dialogs each, all others display a single dialog. The defualt are dialogs under the DIALOGS\TEMPLATES directory and may be edited using the Edit/Dialog Templates option.

Installation Expert Dialog Setting

The ReadMe standard dialog requires additional information. You must specify an text based file that will be displayed in this dialog. You do not need to include the file into your installation as it will be done automatically.

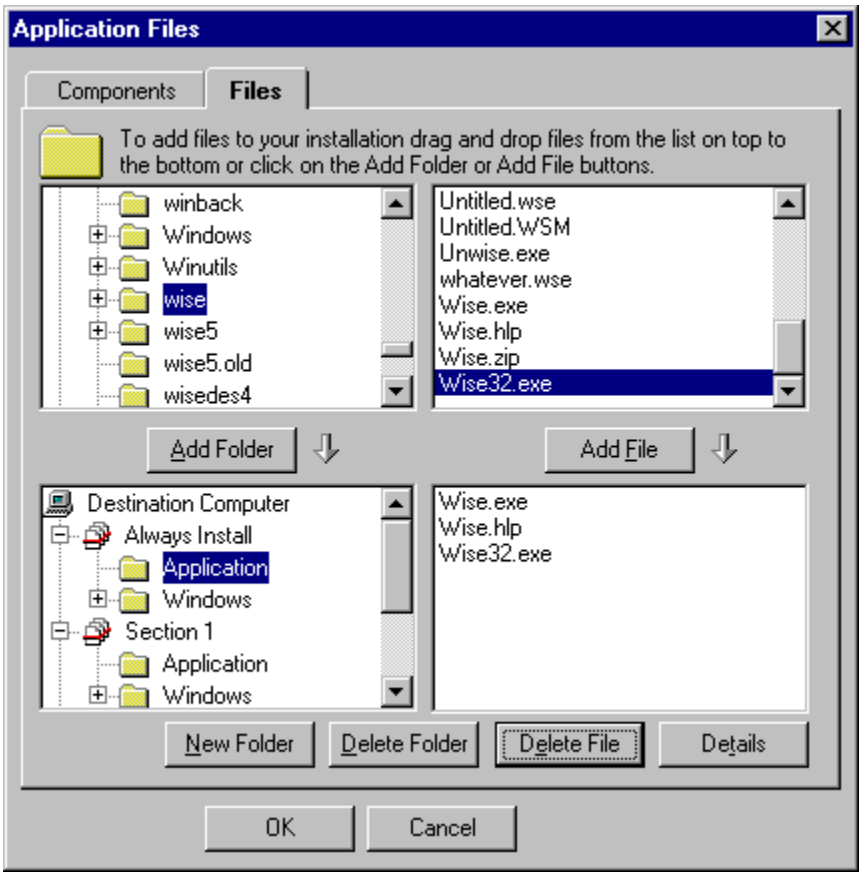
Installation Dialogs

There are nine standard dialog templates that may be included in your installation. The Branding/Serializing and Backup Replaced files will display 2 dialogs, all others display a single dialog. The defualt dialogs are under the DIALOGS\TEMPLATES directory and may be edited using the Edit/Dialog Templates option.You may include as many or as few dialogs as you like, however we suggest at least Welcome, Destination Directory, and Finished dialogs.



Installation Files

The top half of this dialog is your machine (My computer) and the bottom half is the customer's machine (Destination computer). Click on each section of the dialog for further information.



Installation Expert Files

This is a modified version of the Windows 95 explorer and a file open dialog. This is a list of all drives/folder on your machine. You may add the contents of a folder and all sub-folders (to the highlighted folder below) by clicking on the "Add Folder" button. You may move around the directory tree by clicking on an entry.

Installation Expert Files

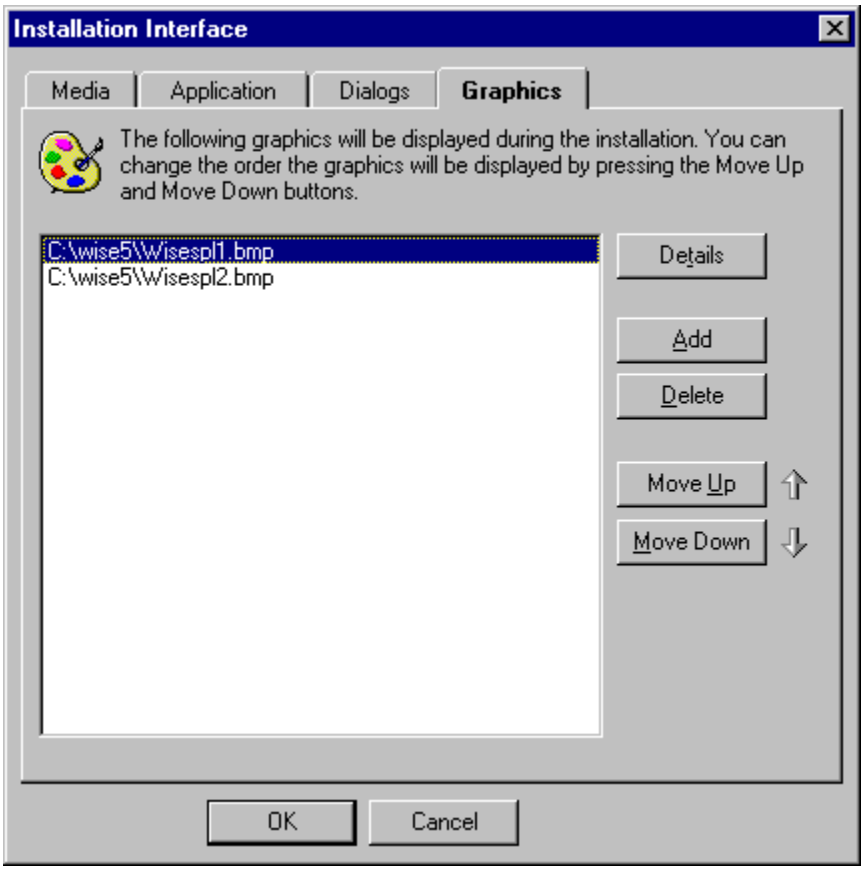
This box contains a list of files in that folder, you may either double click on a file or use the "Add File" button to include a file.

Installation Expert Files

The is a representation of how/where installation files will be installed. If you do not have any components installed, then the structure will have two folders off of the destination computer. The Application folder may contain files or more sub-folders. The Windows folder will contain files to be installed into the Windows directory and sub-folders for Fonts, System, and System32. You may only create folders under the Applications folder, not under the Windows folder. If you have components in your installation then the level just below will list your components and then have the other directories below that level. You may change the installation properties of a file by clicking on the details button.

Installation Graphics

The graphics that you add here by default will be timed and the first one will be displayed right before the first file is installed. You may change the properties on a graphic by clicking on the details button after you have selected a graphic. The order that they appear on the screen is the order they will appear in the installation.



Installation Expert Graphics List

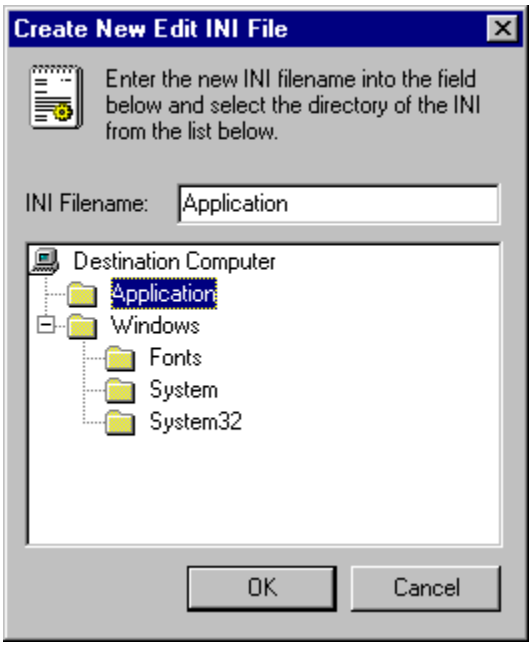
Each graphic must a 16 color or 256 color non-compressed Windows bitmap. You may double click on a single graphic to change the properties. By default the graphics will be timed for equal display throughout the installation.

Installation Expert INI Editing

You can Cut and Paste from any standard INI file. If you include a section (i.e. [Section]) with no entries after it, the Section will be deleted. If you include an entry in a section with no value (i.e. "LastFile=") then that line will be deleted. You may do multiple events in a single "Edit INI File". For example: you may delete a section(s), add a section, add an entry(s), remove an entry(s) all within one entry.

Installation Expert INI Editing

You can name the INI file that you will be editing. After you create the entry in the Installation Expert click on the details button to edit the INI.



Installation Expert INI Editing

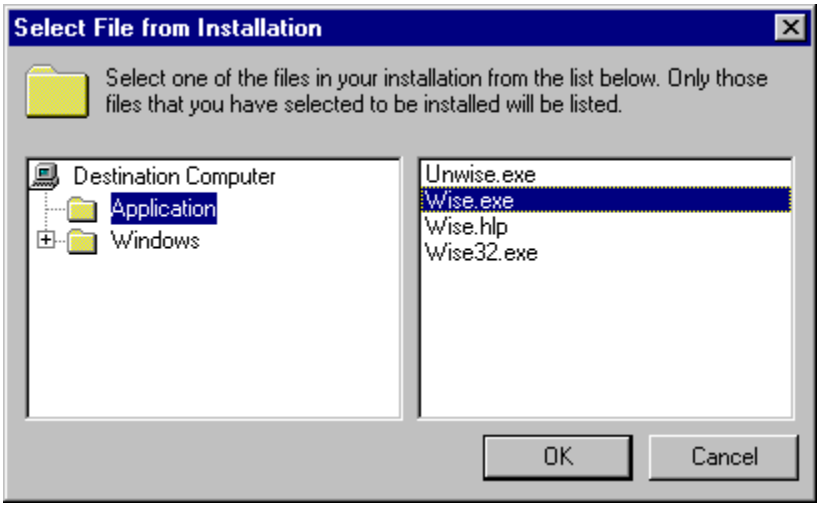
You must specify the INI name and where you would like that INI file placed when the installation is run.

Installation Expert Add Icon

You may select any number of icons to be added during the installation. When you click on the Add button it will bring up a list of the files being installed. Under Windows 3.1 there are no restrictions for adding icons. Under the Windows 95 guidelines it is suggested that you add no more than one or two icons.

Select File from Installation

This dialog allows you to select a single file from the files that are currently being installed to the destination computer.

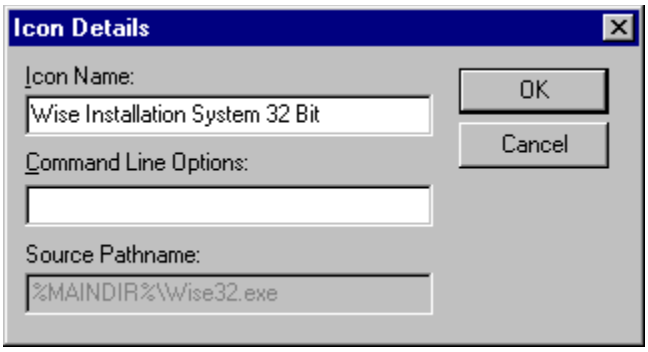


Installation Expert Icon Command Line

This is the command line that will be passed to your program when your program is executed.

Adding Icons/Shortcuts

You may select any number of icons to be added during the installation. Under Windows 3.1 there is no guidelines for adding icons. Under the Windows 95 guidelines it is suggested that you add no more than one or two icons.



Installation Expert Icon Group Name

The default group name will be used as the Program Manager Group under Windows 3.1 and Windows NT 3.51, and as a sub-menu under Windows NT 4.0 and Windows 95. The icon name you use will be adapted for use under all operating systems

Installation Expert Icon Name

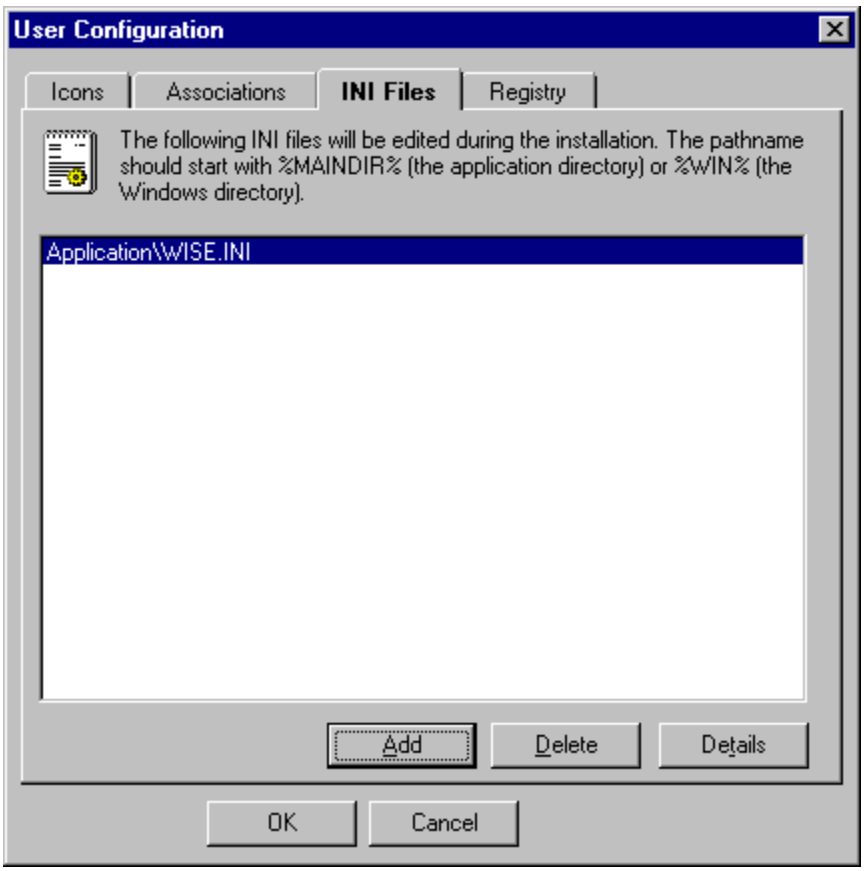
This is the name of the icon that will appear under the picture in the Windows 3.1 or Windows NT 3.51 Program Manager or on the Start Menu under Windows 95 or Windows NT 4.0.

Adding Icons/Shortcuts

You need to use the Add Button to create a new INI file within your installation or to edit an existing file. Once you specify the INI file, click on the details button to insert the text.

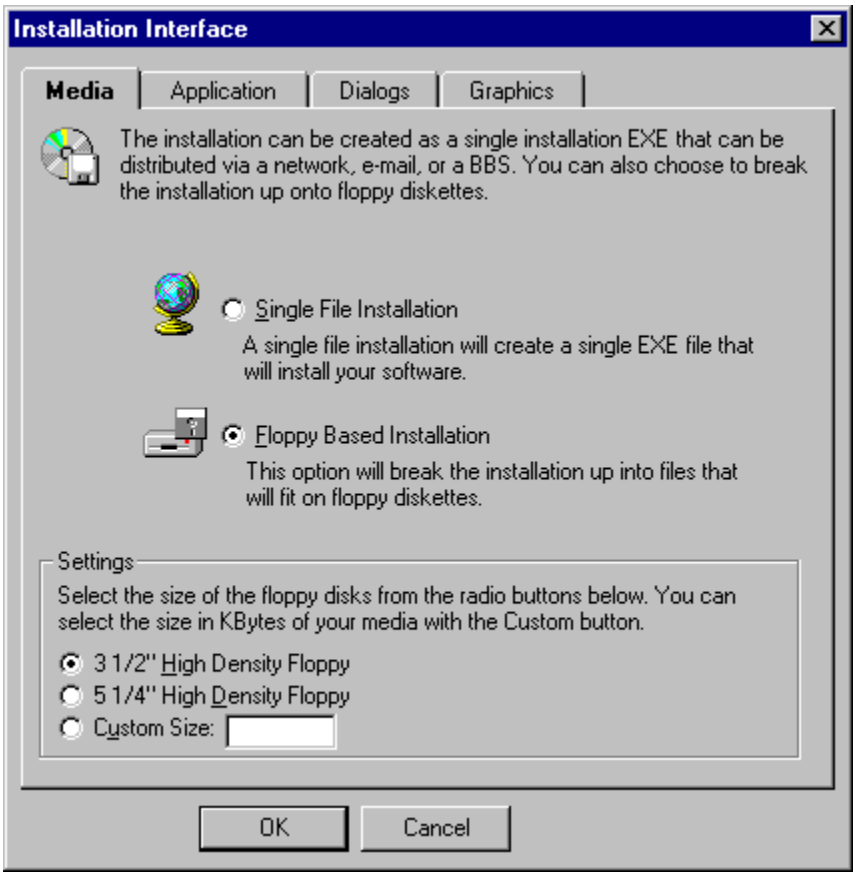
Editing INI Files

The .INI file is a special file that contains the shared configuration options for your application. Using an .INI file is easier and more accurate than using command-line options. When you create the .INI file, you specify all of the options in this file instead of typing them on the command line. The .INI file can include as many options as you need, while the command line is limited to 128 characters.



Distribution Media

You can select either a single file installation or a floppy based installation. The floppy based information has options for which type of floppy or how big the installation executable is.



Media Settings

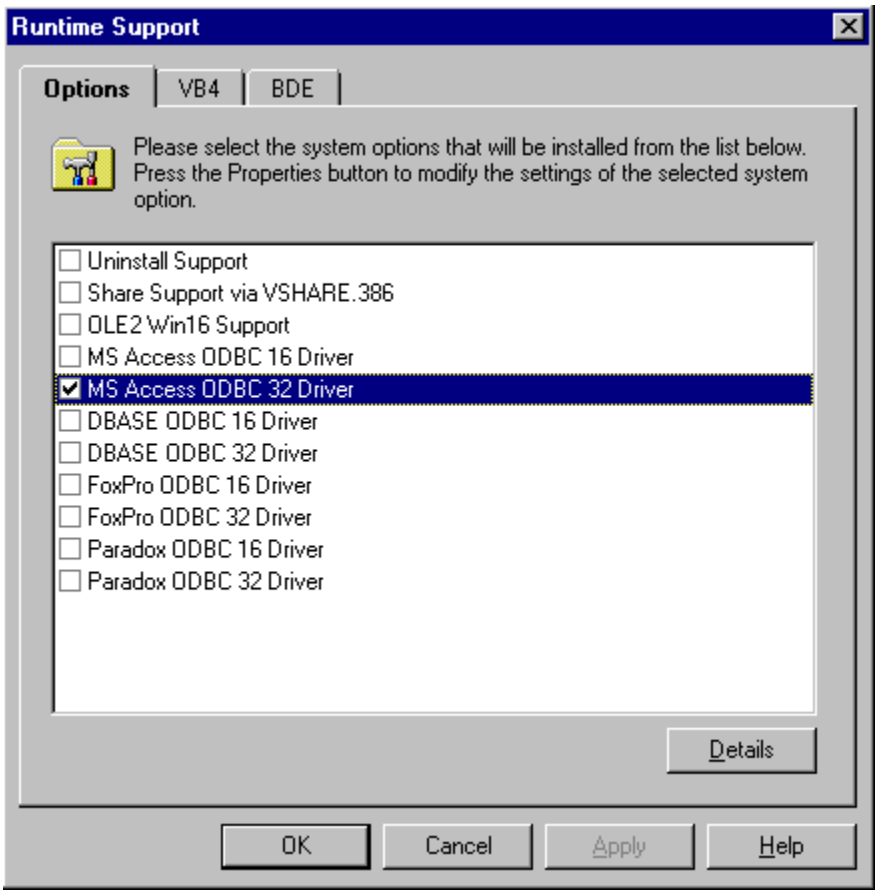
Based on the type of media selected there may be additional settings required.

Installation Media Type

You may select to have the installation be a single file EXE (as large as need to compress your files) or a multiple file installation. With a multiple file installation the size of the file may be specified.

Installing Optional Components

You can click on as many different runtime support components as needed by your application. This list will grow as we develop installation scripts for other various products.

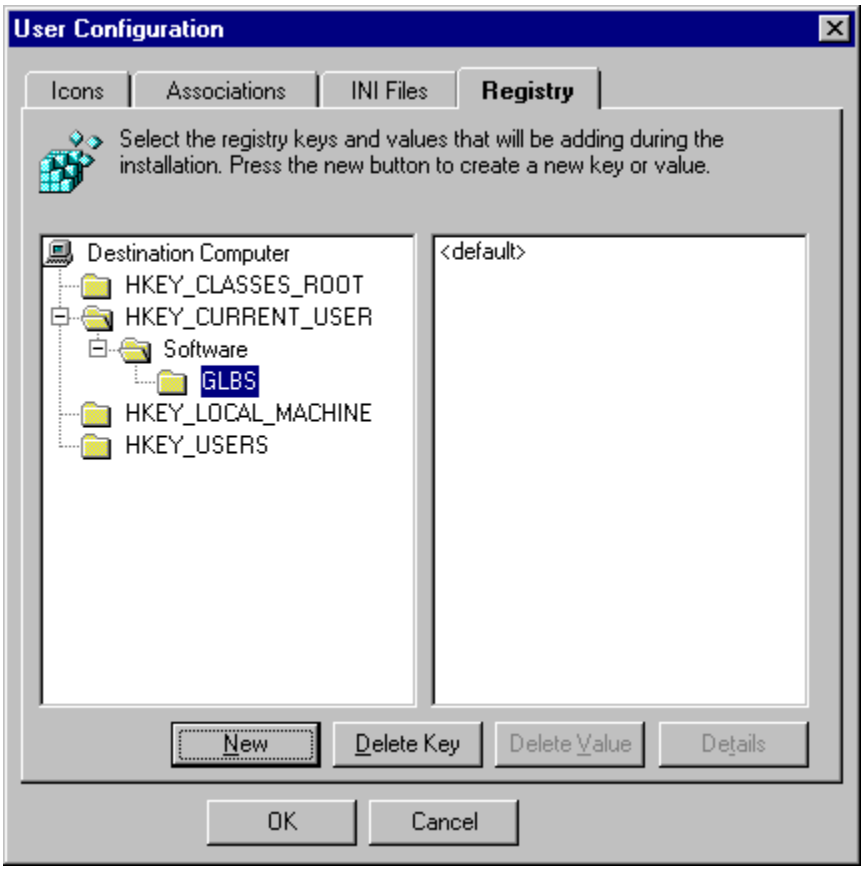


Installation Expert Option Scripts

Each one of thes options have a seperate script that will be included within your script. The default scripts can be found in the INCLUDE directory below the WISE root directory. NOTE: Any modifications that are may to the scripts may effect the final installation result.

Editing the Registry

The registry is a hierarchical structure. Each node in the tree is called a key. Each key can contain subkeys and data entries called values. Key names cannot include a space, backslash (\), or wildcard character (* or ?). In the HKEY_CLASSES_ROOT key, names beginning with a period (.) are reserved for special syntax (filename extensions), but you can include a period within a key name. The name of a subkey must be unique with respect to its parent key. Key names are not localized into other languages, although their values may be.



Installation Expert Registy Editing

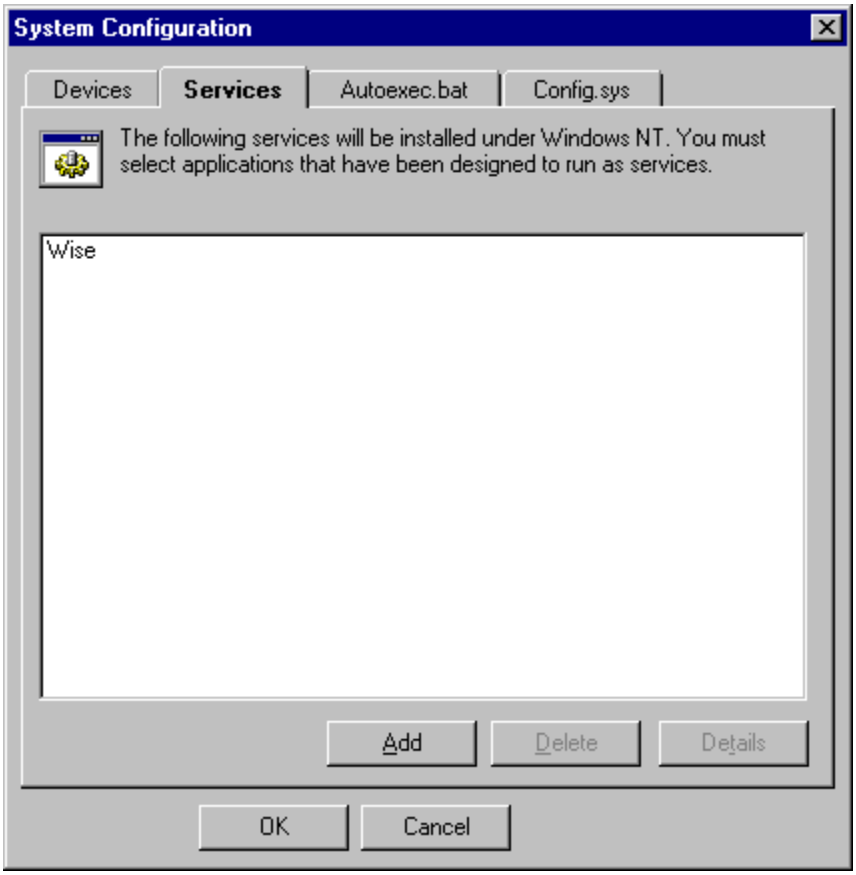
This dialog functions very much like Regedit. The registry that you see is a representation of the changes made to the destination computer. You may create any number of sub-keys and values within the sub-keys. The default type of value created is String, you may change this by clicking on the details button.

A key can have any number of values. A value entry has three parts: the name of the value, its data type, and the value itself. Value entries larger than 2048 bytes should be stored as files with their filenames stored in the registry.

When the user installs your application, register keys for where application data is stored, for filename extensions (file associations), icons, shell commands, OLE registration data, and for any special extensions.

Adding Services

This allows you to add a Service (background process) to a Windows NT system. You should consult Microsoft's Windows NT documentation for more information about creating a service.

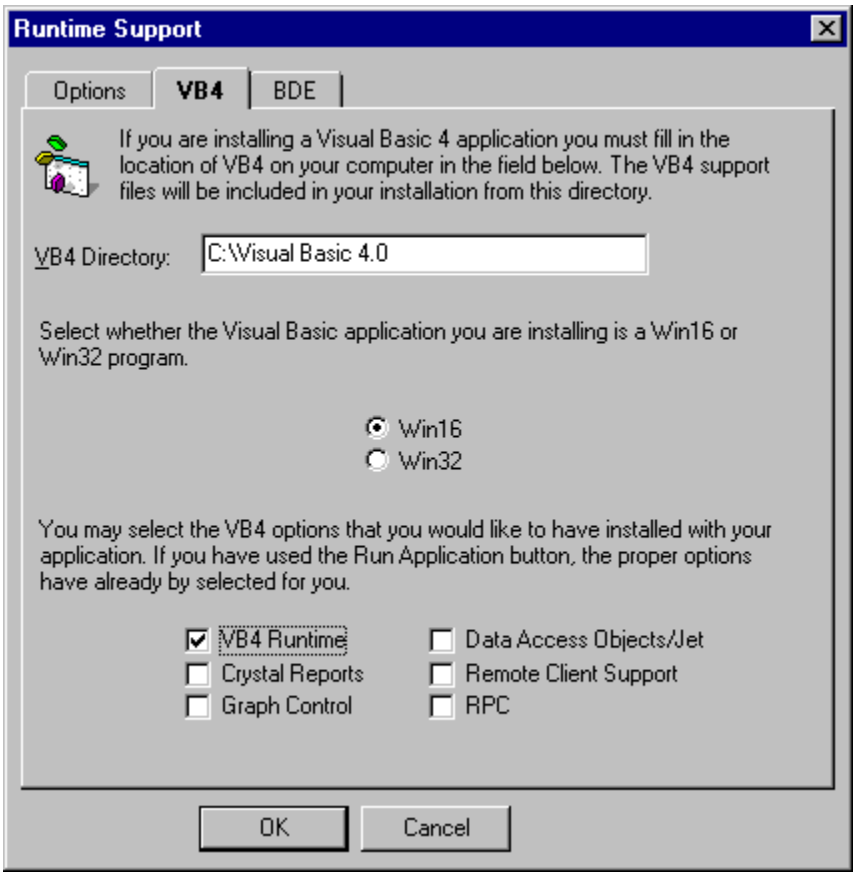


Installation Expert Adding Services

.When you press the Add button you will be prompted for which service to add. This file must already have been selected to be installed. The file name will be used for the Service Name and Display Name as a defualt. You may click on the details button to define additional information about your service.

Visual Basic 4.0 Runtime Support

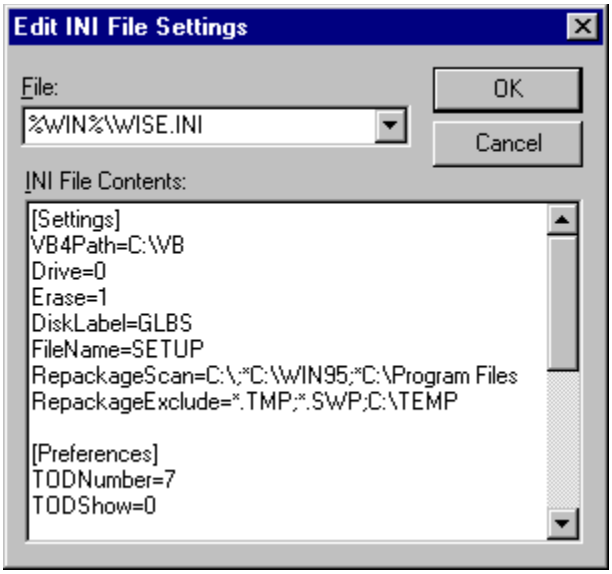
WISE supports both VB4-16 and VB4-32 installations. Some of the more common components are listed at the bottom of the dialog. You need to specify which directory your VB system is installed in so that WISE may know where to retrieve the runtime files. You may select which run time components of Visual Basic 4.0 you would like to install. For more information refer to your Visual Basic 4.0 user manual.



INI File

This script item should be used to add, delete, and modify the values in various Windows INI files.

NOTE: You should NOT use this script item to edit device entries in the SYSTEM.INI file! This should be used to edit the WIN.INI or your private INI files only.



INI Filename

Enter the name of the INI file you want to edit. If you leave this field blank, the WIN.INI file will be edited. You can use variables to prefix the filename if you want to edit INI files in program directories.

INI Section

This field holds the name of the section to edit. This field is required.

INI Item Lines

You place items into this list to edit the INI file. You should type the information in the edit field exactly how you would like it to appear in the INI. You may specify one or more sections within the edit box. Do not use hardcoded pathnames in your ini, you must always use variables.

Icon Number

If your executable file has more than one icon in it, you can select the icon number in the executable file with this field.

Show All Files

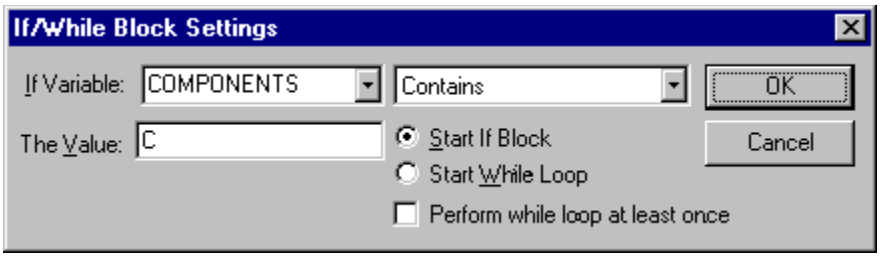
Normally, only your installations programs are displayed. Check this box to show all of your installation files.

Icon Path

This is the pathname of the icon file to use for display. If this is blank, the icon is pulled from the file itself.

If Statement

The if statement can be used to check the contents of a variable and branch depending on the value.



If Compare

The type of comparison to make is listed in this field. The contains comparison can be useful when used with a radio dialog or a component dialog.

If Value

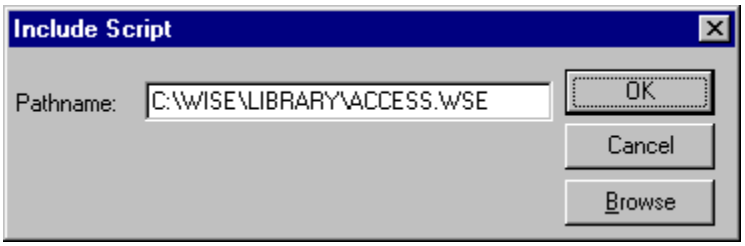
The value to compare against the variable. This value is case sensitive, that is the value 'a' is different that the value 'A'.

If Variable

This field holds the name of the variable to check. This variable must have been previously created.

Include Script

This script item will include another installation script into the current script. This can be used to create libraries of often used script items that can be re-used by many of your installations.



Include Pathname

You should enter the full pathname of the installation script that will be included into this field.

Include Sub-Directories

Checking this box will cause all matching files in all of the directories below the directory you specified to be scanned as well.

Include Sub-Directories

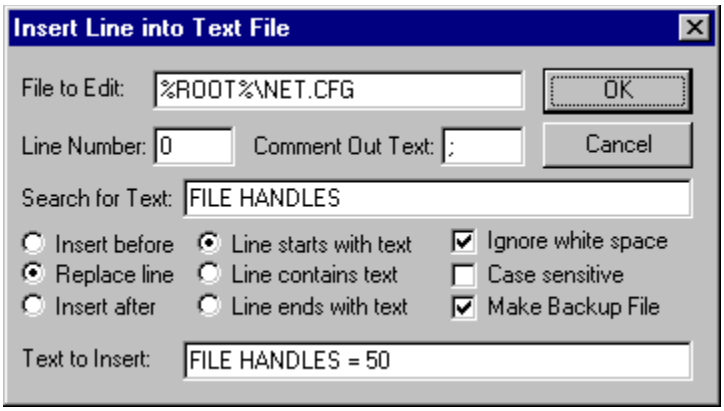
Checking this box will cause the installation to scan all of the directories below the directory you specify for matching files.

Insert Before

When this box is checked, the new script item that you all will be inserted before the currently select script item. If this box is not checked, the item will be added after the current item.

Insert Line into Text File

This script item will insert a new line of text into a text file. If the file does not exist it will be created.



Install DirectX

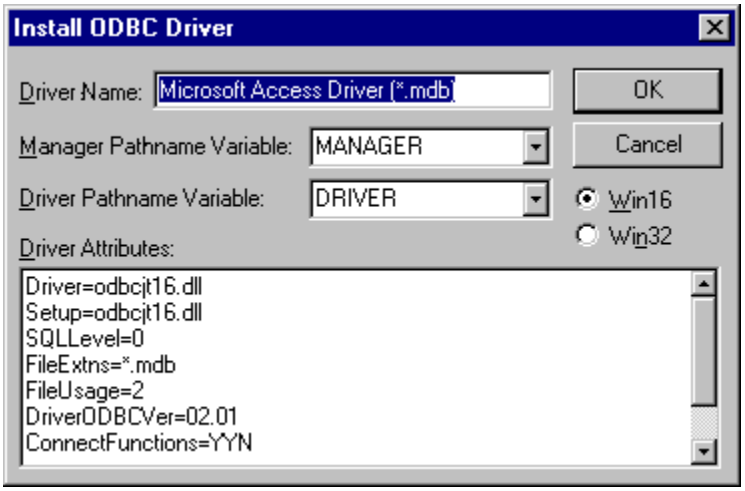
This script item can be used to install DirectX (DirectDraw, DirectSound, or DirectPlay) drivers onto a computer. It calls the Microsoft provided DirectXSetup API call within the DSETUP.DLL file. This script item can only be used under Windows 95 or Windows NT 4.0 or higher.

To install DirectX you must install the DirectX installation files (this includes a directory sub-tree of files) and the DSETUP.DLL as well as DSETUP16.DLL files. The DSETUP.DLL and DSETUP16.DLL files must be in the same directory and the entire DirectX sub-tree must be available. In addition, the DSETUP16.DLL file must be in the same directory as the installation EXE.

- DSETUP.DLL Pathname: This field holds the full pathname on the destination computer where the DSETUP.DLL file is located. The DSETUP16.DLL file must be located in the same directory. For example, when installing DirectX from a CD-ROM this field could be set to %INST% \REDIST\DSETUP.DLL.
- DirectX Directory Path: This field holds the directory name where the DirectX installation files are installed. For example, when installing DirectX from a CD-ROM this field could be set to %INST% \REDIST\DIRECTX.
- Installation Options: It is recommended that you select the Complete Installation of DirectX box. Otherwise, you can select to install DirectDraw, DirectSound, or DirectPlay. If you check the Reinstall DirectX Files the driver files will be reinstalled even if the same version of the files are already on the destination computer.

Install ODBC Driver

This script item will install an ODBC driver and/or the ODBC driver manager INI files. The ODBCINST.DLL and ODBCINST.HLP files must already be installed before you use this script item. You must install the ODBC.DLL file into the directory returned into the Manager Pathname Variable. You must install all of the files for the ODBC driver into the directory returned into the Driver Pathname Variable. If ODBC could not be installed, the returned variables will be blank.



Install Log Pathname

Normally the INSTALL.LOG file is placed into the first directory that files are copied to. You can place the full pathname (including the filename) of the INSTALL.LOG into this field to place the INSTALL.LOG into another directory. The directory must exist for the INSTALL.LOG to be placed into that directory. You should create the directory before you install any files to the destination computer.

Installation Password

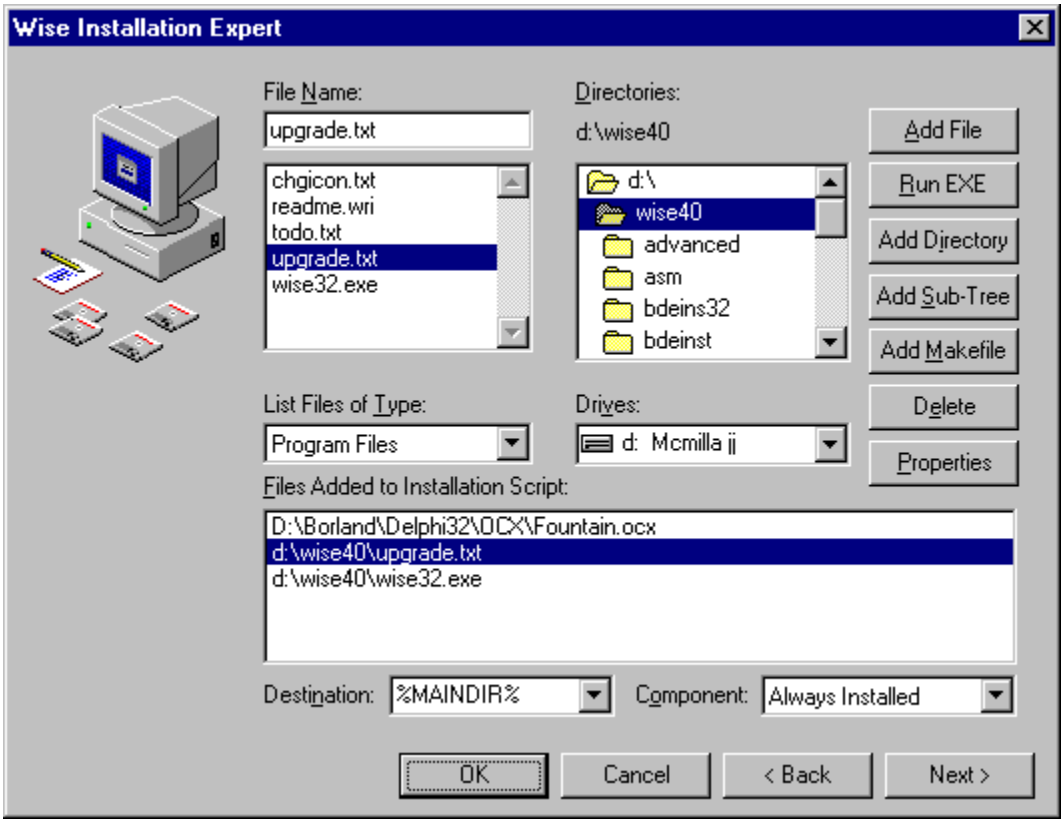
The installation password can be used to protect part or all of your installations from being installed without the user entering the proper password. If you prompt the user for the password from the installation you must place the password into the variable PASSWORD. If the PASSWORD variable is blank or incorrect the installation will prompt the user for a proper password when a file that requires a password is first encountered.

Installation Executable

The Windows self-installating executable that is created by the Wise Installation System.

Installation Expert - Add Files

This dialog adds files to your installation.



Installation Log

A log is created during the installation that describes the changes that were made to the destination system. It lists the files that were copied, the icons that were added, and the INI files that were edited.

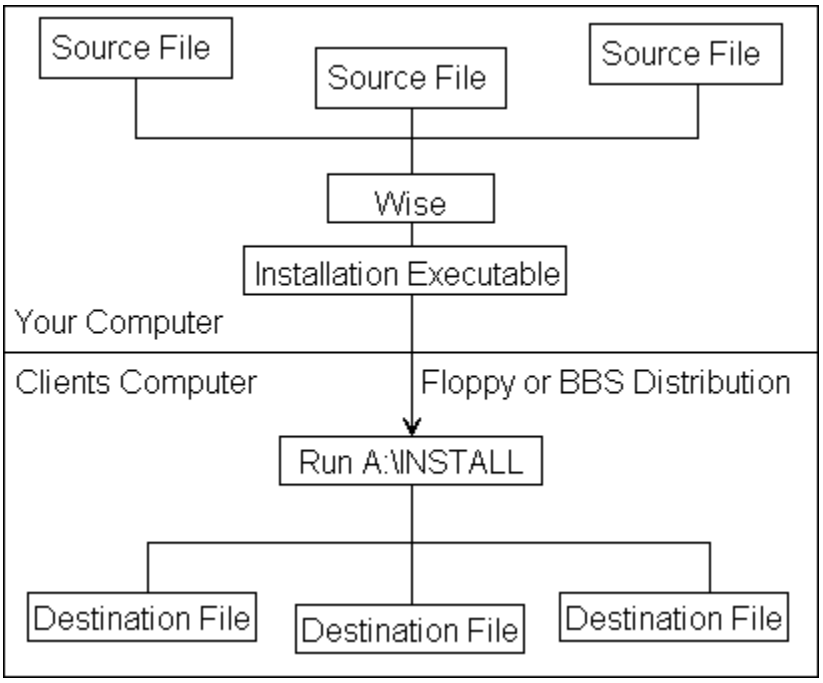
The installation log is called INSTALL.LOG and it is placed in the same directory as the first file that you copy to the destination system. You should make sure that your first file is copied into your main application directory.

Installation Process

There is a two step process in the installation. First you (the programmer) use Wise to create an installation script. After you have created the installation script, you press the Create button to copy all of the files into an installation executable. You can then upload this executable to a BBS or place it on a floppy.

Your client (the installer) then receives the installation executable from you either via modem or floppy. They run the installation executable under Windows (either via Program Manager File->Run or File Manager double-click). The installation executable then extracts the files from itself and copies them onto the client's computer.

The following graphic describes the installation process:



International Orders

If you are ordering outside of the United States, you may contact one of the following distributors:

Australia
GUI Computing Pty. Ltd.
Phone: +61-03-804-3999
Fax: +61-03-804-3414
26 Cato Street
Hawthorn 312
CIS: 100012,3353

Aeronaut
Phone: +61-02-436-1175
Fax: +61-02-436-1184
Ground Floor, 48 Chandos Street
ST. LEONARDS NSW 2065

Denmark
Danish Know*How ApS
Phone: +45-47-10-03-73
Fax: +45-42-17-71-72
Mirnosevej 4
DK 3650 Olstykke

France
KDD
Phone: +33-1-30-15-00-05
Fax +33-1-30-15-00-10
11, Allee des hautes bruyeres
78290 Croissy sur Seine
CIS: 101341,3222

Germany
Downsizing Systems
Phone: 0221-951479-0
Fax: 0221-951479-99
Venloer Str. 315
50823 Koln
CIS: 100115,1547

Nexus Software GMBH
Phone: 0751-351340
Fax: 0751-351345
Zeppelinstr 2
D-88212 Ravensburg

ProLib Software GmbH
Ludwig-Thoma-Str. 13
83358 Seebruck
Phone: (08667) 888321
Fax: (08667) 888328
CIS: 100015.676

Hungary
EastCom
Phone: +36-30-532-100
Fax: +36-62-326-133
Bologasszony sgt 31
PO Box 730
H-6722 SZEGED
CIS: 100324,76

Italy
Silicon Valley
Phone: +39-49-871-34.44
Fax: +39-49-871.30.55
Mirco Sarto
22, Via Vicenza
35138 Padova

Netherlands
AIMED Solutions
Phone: +31 (0) 30 2613559
Fax: +31 (0) 30 2613559
Kaap Hoorndreef 10
NL-3563 AS Utrecht
The Netherlands
CIS: 100723,2271

Norway
Kvam Data as
Phone: +47-51 62 37 66
Fax: +47-51 62 39 85
Nikkelveien 14
Postbox 1101 Lura
N-4301 Sandnes

South Africa
Barras-Baker & Associates
Phone: +27-011-802-3797
Fax: +27-011-802-3741
1 Woodmead Avenue
PO Box 781093
Woodmead
SANDTON 2146

Spain
Visual Programming
Phone: +34-87 22 99 63
Fax: +34-87 24 96 38
La Gaviota 2
24191 Pinilla - LEON
CIS: 100024,1475

Sweden
Linsoft AB
Phone: +46-13 31 15 88
Fax: +46-13 13 56 95
S:t Larsgatan 9D

S-582 24 Linkoping

Switzerland
Visual Tool Shop AG
Phone: +41 41-420 00 11
Fax: +41 41-420 01 11
Schachenweg 13
CH 6030 Ebikon

United Kingdom
QBS Software Ltd.
Phone: +44-181-956-8000
Fax: +44-181-956-8010
10 Barley Mow Passage
London W4 4PH
CIS: 100016,573

Icon Name

This is the name of the icon that you are adding or deleting. If the icon already exists in the Program Manager group you have specified, it will be updated with the new information.

Copy Default Language

This checkbox controls whether the text of the default language will be copied to all other languages in your script items. This can be useful when translating from one language to the next. For example, if you set english as your default language (in the far left column), the english text you enter would be copied to any other languages you define in your script. After you have translated the english to another language, you would delete the english text.

Default Language

This column defines which of the languages is the default. The default language can be copied to any other with the Copy Default checkbox.

Language Dialog Text

This field holds the text that is displayed in the language dialog that is initially displayed. You should instruct the user that they should select the proper language from the list below (you may want to repeat the destription in a few different languages).

EXE Language Order

This column controls which of the languages in the installation script will be written into the installation executable. If you know which language your client speaks, you may want to create an installation executable with only that language in it. The number appearing next to the checkbox defines the order that the languages will appear to the person running the installation executable.

Langage Messages

This list holds all of the messages that can be displayed during the installation. You should select an item from this list to change its value.

Language Name

This list holds the names of all of the languages that Wise supports. These languages will be detected on the destination computer. Select a language from this list to edit the messages for that language.

Language Message Text

This field holds the actual text of the message that will be displayed on the destination computer.

Language Dialog Title

This field holds the title of the dialog box that will be displayed when the installation initially starts. This dialog will let the user choose which language the installation should use.

Language Translated Name

This field holds the name of the language translated into the language itself. This value will appear in the list of languages that can be selected when the installation starts.

Languages Used

This column selects which of the 15 available languages your installation script will use. Each language you select must have been translated in the Installation Messages dialog.

Language INI Name

You may specify the path to the INI that contains the language translations for the installation expert.

Languages Overview

Wise supports multiple languages in a single installation script. When the installation executable is first run, the user is prompted for the language they would like the installation to be in. There are two steps required in creating a multi-lingual installation executable:

1 Create language specific messages for the installation by selecting the Languages menu and Installation Message menu item. You will need to translate all of the messages that may appear during this installation.
2 Define which of the 15 languages you would like you installation script to use. To do this, select the Languages Menu and Scipt Languages item.
3 Create the language specific messages for an installation script by changing the Current Language list on the main screen.

Languages Dialog

This dialog box edits the messages that may be displayed during the installation. You may customize the messages or translate them into different languages. You may use the ampersand character (&) with the OK and Cancel button texts. You can modify the messages that installations will display by selecting a language (from the Language Name list) and a message (from the Messages list) and editing the text in the Message Text field.

The Dialog Title and Dialog Text fields are used when an installation includes more than one language. These fields will be displayed to allow the user to select which language to run the installation in.

Copyright/License/Warranty

License Agreement

You should carefully read the following terms and conditions before using this software.

Demo Version

You are hereby licensed to: use the Demo version of the software for a 30 day evaluation period; make as many copies of the Demo version of this software and documentation as you wish; give exact copies of the original Demo version to anyone; and distribute the Demo version of the software and documentation in its unmodified form via electronic means. There is no charge for any of the above.

You are specifically prohibited from charging, or requesting donations, for any such copies, however made; and from distributing the software and/or documentation with other products (commercial or otherwise) without prior written permission.

Unregistered use of the Wise Installation System after the 30-day evaluation period is in violation of federal copyright laws.

License and Royalties

One registered copy of the Wise Installation System may either be used by a single person who uses the software personally on one or more computers, or installed on a single workstation used nonsimultaneously by multiple people, but not both.

You may access the Wise Installation System through a network, provided that you have obtained individual licenses for the software covering all workstations that will access the software through the network.

You may redistribute the installation programs created by the Wise Installation System freely and without royalties to Great Lakes Business Solutions Inc. In addition, you may freely redistribute the UNWISE.EXE program with those installation programs created by the Wise Installation System. Software distributors may not use the Wise Installation System to redistribute the software of a third party.

Governing Law

This agreement shall be governed by the laws of the State of Michigan.

Disclaimer of Warranty

THIS SOFTWARE AND THE ACCOMPANYING FILES ARE SOLD "AS IS" AND WITHOUT WARRANTIES AS TO PERFORMANCE OF MERCHANTABILITY OR ANY OTHER WARRANTIES WHETHER EXPRESSED OR IMPLIED. Because of the various hardware and software environments into which Wise may be put, NO WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE IS OFFERED.

Good data processing procedure dictates that any program be thoroughly tested with non-critical data before relying on it. The user must assume the entire risk of using the program. ANY LIABILITY OF THE SELLER WILL BE LIMITED EXCLUSIVELY TO PRODUCT REPLACEMENT OR REFUND OF PURCHASE PRICE.

Shell Link Description

If the shell link you are creating is an MS-DOS program, you can use this field to describe the shell link.

Shell Link Destination Pathname

This field holds the name of the new shell link that will be created. The link should always end with the .lnk extension. If you create a link in the %WIN%\Desktop directory the link will appear on the users desktop. Creating links in the %WIN%\Start Menu\Programs directory will add the link to the Start menu.

Shell Link Icon Number

This number sets the icon number within the Icon Pathname file to use.

Shell Link Icon Pathname

Normally, the icon for the shell link is taken from the Source Pathname. You can use the icon of another file by setting this field.

Shell Link Hot-Keys

You can set a Hot-Key that will cause the linked program to execute when it is pressed.

Shell Link Command Line Options

This field holds any command line options for the shell link. These options are passed to the shell link program.

Shell Link Window Size

These buttons select whether the programs inital window will be normal, maximized, or minimized.

Shell Link Source Pathname

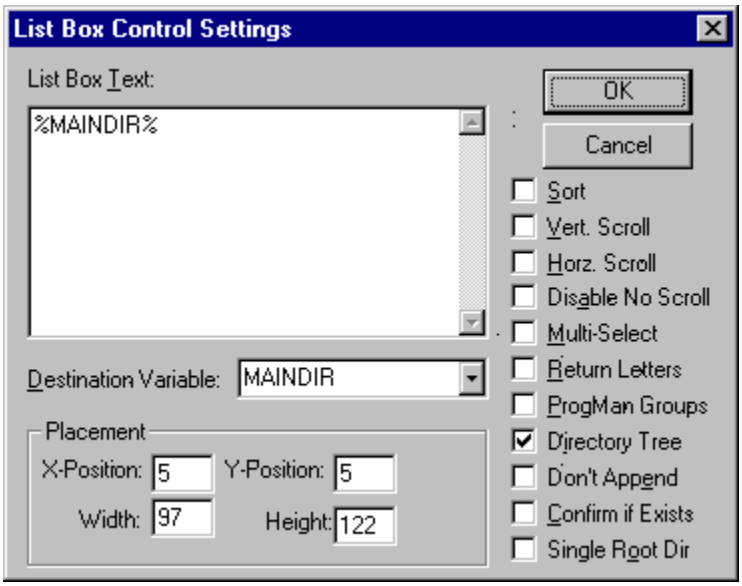
This field holds the name of the file that will be linked (i.e. the name of an already existing file). The new shell link will reference this file.

Shell Link Working Directory

This field contains the working directory for the shell link. This directory will be set as the default directory when the program is executed.

Listbox Control

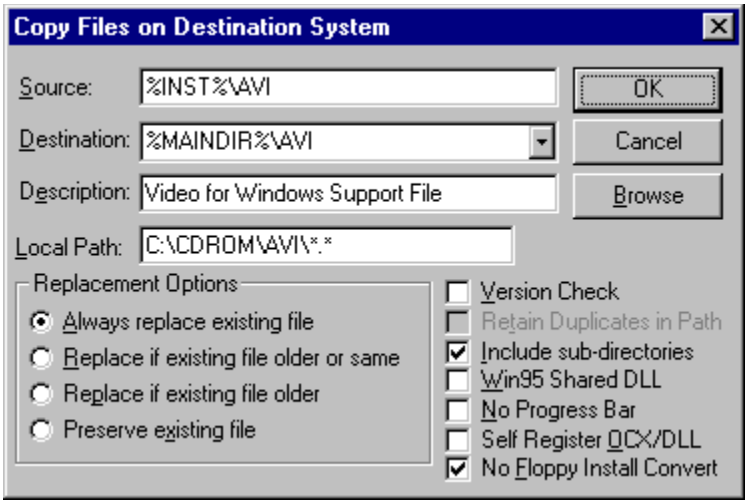
The listbox control allows the user to select one or more choices from a list. The actual text of the selection or a letter representing the position of the choice can be returned into the specified variable.



Copy Local File(s)

The Copy Local File(s) script item can be used to copy files that are not compressed inside of the installation executable. You can use this script item to copy a license or serial number file from the installation floppy to the users hard disk. You can also use this script item to copy files from a CD-ROM to the hard disk.

There are two ways to copy multiple files with this script item. If you specify a local pathname ending in a wildcard in the Local Path field then the files on your computer will be used to get the list of files (and their sizes) that will be copied when the installation is executed. If you leave the Local Path field blank and specify a wildcard in the Source field the files will be copied but the progress bar will not be updated during the copy. In both cases you must specify a directory name in the Destination field.



Local Path

Wise needs to create a list of the files that will be copied and their sizes. This field holds the pathname or wildcards of the files on your computer that will be copied. If you leave this field blank, you can only copy single files and the progress bar will not be updated as the file is being copied (the size of the file will not be known until install time). If you place a directory name of use wildcards in this field, the Source and Destination fields should contain directory names.

Long Filename Support

The Windows NT and Window 95 operating systems support filenames longer than the DOS standard 8.3 filenames. The installations created by the Wise support long filenames under these operating systems. There are a number of compatibility issues that must be considered when using long filenames:

- Some programs do not support long filenames, you should convert any long filenames to short filenames (via the Set Variable script item) before writing your long filenames to INI files or the Registry.
- The Windows NT Program Manager version 3.1 does not support long filenames when adding icons. You should convert your filenames to short filenames before adding icons to the Program Manager.
- You can use long filenames in your installation even if your program does not support long filenames. Make sure you convert any references to your program directory before writing them to your applications INI file.
- If a long filename contains spaces it will not be handled properly on a command line (i.e. when adding an entry to the Add/Remove applet). You should convert long filenames to short filenames before trying to execute a program.

Support Long Filenames

The Windows NT and Windows 95 operating systems support long filenames. You can check this box for your installation to support creating files and directories that have long filenames. You must ensure that your program can handle the long filenames that may be entered by the user before selecting this option.

Main Menu Bar

The Wise Installation System menu bar.

Main Screen Toolbar

The toolbar icons can be used to open, save, cut, copy, paste, compile, test, and run your installation.

Make Floppy Disks Button

This button will copy the installation executable file to a floppy. If you have split your installation executable, you will be prompted for a new floppy for each file.

Message/Message Block

The message item will display a message on the screen. It can also be used to ask the installer a Yes/No question. When you select the Start Block checkbox a Yes/No dialog will be displayed. If the user answers "No" then none of the script lines will executed until the matching End Message Block is reached.

Message Box Character Set

This field holds the character set number for the text that is displayed in Display Message script items. If you translate your installation to Japanese you should set this field to 128 (and set the Message Box Font to MS Gothic), otherwise this field should be set to 0.

Message Box Font

This field holds the name of the font to use for displaying text in Display Message script items and internal Wise messages.

Message Box Font Size

Set this field to the point size of the font to display in Display Message script items and internal Wise messages.

Minimum Buffers

You should enter the smallest number of buffers your program will run with. If the number of buffers in the CONFIG.SYS is lower that this number, it will be increased to this number. You may leave this field blank to leave the number of buffers unchanged.

Minimum Files

You can set the minimum files in the CONFIG.SYS with this box. If you leave the box empty the Files= line will not be changed.

Destination Pathname

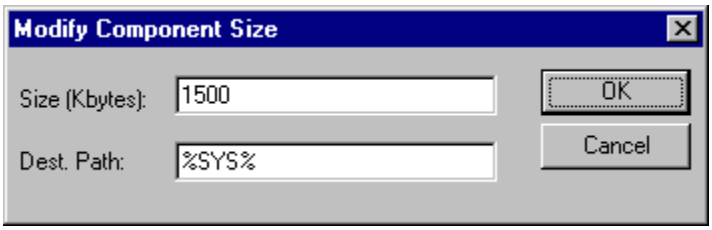
This field holds the pathname (starting with a variable reference) that will receive the files to be installed. This information is used by the Check Disk Space script item.

Component Size

Set this field with the amount of additional K bytes to add to the size of the component.

Modify Component Size

If you are calling external installations you can use the Modify Component Size script item to have the files installed by the external installation taken into account by the Select Components and Check Disk Space script items.



Move Down Button

Use this button to move the currently selected item down in the list.

Move Up Button

Use this button to move the currently selected item up in the list.

Asterisk Icon

An icon of an asterisk will be displayed in the upper left hand corner of the message dialog box.

Start Block

If this box is checked the message will not display OK and Cancel buttons. Instead it will display Yes, No, and Cancel. If the user enters No, all script items between this item and the matching End Message Block item will be ignored.

You can use this feature to implement conditional installation of components of your software. Message blocks can be nested within each other.

Exclamation Mark Icon

An icon of an exclamation mark will be displayed in the upper left hand corner of the message dialog box.

No Icon

Selecting this radio button will display the message without any icon at the upper left corner.

Question Mark Icon

An icon of a question mark will be displayed in the upper left hand corner of the message dialog box.

Stop Sign Icon

An icon of a stop sign will be displayed in the upper left hand corner of the message dialog box.

Message Text

This is the actual text that will be displayed on the screen. You can use the Ctrl-Enter key combination to put blank lines in the message box. You can use variable names in your message text.

Message Title

This will appear in the window title of the message that is displayed on the screen.

Network and CD-ROM Installations

In network and CD-ROM installations uncompressed files are copied from a network server disk or a CD-ROM onto a users computer. The files should be copied with the Local File Copy checkbox on the file script item. The source field should contain the location of the file to copy on the destination computer. The %INST% variable holds the pathname of the installation executable. This variable can be used to locate the files to copy.

New Item

This will create a new item at the current selected point.

New Folder

This will create a new folder inside your installation. You may then highlight this folder and use the Add File button to place files into it.

Don't Append Directory Name

Normally, the default directory name is appended to the end of the default directory is the directory name field. This checkbox will suppress this feature. This can be useful if you want the user to select an existing directory.

No Cancel Button

If this box is checked, the message will not have a Cancel button in it.

No Floppy Install Convert

Check this box if the Copy Local File should not be converted to an Install File(s) if the Convert CD-ROM to Floppy Install box is checked in the properties dialog.

No Background Gradient

No background gradient will be displayed during the installation. This option can be used if a full screen graphic or tiled graphic will be displayed througout the installation.

No Installation Log

Checking this box will cause no installation log file to be created. This can be useful if you are copying files only to the Windows, System, or Temporary directories. You cannot use the Wise Uninstaller if you check this box.

Do not scale bitmap graphic

Normally a bitmap graphic will be scaled if the destination computer has selected to display large fonts. You can check this box to suppress this scaling.

Suppress Version Error

This option will allow you to check the Version Check option in the install file script item on files that do not have version resources. When a file is detected without a version resource version checking is suppressed an no error is generated.

OCX Registration

If you would like to register an OCX already on the system fill the pathname above with that OCX and check the "Queue Existing file for self-registration" button. If you would like register all pending OCX files fill the description above with a message and check the "Register all pending OCX/DLL's" button.

Clients Computer

This is the computer that will receive the copy of the software that you are distributing.

Display Configuration Dialogs

If this box is checked the source configuration dialogs will be displayed to the user. These dialogs allow the user to configure the source for their computer system. If this box is not checked, the source will be installed silently.

ODBC Type

These radio buttons select whether the ODBC drivers are installed for Win16 or Win32. The Win16 ODBC installation require the Win16 version of ODBCINST.DLL and the Win32 ODBC installation requires ODBCCP32.DLL.

Driver Name

This is the name of the ODBC driver that this data source uses. This driver must have already been added to the system as well as it's support files.

Driver Attributes

This field contains a list of attributes for the ODBC driver. This information is contained in the ODBCINST.INI file.

Driver Name

This field holds the name of the ODBC driver to install. This name will appear in the ODBC drivers list.

Driver Pathname Variable

This field holds the variable that will receive the pathname that all of the driver files should be placed into.

Manager Pathname Variable

If this field is not blank, the directory that the ODBC.DLL file should be installed into will be placed into this variable. If you are installing multiple ODBC drivers, only the first driver needs to fill in this field.

Source Attributes

The attributes for the ODBC data source, these can be found in the ODBC.INI file in the Windows directory.

Source Name

This field holds the name of the ODBC data source. The value of this field will be displayed in the ODBC data sources list.

Destination File

These files are copies of the ones that are on your computer. They are created on your clients computer by the installation executable.

Floppy or BBS Distribution

The installation executable must be delivered to your client for them to execute it. You can place it on a BBS or other on-line service. You can also mail it to them on any type of floppy disk or CD-ROM.

Installation Executable

The installation executable is a Windows EXE file (located in the same directory as your installation script) that contains all of the files and instructions from the installation script. This file is uploaded to a BBS or placed on a floppy and mailed to your client (the installer).

OK Button

Press this button when you are done with this dialog box. This button will save the changes that you have made to this dialog.

Run Installation Executable

Your client runs the installation executable on their computer. They usually select the Run item from the File menu in the Program Manager. They then type the name of your installation executable and press the OK button.

Source File

These are the files that comprise your program. The are usually the EXE, VBX, DLL, HLP, and other support files that are located on the programmer's computer.

Wise Installation System

The Wise Installation System reads the files that are given in the installation script and creates an installation executable.

Your Computer

The is the computer where the program that is to be distributed has been written. It holds the program and all of its support files.

Open/Close INSTALL.LOG

This script item can be used to control whether entries are made into the installation log file. In addition, this script item can be used to create a new installation log.

Order Form

Print Order Form

```
Wise Installation System Version 5.0: $199 each = ______
SmartPatch Add-In (requires Wise 4.0):$169 each = ______
Upgrade from Version 4.0 to 5.0:       $50 each = ______
Upgrade from Version 3.0 or earlier   $100 each = ______
One Year Maintenance Contract:         $95 each = ______
Michigan residents add 6% sales tax             + ______

Shipping (you MUST choose one of the following):
                         UPS Ground for U.S.  $8  ______
                     UPS Second Day for U.S. $12  ______
                       UPS Next Day for U.S. $18  ______
                      International Air Mail  $8  ______
       International Express Mail (3-5 days) $25  ______
                  UPS International Express *$50  ______
              (*Some locations may be higher)

Additional Requests:
Request BBS/FTP password to download (Free)   $0  ______

                                   Total payment  ______

            Name: _______________________ Date:___________

         Company: ______________________________________

         Address: ______________________________________

City, State, Zip: ______________________________________

         Country: ______________________________________

       Day Phone: _______________ Eve:_________________

      Fax Number: _______________

   Credit Card Type: ( ) Visa  ( ) MasterCard  ( ) Amex

   Credit Card #: _______________________ Exp Date:______

Cardholder Signature: ___________________________________

Electronic Mail address: ________________________________

How did you hear about the Wise Installation System?
_________________________________________________________

Comments:
```

Ordering Information

Wise Installation System Order Form/Invoice

Ordering by phone: You can place orders with Visa, MasterCard, or American Express by calling toll free 1-800-554-8565 (International callers: 313-981-4970) or by faxing this order form to 313-981-9746.

Ordering by mail: To order by mail send this order form and either 1) a personal or cashiers check or 2) a Visa, MasterCard, or American Express credit card number to:

Great Lakes Business Solutions, Inc.
2200 N. Canton Center Rd., Suite 220
Canton, Michigan 48187

Payments must be in US dollars drawn on a US bank, or you can send international postal money orders in US dollars.

Any questions about the status of the shipment of the order, refunds, order options, product details, technical support, volume discounts, dealer pricing, site licenses, etc, should be directed to 313-981-4970.

Maintenance Contract: The Wise Installation System is updated regularly to include features for new operating systems and to enhance the product. The maintenance contract provides one year of upgrades that will be sent immediately upon the shipment of a new version of Wise. In addition, the maintenance contract entitles the user to telephone based technical support past the initial 30 day period.

Order Form
International Orders

Overview

The Wise Installation System makes it easier for you to distribute your programs, and easier for your clients to install those programs. The Wise Installation System also supports splitting your installation file across multiple disks. You can split the installation executable up to make downloading from a BBS easier. With other installation systems you must do the following to install a software package downloaded from an On-Line service:

- Find the correct "Dearchive" program if the file is not a self-extracting DOS program
- Create a temporary directory
- Execute the self-extracting program or run the "Dearchive" program with the proper command line arguments
- Switch back to windows and execute the Windows Setup program
- Delete the temporary directory if the installation was successful

On the other hand, when you install a Wise executable you need to do the following:

- Double-Click the installation programs name in the File Manager

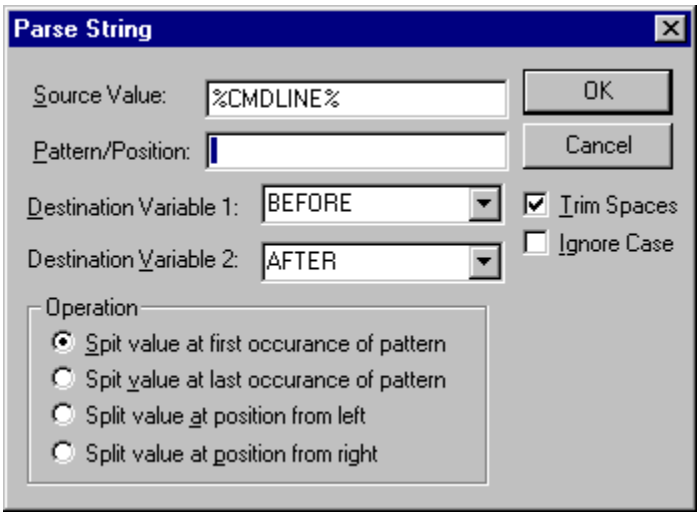
Wise Features
Installation Log
Variables
Installation Process
Installation Executable Options

Command Line

This is the command line to execute when the icon is double-clicked in the Program Manager. You should use a variable name to select where the program is located. For example: %DESTDIR% \MYPROG.EXE

Parse String

This script item can be used to split a string into two parts. The two destination variables will hold the two sections of the string that was parsed.



Destination Variable 2

Enter the name of a variable to receive the ending portion of the parsed string. You can leave this field blank.

Destination Variable 1

Enter the name of a variable to receive the first portion of the parsed string. You can leave this field blank.

Ignore Case

Checking this box will cause a case insensitive search to be performed when splitting the string.

Operation

These buttons select how to split the source value. The first two radio buttons will split the string based on the patterned entered. The second two buttons will split the string based on the number entered into the Pattern/Position field.

Pattern / Position

This field holds either the pattern to search for in the source value or a numeric value (starting at 1) that indicates where to split the string.

Source Value

Enter the value to parse into this field. This field can contain variable references.

Trim Spaces

Check this box to trim spaces (both leading and trailing) from the destination variables.

Existing File Pathname

This field is used by SmartPatch to locate the existing version of the file that is to be updated. The file that is specified by this field must exist on the destination computer and must be identical to one of the files listed in the Previous File Versions list. If wildcards are used in the Source field this field should point to a directory.

NOTE: The use of this field requires the SmartPatch Add-In for the Wise Installation System. The SmartPatch Add-In is a separately purchased product. Please contact GLBS, Inc. at 800-554-8565 or 313-981-4970 to order the SmartPatch Add-In.

Previous file versions

This field holds the list of files that are older versions of the file(s) to be installed. If wildcards are used in the Source field this list should be to directories. These files will be used to create a "patch" file that contains only the differences between the previous files and the file to be installed. The resulting installation is normally much smaller than a full installation. Since only the differences are being shipped, the computer the upgrade will be installed to must contain one of the files listed in this field. You may specify up to 255 previous file versions in this field.

NOTE: The use of this field requires the SmartPatch Add-In for the Wise Installation System. The SmartPatch Add-In is a separately purchased product. Please contact GLBS, Inc. at 800-554-8565 or 313-981-4970 to order the SmartPatch Add-In.

Place at Bottom

This box will place the graphic against the bottom of the installation window.

NOTE: Due to rounding errors in Windows metafiles this option may be slightly inaccurate when used in custom graphics with the Scale to Screen box selected.

Place at Right

Checking this box will place the graphic against the right side of the installation window.

NOTE: Due to rounding errors in Windows metafiles this option may be slightly inaccurate when used in custom graphics with the Scale to Screen box selected.

Copy Window Placement

This list selects where the copy dialog box will appear during the installation. If you want to place graphics on the screen during the installation, you can move the box to another part of the screen.

Play Wave File

This script item will load the specified WAV file into memory and play it. The Wave Pathname field should hold the full pathname of the WAV file. There must be enough memory to load the entire WAV file into memory to play it. The Loop continuously checkbox can be used to cause the WAV file to be played over and over. You can turn off playing of a WAV file by setting the Wave Pathname field to blank.

Confirm Dialog

This checkbox will cause the user to be prompted if they want to add the icons to the Program Manager. Otherwise, the icons will be added automatically.

Files to Add Icons For

You should select those files from this list to add icons to the Program Manager.

Default Group Name

This group name will be the default that is placed into the prompt that is displayed to the user.

Installation Expert - Add Icons

This dialog controls which icons will be placed into the Program Manager at the end of the installation. Enter the name of the Program Manager group that will be the default location of the icons. Select the files to add icons for from the list of files in this dialog.

Backup Replaced Files

You can check this box if you want to ask the user if the files replaced files during the installation should be backed up.

Branding

This checkbox controls whether the user is asked for their name and a company name during the installation. The name and company will be stored on the floppy disk the software was installed from.

Browse for ReadMe Path

Press this button to locate a text file to display as your read me file.

Software Components

This text box contains a list of the components of your software. If your software will be installed as a single component, leave this field blank. Each line should contain a label for the components. You will be prompted for the files that are associated with each component.

Default Directory

This directory is the default that the user will be prompted with during the installation. It should be a top level directory, normally located on the C drive.

OLE 2

This checkbox will copy the OLE 2 files onto the destination computer. You must have a working copy of OLE 2 on your computer to check this box.

ReadMe PathName

If you want to display a read me file before the installation begins, place the pathname of a text file into this field.

File Sharing via Share

This checkbox will cause file sharing to be installed onto the destination computer system. The file VSHARE.386 will be installed if share is not already installed.

Software Title

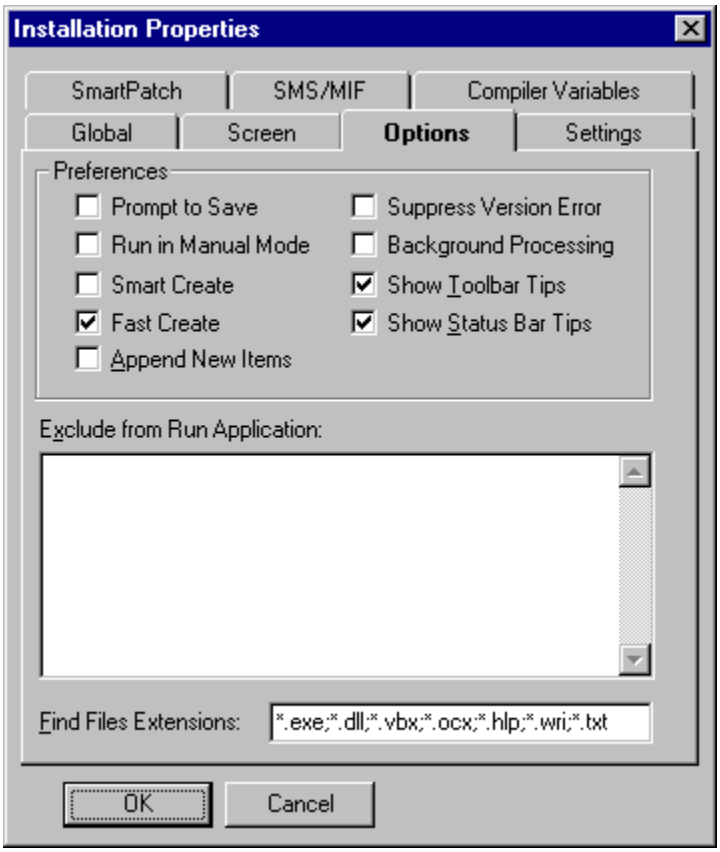
This field holds the title of your software package. An example of a software title is "Wise Installation System". This title will be displayed in messages and in the title bar during the installation.

Uninstall

You should check this box if you want the Wise Uninstall program to be copied to the destination computer. An icon will be added that will allow the user to remove the software from their computer.

Options

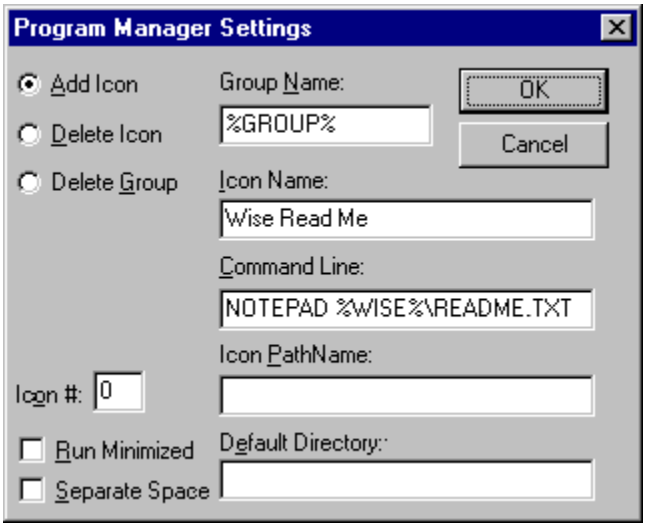
This property sheet contains the options of your installation script. To get more detailed information on the options, click on the dialog below.



Program Manager Icons

You can add this item to your script to add and delete icons from the Windows Program Manager. Click on the picture below for more information about the information you must provide.

NOTE: Not all Program Manager replacement shells allow parentheses "( )" and commas "," in the icon name. You should avoid using these characters in your icon names.



Progress Bar DLL

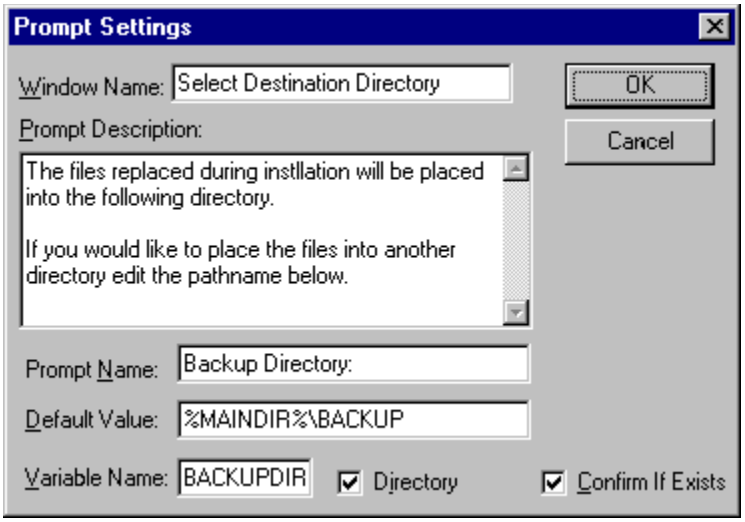
You may specify a custom DLL that will be used for the progress bar.

Progress Bar is Script Based

The progress of the installation can be based on three different criteria. You can base the progress bar on 100% being the entire installation (ie if only part of the files are installed the progress bar may not flow from 0 to 100%). You can base the progress bar on 100% being all of the script items being executed (ie each script item will get the exact same percentage of time on the bar regardless of how long the item may take). You can also base the progress bar on 100% being only the files selected for installation (ie this would show 0 to 100% for any combination of components installed).

Prompt for Text

This script item will prompt the installer for information. This is usually a directory name where they would like to install your program. It can also be a group name of a Program Manager group to add your icons to. If you choose, you can have a default value already filled in the prompt to make your installation easier.



Prompt Default Value

If you place a value in this field, it will be pre-filled in when the prompt is displayed.

Prompt Description

The text entered into this field give the installer information about what they should enter into the prompt.

Directory Name

If this box is checked, any trailing backslashes will be deleted from the string that the user enters in the prompt. This is useful when prompting for a destination directory name.

Prompt Name

This should contain a description of the information you are asking for. For example, if you want to ask for the directory to install you program, you should enter: Enter Directory to Install My Program:

NOTE: You should end the title with a ":" colon.

Prompt To Save

If this checkbox is checked you will be prompted if you want to save your installation script each time you create a new installation executable. If this box is not checked the installation script will be automatically saved for you each time you create a new installation EXE.

Prompt Variable Name

This is the name that you will refer to this prompt as in your other script items. This must not contain a percent sign (%). It should be no longer than 14 characters.

Prompt Window Name

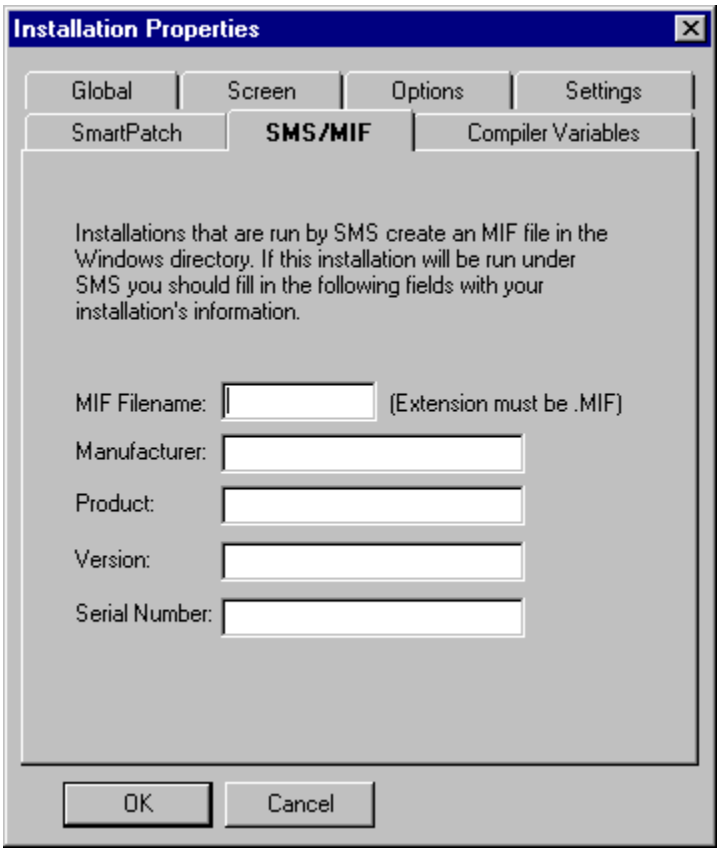
This field holds the name of the window that the prompt will appear in. An example of a prompt window name would be: Select Destination Directory

Compiler Variables

Compiler variables may be used within the installation script to perform branching or as variable information required at compile. The compiler variables have the same appearance as standard variables however they need to be defined in the Installation Properties, Compiler Variables dialog.

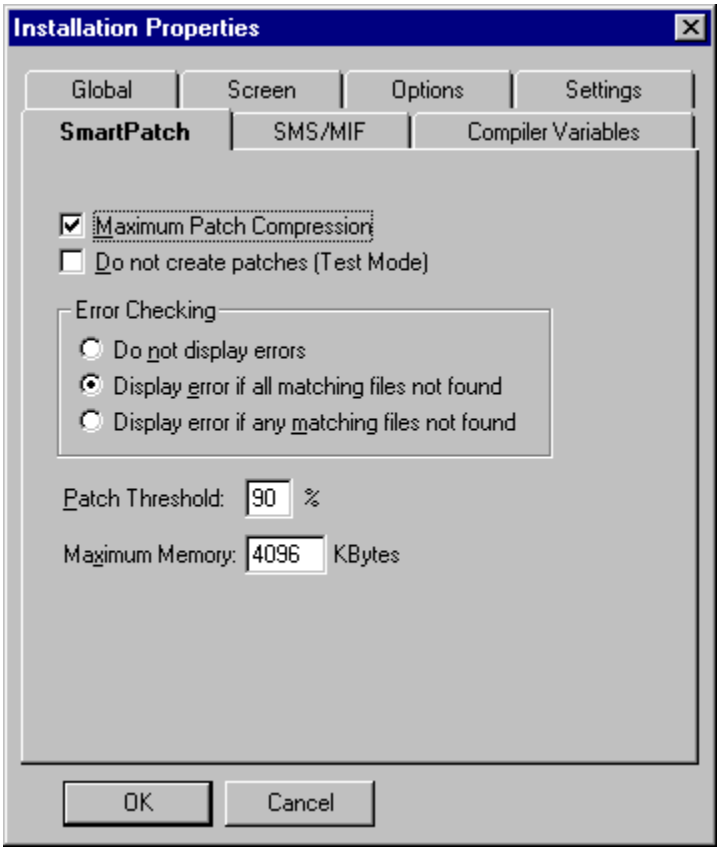
SMS Properties

You can specify the name of the MIF file created for use with an SMS type distribution system. This file will be created in the Windows directory. These fields will be inserted into to the MIF file that is created at the end of the installation. You should consult your SMS manual for information about the values of these fields.



SmartPatch Properties

The SmartPatch Add-In for the Wise Installation System can be used to install upgrades to software and datafiles without having to ship an entire new copy of the files. The differences between the older versions of the files and the latest version are placed into the installation EXE. This normally results in a much smaller installation as well as providing secure upgrades, only users that have the previous versions of the software can perform the upgrade.



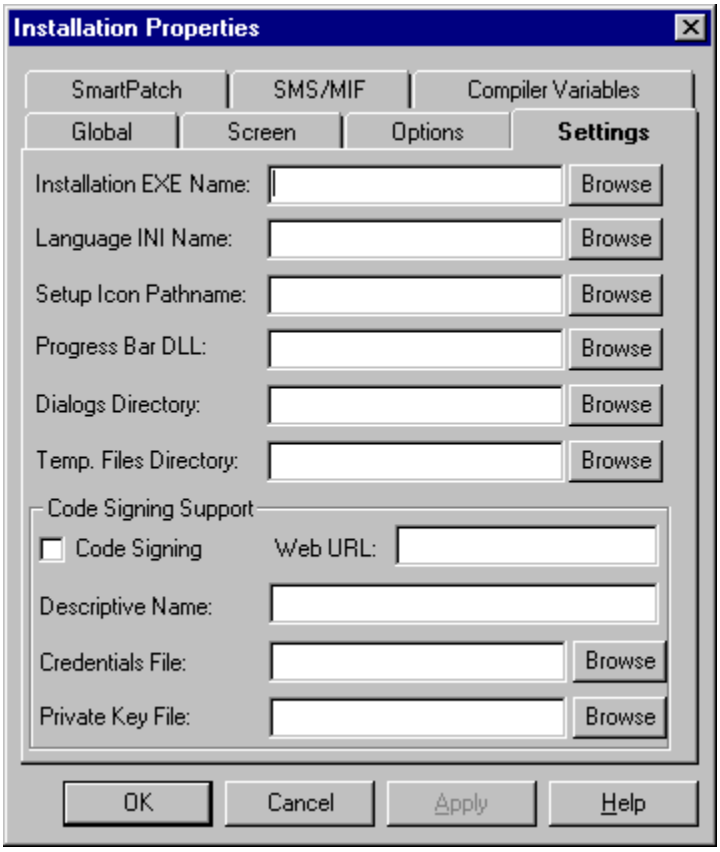
NOTE: The use of this field requires the SmartPatch Add-In for the Wise Installation System. The SmartPatch Add-In is a separately purchased product. Please contact GLBS, Inc. at 800-554-8565 or 313-981-4970 to order the SmartPatch Add-In.

Code-Signing

These fields control whether the installation EXE is signed with a digital certificate. A digital certificate is used to identify the person or company that created the installation. The installations that are created with code signing are Win32 only. (i.e. they will only run under Windows 95, Windows NT, or Win32s) To create an installation EXE that supports code signing you should check the Code Signing box. This will add about 5 KBytes to the size of the installation. In addition, you can fill in the Description Name, Credential File, and Private Key File fields to have your installation EXE automatically signed each time it is compiled. You must have a copy of SignCode.exe installed in your Windows directory. A copy of SignCode.exe is available as part of the ActiveX SDK available at http://www.microsoft.com.

Compiler Settings

These options effect the way the installation is compiled. You can click to on the dialog box below to get more detailed information on the global properties.



Properties Button

You may view the properties of each of the items by selecting this button.

Back

Press this button to move to the previous page.

Next

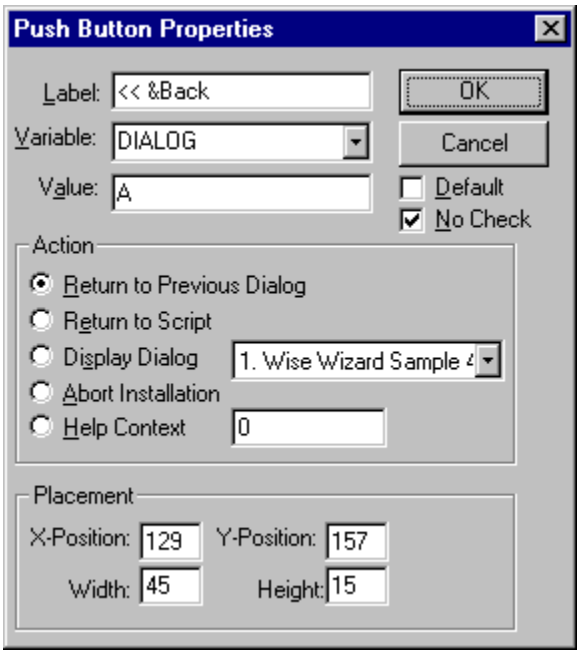
Press this button to move to the next page.

Page Tabs

You can change from one page to another by pressing these tabs.

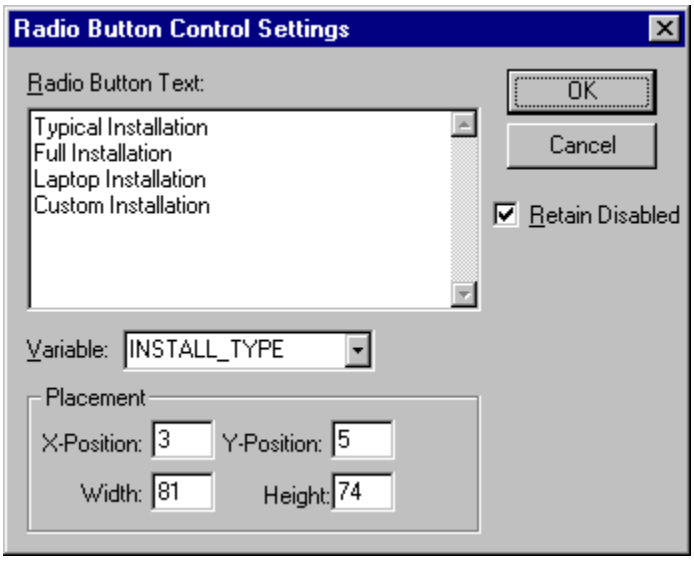
Push Button Control

The push button control will perform an action when it is pressed. It may also set a variable to a specified value.



Radio Button Control

The radio button control gathers either-or information from the user. It allows the user to choose one of a specified list of possible values. The radio button chosen by the user is returned as a single letter (A for the first button, B for the second, etc.) into the specified variable.



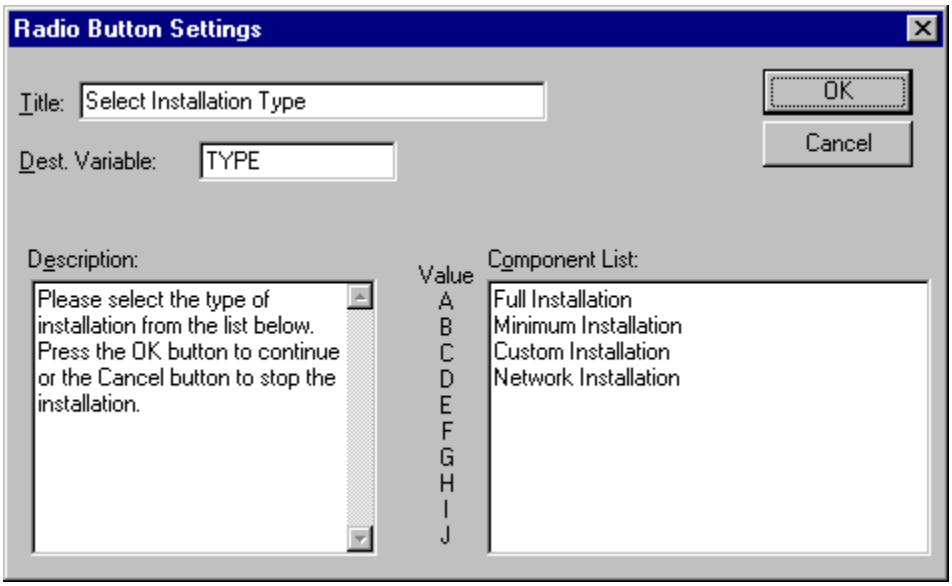
Radio Button Dialog

This script item will display a dialog with a list of radio buttons. The letter that corresponds to the button the user selected will be returned in the destination variable.

To have a radio button other than the first initially selected use the Set Variable script item to set the value of the destination variable to the letter that correspond to the radio button you want selected. For example, if you want the second initially selected you use the script item:

Set Variable COMPONENTS to B

You must use uppercase letters in the set variable script item and the script item must preceed the radio button dialog script item.

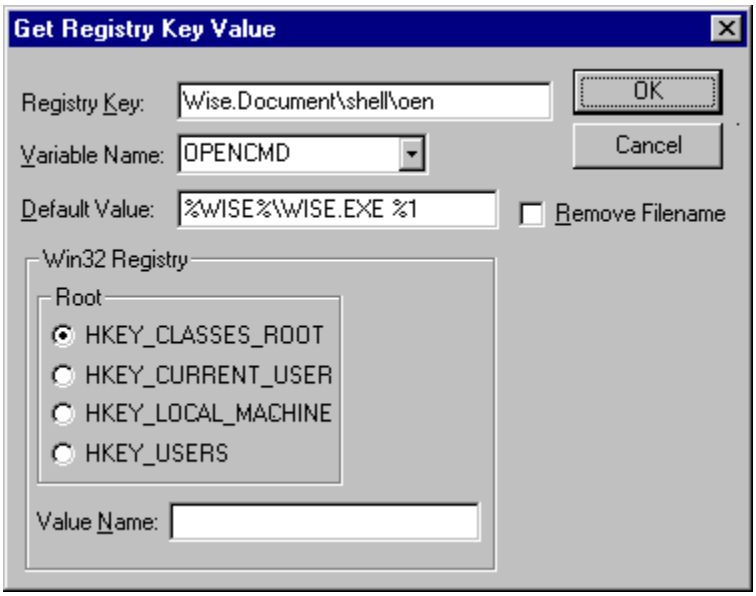


Read Environment Variable

This field holds the name of the environment variable to read. For example, COMSPEC, PATH, and PROMPT are all valid environment variables. This field is case sensitive (for Windows NT).

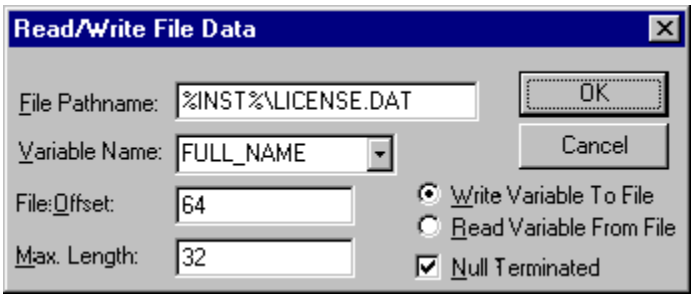
Read Registration Database Key

This script item will read the value of a Registration Database Key into a variable. If the key does not exist, the default value will be returned.

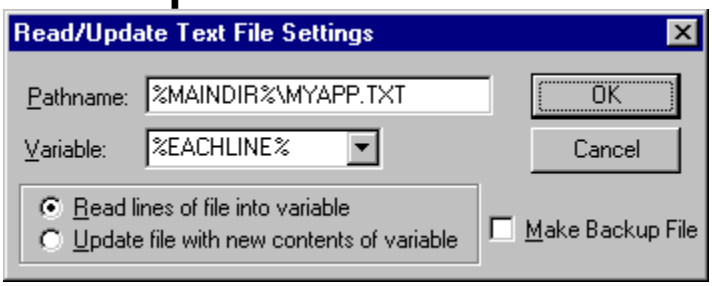


Read/Write File Data

This script item will write the value of a variable into a binary file at the specified location. This can be useful for place a serial number into your EXE files.



Read Update Text File



Registry Data Type

Each key in the Win32 registry can have different possible types. Windows 95 supports String and Double Word types. Windows NT supports all possible data types while Win16 only supports the String data type. If you use the Double Word data type you must enter the number as a decimal integer into the value field.

Registration Database Key Name

This holds the name of the key in the registration database to add. It is added directly below the root specified by the Root radio buttons. An example of a valid key name is:

NewAppDocument\protocol\StdFileEditing

Remove Registry Key Tree

This box can be checked to have the registry key and all of it's sub-keys removed. The Value Name and Key Value are ignored if this box is checked.

NOTE: The registry key that is removed will not be restored by UnWise.exe.

Win32 Registry Root

The Win32 registry consists of a set of keys, arranged hierarchically, all beginning at the root key. The four possible root keys are: HKEY_LOCAL_MACHINE, HKEY_USERS, HKEY_CURRENT_USER, and HKEY_CLASSES_ROOT. You must use the HKEY_CLASSES_ROOT root to add an entry to the Win16 registration database.

Registration Database Key Value

This field holds the value of the key you are adding. You can use variable names in this field. If the data type is Multiple Strings, you can enter a list of values, one per line, into this field. For all other data types you must enter only a single line of information.

Registry Value Name

The Win32 registry allows multple values to be placed into a single registry key. This field holds the name of the value to be placed into the key. The Win16 registration database does not support value names.

Self-Register OCX/DLL

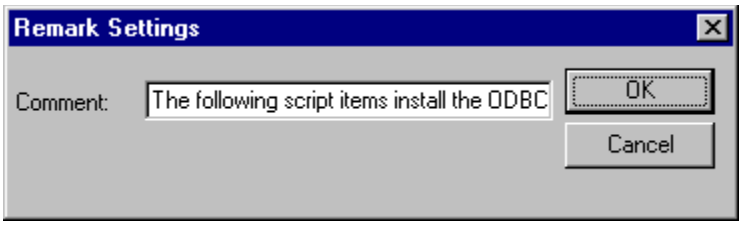
Many OCXs and some DLLs support self-registration. This function can be used to have the OCX or DLL register itself in the Windows registry before it is first used. The OCX/DLL will NOT be registered when the file is installed. The Self-Register OCXs/DLLs script item must be used to actually register the OCX/DLL. This is done to ensure that all support files (i.e. DLL files that are required by the OCX or DLL) have been installed before the OCX/DLL is registered.

Remark Comment

The comment you would like to place into your installation script should be entered into this field.

Remark

The remark script item can be used to place comments into your installation script.



Remove Directory Containing Files

This option will remove any directories that contain matching files. The directory must be empty (i.e. you must have deleted all of the files in ti) for this option to work.

Rename File/Directory

This script item can be used to rename a file or directory on the destination system. You must specify the full pathname starting with a variable reference in the Old Pathname field. The New Filename field could contain the new name of the file or directory. A full pathname must not be included in the New Filename field.

Replace Text in Installation Script

This dialog will replace occurances of the text in the Find What field with the text entered into the Replace With field. Press the Replace button to replace the text for the current script item. You can press the Find Next button to not replace the text and search for the next occurance of the text in the installation script. The search is not case sensitive. The Replace All button will replace all further occurances of the text in Find What with the text in Replace With.

Replace in-use Files

Windows does not allow a DLL/VBX/OCX to be updated if it is currently being used by another program. You can check this box to have your installation gather a list of all files that need to be updated but were in-use. At the end of the installation the user will be prompted to restart Windows if there were in-use files. The files will be updated as Windows restarts. Checking this option will add 15 k bytes to the installation overhead.

Replacement Options

These radio buttons control when an existing file is replaced during the installation. The replacement options are:

- Always replace existing file: will replace any existing file, this option can be used with Version Checking to check the version resource of a file.
- Replace if existing file older or same: will replace an existing file it's date/time modified is the same or earlier than the file to install.
- Replace if existing file older: will replace an existing file it's date/time modified is the earlier than the file to install. If the existing file has the same date/time modified as the file to install, it will not be replaced.
- Preserve existing file: will not install the file if a file already exists with the same pathname.

Require Password

If this box is checked the user will be prompted for a password before this file is installed. The password will be prompted for only once during the installation. You can use this option to select certain files that can be installed without a password (i.e. demo files). The option will not be displayed if your installation does not have a password.

Known Problems/Restrictions

Any complex program such as the Wise Installation System will have "Features" which are more commonly referred to as bugs. Even though this program has been tested thoroughly, please test any installation executables before distributing them. If you find a bug in Wise please send e-mail describing the problem to:

Compuserve: 75111,606
America On-Line: GLBSInc

The following is a list of know problems or restrictions with this version of Wise:

- The demo version creates installation executables that can only be used on the computer that created them.
- Windows 3.1 is supported only (Windows 3.0 would make the installation program larger since it would have to include copies of VER.DLL, DDEML.DLL, LZEXPAND.DLL)
- The graphics import supports only Windows Paintbrush BMP file format. If you have problems importing graphics from other packages, load and re-save the file from Paintbrush.
- The background pattern is slightly distorted when run using an ATI Graphic Ultra video controller using version 2.1 of the drivers. This appears to be a problem in the driver dithering code.

Retain Disabled

You can disable specific buttons in the control by pre-setting the destination variable with lower case letters (i.e. 'a' to disable the first button, 'b' for the second, etc). If you check this box the list of disabled buttons will be retained in the destination variable. This should be used if the custom dialog will be displayed as part of a wizard.

Retain Duplicates in Path

Normally version checking will remove existing copies of the installed DLL that are found in the PATH directory list. This is done to ensure that different versions of the same DLL are not in the PATH. You can suppress this feature by checking this box.

Run Executable

This button will run the installation executable and install the software on your system. It runs the installation in manual mode. You will be prompted for where you want the Windows, System, and Temp directories to be placed.

Run Application

Wise can run your application and determine which DLL/VBX/OCX files are required. In addition, this feature can determine which run-time files are required for VB4 and Delphi applications.

Run Installation

The run installation function will execute your installation exactly as if run by the end-user.

Run in Manual Mode

The installations created by Wise support a manual mode that allows you to select where to place files that will normally be copied to the Windows, System, or installation directories. If you check this box, you will be prompted to select the locations of these directory each time you run your installation from inside of Wise.

Run Minimized

This checkbox will cause the program to be run as an icon when the user double-clicks on it in the Program Manager.

SMS Information

For a description of what information needs to be included with an SMS installation please refer to your SMS documentation.

Default Value

This field holds the value that the variable will be set to if the file is not found in any of the search paths.

Error Checking

The SmartPatch addon can create patches that will upgrade any of a number of previous versions of a file. This field controls when an error message is displayed during the compilation. If no matching previous version of a file is found in the list and errors are disabled a normal file installation will be created. This applies to wildcard Install File(s) script items.

Filename

This field holds the name of the file to search for. All of the directories in the PATH environment variable will be searched for this file.

Maximum Memory

The creation of patches is very memory intensive for large files (i.e. greater than 3 Mbytes). This field controls the amount memory that can be used during the creation of patches. You should set this amount to 2 Mbytes less than the amount of physical memory your computer has.

Maximum Patch Compression

Check this box to create the smallest possible patch. This will cause the compilation process to take much longer.

Test Mode

The creation of patches is a very time consuming process. You can check this box to temporarily turn off the creation of patches and to include the full copy of files. This can be used during testing to speed up the compilation process. This can also be used to create a full installation from an upgrade.

Patch Threshold

If a patch is created from many previous versions of files and there are many differences between the different versions the patch file may get larger than the file itself. This threshold sets the size the patch file can get before a copy of the file itself is used instead of a patch. Even if the new file itself is used, the patch will only be applied if the user has a valid previous version of the software.

Variable Name

This variable will receive the full pathname for the file if it is found. If the file does not exist in any of the directories searched, it will contain the default value.

SQL Installation

These checkboxes control which of the SQLLink drivers are installed into the IDAPI directory. The SQL Installation does not install the SQL Client software.

Sample Wise Installation Script

In the SAMPLE subdirectory below the WISE main directory there is a sample installation script. This is the same installation script that installed Wise onto your system. Included in this directory are the two bitmap graphics that were used during the installation.

The pathnames in this sample installation script assume that you installed the Wise Installation System into the directory C:\WISE. If you get "cannot find file" errors, edit the source pathnames to point to the directories where you installed Wise.

Scale To Screen

Windows supports many different screen resolutions. If you check this checkbox, the graphic image will be scaled to be the same size at any screen resolution.

Scan EXE Header

Checking this box will cause all executable files (EXE, DLL, VBX) to be scanned for the DLL files that they use. This will slow down the create processes somewhat. The list of DLLs will be checked against the files that you have included in your installation script. Any DLLs that are referenced but not included will be displayed after the installation executable is created.

NOTE: The EXE header is scanned for import references. Any DLLs that are loaded dynamically (i.e. with LoadModule) will not be detected.

Scan Entire EXE File

Checking this box will cause all EXE files to be scanned for the VBXs/DLLs files that they use. This will slow down the create process somewhat. The list of VBXs/DLLs that are referenced will be checked against the files that you have included in your installation script. Any VBXs that are referenced but not included will be displayed after the installation executable is created.

NOTE: Since VBX files can be loaded dynamically, there are cases where a VBX reference may not be detected. There is also the possibility that extra references will be reported.

Add All Files in Directory

This button will add all of the selected files in the file names listbox to your installation script. No files in sub-directories will be added.

Add File

Press this button to add the currently selected file in the File Name list to your installation script. The file will be added to the list to the left.

Add Makefile

Press this button to add all of the files in a Visual Basic makefile into your installation script. Only Visual Basic makefiles are currently supported.

Add Directory Sub-Tree Files

Pressing this button will add all of the selected files for the current directory and all sub-directories below it.

Component

This field selects which component's files are displayed in the file list. You should change this value to add files to all of the components in your installation.

Delete

Press this button to delete the currently selected file in the "Files Added to Installation Script" list.

Destination Directory

This directory variable will be used to select the destination of the installed files.

Directory

This is allows you to select the directory where the files to install are located.

Drives

This list select which of your disk drives to locate the files to place into the installation script.

Files Added to Installation Script

This list contains all of the files that will be added to the installation script. The buttons to the right control the adding and deleting of files from this list.

FileName

This list selects the files to add to your installation script.

Run EXE

This button will execute the selected program and monitor the DLLs and VBXs that the program loads, adding them to the files to install. When you press this button you should test all of the features of your program to ensure that all DLL/VBXs are detected. When you are done, exit your program. This feature currently works with Win16 programs only.

File Type

You can select which type of files to display in the files list. The Program Files list can be customized in the View->Preferences dialog.

This list box will hold your installation script items. This script determines how your software will be installed when the Wise executable is run. If you double-click on a item in this box, you can edit the information about that item. The buttons to the right should be used to add, edit, and delete items from this list.

Installation Expert

This dialog will allow you to test, run, and create floppy disks that hold your installation. All of the files in your installation will be compressed and split across the floppies.

Script Actions

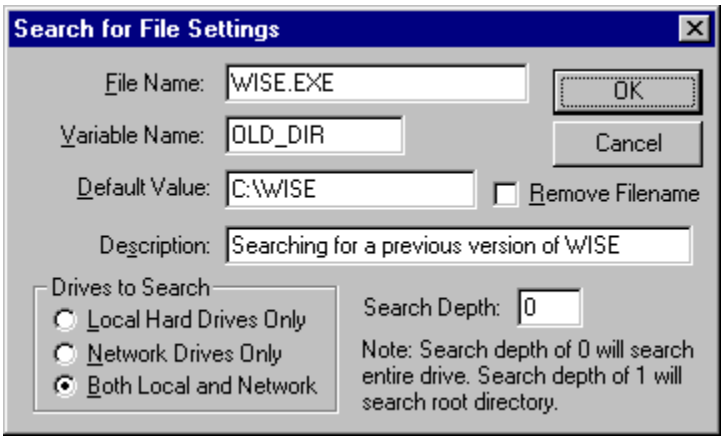
This list contains all of the possible actions that can be performed by Wise during the installation. You can add a new action to your installation script by double clicking on an item within this list or by dragging an action to the appropriate location in the installation script.

Script Languages

The script languages dialog is used to determine the languages that your installation script can use. It also defines which of the languages should be exported into the installation executable (and in which order). Select all of the languages to include in your installation from the list. The number to the right indicates the order the languages will be displayed to the user at the beginning of the installation. The Copy Default checkbox will copy all of the text from the default language to all of the other languages (to assist the translation).

Search for File

This script item will scan the users disk drives for the file name given. No wildcard character may be used. If the file is not found, the default value is returned.



Select Files to Install

This dialog is used to select which files will be placed into the installation script.

Self Register OCXs and DLLs

The Self-Register OCX/DLL checkbox in the Install File(s) and Copy Local File(s) script item does not register the files when they are installed. This is done to ensure that all of the support files for the OCX/DLL are installed first before the file is registered. This script item should be used after all OCXs and their support files have been installed. When this script item is executed all of the files selected for Self-Registration will be actually registered.

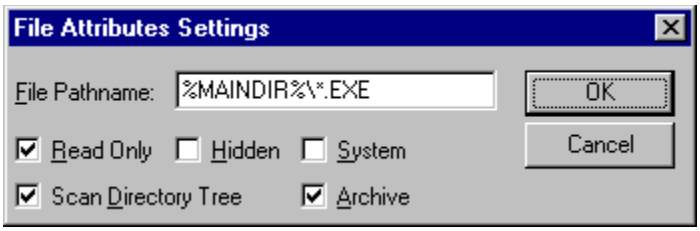
The optional Description field holds the text that will be displayed on the screen while the OCXs are being registered. If this script item is omitted from the installation the OCXs/DLLs will be registered at the end of the installation.

Separate Address Space

You should check this box if you want the program to run in a separate address space under Windows NT 3.5 or NT 3.51. This box can only be checked if your are installing to a Windows NT 3.5 or NT 3.51 system. NOTE: This feature is not supported under Windows NT 4.0.

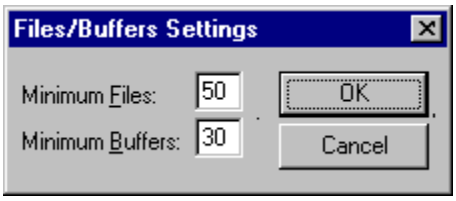
Set File Attributes

This script item can be used to modify the file attributes for one or many files.



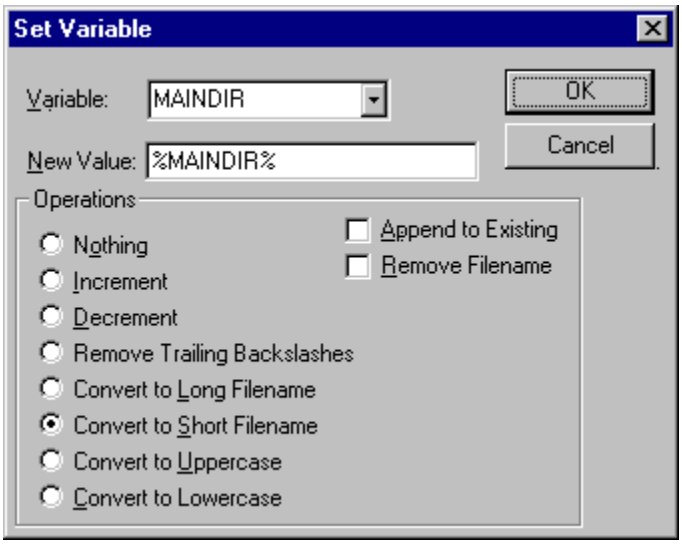
Set Minimum Files/Buffers

This script item edits the CONFIG.SYS file and changes the "Files=" or "Buffers=" settings if they are below the values indicated.



Set Variable Value

This script item can be used to set the value of a variable. A number of operations can be performed on the variable as it is being set.



Setup Icon

You may specify the icon that will be incorporated into the installation executable.

Win95 Shared DLL Counter

Windows 95 supports the shared DLL/VBXs by using a counter stored in the registry. Each time an application that uses the DLL attempts to install the file, the counter is incremented. When an application is un-installed this counter is decremented. If the counter returns to zero, the user is prompted to delete the file from their system. You should check this box on any DLL/VBX files that will be shared by more than one application.

Shareware

Shareware software allows you to try a software package for a limited time (in this case 30 days) before you decide to purchase it. If you find that you want to continue using this software you can purchase a registered copy.

Shell Extensions

You can add support for your program's documents within the File Manager and Windows 95. Select your programs main EXE from the list of executables in your installation. You should then enter the three letter file extension for your applications documents. If you program supports printing via command line options you should enter the command line options as well.

The Windows 95 operating system supports private application paths (much like the DOS PATH environment variable) for every Win32 application. You can enter a list of directories (starting with a variable reference) separated by semi-colons.

Windows 95 also supports associating an icon with a document extension. You can fill in the full pathname of a file that contains your document icon, a comma, and the icon number within that file.

Application Path

Windows 95 supports a separate PATH list for every application. You can fill this field with a list of directories that will be searched for DLLs/VBXs used by your program.

Document Icon

Windows 95 can display an icon for your application documents. This field holds the pathname of the file that contains the document icons, a comma, and the icon number within that file.

Main Program Executable

This list displays all of the programs in your installation. Select the executable from this list you would like to add a file association for.

Document Extension

This field holds the three letter file extension of your programs document files. Files with this document will be associated with your main executable program. If a user double-clicks on a file with this extension in the File Manager, the main executable you selected will be executed.

Application Identifier

The application identifier is used internally by the File Manager to store the file association. If you leave this field blank, a value will be generated from your application name.

Print Command Line Options

If your program supports printing from the command line add the command line parameter here. This will allow users to print your documents from the File Manager.

Show TitleBar

You can check this box to display the title bar at the top of the installation screen. If this box is not checked, no title will be displayed.

Smart Create

The smart create option will check the date time modified and size of all of the files in your installation to see if they have changed since the last time the installation executable was created. If any of the files has changed, the installation executable will be re-created. This option will add a small amount of time to the Run and Make Disk commands.

Control Install Speed

This checkbox controls whether the installation process is slowed on fast computers that run the installation executable from a hard disk. This ensures that the installer will have enough time to read the graphics that you place on the screen.

Split EXE

This selects how big a Wise executable can become before it is split into another file. If this is left empty, the executable can be as large as necessary. You would normally set this to be the size of the floppy disks that you ship your software on. The split files will have the file extension W02, W03, etc.

Start If Block

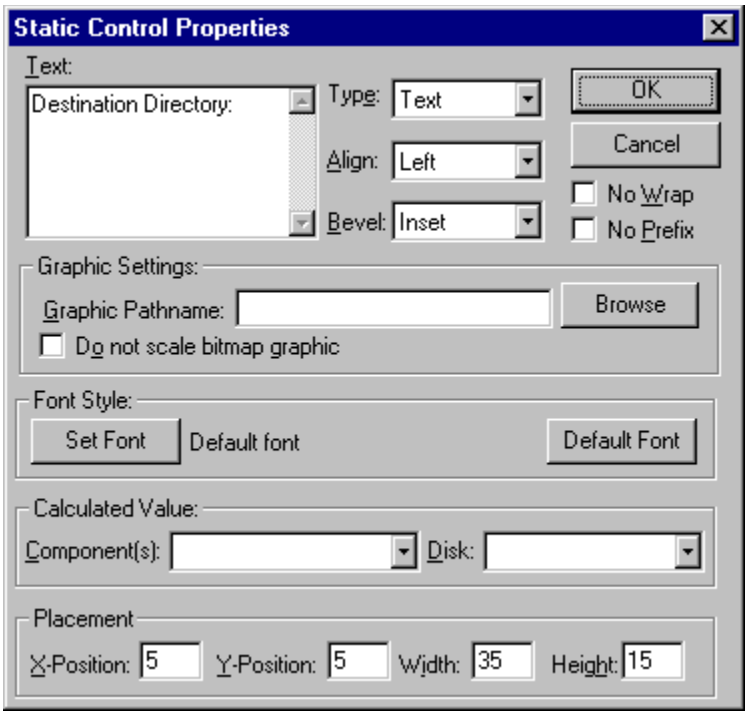
Select this option if you want to script items in the enclosed block to be executed once if the If statement is true.

Start While Loop

Select this option to cause the enclosed script items to be executed repeatedly until the condition is false.

Static Control

The static control is used to display information, either textual or graphical in the dialog box.



Show Status Bar Tips

Checking this box will enable the status bar tips. As you move the mouse over the tool bar buttons, a description of what each button does will appear in the status bar at the bottom of the window.

Technical Support

Free technical support for Wise is available via the following:

- Compuserve forum: GO WISEINSTALL
- Compuserve e-mail: 74777,3314
- American OnLine e-mail: GLBSInc
- Web Page: http://www.glbs.com
- Internet e-mail: 74777.3314@compuserve.com
- Technical Support Fax: 313-981-9746

In addition, 30 days of free technical support via telephone is provided after the purchase of the Wise Installation System. You may purchase a maintance contract to receive automatic upgrades to Wise and technical support via telephone. You may call 800-554-8565 or 313-981-4970 to order the maintenance contract.

- Technical Support Phone: 313-981-4970

Temp Directory

You may specify which directory you would like to use to store temporary files created by the editor or during the compilation process.

Test Executable

Press this button to test your installation script. In this the test mode, no changes will be made to your computer system.

Test Installation

The test function will simulate your installation by displaying all graphics and dialogs. During the test no modifications will be made to your system (ie file installations, registry changes, INI modifications, etc.)

Tile Graphic as Background

You should check this box to have the graphic used as a tiled background during the installation. The graphic will be repeated for the entire size of the background window.

Timed Display

Check this box to display a set of graphics evenly throughout the installation. Each graphic that has this box checked will be displayed one after each other during the installation.

Window Title

This holds the title of the installation script. This will appear at the top of the window during the installation.

Example: A common title would be "MyProgram Installation"

About Wise

This button will bring up a dialog box describing the version of Wise you are running.

Comment Script Items

This button will "comment out" the selected script items. Script items that are commented out are ignored by the Create, Test and Run commands. Pressing this button again will uncomment the selected items if they are already commented out.

Copy Script Item

This button will copy the currently selected script item into the clipboard.

Cut Script Item

This button will delete the currently selected script item and place it into the clipboard.

Move Script Item Down

Pressing this button will move the currently select script item down in the installation script.

New File

Pressing this button will clear your workspace so you can start a new installation script. If you have made changes to your old installation script, you will be prompted to save your changes.

Open Existing File

Press this button to open an installation script that you have already created. You will be prompted for the pathname of the installation script.

Paste Script Item

Pressing this button will insert the script item that is in the clipboard into the script before the currently selected script item. If no script items are selected, it will be appended to the end of the script.

Print Installation Script

This button, when pressed, will print your installation script to a printer. You must have the Professional Version of Wise to print installation scripts.

Save Installation Script

Press this button to save your changes to the installation script. If you have not saved it previously, you will be prompted for the name of this installation script.

Show Tool Bar Tips

Checking this box will enable the tool bar tips. If you stop the mouse over a button in the toolbar, a small window will appear describing what that button does.

Move Script Item Up

Pressing this button will move the currently selected script item up by one.

Add Item

You should select this button to add a new Icon to the Windows Program Manager. You will have to fill in the Group Name, Item Name, and Command Line to the right.

Start Block If Return Value True

The DLL Function can return TRUE (non-zero) or FALSE (zero). If it returns FALSE, all script items will be ignored until the matching End Message Block is found.

Distribute

Based on your Media selection Wise will compile your installation into a single exe or multiple exe files (one for each disk).

Transparent Graphic

This button allows you to place a graphic into the installation where parts of the background show thru. Everywhere that you use the solid blue color (Red=0, Green=0, Blue=255) will become transparent to the background and to each other.

Tree Description

This field holds the description that will be displayed while the files are being copied.

Tree Destination

This is the name of the directory where the files will be placed. This should start with a variable name.

Tree Source

This is the pathname of the directory tree that you want to place into your installation executable. You can use the browse button to select the tree.

Installation Expert - VB4 Options

This dialog should be used to install the runtime files required by VB4 applications.

Visual Basic 4.0 Directory

This must point to the directory where VB4 was installed. Wise will pull the standard setup files from that tree.

Visual Basic 4.0 Run Time

Some of the more common components are grouped together to allow a single checkbox to include these files. You may use the Files dialog to include single OCX/DLL files that are not grouped together.

Visual Basic Run Type

You can choose the type of Visual Basic 4.0 installation based on your executable. Wise will include the run time files for the selected operating system.

Variables

The use of variables is fundamental to the creation of an installation script. Variables hold the pathnames of where your files will be copied. They can also be used to hold the names of groups you will add to the Program Manager. Variables can be created by a Prompt script item or in a DLL function call. When referencing a variable, you must enclose it in percent (%) signs. There are four pre-defined variables:

1 WIN: This variable holds the pathname to the Window directory (usually C:\WINDOWS).
2 SYS: The Windows System directory pathname is contained in this variable (usually C:\WINDOWS\SYSTEM).
3 TEMP: This varible holds the directory where temporary files may be placed. This variable is useful for placing DLLs before you call their functions.
4 INST: The pathname of the location of the installation executable is held in this variable. This can be useful if you want to display a readme.txt that is located on the same disk as the installation executable.
5 RESTART: When this variable is set to 'W' Windows will be restarted at the end of the installation. If the variable is set to 'S' the system will be rebooted. If this variable is set to E followed by a command line, the command will be executed during a Windows restart. A number of advanced script items (editing autoexec/config) set this variable.
6 BACKUPDIR: If this variable is set, all files replaced by the installation will be placed into this directory.
7 PASSWORD: If your installation is password protected this variable will be used as the password. If this variable is not set, the user will be prompted to enter the password.
8 CMDLINE: This variable holds the command line options that were passed to the installation executable.
9 LANG: This variable holds the language the user selected in a multi-language installation.
10 HELPFILE: You can set this variable to the pathname of a Windows Help file that will be used in any custom dialogs.

In addition to the four pre-defined variables, there is a special variable that you may choose to place in a Prompt script item. If you create the variable BACKUPDIR, all files replaced by the installation will be placed into that directory. If you need to place a percent sign into your script, you must place two percent signs next to each other as in:

```
There is a 100%% chance this will work!
```

When the installation executable is run, this will display as:

```
There is a 100% chance this will work!
```

Version Checking

Wise uses the VerFindFile and VerInstallFile routines to perform version checking on installed files. If you install a shared DLL/VBX to the Windows or System directory these routines will normally move a file from the Windows directory into the System directory on a non-network Windows installation. In a networked environment the shared DLL/VBX will be placed into the Windows directory.

If an existing file has the exact same version number as the file to be installed it is still replaced. This is done to ensure a proper installation. If the existing file has become corrupted and it is not re-installed your software would not function properly.

Version Check

You should use this checkbox to install DLLs or VBXs that contain a version resource. A good example of files to use version checking on are CTL3D.DLL and THREED.VBX. You may use version checking for files that are not to be installed into the Windows System directory (i.e. application private files). This check reads the version resource from each of the files and installs the file if it is newer than the existing file.

WISE

W.I.S.E. stands for WIndows Self-Installing Exectable. This is a program that contains compressed copies of software and the installation script.

Whats New?

The following features have been added to Wise version 5.0:

Script File (.wse)

The script file is now ascii!! You can now have two copies of WISE running at the same time and cut/paste script items between them. You can also cut/paste to the clipboard, an e-mail message, or just about anywhere. We do not have documentation completely describing every command and the syntax yet, however, we do hope to eventually have something published.

We feel that the syntax we have chosen is self-documenting. Currently, just about everything is built on the premise of being an item. There are two types of items (i) Global, which describes the installation itself; (ii) script items, one for each script item in the installation. The syntax for item statement is "item: xxx". The block will always end with an "end" statement.

Each item statement may have properties and/or items of it's own. For example: a set variable script item will have only properties within the item block. A custom dialog script item will have properties and items such as pushbutton, listbox, static, etc.

The values of each item are defined to make it easy to create a script in a programming language or a text editor. The only values that may be hard to decipher are the flags. If you want to duplicate the flag values you may need to create a script item with your options to see the flags that you want to set.

Once you have created your text script outside of the editor you can run WISE from the command line with the following syntax: "WISE (or WISE32) /C myapp.wse". This will create the executable as defined in the script.

Installation Properties:

Global:
Progress Bar Based on:
Total Installation, Total Script, Files to be installed

Settings:
Installation EXE Name:
By default the Installation Executable(s) will be created in the same directory as your .WSE file. If you would like to change where the installation executable is created fill in the directory and filename.

Language INI Name:
The Installation Expert can create multi-lingual scripts. This INI file needs to contain the translations of the messages used in the dialogs by the Expert.
(i.e. Welcome, Browse for Directory, Installation Complete, Etc.)

Setup Icon Pathname: This ICO file will be used as the icon included with the installation executable.

Progress Bar DLL:
You may create a DLL that will be used in place of the standard progress bar. You will find a complete sample DLL that may replace the progress bar in the directory DLL\PROGRESS below your main WISE Directory.

Dialogs Directory:
The Installation Expert uses 9 standard dialog boxes to create the Wizard installation. By default these can be found in the DIALOGS\TEMPLATES directory under your main WISE Directory. If you would like to customize these, we suggest you make a copy of them in a different directory and fill these fields in with the pathname.

Temp Files Directory:
By default WISE will create temporary files in the TEMP directory defined by Windows. You may specify a different directory in this field, possibly one that might have more disk space.

Code Signing Support:
These fields control whether the installation EXE is signed with a digital certificate. A digital certificate is used to identify the person or company that created the installation. The installations that are created with code signing are Win32 only. (i.e. they will only run under Windows 95, Windows NT, or Win32s) To create an installation EXE that supports code signing you should check the Code Signing box. This will add about 5 KBytes to the size of the installation. In addition, you can fill in the Description Name, Credential File, and Private Key File fields to have your installation EXE automatically signed each time it is compiled. You must have a copy of SignCode.exe installed in your Windows directory. A copy of SignCode.exe is available as part of the ActiveX SDK available at http://www.microsoft.com. To receive credentials for code-signing contact VeriSign (www.verisign.com).

SMS/MIF:

MIF Filename:
You can specify the name of the MIF file created for use with an SMS type distribution system. This file will be created in the Windows directory.

Manufacturer, Product, Version, Serial Number:
These fields will be inserted into to the MIF file that is created at the end of the installation. You should consult your SMS manual for information about the values of these fields.

Compiler Variables:
Compiler variables are used exactly like an IFDEF in a C type programming language. These variables will look exactly like standard WISE variables except they need to be defined in this dialog. These can be used to create conditional compiles of an installation for deployment for testing. WISE uses a standard of an underscore ("_") as the first and last character of a compiler variable to make it simpler to determine which type of variable it is. Once a variable has been

defined you may set a default value to be used if it is not prompted for. You may also specify when you want to be prompted for the value. For example during testing (compiling from the IDE) you may not want to be prompted every time for each compiler variable. However, when you are creating a final installation (either from the IDE or from the Command Line) you may want to be prompted for the values. You can run WISE from the command line with the following syntax: "WISE (or WISE32) /C myapp.wse". This will create the executable as defined in the script.

Dialog Templates:

Editing Dialog Templates:
The Installation Expert has 9 dialog templates as well as 11 other script items have dialog templates that you may edit. We suggest that you make a copy of the dialog templates before modifying them. You can change the default directory for the dialogs in the Installation Properties/Settings property page. You can modify any of the standard templates such as Browse for Directory, Prompt for Text, Select Components, Get Name Serial Number, Etc.

Events:

The Mainline "event" is actually the full installation script. WISE currently supports two other events:

Exit:
The Exit "event" is executed when the script runs to completion or executes an "Exit Installation" script item. You may add code that may be executed at this time such as Prompting to Execute the program just installed or any other common task that you do not want to duplicate code for within the MAINLINE.

Cancel:
The "Cancel" event is executed whenever the user chooses to cancel the installation. This can be on any dialog or during an actual file installation. When a user chooses to cancel an installation, WISE will bring up a dialog that asks if they are sure. If the user confirms to cancel the installation then the script items in this event will be executed. You may check to see if a partial installation was done and notify the user that he may need to copy some of the BACKUP files into production. If you did not want the user to intervene the script could look for files in the backup directory and copy them into production.

Script Items:

Compiler Variable IF:
This script item works exactly the same as a standard IF Block, however, it will only accept compiler variables. You need to define the compiler variables in the Installation Properties/Compiler Variables property page. An example of using this statement would be if you created a single installation that installed both the 16bit version of your program and the 32bit version. You could use a compiler variable to determine what type of installation you would be creating and choose only to include the required files for that platform.

Compiler Variable Else, Compiler End:
These are used in the exact same manner as "Else Statement" and "End Block". They may only be used in conjunction with a "Compiler Variable If" script item.

Copy Local File:
This statement now supports wildcards in the Source field. If you do use a wildcard in the Source then your Local Path must be blank. WISE will determine what files will be copied at the time the installation is run, not when the installation is compiled. Remember, since we do not know how many files, or what size they are, the progress bar will not move while this statement is being executed. The best way to have the progress bar look normal is to have it be Script Item Based.

Create Service:
This script item gives you the ability to add a Service (background process) to a system. You should consult Microsoft's MSDN for more information about creating a service.

Edit INI File:
This script item has been changed so that you only need one line for each INI file being edited. You only specify the file and the "INI File Contents" will look exactly like your INI file. You can cut and Paste from any standard INI file. If you include a section (i.e. [Section]) with no entries after it, then the Section will be deleted. If you include an entry in a section with no value (i.e. "LastFile=") then that line will be deleted. You may do multiple events in a single "Edit INI File". For example: you may delete a section(s), add a section, add an entry(s), remove an entry(s) all within a single script item.

Edit Registry:
The new addition to this script item is that you may edit a Binary Value. When you are adding a new Binary Value you need to enter the Key Value as a Hex String. For example: "1E 2F 3C BB 23 65", any digit that is not a hex value will be converted to a "0" (zero).

Execute Program:
We have added the "Minimized" and "Normal" option for the Window Size when executing an external program.

Find Files In Path:
We have added an edit box that will allow you to specify what additional directories to search in, as well as the PATH defined by Windows.

Get System Information:
We have added three new system information:

Volume Serial Number/Volume Label:
This will return the serial number/volume label of the media defined in PATHNAME. Make sure that you use a variable reference and not a hardcoded pathname.

Network Username:
This information is obtained by calling the WNetGetUser API Call.

Install DirectX:
We have added the option to install Direct3D. You should consult the Microsoft MSDN for more information about installing DirectX.

Read/Update Text File Settings:
This script item creates a loop block, executing the loop once for each line of text in the file. You need to fill in the pathname with a file that exists on the user's system. Make sure that you use a variable reference and not a hardcoded pathname. The Variable that you use will hold a single line of text from the file each time the loop is executed. If you change the value of the variable inside of the loop the line will be updated in the file if you select to update the text file.

Search for File:
This script item now allows you to search for Files on Local Drives Only, Network Drives Only, or Everywhere. We suggest that if you are searching either on Network drives or everywhere that you limit the search depth. The search depth specifies how many levels (or directories) WISE will go down to on any single drive. A search depth of 0 will search to the bottom of a directory. We suggest that when you search a network drive you limit the level to 2 or 3 so the user is not waiting for a very long time.

Self-Register OCXs/DLLs
You can fill in the Description/Pathname with a pathname and check the radio button "Queue existing file for self-registration" to register a file that you did not install with the installation. This works very well for a Workstation installation where all the files are already installed on the network server. You could then use the same script item with a description to register all the queued OCXs/DLLs.

Wizard Block:
This script item controls the entire Wizard section of the installation. To better understand this, we suggest you use the Installation Expert to create a script so that you can see how this script item works. You need to end the Wizard Block with an "End Statement" script item.

Dialog Boxes:
This is a list of all the dialog boxes that are within the Wizard. We suggest that each Custom Dialog script item only contain one Dialog. Each dialog may have a "Skip Check" placed on it with a simple if statement. By default, the dialog will be displayed. If the statement returns true, then the dialog will be skipped. An example of when you would like to skip a dialog would be when prompting for a backup directory. One dialog would ask if you want to backup the files being overwritten and the next dialog would ask where you wanted the backup files placed. If you stored the question about backing up the files into the variable DOBACKUP, then the next dialog could reference that variable to see if it needed to be skipped. The Statement would be:

"If Variable" DOBACKUP "Equals" "The Value" "NO"

This would skip the dialog that would prompt the user for the directory to store the backup files.

Destination Variable:
This variable holds the value of what direction the Wizard is heading. All of the next buttons will set this variable equal to "N" (for next), and all of the back buttons will set this variable equal to "B" (for back).

Display Variable:
This variable will hold the name of the dialog that will be executed this time in the loop.

Wizard Bitmap Pathname:
You can display a single bitmap in all the wizard dialogs. This should be a standard size graphic that will fit within the wizard dialogs. A sample graphic (WIZARD.BMP) can be found in the DIALOGS\TEMPLATES directory below your main WISE Directory.

Wizard Bitmap X-Pos/Y-Pos:
This is a relative placement of the graphic within the dialog.

Wizard Bitmap Do not resize bitmap:
Some graphics do not look well when they are resized, you may want the graphic to remain the exact size no matter what resolution the end-user has.

Wizard Bitmap 3D Border:
This will add a 3D border to your graphic.

Wizard Bitmap Filler Color:
This color will be used around the edge of the graphic if the graphic is not sized at different resolutions.

Installation Expert:

Installation Interface:

Media:
You can select either a single file installation or a floppy based installation. The floppy based information has options for which type of floppy or how big you want the installation executable to be.

Application:
This dialog defines the software title and the default directory. The software title is used for the for wizard dialog titles, as a background graphic, as the primary icon name, and as the title in Add/Remove Software dialog under Windows 95. The default directory will appear exactly how you type it under Windows 3.1 and Windows NT 3.51, however it will be changed to be under "Program Files" for Windows NT 4.0 and Windows 95.

Dialogs:
The dialogs that are selected will be used in your installation as wizard dialogs. Remember that some dialogs require additional information such as the ReadMe.

Graphics:
The graphics that you add here by default will be timed and the first one will be displayed right before the first file is installed. You may change the properties on a graphic by clicking on the details button after you have selected a graphic. The order that they appear on the screen is the order they will appear in the installation.

Application Files:

Components:
You may create components of your installation that the user may choose to install or not install. Each component that you create in this dialog will be listed in the same fashion as a directory in the Files dialog.

Files:
The top half of this dialog is your machine (My computer) and the bottom half is the customer's machine (Destination computer).

Source Computer:
This is a modified version of the Windows 95 explorer and a file open dialog. On the left hand side is a list of all drives/folder on your machine. You may add the contents of a folder and all sub-folders (to the highlighted folder below) by clicking on the "Add Folder" button. You may move around the directory tree by clicking on an entry. The right hand side contains a list of files in that folder, you may either double click on a file or use the "Add File" button to include a file.

Destination Computer:
The left hand side has a representation of how/where installation files will be installed. If you do not have any components installed, then the structure will have two folders off of the destination computer. The Application folder may contain files or more sub-folders. The Windows folder will contain files to be installed into the Windows directory and sub-folders for Fonts, System, and System32. You may only create folders under the Applications folder, not under the Windows folder. If you have components in your installation then the level just below will list your components and then have the other directories below that level. You may change the installation properties of a file by clicking on the details button.

Runtime Support:

Options:
You can click on as many different runtime support components as needed by your application.

Currently we have support for Uninstall, Share, OLE2, MS-Access 16bit ODBC and MS-Access 32bit ODBC. This list will grow as we develop installation scripts for other various products.

VB4:
WISE supports both VB4-16 and VB4-32 installations. Some of the more common components are listed at the bottom of the dialog. You need to specify which directory your VB system is installed in so that WISE may know where to retrieve the runtime files.

BDE (Borland Database Engine):
WISE supports both BDE-16 and BDE-32 installations. You need to specify what platform, and for the 32bit, you need to specify whether you will be creating a full or partial installation. You may optionally specify an Alias for your database.

User Configuration:

Icons:
You may select any number of icons to be added during the installation. The default group name will be used as the Program Manager Group under Windows 3.1 and Windows NT 3.51, and as a sub-menu under Windows NT 4.0 and Windows 95. The icon name you use will be adapted for use under all operating systems.

Associations:
You may specify a file extension association for your executable. For example: Microsoft would associate the .DOC extension with the WINWORD.EXE file.

INI Files:
You need to use the Add Button to create a new INI file within your installation or to edit an existing file. Once you specify the INI file, click on the details button to insert the text. You can Cut and Paste from any standard INI file. If you include a section (i.e. [Section]) with no entries after it, the Section will be deleted. If you include an entry in a section with no value (i.e. "LastFile=") then that line will be deleted. You may do multiple events in a single "Edit INI File". For example: you may delete a section(s), add a section, add an entry(s), remove an entry(s) all within one entry.

Registry:
This dialog functions very much like Regedit. The registry that you see is a representation of the changes made to the destination computer. You may create any number of sub-keys and values within the sub-keys. The default type of value created is String, you may change this by clicking on the details button.

System Configuration:

Devices:
You may add entries to the [386Enh] section of the system.ini file. You must first install the file before you can reference it in this dialog. By default the file should have a 386 extension.

Services:
This dialog allows you the ability to add a Service (background process) to a system. You should consult Microsoft's MSDN for more information about creating a service.

Autoexec.bat:
You can add a path to the autoexec by filling in the Edit Box. Make sure that the path that you are adding starts with a variable and is not a hardcoded path. You can also add commands to the autoexec.bat file by clicking the Add Button.

Config.sys:
You can also add commands to the config.sys file by clicking the Add Button.

UNINSTALL SUPPORT:

Executing An External Program:
You can execute an external program (or programs) from within the uninstall by adding a line to the install.log. The syntax is:

Execute Program: C:\Program Files\My Application\Uninstall\RemoveMe.exe

When adding this line to the install.log remember to use a variable for the path and not a hardcoded pathname. We recommend that you have little to no user interface within this program so the uninstall will seem to be a single program. During "Automatic" uninstall all programs will be executed. If the user chooses to do a "Custom" uninstall they will be prompted as to which programs to execute.

Remove a Sub-Tree of Files:
With some applications creating files after install we have added the following command syntax to the unwise program:

File Tree: C:\Program Files\My Application*.*

This line in an install.log will delete all files under the My Application sub-directory (including the My Application directory). This command will never allow you to delete any files under the Windows/System directory or from the root of a drive.

Deleting a Registry Tree:
To delete an entire registry tree of keys you will need to add the following lines to the install.log:

RegDB TREE: SOFTWARE/GLBS

RegDB Root: 2

Where the RegDB Root value can be one of the following:
0 - HKEY_CLASSES_ROOT

1 - HKEY_CURRENT_USER
2 - HKEY_LOCAL_MACHINE
3 - HKEY_USERS

Adding customization with UNWISE.INI

The following sections have been created in the UNWISE.INI file:

[Language]
msg0=...
msg1=...
msg2=...

The following files may be found in the WISE directory:

UNWISE.FRE - French Translation
UNWISE.SPA - Spanish Translation
UNWISE.ITA - Italian Translation
UNWISE.GER - German Translation

If you merge the appropriate language file into a UNWISE.INI and install the UNWISE.INI into the same directory as the UNWISE.EXE the UNINSTALL will be translated.

You can also set the background color displayed during the uninstall. You can look at the properties (background tab) for an example of how the colors will display.

[Settings]
Start Red=0
Start Green=0
Start Blue=255
End Red=0
End Green=0
End Blue=0
(this will display the standard blue faded screen).

If the No Background is set to 1, then the uninstall will not display any colored background.

[Settings]
No Background=

The default font for the uninstall is Times New Roman. You may choose another font by:

[Settings]
Title Font=Times New Roman

Window Preview

A small preview of the background window is displayed here.

Background Window Size

These three radio buttons determine the size of the background window. You can make the background window the size of the entire screen, most of the screen, or to not have a background window.

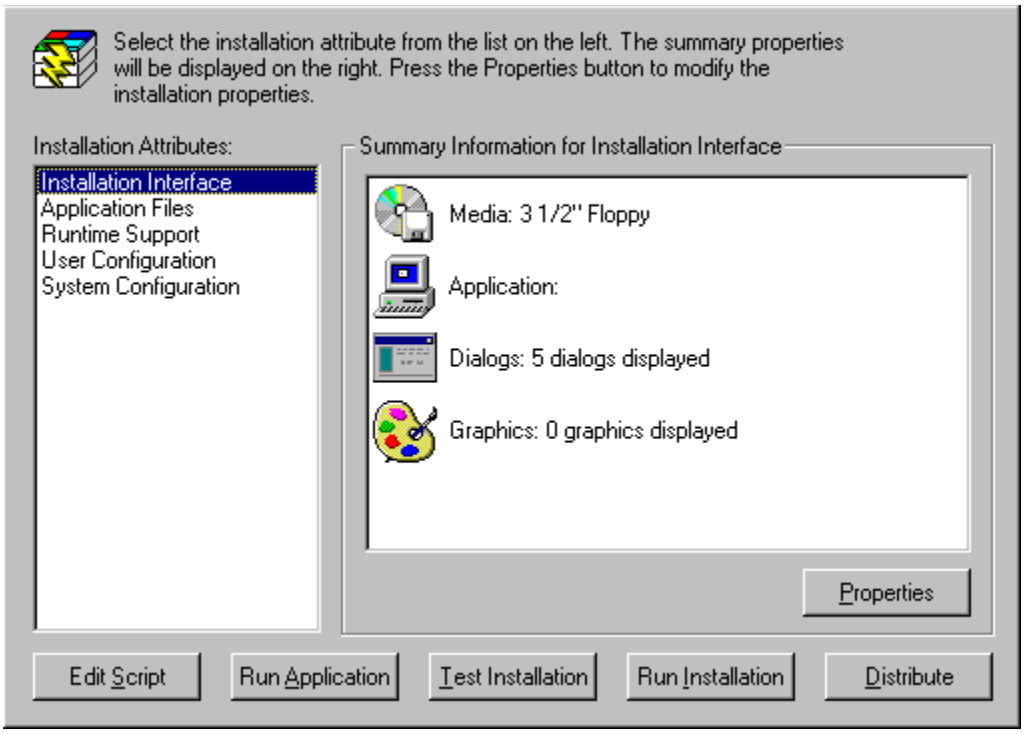
Wise Features

The Wise Installation System gives you great flexibilty in creating professional quality installation programs. It supports the following features in both the 16 and 32 bit editions:

- Single File, self-installing, Windows executable
- Bitmap graphics, both transparent and opaque displayed during the installation
- Multiple prompts for file locations
- Subsets of the files can be loaded
- Very low installation program overhead
- Very high compression ratios
- Executing external programs
- Version Checking
- Disk space checking
- Installation Log
- Making backup copies of replaced files
- Adding keys to the registration database
- Checking DLL/VBX dependencies
- Multiple Disk Installations
- Calls to DLL based functions
- Customizing the installation with Visual Basic programs
- A free distributable uninstaller
- Support for multiple languages in a single installation script
- Editing the AUTOEXEC.BAT/CONFIG.SYS/SYSTEM.INI
- Component based installations
- Searching for a file on the destination system
- Radio button dialogs
- Reading the values of Envronment Variables, INI entries, and Registration Database keys into script variables.
- Custom Dialog Editor
- Custom Graphic Editor
- 256 Color Bitmap Support
- Long filename support under Windows NT and Window 95
- Password Protected Installations
- Automatic replacement of in-use files

Wise Installation Expert

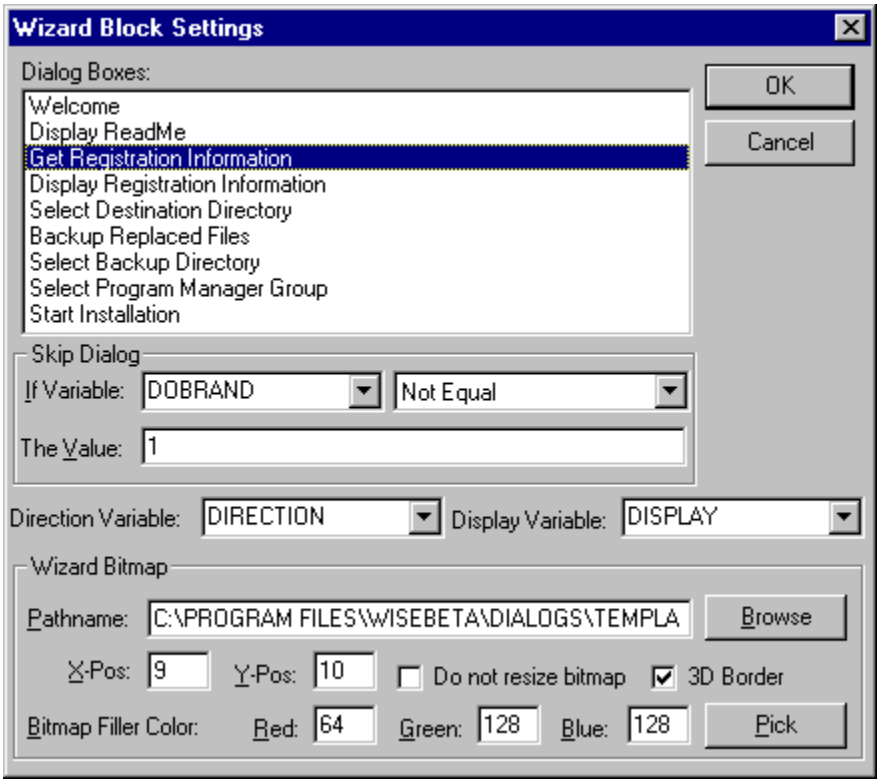
The Wise Installation Expert creates and edits your installations.



Wizard Block

This script item controls the display a set of Wizard dialog boxes. To better understand this, we suggest you first use the Installation Expert to create a script so that you can see how this script item works.

You need to end the Wizard Block with an "End Statement" script item.



Wizard Block Bitmaps

Wizard Bitmap Pathname: You can display a single bitmap in all the wizard dialogs. This should be a standard size graphic that will fit within the wizard dialogs. A sample graphic (WIZARD.BMP) can be found in the DIALOGS\TEMPLATES directory below your main WISE Directory.

Wizard Bitmap X-Pos/Y-Pos: This is a relative placement of the graphic within the dialog.

Wizard Bitmap Do not resize bitmap: Some graphics do not look good when they are resized. You may want the graphic to remain the exact size no matter what resolution the end-user has.

Wizard Bitmap 3D will add a 3D border to your graphic.

Wizard Bitmap Filler Color: This color will be used around the edge of the graphic if the graphic is not sized at different resolutions.

Wizard Block Dialog Boxes

This is a list of all the dialog boxes within the Wizard. We suggest that each Custom Dialog script item contain only one Dialog. Each dialog may have a "Skip Check" placed on it with a simple If statement. By default, the dialog will be displayed. If the statement returns true, then the dialog will be skipped.

Wizard Block Direction Variable

The direction Variable holds the value specifying the direction the Wizard is heading. All of the next buttons will set this variable equal to "N" (for next), and all of the back buttons will set this variable equal to "B" (for back).

Wizard Block Display Variable

The Display Variable will hold the name of the dialog that will be executed this time in the loop. This variable should be placed into the Dialog Set Properties dialog in the Custom Dialog Editor for all of the dialogs in the wizard block.

Wizard Block Skip Dialog

By default all dialogs will be executed. In some circumstances you may not want to execute a specific dialog or set of dialogs. For example, you would likely skip a dialog when prompting for a backup directory. One dialog would ask the user if they want to backup the files being overwritten, and the next dialog would ask here the backup files should be placed. If you stored the question about backing up the files into the variable DOBACKUP, then the next dialog could reference that variable to see if it needed to be skipped. The Statement would be:

```
"If Variable" DOBACKUP "Equals" "The Value" "NO"
```

This would skip the dialog that would prompt the user for the directory to store the backup files.

Maximum Length to Read or Write

This field holds the maximum number of bytes to write to the file from the variable (for writing). For reading, this field holds the number of bytes to read from the file. And trailing spaces will be removed from the read data.

X-Position

This determines where the graphic is placed in the installation window. A larger value will move the graphic more to the right.

Y-Position

This value determine how high up in the window the graphic will appear. A larger value will move the graphic down in the window.

Zip Compatible Directory

You can check this box to make your installation executable compatible with programs that read the standard ZIP file format. This is useful when uploading a file to a public on-line service or BBS system. Your file can be scanned for viruses automatically by the on-line service or BBS operator. If you make your installation ZIP compatible, you cannot place a password on it or split the EXE across multiple floppies.