MicroHelp, Inc.

UnInstaller™ 2

Welcome

General Primers

UnInstaller Features Primer

List Boxes Primer

File Deletion Primer

SmartLinks Primer

Getting Help Primer

Report Setup Primer

Print Setup Primer

SmartDecoy Primer

UnInstaller Basics

MicroHelp UnInstaller 2

Programming Team:

Tim O'Pry
Ken Spreitzer
Mark Novisoff
Bill Locke
Roy Taylor
Bob Flickinger

Documentation Team:

Mark Novisoff
B.J. Traylor
Tina Crawford
Craig Leach
Danny Sullivan

Published by:

MicroHelp Inc.
4359 Shallowford Ind. Pkway
Marietta, GA   30066-1135   (USA)

Sales:	(800) 922-3383
Tech Support:	(404) 516-0898
GA & outside the U.S.:	(404) 516-0899
Fax:	(404) 516-1099
FastFax:	(404) 591-6454

Introduction

A personal computer should be easy to use.   Microsoft Windows tries to make this so by using icons to represent computer applications such as spreadsheets and word processors.   To run an application, you simply use your mouse to point and click on the application's icon.   This eliminates the need to type a command such as C:\123\123 in order to use your spreadsheet.

You'd think that removing an application would be a simple matter of deleting it's icon   unfortunately, this is not the case.   Instead, a computerized "seek-and-destroy" mission is necessary to properly remove a Windows application from your system.   Its an effort that can challenge even the most experienced users.

MicroHelp's *UnInstaller* application, part of *The MicroHelp Utilities for Windows*, makes the task of removing an application from your computer much easier.   *UnInstaller* analyzes your system and offers to delete everything necessary to properly remove an application.

UnInstaller helps you keep your computer free of unnecessary files and information.   As a result, you can regain valuable disk space, and you might even find that Windows runs faster!

Using Online Help

Help is at hand

All the main windows of UnInstaller (**UnInstall Applications**, **System Cleanup**, **INIClean**, **DupeFinder** and **OrphanFinder**) have interactive help.   When you see a picture of the screen in the help file, move your mouse across the picture. When the cursor changes to a hand, click the mouse to get further help about that area of the screen.

You Should Have

Please take a moment to verify that your package contains the following items and that your computer is properly equipped to run the software.

Package Contents

- 2 Installation disks, 3.5" 1.44 MB
- User Manual (this manual)
- Registration card

If any items are missing, please contact MicroHelp Customer Service immediately at (404) 516-0899.

Note: If your system has only 5.25" disk drives, please refer to your registration card. In the "Comments and Suggestions" area, please write "I need 5.25 inch disks." When we receive your registration card, we'll send you the 5.25" disks at no charge.

System Requirements

- Microsoft Windows version 3.1 or higher
- Microsoft Windows file PROGMAN.EXE. *Windows Program Manager does not need to be loaded but the file must be present.*
- Monitor supporting VGA or higher resolution.
- 3 MB hard disk space (on any drive) for storage of the program.
- 3.5" 1.44 MB floppy drive. *A 3.5" 720K drive cannot read the 1.44 MB setup disk, even though the disk fits in the drive*.
- Mouse (not required but recommended).

Reaching Us

MicroHelp's offices can be reached by phone any business day, between the hours of 9 a.m. and 5 p.m., Eastern Time.

Technical Support:	(404) 516-0898
Sales and Service:	(800) 922-3383 or (404) 516-0899
Fax:	(404) 516-1099
FastFax Service:	(404) 591-6454

Please note that technical support questions submitted in writing to MicroHelp, whether by letter or fax, are replied to via US Mail.

You may write to:

MicroHelp, Inc.
4359 Shallowford Industrial Parkway
Marietta, GA 30066

We also operate a 24-hour, multi-node Bulletin Board System (BBS) that you can call for technical support questions and product information. The telephone number is (404) 516-1497. The communications parameters are: no parity, 8 data bits and 1 stop bit (N-8-1). The BBS supports baud rates up to 19200.

You can also reach us in the MSBASIC forum on CompuServe. At the CompuServe prompt, type in GO MSBASIC and leave your message in the section "3rd Party Products" (section 17 as of this writing).

If you have a problem with UnInstaller, please include your serial number when you leave a message so we can locate your record in our customer database.

In addition to utility programs, MicroHelp publishes tools for programmers. From your fax phone, please call our FastFax Service at (404) 591-6454 for more information on any MicroHelp product.

UnInstaller Primer

What does UnInstaller do?

Imagine youve hired a secretary to help you with your work. When you want to write a letter, the secretary brings you a pen and paper. If you want to balance your checkbook, the secretary brings you a calculator.

Now picture Windows as your electronic secretary. As with a human secretary, Windows helps you do things, but with your computer. When you want to write a letter, Windows makes your word processor appear when you click on its icon. If you want to balance your checkbook, Windows has your spreadsheet ready for use.

Brains are the big difference between a human secretary and Windows. The human has them but Windows only seems like it does (usually!).

To illustrate this idea, suppose that you buy a new printing calculator for your office and that your secretary will always operate it for you. In the human world, you give the calculator to your secretary, who takes it out of its packaging and puts it in the storage cabinet for future use. The secretary also reads the instruction manual to learn how to use the new machine.

In the computer world of Windows, the calculator must be smart enough to tell Windows exactly what to do. It must jump out of its box, hand its instruction manual to Windows, and put itself away in the storage cabinet. After that, Windows reads the machines instruction manual everyday, so that it knows how to use the calculator when you need it.

Now imagine you want to get rid of the calculator, which you no longer use. A human secretary would merely remove the machine from the storage cabinet and get rid of it.

Windows, on the other hand, wouldnt know how to get rid of the calculator. It would remain in your storage cabinet, taking up valuable space. Windows also would continue reading the instruction manual everyday, even though you have no plans to use the calculator again.

◆ UnInstaller: Helping Windows take out the trash

UnInstaller comes to the rescue when Windows is loaded down with unnecessary applications. In our example, UnInstaller would remove the calculator and its instruction manual, freeing up storage space and keeping Windows from wasting time reading unnecessary information.

In computer talk, UnInstaller deletes an applications files from your hard disk your computers storage cabinet. This gives you room on the hard disk for other applications. UnInstaller also erases the instructions that applications leave for Windows.

Be sure to read *Smart Uninstalling* for an overview of what UnInstaller removes from your computer. For more details and a complete explanation of each option, see *UnInstallable Items*.

System Cleanup: Removing the "extras" you'll never use

The **System Cleanup** feature helps you remove extraneous files that you no longer need. When Windows sets itself up, it installs a large number of files, many of which serve the same purpose, "just in case" you want to use them. Most likely, you will only use a few of these files, while the "extras" remain on your system and take up disk space. **System Cleanup** will show you these files and let you "house clean" your system and unload the "extras," much like you would clean your house by having a garage sale. See *Using System Cleanup* for more details.

INIClean: Cleaning up the messes made before you had UnInstaller

The **INIClean** feature helps you when youve deleted an application without using UnInstaller. You should use **INIClean** if the application that you've removed (before acquiring UnInstaller) was simple and probably added only an INI file to your computer. If the application was more complicated, you can use UnInstaller to find *all* remaining traces of the application. You can also use UnInstaller to remove all remaining traces of an application which you've deleted manually. See *Using INIClean* for more information.

Duplicate File Finder: The "de-redundancy" treatment

The **Duplicate File Finder** (from now on, we'll call it **DupeFinder**) feature helps you locate and remove extra copies of files that reside on your system. Many times, applications use the same file and will therefore place copies of the file in various directories. As long as one copy of the file is present in the search path (see *DOS Primer*) the other copies can usually be removed. See *Using DupeFinder* for more details.

OrphanFinder: Tossing out the leftovers

OrphanFinder is used to view and remove orphaned files. (The definition of an "orphan" is a Windows EXE that is not referenced, or installed, in your shell (desktop), or a DLL that is not used by any of the applications referenced in the shell.) See *Using Orphan Finder* for more details.

• Restore Backup

Restore Backup restores applications that were uninstalled using UnInstaller, provided that the Backup check box was checked. Restore Backup restores an application's files, sections in WIN.INI and document files to their original location. See *Using Restore Backup* for more details.

SYSTEM.INI and WIN.INI Primer

Helping the secretary know what to do

Lets go back to the idea that you have a secretary helping you work. This secretary is super-efficient and always knows what you want. For example, when you come to work, you always find a copy of your favorite newspaper sitting out on your desk.

In order to maintain the efficiency level, your secretary keeps lists of important things to remember. One list tells the secretary about the basic functions of the office, where the light switches are, how to operate the alarm systems, and so on. The other list tells the secretary about things needed for specific tasks, i.e. where to find the type of writing paper you like, where the office dictionary is kept, which newspapers you want placed on your desktop when you arrive at work in the morning, etc.

Like the human secretary, your electronic secretary, Windows, keeps lists of important things to remember. These lists are kept in two files called ***SYSTEM.INI*** and ***WIN.INI,*** which are read whenever you start Windows. In addition, Windows may read WIN.INI again when you run certain applications.

SYSTEM.INI: Your needs may change

SYSTEM.INI is the first INI file that is loaded upon Windows startup. It contains information about how the basic resources of Windows should be used to accommodate specific hardware, drivers and certain applications. When you first install Windows, the SYSTEM.INI list of instructions is set to defaults. As you install new applications or hardware, or make modifications to your Windows environment, these settings may be changed and new information or sections may be added.

Lets return to the human world, where youve instituted a new security system in your office that utilizes sound sensors to alert a security company when they are tripped. This system comes with a manual that is presented to the secretary. It teaches her how the keypad and passcode systems work. The manual also tells her:

- As soon as you come in, turn off the alarm system.
- The passcode that the alarm system responds to is 13556.
- Always turn the alarm system on again at night.

Thanks to these instructions, your secretary knows how to operate the new security system.

Later on, you decide to implement a new security system that uses security guards instead of a remote alarm setup. Your secretary no longer needs to make use of the remote alarm system's keypads. In fact, you no longer need the keypads at all. But unless you change your instructions to the secretary and let her know that this is no longer necessary, she will keep spending resources (time and energy) turning the alarm system on and off, and memory to remember the passcodes, while still more resources (money) are being spent elsewhere on the same thing (the guards).

Hardware drivers and applications written for Windows, like the remote alarm system, come with instructions that Windows reads so it knows how to install and run them.   Information is sometimes entered into SYSTEM.INI for drivers and applications that need basic Windows resources allocated for them.   But when you want to get rid of one of these applications, these instructions are usually left behind in the SYSTEM.INI file.

Though SYSTEM.INI is loaded only once, cleaning the SYSTEM.INI file may free Windows resources that are allocated upon starting Windows and held in reserve throughout your Windows session by drivers that you may no longer be using.

WIN.INI: A long list can be a bad thing

When you first install Windows, the WIN.INI list of instructions is relatively short. It usually grows longer each time you install a new Windows application, since many applications add instructions to WIN.INI that help Windows do its job.

Lets return once again to the office, where youve purchased a new calculator. The calculator is smart.   The first time you plug it in, it immediately seeks out your secretarys things to remember list and writes:

- The calculator is kept on the shelf behind your desk.
- Always put the calculator on your boss desk in the morning.
- Always turn the calculators print function on.
- Always put the calculator away at night.

Thanks to these instructions, your secretary knows what to do with the new machine.   When you get to work, the calculator is sitting out on your desk, ready to use.

Windows applications, like the calculator, are smart enough to teach Windows what to do.   But when you want to get rid of these applications, only a few of them are smart enough to erase the files that theyve added to your computer. An even smaller number of them can remove all the instructions theyve left for Windows to read in the WIN.INI file.

Remember, Windows may read through the WIN.INI list more than once. Eliminating unnecessary information helps prevent Windows from spending unnecessary time going through the list, and helps you get back to work quicker. Disk space is also freed when you get rid of unnecessary files.

The INI Files and UnInstaller

UnInstaller reads SYSTEM.INI and WIN.INI to find instructions pertaining to the application that you are removing.   It then offers to remove those instructions for you, so that these files are kept uncluttered and aren't wasting resources.

The **INIClean** feature helps you review and clean-up the WIN.INI file when you are not removing an application.   You can use **INIClean** to display non-standard sections of SYSTEM.INI and WIN.INI.   Non-standard sections usually have information that was not present when you first installed Windows right out of the box. By viewing only non-standard sections, you can easily spot instructions which may no longer be needed.

Note: Changes made to SYSTEM.INI or WIN.INI do not take effect until Windows is shut down and restarted.

Note: Though you can review SYSTEM.INI using **INIClean**, you cannot make modifications to it.   The SYSTEM.INI file is critical to the operation of Windows and should only be modified by advanced users.

Associations Primer

How Windows knows what to run

Applications have a way of telling Windows about *associations*. To understand associations, imagine youre writing a letter to your friend Anna and you're using the Microsoft Word for Windows word processor. When youve finished writing your letter and try to save it, Word asks you to name the file. You might type in ANNA, in which case Word calls the document ANNA.DOC.

Word adds the ".DOC" portion of the name for the same reason that a child gets a parents last name so that you know who the parents are. Word sees .DOC and knows that ANNA is one of its babies. In computer terms, the ".DOC" portion of the file name is called an *extension*.

If you double-click on the filename ANNA.DOC in File Manager, Windows sees the DOC extension and immediately knows that the letter was written using Word. Windows loads Word, which in turn displays your letter to Anna. This is much more convenient than loading Word and issuing a command for Word to load your letter.

Teaching Windows

When Windows is installed, it doesn't know that DOC is the extension used for Word files it has to be given instructions that tell it about this type of association. Setting up this kind of instruction is one of the things that applications do when they are installed on your computer.

For instance, when you install Word, the Setup program writes a note for Windows that basically says, "*All files that have the extension DOC should be associated with the Word application*." Word leaves this note in the WIN.INI file, which we talked about in the previous primer. Without that note, Windows wouldnt know to load Word when you double-click on the file ANNA.DOC in the File Manager.

Note: Windows does understand a few associations when it's first installed on your system. For example, if it sees an extension of BMP, it automatically loads the Paintbrush application. Likewise, an extension of TXT causes Windows to load a file into the Notepad application.

Extensions and applications that Windows knows about when it's first installed are called *standard associations*. **INIClean** can be instructed not to display the standard associations, so that you can concentrate only on associations that have been added after Windows was initially installed.

Associations and UnInstaller

UnInstaller finds instructions in WIN.INI that applications leave about their associations, and offers to delete them. UnInstaller also finds data files that are associated (by their extensions) with these applications. You then have the choice of deleting all data files, deleting some of them, or keeping all of them.

The **INIClean** feature shows you the associations listed in WIN.INI and tells you to which application each extension is associated. Armed with that information, you can delete any associations that belong to applications that are no longer installed on your computer.

Sections Primer

Where applications tuck away instructions

As described earlier, many applications leave notes about associations or other comments in the SYSTEM.INI or WIN.INI file all notes about filename extensions are kept in the [Extensions] section of WIN.INI. However, SYSTEM.INI and WIN.INI have many different sections, some of which are used by Windows itself and some of which are added by other applications.

For example, Word creates a [Word] section in WIN.INI. This section tells Word where to find special files used to import and export documents created by other word processing applications.

In general, it is considered rude for an application to add sections to SYSTEM.INI or WIN.INI a "well-behaved" application should create its own INI file (for example, WORD.INI) to store its private information. In fact, such files are referred to as "private INI files." Some applications (such as Word) create their own INI file but also add sections to WIN.INI as well. This is sometimes necessary, especially if the information is to be shared among several applications. (The next subchapter discusses private INI files.)

Sections and UnInstaller

UnInstaller lists the sections in WIN.INI that were created by the application you are removing. Its up to you to determine whether or not to delete a section. This is because some applications may share sections and it can be difficult to guess which sections are used by which application. You can learn more about this topic by reading the section called *Smart Uninstalling*.

The **INIClean** feature also lists sections in SYSTEM.INI and WIN.INI so that you can eliminate (WIN.INI only) those that are no longer needed. If you prefer, **INIClean** can hide standard sections (those sections that are part of SYSTEM.INI and WIN.INI when Windows is first installed.)

INIClean also allows you to look inside sections, and even make changes to sections in WIN.INI, which is useful for advanced users.

Private INI Files Primer

Yet another place where instructions are kept

Some applications add entries to the SYSTEM.INI or WIN.INI in order to run correctly.   Other applications simply create their own "private" INI files to hold the information they need.

INI files and UnInstaller

UnInstaller finds and suggests deletion of the INI files which belong to applications you remove.   The **INIClean** feature lists the INI files in your Windows directory so that you can eliminate those that are not needed. **INIClean** also allows you to view and edit the contents of INI files.   As with associations, **INIClean** can be configured so it doesn't show *standard INI files* (the INI files that are part of Windows when it is first installed).

Group Files Primer

Departments of Windows

Groups are the main "boxes" that organize applications. Typical group titles are: Main, Accessories, Games, Startup, etc. Group files (having an extension of GRP) are the "maps" of each group. These files contain the information about the applications in each group, and the locations of the files that are needed to start and run these applications. The shell program (Program Manager, for example) that is being used may have its own file or files in which it keeps group information, or it may use the Program Manager's GRP files.

Let's use our office setting as an example once again. Your secretary happens to be in charge of the other secretaries in the office. Every morning as one of her duties, she passes out group or department maps to each group's secretary. Each secretary gets a map of her own group which tells her who works in that group and where each person's desk is. It also tells her what and where the important items are that each person needs in order to do their job. Each secretary needs to know only about her own group. Your secretary keeps track of how to get information about any group by using her list of group secretaries.

Relating this to the computer, each group secretary is a GRP file with all the information about each group. Each person in the group is an application. The items at each person's (application's) desk that they use to work are the application's file information or properties.

GRP Files and UnInstaller

UnInstaller will always update GRP files so that changes made by uninstalling an application will be reflected in your desktop. Some shell programs, such as PCTools for Windows, interfere with this process. In such cases, you will need to update the changes to your groups manually by deleting icons or removing a group if it is no longer necessary.

Some shell programs, such as Norton Desktop for Windows™ (NDW) and Dashboard™, allow you to create subgroups, or groups within groups. If you want to uninstall an application contained within a subgroup, you can do so by using any one of the following methods:

- Move the application to a main group and then uninstall normally.
- Choose **UnInstall Applications** from the Tools menu and press the **Find**... button.
- Choose **UnInstall Applications** from the Tools menu and press the **Scan**... button.

Fonts Primer

What is a font anyway?

In Windows, *font* is the name for a typeface. Fonts are used to display letters on devices, such as video monitors and printers. Fonts are kept in various types of files, depending on how they were designed and implemented. When Windows is installed, it loads some fonts onto your hard disk and inserts references to them into the WIN.INI file this gives you the minimum number of fonts required to run Windows.

With the introduction of Windows version 3.1, Microsoft® added TrueType® for Windows technology and distributed several TrueType font files with the Windows package.

Windows font types

The three types of fonts that Windows uses are raster fonts, vector fonts, and TrueType fonts.

Raster fonts are bit-mapped fonts that use dots for displaying and/or printing. Raster fonts are not designed to be scaled and are most attractive at small sizes.

Vector fonts are rendered mathematically by connecting dots with lines. They are perfectly suited for output to pen plotters because they can be scaled and rotated, but are not very attractive for monitor or printer output. Their use has diminished considerably since the introduction of technologies such as TrueType and Postscript.

TrueType and **PostScript Type 1 fonts** are outline fonts that can be scaled to any size.

See also: *Standard fonts* and *Standard Font File extensions*.

Fonts and UnInstaller

You should probably keep the System fonts on your system, since they are used throughout Windows as the default for many Windows applications. If you do not have a pen plotter, you can remove the vector fonts Modern, Roman, and Script. If you use TrueType or Adobe Type Manager (ATM), you can remove the other raster fonts, including Courier, MS Sans Serif, MS Serif, Small, and Symbol.

Removing TrueType and/or ATM fonts is a personal decision, depending on the typefaces you use in your applications. If you try to use a font that is not installed, Windows does its best to make an accurate substitution.

Drivers Primer

Talking to your peripherals

Note: Removing drivers from a system should be done only by advanced users who have a good understanding of Windows and their systems.

Windows interfaces with peripherals such as monitors, keyboards, printers, plotters, pointing devices, networks and memory through the use of *drivers*. Basically, a driver is a translator between Windows-specific commands and peripheral-specific commands. Think of this as two people speaking different language, using an interpreter to translate their words. The interpreter's function is the same as a driver's function.

The standard extension for driver files is DRV. However, driver files can have any legal DOS file extension. Other extensions used for drivers are 386 (Windows enhanced mode), DLL (Dynamic Link Library) and EXE (Executable).

There may also be related files for drivers. For instance, when using a Postscript printer and PSCRIPT.DRV (the standard Postscript driver), there may be a corresponding WPD (Windows Postscript Definition) file for your printer.

Most of the drivers that Windows uses are listed in various sections of the SYSTEM.INI file. Use ***INIClean*** to list and view (but not edit) the sections of the SYSTEM.INI file, or let UnInstaller remove display drivers for you. In either case, make a backup copy of the SYSTEM.INI file before editing it.

Caution: We cannot stress the importance of the integrity of the SYSTEM.INI file enough. If it becomes corrupted, you may not be able to restart Windows.

Drivers and UnInstaller

UnInstaller's ***System Cleanup*** feature can find and clean up the unused non-standard display drivers from your system.

DOS Primer

The driving force

Another place that applications may add or modify information is in the main instruction files that DOS (Disk Operating System) uses CONFIG.SYS and AUTOEXEC.BAT. Perhaps it's best to think of DOS as the vehicle that takes you to Windows. Let's go back to the human world of business.

Suppose you are the president of the company where you work, and the company is doing very well. You decide that you'd like to get some work done on your long trips to and from work, so you hire a chauffeur. (You still go to work because you love your work!) Your chauffeur, just like your secretary, keeps lists of things to do. One of these lists tells him how to maintain and start up the car. Those instructions might include checking the oil, turning the ignition key, etc. The other list tells him how to arrange things to your tastes and preferences (temperature settings, radio volume, etc.).

Let's translate this into computer talk. Your chauffeur (DOS) brings you to your office (Windows) everyday. The "list" DOS uses to get everything running properly is the CONFIG.SYS file. The "list" DOS uses to know how to set things according to your preferences is the AUTOEXEC.BAT file.

Some Windows applications require certain drivers to be loaded. They do so because they need access to resources before Windows has the chance to use them up. The installation program of the application makes modifications to the AUTOEXEC.BAT or CONFIG.SYS. This would be the equivalent of your secretary leaving a message for your chauffeur, asking him to take care of something every morning before he brings you into work (Windows).

Another way of thinking of the CONFIG.SYS and AUTOEXEC.BAT files is that they are the DOS versions of the Windows' SYSTEM.INI and WIN.INI files, respectively.

Paths

One of the changes that a Windows application might make to your AUTOEXEC.BAT file is an addition to your PATH= statement. In order to explain paths better, let's go back to the human world. Let's say that your chauffeur moved here from another country. He was flown in by plane so that when he arrived, he didn't know any of the roads or places that you might want to go. During his training, he was given directions to most of the places that you might want to go. Every morning when he gets up, he needs to review his list of directions so he can remember how to get to any of the places on your list. Let's say the first place on the list is your home, the second is your office, the third is your favorite restaurant, and so on. This list is the equivalent of the path statement: a list of places to search through when you ask your chauffeur (DOS) to find something.

Let's say you are at the library and you ask your chauffeur to go pick up a pack of greeting cards because you have run out. Your chauffeur (DOS) has an unusual method of trying to find something. First he tries to find what you are looking for at the current location in this case, the library. Obviously, he wouldn't find the greeting cards there. Next, he would go to the first location in his list, home, and see if what he's looking for is there. If it is not there, he will continue on to the next location on the list, the office, search there, and so on. If he has not been given a direction sheet instructing him how to get to a gift shop, grocery store or mall, he will return and tell you he could not find what you asked for.

Although the way DOS searches the Path may sound inefficient, your chauffeur has an extraordinary talent: he can search all of the places on his list in less than one second!

This is how DOS operates. It searches the path statement, its list of directions, from first location to last location in search of the file you ask it to find. If it cannot find the file you requested in one of the locations specified in your path, it will return the message that it could not find the file. You will then have to go find the file yourself.

A Windows application may need to add a location to the list so that its files can be found by DOS. This is especially true for programs that must load drivers in DOS before they can run in Windows. Therefore, these applications will add to the path statement (list), which is located in the AUTOEXEC.BAT.

DOS and UnInstaller

UnInstaller finds the instructions and references that applications leave in the AUTOEXEC.BAT and CONFIG.SYS files, and offers to remove them. UnInstaller also offers to remove primary batch files and PIF files that may be used to run DOS applications through Windows.

Note: UnInstaller will search and remove only *primary* batch files, and the associated executables, but it will not search or remove a "chain" of batch files that are called within other batch files.

The **Duplicate File Finder** feature shows you how many copies of files you have and in which directories they may be found so that you may remove the extra copies and free up disk space. You should be certain that you are thoroughly familiar and comfortable with the concept of Path before you remove any duplicate files. For further information, see *Smart Duplicate File Deletion*.

Shells Primer

What does your Windows look like?

A "shell" is the program that determines what your Windows environment looks and "feels" like   it's the first thing you see when you start Windows.   The shell will always have a title bar at the top to let you know what shell program you are running.   It may or may not have a menu.   The shell organizes your application icons and group windows.   Your shell program, in short, is your "desktop" organizer.

The standard Windows shell is Program Manager.   Other "aftermarket" shell programs from various vendors are also available.   Windows determines which shell program to run by looking at the [boot] section of the SYSTEM.INI file.   In that section, you'll find a line beginning with "*shell* = " and ending with the name of the executable (EXE) of the shell program to be used.   For instance, if you are using Program Manager as your shell, the line reads:

```
shell = progman.exe
```

When you install an aftermarket shell program, the install program for that shell will modify this line appropriately.   The different types of shells can be broken down into 3 categories:

Program Manager clones

This type of shell reads and updates the Program Manager GRP files directly. Dashboard is an example of this type of shell.

Program Manager imitators

This type of shell intercepts and processes all Program Manager requests transparently, without loading Program Manager.   Norton Desktop for Windows (NDW) is an example of this type of shell.

Program Manager replacements

This type of shell is not aware of Program Manager except when it is installed, at which time it copies the current GRP settings to its own files.   These shells keep all data in proprietary files and ignores all Program Manager requests.   PCTools for Windows (PCTW) is an example of this type of shell.

Note: This chapter contains primers for Norton Desktop for Windows and PCTools for Windows

Shells and UnInstaller

UnInstaller has been tested with the most popular aftermarket and shareware shell programs, including Dashboard, Norton Desktop for Windows (NDW) and PCTools for Windows (PCTW).   All except PCTW will respond normally to UnInstaller.   PCTW does not interface with Program Manager in any way and uses its own inaccessible proprietary database instead of Program Manager's GRP files.   This means that when you go to uninstall an application, you may need to search for the application using the **Find...** button rather than selecting it from the application list.   PCTW also ignores DDE read requests, meaning that UnInstaller is unable to delete PCTW folders when they are no longer needed. Because of this, you will need to remove uninstalled application icons and group folders manually.

UnInstaller is capable of uninstalling a shell program, even if it is the currently operating shell program. If the shell selected for removal is the current shell, UnInstaller will modify the "shell=" line in SYSTEM.INI and then restart Windows. You may then select the old shell program for removal as you would any other application.

Norton Desktop for Windows Primer

What if you are using NDW?

The following information is a summary of all the text in this manual that is applicable to Norton Desktop for Windows (NDW). We've gathered it together simply so that you'll have everything about NDW in one place.

NDW is known as a "Program Manager imitator," because it intercepts and processes all Program Manager requests transparently, without loading Program Manager.

Best results are obtained if you do not load Program Manager while running NDW. The reason is that NDW and UnInstaller communicate directly via DDE (Dynamic Data Exchange). If Program Manager is loaded, it will process our DDE requests without passing them along to NDW. The result is that applications installed only in NDW, and not in Program Manager, won't appear in the list of applications from which you choose to uninstall. In addition, the application's icon will be deleted in Program Manager but not in NDW.

If you are using NDW and have the desktop minimized, the desktop will automatically normalize whenever UnInstaller queries the shell for group or program information. We have no control over this as NDW does this on each DDE request that we make.

UnInstaller is impacted by NDW in two ways:

- An applications icon may not be removed from its shell program group window. This happens if the icon for the application that you are uninstalling is in a different group under the shell than it is under Program Manager. If this happens, then you can manually delete the icon from the shell program: simply switch to your shell program (NDW, PCTW, etc.), open the appropriate group window, select the icon to be deleted, and follow directions for deleting icons provided in your shell program manual. (Use the **<Del>** key to tell NDW to delete an icon.)
- Groups within groups (or subgroups) are **not** supported. The master group will appear in the list box, but double-clicking on it will give a DDE error.

UnInstaller can remove NDW, even if it is the current shell! Here's how it works:

- Select the NDW icon in UnInstaller. If NDW is the current shell, UnInstaller will modify the "shell=" line in SYSTEM.INI so that Program Manager is the shell and then restart Windows. (Note: If Program Manager cannot be found, an error message will be displayed and the selection will be canceled.)
- Select NDW for removal as you would any other application.

PCTools for Windows Primer

What if you are using PCTW?

The following information is a summary of all the text in this manual that is applicable to PCTools for Windows (PCTW). We've gathered it together simply so that you'll have everything about PCTW in one place.

PCTW is considered a Program Manager replacement. When you install PCTW, it copies the current GRP settings to its own files. From that point on, PCTW keeps all data in proprietary files and ignores all Program Manager requests. This means that when you go to uninstall an application, you may need to search for the application using the **Find...** button rather than selecting it from the application list. PCTW also ignores DDE requests, meaning that UnInstaller is unable to delete PCTW folders when they are no longer needed. Because of this, you will need to remove uninstalled application icons and group folders manually.

Orphan Finder uses DDE to request a list of all installed applications from the shell. Because PCTW does not support DDE, ***Orphan Finder*** cannot obtain a complete list of applications. (In case you are interested, we have contacted Central Point Software, the publishers of PCTW, explained the problem and asked for assistance. They have not responded as of this writing.)

If you are using PCTools for Windows (PCTW), or another shell that doesn't support DDE requests, you should start the ***Orphan Finder*** feature and use the **Scan Shell** feature, described under *Advanced Options*. After scanning the shell, you should use the **Add File** button to make sure that ***Orphan Finder*** knows about all the installed Windows applications in your system. Press the **Add File** button, and add the file name, for each Windows application that is installed in your system, but not shown in the list box generated as a result of the **Scan Shell** operation.

UnInstaller is impacted by PCTW in three ways:

- When selecting an application to be uninstalled, UnInstaller searches only your Program Manager groups for installed applications. If an application was installed only into the shell program but not Program Manager, then UnInstaller will not find or list it. In this case, choose the **Find...** button to manually locate the application by filename.

- An application's icon may not be removed from its shell program group window. This happens if the icon for the application that you are uninstalling is in a different group under the shell than it is under Program Manager. If this happens, then you can manually delete the icon from the shell program: simply switch to your shell program (NDW, PCTW, etc.), open the appropriate group window, select the icon to be deleted, and follow directions for deleting icons provided in your shell program manual. (The key is used to tell PCTW to delete an icon.)

- Groups within groups (or subgroups) are **not** supported. The master group will appear in the list box, but double-clicking on it will give a DDE error.

UnInstaller can remove PCTW, even if it is the current shell! Here's how it works:

- Select the PCTW icon in UnInstaller. If PCTW is the current shell, UnInstaller will modify the "shell=" line in SYSTEM.INI so that Program Manager is the shell and then restart Windows. (Note: If Program Manager cannot be found, an error message will be displayed and the selection will be canceled.)

- Select PCTW for removal as you would any other application.

I use PCTW and tried to start up Program Manager but it wouldn't load. Is there some way I can start it, or is something wrong?
Yes. UnInstaller loads Program Manager in the background and hides it. When this occurs, an **Options** Menu selection, **Show Program Manager** will appear. Selecting this option will display Program Manager as a minimized application.

If UnInstaller successfully loaded Program Manager, it will automatically close Program Manager when the uninstall session is completed. If you shut down Program Manager manually before ending your uninstall session, it will usually be reloaded into the background again. However, you may get a warning of a "failed DDE request." The best advice is to avoid using Program Manager while using UnInstaller with PCTW.

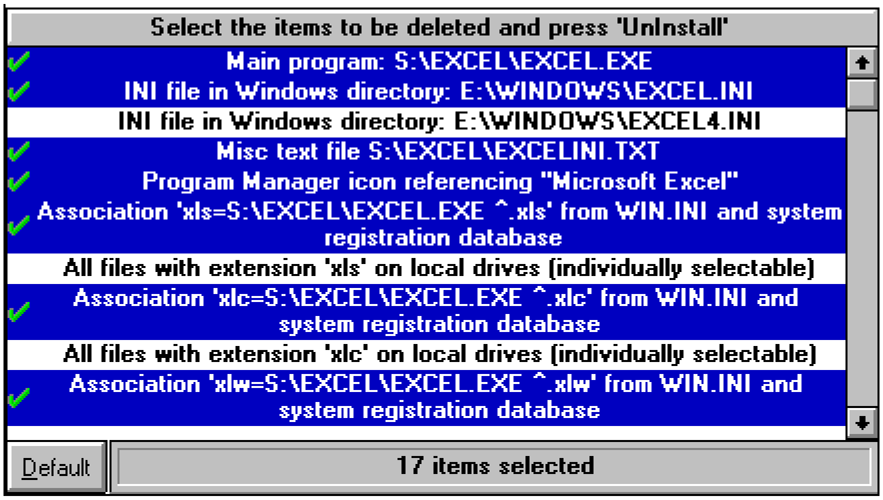
List Boxes Primer

How to select items in a list box

Several of UnInstaller's tools let you select one or more items within a list box. When an item in a list box is selected, or *highlighted*, Windows changes the color used to display the item. If you are using the Windows default colors, selected items are shown in white text on a blue background. However, you can change the colors used to highlight items using the Color option in the Windows Control Panel. Listed below are the two types of list boxes that the UnInstaller tools use for item selection.

Simple selection list boxes

Each item in the list box can be selected or deselected individually. You may select any number of items in the list box.



Select the items to be deleted and press 'UnInstall'

✓ Main program: S:\EXCEL\EXCEL.EXE
✓ INI file in Windows directory: E:\WINDOWS\EXCEL.INI
INI file in Windows directory: E:\WINDOWS\EXCEL4.INI
✓ Misc text file S:\EXCEL\EXCELINI.TXT
✓ Program Manager icon referencing "Microsoft Excel"
✓ Association 'xls=S:\EXCEL\EXCEL.EXE ^.xls' from WIN.INI and system registration database
All files with extension 'xls' on local drives (individually selectable)
✓ Association 'xlc=S:\EXCEL\EXCEL.EXE ^.xlc' from WIN.INI and system registration database
All files with extension 'xlc' on local drives (individually selectable)
✓ Association 'xlw=S:\EXCEL\EXCEL.EXE ^.xlw' from WIN.INI and system registration database

17 items selected

A simple selection list box

To select or deselect multiple nonsequential items:

Mouse

- Click on the item.

Keyboard

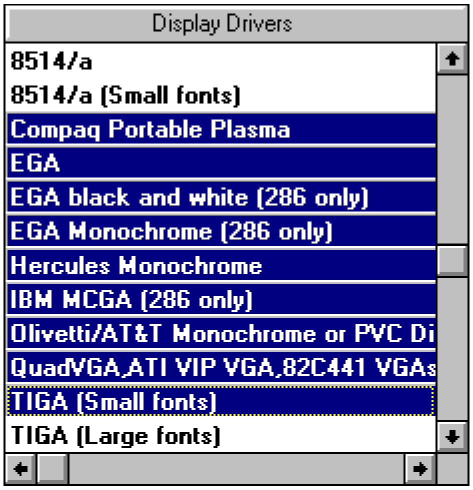
- Use the arrow keys to move to an item, then press the **<Space Bar>**.

A simple selection list box is used in the following tool:

Tool	**List Box Caption**
UnInstaller	Select the items to be deleted...

Extended selection list boxes

Multiple items in a list box can be selected as a group.



Display Drivers

8514/a
8514/a (Small fonts)
Compaq Portable Plasma
EGA
EGA black and white (286 only)
EGA Monochrome (286 only)
Hercules Monochrome
IBM MCGA (286 only)
Olivetti/AT&T Monochrome or PVC Di
QuadVGA,ATI VIP VGA,82C441 VGAs
TIGA (Small fonts)
TIGA (Large fonts)

An extended selection list box

To select or deselect multiple items:

Mouse

- Using the mouse, click on the first item to be selected, and while holding down the left mouse button, drag the cursor to the last item to be selected.

Or:

- Using the keyboard, **<Shift+Click>** extends the selection from the previously selected item to the current item. **<Ctrl+Click>** selects or deselects an item in the list.

Keyboard

- **<Shift+Arrow key>** extends the selection from the previously selected item to the current item.
- **<Ctrl+/>** selects all items in the list.
- **<Ctrl+\>** deselects all items in the list.
- **<Ctrl+*letter*>** selects the first item found in the list that starts with the specified letter.
- **<Shift+*letter*>** selects all items between and including the currently selected item and the first item found in the list that starts with the specified letter.

Note: When you begin selecting a new item or items, previously selected items are deselected.

Extended selection list boxes are used in the following tools:

Tool	**List Box Caption**
UnInstaller - Confirm Deletions	Delete all local documents
System Cleanup - Display Drivers	Display Drivers
System Cleanup - System Support Files	System Support Files
System Cleanup - Windows Fonts	Installed Fonts
System Cleanup - Windows Fonts	Available Fonts
System Cleanup - Non-Windows Apps	Support Files
Orphan Finder - Add File	Select the program to be added
Duplicate File Finder	File, Path, etc.

File Deletion Primer

How does UnInstaller select files for deletion?

UnInstaller selects files by looking for certain file extensions and comparing those file names with the name of the executable file. The first step in the process is to determine the name of the executable file associated with the application. For items installed in the shell, this is simply a matter of requesting the EXE name. For other applications, UnInstaller has a **Find...** and **Scan...** feature, both of which let you choose the executable.

After determining the EXE name, UnInstaller looks for files where the first three letters of the file name match the first three letters of the executable file and have one of the following file extensions: INI, CFG, FLT, TXT, DAT, SET, LOG, ICO, or BAK. The exception to this is if the first three letters of the executable are WIN in this case, UnInstaller will use the first four letters instead. Also, if the name of the executable is shorter than three letters, it will use exactly the number of letters that are present. For example, if the application executable file is AB.EXE, UnInstaller will look for AB*.INI, etc.

For instance, if you are uninstalling Microsoft Word for Windows (WINWORD.EXE), UnInstaller will search for WINW*.INI, WINW*.CFG, WINW*.FLT, WINW*.TXT, WINW*.DAT, WINW*.SET, WINW*.LOG, WINW*.ICO, and WINW*.BAK in the \WINDOWS and \WINDOWS\SYSTEM directories. It will place the matching files into the **Items to be deleted** list box. None of these selections will be highlighted automatically UnInstaller will require you to select them if you want them to be deleted.

UnInstaller also scans the executable file (EXE) for DLL files that are referenced within the application and includes those files in the list box.

SmartLinks Primer

Performing a more thorough uninstall

Note: This primer is for advanced users who have a good general understanding of how Windows operates and of the distinction between EXE and DLL files.

Note: The ***Orphan Finder*** feature also makes use of the SmartLinks data files. If the **EXEs Only** check box is NOT checked, ***Orphan Finder*** will perform a SmartLinks scan and list any "orphaned" DLLs found. A SmartLinks scan is not performed if the **EXEs Only** box is checked.

UnInstaller includes an advanced feature, called SmartLinks, that scans all executable code on your system and attempts to determine which applications use which DLLs. This scanning may take several minutes the first time it is performed.

The results of the scan are saved to files named UNI2SML2.DTB and UNI2SML1.DTB in your UnInstaller program directory. Subsequent scans merely update these data lists and are therefore completed more quickly.

To perform a SmartLinks scan, select the **Create/Update SmartLinks** option from the **File** menu at any time. Once your system has been scanned, UnInstaller can access the SmartLinks cross-reference files which list applications and related DLLs.

UnInstaller will access the SmartLinks data files, if they exist, when it analyzes an application for removal. UnInstaller will suggest for deletion all DLLs that it determines are used by the application. If SmartLinks does not know of any other applications that use the DLL, UnInstaller will display the message "**DLL *filename.ext* not used by scanned programs**." If SmartLinks knows that other applications use a DLL, the DLL is listed in a confirmation dialog box and the message "**DLL *filename.ext* is used by other programs**" is displayed.

If a SmartLinks scan has never been performed or a DLL cannot be found in the SmartLinks data files, the message "**DLL *filename.ext* may be used by other programs**" will appear. It is recommended that you Cancel the UnInstall and select the **Create/Update SmartLinks** menu option to perform a SmartLinks scan. You should then reselect the application for removal.

Note that it is not possible to create a totally perfect file links system with the range of applications available today. However, our testing has shown that SmartLinks is approximately 90% accurate in its assessment of your systems file links. Your mileage may vary!

Getting Help Primer

Help is just a keypress away

Help is always available when using UnInstaller by pressing the **<F1>** Key, the Help button, or choosing from the Help menu.

F1 Key: Help about what you are currently doing

Press **<F1>** anytime you have a question about something you are doing with UnInstaller to make information appear about the task at hand.

The Help menu: Full-service support

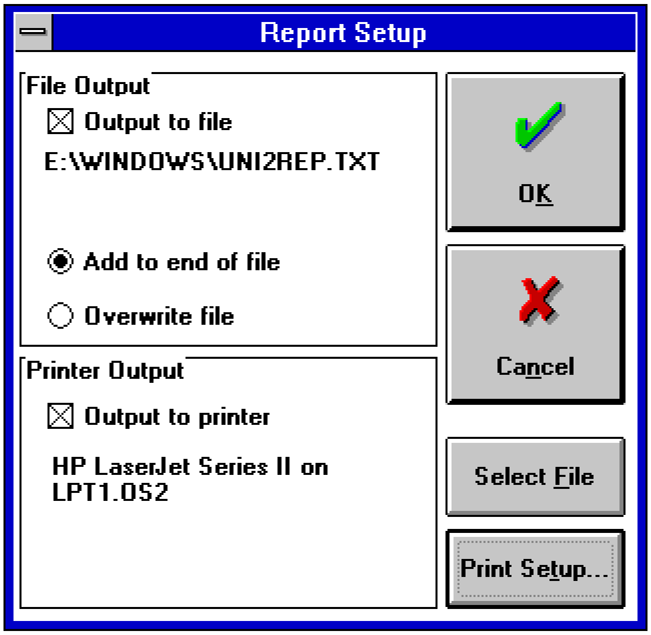
The Help menu is available on the main window of UnInstaller.   It provides several choices for assistance:

- **Contents** brings up the contents page of UnInstaller Help.
- **Search For Help On** brings up the UnInstaller Help search dialog.
- **Technical Support** displays MicroHelps technical support phone number.
- **How to Use Help** displays the help file that gives you detailed instructions on using Windows Help in general.
- **About...** displays information about UnInstaller.   This menu command also displays the MicroHelp technical support phone number and the amount of disk space freed by UnInstaller.

Report Setup Primer

To set report options

1. Select **Report Setup** from the **File** menu or press the report setup button shown above.



☒ Output to file

E:\WINDOWS\UNI2REP.TXT

● Add to end of file

○ Overwrite file

☒ Output to printer

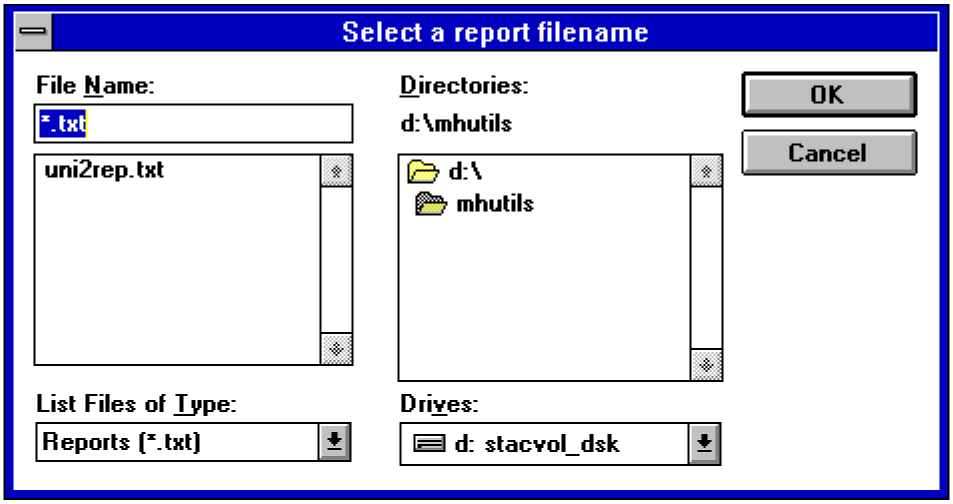
HP LaserJet Series II on LPT1.OS2

The Report Setup window

2. Check the **Output to file** box if you would like to create a report file or the **Output to printer** box if you would like to print out the report automatically. Check both boxes if you would like both options.

3. **Output to file:** Select where you would like the new report information to be placed. Select **Add to end of file** to add the new report information to the end of an existing file or select **Overwrite file** to overwrite a previous report's information with the new report information.

4. **Output to file:** If you would like to select a different file than the one specified for the report output, press the **Select File** button. The following dialog will be displayed:



File Name: *.txt

uni2rep.txt

Directories: d:\mhutils

d:\
mhutils

List Files of Type: Reports (*.txt)

Drives: d: stacvol_dsk

Use this dialog to select a file name if you want report output to go to a file

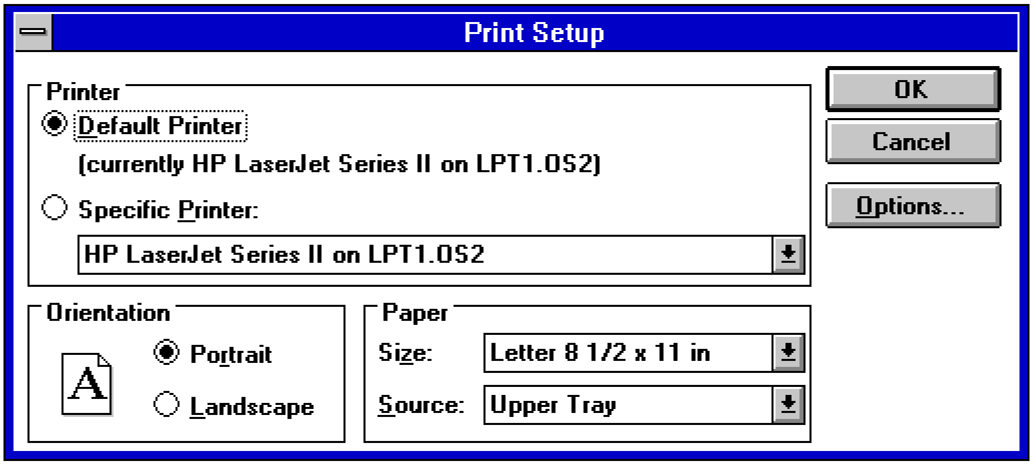
5. **Output to file:** Select or type in the name of the file you wish to send the report information to and press the **OK** button.

6. **Output to printer:** If you would like to select a different printer than the default printer, press the **Print Setup** button to display the **Print Setup** dialog. For further information, please see *Print Setup Primer*.

Print Setup Primer

To set printer options

1. Select **Print Setup** from the **File** menu or press the print setup button (shown above) on the button bar.



Print Setup

Printer

● Default Printer
(currently HP LaserJet Series II on LPT1.OS2)

○ Specific Printer:
HP LaserJet Series II on LPT1.OS2

Orientation

● Portrait
○ Landscape

Paper

Size: Letter 8 1/2 x 11 in
Source: Upper Tray

The Print Setup dialog

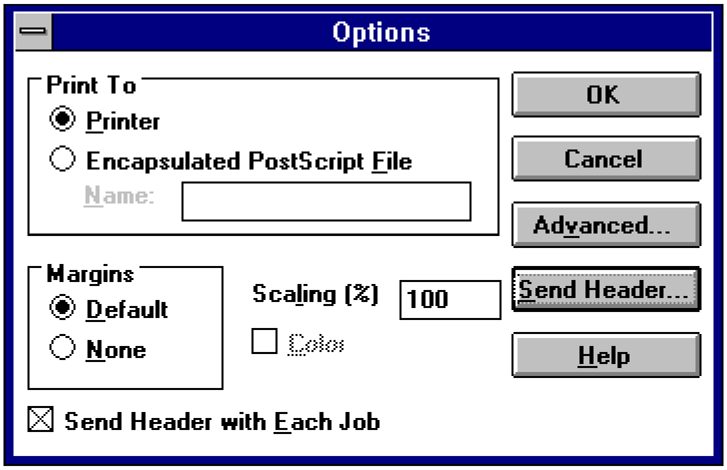
2. **Printer:** To select a printer, choose either the **Default Printer** or one of the printers from the **Specific Printer** drop down list.

3. **Orientation:** This will usually be selected automatically, depending on the printer selected, but you may change the default by selecting either the **Portrait** or **Landscape** option.

4. **Paper:** To make selections other than the defaults shown, both **Size** and **Source** have drop down lists from which you can make selections.

5. When you are satisfied with your choices, press the **OK** button to implement your changes or **Cancel** to undo any changes made.

6. **Options:** If you would like to view the non-standard print options, press the **Options** button to display the options dialog. Make the selections of your choice and, when complete, press the **OK** button to implement your changes or **Cancel** to undo any changes and return to the previous screen. For further information or assistance, press the **Help** button.



Options

Print To
● Printer
○ Encapsulated PostScript File
Name:

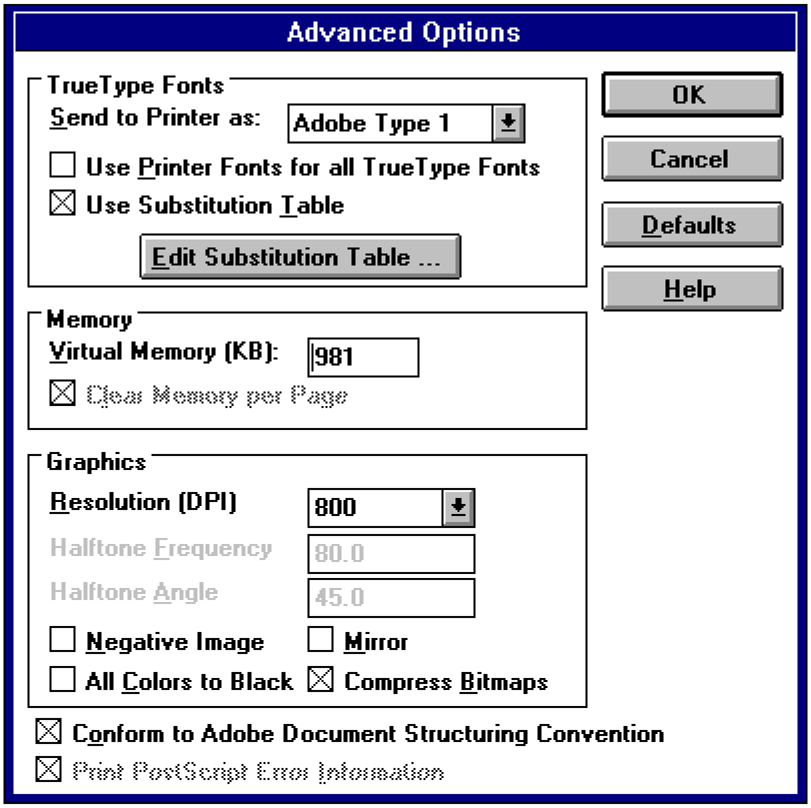
Margins
● Default
○ None

Scaling (%) 100
□ Color

☒ Send Header with Each Job

The Print Setup Options dialog

7. **Advanced:** If you would like to make advanced printer selections, press the **Advanced** button and the **Advanced Options** dialog will be displayed. Make the selections of your choice and, when complete, press the **OK** button to implement your changes or **Cancel** to undo any changes and return to the previous screen. Press the **Defaults** button to return all selections to default. For further information or assistance, press the **Help** button.



Advanced Options

TrueType Fonts
Send to Printer as: Adobe Type 1
□ Use Printer Fonts for all TrueType Fonts
☒ Use Substitution Table

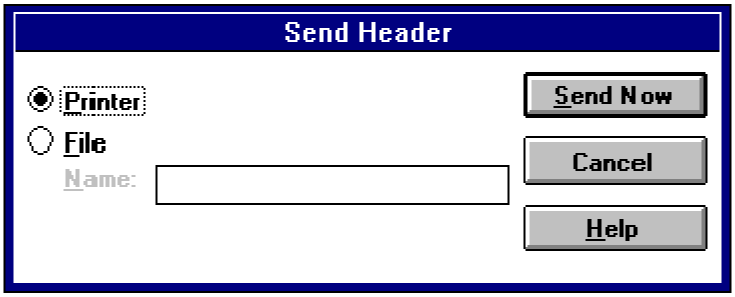
Memory
Virtual Memory (KB): 981
☒ Clear Memory per Page

Graphics
Resolution (DPI) 800
Halftone Frequency 80.0
Halftone Angle 45.0
□ Negative Image □ Mirror
□ All Colors to Black ☒ Compress Bitmaps

☒ Conform to Adobe Document Structuring Convention
☒ Print PostScript Error Information

The Print Setup Advanced Options dialog

8. **Send Header:** If you would like to send a header to a printer or to a file, press the **Send Header** button and the **Send Header** dialog will be displayed. Make the selections of your choice and, when complete, press the **Send Now** button to send the header or the **Cancel** button to undo any changes and return to the previous screen. For further information or assistance, press the **Help** button.



Send Header

● Printer

○ File

Name:

The Print Setup Send Header dialog

SmartDecoy Primer

Making life easier for network users

This primer is only for those users who have a network version of UnInstaller. The SmartDecoy option is not included in the single-user version of this product.

If you are the network administrator of a large network, consider this problem: UnInstaller makes it easy enough to uninstall an application from *your* system, but how do you remove all traces of the application for the *rest* of the users on your network?

Using SmartDecoy technology, UnInstaller can effortlessly clean up all systems on a computer network. This feature can save a network administrator countless hours of repetitive (and boring!) cleanup work across the network.

How SmartDecoy technology works

When you uninstall an application from a network, UnInstaller offers to create a SmartDecoy. If you decide to leave a SmartDecoy, a decoy copy of UnInstaller gets written (and renamed) over the original program filename. For example, if you are uninstalling N:\WINWORD\WINWORD.EXE, then a decoy copy of UnInstaller is written over WINWORD.EXE, just as if you had issued the DOS command:

```
COPY   UNINSTAL.EXE   N:\WINWORD\WINWORD.EXE
```

Then, whenever a user on the network tries to run Word, they actually run the SmartDecoy copy of UnInstaller. UnInstaller informs the user that the application has been uninstalled and offers to clean up the user's system.

Users who have multiple installations of Windows on their system (for example, separate Windows 3.0 and Windows 3.1 installations) may wish to leave a SmartDecoy installed so that it can clean up their system when they are running under a different installation of Windows.

Using SmartDecoy

When you press the Analyze button on the UnInstaller window, and you have selected an application that is installed on a network drive, one of the items that appears in the list is a line that says "Select this line to create a SmartDecoy." That's all you have to do!

Quick Uninstalling

Uninstalling an application is a four-step process. First, you select an application to uninstall. UnInstaller will analyze your system to find which items need to be deleted to remove the application. Second, you review the list of suggested deletions and, if desired, modify the list. Third, you choose whether you want a report detailing the uninstallation. Finally, you begin the actual uninstall process and confirm the deletion process when prompted.

Step 1: Selecting what to uninstall

Step 2: Review deletion suggestions

Step 3: The uninstall report

Step 4: Confirming deletions

Step 1: Selecting what to uninstall

1. You start the uninstall process by choosing **Tools-UnInstall Applications...** from the menu or by clicking the button shown above. Applications are listed under their shell groups.

2. Now do one of the following:

- Double-click on a group name to show the applications within that group. Click on the application to be removed and choose **Analyze**.

  Note: Pressing the **Analyze** button does not cause the application to be uninstalled.

- Choose **Scan...** to find applications that do not appear in the list. (You may need to choose Scan if you use a shell that does not support DDE requests, such as DDE.) The Scan feature will scan for all Windows executables on the specified drive, show you the icon and file description and allow you to select the application for removal.
- Choose **Find...** if the application you want to uninstall is not listed. Use the find dialog to select the application you want to uninstall. Choose the **OK** button in the **Find...** window and then choose **Analyze**.

  Note: If you select an application that is installed on a network server, and UnInstaller thinks that the application is in use by another person, UnInstaller will notify you and ask you if you wish to continue the uninstall process.

- Choose **Close** to close the Uninstall Applications window and return to the main window without selecting an application to uninstall.

3. When you select **Analyze**, UnInstaller analyzes your system to find all traces of the application you want to uninstall. Every trace that it finds is entered into the **Items to be deleted** list box.

4. If an application's name is four characters or shorter, a message appears warning you to look closely at the suggested deletions. This is necessary because UnInstaller recommends items for deletion by searching for the filename of the application that you are trying to delete. When you choose an application with a short file name, it's possible that the file name may be found elsewhere, but as a portion of a longer file name. For example, if you are a dentist and you have an application called "Pain," UnInstaller may find occurrences of "Paintbrush" that appear to be matches for "Pain."

Step 2: Review deletion suggestions

1. Review the Uninstall Application window's list box, which shows the programs, files and other items that UnInstaller suggests for deletion. One or more items will be pre-selected for deletion.
2. Select or deselect the items shown by clicking on them with the mouse or using the buttons below the list box. Using the keyboard, move with the arrow keys and select or deselect items by pressing the space bar.
3. Never select anything for deletion that you have doubts about. Refer to *Smart Uninstalling* for detailed advice about what should be removed from your system.
4. Advanced users: Use the **Add file** or **Add section** buttons to add items not shown on the list.
5. Advanced users: If you wish to bypass most delete confirmations, clear the **Confirm all deletions** check box on the main window. However, we recommend that you leave the box checked to provide maximum safety during the uninstall process.
6. Network administrators: Select the SmartDecoy item (always the last item in the list) if you wish to create a SmartDecoy. See *SmartDecoy Primer* for an introduction to SmartDecoys.
7. Select the **Backup** option to have UnInstaller make a backup of the primary files and INI changes it makes.

Caution: This is a *temporary* backup. If you use this option, you should uninstall the application and then test Windows and other Windows applications to see that they run correctly. If you must use the restore option, UnInstaller will restore everything back exactly as it was before the uninstallation. This includes directories, primary files, and INI entries. If you change drive mappings/assignments or move the UnInstaller .EXE file, the restore will not work correctly.

UnInstaller does not backup the "associated" files that are deleted in conjunction with the uninstall process. If you may need those files in the future, you should use a separate backup utility to back them up before uninstalling .

The backup feature does not save "registration database" information as used with OLE (Object Linking and Embedding). The only reason you might wish to restore the registration database is if you truly want to re-install the application. In that case, you should re-install the application from the original disks.

8. If you want UnInstaller to create a report, select the **Create Report** check box. See ***Report Setup Primer*** for details.

Step 3: The uninstall report

See **Report Setup Primer** for details.

Step 4: Confirming deletions

1. Choose **Uninstall** to begin removing items.

Note: If you choose to backup the application, the backup process is performed before any deletions take place. The completion percentage shown during the backup process is based on the number of items selected for deletion in the list box it is not based on the number of files to be backed up.

Note: UnInstaller never deletes a file that is currently in use by Windows.

If you did not clear the **Confirm all deletions** check box, UnInstaller prompts you before deleting each item you've selected for deletion. Even if you did clear the check mark, you will still be prompted to confirm deletion of any important items.

The following are a few notes to keep in mind during the uninstall process:

- UnInstaller may have to open Program Manager, particularly if you use an alternative desktop such as Central Point PCTools Desktop™. Dont worry about this. If UnInstaller runs Program Manager, it will be hidden initially. You can show Program Manager by choosing the **Show Program Manager** item in the **Options** menu. UnInstaller will shut Program Manager down before exiting.

- UnInstaller may tell you that errors occurred while trying to delete icons from Program Manager. If you see this error message, please refer to *Q&A on UnInstaller* for more details.

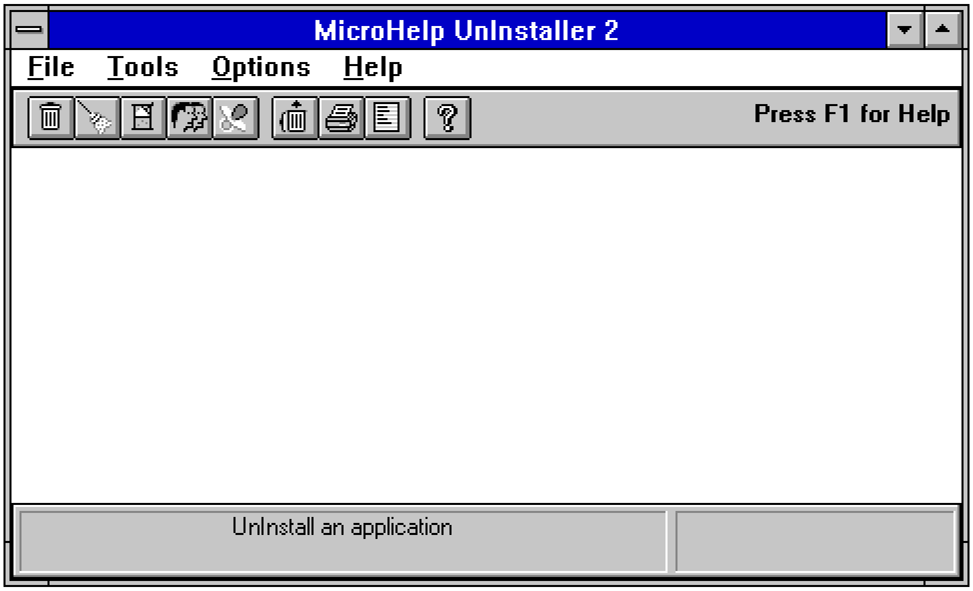
- In confirmation boxes and windows: **Yes** allows a deletion; **No** continues the uninstall without deleting an item; **Cancel** stops the entire uninstallation process.

- Some confirmation windows display detailed information about an item. In these windows, the **View...** (or **View File...**) button allows you to view an items or files contents, if possible. The **Yes to all** button allows you to cancel most of the subsequent confirmation messages during the uninstall process and is equivalent to clearing the **Confirm all deletions** check box on the **Uninstall Applications** window. Note that even if you choose **Yes to all**, some items still prompt for confirmation.

- If a selected item would cause the deletion of more than one file, the next step depends on the state of the **Confirm all deletions** check box. If its checked, all of the files are displayed in a list box, from which you choose the ones to be deleted. You may select all, some, or none of the files for deletion. If unchecked, you are prompted with a window that asks for confirmation that you want to delete the directory and all the files. Choose **Yes** to delete the directory and all the files, **No** to load a list of the files into a list box (the same as when **Confirm all deletions** is checked) or **Cancel** to cancel the uninstall process.

- UnInstaller tells you when it has finished.

• Main Window Overview



The UnInstaller main window. Press the UnInstall button and then choose a group.

When you launch UnInstaller, you'll see the main window shown above. At this point you have several choices, each of which can be accessed from the menu bar. All main options under the **Tools** menu can be accessed via command buttons.

Using the Main Menu

The following sections describe the commands available under each menu selection.

File Tools Options Help

See Also:

Using The Button Bar About The Status Bar

File Menu Commands

The following table describes the commands found on the File menu.

Use	To
Create/Update SmartLinks	See *SmartLinks*
Restore Backup...	Restore an uninstalled application
View Report File...	Views report file created during uninstall
Print Setup...	Display the **Print Setup** dialog
Report Setup...	Display the **Report Setup** dialog
Exit	Close UnInstaller

Tools Menu Commands

The following table describes the commands found on the Tools menu.

Use	To
UnInstall Applications...	Start the **UnInstall** feature
System Cleanup...	Start the **System Cleanup** feature
INIClean...	Start the **INIClean** feature
Duplicate File Finder...	Start the **Duplicate File Finder** feature
Orphan Finder...	Start the **Orphan Finder** feature

Options Menu Commands

The following table describes the commands found on the Options menu.

Use	Check to
Button Bar	See Button Bar on screen
Status Bar	See Status Bar on screen
Tool Descriptions	See summaries when tools are used
3-D Borders	See 3-D effects on borders
3-D Text	See 3-D text
3-D Background	See 3-D backgrounds
Sound On	Turn sound on

Help Menu Commands

The following table describes the commands found on the Help menu.

Use	To
Contents	View contents of UnInstaller Help
Search For Help On...	Enter a search topic for UnInstaller Help
Technical Support...	View information about technical support
How to Use Help	View information about Windows help in general
About...	View information about UnInstaller

Note: When you select **About** from the **Help** menu, UnInstaller displays the amount of disk space freed since the time you installed UnInstaller.

Using the Button Bar

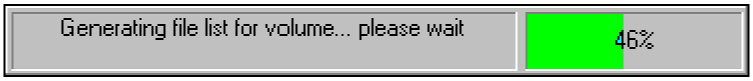
The buttons below the menu on the Main Window are a quick way to select the following features:

Use	To
● UnInstall Applications...	Start the **UnInstall** feature
● System Cleanup...	Start the **System Cleanup** feature
● INIClean...	Start the **INIClean** feature
● Duplicate File Finder...	Start the **Duplicate File Finder** feature
● Orphan Finder...	Start the **Orphan Finder** feature
● Restore Backup...	**Restore** an uninstalled application
● Print Setup...	Display the **Printer Setup** dialog
▤ Report Setup...	Display the **Report Setup** dialog
● Help	Display **Help** about the task at hand

About the Status Bar

The status bar at the bottom of the main UnInstaller window displays messages that tell you which function is currently being performed.

On the right side of the status bar is a section that is used by some functions to let you know the progress of a procedure.   The completion indicator is shown using a colored gauge display.



Generating file list for volume... please wait

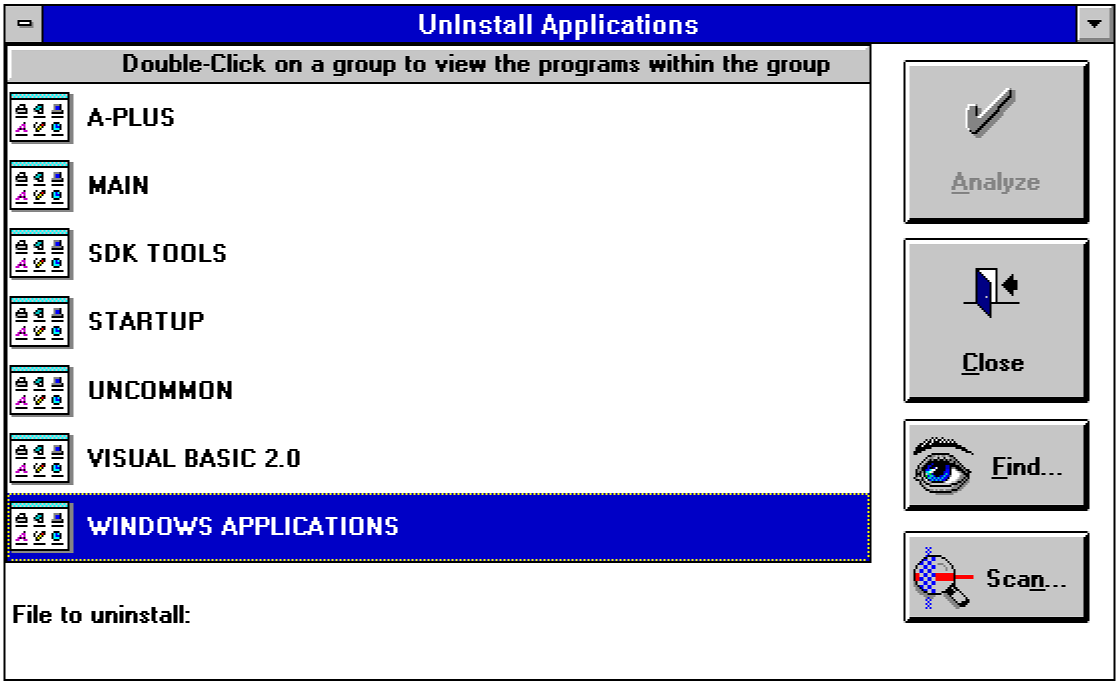
46%

The Status Bar

• Uninstalling an Application

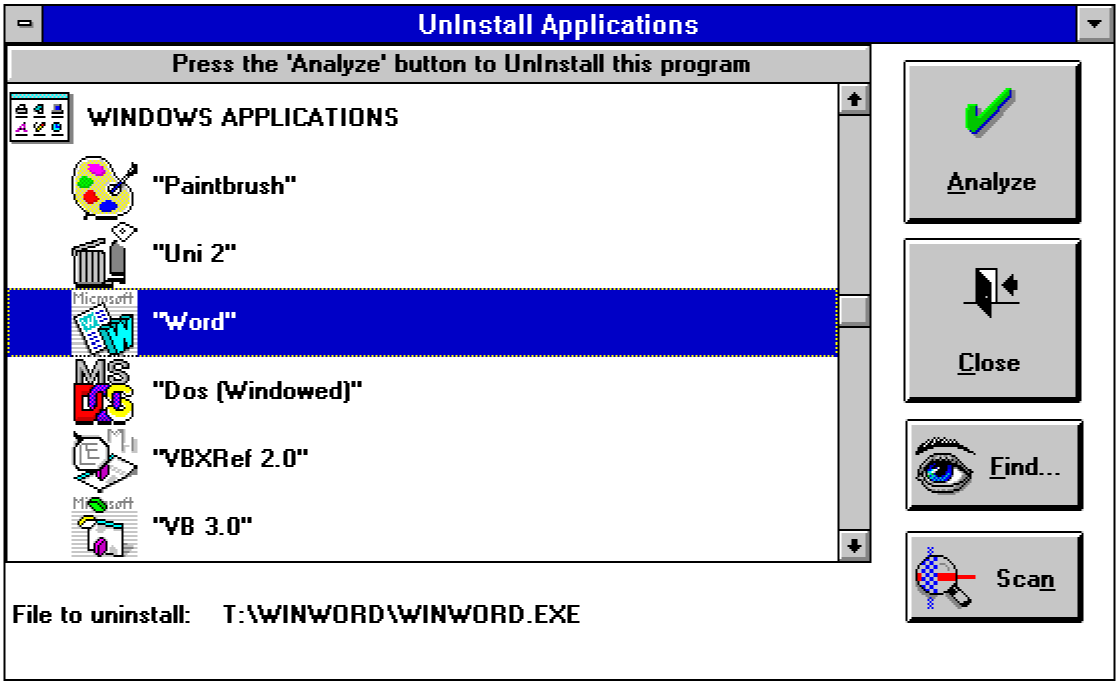
Getting down to the nitty-gritty

You start the uninstall process by choosing **Tools-UnInstall Applications...** from the menu or by clicking the button shown above.



Double-Click on a group name to view the programs in that group

The main window, titled **Double-Click on a group to view the programs within the group,** automatically lists all the applications that have been logged into the shell program (Program Manager by default). The applications are shown by groups. Double-click on the group in which the application to be uninstalled is located. The list will expand to show the applications within that group, as shown below. If you double-click on a group that does not have any application icons in it, UnInstaller will advise you accordingly and offer to delete the group.



File to uninstall: T:\WINWORD\WINWORD.EXE

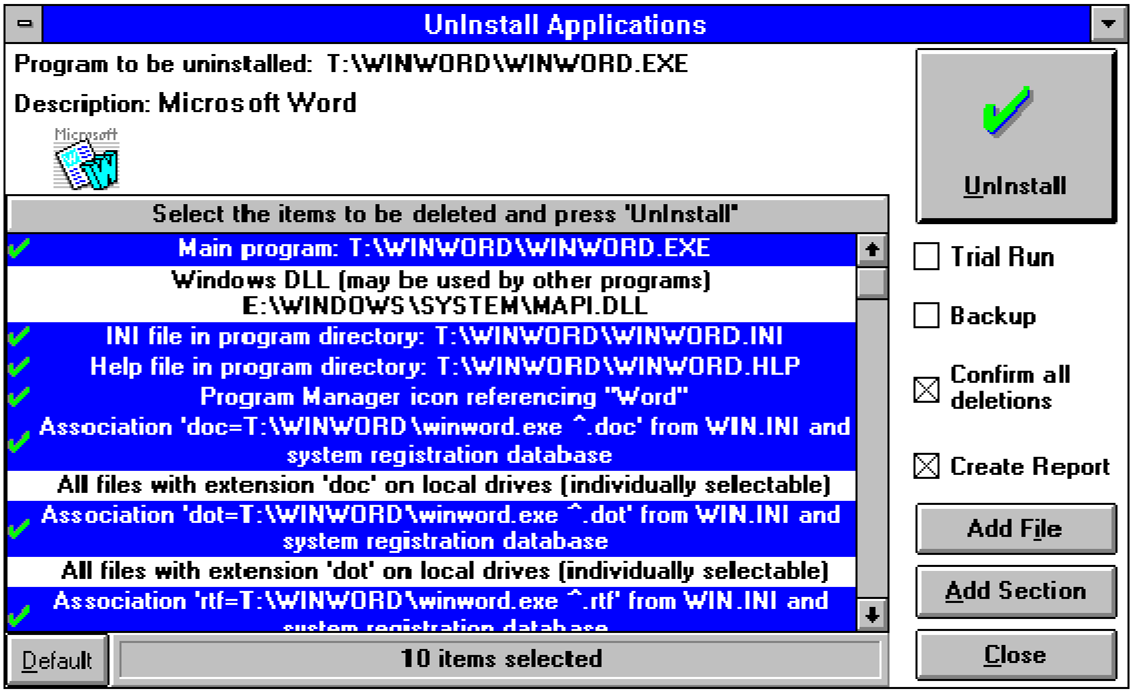
The UnInstall Applications dialog.   After choosing an application, press the Analyze button.

If the application you want to uninstall is not listed, you can manually select it by pressing the **Find...** button. Or, you can use the **Scan** button to scan a drive for all Windows applications on that drive.

After you have selected an application to UnInstall, press the **Analyze** button to tell UnInstaller to examine the application.

Note: Pressing the **Analyze** button does not cause the application to be UnInstalled.

After UnInstaller has analyzed the selected application, you are presented with a list of items that UnInstaller suggests for removal from your system.   In addition, the selected applications icon appears at the top of the window, along with the applications file name and description.



Program to be uninstalled: T:\WINWORD\WINWORD.EXE

Description: Microsoft Word

Select the items to be deleted and press 'UnInstall'

- ✓ Main program: T:\WINWORD\WINWORD.EXE
- Windows DLL (may be used by other programs) E:\WINDOWS\SYSTEM\MAPI.DLL
- ✓ INI file in program directory: T:\WINWORD\WINWORD.INI
- ✓ Help file in program directory: T:\WINWORD\WINWORD.HLP
- ✓ Program Manager icon referencing "Word"
- ✓ Association 'doc=T:\WINWORD\winword.exe ^.doc' from WIN.INI and system registration database
- All files with extension 'doc' on local drives (individually selectable)
- ✓ Association 'dot=T:\WINWORD\winword.exe ^.dot' from WIN.INI and system registration database
- All files with extension 'dot' on local drives (individually selectable)
- ✓ Association 'rtf=T:\WINWORD\winword.exe ^.rtf' from WIN.INI and system registration database

10 items selected

□ Trial Run

□ Backup

☒ Confirm all deletions

☒ Create Report

Select (highlight) the items you want to delete and then press the UnInstall button

Each line in the list box represents an item or items that UnInstaller suggests for removal. Some of the listed items are files, while others may refer to changes to your WIN.INI file, icons or directories.

Most of the lines will probably be selected automatically. This means that UnInstaller will remove the item(s) when you press the **UnInstall** button. As a safety feature, one or more of the suggested items must be manually selected. See *Smart Uninstalling* for more details about how to make the right choices.

• UnInstaller Controls
Buttons and boxes

The controls on the UnInstaller Window let you supervise the uninstall process.

The **Default** button selects the items UnInstaller originally suggested for deletion and deselects the items not originally suggested. Use this button if you make mistakes in choosing items for deletion and you want UnInstaller to redisplay its automatically-selected items.

The **UnInstall** button starts the actual uninstall process. See *Starting the UnInstall* for details.

When checked, the **Trial Run** check box tells UnInstaller to go through the motions of uninstalling the selected application, but without deleting any files and without changing or removing any sections in WIN.INI. Going through a trial run tells you how many files would have been deleted and how much disk space would have been saved if the files were actually deleted. If the **Create Report** check box is checked, a report is generated so that you can determine what would have been deleted.

When checked, the **Backup** check box tells UnInstaller that you want a backup copy of the application you are uninstalling. If you tell UnInstaller to create a backup copy, you can use the ***Restore Backup*** feature at a later time to restore the application.

Note: If you choose to backup the application, the backup process is performed before any deletions take place. The completion percentage shown during the backup process is based on the number of items selected for deletion in the list box it is not based on the number of files to be backed up.

When checked, the **Confirm all deletions** check box tells UnInstaller that you want to be prompted to confirm deletion of each and every item you've selected in the list box.

When checked, the **Create Report** check box tells UnInstaller that you want a report created each time you uninstall an application. (See *Report Setup Primer* for instructions on creating printed reports and a report file.)

The **Add File...** and **Add Section...** buttons allow advanced users to mark additional files and/or WIN.INI sections that may not be listed by UnInstaller.

• UnInstaller Reports

Keeping tabs on UnInstaller

UnInstaller can create a report of everything it does. To have a report automatically created, simply check the **Create Report** box. The report shows which application you chose to remove and which files and WIN.INI associations were deleted. It also lists all the items you chose not to delete. Whenever you're in doubt about deleting items, we recommend that you tell UnInstaller to create a report by selecting the **Create Report** check box. Then, if there's a problem, you can show the report to an expert and request advice.

You can also create a "what if" report by checking the **Trial Run** check box. That causes UnInstaller to determine which files would be deleted and how much disk space would be saved, but without actually deleting anything.

• Starting the UnInstall

Ready, set, go...

Once you have selected items to be deleted, press the **UnInstall** button to begin the uninstallation process. The act of pressing this button *does not* automatically delete anything. Instead, UnInstaller asks for confirmation as it works down the list of items to remove. If you don't want to have to confirm all the deletions, most confirmations can be avoided by clearing the **Confirm all deletions** check box before pressing the **UnInstall** button.

If the removal of a single item would result in the deletion of more than one file, UnInstaller lists the files in a separate window. You may pick and choose exactly which files you want removed.

If you would like to make a backup of the primary files removed and INI file changes made, check the **Backup** box. (Checking the **Backup** box means that you will be able to use the ***Restore Backup*** feature to restore the application.) This is advisable if you are in any way unsure of the purpose of any of the items suggested for removal by UnInstaller.

Note: If you choose to backup the application, the backup process is performed before any deletions take place. The completion percentage shown during the backup process is based on the number of items selected for deletion in the list box it is not based on the number of files to be backed up.

UnInstaller tells you when it has finished and sends its report, if requested, to the printer, a disk file or both as specified in the **Report Setup** dialog. The **Report Setup** dialog is accessible from the **File** menu. See *Report Setup Primer* for details.

Uninstalling the Uninstalled

Quick cleanup of previously-removed applications

1. Choose **Tools-UnInstall Applications** from the main menu or press the button shown above.
2. Press the **Find...** button.
3. Using the **Select the program to be uninstalled...** dialog, type in the filename, including the directory name (if you know it) of the program that has been removed and press the **OK** button. For example, enter EXCEL.EXE. You will get a warning from UnInstaller that the file you've selected does not exist. Don't worry about that. Simply answer 'yes' to allow UnInstaller to continue with your selection.
4. Press the **Analyze** button.
5. Select the items to be deleted from your system in the **Select the items to be deleted and press 'UnInstall'** list box.
6. Press the **UnInstall** button.

• INIClean vs. UnInstaller
Deciding which is best for the job

UnInstallers **INIClean** feature is designed to help clean up your system if you have previously removed an application without the benefit of UnInstaller. However, it's sometimes better to use UnInstaller as a cleaner, even if the application has already been deleted.

In general, use **INIClean** if you think an application has made only minor changes to Windows, such as adding a section to WIN.INI.

Use UnInstaller to clean up after applications that make many changes to your system, such as creating INI files in several locations, creating documents or installing DLL files.

◆ Smart UnInstalling
Tips, tricks & techniques

UnInstaller usually knows what should be removed from your computer to properly uninstall an application.   However, UnInstaller is not perfect.   For example, an application with a very short name can be confused with another application having a similar, but longer name.   If an application's name is four characters or shorter, a message appears warning you to look closely at the suggested deletions.   This is necessary because UnInstaller recommends items for deletion by searching for the filename of the application that you are trying to delete.   When you choose an application with a short file name, it's possible that the file name may be found elsewhere, but as a portion of a longer file name. For example, if you are a dentist and you have an application called "Pain," UnInstaller may find occurrences of "Paintbrush" that appear to be matches for "Pain."

In addition, more than one application may depend on a WIN.INI section that UnInstaller suggests for deletion.   In either case, letting UnInstaller delete something it shouldn't could bring your Windows system or Windows applications to a halt.

Dont panic!

UnInstaller never automatically suggests an essential Windows component for deletion.   Furthermore, it never automatically suggests for deletion an item that may be used by other applications.   Only items that UnInstaller thinks are used *only* by the application being uninstalled are automatically suggested for removal.

When UnInstaller has doubts about an item, its up to you to decide whether it should be deleted.   If you keep the following rule in mind, you wont go wrong uninstalling an application:

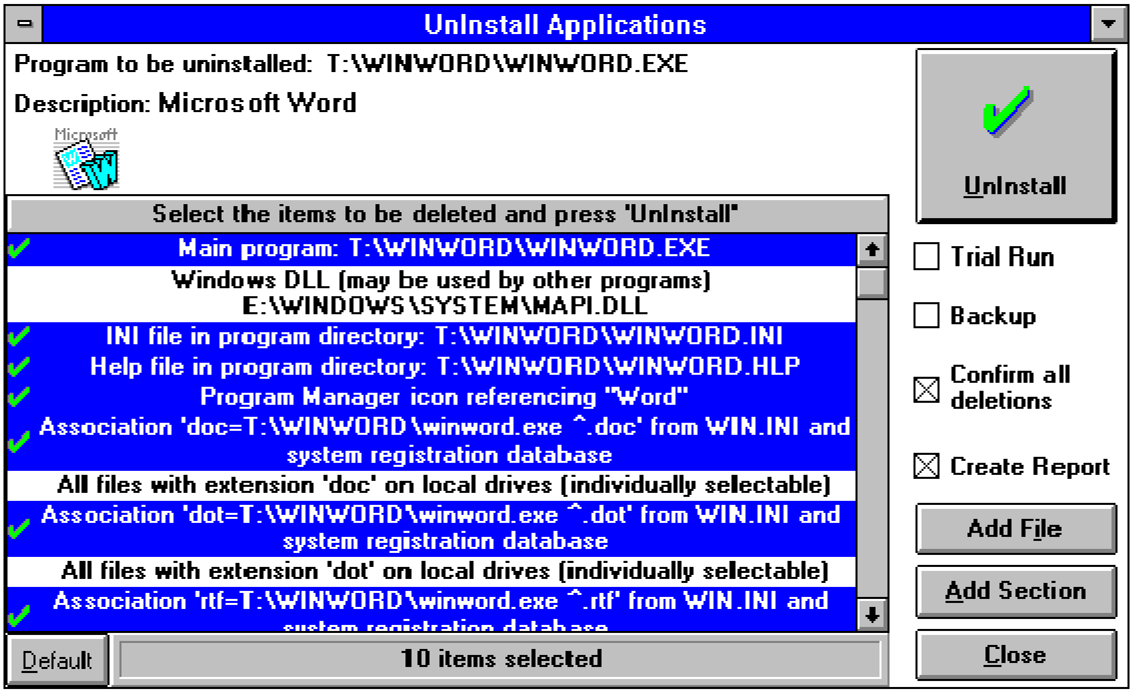
Dont delete anything unless you know that its safe to do so.

How do you know what should be removed? The following sample uninstallation gives an overview of what exactly UnInstaller removes or suggests for removal. A more detailed and lengthy explanation of uninstallable items is given in *UnInstallable Items*.

Another aid is the UnInstaller report, which shows items that were removed as well as items that were *suggested* for removal but not deleted.   This lets you seek advice from an advanced user, a Windows expert or the technical support line of the company that publishes the application you are uninstalling.   Note that you can check the **Trial Run** check box to get a report of what will be deleted, but without actually deleting anything.

Portrait of an uninstall

UnInstaller lists everything suggested for removal in the **Select the items to be deleted and press 'UnInstall'** list box of its main window. These items include an applications main executable file (EXE file), help file (HLP) and data files. Items also include associations listed in WIN.INI, sections of WIN.INI and private .INI files used by the application. Heres an example of what would be suggested for deletion when uninstalling Microsoft Word for Windows:



Program to be uninstalled: T:\WINWORD\WINWORD.EXE

Description: Microsoft Word

Select the items to be deleted and press 'UnInstall'

- ✓ Main program: T:\WINWORD\WINWORD.EXE
- Windows DLL [may be used by other programs] E:\WINDOWS\SYSTEM\MAPI.DLL
- ✓ INI file in program directory: T:\WINWORD\WINWORD.INI
- ✓ Help file in program directory: T:\WINWORD\WINWORD.HLP
- ✓ Program Manager icon referencing "Word"
- ✓ Association 'doc=T:\WINWORD\winword.exe ^.doc' from WIN.INI and system registration database
- All files with extension 'doc' on local drives (individually selectable)
- ✓ Association 'dot=T:\WINWORD\winword.exe ^.dot' from WIN.INI and system registration database
- All files with extension 'dot' on local drives (individually selectable)
- ✓ Association 'rtf=T:\WINWORD\winword.exe ^.rtf' from WIN.INI and system registration database

10 items selected

☐ Trial Run

☐ Backup

☒ Confirm all deletions

☒ Create Report

*This sample **Select the items to be deleted** list box from the UnInstall Applications window shows a typical list of what would be shown when uninstalling Microsoft Word for Windows.*

Main program: T:\WINWORD\WINWORD.EXE
At the heart of every Windows application is its *main program* file. This file contains the code that makes an application work. Deleting WINWORD.EXE removes the core of Microsoft Word from your computer.

Windows DLL [may be used by other programs]: E:\WINDOWS\SYSTEM\MAPI.DLL
DLL (Dynamic Link Libraries) files are used by applications as part of the executable program. The catch here is that DLL files can be (and often are) shared by multiple programs. UnInstaller tells you that a DLL *may be used by other programs* because it has scanned the WINWORD.EXE file and determined it *may* use this DLL file. These items are not automatically selected.

INI file in program directory: T:\WINWORD\WINWORD.INI
INI files are described under INI Files Primer. INI files are typically used by applications to store information pertaining to the application between uses. They should be removed when the application is uninstalled. If WINWORD.EXE gets deleted, then WINWORD.INI should be deleted at the same time.

Help file in program directory: T:\WINWORD\WINWORD.HLP
Most Windows applications include a help file. (Help files usually have an

extension of HLP.) Help files should be deleted when the corresponding application is deleted. If WINWORD.EXE gets deleted, then WINWORD.HLP should get deleted too.

Program Manager icon referencing "Word"
UnInstaller can remove the icon that represents the application from Program Manager. If WINWORD.EXE is removed from your computer, then Program Manager no longer needs the icon and the icon-related information referring to the application.

Association doc=T:\WINWORD\winword.exe ^.doc from WIN.INI and system registration database
Association dot=T:\WINWORD\winword.exe ^.dot from WIN.INI and system registration database
Association rtf=T:\WINWORD\winword.exe ^.rtf from WIN.INI and system registration database
If an application is uninstalled, its associations in the [Extensions] section of the WIN.INI file and system registration database should also be removed.

All files with extension doc on local drives (individually selectable)
All files with extension dot on local drives (individually selectable)
All files with extension rtf on local drives (individually selectable)
UnInstaller can scan your hard disk and offer to delete all data files created by an application. UnInstaller lists the data files created by the application being uninstalled and lets you select which ones should be deleted.

Section [MICROSOFT WORD 2.0] of WIN.INI
Section [MS PROOFING TOOLS] of WIN.INI
Section [MSWORD TEXT CONVERTERS] of WIN.INI
Some applications add entire sections of instructions to WIN.INI and the entire section should be removed when the application is uninstalled. As a safety measure, UnInstaller never automatically suggests a WIN.INI section for removal. One reason for this is that several applications may use information contained in the same section. For example, the [ALDUS] section of a WIN.INI file might be used by both *Aldus PageMaker* and *Aldus Freehand*. If you remove the ALDUS section when uninstalling PageMaker, Freehand may not run. To avoid problems like this, you should select for deletion only those sections you know can be safely removed.

Note that in the above example, [MICROSOFT WORD 2.0] and [MSWORD TEXT CONVERTERS] clearly refer only to Microsoft Word and can be removed. [MS PROOFING TOOLS], however, could refer to any Microsoft product, so it should not be removed.

Checking an application's instruction manual (also called User's Guide) or calling a company's technical support line can help you determine if a section should be removed.

When in doubt, leave questionable sections on your computer. You can use the UnInstaller report to record the sections name, and then later try to find out more about it. If it turns out that the section can be removed, you can delete it using **INIClean**. Or, if you prefer, you can select the section for deletion, and when you see the confirmation dialog box, use the **View** button to view the contents of the section before deciding whether or not to delete the section.

OLE object type Word Document
This line appears if an application is listed as an *OLE server* in WIN.INI. This

item should be deleted only if the OLE server program is being uninstalled. Otherwise, other applications may still think they can embed Word document objects into their documents!

Remove T:\WINWORD (and subdirectories) if empty when done
Delete all files in T:\WINWORD and its subdirectories (individually selectable)
After UnInstaller has removed the traces left by an application, it lists all the files remaining in the applications directory and subdirectories and gives you the option of deleting them. In most cases, youll want to delete all the files, since the files are useless without the application. If you delete all files in a directory, UnInstaller will offer to remove the directories for you.

Select this line to create a SmartDecoy
Network administrators should select this option if the application they are uninstalling is shared by multiple users on the network. This line item is always listed last and is never automatically selected. Please refer to *SmartDecoy Primer* for more information on SmartDecoys. This option is only available in the network version of UnInstaller.

Q&A On UnInstalling

When should I use UnInstaller?
Use UnInstaller to remove applications from your computer. Say, for example, you change the brand of spreadsheet you use and begin using *Microsoft Excel* rather than *Lotus 1-2-3 for Windows*. Use UnInstaller to remove Lotus 1-2-3 and all files, associations and other traces that belong to Lotus 1-2-3.

How do I find the applications to uninstall?
When the UnInstaller main window appears, double-click on the group icon in which the application can be found. A list box expands, listing applications that you can remove. If the application that you want to uninstall is not listed in this window, check other groups or choose the **Find...** button.

The application that I want to remove isnt listed. Why not? How can I find it?
UnInstaller lists every application that appears in Program Manager. But sometimes, Program Manager doesnt know that an application is on your system.

For example, sometimes users delete an applications icon and Program Manager thinks the application is gone. In reality, it remains on your system. Another problem is if you use a desktop alternative such as PCTools for Windows (PCTW). It may be that you installed an application into PCTW, but Program Manager knows nothing about it.

You can help UnInstaller around these problems by manually finding the application's executable file (usually having an extension of EXE). Choose either the **Find...** or **Scan...** button in the **Double-Click on a group...** window to search your computer for the application you want to remove. Once you have found it, click **OK** and then click the **Analyze** button to begin uninstalling.

Can I use drag-and-drop to uninstall an application?
Yes, if you use a desktop alternative (such as Norton Desktop for Windows) which supports drag-and-drop program launching. Drag and drop UnInstallers icon to the desktop to create an UnInstaller desktop button. Then drag an application icon and drop it onto the UnInstaller desktop button. This will launch UnInstaller, which will automatically analyze the application.

UnInstaller told me that errors occurred while trying to delete icons from Program Manager. Is everything OK?
Program Manager can be difficult to work with, especially when UnInstaller is trying to delete one of its icons. As a result, you may get this error message even though everything worked perfectly OK. To be certain, check Program Manager to see if the applications icon was deleted. If not, delete it manually by selecting it, then pressing the

<Del> key on your keyboard (or choosing **Delete** from Program Managers **File** menu).

Another way to handle this situation is to let the uninstall procedure complete, then choose the **Uninstall** button again, immediately. Sometimes Program Manager will delete icons the second time around!

What exactly is UnInstaller removing from my computer? What are these items for deletion that I keep reading about?
The *Smart Uninstalling* section in this chapter goes into detail about what is being taken off your computer.

What items are not automatically selected for deletion by UnInstaller?
UnInstaller is very smart at deciding which items can safely be removed from your system. Items that UnInstaller is confident about are automatically selected for deletion during the analyze procedure. However, UnInstaller is cautious and does not select two common items for deletion:

Data files, such as the letters youve written in a word processor or spreadsheets you've created, are never automatically selected for deletion. UnInstaller merely reminds you that they are there and offers to delete them.

Entire sections of WIN.INI are never automatically chosen for removal. This is because some sections might be used by other applications, and it's difficult to tell which sections are used by which programs.

"All other files" in the same directory is not automatically selected if the directory name isn't similar to the application's name. In fact, if the application is installed in the \WINDOWS or \WINDOWS\SYSTEM directory, this item doesn't appear in the list box at all.

I have more than one copy of Windows on my system though I only have one in use. Will UnInstaller be able to locate my applications properly?
UnInstaller will not have difficulty locating the application you are looking for as long as it is in the currently running version of Windows. The other copies of Windows will not reflect the removal of an application. Please note that UnInstaller only prevents you from deleting a "protected" Windows file from the currently loaded copy of Windows. All other file copies are considered expendable. Please check the list of files to be deleted carefully if you have multiple copies of Windows on your machine.

What will UnInstaller fail to find?
There is a great range of applications available that modify your system in many ways. Here is a partial list of items that UnInstaller will *not* find when uninstalling an application:

DLLs which hook standard drivers, such as remote control programs for Windows which replace your standard video, keyboard and mouse drivers
Some entries to your SYSTEM.INI file

I want to delete some data files with UnInstaller, but not all of them. Can I be selective?
Yes, you can. UnInstaller allows you to remove none, some, or all data files that end in a file extension associated with an application. If you select the All files with extension EXT on local drives line item, UnInstaller scans all of your local hard disks (i.e., all non-network hard disks) to find all files with the given extension. These files are listed in a window and you can select individual files to be deleted. You can view the contents of a file before deleting it by selecting the filename and then choosing the **View** button in the confirmation window.

Once you have selected the documents to delete and you press the **Delete** button, UnInstaller prompts you once more for confirmation. Answering **Yes** deletes all the selected files at once.

Cancel stops the entire uninstallation process. Choose **Close** to close the window without deleting any of the files.

Why does UnInstaller sometimes open Program Manger?
Some desktop alternates, such as PCTools for Windows, do not update Windows Program Manager files. If you tell UnInstaller to uninstall an application, UnInstaller may run Program Manager so that it can update files and delete icons.

I use a shell program, such as PCTools for Windows (PCTW) or Dashboard, and tried to start up Program Manager but it wouldn't load. Is there some way I can start it or is something wrong?
If you have a shell replacement, with the exception of Norton Desktop for Windows (NDW), UnInstaller loads Program Manager in the background and hides it. When this occurs, an **Options** Menu selection, **Show Program Manager** is created. Selecting this option will display Program Manager as a minimized application. Please note that if you are running NDW, you should not load Program Manager while running UnInstaller, because the information regarding any system changes made during that uninstall session will not be passed along to NDW.

If UnInstaller successfully loaded Program Manager, it will automatically close Program Manager when the uninstall session is completed. If you shut down Program Manager manually before ending your uninstall session, it will usually be reloaded into the background again. However, if you use PCTools for Windows as your shell, you may get a warning of a "failed DDE request." The best advice is to leave Program Manager loaded until you have finished uninstalling and closed the UnInstaller application.

I use a shell program such as Norton Desktop for Windows (NDW) or PCTools for Windows (PCTW). Will I have problems using UnInstaller?

UnInstaller is impacted by shell programs in three ways:

(Applies to PCTW and sub groups) When selecting an application to be uninstalled, UnInstaller searches only your Program Manager groups for installed applications. If an application was installed only into the shell program but not Program Manager, then UnInstaller will not find or list it. In this case, choose the **Find...** button to manually locate the application by filename.
An applications icon may not be removed from its shell program group window. This happens if the icon for the application that you are uninstalling is in a different group under the shell than it is under Program Manager. If this happens, then you can manually delete the icon from the shell program: simply switch to your shell program (NDW, PCTW, etc.), open the appropriate group window, select the icon to be deleted, and follow directions for deleting icons provided in your shell program manual. (The <Del> key is used to tell most shell programs to delete an icon.)
Groups within groups (or subgroups) are *not* supported. The sub group will appear in the list box, but double-clicking on it will give a DDE error.

Note: If you are using Norton Desktop for Windows (NDW) and have the desktop minimized, the desktop will automatically normalize whenever UnInstaller queries the shell for group or program information. We have no control over this as NDW does this on each DDE request that we make.

Can I use UnInstaller to remove a shell program such as Norton Desktop for Windows (NDW) or PCTools for Windows (PCTW)?

UnInstaller can remove a third party shell, even if it is the current shell! Here's how it works:

Select the shell program's icon in UnInstaller. If the shell selected for removal is the current shell, UnInstaller will modify the "shell=" line in SYSTEM.INI so that Program Manager is the shell and then restart Windows. (Note: If Program Manager cannot be found, an error message will be displayed and the selection will be canceled.)
Select the old shell program for removal as you would any other application.

How can I "uninstall" UnInstaller?

Be sure that UnInstaller is not running.

Use File Manager to delete all of the files in the directory where you installed UnInstaller.

If you are running Windows from your local hard disk, use File Manager to delete VBRUN300.DLL from your \WINDOWS\SYSTEM directory.

If you are running Windows from a network drive, use File Manager to delete VBRUN300.DLL from your \WINDOWS directory.

Delete the icons from your shell program.

UnInstaller makes no changes to INI files so there is no need for a complicated uninstall procedure for removing UnInstaller.

I just installed UnInstaller 2 and I didn't let the installation program remove or upgrade UnInstaller 1. How can I remove UnInstaller 1?

The easiest way is to use UnInstaller 2 to remove version 1. However, if you wish to do that, be sure to install version 2 to a different directory than version 1.

For those who prefer to delete files themselves, please refer to pages 99 and 100 in your UnInstaller version 1 manual for the lists of files that were installed by UnInstaller 1. The directory and all the files listed on page 99 can be removed. On page 100, the only files listed that need to be removed are: MHEN200.VBX and MHUTIL.VBX. These should not be removed if you have any MicroHelp programming tools.

Warning: Do *not* remove the other three files (.DLLs) listed! (These should have been on your machine before UnInstaller was ever installed as they are required by Windows.) Simply remove the two files noted above (.VBXs) from your machine using File Manager or some other file management program.

Finally, delete the old icons from Windows. UnInstaller 1 made no changes to INI files so there is no need for a complicated uninstall procedure for removing UnInstaller.

Who are the experts who can help?

If you need an experts help, make a note of the information (file names, error messages, etc.) that you are confused about. Then you can seek out advice in several places.

Online information services, such as CompuServe and Prodigy, offer a widespread forum for seeking advice. Many computer magazines offer help to users with questions about Windows and Windows applications. Product support lines maintained by software publishers can tell you what associations, WIN.INI sections and INI files their applications create.

MicroHelp will be happy to assist you with technical problems directly relating to the functionality of UnInstaller, but MicroHelp cannot offer guidance in determining what should be removed from your computer. There are simply too many applications out there to keep track of.

• Using System Cleanup

Note: Everything that **System Cleanup** can delete can be easily restored by running the Windows Setup program, *except* for third party (additional) font and video drivers. If you do *not* have your Windows installation disks, you might not want to use the **System Cleanup** feature. It is possible that if you delete the wrong file(s), some of your Windows applications or even Windows itself, will not run.

Note: We recommend that you execute a trial run and print a report of your cleanup selections before you tell UnInstaller to actually remove any files. Seek expert advice and review the report carefully before actually performing the cleanup. Prior to performing the cleanup, it is advisable to make a backup of the specific files to be removed as listed in the report.

Note: No files are deleted if the **Trial Run** check box is checked. Performing a trial run allows you to determine how many files will be deleted and how much disk space will be freed without actually removing the files.

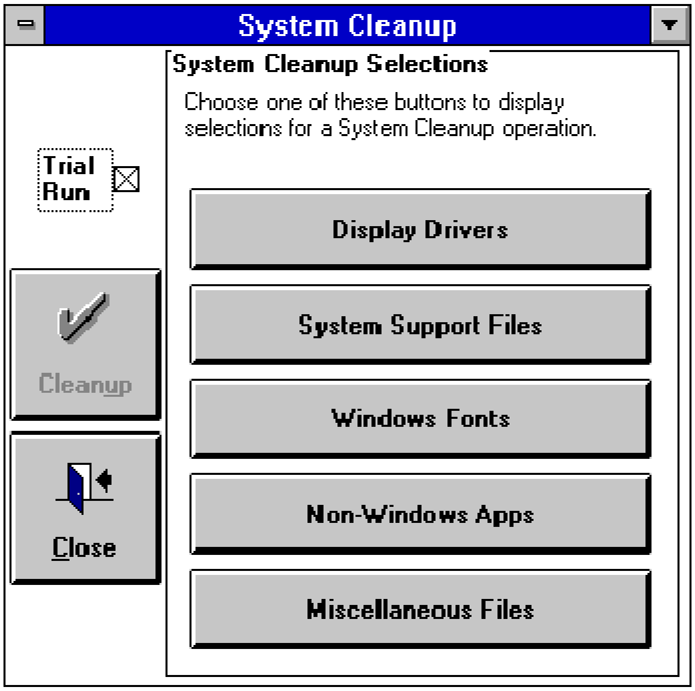
● Quick System Cleanup

To Cleanup Your System

1. Choose **Tools-System Cleanup** from the main menu or press the button shown above.
2. Select the type of items you want to remove by pressing one of the following buttons: **Display Drivers**, **System Support Files**, **Windows Fonts**, **Non-Windows Apps** or **Miscellaneous Files**. (Quick instructions for each type of item can be found beginning with *Removing Display Drivers*.)
3. Press the **Cleanup** button to perform the actual cleanup for the type of items you have selected.
4. Repeat steps 2 and 3 for all types of items.

• System Cleanup Overview

The **System Cleanup** feature helps you clean display drivers, system support files, Windows fonts, and non-Windows application support files from your system. When you run UnInstaller's **System Cleanup** feature, the following window will appear:



Choose one of these buttons to display selections for a System Cleanup operation.

The System Cleanup main window

Note: No files are deleted if the **Trial Run** check box is checked. This allows you to determine how many files will be deleted and how much disk space is reclaimed without actually removing the files.

Caution: If you are not 100% sure about whether a file should be removed, don't remove it!

Listed below are descriptions of the controls in the **System Cleanup** window:

Trial Run check box

When this check box is checked, UnInstaller will not delete any files or make any changes to files. Use this option until you are sure about what you want to do. Once you are sure, clear the check box and then press the **Cleanup** button to actually remove the files.

Cleanup button

Carries out the actions you have specified.

Display Drivers button

Allows selection and cleanup of OEM display drivers. For more information, please refer to *Drivers Primer* and *Removing Display Drivers*.

System Support Files button
Suggests support files that may be deleted. For instance, standard mode files that Windows does not use when running in enhanced mode. For more information, please refer to *Removing System Support Files*.

Windows Fonts button
Displays the names of, and allows deletion of, both installed and available fonts. For more information, please refer to *Fonts Primer* and *Removing Fonts*.

Non-Windows Apps button
These are ***support files*** required to run DOS programs under Windows 386 enhanced mode. If you never run DOS programs in Windows, you can delete these files. For more information, please refer to *Removing Non-Windows Apps*.

Miscellaneous Files button
Presents a list of various files that you may delete if they are not being used. UnInstaller color codes the files that it suggests for removal. File names that are grayed out are not present on your system. Files displayed in green can be safely removed. If a file's name is displayed in yellow, use caution before telling UnInstaller to delete the file. When you see a file displayed in red, you should delete the file only if you are absolutely sure you don't need it. The files named RAMDRIVE.SYS and EMM386.EXE are shown in red if they are in use. For more information, please refer to *Removing Miscellaneous Files*.

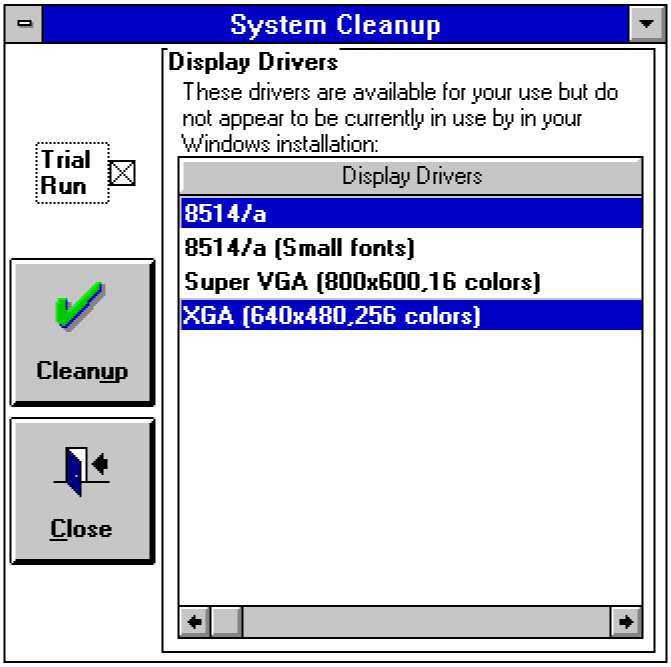
Note: Everything **System Cleanup** can delete can be easily restored by running the Windows Setup program, *except* for third party (additional) font and video drivers. If you do *not* have your Windows installation disks, you might not want to use the **System Cleanup** feature. It is possible that if you delete the wrong file(s), some of your Windows applications or even Windows itself, will not run.

Note: We recommend that you execute a trial run and print a report of your cleanup selections before you tell UnInstaller to actually remove the files. Seek expert advice and review the report carefully before actually performing the cleanup. Prior to performing the cleanup, it is advisable to make a backup of the specific files to be removed as listed in the report.

• Removing Display Drivers

Warning: Deletions are immediate and cannot be undone by UnInstaller. It is highly recommended that you execute a trial run (by checking the **Trial Run** check box) and print a report of the cleanup of any drivers you wish to remove. Seek expert advice and review the report carefully before actually removing these files. Prior to performing the cleanup it is advisable to make a backup of the specific files to be removed, as listed in the report.

To remove display drivers



Display Drivers

These drivers are available for your use but do not appear to be currently in use by in your Windows installation:

Display Drivers
8514/a
8514/a (Small fonts)
Super VGA (800x600,16 colors)
XGA (640x480,256 colors)

System Cleanup - Display Drivers

1. Press the **Display Drivers** button.
2. Currently unused drivers will appear in the list box. UnInstaller will suggest all non-standard (i.e., those drivers not included with Windows) drivers for deletion. Select any drivers that you wish to remove. Note that those which UnInstaller believes are safe to remove are automatically selected.
3. Press the **Cleanup** button to delete the drivers.
4. Information relating to disk space regained will be shown in the **Cleanup Statistics** boxes on the right. Please note that it is possible to select drivers that appear in the OEM.INF files, run a trial run and the results would state "No files would be deleted." This could be caused by one of two things:

   The associated driver has been previously deleted and only the detected entries in INF file(s) remain.

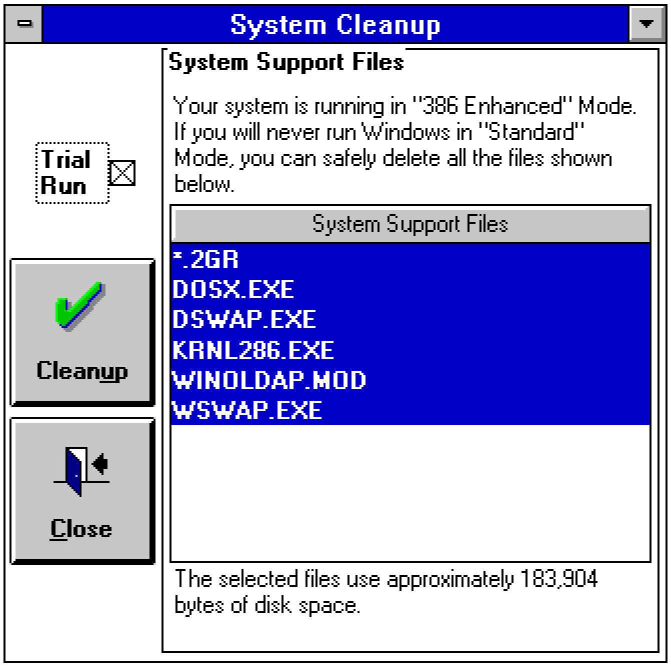
OR

   The selected driver is used by other unselected drivers, so they cannot be deleted without selecting all of the associated drivers. This is common with OEM drivers which should all have the same manufacturer's name with different resolution descriptions. In order to free up disk space you will need to select all of each group of drivers. If all are not selected, **System Cleanup** will just remove the entry from the INF file.

Caution: Extreme care should be exercised when removing these drivers. Some manufacturers rename their drivers when they are selected by Windows Setup for use. The driver might be highlighted as appearing to be unused, but in fact it is in use, just under another name. If you find that you have removed a currently selected driver, reinstall it by running Windows setup. From the DOS prompt, simply type **SETUP <Enter>** from the \WINDOWS directory.

• Removing System Support Files

To remove system support files



System Support Files

Your system is running in "386 Enhanced" Mode. If you will never run Windows in "Standard" Mode, you can safely delete all the files shown below.

System Support Files

*.2GR
DOSX.EXE
DSWAP.EXE
KRNL286.EXE
WINOLDAP.MOD
WSWAP.EXE

The selected files use approximately 183,904 bytes of disk space.

System Cleanup - System Support Files

Warning: After you press the **Cleanup** button, deletions are immediate and cannot be undone by UnInstaller.

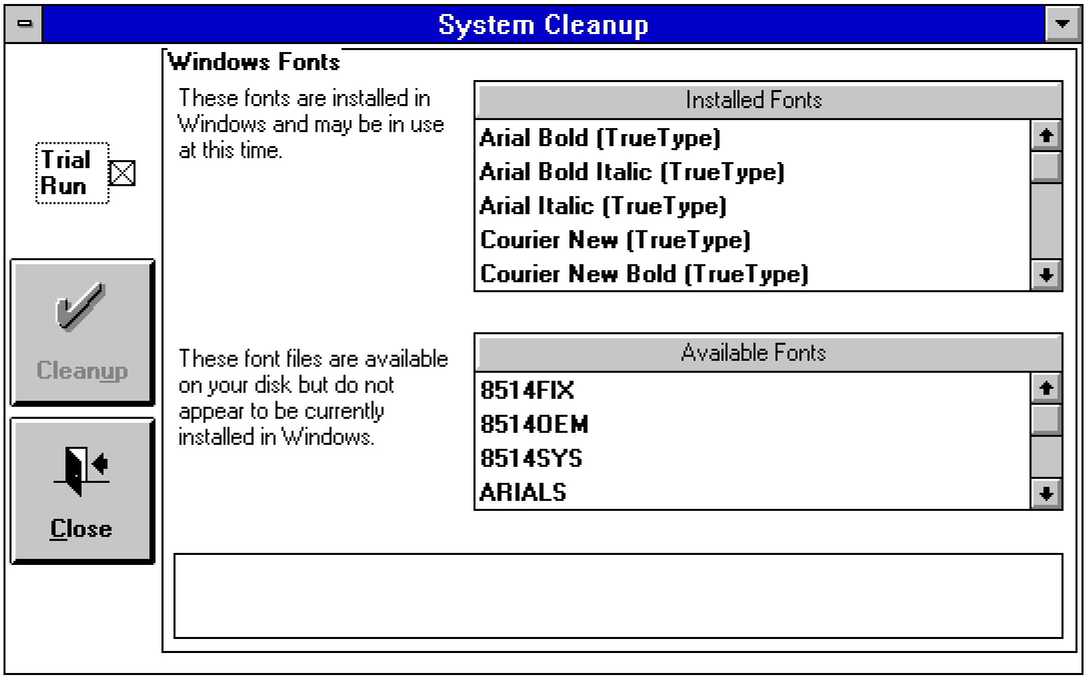
1. Press the **System Support Files** button.
2. Unused files will appear in the list box. Select any files that you wish to remove. Note that those which UnInstaller believes are safe to remove are automatically selected.
3. Press the **Cleanup** button to delete the files.
4. Information relating to disk space regained will be shown in the **Cleanup Statistics** frame on the right.

• Removing Windows Fonts

Warning: After you press the **Cleanup** button, deletions are immediate and cannot be undone by UnInstaller.

Note: Use caution when deleting "Installed Fonts". Remember that these fonts are known by Windows, but they are not currently in use. You may have one or more applications that expect to be able to use these fonts when they load.

To remove fonts



These fonts are installed in Windows and may be in use at this time.

Installed Fonts

- Arial Bold (TrueType)
- Arial Bold Italic (TrueType)
- Arial Italic (TrueType)
- Courier New (TrueType)
- Courier New Bold (TrueType)

These font files are available on your disk but do not appear to be currently installed in Windows.

Available Fonts

- 8514FIX
- 8514OEM
- 8514SYS
- ARIALS

System Cleanup - Windows Fonts

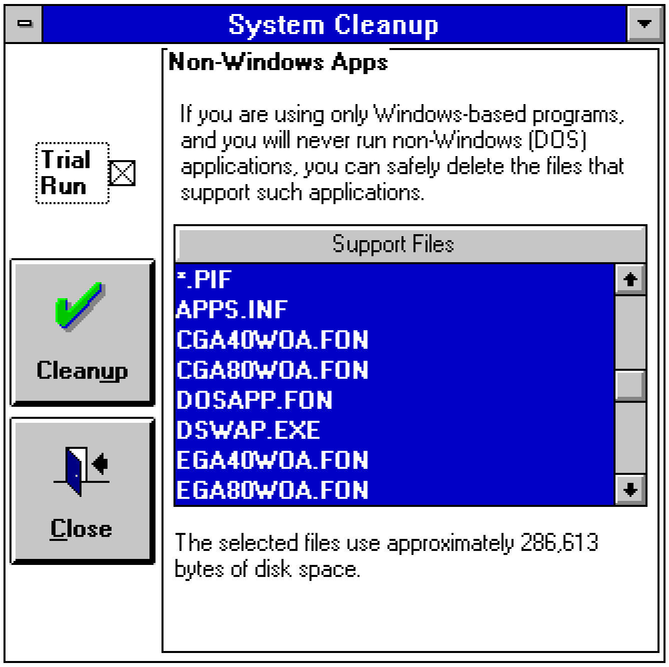
1. Press the **Windows Fonts** button.
2. Two sets of font listings will appear: **Installed fonts** may be in use by your system; **Available fonts** are not currently in use and can be removed. Select any font files that you wish to remove.
3. Press the **Cleanup** button to delete the files.

• Removing Non-Windows Apps Support Files

Note: The files displayed by this tool are required to run DOS applications in a windowed DOS box. This capability is available only in Windows 386 enhanced mode. Files used for your current resolution (VGA, EGA, CGA) are *not* displayed or available for deletion. You may safely delete these files as long as you won't be running windowed DOS applications in a different resolution than the current one.

Warning: After you press **Cleanup**, deletions are immediate and cannot be undone by UnInstaller.

To remove non-Windows Application support files



Non-Windows Apps

If you are using only Windows-based programs, and you will never run non-Windows (DOS) applications, you can safely delete the files that support such applications.

Support Files

```
*.PIF
APPS.INF
CGA40WOA.FON
CGA80WOA.FON
DOSAPP.FON
DSWAP.EXE
EGA40WOA.FON
EGA80WOA.FON
```

The selected files use approximately 286,613 bytes of disk space.

System Cleanup - Non-Window Apps Support Files

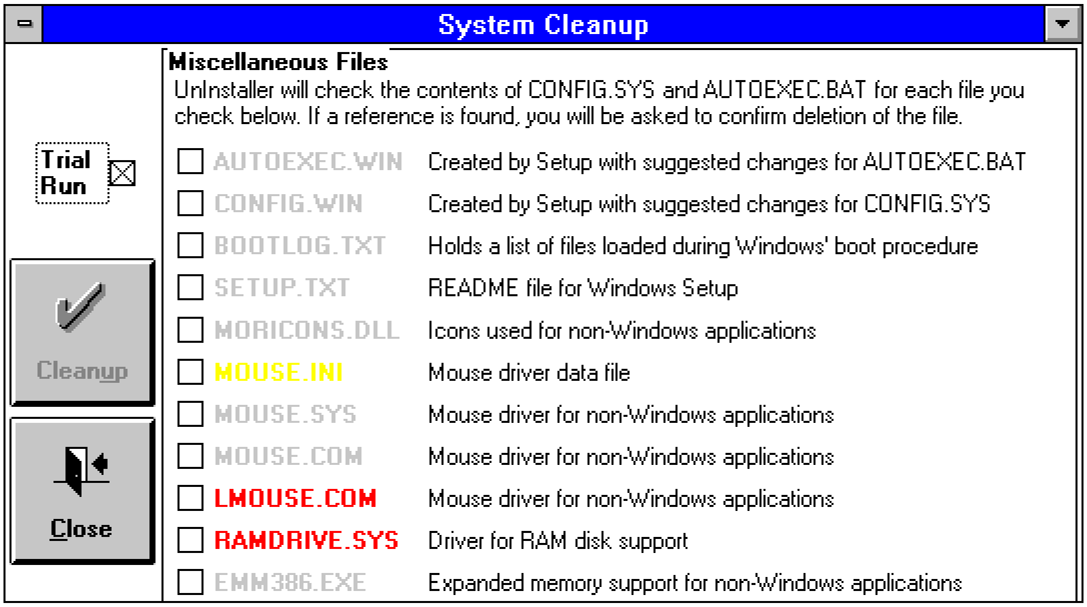
1. Press the **Non-Windows Apps** button.
2. Unnecessary files will appear in the list box. Select any files that you wish to remove.
3. Press the **Cleanup** button to delete the files.

• Removing Miscellaneous Files

UnInstaller color codes the files that it suggests for removal. File names that are grayed out are not present on your system. Files displayed in green can safely be removed. If a file's name is displayed in yellow, use caution before telling UnInstaller to delete the file. When you see a file displayed in red, you should delete the file only if you are absolutely sure you don't need it.

To remove miscellaneous files

Warning: Deletions are immediate and cannot be undone by UnInstaller.



Miscellaneous Files

UnInstaller will check the contents of CONFIG.SYS and AUTOEXEC.BAT for each file you check below. If a reference is found, you will be asked to confirm deletion of the file.

□	AUTOEXEC.WIN	Created by Setup with suggested changes for AUTOEXEC.BAT
□	CONFIG.WIN	Created by Setup with suggested changes for CONFIG.SYS
□	BOOTLOG.TXT	Holds a list of files loaded during Windows' boot procedure
□	SETUP.TXT	README file for Windows Setup
□	MORICONS.DLL	Icons used for non-Windows applications
□	MOUSE.INI	Mouse driver data file
□	MOUSE.SYS	Mouse driver for non-Windows applications
□	MOUSE.COM	Mouse driver for non-Windows applications
□	LMOUSE.COM	Mouse driver for non-Windows applications
□	RAMDRIVE.SYS	Driver for RAM disk support
□	EMM386.EXE	Expanded memory support for non-Windows applications

System Cleanup - Miscellaneous Files

1. Press the **Miscellaneous Files** button.
2. Check the names of any files that you wish to remove. Grayed file names were not found on your machine and are therefore unavailable for removal. When you click the **Cleanup** button, **System Cleanup** will search for any references to each checked file. If no references are found, the file will be immediately deleted. If a reference is found, the file is probably in use and you will be prompted for confirmation.
3. Press the **Cleanup** button to delete the files.

- ## Using INIClean

• Quick Viewing and Deleting of Associations

To view associations

1. In the **Display** box, choose **Associations in WIN.INI**.
2. In the **Which ones?** box, choose a display option.
3. Associations appear alphabetically in the List box. Lines begin with a file extension and show what application the extension launches.
4. To see exactly how an association is written to the WIN.INI file, double-click on it, or select it and choose the **View File...** button.

To delete associations

Warning: Deletions are immediate and cannot be undone by UnInstaller.

1. In the **Display** box, choose **Associations in WIN.INI**.
2. In the **Which ones?** box, choose a display option.
3. Associations appear alphabetically in the List box, sorted according to the extension. Lines begin with the file extension and show what application that extension launches.
4. Select the association you wish to delete. Associations can be selected only one at a time.
5. Choose the **Delete File/Item** button. Deletion does not take place unless you have selected an association.
6. A warning message appears the first time you try to delete anything. Choose **OK**. This does *not* remove the association.
7. A dialog box asks you to confirm the deletion. This dialog box also warns you in the event you are about to delete a standard Windows association (integral parts of Windows.) Remove such associations only if you know what you are doing. Choose **Yes** to delete the association, or **No** if you don't want to delete the association.

Second Warning: Deletions are immediate and cannot be undone by UnInstaller.

• Quick View, Delete & Edit of WIN.INI Sections

To view sections

1. In the **Display** box, choose **Sections in WIN.INI**.
2. In the **Which ones?** box, choose a display option.
3. Sections appear alphabetically in the List box. Each line begins with the sections name.
4. To view the contents of a section, double-click on the section name or click the section name once (to select it) and choose the **View File...** button.
5. When done viewing, choose **Close** or press **<Esc>.**

To delete sections

Warning: Deletions are immediate and cannot be undone by UnInstaller.

1. In the **Display** box, choose **Sections in WIN.INI**.
2. In the **Which ones?** box, choose a display option.
3. Sections appear alphabetically in the List box. Each line begins with the sections name.
4. Select the section you wish to delete. Note that Sections can be selected only one at a time.
5. Choose the **Delete File/Item** button. (The **Delete File/Item** button is not enabled unless you have selected a section.)
6. A warning message appears the first time you try to delete anything. Choose **OK**. This *does not* remove the section.
7. A dialog box asks you to confirm the deletion. This dialog box also warns in the event you are about to delete a standard Windows section (integral parts of Windows). Remove such sections only if you are positive that you know what you are doing. Choose **Yes** to delete the section or **No** if you don't want to delete the section.

Second Warning: Deletions are immediate and cannot be undone by UnInstaller.

Editing a section

Warning: Only advanced Windows users should use the edit option.

1. In the **Display** box, choose **Sections in WIN.INI**.
2. In the **Which ones?** box, choose a display option.
3. Sections appear alphabetically in the List box. Each line begins with the sections name.
4. Select the section you wish to edit, then choose the **Edit INI File...** button.
5. The Editor window appears, listing the contents of the section.
6. To change a line, select it and then make changes in the **Keyword** or **Value** edit fields. Choose the **Write** button to save the changes to disk. (The **Write** button is not enabled until you actually change an entry.) If you decide that you dont want to keep the changes to the selected line, choose the **Undo** button.
7. A line within a section can be removed by selecting the line and choosing the **Delete** button. All deletions are confirmed.
8. When finished, choose **Close** or press **<Esc>** to return to the **INIClean** window.

Quick Viewing and Deleting INI Files

To view INI files

1. In the **Display** box, choose **List of INI files.**
2. In the **Which ones?** box, choose a display option.
3. INI files appear alphabetically in the list box, along with their size and last date changed.
4. To view the contents of an INI file, double-click on the line in the list box or select the line and choose the **View File...** button.
5. When done viewing, choose **OK** or press **<Esc>**.

To delete INI files

Warning: Deletions are immediate and cannot be undone by UnInstaller.

1. In the **Display** box, choose **List of INI files.**
2. In the **Which ones?** box, choose a display option.
3. INI files appear alphabetically in the list box, along with their size and last date changed.
4. Select the INI file you wish to delete.   Only one INI file can be selected at a time.
5. Choose the **Delete File/Item** button.   (The button is not enabled until you have selected an INI file.)
6. A warning message appears the first time you try to delete anything. Choose **OK**.   This does *not* remove the file.
7. A dialog box asks you to confirm the deletion.   This dialog box also warns you if you are about to delete a standard Windows INI file (integral parts of Windows.)   Remove them only if you know what you are doing. Choose **Yes** to delete the INI file, or **No** if you do not want to delete the INI file.

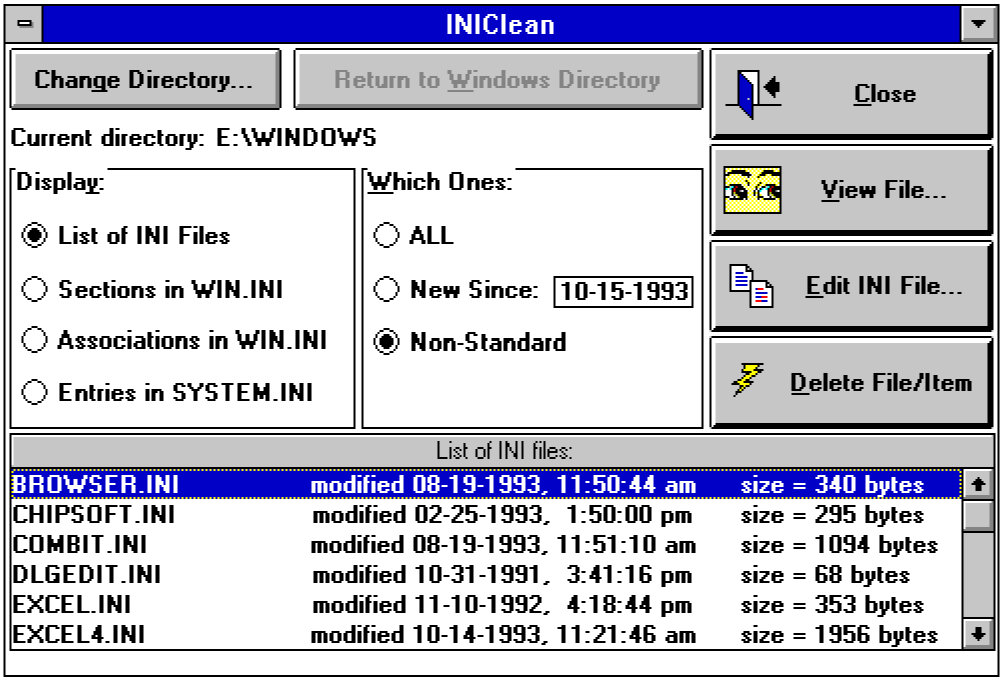
Editing an INI file

Warning: Only advanced Windows users should use the edit option.

1. In the **Display** box, choose **List of INI files**.
2. In the **Which ones?** box, choose **All**.
3. If you wish to edit an INI file that is located in a directory other than the current directory, press the **Change directory...** from the **File** menu, choose *any* INI file in the desired directory, and choose **OK**.
4. All INI files in the current directory appear alphabetically in the list box. Each line begins with the files name.
5. Select the file you wish to edit, then choose the **Edit INI File...** button. Note that **INIClean** does not allow editing of SYSTEM.INI because of the possible serious consequences involved in changing it.
6. A window appears that lists all the sections in the file that you are editing. Select the name of the section you wish to edit and choose the **Edit...** button.
7. The Editor window appears, listing the contents of the section.   To change a line, select it and make changes in the **Keyword** or **Value** edit fields.   Choose the **Write** button to save the changes to disk.   If you decide that you dont want to keep the changes to the line, choose the **Undo** button.
8. A line within a section can be removed by selecting the line and choosing the **Delete** button.   All deletions are confirmed.
9. When finished, choose **Close** or press **<Esc>** to return to the **INIClean** window.

• INIClean Overview

The **INIClean** feature of UnInstaller assists you in cleaning your INI files.   To use **INIClean,** choose **Tools-INIClean** from the main menu or press the button shown above.



Current directory: E:\WINDOWS

Display:

- ● List of INI Files
- ○ Sections in WIN.INI
- ○ Associations in WIN.INI
- ○ Entries in SYSTEM.INI

Which Ones:

- ○ ALL
- ○ New Since: 10-15-1993
- ● Non-Standard

List of INI files:

```
BROWSER.INI     modified 08-19-1993, 11:50:44 am   size = 340 bytes
CHIPSOFT.INI    modified 02-25-1993,  1:50:00 pm   size = 295 bytes
COMBIT.INI      modified 08-19-1993, 11:51:10 am   size = 1094 bytes
DLGEDIT.INI     modified 10-31-1991,  3:41:16 pm   size = 68 bytes
EXCEL.INI       modified 11-10-1992,  4:18:44 pm   size = 353 bytes
EXCEL4.INI      modified 10-14-1993, 11:21:46 am   size = 1956 bytes
```

The INIClean Window

Use the **Display** frame and the **Which Ones** frame to tell **INIClean** what you want to view.   (Both are explained below.)

Change Directory button

Displays a dialog that lets you change the directory to search for INI files.   The default is the directory where you have Windows installed.

Return to Windows Directory button

Returns to the directory where you have Windows installed.

Display frame

Choose what you would like to work with.   **List of INI Files** lets you specify the INI file.   **Sections in WIN.INI** lists all the sections in the WIN.INI file. **Associations in WIN.INI** lists the file extensions and the programs with which they are associated.   **Entries in SYSTEM.INI** lets you view the lines in SYSTEM.INI, exactly as they appear in the file.

Which Files frame

You may choose between listing **ALL** files, files that are **New Since** a certain date and **Non-Standard** files.

Close button

Click to close the **INIClean** window.

View File... button
Click to view the file you've selected from the list.   You may not edit the file when using this selection.

Edit INI File button
Choose this button to edit the INI file you have selected.

Delete File/Item button
Click this button to delete the file or item that is currently selected in the list box.

• Smart INICleaning

Tips, tricks & techniques

INIClean makes it easy to home in on the changes that applications make to Windows. It displays the associations and sections that have been added to WIN.INI and the INI files that have been added to your Windows directory. It will also show you the entries in your SYSTEM.INI file. However, **INIClean** cannot tell you what *should* be removed.

Sometimes common sense is enough to help you decide if something should be removed. For example, an association that says doc launches Microsoft Word is unnecessary if you no longer have Microsoft Word on your system.

Other times, deciding is not so easy. What do you think is dependent on a section in WIN.INI called MacWordConv? Maybe it has something to do with Microsoft Word or maybe it doesnt. (As it turns out, it does!)

To help you with these decisions, this section explains what **INIClean** shows you about associations, sections and INI files. Armed with this information, and a little common sense, youll be able to remove some of the clutter from your system. However, you should expect that in some cases, its best to leave well enough alone until an expert can tell you what to remove.

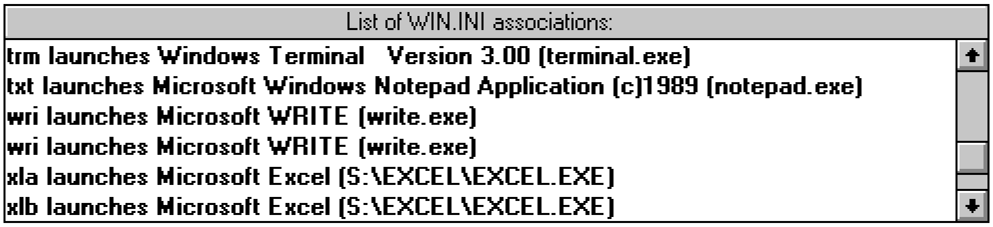
Make life easier: Narrow your search

The **Non-Standard** display option is the best friend of many users who clean Windows with **INIClean**. Choosing **Non-Standard** in the **Which ones** box of the **UnInstall Applications** window filters out all the associations, sections and INI files that are part of Windows when it is first installed. Filtering out these standard components makes it easier to concentrate on finding the things that may not be necessary on your system.

Associations: No sweat with INIClean

Associations are the easiest thing to clean-up when using **INIClean**. Most of the time, youll know right away whether an association is no longer needed.

INIClean lists associations alphabetically, by file extension. Each extension is followed by the name of the application it launches, as shown in the window below.



List of WIN.INI associations:

```
trm launches Windows Terminal   Version 3.00 (terminal.exe)
txt launches Microsoft Windows Notepad Application (c)1989 (notepad.exe)
wri launches Microsoft WRITE (write.exe)
wri launches Microsoft WRITE (write.exe)
xla launches Microsoft Excel (S:\EXCEL\EXCEL.EXE)
xlb launches Microsoft Excel (S:\EXCEL\EXCEL.EXE)
```

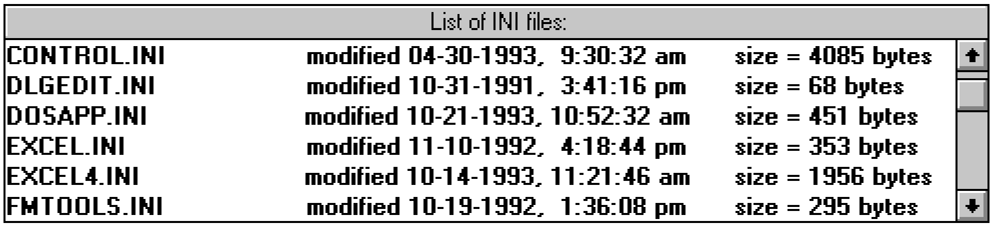
List of WIN.INI Associations

In some cases the name of the product appears, while other times only the filename is shown. **INIClean** lists the product names whenever it can find them.

In the above example, its easy to know whether to remove lines mentioning Microsoft Excel. All you have to do is ask yourself if you still have that application on your system. If not, then the associations are unnecessary and can be deleted.

You can see exactly how an association is written to the WIN.INI file by double-clicking on a line or selecting a line and choosing **View File...**.

INI files: For the advanced user, easy to delete



```
List of INI files:
CONTROL.INI     modified 04-30-1993,  9:30:32 am    size = 4085 bytes
DLGEDIT.INI     modified 10-31-1991,  3:41:16 pm    size = 68 bytes
DOSAPP.INI      modified 10-21-1993, 10:52:32 am    size = 451 bytes
EXCEL.INI       modified 11-10-1992,  4:18:44 pm    size = 353 bytes
EXCEL4.INI      modified 10-14-1993, 11:21:46 am    size = 1956 bytes
FMTOOLS.INI     modified 10-19-1992,  1:36:08 pm    size = 295 bytes
```

List of INI files

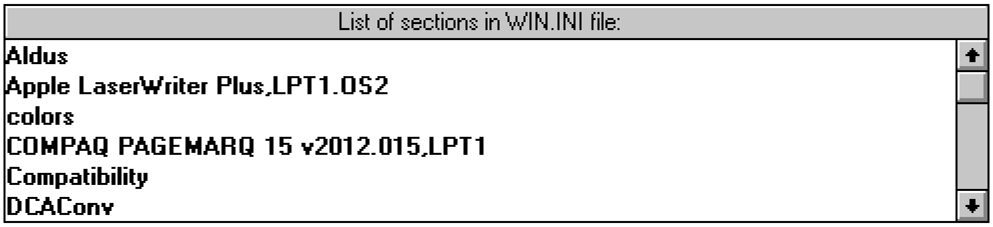
Sometimes, youll be able to tell which application created an INI file merely by looking at its filename.   Other times, you may be able to tell by viewing the contents of the file.   For example:

In the example above, EXCEL4.INI is a clear reference to Microsoft Excel and FMTOOLS.INI refers to MicroHelp FMTools.   If you no longer have either application, these INI files could be removed.   For any INI file you don't immediately recognize, you may need to consult with a Windows expert to tell you to which application it belongs.

The contents of an INI file can be viewed by double-clicking on a line or selecting a line and then choosing **View File...**.   Viewing the INI file may help you determine which application created the file.

WIN.INI sections: Sometimes a challenge.

Management of WIN.INI sections is tricky at best.   Sometimes a section takes an applications name, making it easy to know if it should be removed.   But often, theres just no way to know by looking.   Heres a sample WIN.INI section list:



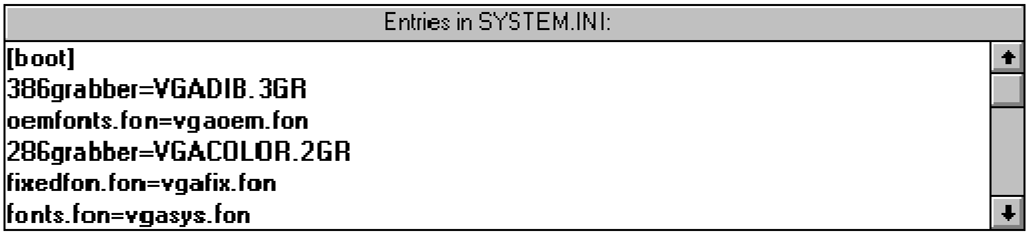
```
List of sections in WIN.INI file:
Aldus
Apple LaserWriter Plus,LPT1.OS2
colors
COMPAQ PAGEMARQ 15 v2012.015,LPT1
Compatibility
DCAConv
```

List of Sections in WIN.INI

This list can be loaded with trouble.   You might wish to remove the Aldus section from WIN.INI if you no longer have Aldus PageMaker on your system.   However, other Aldus products, such as Freehand, might be dependent upon the Aldus section.   Be certain before you delete such sections!

Entries in SYSTEM.INI: Danger lurks here

Warning: Changes to the SYSTEM.INI file should be made only by an expert and only when Windows is not running.   Because such changes can result in your not being able to restart Windows, **INIClean** does not have the ability to make these changes.



```
[boot]
386grabber=VGADIB.3GR
oemfonts.fon=vgaoem.fon
286grabber=VGACOLOR.2GR
fixedfon.fon=vgafix.fon
fonts.fon=vgasys.fon
```

List of entries in SYSTEM.INI

Only entries required by Windows should be found in the SYSTEM.INI file. Notice that section names appear in [square brackets] and each entry has an equal sign.

If you absolutely, positively have to make a change to one of these entries, we recommend the following steps:

1. Select the entry and press **<F1>** to display help for the entry.
2. Read and understand the help to make sure that what you are about to do is correct.
3. Make a backup copy of your SYSTEM.INI file.
4. Close UnInstaller and launch an edit program or Control Panel to make the desired changes. We will not assist you in making changes to your SYSTEM.INI file.
5. Restart Windows so that the changes take effect. Make sure everything works correctly.

Keeping up with the changes

Once youve given your system a good cleaning with **INIClean**, the **New since** display option offers a good way to stay on top of things. When selected, it shows you changes made since the last time you used **INIClean**. This makes it easy to review whats new on your system.

• Using INIClean Help

Help is at hand

INIClean has some very detailed information about the standard sections and entries in the WIN.INI and SYSTEM.INI files. You can access this information from the Help menu or by selecting a section or entry and pressing the **<F1>** key.

The on-line help will give you information about the default value (if any), limits (if any), how to change an entry and whether you should change an entry.

All the main windows of UnInstaller (**UnInstall Applications**, **System Cleanup**, **INIClean**, **DupeFinder** and **OrphanFinder**) have interactive help. When you see a likeness of the screen in the help file, move your mouse across the picture. When the cursor changes to a hand, you can get further help about that area by clicking your mouse.

- ## Using DupeFinder

Quick Dupe Finding

Quick Refreshing of Duplicate File List

Quick File Viewing

DupeFinder Overview

Smart Duplicate Deletion

- ## Quick Dupe Finding

To delete duplicate files

1. Select **Tools-Duplicate File Finder...** from the menu or press the button shown above**.**
2. Select the disk drive to analyze.
3. Press the **Search** button.
4. Select the files to be deleted.
5. Press the **Delete File** button.

Quick Refreshing of Duplicate File List

To refresh the duplicate file list

1. Uncheck the **Use file map from (date)** by clicking on it.
2. Click on the **Search** button.

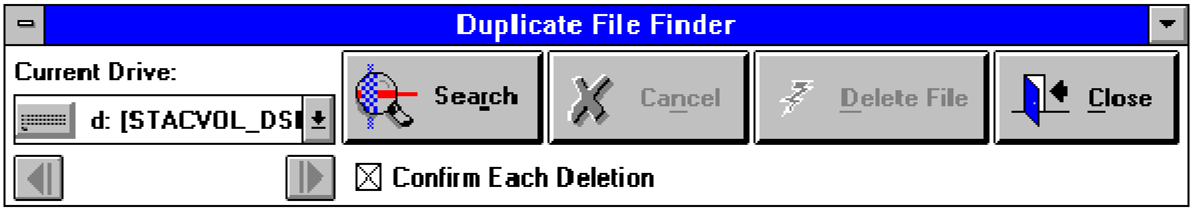
Quick File Viewing

To view files

1. Select the file you wish to view. Make certain it is the correct copy of the file by checking the path, date, time and size information on the line selected. (Note: If the file viewer is already open, you may also drag a file from Windows File Manager and drop it onto viewer window.)
2. Click on the **View** button. (If you prefer, you can simply double-click a file's name to view the file.)
3. The view screen will appear and display the file's contents. You can scroll through a long file using the scroll bars on the edges of the window or by pressing the arrow and page keys.
4. When you are finished viewing the file, either select **Exit** from the **File** menu or click on the **Control Box** (in the upper left hand corner of the window) and select **Close**.

• DupeFinder Overview

Duplicate File Finder (called **DupeFinder** from now on) is used to view and/or delete duplicate files found on a selected drive. To use **DupeFinder,** choose **Tools-Duplicate File Finder...** from the main menu or press the button shown above. The following window will be displayed:



Duplicate File Finder window

First, select which drive you want to search from the **Current Drive** drop down list. Next, press the **Search** button and **DupeFinder** will create a map file for that drive. Once the map file has been created, all files having the same name that exist in more than one place on the selected drive will be listed in the large list box.

Note: You can use or re-use the current map file by checking the box **Use file map from (date)** and pressing the **Search** button. This is helpful when you've deleted one or more duplicate files and you want to know what duplicates remain on your disk.

Note: If you have the single-user version of UnInstaller, **DupeFinder** will not search network drives it searches only local drives.

At this point you may want to be certain the **Confirm Each Deletion** box is checked. Scroll through the list, and select any files you would like to view or delete. You may select **Search** at any time to create a new map file that will display the changes that have been made.

To view the contents of a file, simply select the file that you wish to view and double-click on it or press the **View** button. To delete one or more files, select the file(s) and press the **Delete File** button. If the **Confirm Each Deletion** box is checked, you will be asked to confirm the deletion of each file. You can now select and search another drive, choose more files to delete, or close **DupeFinder**.

Note: If you select for deletion all files of the same name, and then press the **Delete File** button, **DupeFinder** will prompt you prior to deleting the last file having that name.

Please note that it is up to you to know what you are deleting. When in doubt, view the files, compare the dates and sizes of the files and make sure they are truly duplicates.

Listed below are descriptions of the controls in the **DupeFinder** main window:

Current Drive drop down list

Use this list to select the drive you want to search for duplicate file names. To select a different drive than the one shown, click on the down arrow and scroll through the list until you find the desired drive, then click on the desired drive. The list will close and the desired drive will be displayed as the current drive.

Note: If you have the single-user version of UnInstaller, **DupeFinder** will not search network drives   it searches only local drives.

Search button

Begins searching for duplicate files.

Note: This process may take a few minutes to complete.   The length of time depends on the number of files on the drive as well as the number of duplicates found.

View button

Note: This button appears only after duplicate files have been located.   This button is overlaid by the **Cancel** button when appropriate

Views the contents of the selected (highlighted) file.   If more than one file is selected, the last one selected is displayed.

UnInstaller understands approximately 20 file types.   If UnInstaller does not recognize the file type, it will default to a hexadecimal dump of the file.

If the viewer window is open, you may also drag files from Windows File Manager and drop them onto the viewer window.

Cancel button

Stops the current task.   This button is enabled at several points during the search process, which enables you to cancel the process.   (This button overlays the **View** button when appropriate.)

Delete File button

Deletes the selected file(s) from the drive.

Close button

Closes **DupeFinder**.

Confirm Each Deletion check box

Check this box to require confirmation before deletion of any file.

Use file map from (date) check box

You can use or re-use the current map file by checking the box **Use file map from (date)** and pressing the **Search** button.   This is helpful when you've deleted one or more duplicate files and you want to know what duplicates remain on your disk.

Previous Page and Next Page arrows

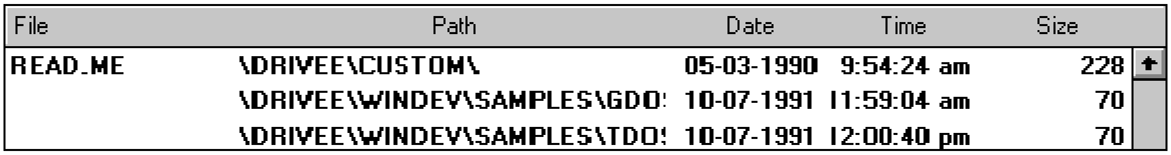
If you have a large number of duplicate files on the selected drive, and all the file names won't fit into memory at once, these buttons become enabled and you can press them to scroll through the list.   (Note These arrows are located under the **Current Drive** drop-down list.)   If all duplicate file names are present in the list box, these buttons will not be visible.

• Smart Duplicate Deletion
Tips, tricks & techniques

Using the **DupeFinder** feature, you can find which files may have extra copies of themselves in other directories on your machine. This duplication can obviously take up a lot of valuable disk space, but you need to be careful when deleting these duplicate files. **DupeFinder** cannot tell you what *should* be removed.

The most important concept to understand is that each program that makes use of a file needs to have access to that file. The application may be able to find and use the file if it's in another directory that is in the path (see *Paths*), but this is not always the case. Some applications may not be able to use a file if it's not in the main application directory in which it was originally installed.

There can be many different problems to consider with each duplicate file scenario. Let's say you find three instances of a READ.ME file in three separate application directories. There are a few possibilities: You may have three different applications that all use the same file or, more likely in this instance, these may be three totally unique files that all have the same name. (A third scenario is that two of the applications are using identical READ.ME files and the third is using a file containing a different version. In the case of a READ.ME file, this would be unlikely.) In this case, the only way to know for sure is to use the file viewer. Unfortunately, not all file types can be properly viewed and compared.



File	Path	Date	Time	Size
READ_ME	\DRIVEE\CUSTOM\	05-03-1990	9:54:24 am	228
	\DRIVEE\WINDEV\SAMPLES\GDO	10-07-1991	11:59:04 am	70
	\DRIVEE\WINDEV\SAMPLES\TDO	10-07-1991	12:00:40 pm	70

Three files with the same name

To help you with these decisions, this section explains what **DupeFinder** shows you about your duplicate files. Armed with this information, and a little common sense, youll be able to remove some of the clutter from your system. However, you should expect that in some cases, its best to leave well enough alone until an expert can tell you what to remove.

Note: The columns in the file listing shown above can be re-sized by you. As you move the mouse cursor between sections of the list box (between "Path" and "Date", for example), the cursor changes shape to indicate that you can size the column by holding the left button down and dragging the column's outline.

First Priority: BACKUP!

Always make a backup of ANY files you are considering for removal before using this feature. Either use a backup utility that you are familiar with, backup to floppies or use any other method you feel comfortable with.

Hurdle number one: The date/time/size test

The first thing to check when deciding whether or not to remove a file is the date, time and size of the file. If they don't match the other file(s) of the same name, and you are not familiar with the files, use the file viewer to compare the contents. If the file is not a readable format and you are not specifically familiar with the file as one you've made use of, such as a Word document you've updated and made copies of, it's best not to remove the file. If you do choose to remove the file or files, be certain to make backups of each one.

Hurdle number two: Who owns it?

If the files in question have the same date, time and size, the next step is to figure out which applications are making use of the file. Generally speaking, this can be done simply by determining which application created the directory in which the file is found (path). Many applications place files directly into the \WINDOWS and \WINDOWS\SYSTEM directories, so any files in those directories will be difficult to track or may be used by Windows itself. Files in those directories should be the last to be removed or left entirely alone. It is especially important to check the functioning of applications after removing files from these directories.

After each deletion, we suggest that you check the functioning of applications relating to the file that was removed. If you cannot determine which application or applications are using a file so that you can test the effects of the deletion properly, it would be best not to remove the file.

Hurdle number three: The path test

If a file being considered for removal is not in either the \WINDOWS or the \WINDOWS\SYSTEM directories, and it is not in your path (see *Path*), we recommend that you either:

> Place a copy of the file in your \WINDOWS directory, which could further clutter that already crowded directory, or
>
> Add the directory where the file is located to your path statement, which could create an overly long path.

After backing up the files, remove them and see if the affected applications still run properly. If they do not, the applications may not be able to use the file if it's not in the main application directory in which it was originally installed. Restore the file to its original location from the backup and move on to your next file.

Note: Changes made to your path statement in the AUTOEXEC.BAT will not take effect until the machine is rebooted. Changes made to the path statement from the DOS command line must not be done through a DOS shell you must exit Windows, make the path change and then restart Windows.

Who are the experts who can help?

If you need an experts help, make a note of the name, the directories (path), date, time and size information about the file(s) that you are confused about. Then you can seek out advice in several places.

Information services such as CompuServe or Prodigy offer a widespread forum for seeking advice. Many computer magazines offer help to users with questions about Windows and Windows applications. Product support lines maintained by software publishers can tell you what files their applications add or create.

MicroHelp cannot offer guidance in knowing what should be removed from your computer. There are simply too many applications out there to keep track of.

Using Orphan Finder

Quick Orphan Finding

To view a list of orphans

1. Start **Orphan Finder** by either selecting **Orphan Finder** from the **Tools** menu or by pressing the button shown above.
2. If you have never run **Orphan Finder** before, or if you wish to generate a fresh listing, press **Search**.
3. If you have run **Orphan Finder** before and you want to view the list of orphans found previously, press **Load Existing Orphans**.
4. Orphaned files will appear alphabetically in the list box. Lines begin with the file path (directory) and end with the file name.

• Quick Orphan Deletion

To delete orphans

1. Start **Orphan Finder** by either selecting **Orphan Finder** from the **Tools** menu or by pressing the button shown above.
2. If you have never run **Orphan Finder** before, or if you wish to generate a fresh listing, press **Search**.
3. If you have run **Orphan Finder** before and you want to view the list of orphans found previously, press **Load Existing Orphans**.
4. Orphaned files will appear alphabetically in the list box. Lines begin with the file path (directory) and end with the file name.
5. Select an orphaned file from the orphans list and press the **UnInstall** button.
6. The selected orphan file will automatically be loaded into the UnInstaller module. Review the list of recommended deletions as you would for any uninstallation. See *UnInstaller Basics* for more details.
7. When you are satisfied with the listing of recommended changes and deletions to be made, press the **UnInstall** button.
8. When the uninstall is complete, press **Close** to return to **Orphan Finder**. (If all the items in the list were selected for deletion and successfully deleted, you will automatically be returned to the **Orphan Finder** window.)

- ## Quick Shell Scan

To view shell references

1. Start **Orphan Finder** by either selecting **Orphan Finder** from the **Tools** menu or by pressing the button shown above.
2. Press the **Scan Shell** button.
3. Files referenced by your shell will appear alphabetically in the list box. Lines begin with the file path (directory) and end with the file name. These files are referenced by your shell and will not be listed as orphans.

• Quick Shell Reference Additions

To add shell references

1. Start **Orphan Finder** by either selecting **Orphan Finder** from the **Tools** menu or by pressing the button shown above.
2. Press the **Scan Shell** button.
3. Files referenced by your shell will appear alphabetically in the list box. Lines begin with the file path (directory) and end with the file name. These files are referenced by your shell and will not be listed as orphans.
4. Press the **Add File...** button to add an application file to the shell references list. Any files added to this list will no longer be considered orphans and will not be displayed on the orphaned files list. Note that because UnInstaller uses an extended selection list box, you can add multiple files from the same directory. See Extended Selection List boxes for details on using extended selection list boxes.
5. Select one or more files.
6. Press the **OK** button.
7. To add more files, press **Add File...** and repeat steps 5 and 6.

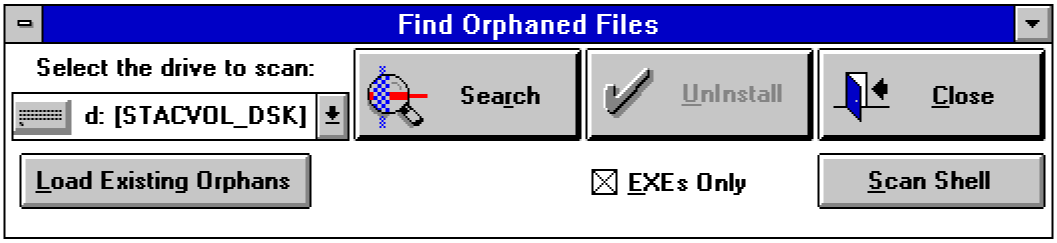
• Orphan Finder Overview

Note: If you are using PCTools for Windows (PCTW), or another shell that doesn't support DDE requests, you should use the **Scan Shell** feature, described under *Advanced Options*. After scanning the shell, you should use the **Add File** button to make sure that **Orphan Finder** knows about all the installed Windows applications in your system. Press the **Add File** button, and add the file name, for each Windows application that is installed in your system, but not shown in the list box generated as a result of the **Scan Shell** operation.

The reason this is necessary is that **Orphan Finder** uses DDE to request a list of all installed applications from the shell. Because PCTW does not support DDE, **Orphan Finder** cannot obtain a complete list of applications. We have contacted Central Point Software, the publishers of PCTW, explained the problem and asked for assistance. They have not responded as of this writing.

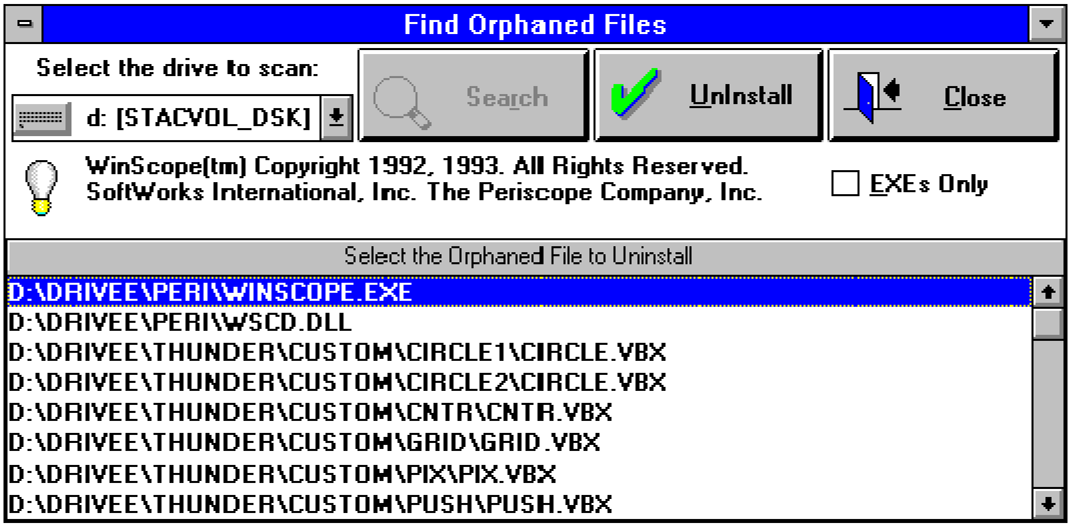
Orphan Finder is used to locate and remove orphaned files. (The definition of an "orphan" is a Windows EXE that is not referenced, or installed, in your shell (desktop) or a DLL that is not used by any of the applications referenced in the shell.)

You specify the drive you would like to scan for orphans and then press the **Search** button to view a listing of all orphaned files found. If you use **Orphan Finder** more than once during an UnInstaller session, you can press the **Load Existing Orphans** button to recall the most recent list of orphans that was generated. (The orphan list is not deleted when you exit UnInstaller the list is saved between UnInstaller sessions.)



The main Orphan Finder window.

You may then choose to uninstall any orphan found in the list by selecting an orphaned file and pressing the **UnInstall** button.



Select the drive to scan: d: [STACVOL_DSK]

WinScope(tm) Copyright 1992, 1993. All Rights Reserved.
SoftWorks International, Inc. The Periscope Company, Inc.

```
D:\DRIVEE\PERI\WINSCOPE.EXE
D:\DRIVEE\PERI\WSCD.DLL
D:\DRIVEE\THUNDER\CUSTOM\CIRCLE1\CIRCLE.VBX
D:\DRIVEE\THUNDER\CUSTOM\CIRCLE2\CIRCLE.VBX
D:\DRIVEE\THUNDER\CUSTOM\CNTR\CNTR.VBX
D:\DRIVEE\THUNDER\CUSTOM\GRID\GRID.VBX
D:\DRIVEE\THUNDER\CUSTOM\PIX\PIX.VBX
D:\DRIVEE\THUNDER\CUSTOM\PUSH\PUSH.VBX
```

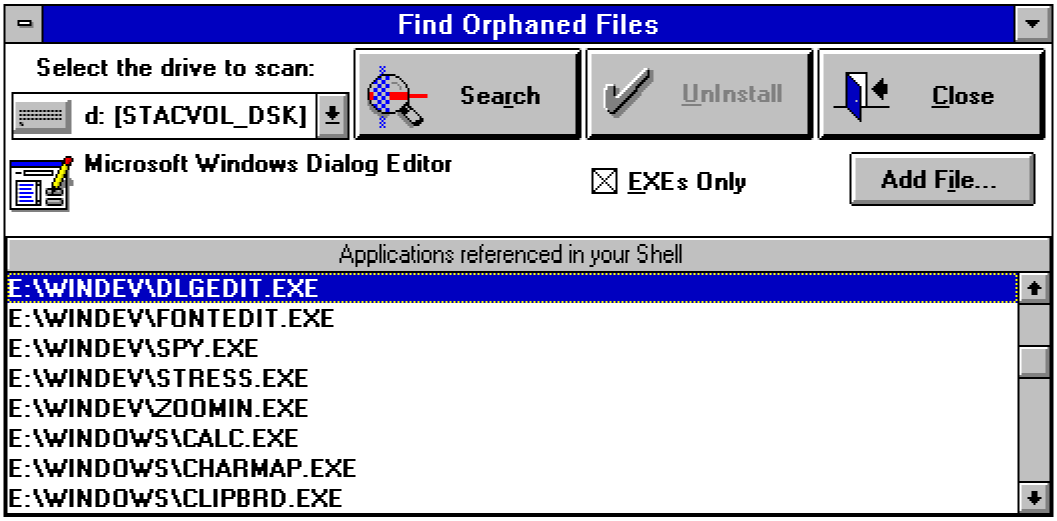
A list of orphaned .EXEs found. The next step is to choose one to uninstall.

When you press the **UnInstall** button, the selected file is passed to the UnInstaller tool for analysis. At that point, you can uninstall the orphan just as you would uninstall any other application. (For more details on uninstalling an application, please see *UnInstalling an Application*.) When you have finished uninstalling orphaned files, press the **Close** button to return to the **Orphan Finder** window.

• Orphan Finder Advanced Options

Before you scan for orphaned files, there are two options you can select. To search only for orphaned EXE files, check the **EXEs Only** check box. (When this box is unchecked, **Orphan Finder** searches for both orphaned EXEs and DLLs, which takes longer than searching only for EXEs.)

To view a listing of application files that are referenced or installed in your shell, press the **Scan Shell** button. All applications recognized by your shell program will appear in the list. If you find that an application you use is *not* listed, you may add it to the list of referenced applications by pressing the **Add File...** button and selecting a file to add. When constructing the orphaned file list, **Orphan Finder** will exclude any files shown in this list box and will not display them on the orphaned files file list. In addition, the added files are scanned for any dependent DLLs and they can also be excluded.



Select the drive to scan: d: [STACVOL_DSK]

Microsoft Windows Dialog Editor

☒ EXEs Only

Applications referenced in your Shell

```
E:\WINDEV\DLGEDIT.EXE
E:\WINDEV\FONTEDIT.EXE
E:\WINDEV\SPY.EXE
E:\WINDEV\STRESS.EXE
E:\WINDEV\ZOOMIN.EXE
E:\WINDOWS\CALC.EXE
E:\WINDOWS\CHARMAP.EXE
E:\WINDOWS\CLIPBRD.EXE
```

These applications are referenced and installed in the shell (desktop) and will not be listed as orphans.

When you are satisfied that all of your usable applications are listed properly, press the **Find Orphans** button to search for orphaned files.

◆ Smart Orphan Deletion

Orphan Finder helps you find disconnected or "orphaned" files in your system. These files are extraneous, unused files that can waste valuable hard disk space. **Orphan Finder** will display all the orphaned files that it finds in an easy to view list so you can remove them and free up hard disk space quickly. However, **Orphan Finder** cannot tell you what *should* be removed.

The definition of an "orphan" is:

1. A Windows EXE that is not referenced or installed in your shell., or
2. A Windows DLL that is not used by any of the applications referenced in your shell. (A shell refers to either the Windows Program Manager or a third party shell program such as Norton's Desktop for Windows - NDW.)

Note: Some applications do not install ALL of the required EXE files into your shell. The installed applications execute these "other" program files directly through Windows. Since these "other" program files are not directly installed in your shell, they would be considered orphans, when in fact they are not.

Sometimes it's relatively easy to recognize a "false" orphans that should not be removed. For example, let's say you're using Dashboard as your shell program instead of Program Manager. In your list of "orphaned" files you see C:\DASH\DASHGRP.EXE. This is not a recognizable EXE file that you are familiar with, but if you're still running Dashboard, it's safe to assume that it's an integral part of the Dashboard program and should be left alone.

Other times, recognizing a true orphan is not so easy. Let's say you see a listing for C:\WINDOWS\SYSED.EXE. It's obviously associated with your Windows package, but is it an "extra" companion tool that you never actually use? Or is it a file necessary for the proper functioning of Windows that is used "invisibly?" (As it turns out, the file is an extra tool that is very useful for viewing and editing your CONFIG.SYS, AUTOEXEC.BAT and INI files, but is not necessary for the proper functioning of Windows.)

To help you with these decisions, this section explains what **Orphan Finder** shows you about these "long-lost" files, or orphans, floating around on your hard disk. Armed with this information, and a little common sense, youll be able to remove some of the clutter from your system. However, as with many of these tools, you should expect that in some cases, its best to leave well enough alone until an expert can tell you what to remove.

Process of elimination: Shorten your list

To help you make your decisions, **Orphan Finder** will display icons and descriptions for those applications that have them. Using these descriptions, you may discover a long-lost "orphan" you wish to adopt into your regular working environment again!

If you go through your orphans list and find application files that you don't wish to delete (either because they are "false" orphans or because you just prefer to keep them), you may wish to add them to your shell reference list. In that case, select **Close,** restart **Orphan Finder** and then select **Scan Shell**. The list of files displayed are those applications that will NOT be considered orphans because they are referenced by your shell. Select **Add File** and choose a file you wish to remove from the orphans list and add it to the referenced files list.

The files you select will be added to the list of applications that are not considered orphans. They will no longer be displayed on the orphans listing so long as you select to **Load Existing Orphans** when you start **Orphan Finder** in the future.

Note: Each time you select **Scan Shell**, any changes made to this list are lost and the list will be regenerated.

EXEs vs. DLLs: Which to display?

Check the **EXEs Only** check box for the fastest operation. But, beware that Orphan Finder will not find DLLs from executables that have already been deleted. If you wish to view DLLs *and* EXEs, then leave **EXEs Only** unchecked.

In terms of deciding which files should be removed as orphans, it's always safer to assume that files found in directories of applications that you still wish to use should be left alone. For example, if you still use Dashboard and you see C:\DASH\DASHGRP.EXE listed, it would be safer to assume that this file should NOT be removed.

Keeping your listings current

Once youve removed orphans and updated the listings of your recognized applications for the first time, selecting **Load Existing Orphans** will continue to use your listings of recognized applications. As long as you remove any future applications using UnInstaller, you should not need to create a new map file of orphans in your system. If for some reason, you wish to do a fresh scan, pressing **Search** will create a new map file of any orphans found, however any additions you have made to your recognized applications list will be lost.

Who are these experts who can help?

If you need an experts help, note the name of the file that you are confused about. Then you can seek out advice in several places.

Information services such as CompuServe or Prodigy offer a widespread forum for seeking advice. Many computer magazines offer help to users with questions about Windows and Windows applications. Product support lines maintained by software publishers can tell you what files their applications create and use.

MicroHelp can not offer guidance in knowing what should be removed from your computer. There are simply too many applications out there to keep track of.

- ## Using Restore Backup

Quick Backup Restoration

Overview

Recovering Disk Space Used by Backup

Smart Backup Restoration

- ## Quick Backup Restoration

To restore an application

1. Select **File-Restore Backup** from the menu or press the button shown above**.**
2. Select the application to be restored.
3. Press the **Restore** button.
4. If the restore operation is successful, you are returned to the main UnInstaller window.

Note: Once an application has been restored, you can return to the Restore Backup window and press the **Delete File** button to recover the disk space used by the backup file.

• Restore Backup Overview

Reinstalling the UnInstalled

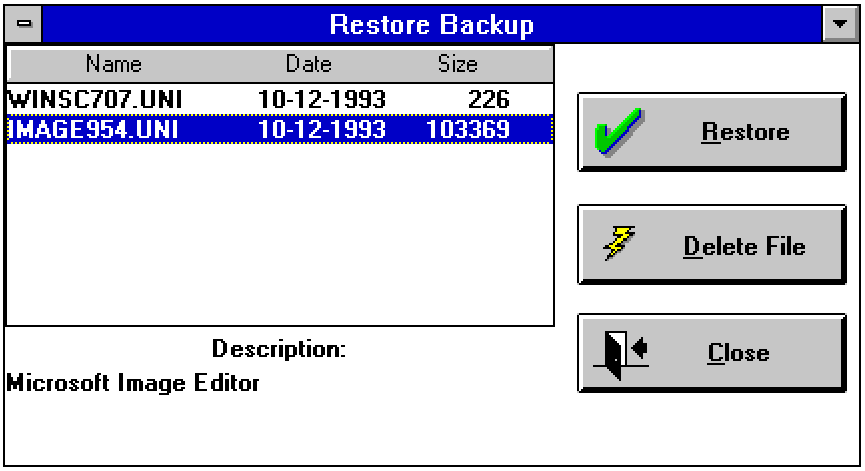
Restore Backup restores applications that were uninstalled using UnInstaller, with the **Backup** check box checked. **Restore Backup** restores an application's files, sections in WIN.INI and the application's icon to your shell program.

You start the **Restore Backup** process by choosing **File-Restore Backup** from the menu or by clicking the button shown above.

Note: The **Restore Backup** feature is not intended as a means of backing up applications. It is intended to be used only to reinstall an application immediately after it has been UnInstalled in case an error occurred..

Note: The file name UnInstaller creates for the backup is made up of the first five letters of the application's file name, followed by three random numbers and the UNI extension. PKZIP can list the files in the archive but it and other programs cannot uncompress the files.

Caution: The **Restore Backup** feature cannot restore "registration database" information as used with OLE (Object Linking and Embedding). The only reason you might wish to restore the registration database is if you truly want to re-install the application. In that case, you should re-install the application from the original disks.



Restore Backup

Name	Date	Size
WINSC707.UNI	10-12-1993	226
IMAGE954.UNI	10-12-1993	103369

Description:

Microsoft Image Editor

Restore Backup main window

Backup restoration is automatic. Simply select the backup you want to restore and select **Restore**. Note that each time a backup file is created during the uninstall process, a description of the application is stored in the backup file. When you select a line in the list box, the description of the application (assigned when you originally installed it) is shown below the "**Description:**" caption. If no description exists, UnInstaller uses the complete file name as the description.

• Recovering Disk Space Used by Backup

After you are sure that a backup was successfully restored to your system, we recommend that you again invoke the **Restore Backup** option, select the file you used to restore the application, then press the **Delete File** button to remove the backup file from your hard disk. Press the **Close** button to leave the **Restore Backup** window.

Note: If a backup file remains on your hard disk for seven days or more, UnInstaller will remind you that the file is still there whenever you launch UnInstaller. If you are satisfied that removal of the application was successful, you may wish to delete the backup file.

◆ Smart Backup Restoration

Further Precautions

Do not move the UNINSTAL.EXE file to another directory. The backup files are saved in the same directory as UNINSTAL.EXE and UnInstaller will not be able to find them if you move its executable file.

If you chose not to delete an application's icon when you uninstalled the application, the **Restore Backup** feature will reinstall the same icon and you will have multiple icons for that application in your group. You can delete one of them manually.

If you chose not to delete a file during the uninstall process, and that file is still present, **Restore Backup** will ask if you want to replace it with the file from the backup.

Don't remove the backup file itself until you are certain Windows and the restored application both operate normally.

When you get rid of a backup file by pressing the **Delete File** button in the **Restore Backup** window, the file is deleted immediately and is not recoverable by UnInstaller. However, the DOS "undelete" command can rescue the file.

When you select **Restore Backup**, you do not have an option as to which files are restored. All of the application's primary files which UnInstaller removed will be restored when **Restore Backup** is selected.

Don't expect to have documents restored by **Restore Backup**. UnInstaller does not back up documents. It only backs up an application's primary files as well as sections and entries that were removed from the WIN.INI file, the application's icon and the "all other files in directory", if you selected that option.

Uninstallable Items

Uninstallable Items Overview

When you select an application to uninstall and press the **Analyze** button, UnInstaller searches for the items listed in this section.

The items are listed alphabetically, using the following format:

Description
A brief description of the item.

Default selection
Auto-selected means the item is automatically selected for deletion. Not auto-selected means that UnInstaller displays the item but the user must select it if deletion is desired.

Recommendation
General guidance on deleting the item.

Confirmation
Always confirmed means that UnInstaller always asks you to confirm deletion of the item, even if the **Confirm all deletions** check box has been cleared. Depends on the state of the **Confirm all deletions** check box means that you are not prompted to confirm deletion of the item if the **Confirm all deletions** check box is cleared.

Backed up
Indicates if this item is backed up if the **Backup** check box is checked. If the item is backed up, it can be restored at a later time using the **Restore Backup** feature.

Alphabetical Listing of Uninstallable Items

All files with extension *ext* on local drives (individually selectable)

Association *ext=Program.EXE* from WIN.INI and system registration database

Batch file in PIF: *filename.BAT*

Delete all other files in *directory* and its subdirectories (individually selectable)

DLL *filename.ext* (is used by other programs)

DLL *filename.ext* (not used by scanned programs)

DLL *filename.ext* (may be used by other programs)

Extension *ext* from DOCUMENTS= line of WIN.INI

Help file in program directory: *filename.HLP*

Help file in Windows directory: *filename.HLP*

Help file in System directory: *filename.HLP*

Help annotation file: *filename.ANN*

INI file in program directory: *filename.INI*

INI file in Windows directory: *filename.INI*

INI file in System directory: *filename.INI*

Main program: *filename.ext*

OLE object type *Object name*

Program *Program* from LOAD= line of WIN.INI

Program *Program* from RUN= line of WIN.INI

Program Manager icons referencing *Program* (with individual confirmation)

Program Manager icons with extension *ext* (with individual confirmation)

Remove *directory* (and subdirectories) if empty when done

Section [*SectionName*] of WIN.INI

Section [SPOOLER] of WIN.INI

Set SPOOLER=NO in WIN.INI

Select this line to create a SmartDecoy

All files with extension *ext* on local drives (individually selectable)

Description

This is a convenient way to remove data files (documents, spread sheets, etc.) that an application has created. When you select this option, UnInstaller scans your local hard disk or disks (but not network drives) to find all files ending with an extension associated with the application being removed. These files are then listed and you can select the ones you want to delete. A viewer window is provided so that you can view a file before deciding if you want to delete it. See *Quick Working with File Viewer* for more information.

Default selection

Not auto-selected.

Recommendation

Select if you would like to remove some or all of the documents that an application has created with the given extension.

Confirmation

Always confirmed.

Backed up

No

Association *ext=Program.EXE* from WIN.INI and system registration database

Description

If you check this item, it causes UnInstaller to remove the association from the [Extensions] section of WIN.INI and from the system registration database. See *Associations Primer* for a full explanation of file associations.

Default selection

Auto-selected.

Recommendation

Delete the association.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Batch file in PIF: *filename.BAT*

Description

DOS applications may have one or more batch files that are executed when the application is run. These batch files (*.BAT) are referenced in a DOS application's main program PIF file. See *DOS and UnInstaller* for more information on PIF files.

Default selection

Auto-selected.

Recommendation

Delete this file to remove the program from your system.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Delete all other files in *directory* and its subdirectories (individually selectable)

Description

UnInstaller provides the opportunity to delete files remaining in an application's directory or subdirectories, after it has deleted the main program files and other selected items.   UnInstaller lists all the remaining files and you can select individual files to be deleted.

Default selection

Auto-selected only if the directory name matches the main program name.

Recommendation

Delete all the remaining files if the only application in that directory is being uninstalled.

Confirmation

Always confirmed.

Backed up

Yes

Note: This item is not listed if the program is located in your \WINDOWS or \WINDOWS\SYSTEM directory or in the root directory of any drive.

DLL *filename.ext* (is used by other programs)

DLL *filename.ext* (not used by scanned programs)

Description

Any DLLs used by the application being uninstalled are listed. DLLs are additional program files used by the main application. It is normal for multiple applications to share a DLL. If **SmartLinks** (see *SmartLinks*) knows of other applications using a DLL, the DLL is listed in a confirmation dialog box with the notation **(is used by other programs)**. If **SmartLinks** does not know of any other applications that use the DLL, the DLL is listed with the notation **(not used by scanned programs)**. Please note that applications that are not located in your current shell may not be scanned. For further information, please see *File Deletion Primer*.

Default Selection

"is used by other programs" not auto-selected. "not used by scanned programs" auto-selected.

Recommendation

The best strategy is to make sure that any applications you wish to use are in your current shell. You can then decide whether each DLL should be deleted. It may be advisable to make a floppy disk backup copy of a particularly questionable DLL file and test your applications for proper functioning after removal of the DLL.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

DLL *filename.ext* (may be used by other programs)

Description

DLLs are additional program files used by the main application. These DLLs may be used by the application being uninstalled or other applications and are listed individually. It is normal for multiple applications to share a DLL. If UnInstaller cannot verify that the DLL is either used or not used by any other applications, the DLL is listed with the notation **(may be used by other programs)**. This will occur if UnInstallers **SmartLinks** (see *SmartLinks*) database has not been created or updated to contain the selected application. This means that the DLL cannot be verified in the **SmartLinks** database and should not be deleted until the **SmartLinks** database has been updated for verification.

Default Selection

Not auto-selected.

Recommendation

The best strategy is to update/create the **SmartLinks** database using the **File-Create/Update SmartLinks** menu option. Then uninstall the application and when one of the more definite DLL usage verifications is noted, you can decide whether each DLL should be deleted. For more information, see *SmartLinks*.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Extension *ext* from DOCUMENTS= line of WIN.INI

Description

Some applications list their document extensions on the DOCUMENTS= line of WIN.INI.

Default selection

Auto-selected.

Recommendation

Delete the extension from the DOCUMENTS= line of WIN.INI.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

File: *filename.ext*

Description

Represents a file you manually added to the list for deletion.   You can choose the **Add file...** button to specify any additional files to be deleted.

Default selection

Auto-selected.

Recommendation

If you added a file using the **Add File...** button, then you probably already know that it can be deleted.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Help file in program directory: *filename.HLP*

Help file in Windows directory: *filename.HLP*

Help file in System directory: *filename.HLP*

Description

Most applications provide on-line help, contained in a file with an extension of HLP.   The help file is usually located in the same directory as the main program file, but it can be located in your \WINDOWS or \WINDOWS\SYSTEM directory.

Default selection

Auto-selected.

Recommendation

If the applications main program file is being deleted, then delete the help file also.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Help annotation file: *filename.ANN*

Description

Many applications have on-line help files to which you can add comments. These comments are stored in a help annotation file (with an extension of ANN) and are kept in your Windows directory.

Default selection

Auto-selected.

Recommendation

If you delete the applications help file, then you should also delete the help annotation file.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

INI file in program directory: *filename.INI*

INI file in Windows directory: *filename.INI*

INI file in System directory: *filename.INI*

Description

Many applications store settings and information that they need to run in a file with an extension of INI. The INI file is usually located in the Windows directory but can reside in the same directory as the main program file or in your System directory. Please refer to *INI Files Primer* for more details about INI files.

Default selection

Auto-selected.

Recommendation

If you are deleting the main program file, then delete the INI file, unless you know of another application on your system which needs the same INI file.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Main program: *filename.ext*

Description

Applications usually have one main file that is executed when the application is run. This is the main program file, and it usually has an extension of EXE. The main program file of a DOS based application that you use from within Windows will usually have an extension of PIF. See *DOS and UnInstaller* for more information on PIF files.

Default selection

Auto-selected.

Recommendation

Delete this file to remove the program from your system.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

OLE object type *Object name*

Description
If the program is an OLE server, the objects that it can create are listed here.

Default selection
Auto-selected.

Recommendation
If you are deleting the application from your system, then you should delete the OLE object type entries as well.

Confirmation
Depends on the state of the **Confirm all deletions** check box.

Backed up
No

Program *Program* from LOAD= line of WIN.INI

Program *Program* from RUN= line of WIN.INI

Description

Your WIN.INI file contains two types of lines which you can use to specify applications that are automatically launched when Windows starts up.   The lines can be identified because they start with either "LOAD=" or "RUN=".

These lines are commonly used to start applications such as a clock, network utilities or reminder (tickler) programs.

Default selection

Auto-selected.

Recommendation

If you are deleting the main program file, then you should also delete its entry on the LOAD= or RUN= line of WIN.INI.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Note: UnInstaller will not remove a program that is currently running.   You must close the application before attempting to uninstall it.

Program Manager icons referencing *Program* (with individual confirmation)

Program Manager icons with extension *ext* (with individual confirmation)

Description

When an application is installed on your system, one or more icons are usually created in Program Manager to represent the application. You launch the application by double-clicking on its icon. In addition, icons can be created to represent documents created by the application (for example, README.WRI, created by Write).

UnInstaller can remove both types of icons, as well as their associated files.

Default selection

Auto-selected.

Recommendation

The best way to use this line is to select it and then allow UnInstaller to prompt you to confirm the deletion of each icon.

Confirmation

Always confirmed.

Backed up

Yes

Note: Sometimes Program Manager does not correctly respond to UnInstallers attempts to delete icons. If this happens and UnInstaller reports that there were problems, choose the UnInstall button a second time immediately after your current uninstall process finishes. Often, this will solve the problem. If UnInstaller cannot delete the icon from Program Manager, you can do so manually by selecting the icon and pressing the **<Del>** key.

Remove *directory* (and subdirectories) if empty when done

Description

When UnInstaller is finished deleting files from the applications directory, it can delete any empty directories.

Default selection

Auto-selected.

Recommendation

Allow UnInstaller to remove empty subdirectories.   Any directory that still has files in it will not be affected.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Note: If you have the Network version of UnInstaller and you are creating a SmartDecoy, this item is not listed.

Section [*SectionName*] of WIN.INI

Description

Many applications add sections to your WIN.INI to store settings between sessions. This type of section should be removed if the section is not used by any other application remaining on your system.

Default selection

Not auto-selected.

Recommendation

It is sometimes difficult to determine which section belongs to which application and whether a section is shared by any other application. Therefore, the safest course of action is not to delete any section if you are not certain it can be deleted without effecting other applications. You may want to create a trial run uninstall report to make note of the section name and you can then use **INIClean** to delete the section later. Alternatively, you can select the section and when you are asked for confirmation to delete the section, you can choose the **View...** button to display the contents of that section.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Section [SPOOLER] of WIN.INI

Set SPOOLER=NO in WIN.INI

Description

If you are uninstalling the Windows Print Manager, these two items are present so you can remove Print Managers settings from your WIN.INI file.

Default selection

Auto-selected.

Recommendation

If you are uninstalling Print Manager, you should allow UnInstaller to make these changes.

Confirmation

Depends on the state of the **Confirm all deletions** check box.

Backed up

Yes

Select this line to create a SmartDecoy

Note: The SmartDecoy feature is available only in the Network version of UnInstaller.

Description

Select this item if you wish to create a SmartDecoy of the application that you are uninstalling.   See *SmartDecoy Primer* for a full explanation of SmartDecoys.

Default selection

Not auto-selected.

Recommendation

If you are uninstalling an application that is located on a network and it is used by other users on the network, then you should leave a SmartDecoy.

Confirmation

Always confirmed.

Backed up

Not applicable

Note: This item is handled differently from most of the other items because selecting it does not actually *delete* anything.   Instead, it enables the creation of a SmartDecoy.

Standard Windows Components

- Standard SYSTEM.INI Sections
- Standard WIN.INI Sections
- Standard Associations
- Standard INI Files
- Standard Fonts
- Standard Font File Extensions

Standard SYSTEM.INI Sections

The following sections of the SYSTEM.INI file are considered to be standard:

Section	Purpose
Boot	Lists drivers and Windows modules
Boot.description	Lists the names of devices you can change using Windows Setup
Drivers	Contains a list of aliases (or names) assigned to installable driver files
Keyboard	Contains information about the keyboard
MCI	Lists Media Control Interface (MCI) drivers
NonWindowsApp	Contains information used by non-Windows applications
Standard	Contains information used by Windows in standard mode
386enh	Contains information used by Windows in 386 enhanced mode

Note: There may be other sections required for the proper operation of your system.

Standard WIN.INI Sections

The following sections of the WIN.INI file are considered to be standard:

Section	Used by
Cardfile	Cardfile (CARDFILE.EXE)
Clock	Clock (CLOCK.EXE) (Windows 3.0)
Colors	Windows
Compatibility	Windows 3.1
Desktop	Windows
Devices	Windows
Display driver	Pen Windows
Embedding	Windows (Object linking & embedding [OLE] information)
Extensions	Windows
Fonts	Windows
FontSubstitutes	Windows
Intl	Windows
MCI extensions	Windows
MCI.*	Windows
MSCharMap	Character map (CHARMAP.EXE)
MSWrite	Write (WRITE.EXE)
Network	Network
Networks	Network
OldFonts	Windows
Paintbrush	Paintbrush (PBRUSH.EXE)
Ports	Windows
PrinterPorts	Windows
Programs	Windows 3.1
RegEdit	Registration Editor (REGEDIT.EXE)
Solitaire	Solitaire (SOL.EXE)
Sounds	Windows
Spooler	Print Manager (PRINTMAN.EXE)
Terminal	Terminal (TERMINAL.EXE)
TrueType	Windows
Windows Help	Windows Help (WINHELP.EXE)
Windows	Windows

Note: Not all sections are required for the proper operation of every system.

Standard Associations

The following extensions are considered standard:

Ext	Associated application	
BMP	Paintbrush (PBRUSH.EXE)	
CAL	Calendar (CALENDAR.EXE)	
CLP	Clipboard viewer (CLIPBRD.EXE)	
CRD	Cardfile (CARDFILE.EXE)	
HLP	Help (WINHELP.EXE)	
INI	Notepad (NOTEPAD.EXE)	
MID	Media player (MPLAYER.EXE)	MIDI file
MMM	Media player (MPLAYER.EXE)	Movie file
PCX	Paintbrush (PBRUSH.EXE)	
REC	Macro recorder (RECORDER.EXE)	
RMI	Media player (MPLAYER.EXE)	Sequencer file
TRM	Terminal (TERMINAL.EXE)	
TXT	Notepad (NOTEPAD.EXE)	
WAV	Sound recorder (SOUNDREC.EXE)	Waveform audio file
WRI	Write (WRITE.EXE)	

Note: **INIClean** does *not* verify that a standard extension is actually associated to the standard application.

Standard INI Files

The following INI files are considered to be standard by UnInstaller:

File	Used by
ATM.INI	Adobe Type Manager
CLOCK.INI	Clock (CLOCK.EXE)
CONTROL.INI	Control Panel (CONTROL.EXE)
DOSAPP.INI	DOS applications settings (Windows 3.1)
INICLEAN.INI	INIClean (INICLEAN.EXE)
MOUSE.INI	Mouse settings
MPLAYER.INI	Media Player (MPLAYER.EXE)
MSD.INI	Microsoft Diagnostics (MSD.EXE)
NDW.INI	Norton Desktop for Windows (NDW.EXE)
PENWIN.INI	Pen Windows
PROGMAN.INI	Program Manager (PROGMAN.EXE)
SYSTEM.INI	System & boot-up configuration
WIN.INI	File Manager (WINFILE.EXE)
WINMINE.INI	Minesweeper game (Windows 3.1) (WINMINE.EXE)

Standard Fonts

The following fonts are considered to be standard by UnInstaller:

Font	Font type, spacing, and default sizes
Arial	TrueType, proportional, scalable
Arial Bold	TrueType, proportional, scalable
Arial Bold Italic	TrueType, proportional, scalable
Arial Italic	TrueType, proportional, scalable
Courier	Raster, fixed, 10,12,15
Courier New	TrueType, fixed, scalable
Courier New Bold	TrueType, fixed, scalable
Courier New Bold Italic	TrueType, fixed, scalable
Courier New Italic	TrueType, fixed, scalable
Modern	Vector, proportional, scalable
MS Sans Serif	Raster, proportional, 8,10,12,14,18,24
MS Serif	Raster, proportional, 8,10,12,14,18,24
Roman	Vector, proportional, scalable
Script	Vector, proportional, scalable
Small	Raster, proportional, 2,4,6
Symbol**	Raster, proportional, 8,10,12,14,18,24
Symbol**	TrueType, proportional, scalable
System	Raster, proportional, display-dependent size
Terminal*	Raster, fixed, display-dependent size
Times New Roman	TrueType, proportional, scalable
Times New Roman Bold	TrueType, proportional, scalable
Times New Roman Bold Italic	TrueType, proportional, scalable
Times New Roman Italic	TrueType, proportional, scalable

* OEM character set, rather than ANSI character set

** Symbol character set, rather than ANSI character set

Standard Font File Extensions

The following are some standard font file extensions:

Extension	Font Type
FON	Standard Windows
FOT	TrueType
PCM	Printer Cartridge Metrics
PFB	ATM Type 1
PFM	Windows Font Metrics
TTF	TrueType
USP	HP Downloadable

UnInstaller Errors

UnInstaller Errors Overview

The subjects discussed in this area are presented in Problem/Solution format. The problems are shown in bold type in the header and the solutions, or recommended actions, are in regular type.

Installation Errors

Error - Write protected disk

Machine hangs immediately - no part of installation is accomplished

Error - Write protected disk

The installation disk has been write protected and the license file cannot be updated.   Disable write protection on the disk and begin installation as normal.

Machine hangs immediately - no part of installation is accomplished

If Facelift by BitStream is loaded, this may occur. Temporarily unload Facelift and/or any other third party drivers and start installation as normal.

Errors Running UnInstaller

I try to launch UnInstaller, but it will not run

I received a message "Unable to load UnInstaller: incorrect VBX version for [name]. Please insure that the VBX files that were included with this product are accessible to UnInstaller."

I received a message "The file [name] is out of date. This program requires a newer version."

My computer reboots midway through an uninstall

UnInstaller beeps many times and then runs normally

I try to launch UnInstaller, but it won't run

This can be the result of a corrupted VBRUN300.DLL file.   Use **File Manager** or another utility to search your system for all instances of the file VBRUN300.DLL. A copy should be located in the \WINDOWS\SYSTEM directory (or the \WINDOWS directory, if you are running Windows from a network).   If there are any other copies on your system, make sure that they are not located earlier in your path than the copy in the \WINDOWS\SYSTEM or \WINDOWS directory. Rename the VBRUN300.DLL file to VBRUN300.OLD and reinstall the UnInstaller.   If you are not familiar with the 'Path', please see *Paths*.

I received a message "Unable to load UnInstaller: incorrect VBX version for [name]. Please insure that the VBX files that were included with this product are accessible to UnInstaller."

This is probably because you moved UnInstaller's executable file (UNINSTAL.EXE) to a different directory from the one where you installed it. First, use File Manager to look at the directory where you installed UnInstaller and make a list of the files having an extension of VBX. Then search the \WINDOWS and WINDOWS\SYSTEM directories, as well as any directories listed in your path, for files having the same name as the UnInstaller VBX files. If you find duplicate files, delete all but the most recent file (i.e., the one with the most recent date and time).

I received a message "The file [name] is out of date. This program requires a newer version."

This is probably because you moved UnInstaller's executable file (UNINSTAL.EXE) to a different directory from the one where you installed it. This can happen when you are running another Windows application that uses a VBX file with the same name as one of the UnInstaller VBX files. The solution is to exit Windows, restart Windows and then immediately run UnInstaller. Use **DupeFinder** to find duplicates of the VBX file and delete all but the most recent one.

My computer reboots midway through an uninstall

This error can be caused by a conflict between UnInstaller and Norton's Anti-Virus virus scanner. Disable the virus protection and begin the uninstall again.

UnInstaller beeps many times and then runs normally

Your system is busy trying to complete some other task. This could happen on a network or if you are running a program in the background.

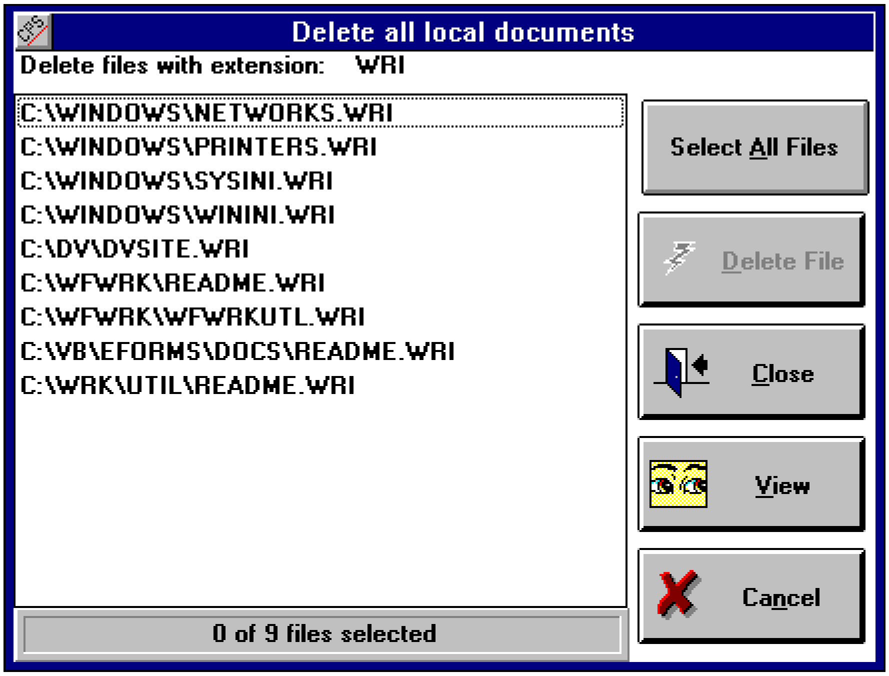
Troubleshooting

If none of the errors mentioned earlier in this chapter describes your problem, and you've reviewed the installation/operation instructions, and you're doing everything according to instructions, the following suggestion may offer some insight to your problem.

Exit out of Windows and, at the DOS prompt, type WIN /S. When Windows has loaded, attempt to complete what you were doing when you had the problem, whether it was an installation or an uninstall. If you find that this solves the problem, then there are probably third-party drivers (see *Drivers Primer*) or other applications that are being loaded when Windows is run in enhanced mode and one of those conflicts with the task you were attempting to perform.

The changes that may need to be made to correct this type of problem should be done only by advanced users. This is because the information these drivers use to load will be located in the "386enh" section of the SYSTEM.INI file. If you are not thoroughly familiar with this information, seek expert advice rather than making any modifications yourself.

Delete All Local Documents



Delete files with extension: WRI

C:\WINDOWS\NETWORKS.WRI
C:\WINDOWS\PRINTERS.WRI
C:\WINDOWS\SYSINI.WRI
C:\WINDOWS\WININI.WRI
C:\DV\DVSITE.WRI
C:\WFWRK\README.WRI
C:\WFWRK\WFWRKUTL.WRI
C:\VB\EFORMS\DOCS\README.WRI
C:\WRK\UTIL\README.WRI

0 of 9 files selected

When you are uninstalling an application, you are given the option to select which items you would like to delete. Frequently, an application has certain files associated with it. For example, files with the extension .WRI are associated with Windows Write. If you select "All files with extension 'wri' on local drives (individually selectable)" from the "select the items to be deleted..." list box, UnInstaller will search your hard drive for files that have the extension 'wri' and present them to you in a list box. You then have the option to delete one of the 'wri' files, some of the 'wri' files or all of the 'wri' files. Once you select the files you want and then press the **DELETE** button, UnInstaller will present you with a dialog box asking you to confirm your intention to delete the selected files. If you press **YES**, all the selected files will be deleted. If you press **NO**, you will be returned to the ***Delete All Local Documents*** window.

WIN.INI

[windows]

[Desktop]

[Extensions]

[intl]

[ports]

[fonts]

[FontSubstitutes]

[TrueType]

[mci extensions]

[network]

[embedding]

[Windows Help]

[sound]

[printerPorts]

[devices]

[programs]

[colors]

[windows] section

File: WIN.INI

This section contains entries that affect startup applications, the warning beep, printing, window border width, keyboard speed, mouse settings, and definitions of files as documents or programs.

You will find entries in this section for:

Beep
BorderWidth
Coolswitch
CursorBlinkRate
DefaultQueueSize
Device
DeviceNotSelectedTimeout
Documents
DosPrint
DoubleClickHeight
DoubleClickWidth
DoubleClickSpeed
KeyboardDelay
KeyboardSpeed
Load
MenuDropAlignment
MenuShowDelay
MouseSpeed
MouseThreshold1
MouseThreshold2
MouseTrails
NetWarn
NullPort
Programs
Run
ScreenSaveActive
ScreenSaveTimeOut
Spooler
SwapMouseButtons
TransmissionRetryTimeout

[Desktop] section

File: WIN.INI

This section contains entries that define the appearance of Windows.

You will find entries in this section for:

GridGranularity
IconSpacing
IconTitleFaceName
IconTitleSize
IconTitleWrap
IconVerticalSpacing
Pattern
TileWallpaper
Wallpaper

[Extensions] section

File: WIN.INI

This section contains definitions that associate file extensions with programs.

You will find entries in this section for:

file associations

[intl] section

File: WIN.INI

iCountry
iCurrDigits
iCurrency
iDate
iDigits
iLZero
iMeasure
iNegCurr
iTime
iTLZero
s1159
s2359
sCountry
sCurrency
sDecimal
sLanguage
sLis
sLongDate
sShortDate
sThousand
sTime

[ports] section

File: WIN.INI

This section lists the available communications and printer ports, defines default values, and lists files to which printer output can be sent.

This section can contain one or more entries in this format:

portname:=*baud-rate, parity, word-length, stop-bits* [[, p]]

[fonts] section

File: WIN.INI

This section lists the screen font files that Windows loads at startup.

Use the Fonts icon in Control Panel to change this section.

The entries in this section are in the following format:

font-name=font-filename

Where *font-name* is the descriptive name of a font and *font-filename* is the name of the file that contains the font resources.

Note: Adding entries to this section will not install a font. You must use the Fonts icon in the Control Panel to install new fonts.

[FontSubstitutes] section

File: WIN.INI

This section describes fonts recognized by Windows as identical to another typeface and contains one or more occurrences of the following:

font-name=*font-name*

Edit WIN.INI to change this section.

[TrueType] section

File: WIN.INI

This section describes options that affect the use of TrueType fonts in your applications.

You will find entries in this section for:

OutlineThreshold
TTEnable
TTIfCollisions
TTOnly

[mci extensions] section

File: WIN.INI

This section contains entries that associate types of media files with Media Control Interface drivers. It can contain one or more occurrences of the following:

extension=mcidevice-type

[network] section

File: WIN.INI

This section contains network settings. It can contain the following sections:

drive
InRestoreNetConnect
port

[embedding] section

File: WIN.INI

This section contains Object Linking and Embedding (OLE) objects, their description , the program that creates them and their file format. The [embedding] section is included only for backwards compatibility with Windows 3.0.

Entries in this section have the following format:

object=description, description2 program file, format

Where:

object = the name of the server object.

description = a description of the server object.

description2 = what appears in the list of registered file types in the Registration Info Editor.

format = the format of the file.

Use the Registration Info Editor to change these entries.

[Windows Help] section

File: WIN.INI

This section contains settings that specify the size, placement, and appearance of Help windows.

You will find entries in this section for:

M_WindowPosition
H_WindowPosition
A_WindowPosition
C_WindowPosition
IFJumpColor
IFPopupColor
JumpColor
MacroColor
PopupColor

[sound] section

File: WIN.INI

This section contains entries for system events that support sound. The [sound] section can contain one or more occurrences of the following entry:

systemevent=filename,description

Where:

system event = the defined system event

filename = the name of the sound file

description = a description of the system event

Use the Sound icon in Control Panel to change this entry.

[printerPorts] section

File: WIN.INI

This section contains a list of the active and inactive printers that can be accessed by Windows, their associated communications ports, and their timeout values. The [printerPorts] section can contain one or more entries in the following format:

device=driver, port, DeviceTimeout, RetryTimeout [, other ports ...]

Where :

device = the name of the device

driver = the driver filename

port = the port the device is connected to

DeviceTimeout = seconds that Print Manager will wait for the device to be turned on

RetryTimeout = seconds before attempted transmission retries

Use the Printers icon in Control Panel to change this entry.

[devices] section

File: WIN.INI

This section lists the active printers, but is only necessary for compatibility with Windows 2.x applications. The [devices] section can contain one or more entries in the following format:

device=driver, port [, other ports ...]

Where :

device = the name of the device

driver = the driver filename

port = the port the device is connected to

Use the Printers icon in Control Panel to change this entry.

[programs] section

File: WIN.INI

This section identifies the paths that Windows will search to start an application. This is in addition to the path in your AUTOEXEC.BAT file. The [programs] section can contain one or more of the following entries:

program file=drive:directory\program file

Where:

program file = the name of the program you are trying to run

drive = the drive on which the program is located

directory = the directory in which the program is located

program file = the name of the program you are trying to run

To change this entry, complete the dialog box in File Manager that prompts for the path when it can't find the application.

[colors] section

File: WIN.INI

This section defines the colors of the Windows display. It can contain entries in the following format:

component=red-value green-value blue-value

Where:

component = one of the following:

- ActiveBorder
- ActiveTitle
- AppWorkspace
- Background
- ButtonFace
- ButtonShadow
- ButtonText
- GrayText
- Hilight
- HilightText
- InactiveBorder
- InactiveTitle
- InactiveTitleText
- Menu
- MenuText
- ScrollBar
- TitleText
- Window
- WindowFrame
- WindowText

Beep

Beep=*Yes* | *No*

Default : Yes

Purpose : If Yes, a beep sounds when you do something illegal.

Use Sound icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

BorderWidth

BorderWidth=*number*

Default : 3

Purpose : Sets sizable window border.

Use Desktop icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

CoolSwitch

CoolSwitch=*0* | *1*

Default : 1

Purpose : Turns fast task switching on (1) or off (0).

Use Desktop icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

CursorBlinkRate

CursorBlinkRate=*milliseconds*

Default : 530

Purpose : How much time between blinks of the cursor

Use Desktop icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

DefaultQueueSize

DefaultQueueSize=*number*

Default : 8

Purpose : The number of messages an application's queue can hold.

Edit WIN.INI to change this.

File : WIN.INI Section : [windows]

Device

Device=o*utput-device-name, device-driver, port-connection*
Purpose : Defines the default printer.
Use Printers icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

DeviceNotSelectedTimeout

DeviceNotSelectedTimeout=*seconds*

Default : 15

Purpose : How many seconds Windows will wait for a device to be switched on.

Use Printers icon in Control Panel to change this value.

File : WIN.INI   Section : [windows]

Documents

Documents=*extensions*

Default : none

Purpose : Defines extensions for documents not associated with an application

Edit WIN.INI to change this entry.

File : WIN.INI Section : [windows]

DosPrint

DosPrint=*Yes* | *No*

Default : No

If Yes, Windows uses MS-DOS interrupts when printing.

If No, Windows sends output directly to the printer port.

Use Printers icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

DoubleClickHeight

DoubleClickHeight=*pixels*

Default : 4

Purpose : How much the mouse can move vertically between clicks in a double-click.

Edit WIN.INI to change this entry.

See also : DoubleClickSpeed, DoubleClickWidth

File : WIN.INI Section : [windows]

DoubleClickWidth

DoubleClickWidth=*pixels*
Default : 4
Purpose : How much the mouse can move horizontally between clicks in a double-click.
Edit WIN.INI to change this entry.
See also : DoubleClickHeight, DoubleClickSpeed

File : WIN.INI Section : [windows]

DoubleClickSpeed

DoubleClickSpeed=*milliseconds*

Default : 452

Purpose : Amount of time allowed between clicks in a double-click

Use Mouse icon in Control Panel to change this value.

See also : DoubleClickHeight, DoubleClickWidth

File : WIN.INI Section : [windows]

KeyboardDelay

KeyboardDelay=*milliseconds*

Default : 2

Purpose : How much time elapses before key starts repeating while you hold down key.

Use Keyboard icon in Control Panel to change this value.

See also : KeyboardSpeed

File : WIN.INI		Section : [windows]

KeyboardSpeed

KeyboardSpeed=*milliseconds*

Default : 31

Purpose : How much time elapses between repeats while you hold down a key.

Use Keyboard icon in Control Panel to change this value.

See also : KeyboardDelay

File : WIN.INI   Section : [windows]

Load

Load=*filename(s)*

Purpose : Applications to start minimized every time Windows starts.

Add an application to Startup group and set Minimize On Use to set this

File : WIN.INI Section : [windows]

MenuDropAlignment

MenuDropAlignment=*0* | *1*

Default : 0

Purpose : Specifies whether menus open right-aligned (1) or left-aligned (0)

Edit WIN.INI to change this entry.

File : WIN.INI   Section : [windows]

MenuShowDelay

MenuShowDelay=*milliseconds*

Default : 0 for 386 systems; 400 for 286 systems

Purpose : Specifies how long to wait before displaying a cascading menu.

Edit WIN.INI to change this entry.

File : WIN.INI Section : [windows]

MouseSpeed

MouseSpeed=*0* **|** *1* **|** *2*

Default : 1

Purpose : Sets relationship between mouse and cursor movement if either MouseThreshold1 or MouseThreshold2 is exceeded.

0 = no acceleration.

1 = 2 times normal speed when MouseThreshold1 is exceeded.

2 = 2 times normal speed when MouseThreshold1 is exceeded. 4 times normal speed when MouseThreshold2 is exceeded.

Use Mouse icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

MouseThreshold1

MouseThreshold1=*pixels*

Default : 5

Purpose : *Sets maximum number of pixels that the mouse can move before Windows speeds up cursor movement.*

Use Mouse icon in Control Panel to change this value.

See also: MouseSpeed, MouseThreshold2

File : WIN.INI Section : [windows]

MouseThreshold2

MouseThreshold2=*pixels*

Default : 10

Purpose : *Sets maximum number of pixels that the mouse can move before Windows speeds up cursor movement.*

Use Mouse icon in Control Panel to change this value.

See also: MouseSpeed, MouseThreshold2

File : WIN.INI Section : [windows]

MouseTrails

MouseTrails=*number*

Default : 0

Purpose : The number of pointers trailing behind the cursor.

Use Mouse icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

NetWarn

NetWarn=*0* | *1*

Default : 1

Purpose : Specifies whether Windows gives a warning when network not available.

1 = Display a warning if network Windows expects a network but it is not available.

0 = No warning.

Use Network icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

NullPort

NullPort=*string*

Default : "None"

Purpose : Specifies the name for a null port. Used for installed devices that aren't connected to a port. Edit WIN.INI to change this setting.

File : WIN.INI Section : [windows]

Programs

Programs=*extensions*

Default : com exe bat pif

Purpose : Defines which files Windows regards as applications.

Edit WIN.INI to change this entry.

File : WIN.INI Section : [windows]

Run

Run=*filename(s)*

Purpose : Windows starts these applications on startup.

Add or remove the application to the Startup group to change this entry.

File : WIN.INI Section : [windows]

ScreenSaveActive

ScreenSaveActive=*0* | *1*

Default : 0

Purpose : Specifies whether is a screen save is turned on.

1 = Screen saver is turned on.

0 = Screen save is turned off.

Use Desktop icon in Control Panel to change this value.

See also : ScreenSaveTimeOut

File : WIN.INI Section : [windows]

ScreenSaveTimeOut

ScreenSaveTimeOut=*seconds*

Default : 120

Purpose : Specifies amount of idle time before Windows starts the screen saver

Use Desktop icon in Control Panel to change this value.

See also : ScreenSaveActive

File : WIN.INI Section : [windows]

Spooler

Spooler=*Yes* | *No*

Default : Yes

Purpose : Specifies whether print output is sent through Print Manager.

Yes = Goes through Print Manager.

No = Does not go through Print Manager.

Use Printers icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

SwapMouseButtons

SwapMouseButtons=*0* | *1*

Default : 0

Purpose : Specifies whether Windows swaps left and right mouse buttons.

0 = buttons not swapped.

1 = buttons are swapped.

Use Mouse icon in Control Panel to change this value.

File : WIN.INI Section : [windows]

TransmissionRetryTimeout

TransmissionRetryTimeout=*seconds*

Default : 90 for PostScript printers; 45 for all other printers

Purpose : The system default time that Windows uses for attempted transmission retries.

Use Printers icon in Control Panel to change this value for a particular printer.

Edit WIN.INI to change the default.

File : WIN.INI Section : [windows]

GridGranularity

GridGranularity=*number*

Default : 0

Purpose : Specifies the size of the grid used to position windows on the screen.

Use Desktop icon in Control Panel to change this value.

File : WIN.INI   Section : [Desktop]

IconSpacing

IconSpacing=*pixels*

Default : 77

Purpose : The number of pixels between icons horizontally

Use Desktop icon in Control Panel to change this value.

File : WIN.INI Section : [Desktop]

IconTitleFaceName

IconTitleFaceName=*fontname*

Default : MS Sans Serif

Purpose : The font used to display icon titles.

Edit WIN.INI to change this entry.

File : WIN.INI   Section : [Desktop]

IconTitleSize

IconTitleSize=*number*

Default : 8

Purpose : The size of the font used to display icon titles.

Edit WIN.INI to change this value.

File : WIN.INI Section : [Desktop]

IconTitleWrap

IconTitleWrap=*0* | *1*

Default : 1

Purpose : Specifies whether to wrap icon titles.

0 = No icon title wrapping

1 = Allow icon title wrapping and increase vertical spacing by 3 lines

Edit WIN.INI to change this value.

File : WIN.INI Section : [Desktop]

IconVerticalSpacing

IconVerticalSpacing=*pixels*
Default : based on title font and your display type
Purpose : Number of pixels between icons vertically.
Edit WIN.INI to change this value.

File : WIN.INI Section : [Desktop]

Pattern

Pattern=*b1 b2 b3 b4 b5 b6 b7 b8*

Default : "(None)"

Purpose : The pattern used for the screen background

Use Desktop icon in Control Panel to change this value.

File : WIN.INI Section : [Desktop]

TileWallpaper

TileWallpaper=*0* | *1*

Default : 0

Purpose : Specifies whether wallpaper is tiled or centered.

0 = centered.

1 = tiled.

Use Desktop icon in Control Panel to change this value.

File : WIN.INI Section : [Desktop]

Wallpaper

Wallpaper=*bitmap-filename*

Default : "(None)"

Purpose : The bitmap filename for the screen background

Use Desktop icon in Control Panel to change this value.

File : WIN.INI Section : [Desktop]

extension (file associations)

extension=command-line

Where:

extension = a filename extension of 1 to 3 characters.

command-line = a normal MS-DOS command line. (The application filename followed by any required parameters.

Purpose : To associate a file extension with its program.

Choose File-Associate in File Manager to change these entries.

File : WIN.INI Section : [Extensions]

iCountry

iCountry=*country number*

Default : 1.

Purpose : Specifies the country code. (The country's international telephone code except for Canada, which is 2).

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

iCurrDigits

iCurrDigits=*number*

Default : 2

Purpose : The number of digits after the decimal separator in currency.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

iCurrency

iCurrency=*number*

Default : 0

Purpose : The currency format.

0 = $2

1 = 2$

2 = $ 2

3 = 2

Note : The actual currency symbol is set by sCurrency.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

iDate

iDate=*number*

Default : 0

Purpose : Specifies a Windows 2.x compatible date format.

0 = 12/31/90

1 = 31/12/90

2 = 90/12/31

Note : The date separator is set by sShortDate

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

iDigits

iDigits=*number*

Default : 2

Purpose : The number of digits to display after the decimal in numbers.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

iLZero

ilZero=*0* | *1*

Default : 0

Purpose : Specifies whether to put leading zeroes in decimal numbers.

0 = .7

1 = 0.7

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

iMeasure

iMeasure=*0* | *1*

Default : 1

Purpose : Whether to use English or metric measurements.

0 = metric

1 = English

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

iNegCurr

iNegCurr=*number*

Default : 0

Purpose : Specifies one of the following number formats:

0 = ($1)   1 = -$1   2 = $-1   3 = $1-   4 = (1$)
5 = -1$ 6 = 1-$ 7 = 1$-   8 = -1 $ 9 = -$ 1
10 = $ 1-

Use the International icon in Control Panel to change this value.

File : WIN.INI   Section : [intl]

iTime

iTime = number

Default : 0

Purpose : Specifies whether to format time using a 12 or 24 hour clock.

0 = 1:00

1 = 13:00

Note : The time separator is specified by sTime

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

iTLZero

iTLZero=number

Default : 0

Purpose : Whether to put leading zeroes in time.

0 = 2:30

1 = 02:30

Note : The time separator is specified by the sTime

File : WIN.INI   Section : [intl]

s1159

s1159=string

Default : AM

Purpose : The string that follows times before noon in 12-hour format.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

s2359

s2359=string

Default : PM

Purpose : The string that follows times after noon in the 12-hour format and at all times in the 24-hour format.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sCountry

sCountry=string

Default : United States

Purpose : The name of your country

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sCurrency

sCurrency=string

Default : $

Purpose : The currency symbol.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sDecimal

sDecimal=*string*

Default : . (a period)

Purpose : Specifies the decimal separator.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sLanguage

sLanguage=string

Default : enu

Purpose : The language you want to work in specified by one of the following:

dan = Danish
deu = German
eng = International English
enu = U. S. English
esn = Modern Spanish
esp = Castilian Spanish
fin = Finnish
fra = French
frc = French Canadian
isl = Icelandic
ita = Italian
nld = Dutch
nor = Norwegian
ptg = Portugese
sve = Swedish

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sLis

sLis=string

Default : , (a comma)

Purpose : The character used to separate items in a list.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sLongDate

sLongDate=format

Default : dddd, MMMM dd, yyyy

Purpose : Specifies your choices for the long date format. Values are:

d = Day (1-31)
dd = Day (01-31)
ddd = Day (Mon-Sun)
dddd = Day (Monday-Sunday)
M = Month (1-12)
MM = Month (01-12)
MMM = Month (Jan-Dec)
MMMM = Month (January-December)
yy = Year (00-99)
yyyy = Year (1900-2040)

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sShortDate

sShortDate=format

Default : M/d/yy

Purpose : Specifies your choices for the short date format. Values are:

d = Day (1-31)
dd = Day (01-31)
ddd = Day (Mon-Sun)
dddd = Day (Monday-Sunday)
M = Month (1-12)
MM = Month (01-12)
MMM = Month (Jan-Dec)
MMMM = Month (January-December)
yy = Year (00-99)
yyyy = Year (1900-2040)

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sThousand

sThousand=*string*

Default : , (a comma)

Purpose : The symbol used to separate thousands.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

sTime

sTime=*string*

Default : : (a colon)

Purpose : The character used to separate hours, minutes, and seconds.

Use the International icon in Control Panel to change this value.

File : WIN.INI Section : [intl]

OutlineThreshold

OutlineThreshold=*number-of-pels-per-em*

Default : 256

Purpose : The number of pels-per-em at which Windows will render TrueType fonts as outline fonts instead of bitmap fonts.

Edit WIN.INI to change this value.

Caution : Do not specify a value over 300

File : WIN.INI Section : [TrueType]

TTEnable

TTEnable=*0* | *1*

Default : 1

Purpose : Specifies whether TrueType is enabled.

0 = TrueType fonts not available to applications

1 = TrueType fonts are available to applications.

Use the Fonts icon in Control Panel to change this value.

File : WIN.INI Section : [TrueType]

TTIfCollisions

TTIfCollisions=*0* | *1*

Default : 0

Purpose : Specifies whether to use TrueType fonts if both types of fonts have the same name

0 = Use TrueType

1 = Use bit mapped font

Edit WIN.INI to change this section.

File : WIN.INI Section : [TrueType]

TTOnly

TTOnly=*0* **|** *1*

Default : 0

Purpose : Specifies whether to make only TrueType fonts available to applications.

0 = All fonts are made available.

1 = Only TrueType fonts are mad available.

Use the Fonts icon in Control Panel to change this value.

File : WIN.INI Section : [TrueType]

drive

drive=network-server-and-share

Purpose : Shows the network connections that are restored each time Windows is started. Use the Disk - Network Connections command in File Manager to change this entry.

File : WIN.INI Section : [network]

InRestoreNetConnect

InRestoreNetConnect=*0* | *1*

Default : 1

Purpose : Specifies whether to reconnect to the network servers you were connected to in your previous Windows session.

0 = Do not reconnect.

1 = Reconnect.

Use the Networks option from Control Panel to change this entry.

File : WIN.INI Section : [network]

port

port=*network-printer-path*

Purpose : Defines path and port for a network printer.

Use the Printers icon in the Control Panel to change this entry.

File : WIN.INI Section : [network]

M_WindowPosition

M_WindowPosition=*number1,number2,number3,number4,number5*

Purpose : Sets the default size and placement of the main Help window.

number1 = the upper left corner horizontally

number2 = the upper left corner vertically

number3 = the width

number4 = the height

number5 = (0 **|** 1) 0-use defaults   1-maximize the window

Move or size the main Help window to change this entry.

File : WIN.INI   Section : [Windows Help]

H_WindowPosition

H_WindowPosition=*number1,number2,number3,number4,number5*

Purpose : Sets the default size and placement of the History window.

number1 = the upper left corner horizontally

number2 = the upper left corner vertically

number3 = the width

number4 = the height

number5 = 0

Move the History dialog to change this entry.

File : WIN.INI Section : [Windows Help]

A_WindowPosition

A_WindowPosition=*number1,number2,number3,number4,number5*

Purpose : Sets the default size and placement of the Annotate window.

number1 = the upper left corner horizontally

number2 = the upper left corner vertically

number3 = the width

number4 = the height

number5 = 0

Move the Annotate dialog to change this entry.

File : WIN.INI Section : [Windows Help]

C_WindowPosition

C_WindowPosition=*number1,number2,number3,number4,number5*

Purpose : Sets the default size and placement of the Copy window.

number1 = the upper left corner horizontally

number2 = the upper left corner vertically

number3 = the width

number4 = the height

number5 = 0

Move the Copy dialog to change this entry.

File : WIN.INI Section : [Windows Help]

IFJumpColor

IFJumpColor=red-value green-value blue-value

Purpose : Specifies the color of text that makes jumps to another section of the Help file.

Edit WIN.INI to change this setting.

File : WIN.INI Section : [Windows Help]

IFPopupColor

IFPopupColor=red-value green-value blue-value

Purpose : Specifies the color of text that displays a popup panel for further Help.

Edit WIN.INI to change this setting.

File : WIN.INI Section : [Windows Help]

JumpColor

JumpColor=red-value green-value blue-value

Purpose : Specifies the color of text that makes jumps to another section of the Help file.

Edit WIN.INI to change this setting.

File : WIN.INI Section : [Windows Help]

MacroColor

MacroColor=red-value green-value blue-value

Purpose : Specifies the color of text that runs a Help macro.

Edit WIN.INI to change this setting.

File : WIN.INI Section : [Windows Help]

PopupColor

PopupColor=red-value green-value blue-value

Purpose : Specifies the color of text that displays a popup panel for further Help.

Edit WIN.INI to change this setting.

File : WIN.INI Section : [Windows Help]

ActiveBorder

ActiveBorder=*red-value green-value blue-value*
Purpose : Sets the color of the border of the active window.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

ActiveTitle

ActiveTitle=*red-value green-value blue-value*
Purpose : Sets the color of the title bar of the active window.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

AppWorkspace

AppWorkspace=*red-value green-value blue-value*
Purpose : Sets the color of the application workspace for Windows applications.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

Background

Background=*red-value green-value blue-value*
Purpose : Sets the color of the desktop background.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

ButtonFace

ButtonFace=*red-value green-value blue-value*
Purpose : Sets the color of the button faces.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

ButtonShadow

ButtonShadow=*red-value green-value blue-value*
Purpose : Sets the color of the shadow behind buttons.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

ButtonText

ButtonText=*red-value green-value blue-value*
Purpose : Sets the color of the text on buttons.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

GrayText

GrayText=*red-value green-value blue-value*
Purpose : Sets the color of the text that is dimmed.

Hilight

Hilight=*red-value green-value blue-value*
Purpose : Sets the color of the background of highlighted text.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

HilightText

Hilight=*red-value green-value blue-value*
Purpose : Sets the color of hilighted text.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

InactiveBorder

InactiveBorder=*red-value green-value blue-value*
Purpose : Sets the color of the border of an inactive window.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

InactiveTitle

InactiveTitle=*red-value green-value blue-value*
Purpose : Sets the color of the title bar of an inactive window.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

InactiveTitleText

InactiveTitleText=*red-value green-value blue-value*
Purpose : Sets the color of the text in an inactive title bar.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

Menu

Menu=*red-value green-value blue-value*
Purpose : Sets the color of the background of menus.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

MenuText

MenuText=*red-value green-value blue-value*
Purpose : Sets the color of the text in a menu.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

ScrollBar

ScrollBar=*red-value green-value blue-value*
Purpose : Sets the color of the scrollbars.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

TitleText

TitleText=*red-value green-value blue-value*
Purpose : Sets the color of the text in the title bar.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

Window

Window=*red-value green-value blue-value*
Purpose : Sets the color of the workspace in a window.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

WindowFrame

WindowFrame=*red-value green-value blue-value*
Purpose : Sets the color of the window frame.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

WindowText

WindowText=*red-value green-value blue-value*
Purpose : Sets the color of the text in a window.

Use the Color icon in the Control Panel to change this entry.

File : WIN.INI Section : [colors]

SYSTEM.INI

Windows Setup creates the SYSTEM.INI file when you install Windows. It contains global information that Windows uses on startup. The following sections may appear in SYSTEM.INI:

[boot]

[boot.description]

[drivers]

[keyboard]

[mci]

[NonWindowsApp]

[standard]

[386enh]

[boot]

File: SYSTEM.INI

The [boot] section contains a list of the drivers and Windows modules that Windows uses to configure itself each time you start it.

The [boot] section can contain the following entries:

286grabber

386grabber

CachedFileHandles

comm.drv

display.drv

drivers

fixedfon.fon

fonts.fon

keyboard.drv

language.dll

mouse.drv

network.drv

oemfonts.fon

shell

sound.drv

system.drv

TaskMan.Exe

286grabber

286grabber=filename

This entry specifies the filename of the grabber, which is the device driver   that makes a non-Windows application visible when you run Windows in standard mode. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the Display setting. If you are installing a device driver that is not included with Windows, exit Windows and run Setup from MS-DOS.

File: SYSTEM.INI   Section: [boot]

386grabber

386grabber=filename

This entry specifies the filename of the grabber, which is the device driver that makes a non-Windows application visible when you run Windows in 386 enhanced mode. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the Display setting. If you are installing a device driver that is not included with Windows, exit Windows and run Setup from MS-DOS.

File: SYSTEM.INI Section: [boot]

CachedFileHandles

CachedFileHandles=number

This entry specifies the number of the most recently-used executable files (.EXE) and dynamic-link library files (.DLL) that can remain open. Windows keeps these files open so that they can be accessed quickly, ensuring optimal performance. However, some networks have a limit on the number of files that can be open on a server at a time. If you have problems running Windows from a network server, use a lower number for this value. Valid numbers are 2 through 12. The default is 12. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [boot]

comm.drv

comm.drv=filename

This entry specifies the filename of the serial communications driver you are using. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [boot]

display.drv

display.drv=filename

This entry specifies the filename of the display driver you are using. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window. If you are installing a device driver that is not included with Windows, exit Windows and run Setup from MS-DOS.

File: SYSTEM.INI Section: [boot]

drivers

drivers=filename | aliasname

This entry specifies the filenames or aliases of the installable drivers that load when Windows is started. An installable driver is a dynamic-link library that Windows treats like a device driver. If the drivers you want to use include parameters, you must specify them by alias name as defined in the [drivers] section of SYSTEM.INI, where you can list several filenames or aliases. The default is none (blank). Most Setup programs for Windows applications add these settings to the SYSTEM.INI file when installing drivers, so you shouldnt need to change the values. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [boot]

fixedfon.fon

fixedfon.fon=filename

This entry specifies the filename of the fixed system font used by Windows version 2.x applications. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window. This entry also changes when you change the Display setting in Setup.

File: SYSTEM.INI Section: [boot]

fonts.fon

fonts.fon=filename

This entry specifies the filename of the proportionately spaced system font used by Windows 3.1. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window. This entry also changes when you change the Display setting in Setup.

File: SYSTEM.INI Section: [boot]

keyboard.drv

keyboard.drv=filename

This entry specifies the filename of the keyboard driver you are using. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the setting for the Keyboard option.

File: SYSTEM.INI   Section: [boot]

language.dll

language.dll=library-name

This entry specifies the name of the dynamic-link library that supplies language-specific functions. If no language library is specified, Windows uses the built-in U.S. English library. The default is none (blank). To change this entry, choose the International icon from Control Panel and change the setting for the Language option.

File: SYSTEM.INI Section: [boot]

mouse.drv

mouse.drv=filename

This entry specifies the filename of the mouse driver you are using. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the setting for the Mouse option.

File: SYSTEM.INI Section: [boot]

network.drv

network.drv=filename

This entry specifies the filename of the network driver you are using. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the setting for the Network option.

File: SYSTEM.INI Section: [boot]

oemfonts.fon

oemfonts.fon=filename

This entry supplies the name of the font file for the OEM character set, which is associated with the Display setting. The default is none (blank). To change this entry, exit Windows and run Setup from MS-DOS. This entry also changes when you change the Codepage or Display setting in Setup.

File: SYSTEM.INI Section: [boot]

shell

shell=filename

This entry specifies the Windows shell program that runs when you start Windows. Unless you change this entry, Program Manager will run when you start Windows. The default is none (blank). Setup initializes this value as progman.exe. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [boot]

sound.drv

sound.drv=filename

This entry specifies the filename of the system sound driver you are using. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [boot]

system.drv

system.drv=filename

This entry specifies the filename of the system hardware driver you are using. The default is none (blank). To change this entry, exit Windows and run Setup from MS-DOS, then choose a new System setting.

File: SYSTEM.INI Section: [boot]

TaskMan.Exe

TaskMan.Exe=filename

This entry specifies the task-switching application that appears when you press CTRL+ESC. The default is taskman.exe. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [boot]

[boot.description]

File: SYSTEM.INI

The [boot.description] section is a list of strings used to describe the devices you can change when you run Setup. There is no reason to change these entries. If you do change values, you cannot use Windows Setup to update drivers to newer versions.

[drivers] Section

File: SYSTEM.INI

This section contains a list of aliases (or names) assigned to installable driver files. The [drivers] section can contain this entry:

alias=driver filename[parameters]

This entry assigns an alias name to an installable driver and specifies any parameters used by the driver. This entry is associated with the drivers= entry in the [boot] section of SYSTEM.INI. If the driver includes parameters, you must assign an alias to it and then specify the driver by alias name in the [boot] section. You can define multiple aliases by specifying multiple alias lines in this entry. The default is none. Most Setup programs for Windows applications add these settings to SYSTEM.INI when installing drivers, so you shouldnt need to change the values. To change this entry, choose the Drivers icon in Control Panel or use the installation program provided by the manufacturer.

[keyboard] Section

File: SYSTEM.INI

The [keyboard] section provides information about the keyboard.

Important:All entries in this section are required. If you modify or delete one of these entries, Windows will not operate properly. There are no built-in default values for these entries; Setup assigns values based on the system configuration.

The [keyboard] section can contain the following entries:

keyboard.dll
oemansi.bin
subtype
type

keyboard.dll

keyboard.dll=filename

This entry specifies the name of a dynamic-link library that defines the layout for non-U.S. keyboards and keyboards not compatible with IBM-compatible systems. This entry is required for all keyboards except these U.S. keyboards: IBM XT, PC/AT, or enhanced; AT&T type 301 or 302; and Olivetti 83-key. To change this entry, choose the Windows Setup icon from the Main Group window or the International icon from Control Panel.

File: SYSTEM.INI   Section: [keyboard]

oemansi.bin

oemansi.bin=filename

This entry specifies the name of a file that defines OEM/ANSI code-page translation tables for systems not using code page 437 (the U.S. OEM character set). To change this entry, exit Windows and run Setup from MS-DOS. This setting also changes when you change the Codepage or Display settings.

File: SYSTEM.INI Section: [keyboard]

subtype

subtype=number

This entry distinguishes, for some drivers, special features for keyboards that otherwise have identical layouts. This value can also be used by other drivers. To change the subtype= entry, choose the Windows Setup icon from the Main Group window. See the type= entry for information about type values. These values are defined for subtype=.

Type value	Subtype value	Meaning
1	2	Olivetti M24 83-key or AT&T 6300 type 301 83-key.
1	4	AT&T type 302 sometimes used on the 6300 Plus.
2	1	Olivetti 102-key ICO used on M24 systems.

File: SYSTEM.INI Section: [keyboard]

type

type=number

This entry specifies the keyboard type. It can be any one of these values.

Value	Meaning
1	IBM PC or XT compatible (83 keys)
2	Olivetti 102-key ICO
3	IBM AT compatible (84 or 86 keys)
4	IBM compatible, enhanced (101 or 102 keys)

If this entry is blank or missing, the driver selects a default type. For IBM-compatible keyboards (using the KEYBOARD.DRV driver), the default type is determined by the BIOS. To change this entry, choose the Windows Setup icon from the Main Group window.

File: SYSTEM.INI Section: [keyboard]

[mci] section

The [mci] section contains a list of the drivers that use the Media Control Interface (MCI) to play media files. These drivers are installed automatically when you run Setup. To change these values, choose the Drivers icon in Control Panel.

[NonWindowsApp] Section

The [NonWindowsApp] section contains entries that affect the performance of non-Windows applications. This section can contain the following entries:

CommandEnvSize

DisablePositionSave

FontChangeEnable

GlobalHeapSize

LocalTSRs

MouseInDosBox

NetAsynchSwitching

ScreenLines

SwapDisk

CommandEnvSize

CommandEnvSize=bytes

This entry specifies the size of the COMMAND.COM environment. Since running batch files with the extension .BAT starts COMMAND.COM, this setting also applies to batch files. The value for this setting must be either 0 or between 160 and 32768. A value of 0 disables this setting. If the value is not valid, it will be rounded up to 160 or down to 32768. If the value is less than the current size of the actual environment, this setting will be disabled, as if it were set to 0. If you specified the environment size in a PIF for COMMAND.COM, the PIF setting overrides this setting.   The default is 0 with MS-DOS versions earlier than 3.2. Otherwise, the value is the /e: parameter in the shell= line in CONFIG.SYS.   To change this value, you must edit SYSTEM.INI.

File: SYSTEM.INI   Section: [NonWindowsApp]

DisablePositionSave

DisablePositionSave=Boolean

When this entry is 0, the position and font used in a non-Windows application is saved in the DOSAPP.INI file when you quit the application. If this entry is 1, any settings that werent saved previously in DOSAPP.INI will not be saved. If this entry is 1, the setting can be overridden for each non-Windows application by selecting the Save Settings On Exit checkbox in the Fonts dialog box. The default is 0. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [NonWindowsApp]

FontChangeEnable

FontChangeEnable=Boolean

This entry, if 1, provides the ability to change the fonts when running non-Windows applications in a window on systems that use version 3.0 of the grabbers (usually in version 3.0 display drivers). Windows version 3.1 of the video grabbers (used in version 3.1 display drivers) include built-in support for changing fonts when running non-Windows applications in a window. If you are using a 3.0 grabber that has not been updated to include the ability to change fonts and you want to use this feature, set this value to 1. However, when this entry is 1, your screen may lose characters and the cursor may change size and position slightly. The default is 1 on systems that use Windows 3.1 grabbers, and 0 on systems that use Windows 3.0 grabbers. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [NonWindowsApp]

GlobalHeapSize

GlobalHeapSize=kilobytes

This entry specifies the size of a buffer in conventional memory that Windows allocates when running in standard mode. This buffer can be used to share information among all non-Windows applications that are started from Windows. This setting has no effect when Windows is running in 386 enhanced mode. The default is 0. To change this entry, you must edit SYSTEM.INI. (You should never need to change this setting.)

File: SYSTEM.INI Section: [NonWindowsApp]

LocalTSRs

LocalTSRs=list-of-TSR-applications

This entry specifies which terminate-and-stay-resident (TSR) programs work properly if they are copied to each instance of a virtual machine. When you start Windows, it detects any TSR programs that are currently running. If the TSR is on the LocalTSRs= list, Windows will place a copy of the TSR in each virtual machine you run. Many TSRs will not run properly if they are added to this list. Make sure your TSR is fully compatible with Windows and can be copied to a virtual machine before adding it to the list. The default is dosedit, pced, ced. To change this value, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [NonWindowsApp]

MouseInDosBox

MouseInDosBox=Boolean

This entry specifies whether the mouse is supported when running a non-Windows application in a window. Mouse support for non-Windows applications running in a window is available automatically if you are using a Windows 3.1 version of the grabbers. If you are using a Windows 3.0 version of the grabbers and you want mouse support, enable this setting. If you do not want mouse support, disable this setting.   The default is 1 if an MS-DOS mouse driver is loaded that has the extension .COM or .SYS and that supports using a mouse with a non-Windows application. Otherwise, the default is 0. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI   Section: [NonWindowsApp]

NetAsynchSwitching

NetAsynchSwitching=0 | 1

This entry indicates whether you can switch away from an application running in Windows standard mode after it has made an asynchronous network BIOS call. The default value of 0 specifies that such task switching is not possible. Switching away from some applications that make these calls might cause the system to fail. Once Windows detects an asynchronous NetBIOS call, it will not allow switching away from the application even if no more of these calls are made. Set this value to 1 if you are sure the applications you use will not receive network messages while you are switched away from them. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [NonWindowsApp]

ScreenLines

ScreenLines=number
This entry specifies the number of lines displayed on the screen when a non-Windows application runs. An application that specifies a different screen mode can override this entry. The default is 25. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [NonWindowsApp]

SwapDisk

SwapDisk=drive:directory

This entry provides the name of the disk drive and directory to which Windows standard mode swaps non-Windows applications. The default is the directory pointed to by the TEMP environment variable; if there is no TEMP variable, then the default is the boot directory of your first hard disk (usually C:). If your system does not have a hard disk, the default is the root directory on the first floppy drive (usually A:). To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI   Section: [NonWindowsApp]

[standard] Section

The [standard] section contains entries that are specific to running Windows in standard mode. This section can contain the following entries:

FasterModeSwitch
Int28Filter
MouseSyncTime
NetHeapSize
PadCodeSegments
Stacks
StackSize

FasterModeSwitch

FasterModeSwitch=0 | 1

This entry, if set to 1, causes Windows standard mode to use a faster method of switching from protected to real mode on many 80286-based computers. When this entry is 1, Windows responds more quickly to hardware interrupts, allowing better throughput for interrupt-intensive applications, such as communications applications. Also, you should set this entry to 1 if you are using a Zenith Z-248 system and are losing characters while typing, or if you are using an Olivetti M-250-E and lose control of the mouse. The default is 0. To change this entry, you must edit SYSTEM.INI.

Note:The value for FasterModeSwitch= has no effect on 80386-based computers. Some early IBM-AT and compatible computers do not have the BIOS support necessary to use this setting. Setting this entry to 1 on these computers may cause them to hang when starting Windows.

File: SYSTEM.INI Section: [standard]

Int28Filter

Int28Filter=number

This entry specifies the percentage of INT 28h interrupts, generated when the system is idle, that are visible to software loaded before Windows. Windows will reflect every nth interrupt, where n is the value of this entry. For example, a value of 1 reflects every INT 28h interrupt, a value of 2 reflects every second INT 28h interrupt, a value of 3 reflects every third INT 28h interrupt, and so on. Increasing this value might improve Windows performance, but may interfere with some memory-resident software such as a network.

Set this value to 0 to prevent INT 28h interrupts. But setting this value too low adds to system overhead that might interfere with communications applications. The default is 10. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [standard]

MouseSyncTime

MouseSyncTime=milliseconds

This entry specifies the number of milliseconds that can elapse between mouse data bytes before Windows running in standard mode assumes that a mouse data packet is complete. This setting only affects Windows standard mode on computers with an IBM PS/2 mouse interface. The default is 500. To change this value, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [standard]

NetHeapSize

NetHeapSize=kilobytes
This entry specifies the size of the data-transfer buffers that Windows standard mode allocates in conventional memory for transferring data over a network. If an application is not running correctly, your network may require a larger buffer than the default. Increasing this value will decrease the amount of memory available to applications. If no network software is running, this entry is ignored and no memory is allocated. The default is 8. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [standard]

PadCodeSegments

PadCodeSegments=0 | 1

Setting the value of this entry to 1 causes Windows Kernel to pad code segments with 16 bytes. This prevents the last instruction in the segment from being too close to the segment limit for 80286 C2 stepping. Set this value to 1 only for this 80286 stepping. Unfortunately, there is no easy way of telling what stepping a 80286 chip is. Try setting this value to 1 if your 80286 system hangs in standard mode. The default is 0. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [standard]

Stacks

Stacks=number
This entry specifies the number of interrupt reflector stacks used by the standard mode MS-DOS Extender (DOSX) to map an MS-DOS or BIOS API from real mode to protected mode. If you receive a Standard Mode: Stack Overflow message, try increasing this number. You can specify a number between 8 and 64. The default is 12. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [standard]

StackSize

StackSize=kilobytes
This entry specifies the size of the interrupt reflector stacks used by the standard mode MS-DOS Extender. The default is 384. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry.)

File: SYSTEM.INI Section: [standard]

[386enh] Section

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.

This section can contain the following entries:

32BitDiskAccess

A20EnableCount

AllEMSLocked

AllVMsExclusive

AllXMSLocked

AltKeyDelay

AltPasteDelay

AutoRestoreScreen

BkGndNotifyAtPFault

CGA40WOA.FON

CGA80WOA.FON

CGANoSnow

COM1AutoAssign

COM2AutoAssign

COM3AutoAssign

COM4AutoAssign

COM1Base

COM2Base

COM3Base

COM4Base

COMBoostTime

COM1Buffer

COM2Buffer

COM3Buffer

COM4Buffer

COMdrv30

COM1FIFO

COM2FIFO

COM3FIFO

COM4FIFO

COMIrqSharing

COM1Protocol

COM2Protocol

COM3Protocol

COM4Protocol

Device

Display

DMABufferIn1MB

DMABufferSize

DOSPromptExitInstruc

DualDisplay

EBIOS

EGA40WOA.FON

EGA80WOA.FON

EISADMA

EMMExclude

EMMInclude

EMMPageFrame

EMMSize

FileSysChange

Global

HardDiskDMABuffer

IdleVMWakeUpTime

IgnoreInstalledEMM

InDOSPolling

INT28Critical

IRQ9Global

Keyboard

KeyBoostTime

KeyBufferDelay

KeyIdleDelay

KeyPasteCRSkipCount

KeyPasteDelay

KeyPasteSkipCount

KeyPasteDelay

KeyPasteSkipCount

KeyPasteTimeout

KybdPasswd

KybdReboot

Local

LocalLoadHigh

LocalReboot

LPT1AutoAssign

LPT2AutoAssign

LPT3AutoAssign

LPT4AutoAssign

LRULowRateMult

LRURateChngTime

LRUSweepFreq

LRUSweepLen

LRUSweepLowWater

LRUSweepReset

MapPhysAddress

MaxBPs

MaxCOMPort

MaxDMAPGAddress

MaxPagingFileSize

MaxPhysPage

MCADMA

MessageBackColor

MessageTextColor

MinTimeSlice

MinUnlockMem

MinUserDiskSpace

Mouse

MouseSoftInit

NetAsynchFallback

NetAsynchTimeout

NetDMASize

NetHeapSize

Network

NMIReboot

NoEMMDriver

NoWaitNetIO

OverlappedIO

PageBuffers

PageOverCommit

Paging

PagingDrive

PagingFile

PerformBackfill

PermSwapDOSDrive

PermSwapSizeK

PerVMFILES

PSPIncrement

ReflectDosInt2A

ReservedHighArea

ReservePageFrame

ReverseVideoROM

ROMScanThreshold

ScrollFrequency

SGrabLPT

SyncTime

SystemROMBreakPoint

SysVMEMSLimit

SysVMEMSLocked

SysVMEMSRequired

SysVMV86Locked

SysVMXMSLimit

SysVMXMSRequired

TimerCriticalSection

TokenRingSearch

TranslateScans

TrapTimerPorts

UniqueDOSPSP

UseableHighArea

UseInstFile

UseROMFont

VGAMonoText

VideoBackgroundMsg

VideoSuspendDisable

VirtualHDIrq

WindowKBRequired

WindowMemSize

WindowUpdateTime

WinExclusive

32BitDiskAccess

32BitDiskAccess=Boolean

This entry turns 32-bit disk access on or off. The default is Off if your hard disk is capable of supporting 32-bit disk access. Otherwise, this setting is not present in SYSTEM.INI. To change this entry, choose the 386 Enhanced icon in Control Panel. If this setting is not present in SYSTEM.INI, the 32-Bit Disk Access check box will not appear in Control Panel Virtual Memory dialog box.

File: SYSTEM.INI Section: [386enh]

A20EnableCount

A20EnableCount=number

This entry specifies the initial A20 enable count, which identifies the A20 handler that HIMEM.SYS uses to access extended memory. (For more information about the A20 enable count, see the documentation for your extended memory manager.) The default is the value computed by the Win386 loader. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry.)

File: SYSTEM.INI Section: [386enh]

AllEMSLocked

AllEMSLocked=Boolean

This entry, if On, locks the contents of expanded memory used by all Windows and non-Windows applications into memory (instead of swapping it to disk) and overrides PIF settings for EMS Memory Locked. Set this value to On if you are using a disk cache program that uses expanded memory. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

AllVMsExclusive

AllVMsExclusive=Boolean

This entry, when On, forces all applications to run in exclusive full-screen mode, overriding all other settings in the application PIFs. Setting AllVMsExclusive=on might help avoid UAEs when you are running network or memory-resident software that is not compatible with Windows. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

AllXMSLocked

AllXMSLocked=Boolean

This entry, if On, locks the contents of the extended memory used by all Windows and non-Windows applications into memory (instead of swapping it to disk), and overrides PIF settings for XMS Memory Locked. The default is Off. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry).

File: SYSTEM.INI Section: [386enh]

AltKeyDelay

AltKeyDelay=seconds

This entry specifies how much time Windows waits to process a keyboard interrupt after it processes an ALT interrupt. Some applications expect a slower processing rate than Windows 386 enhanced mode usually uses. Increase this value if such an application has trouble handling the ALT key. The default is .005. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

AltPasteDelay

AltPasteDelay=seconds

This entry specifies how much time Windows waits before pasting any characters after the ALT key has been pasted. Some applications may require more time for recognition of the ALT keystroke. The default is .025. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

AutoRestoreScreen

AutoRestoreScreen=Boolean

This entry specifies whether the display for non-Windows applications should be restored by Windows or by the applications themselves when they become active. If this entry is On, Windows saves the display information for a non-Windows application in memory when you switch away from the application. When you switch back to the application, Windows restores the screen. If this entry is Off, the application must restore its own display by repainting the screen. This requires less memory, but can slow down performance since Windows can usually restore the screen faster. This entry only applies to VGA displays and affects only applications that notify Windows that they can update their screens automatically when Windows sends a display update call. The default is On. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

BkGndNotifyAtPFault

BkGndNotifyAtPFault=Boolean

This entry specifies when a notification should be sent to a non-Windows application or to Windows to prevent the application from attempting to access the display (which might corrupt the display of a different application). This notification can be sent while switching to a different application or when attempting to access the actual display. If this entry is Off, Windows sends a notification when switching between applications. Set this entry to Off if you are using a display that has special hardware (for example, 8514 and TIGA). If you are using a VGA display, setting this entry to On should work in most cases. The default is On for VGA displays; Off for 8514 displays. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

CGA40WOA.FON

CGA40WOA.FON=filename

This entry specifies the filename of the fixed-pitch display font used for non-Windows applications with a display of 40 columns and 25 or fewer lines. The default is none. To change this entry, exit Windows and run Setup from MS-DOS, then change the Codepage or Display setting.

File: SYSTEM.INI Section: [386enh]

CGA80WOA.FON

CGA80WOA.FON=filename

This entry specifies the filename of the fixed-pitch display font used for non-Windows applications with a display of 80 columns and 25 or fewer lines. The default is none (blank). To change this entry, exit Windows and run Setup from MS-DOS, then change the Codepage or Display setting.

File: SYSTEM.INI Section: [386enh]

CGANoSnow

CGANoSnow=Boolean

This entry, when On, causes Windows to do special handling to avoid snow appearing on an IBM CGA display device. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

COM1AutoAssign
COM2AutoAssign
COM3AutoAssign
COM4AutoAssign

COM1AutoAssign=number | seconds

COM2AutoAssign=number | seconds

COM3AutoAssign=number | seconds

COM4AutoAssign=number | seconds

These entries indicate the contention detection values for each connected communications port. These values are used by Windows to determine how to arbitrate requests for the use of a device by more than one application, at least one of which is a non-Windows application. If the value is -1, Windows will display a warning message that asks you which application should be given control of the port. If the value is 0, any application can use the device at any time. If the value is a positive integer less than 1000, this value represents the number of seconds after an application stops using the device before another application can use the same device. The default is 2. To change this entry, choose the 386 Enhanced icon from Control Panel.

File: SYSTEM.INI Section: [386enh]

COM1Base
COM2Base
COM3Base
COM4Base

COM1Base=address

COM2Base=address

COM3Base=address

COM4Base=address

These entries specify the base (first) port for the serial port adapter you are using for both standard mode and 386 enhanced mode. Check your hardware documentation for the appropriate value. The defaults are COM3Base=3E8h, and the port address values in the BIOS data area for COM1, COM2, and COM4. To change this entry, choose the Ports icon in Control Panel, and click the Advanced button for the selected COM port.

File: SYSTEM.INI Section: [386enh]

COMBoostTime

COMBoostTime=milliseconds

This entry specifies the time to allow a virtual machine to process a COM interrupt. If a communications application is losing keyboard characters on the display, you can try increasing this value. The default is 2. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

COM1Buffer
COM2Buffer
COM3Buffer
COM4Buffer

COM1Buffer=number

COM2Buffer=number

COM3Buffer=number

COM4Buffer=number

These entries specify the number of characters that the device will buffer on the corresponding communications port. Before changing one of these entries, make sure the corresponding COMxProtocol= entry has the proper value. Buffering may slow down communications on a port, but might be necessary to prevent some communications applications from losing characters at high baud rates. The size of the buffer required will depend on the speed of the machine and the applications needs. The default is 128. Before increasing this value, see COMxProtocol=. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

COMdrv30

COMdrv30=Boolean

This entry, if On, indicates that the Virtual COM Driver (VCD) will use its own copy of the serial communications drivers interrupt handler. This will improve performance of COM ports. Set this entry to On if you are using a Windows 3.0 serial communications driver. Set this entry to Off if you are using the standard Windows 3.1 serial communications driver. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

COM1FIFO
COM2FIFO
COM3FIFO
COM4FIFO

```
COM1FIFO=Boolean
COM2FIFO=Boolean
COM3FIFO=Boolean
COM4FIFO=Boolean
```

These entries specify whether the FIFO buffer of a COM ports 16550 Universal Asynchronous Receiver Transmitter (UART) should be enabled (On) or disabled (Off). If a serial port does not have a 16550 UART, this setting is ignored. These values are used by Windows for both standard and enhanced modes. The default is On.

File: SYSTEM.INI Section: [386enh]

COM1Irq
COM2Irq
COM3Irq
COM4Irq

COM1Irq=number

COM2Irq=number

COM3Irq=number

COM4Irq=number

These entries specify which interrupt line is used by the device on the specified serial port. Check your hardware documentation for the correct value. If there is a hardware conflict between ports, set a value of _1 to disable input for that COM port. The defaults for ISA and EISA machines are COM1Irq=4, COM2Irq=3, COM3Irq=4, and COM4Irq=3; for MCA machines, the defaults are COM1Irq=4, COM2Irq=3, COM3Irq=3, and COM4Irq=3. To change this entry, choose the Ports icon in Control Panel and click the Advanced button.

File: SYSTEM.INI Section: [386enh]

COMIrqSharing

COMIrqSharing=Boolean

This entry specifies whether COM interrupt lines will be sharable between mulitiple serial ports or with other devices. Set this switch if your machine uses the same interrupt for COM3 or COM4 as it does for COM1 or COM2. The default is On for Micro Channel and EISA machines; Off for all other machines. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

COM1Protocol
COM2Protocol
COM3Protocol
COM4Protocol

COM1Protocol=XOFF | blank

COM2Protocol=XOFF | blank

COM3Protocol=XOFF | blank

COM4Protocol=XOFF | blank

These entries specify whether Windows 386 enhanced mode should stop simulating characters in a virtual machine after the virtual machine sends an XOFF character. Set the value for a port to XOFF if a communications application using that port is losing characters while doing text transfers at high baud rates. Windows will resume simulating characters when the virtual machine sends another character after the XOFF character.

Leave this entry blank if the application does binary data transfers; setting this switch might suspend binary transmissions. The default is no entry, which is the same as any entry other than XOFF. Windows will not check for XOFF characters if this entry is blank or set to anything other than XOFF. If the application continues to lose characters after this entry is set, try increasing the related COMxBuffer= value. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

Device

Device=filename | *devicename

This entry specifies which virtual devices are being used with Windows 386 enhanced mode. This value can appear in two ways: either the name of a specific virtual device file (with path if necessary), or an asterisk (*) followed immediately by the device name to refer to a virtual device built into the WIN386.EXE file. Synonyms for Device= are Display=, EBIOS=, Keyboard=, Network=, and Mouse=. Filenames usually include the .386 extension. Multiple device lines are required to run Windows 386 enhanced mode. The default is none. Setup assigns appropriate values based on your system configuration. To change this entry, you must edit SYSTEM.INI.

If Setup detects protected mode block devices for Western Digitalcompatible controllers, it adds the entries device=*int13 and device=*wdctrl to [386enh]. A block device is a virtual device used in 386 enhanced mode to talk directly to disk drive controllers, bypassing MS_DOS and BIOS. To remove block device support, delete these entries in SYSTEM.INI. You do not need to delete files from your hard disk, as these devices are appended to WIN386.EXE.

File: SYSTEM.INI   Section: [386enh]

Display

Display=filename | *devicename

This entry specifies the display device being used with Windows 386 enhanced mode. This entry is a synonym for Device=. The default is none. Setup assigns an appropriate value based on your system configuration. To change this entry, choose the Windows Setup icon from the Main Group window.

File: SYSTEM.INI Section: [386enh]

DMABufferIn1MB

DMABufferIn1MB=Boolean

This entry, if set to On, indicates that the direct memory access (DMA) buffer memory should be in the first 1 MB of memory (above 640K, if possible) to be compatible with 8-bit bus master cards. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

DMABufferSize

DMABufferSize=kilobytes

This entry specifies the amount of memory to be reserved for the DMA buffer. This memory will be allocated above 640K, if possible. Windows 386 enhanced mode will default to a DMA buffer size that will handle disk access. The default is 16. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

DOSPromptExitInstruc

DOSPromptExitInstruc=Boolean

This entry, if On, causes a message box to appear when you start the MS-DOS Prompt, displaying instructions on how to exit and switch away from the MS-DOS Prompt. Set this entry to Off if you do not want to see the message. The default is On. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

DualDisplay

DualDisplay=Boolean

Usually, when Windows is running in 386 enhanced mode, the memory between B000:0000 and B7FF:000F will be used by the general system unless a secondary display is detected. If this entry is On, this memory will be left unused and available for display adapters, or if you do not have a monochrome display card installed and want EMM386.EXE to include this address space as an upper memory block (UMB). If this entry is Off, the address range will be available on EGA systems but not under VGA systems, because the VGA display device supports monochrome modes, which use this address space. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

EBIOS

EBIOS=filename | *devicename

This entry specifies the extended BIOS device being used with Windows 386 enhanced mode. This entry is a synonym for Device=. The default is the value that Setup assigns, based on your system configuration. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

EGA40WOA.FON

EGA40WOA.FON=filename

This entry specifies the filename of the fixed-pitch display font used for non-Windows applications with a display of 40 columns and more than 25 lines. The default is none. To change this entry, exit Windows and run Setup from MS-DOS, then change the Codepage or Display setting.

File: SYSTEM.INI Section: [386enh]

EGA80WOA.FON

EGA80WOA.FON=filename

This entry specifies the filename of the fixed-pitch display font used for non-Windows applications with a display of 80 columns and more than 25 lines. The default is none. To change this entry, exit Windows and run Setup from MS-DOS, then change the Codepage or Display setting.

File: SYSTEM.INI Section: [386enh]

EISADMA

EISADMA=Boolean | channel,size

This entry specifies the mode of operation of an extended DMA channel for Extended Industry Standard Architecture (EISA) machines only. This entrys value can take one of two forms. If you set this entry to Off, Windows treats the machine as non-EISA, avoiding all EISA-related logic. Set this value to Off if you cannot run Windows 386 enhanced mode on an EISA machine. If you are using an EISA machine, you can specify the default transfer size for one or more DMA channels. The channels can operate in these modes: 8-bit (8), 16-bit specified in words (16w), 16-bit specified in bytes (16b), or 32-bit (32). If you are not using an EISA machine, Windows ignores this entry. The default is 0,8; 1,8; 2,8; 3,8; 5,16w; 6,16w; 7,16w. Each pair goes with its own EISADMA= entry. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

EMMExclude

EMMExclude=paragraph-range

This entry specifies a range of memory that Windows will not scan to find unused address space. This has the side effect of turning off the RAM and ROM search code for the range. The range (two paragraph values separated by a hyphen) must be between A000 and EFFF. This scanning can interfere with some adapters that use the same memory area. The starting value is rounded down and the ending value is rounded up to a multiple of 16K. For example, you could set EMMExclude=C800-CFFF to prevent Windows from scanning the addresses C800:0000 through CFFF:000F (the block that some VGA cards use to enhance performance). You can specify more than one range by including more than one EMMExclude= entry. The default is none. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

EMMInclude

EMMInclude=paragraph-range

This entry specifies a range of memory that Windows will scan for unused address space regardless of what may be there. EMMInclude= takes precedence over EMMExclude= if you specify ranges that overlap. The range (two values separated by a hyphen) must be between A000 and EFFF. The starting value is rounded down and the ending value is rounded up to a multiple of 16K. For example, you could set EMMInclude=C800-CFFF to ensure that Windows scans the addresses C800:0000 through CFFF:000F. You can specify more than one range by including more than one EMMInclude= entry. The default is none. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

EMMPageFrame

EMMPageFrame=paragraph

This entry specifies the starting paragraph where the 64K page frame will begin when Windows cannot find a suitable page frame, allowing an EMS page frame in an area containing some unused RAM or ROM. For example, you could set EMMPageFrame=C400 to start the page frame at C400:0000. The default is none. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

EMMSize

EMMSize=kilobytes

This entry specifies the total amount of memory to be made available for mapping as expanded memory. The default allocates the maximum possible amount of system memory as expanded memory. Specify a value for this entry if you run an application that allocates all of the available expanded memory. This will be apparent if, when you run the application, you can never create any new virtual machine. If this value is 0, then no expanded memory will be allocated. This entry does not prevent the EMS driver from being loaded; use NoEMMDriver=on to turn off EMS. The default is 65,536. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

FileSysChange

FileSysChange=Boolean

This entry indicates whether File Manager automatically receives messages anytime a non-Windows application creates, renames, or deletes a file. When this entry is Off, a virtual machine can be run exclusively even when it modifies files. Setting this entry to On can slow down system performance significantly. The default is On for 386 enhanced mode; Off in standard mode. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

Global

Global=device-name

This entry defines MS-DOS devices loaded in CONFIG.SYS that need to be global to the system. The default is that all devices loaded in CONFIG.SYS are global to the system. But certain virtual devices might specify that an MS-DOS device be local (for example, MS$MOUSE). Use this entry to override that local specification. The device-name value must exactly match the case of the device name in CONFIG.SYS, or this entry will not work. Most device names are in all capital letters, so this value must usually be in all caps. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

HardDiskDMABuffer

HardDiskDMABuffer=kilobytes

This entry specifies the amount of memory used for the Direct Memory Access (DMA) buffer. If you are using SMARTDrive and double buffering is turned on, this amount is determined automatically. You may need to change this value if you are using a hard disk that supports DMA and are not using SMARTDrive, or if you do not have double buffering turned on. The default is 0 for AT architecture computers; 64 for Micro Channel computers or computers that use DMA channel 3 (unless you use SMARTDrive with double buffering, in which case the default is 0). To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

IdleVMWakeUpTime

IdleVMWakeUpTime=seconds

This entry causes timer interrupts to periodically wake up idle virtual machines after the specified number of seconds has elapsed. If a virtual machine does not use timer interrupts (INT 8 or INT 1Ch), Windows will not usually force timer interrupts into a virtual machine unless it is active. This entry forces the timer interrupts to occur. The value for this entry is rounded down to the lowest power of 2 (for example, 1, 2, 4, 8, 16, 32, 64). The default is 8. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

IgnoreInstalledEMM

IgnoreInstalledEMM=Boolean

This entry, if set to On, allows Windows to start in 386 enhanced mode even when there is an unknown expanded memory manager (EMM) running. Starting Windows with an unknown EMM running can cause the system to fail if memory-resident software was using expanded memory before Windows started. Set this entry only if no such software is installed or you are sure it will not be active when you run Windows. This entry applies only to EMMs servicing physical EMS hardware; Windows will not disable unrecognized 80386 expanded memory emulators. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

InDOSPolling

InDOSPolling=Boolean

This entry, if set to On, prevents Windows from running other applications when memory-resident software has the InDOS flag set. Setting this entry to On is necessary if the memory-resident software needs to be in a critical section to do operations off an INT 21 hook. Setting this entry to On will slow down system performance slightly. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

INT28Critical

INT28Critical=Boolean

This entry specifies whether a critical section is needed to handle INT 28h interrupts used by memory-resident software. Some network virtual devices do internal task switching on INT 28h interrupts. These interrupts might hang some network software, indicating the need to add an INT28hCritical= entry. If you are not using such software, setting this entry to Off might improve Windows task switching. The default is On. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

IRQ9Global

IRQ9Global=Boolean

This entry, if set to On, converts IRQ 9 masks to global. Set this entry to On if your system hangs when it reads from a floppy drive, or to make sure the system reads the floppy drive before starting Windows. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

Keyboard

Keyboard=filename | *devicename

This entry specifies the keyboard driver for Windows 386 enhanced mode. This entry is a synonym for Device=. The default is none. Setup assigns an appropriate value based on your system configuration. To change this entry, choose the Windows Setup icon from the Main Group window.

File: SYSTEM.INI Section: [386enh]

KeyBoostTime

KeyBoostTime=seconds

This entry specifies how much time an application gets to run with increased priority when it receives a keystroke. Use this entry to increase the response to keystrokes when several background applications are running. The default is .001. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

KeyBufferDelay

KeyBufferDelay=seconds

This entry specifies the time to delay pasting keyboard input after the keyboard buffer is full. Some applications might require more than .2 seconds. The default is .2. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

KeyIdleDelay

KeyIdleDelay=seconds

This entry specifies how long Windows ignores idle calls after simulating a keystroke into a virtual machine. You can set this value to 0 to speed up keyboard input, but some applications might respond sluggishly if you do. The default is .5. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

KeyPasteCRSkipCount

KeyPasteCRSkipCount=number

This entry specifies the number of times that a read-status INT 16 call should return a status of empty for the keyboard buffer after pasting a carriage return and before pasting another character. When Windows pastes data from the Clipboard to a non-Windows application, it must first paste the data to the BIOS keyboard buffer before pasting it into the application. This setting is used to slow down fast pasting from the Clipboard to the keyboard buffer so that the application can handle all incoming characters from the buffer. If you seem to lose characters, or if the screen does not update often enough while pasting information from the Clipboard, increase this value. This setting is related to KeyPasteSkipCount=. The default is 10. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

KeyPasteDelay

KeyPasteDelay=seconds

This entry specifies how long to wait before pasting any characters after a key has been pasted. Some applications might require more time than .003 seconds for recognition of a keystroke. The default is .003. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

KeyPasteSkipCount

KeyPasteSkipCount=number

This entry specifies the number of times that a read-status INT 16 call should return a status of empty for the keyboard buffer before pasting another character. When Windows pastes data from the Clipboard to a non-Windows application, it must first paste the data to the BIOS keyboard buffer before pasting it into the application. This setting is used to slow down fast pasting from the Clipboard to the keyboard buffer so that the application can handle all incoming characters from the buffer. If you seem to lose characters, or if the screen does not update often enough while pasting information from the Clipboard, increase this value. The default is 2. This setting is related to KeyPasteCRSkipCount=. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

KeyPasteTimeout

KeyPasteTimeout=seconds

This entry specifies how much time to allow an application to make the necessary BIOS calls for reading keyboard input before Windows will change from the fast paste (INT 16h) to the slow paste (INT 9h) mechanism. The default is 1. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

KybdPasswd

KybdPasswd=Boolean

This entry specifies whether the Virtual Keyboard Device (VKD) should support PS/2 8042 commands that implement password security. This entry only applies to 8042 keyboard controllers that are compatible with the PS/2 computer. The default is On for IBM PS/2 computers; Off for all other computers. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

KybdReboot

KybdReboot=Boolean

If this entry is On, Windows will attempt to reboot the computer using a keyboard controller command. On some computers this method is unreliable. If your computer hangs while rebooting, then set this entry to Off, in which case Windows will quit and display a prompt to press CTRL+ALT+DEL a second time if you attempt to reboot from the keyboard while Windows is running. The default is On. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

Local

Local=device-name

This entry defines MS-DOS device drivers that need to be local to each virtual machine. The device-name value must exactly match the case of the device name, or this entry will not work. Most device names are in all capital letters, so this value must usually be in all caps. If a device driver is local, a separate copy of it will exist in each virtual machine, and each will keep different state information. Most device drivers will not function properly when this entry is used. An exception is CON, the MS-DOS console device, which should be local to avoid filling a buffer with input from multiple virtual machines. The default is none; but in a standard SYSTEM.INI file, Setup will set Local=CON. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

LocalLoadHigh

LocalLoadHigh=Boolean

This entry determines how the upper memory blocks (UMBs) are used when Windows in running on MS-DOS version 5.0. If this entry is Off, Windows uses all of the upper memory area, leaving no extra UMBs available for virtual machines. If this entry is On, then Windows does not use all of the upper memory area, so the UMBs are available locally to each virtual machine. The default is Off. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

LocalReboot

LocalReboot=On | Off

This entry specifies whether you can press CTRL+ALT+DEL to quit applications that cause an unrecoverable error in 386 enhanced mode, without restarting Windows. If this setting is On, you can quit the application. If this setting is Off, pressing CTRL+ALT+DEL will restart your entire system (as it usually does). To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

LPT1AutoAssign
LPT2AutoAssign
LPT3AutoAssign
LPT4AutoAssign

LPT1AutoAssign=seconds

LPT2AutoAssign=seconds

LPT3AutoAssign=seconds

LPT4AutoAssign=seconds

These entries apply to Windows version 3.0 only. If you have upgraded to Windows 3.1, these settings may still appear in your SYSTEM.INI file, but will have no effect. The default is 60.

File: SYSTEM.INI Section: [386enh]

LRULowRateMult

LRULowRateMult=number

This entry specifies the value used to determine the Least Recently Used (LRU) low paging rate sweep frequency, which is computed by multiplying the value for LRUSweepFreq= by the value specified for LRULowRateMult=. Values between 1 and 65535 can be used. The default is 10. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry.)

File: SYSTEM.INI Section: [386enh]

LRURateChngTime

LRURateChngTime=milliseconds

This entry specifies the length of time that the Virtual Memory Manager (VMM) stays at high rate with no paging before switching to low rate, and the length of time the VMM stays at low rate with no paging before turning the LRU sweep off. The default is 10,000. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry.)

File: SYSTEM.INI Section: [386enh]

LRUSweepFreq

LRUSweepFreq=milliseconds

This entry specifies the time between LRU sweep passes. This is also the high paging rate sweep frequency. The default is 250. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry.)

File: SYSTEM.INI Section: [386enh]

LRUSweepLen

LRUSweepLen=*length-in-pages*

This entry specifies the length in pages of the region swept on each pass. Windows computes this value by dividing the value of LRUSweepReset by the value of LRUSweepFreq. The value must be at least 1. The default is 1024. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry.)

File: SYSTEM.INI Section: [386enh]

LRUSweepLowWater

LRUSweepLowWater=*number*

This entry specifies when the LRU sweeper should be turned on. When the number of free pages drops below this value, the sweeper is turned on. The default is 24. To change this entry, you must edit SYSTEM.INI. (You should never have to change this entry.)

File: SYSTEM.INI Section: [386enh]

LRUSweepReset

LRUSweepReset=*milliseconds*

This entry specifies the time desired for an ACC bit reset divided by 4 MB of pages. Therefore, the time to reset all ACC bits is *number of pages in system + 1023/1024,* where 1024 pages = 4 MB. The minimum value is 100. The default is 500. To change this entry, you must edit SYSTEM.INI. (You should never have to change this entry.)

File: SYSTEM.INI Section: [386enh]

MapPhysAddress

MapPhysAddress=*range*

This entry specifies the address range (in megabytes) in which the memory manager will preallocate physical page-table entries and linear address space. Set a value for this entry if you are using an MS-DOS device driver that needs this contiguous memory (such as an older version of RAMDrive that uses extended memory). The default is none. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

MaxBPs

MaxBPs=*number*

This entry specifies the maximum number of break points (a method for transferring control to Windows 386 enhanced mode) that can be used by the VMM. You may need to increase this value if you are using a third-party virtual device driver that requires more break points than the default value. The default is 200. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

MaxCOMPort

MaxCOMPort=*number*

This entry specifies the maximum number of COM ports supported in 386 enhanced mode. Change this value if you have more than four COM ports in your system. The default is 4. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

MaxDMAPGAddress

MaxDMAPGAddress=*address*

This entry specifies the maximum physical page address that can be used for DMA. This entry works only with hard drives that support DMA. The default is 0FFFh for non-EISA computers; 0FFFFFh for EISA computers. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

MaxPagingFileSize

MaxPagingFileSize=*kilobytes*

This entry specifies the maximum size for a temporary swap file. The default is 50 percent of the available disk space. To change this entry, choose the 386 Enhanced icon in Control Panel, then choose Virtual Memory to set a value.

File: SYSTEM.INI Section: [386enh]

MaxPhysPage

MaxPhysPage=*hexadecimal-page-number*

This entry specifies the maximum physical page number that the VMM can manage as a useable page, allowing pages to be added at a physical address beyond what the VMM recognized during initialization. If the value specified is less than what the VMM determines, the VMM will ignore several physical pages that it would usually use, preventing the use of memory. This is useful if you are using a hardware device that cannot recognize all of the physical memory in your computer (for example, ISA DMA network cards cannot access physical memory above 16 MB). The default is determined by Windows based on the highest physical page number detected by the VMM during initialization. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

MCADMA

MCADMA=*Boolean*

This entry specifies whether Windows should use the Micro Channel (MCA) extensions to DMA. This entry only applies to MCA computers. Set this entry to **Off** if you are using an MCA computer but the DMA extensions are not implemented. The default is **On** for most MCA computers; **Off** for all other computers. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

MessageBackColor

MessageBackColor=*vga-color-attribute*

This entry specifies the background color of message screens, such as the screen that appears when you press CTRL+ALT+DEL. The default is 1 (for blue). To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

MessageTextColor

MessageTextColor=*vga-color-attribute*

This entry specifies the color used to display text in message screens, such as the screen that appears when you press CTRL+ALT+DEL. The default is F (for white). To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

MinTimeSlice

MinTimeSlice=*milliseconds*

This entry specifies the minimum time a virtual machine will be allowed to run before other virtual machines can take over. A smaller value (such as 10 milliseconds) will make multitasking appear smoother, but will diminish the overall system performance. The default is **20**. To change this entry, choose the 386 Enhanced icon from the Control Panel.

File: SYSTEM.INI Section: [386enh]

MinUnlockMem

MinUnlockMem=*kilobytes*

This entry specifies the amount of memory that must remain unlocked and available for use when resuming a virtual machine if more than one virtual machine is running. The default is **40**. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry.)

File: SYSTEM.INI Section: [386enh]

MinUserDiskSpace

MinUserDiskSpace=*kilobytes*

This entry tells Windows how much disk space to leave free when creating a temporary swap file. Use this entry if your systems paging drive has less available space than Windows can use for paging. This entry has no effect if a permanent swap file exists. The default is **500**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

Mouse

Mouse**=**filename | *devicename

This entry specifies the virtual device that handles the mouse hardware in 386 enhanced mode. This entry is a synonym for **Device=**. The default is none. Setup assigns an appropriate value based on your system configuration. To change this entry, choose the Windows Setup icon from the Main Group window.

File: SYSTEM.INI Section: [386enh]

MouseSoftInit

MouseSoftInit=*Boolean*

This entry specifies whether Windows should convert INT 33h function 0 hard initialization calls to function 33 soft initialization calls, which do not reset the mouse hardware. Set this entry to **On** if you want to use a mouse with a non-Windows application started in a window. Set this entry to **Off** if you have problems using the mouse with an application, such as extraneous display output or a distorted cursor. If you set this entry to **Off**, you will not be able to use the mouse with a non-Windows application started in a window. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

NetAsynchFallback

NetAsynchFallback=*Boolean*

This entry, if set to **On**, tells Windows to attempt to save a failing NetBIOS request. When an application issues an asynchronous NetBIOS request, Windows will attempt to allocate space in its global network buffer to receive the data. If there is insufficient space in the global buffer, Windows will usually fail the NetBIOS request. If this entry is **On**, Windows will attempt to save such a request by allocating a buffer in local memory and preventing any other virtual machines from running until the data is received and the timeout period specified by **NetAsynchTimeout=** expires. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

NetAsynchTimeout

NetAsynchTimeout=*seconds*

This entry specifies the timeout period when Windows needs to enter a critical section to service an asynchronous NetBIOS request. This value is used only when **NetAsynchFallback=** is set. This value can include a decimal, such as 0.5. The default is **5.0**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

NetDMASize

NetDMASize=*kilobytes*

This entry specifies the DMA buffer size for NetBIOS transport software if a network has been installed. In this case, the buffer size is the larger value between this value and the value of **DMABufferSize=**. The default is **32** on Micro Channel machines, and **0** on non-Micro Channel machines. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

NetHeapSize

NetHeapSize=*number*

This entry specifies the size, in 4K increments, of the buffers that Windows 386 enhanced mode allocates in conventional memory for transferring data over a network. All values are rounded up to the nearest 4K. The default is **12.** To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

Network

Network**=***filename* **|** **devicename*

This entry specifies the type of network you are using with Windows 386 enhanced mode. This entry is a synonym for **Device=**. The default is none. Setup assigns an appropriate value based on your system configuration. To change this entry, choose the Windows Setup icon from the Main Group window.

File: SYSTEM.INI Section: [386enh]

NMIReboot

NMIReboot=*Boolean*

This entry, if set to **Yes**, causes a reboot to occur when a nonmaskable interrupt is received. The default is **No.** To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

NoEMMDriver

NoEMMDriver=*Boolean*

This entry, if set to **On**, prevents Windows 386 enhanced mode from installing the expanded memory driver. This differs from setting **EMMSize=0**, which prevents UMBs from being allocated, but does not prevent the EMS driver from being loaded. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

NoWaitNetIO

NoWaitNetIO=*On* | *Off*

This entry specifies whether to convert synchronous NetBIOS commands to asynchronous commands. When this entry is **On**, synchronous NetBIOS commands are converted. This can improve overall system performance when running several applications. The default is **On**. To change this value, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

OverlappedIO

OverlappedIO=*Boolean*

This entry, if **On**, allows several virtual machines to make read and write requests to a disk before the first request has been completed. If this entry is **Off**, virtual machines cannot issue a request to read or write to a disk until any previous read and write requests have been completed. The default is **Off** if **InDOSPolling=On**; otherwise, **On**. To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry.)

File: SYSTEM.INI Section: [386enh]

PageBuffers

PageBuffers=*number*

This entry specifies the number of 4K page buffers used to store asynchronous read and write pages. This setting is used

File: SYSTEM.INI Section: [386enh]

PageOverCommit

PageOverCommit=*megabytes*

This entry specifies the multiplier that determines the amount of linear address space the VMM will create for the system, which is computed by rounding up the amount of available physical memory to the nearest 4 MB and then multiplying that value by the value specified for **PageOverCommit**. Increasing this value increases the amount of available linear address space, causing the size of data structures to increase. This also increase paging activity proportionately and can slow down the system. You can specify a value between **1** and **20**. The default is **4**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

Paging

Paging**=***Boolean*

This entry sets demand paging (virtual memory). Set this entry to **Off** only if you need the disk space usually used for a temporary swap file. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

PagingDrive

PagingDrive**=***drive-letter*

This entry specifies the disk drive where Windows 386 enhanced mode will allocate a temporary swap file. This entry is ignored if you have a permanent swap file. If you dont have a permanent swap file and no drive is specified or the specified drive does not exist, Windows attempts to put the temporary swap file on the drive containing the SYSTEM.INI file. If the specified drive is full, paging will be turned off. The default is none. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

PagingFile

PagingFile=*path-and-filename*

This entry specifies the path and filename for the temporary swap file that is created when you start Windows 386 enhanced mode. This file is deleted when you quit Windows. This setting overrides the **PagingDrive=** setting. The default is **WINDOWS\WIN386.SWP**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

PerformBackfill

PerformBackfill=*Boolean*

This entry specifies whether or not to allocate a full 640K of memory to a computer that has less than 640K of base conventional memory. Setting this entry overrides the automatic checking done by Windows. The default is **On** (Windows automatically checks to see if it needs to fill in memory based on how MS-DOS is occupying memory.) To change this entry, you must edit SYSTEM.INI. (You should never need to change this entry, because Windows can automatically detect whether or not to perform a backfill.)

File: SYSTEM.INI Section: [386enh]

PermSwapDOSDrive

PermSwapDOSDrive=*drive-letter*

This entry specifies the disk drive where Windows in 386 enhanced mode will create a permanent swap file. The default is none. To change this entry, choose the 386 Enhanced icon from Control Panel.

File: SYSTEM.INI Section: [386enh]

PermSwapSizeK

PermSwapSizeK=*kilobytes*

This entry specifies the desired size of a permanent swap file. The default is none. To change this entry, choose the 386 Enhanced icon from Control Panel.

PerVMFILES

PerVMFILES=*number*

This entry specifies the number of private file handles Windows should allocate to each virtual machine. Increase this value if an application does not have enough file handles to run. The total number of file handles, including the global handles specified in the **files=** statement in CONFIG.SYS, cannot exceed 255. If it does exceed 255, this value will be rounded down. Set this value to 0 to prevent the allocation of any private file handles. The default is **10**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

PSPIncrement

PSPIncrement=*number*

This entry specifies the amount of additional memory, in 16-byte increments, that Windows should reserve in each successive virtual machine when **UniqueDOSPSP= On**. The value that will work best for your machine might vary depending on the memory configuration and the applications you are running. Valid values are **2** through **64**. The default is **2**. See UniqueDosPSP for more information. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

ReflectDosInt2A

ReflectDosInt2A=*Boolean*

This entry indicates whether Windows should consume or reflect MS-DOS INT 2A signals. If the value is **Off**, Windows will consume these signals and therefore run more efficiently. Set this entry to **On** if you are running memory-resident software that relies on detecting INT 2A messages. The default is **Off.** To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

ReservedHighArea

ReservedHighArea=paragraph-range

This entry specifies a range of memory that Windows will not scan to find unused address space. This also turns off the RAM and ROM search code for the range. This scanning can interfere with some adapters that use the same memory area. The range (two paragraph values separated by a hyphen) must be between **A000** and **EFFF.** The starting value is rounded down and the ending value is rounded up to a multiple of 4K. For example, you could set **ReservedHighArea=E000-EFFF** to prevent Windows from scanning the first 4K of memory starting at E000. You can specify more than one range by including more than one **ReservedHighArea=** entry. The default is none. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

ReservePageFrame

ReservePageFrame=*Boolean*

This entry tells Windows whether to give preference to EMS page frame space or conventional memory when it has to use one of them to allocate MS-DOS transfer buffers. This choice is necessary when Windows cannot find space between 640K and 1MB other than EMS page frame space. If set to **On**, this entry will preserve EMS page frame space at the expense of conventional memory. If you are not going to run non-Windows applications that use expanded memory, set this entry to **Off** to give non-Windows applications more conventional memory. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

ReverseVideoROM

ReserveVideoROM=*Boolean*

This entry tells Windows that the video ROM exists in pages C6 and C7. If the text font appears scrambled when starting non-Windows applications, set this entry to **On**. The default is **Off.** To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry, because the default value causes Windows to detect video ROM correctly.)

File: SYSTEM.INI Section: [386enh]

ROMScanThreshold

ROMScanThreshold=*number*

This entry specifies a parameter used to determine whether an area of memory in the adapter area (usually between C000-EFFF) is ROM when the area has no ROM header or incorrect header information. The number for this entry specifies how many transitions (value changes) must occur within the memory area to determine whether or not it is ROM. If the number of transitions is greater than the value for this entry, Windows recognizes the memory as ROM. If the number of transitions is less than the value, Windows recognizes the memory as usable memory. Specifying **0** causes Windows to recognize the memory area as ROM and then reserve it as unusable. The default is **20**. (You should never need to change this entry.)

File: SYSTEM.INI Section: [386enh]

ScrollFrequency

ScrollFrequency=*number*

This entry specifies the number of lines you can scroll in a non-Windows application that is running in Windows before its display is update. The default is 2. You should never need to change this setting.

File: SYSTEM.INI Section: [386enh]

SGrabLPT

SGrabLPT**=**port-number

This entry routes all printer interrupts on the specified port to the system virtual machine rather than to the current virtual machine. The default is none. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

SyncTime

SyncTime=Boolean

This entry, if **On**, causes Windows to periodically synchronize its time with the computers CMOS clock. If this entry is **Off**, Windows usually maintains the correct time, unless **TrapTimerPorts=Off** and you are running applications that can cause the system time to run faster or slower than the actual time. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

SystemROMBreakPoint

SystemROMBreakPoint=*Boolean*

This entry specifies whether Windows should use ROM address space between F000:0000 and 1MB for a break point. Windows 386 enhanced mode usually searches this space to find a special instruction that is used as a system break point. If this address space contains something other than permanently available ROM, set this entry to **Off**. The default is **On** if Windows is started in real mode of the 80386/486 CPU. **Off** if Windows is started in virtual mode of the 80386/486 CPU. Most 386 memory managers, such as QEMM and 386MAX, require this value to be set to **Off.** If a 386 memory manager is running, Windows is started in virtual mode. Otherwise, Windows is started in real mode. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

SysVMEMSLimit

SysVMEMSLimit=*kilobytes*

This entry specifies how many kilobytes of expanded memory Windows should be permitted to use. Setting this value to **0** prevents Windows from gaining access to any expanded memory. Setting it to **-1** gives Windows all the available expanded memory that it requests. The default is **2048**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

SysVMEMSLocked

SysVMEMSLocked=*Boolean*

This entry indicates whether to swap to the hard disk the expanded memory being used by non-Windows applications. Locking expanded memory can improve the performance of a Windows application that uses it, but locking it slows down the rest of the system. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

SysVMEMSRequired

SysVMEMSRequired=*kilobytes*

This entry specifies how much expanded memory must be free to start Windows. Leave this value at **0** if no non-Windows application running under Windows will require expanded memory. The default is **0**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

SysVMV86Locked

SysVMV86Locked=*Boolean*

This entry, if set to **On**, causes the virtual-mode memory being used in the system virtual machine to remain locked in memory rather that being swappable out to disk. Because Windows handles this process, there is no known reason to set this entry. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

SysVMXMSLimit

SysVMXMSLimit=*kilobytes*

This entry specifies the maximum amount of memory the extended memory driver will allocate to MS-DOS device drivers and memory-resident software in the system virtual machine. Set the value to **-1** to give an application all the available extended memory that it requests. The default is **2048**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

SysVMXMSRequired

SysVMXMSRequired=*kilobytes*

This entry specifies how many kilobytes of extended memory must be reserved by the XMS driver to start Windows. Leave this value at **0** if there are no XMS users in the system virtual machine. The default is **0**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

TimerCriticalSection

TimerCriticalSection=*milliseconds*

This entry instructs Windows to go into a critical section around all timer interrupt code, and specifies a timeout period. Specifying a positive value will assure that only one virtual machine at a time will receive timer interrupts. Some networks and other global memory-resident software may fail unless this entry is used. However, using this entry will slow down performance and can make the system sluggish or seem to stop for short periods of time. The default is **0**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

TokenRingSearch

TokenRingSearch=*Boolean*

This entry tells Windows whether to search for a token ring network adapter on machines with IBM PC/AT architecture. Set this entry to **Off** if you are not using a token ring card and the search interferes with another device. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

TranslateScans

TranslateScans=*Boolean*

This entry indicates whether Windows will translate a keyboards scan codes to make them compatible with standard IBM scan codes. This entry is necessary only for keyboards that generate non-standard scan codes. This entry is used only for the Switcher Screen and message boxes. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

TrapTimerPorts

TrapTimerPorts=*Boolean*

If this entry is set to **Off**, Windows allows applications to read from and write to the system timer ports. In this case, Windows can usually detect when an application has changed the timer interrupt interval and then make any adjustments to the time. However, some applications may prevent Windows from keeping accurate time. If your systems time appears to be running fast or slow, set this entry to **On**, so that Windows will deny access to the timer ports. This ensures that the system time will be accurate, but may cause applications that frequently access the timer ports to run slowly. If you do not want to set this entry to **On**, set **SyncTime=On**, which causes Windows to restore the correct time periodically regardless of whether or not applications access the timer ports. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

UniqueDOSPSP

UniqueDOSPSP=*Boolean*

This entry, if set to **On**, tells Windows to start every application at a unique address (PSP). Each time Windows creates a new virtual machine to start a new application, Windows reserves a unique amount of memory (*i* bytes) below the application. For example, the first application would be loaded at address M, the second at address M+*i*, the third at M+2*i*, and so forth. The amount of memory *i* is determined by **PSPIncrement=**. These entries should help assure that applications in different virtual machines all start at different addresses. Some networks use applications load addresses to identify the different processes using the network. On such networks, failing to set this entry might cause one application to fail when you exit another, because the network interprets them as the same. However, setting this entry will leave slightly less memory for non-Windows applications. The default is **Off**, except if you are running a network based on Microsoft Network or LAN Manager, the default value is **On**. See the NETWORKS.WRI online document to find out if the network you are running is one of these. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

UseableHighArea

UseableHighArea**=**paragraph-range

This entry specifies the range of memory that Windows will scan for unused address space regardless of what may be there. **UseableHighArea=** takes precedence over **ReservedHighArea=** if you specify ranges that overlap. The range (two values separated by a hyphen) must be between **A000** and **EFFF.** The starting value is rounded down and the ending value is rounded up to a multiple of 4K. For example, you could set **UseableHighArea=E100-E3FF** to ensure that Windows scans the first 12K of memory starting at E100. You may specify more than one range by including more than one **UseableHighArea=** entry. The default is none. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

UseInstFile

UseInstFile=*Boolean*

This entry specifies whether Windows should look in the INSTANCE.386 file for information it can use to determine whether data structures within MS-DOS need to be local. Two other methods give Windows this information: internal tables within the device, and INT 2Fh call documented in the OEM Adaptation Kit. Both methods are preferable to using this setting; this entry is provided only for compatibility with Windows/386 version 2.x. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI   Section: [386enh]

UseROMFont

UseROMFont=*Boolean*

This entry, if **On**, causes Windows to use the soft font stored in video ROM for displaying messages that appear when non-Windows applications are running in a full screen and for displaying the text that appears when switching away from a non-Windows application. Set this entry to **Off** if the font used to display messages is different from the font used in the application, or if random dots and shapes appear on your screen. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

VGAMonoText

VGAMonoText**=***Boolean*

This entry tells Windows to ignore the video memory address space in VGA displays, usually used for monochrome adapters. When this entry is **Off**, Windows can use the B000h through B7FF range for upper memory space, provided no hardware device is using these addresses and other applications do not use the monochrome display mode of the VGA adapter. This entry applies only to VGA displays. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

VideoBackgroundMsg

VideoBackgroundMsg**=***Boolean*

This entry, if **On**, causes Windows to display a message when a background application is suspended, or if its display cannot be updated properly because video memory is low. Setting this entry to **Off** turns off the warning message. This entry affects all non-Windows applications currently running. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

VideoSuspendDisable

VideoSuspendDisable=*Boolean*

This entry specifies whether or not to suspend applications running in the background if its display becomes corrupted. If this entry is **On**, the application will continue running. If it is **Off**, applications will be suspended and a warning message will appear if the **VideoBackgroundMsg=On**. This entry applies only to VGA displays. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

VirtualHDIrq

VirtualHDIrq=*Boolean*

This entry allows Windows 386 enhanced mode to terminate interrupts from the hard disk controller, bypassing the ROM routine that handles these interrupts. Some hard drives might require that this entry be set to **Off** for interrupts to be processed correctly. If this entry is set to **Off**, the ROM routine handles the interrupts, which slows the systems performance. The default is **On**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

WindowKBRequired

WindowKBRequired=*kilobytes*

This entry specifies how much conventional memory must be free to start Windows. The default is **256**. To change this entry, you must edit SYSTEM.INI.

File: SYSTEM.INI Section: [386enh]

WindowMemSize

WindowMemSize**=**number **|** kilobytes

This entry limits the amount of conventional memory Windows can use for itself. The default value indicates that Windows can use as much of this space as it needs. Try entering a positive value less than 640 if there is not enough memory to run Windows 386 enhanced mode. The default is **-1**. To change this entry, you must edit SYSTEM.INI.

WindowUpdateTime

WindowUpdateTime=*milliseconds*

This entry specifies the time Windows takes between updates of the display for a windowed non-Windows application. The default is **50**. To change this entry, you must edit SYSTEM.INI.

WinExclusive

WinExclusive=*Boolean*

This entry, id **On**, indicates whether Windows always gets all of the computers processing time whenever a Windows application is in the foreground. If this entry is **On** and a Windows application is in the foreground, no non-Windows application will run. The default is **Off**. To change this entry, choose the 386 Enhanced icon from the Control Panel.

WinTimeSlice

WinTimeSlice**=***number,number*

This entry contains two numbers. The first number specifies the relative processing time (based on **MinTimeSlice=**) given to all Windows applications running in the foreground relative to the time allocated to all non-Windows applications running in the background. The second number specifies the relative processing time given to all Windows applications running in the background when a non-Windows application is running in the foreground. The range of numbers that can be entered is **1** through **10000**. The important value is the ratio of these values to the corresponding settings in the PIFs of the active non-Windows applications. The default is **100,50**. To change this entry, choose the 386 Enhanced icon from the Control Panel.

WOAFont

WOAFont**=**font filename

This entry specifies which font files are loaded into memory when running non-Windows applications. These fonts are also available to Windows applications as long as a non-Windows application is running. If you want to make these fonts always available to Windows applications, add them to your system with the Fonts option in Control Panel. The default is **DOSAPP.FON**. To change this entry, exit Windows and run Setup from MS-DOS, then change the Codepage or Display setting.

XlatBufferSize

XlatBufferSize=*kilobytes*

This entry specifies the size of the low memory buffer used to map MS-DOS calls from protected mode to virtual 386 enhanced mode. Specifying a higher value can improve the performance of protected mode Windows applications that read or write information in a large number of bytes (for example, some database applications). However, this may decrease the amount of memory available for running non-Windows applications and may cause problems on networks that use named pipes. These networks may require a value of 4. The value you specify is rounded to the next 4K increment. For example, if you set this value to 5, Windows rounds it up to 8. The default is 8. To change this entry, you must edit SYSTEM.INI.

XMSUMBInitCalls

XMSUMBInitCalls=*Boolean*

This entry specifies whether or not Windows should call the extended memory drivers UMB management routines. The default is usually **On**, but may be **Off** depending on the extended memory manager you are using. You should never need to change this entry.

Default button

The **Default** button selects the items UnInstaller originally suggested for deletion and deselects the items not originally suggested.   Use this button if you make mistakes in choosing items for deletion and you want UnInstaller to redisplay its automatically-selected items.

UnInstall button

The **UnInstall** button starts the actual uninstall process. See *Starting the UnInstall* for details.

Trial Run check box

When checked, the **Trial Run** check box tells UnInstaller to go through the motions of uninstalling the selected application, but without deleting any files and without changing or removing any sections in WIN.INI.   Going through a trial run tells you how many files would have been deleted and how much disk space would have been saved if the files were actually deleted.   If the Create Report checkbox is checked, a report is generated so that you can determine what would have been deleted.

Backup check box

When checked, the **Backup** check box tells UnInstaller that you want a backup copy of the application you are uninstalling.   If you tell UnInstaller to create a backup copy, you can use the **Restore Backup** feature at a later time to restore the application.

Note: If you elect to backup the application, the backup process is performed before any deletions take place.   The completion percentage shown during the backup process is based on the number of items selected for deletion in the list box--it is not based on the number of files to be backed up.

Confirm all deletions check box

When checked, the **Confirm all deletions** check box tells UnInstaller that you want to be prompted to confirm deletion of each and every item you've selected in the list box.

Create Report check box

When checked, the **Create Report** check box tells Un**I**nstaller that you want a report created each time you uninstall an application. (See *Report Setup Primer* for instructions on creating printed reports and a report file.)

Add File... and **Add Section...** buttons

The **Add File...** and **Add Section...** buttons allow advanced users to mark additional files and/or WIN.INI sections that may not be listed by UnInstaller.

Scan button

Choose **Scan...** to find applications that do not appear in the list. (You may need to choose Scan if you use a shell that does not support DDE requests, such as PCTools for Windows.) The **Scan** feature will scan for all Windows executables on the specified drive, show you the icon and file description and allow you to select the application for removal.

Find button

Choose **Find...** if the application you want to uninstall is not listed. Use the find dialog to select the application you want to uninstall. Choose the OK button in the **Find...** window and then choose **Analyze**.

Close button

Choose **Close** to close the **Uninstall Applications** window and return to the main window without selecting an application to uninstall.

Analyze button

When you select **Analyze**, UnInstaller analyzes your system to find all traces of the application you want to uninstall. Every trace that it finds is entered into the **Items to be deleted** list box.

Create Report check box

If you want UnInstaller to create a report, select the **Create Report** option.   If you do not want a report, be certain that the **Create Report** option is not selected.

Trial Run check box

When this check box is checked, UnInstaller will not delete any files or make any changes to files. Use this option until you are sure about what you want to do. Once you are sure, clear the check box and then press the **Cleanup** button to actually remove the files.

Cleanup button

Carries out the actions you have specified.

Display Drivers button

Allows selection and cleanup of OEM display drivers.   For more information, please refer to *Drivers Primer* and *Removing Display Drivers*.

System Support Files button

Suggests support files that may be deleted. For instance, standard mode files that Windows does not use when running in enhanced mode. For more information, please refer to *Removing System Support Files*.

Windows Fonts button

Displays the names of, and allows deletion of, both installed and available fonts. For more information, please refer to *Fonts Primer* and *Removing Fonts*.

Non-Windows Apps button

These are support files required to run DOS programs under Windows 386 enhanced mode. If you never run DOS programs in Windows, you can delete these files. For more information, please refer to *Removing Non-Windows Apps*.

Miscellaneous Files button

Presents a list of various files that you may delete if they are not being used. UnInstaller color codes the files that it suggests for removal. File names that are greyed out are not present on your system. Files displayed in green can be safely removed. If a file's name is displayed in yellow, use caution before telling UnInstaller to delete the file. When you see a file displayed in red, you should delete the file only if you are absolutely sure you don't need it. The files named RAMDRIVE.SYS and EMM386.EXE are shown in red if they are in use. For more information, please refer to *Removing Miscellaneous Files*.

• UnInstaller: Helping Windows take out the trash

UnInstaller comes to the rescue when Windows is loaded down with unnecessary applications.   In our example, UnInstaller would remove the calculator and its instruction manual, freeing up storage space and keeping Windows from wasting time reading unnecessary information.

In computer talk, UnInstaller deletes an applications files from your hard disk   your computers storage cabinet.   This gives you room on the hard disk for other applications. UnInstaller also erases the instructions that applications leave for Windows.

Be sure to read *Smart Uninstalling* for an overview of what UnInstaller removes from your computer.   For more details and a complete explanation of each option, see *UnInstallable Items*.

● System Cleanup: For removing the "extras" you'll never use

The **System Cleanup** feature helps you remove extraneous files that you aren't using. When Windows sets itself up, it installs a large number of files, many of which serve the same purpose, "just in case" you want to use them. Generally speaking, you will only use a few of these files for your preferences and you won't ever use the rest, and so you're left with the "extras" taking up space. **System Cleanup** will show you these files and let you "house clean" your system and unload the "extras," much like you would clean your house by having a garage sale. See *Using System Cleanup* for more details.

• INIClean: For messes made before you had UnInstaller

The **INIClean** feature helps you when youve deleted an application without using UnInstaller.   You should use **INIClean** if the application that you removed (before acquiring UnInstaller) was simple and probably added only an INI file to your computer. If the application was more complicated, you canuse UnInstaller to find *all* remaining traces of the application.   You can also use UnInstaller to remove all remaining traces of an application which you manually deleted.   See *Using INIClean* for more information.

• Duplicate File Finder: The "de-redundancy" treatment

The **Duplicate File Finder** (from now on, we'll call it **DupeFinder**) feature helps you locate and remove extra copies of files that you have on your system. Many times applications use the same file and each application will place copies of the file in various directories. As long as one copy of the file is present in the search path (see *DOS Primer*) the other copies can usually be removed. See *Using DupeFinder* for more details.

⬣ OrphanFinder: Tossing out the leftovers

OrphanFinder is used to view and remove orphaned files. (The definition of an "orphan" is a Windows EXE that is not referenced, or installed, in your shell (desktop) or a DLL that is not used by any of the applications referenced in the shell.) See *Using Orphan Finder* for more details.

● Restore Backup

Restore Backup restores applications that were uninstalled using UnInstaller, with the Backup check box checked.   Restore Backup restore an application's files, sections in WIN.INI and document files to their original location.

Help button

Help is always available when using UnInstaller by pressing the **<F1>** Key, the Help button, or choosing from the Help menu.

Current Drive drop down list

Use this list to select the drive you want to search for duplicate file names.   To select a different drive than the one shown, click on the down arrow and scroll through the list until you find the desired drive, then click on the desired drive.   The list will close and the desired drive will be displayed as the current drive.

Note: If you have the single-user version of UnInstaller, **DupeFinder** will not search network drives   it searches only local drives.

Search button

Begins searching for duplicate files.

Note: This process may take a few minutes to complete. The length of time depends on the number of files on the drive as well as the number of duplicates found.

View button

Note: This button appears only after duplicate files have been located. This button is overlayed by the **Cancel** button when appropriate

Views the contents of the selected (highlighted) file. If more than one file is selected, the last one selected is displayed.

UnInstaller understands approximately 20 file types. If UnInstaller does not recognize the file type, it will default to a hexadecimal dump of the file.

If the viewer window is open, you may also drag files from Windows File Manager and drop them onto the viewer window.

Cancel button

Stops the current task. This button is enabled at several points during the search process, which enables you to cancel the process. (This button overlays the **View** button when appropriate.)

Delete File button

Deletes the selected file(s) from the drive.

Close button

Closes **DupeFinder**.

Confirm Each Deletion check box

Check this box to require confirmation before deletion of any file.

Use file map from (date) check box

You can use or re-use the current map file by checking the box **Use file map from (date)** and pressing the **Search** button.   This is helpful when you've deleted one or more duplicate files and you want to know what duplicates remain on your disk.

Previous Page and Next Page arrows

If you have a large number of duplicate files on the selected drive, and all the file names won't fit into memory at once, these buttons become enabled and you can press them to scroll through the list. (Note These arrows are located under the **Current Drive** drop-down list.) If all duplicate file names are present in the list box, these buttons will not be visible.

EXEs Only check box

Check the **EXEs Only** check box for the fastest operation. But, beware that Orpan Finder will not find DLLs from executables that have already been deleted. If you wish to view DLLs *and* EXEs, then leave **EXEs Only** unchecked.

Current Drive drop down list

Use this list to select the drive you want to search for orphaned files. To select a different drive than the one shown, click on the down arrow and scroll through the list until you find the desired drive, then click on the desired drive. The list will close and the desired drive will be displayed as the current drive.

Search button

Begins searching for orphaned files.

Note: This process may take a few minutes to complete. The length of time depends on the number of files on the drive.

UnInstall button

Begins the uninstall for the selected orphan.

Close button

Closes **Orphan Finder**.

Load Existing Orphans button

Once youve removed orphans and updated the listings of your recognized applications for the first time, selecting **Load Existing Orphans** will continue to use your listings of recognized applications.   As long as you remove any future applications using UnInstaller, you should not need to create a new map file of orphans in your system.   If for some reason, you wish to do a fresh scan, pressing **Search** will create a new map file of any orphans found, however any additions you have made to your recognized applications list will be lost.

Scan Shell button

To view a listing of application files that are referenced or installed in your shell, press the Scan Shell button.   All applications recognized by your shell program will appear in the list.

Description

A short description of the selected application. This is the description the developer of the application inserted into the executable program file.

Add File button

If you find that an application you use is *not* listed, you may add it to the list of referenced applications by pressing the Add File... button and selecting a file to add.

Orphaned Files list

This is a list of orphaned files on the selected drive.

Shell Applications list

The list box generated as a result of the **Scan Shell** operation. These applications are referenced and installed in the shell (desktop) and will not be listed as orphans.

Backup list

A list of backup files you have created previously that are still present on your hard drive.

Restore button

If you find that you have removed an application in error, you may restore the application by selecting it from the list and pressing this button.

Delete File button

Once you have determined you no longer need the backup for the selected application, press this button to delete the backup file from your hard drive.

Close button

Closes the Restore Backup window.

Description

A short description of the selected application. This is the description the developer of the application inserted into the executable program file.

Output to file and **Output to printer** check boxes

Check the **Output to file** box if you would like to create a report file or the **Output to printer** box if you would like to print out the report automatically.   Check both boxes if you would like both options.

Add to end of file check box

Select **Add to end of file** to add the new report information to the end of an existing file.

Overwrite file check box

Select **Overwrite file** to overwrite a previous report's information with the new report information.

Select File button

If you would like to select a different file than the one specified for the report output, press the **Select File** button.

File name

The name of the report file you have selected for output.

Selected printer

The printer you have selected for output.

OK button

Press the **OK** button when you are through making changes.

Cancel button

Press the **Cancel** button if you have made changes but decide you do not want the changes to take effect.

Print Setup... button

Press this button to bring up the **Print Setup** dialog.

Change Directory button

Displays a dialog that lets you change the directory to search for INI files.   The default is the directory where you have Windows installed.

Return to Windows Directory button

Returns to the directory where you have Windows installed.

Display frame

Choose what you would like to work with. **List of INI Files** lets you specify the INI file. **Sections in WIN.INI** lists all the sections in the WIN.INI file. **Associations in WIN.INI** lists the file extensions and the programs with which they are associated. **Entries in SYSTEM.INI** lets you view the lines in SYSTEM.INI, exactly as they appear in the file.

Which Ones frame

You may choose between listing **ALL** files, files that are **New Since** a certain date and **Non-Standard** files.

Close button

Click to close the **INIClean** window.

View File... button

Click to view the file you've selected from the list.   You may not edit the file when using this selection.

Edit INI File button

Choose this button to edit the INI file you have selected.

Delete File/Item button

Click this button to delete the file or item that is currently selected in the list box.

Exit

Select this option to shut down UnInstaller.