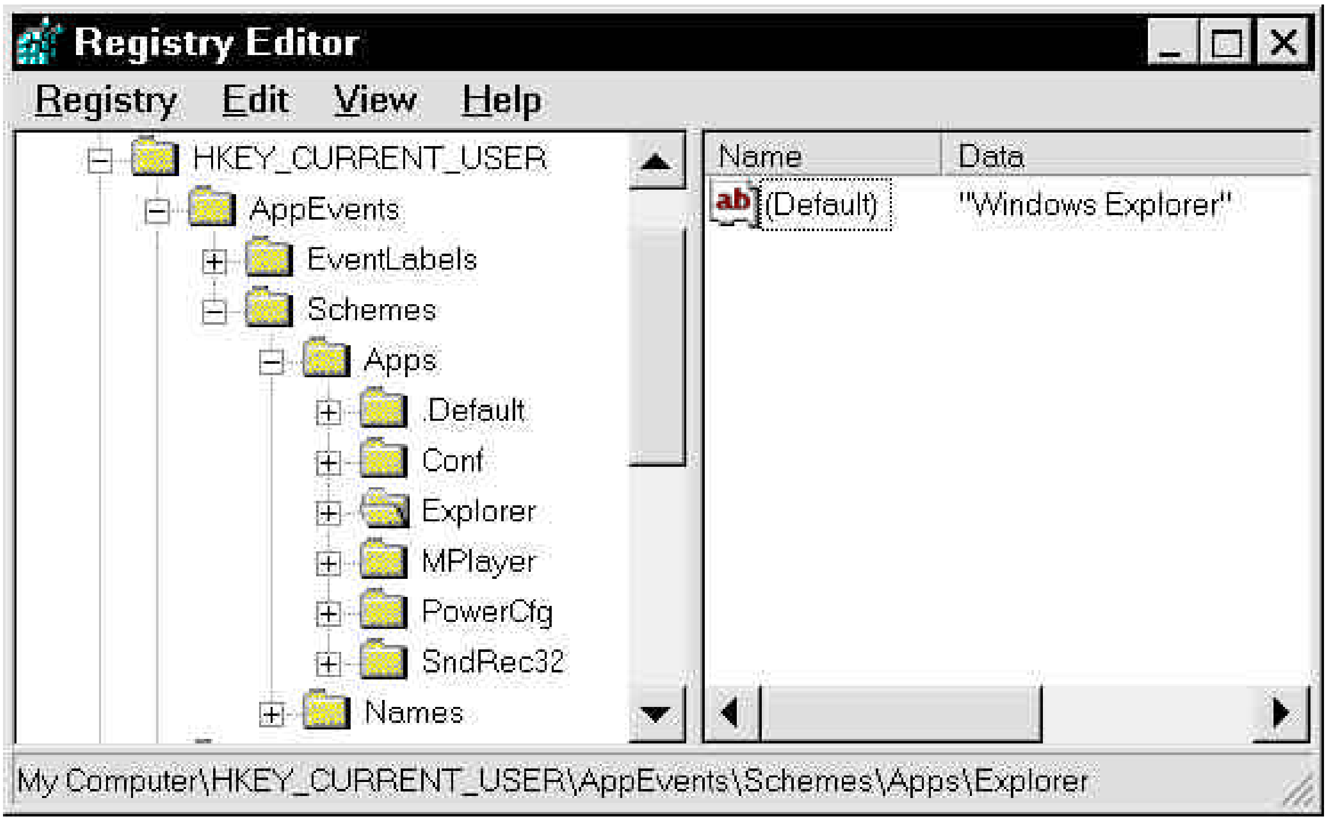
Speed it up!

Windows-Tuning

with the Registry



Name	Data
ab (Default)	"Windows Explorer"

My Computer\HKEY_CURRENT_USER\AppEvents\Schemes\Apps\Explorer

www.KnowWareGlobal.com

Windows Tuning with the Registry

Acrobat Reader: How to ...

F5/F6 open/closes bookmarks - **F4** open/closes thumbnails

In menu View you can set, how the file is displayed

CTRL+0 = Fit in Window, **CTRL+1** = Actual size, **CTRL+2** = Fit width

You can set **SINGLE PAGE**, **CONTINUOUS VIEW** or **CONTINUOUS FACING**

.. try them out and you will see the differences.

Navigation

ARROW LEFT/RIGHT: forward/backwards one page

ALT+ARROW LEFT/RIGHT: same as in a browser: forward/back

CTRL++ zooms in **AND CTRL +-** zooms out

www.knowwareglobal.com

The Registry – I'd rather leave *that* alone!

Have you thought this at least once before? You've heard advice like "it's better to keep your hands off it"? Such statements are understandable. The very thought that the Registry holds all the settings for the operating system in Windows 95 and 98 is enough to make your stomach churn. The fear that you could paralyze your whole system with one false move is not unjustified, but with this booklet in hand, you won't make any mistakes.

What's the purpose of this booklet?

"No one does acrobatic exercises without first having spanned the safety net." The wise man who said this (who was, no doubt, an insurance salesman ☺) was absolutely correct. Here I'll show you exactly how to work with the Registry. Don't be afraid if something goes really wrong; that's what the "Troubleshooting -- Nothing's Working Anymore" section is for. There I'll show you step by step how to bring your system back onto its feet. So enjoy peace of mind while you learn about the crucial role that the Registry plays in Windows 9x. Maybe you don't even know yet exactly what the "Registry" is. That's no problem at all -- quite to the contrary. For, like all KnowWare authors, I've set myself the goal of explaining sometimes truly complex computer problems to beginners.

Even if you're already a bit familiar with the topic, you can use this booklet to learn a lot about your operating system. With the many tips and tricks that it contains, you'll find a wealth of practical knowledge and many opportunities to take advantage of hidden functions of Windows. In addition, I'll introduce you to various programs that can make your work with the Registry easier and eliminate the need for manual intervention in the Registry database. Nevertheless, you should learn what the purpose of the various entries in the Registry is and how you can customize them to suit your particular needs.

Prerequisites

Before you plunge into the depths of the Registry, you should have some basic knowledge about Windows 95/98: you should know how to use the mouse and keyboard, what files and directories are, and how to copy or delete them.

The Basics

What is the Registry?

In a nutshell, the Registry is the system databank of Windows 95/98. It contains all settings, from the sound card to Winword to mouse configuration. An operating system that is as advanced and comfortable for the user as Windows is (aside from a few flaws -- but nobody's perfect and nobody's software is either) must keep track of an almost unimaginable amount of data. Which wallpaper does the user want? How fast should the double-click of the mouse be? Which program does the user prefer for word processing and what's their favorite font? Does the user want to see 16, 256, or even more colors? Who manufactures the user's printer and do they have a scanner too…

As you can see, one can ask question after question. Windows has to remember all this and much more too; the settings can change every day. Maybe several people are working on one computer and they all prefer different settings. The operating system must keep track of each person's preferences separately.

Reconciling all this information requires artistry. Keeping all these settings so that one can still see the "big picture" and so that the user can understand and edit them in a databank requires a veritable miracle.

Microsoft's Warning

Apparently Microsoft doesn't like it when end users play around with the Registry. Therefore, the Registry Editor is not usually found on the Start menu. In addition, the general notices that Microsoft includes with the operating system in the form of various text files, contains the following warning:

"Manual manipulation of the Windows 95 Registry can have serious consequences for your Windows 95 installation. Such consequences can range from the malfunctioning of individual components to the complete paralysis of your system and may lead you to reinstall Windows 95. Microsoft provides no guarantee and assumes no responsibility for supporting problems caused by manipulation of the Windows 95 Registry. Use the Registry editor (REGEDIT.EXE) or other similar tools for manipulating the Windows 95 Registry at your own risk."

This warning is doubtless well-intentioned and it is quite correct, yet it scares inexperienced users away. You as the reader of this booklet should not let it intimidate you. In order to prevent accidents, you should definitely read the section entitled "Securing the Registry" below.

The fact that Microsoft delivers no documention whatsoever about the Registry emphasizes the point that the end user should stay as far away from the system databank as possible. But "no documentation whatsoever" isn't completely correct, as I'll show you in the following tip:

Hidden Information

Only in the Microsoft Resource Kit, which you'll find in the \ADMIN\RESKIT\HELPFILE on your Windows 95 CD, is there any documentation about the Registry. It's best if you copy the files WIN95RK.HLP and WIN95RK.CNT into the directory C:\WINDOWS\HELP.

If you'd like to add these files to the Windows 95 Help system, edit two lines of the file WINDOWS.CNT to read as follows:

:Index Resource Kit =win95rk.hlp and

:include win95rk.cnt

The first line belongs underneath the last entry, which also begins with "Index." You can add the second line to the end of the file.

If you're using Windows 98, you install the Resource Kit by calling up the file SETUP.EXE from the directory \TOOLS\RESKIT on your Windows 98 CD. Then you'll find the Help file by choosing **Start, Programs, Windows 98 Resource Kit, Resource Kit Online Book**.

The History of the Registry

It wasn't always the case that Windows had to keep track of as many configuration details as it does today. For example, in Windows 3.1, there were only about a third as many settings as in Windows 95, but then again, at that time Windows was not as capable as it is today. As the comfort of the graphical user interface increases, the administrative overhead grows exponentially. Internet Explorer 4, which Microsoft built into Windows 95 and which they sold, along with some additional new features, as a completely new operating system (Windows 98), adds even more settings that Windows has to keep track of.

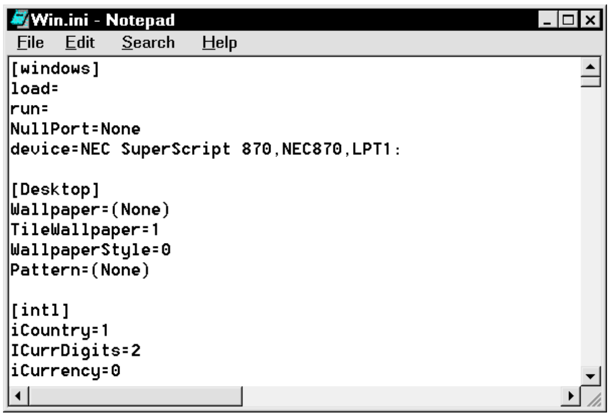
INI-Files – What are those again?

In Windows 3.1, almost all the settings were saved in text files. These became known as INI files because they had the extension INI. The two best known among these were WIN.INI and SYSTEM.INI. These two together held nearly all Windows settings, for example the number of colors available, the monitor resolution, wallpaper, and the installed fonts.

Nearly every program had its own INI file, in which it stored its own settings. Depending on the application, these were called WINWORD.INI or MPLAYER.INI.

The Structure of INI Files

The INI files were structured quite simply: the category names were written in square brackets, kind of like a heading for a block of text. There were categories such as [Windows], [Fonts], or [Microsoft Word]. The rest of the file consisted of so-called keyword pairs. In each line there was a key (a word that served as a name for a Windows function) and an equals sign followed by a designated value. The form of these lines often looked like this: "NumberColors=256". With this example, Windows would know that it was supposed to work in 256 colors.

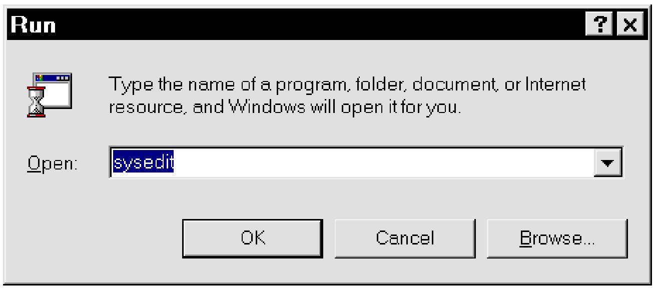


```
[windows]
load=
run=
NullPort=None
device=NEC SuperScript 870,NEC870,LPT1:

[Desktop]
Wallpaper=(None)
TileWallpaper=1
WallpaperStyle=0
Pattern=(None)

[intl]
iCountry=1
ICurrDigits=2
iCurrency=0
```

Illustration 1: A classic WIN.INI file from Windows 3.1

Editing INI Files

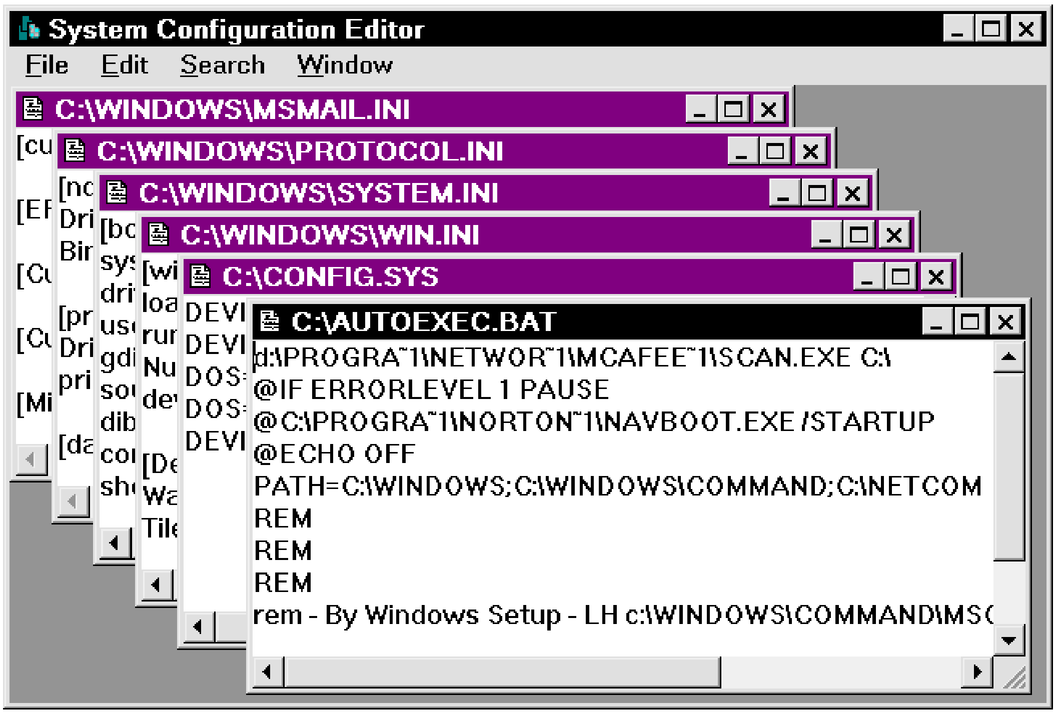
As you'll learn in the course of this booklet, most configuration settings today are stored in the Registry. But if you should still need to edit one of Windows' two crucial INI files (WIN.INI und SYSTEM.INI), you should know about the Windows System Editor. You'll find this little tool if you choose **Start, Run, Sysedit**.



Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Illustration 2: Here's how you start the handy System Editor

The program will start and automatically load up the crucial WIN.INI and SYSTEM.INI files.

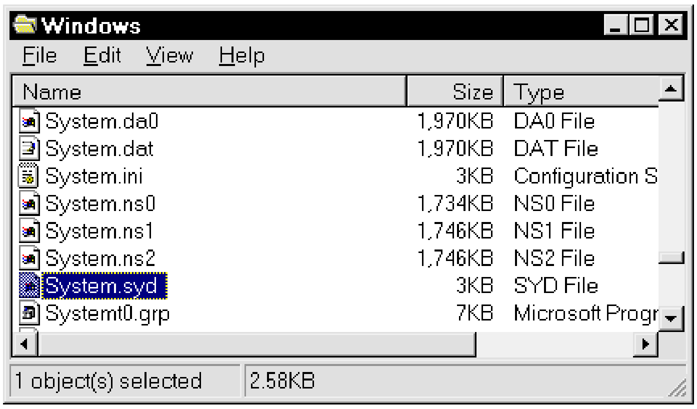


```
d:\PROGRA~1\NETWOR~1\MCAFEE~1\SCAN.EXE C:\
@IF ERRORLEVEL 1 PAUSE
@C:\PROGRA~1\NORTON~1\NAVBOOT.EXE /STARTUP
@ECHO OFF
PATH=C:\WINDOWS;C:\WINDOWS\COMMAND;C:\NETCOM
REM
REM
REM
rem - By Windows Setup - LH c:\WINDOWS\COMMAND\MS(
```

Illustration 3: The System Editor loads up many crucial system files

The best thing is that as soon as you've made changes to one of the files and saved it, Sysedit will automatically create a backup copy of the file with the extension SYD. If, for example, you edit SYSTEM.INI and save the new version, you'll find the old version in the WINDOWS directory under the name SYSTEM.SYD.

Beginning with the Registry



Name	Size	Type
System.da0	1,970KB	DA0 File
System.dat	1,970KB	DAT File
System.ini	3KB	Configuration S
System.ns0	1,734KB	NS0 File
System.ns1	1,746KB	NS1 File
System.ns2	1,746KB	NS2 File
System.syd	3KB	SYD File
Systemt0.grp	7KB	Microsoft Progr

1 object(s) selected 2.58KB

Illustration 4: If you've made changes to the SYSTEM.INI file, Sysedit has created a backup copy for you

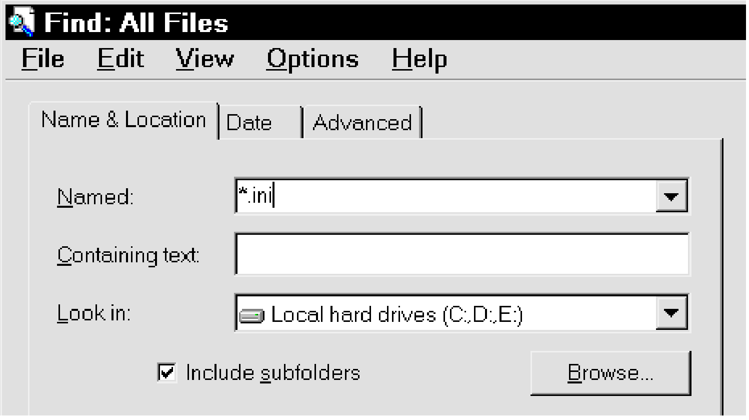
In addition to the INI files, there was a crude ancestor of today's Registry in Windows 3.1: a file called REG.DAT. This file was used for software administration, for example to save information for a data transfer from Program A to Program B. But, as I told you above, all other settings were saved in the files WIN.INI and SYSTEM.INI.

The principle of REG.DAT was expanded under Windows 95. Starting immediately, all settings were to be administered through the Registry. But not really *all* settings! So that older programs, namely ones that were written for Windows 3.1, could function under Windows 95/98, the two files WIN.INI and SYSTEM.INI were carried over to the new system. If you installed an older program, it didn't write its settings to the Registry. No, as in days gone by, most of the entries ended up in WIN.INI. But it was even worse: the older programs didn't even give up their own INI files.

Unfortunately, even the programmers of modern 32-bit software (software for Windows 95/98/NT) didn't honor Microsoft's recommendation that all settings should be saved to a particular part of the Registry.

So you can see: in the Registry there is a divide between the theory on Microsoft's pages and the practices of worldwide computing. This is why if you examine your hard disk, you'll still find so many INI files.

Finding and Deleting Unnecessary INI Files



Named: *.ini

Containing text:

Look in: Local hard drives (C:,D:,E:)

☑ Include subfolders

Illustration 5: Faster than the police allow -- on the trail of INI files

Try a little experiment; choose **Start, Find, Files/Folders** to call up the **Find** dialog box. Under **Named** enter the search criterion ***.ini** and then begin the search with a click on the **Start** button.

There were an astonishing 449 of these INI files on my hard disk; if they bear the name of a long-deleted program, you can go ahead and delete the INI files. If you're not sure of yourself, you should save the file, since they are small and surely not the reason your hard disk always seems too small. In case of emergency you can copy the file to a diskette; then, in case a program gives you an error message, you can just copy the file back to the hard disk again.

Advantages of the Registry Principle

You already know that the INI files have a very simple structure. With this simple structure, which in the end has only two levels, the administration of many and complex pieces of information becomes difficult. And then add to this the limitation under Windows 3.1 that INI files couldn't be any larger than 64 KB, which created problems, especially in the case of WIN.INI.

Under Windows 3.1, creating different user profiles was very complicated. In order to maintain different configurations side-by-side, one had to take the time to copy the INI files with the different settings from one place to another before starting Windows. Some of you users may shudder as you remember files such as WIN.INI, WIN.002, and WIN.003. For each configuration there had to be a separate file.

These problems are foreign to Windows 95/98 with its Registry system. The Registry database is well thought through and clearly structured. Thanks to its hierarchical structure, the system can be expanded to just about any imaginable depth, just like the organizational structure of hard disks. These, with their structure of directories and subdirectories, can accommodate level after level of organization. In the Registry, settings are no longer stored as pure text. Sometimes, binary values (columns of zeroes and ones) are used since they can be processed more quickly. But we'll come to the structure of the Registry somewhat later on.

Such complexity and possibility for expansion also have their price. Beginners have a harder time with the Registry than with the simple text-based INI files. And the expenditure of energy for programmers of application software is also greater, which explains why -- as already mentioned -- some 32-bit programmers still use INI files.

The Structure of the Registry

The Registry consists of the two files USER.DAT und SYSTEM.DAT,which without a doubt are the two most important system files under Windows. If one of these files is missing, the whole system will be lamed.

The Difference between System.dat and User.dat

Maybe you'll find yourself asking now: "What's this all about? Why did Microsoft divide the Registry between two files?" This was done for the following reason: Windows supports user profiles, which means that several users can work on one computer. When someone starts the computer up, they type in their name and password. Then they can configure the computer according to their particular preferences (wallpaper, etc.). When the user enters their name, Windows knows which user is in front of the computer and makes the corresponding settings available.

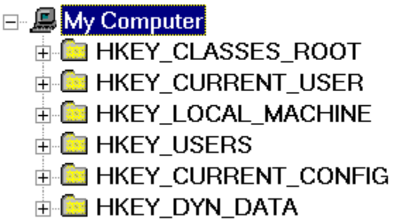
The hardware settings and registry entries of installed software remain the same regardless of which profile is in use; this is why these entries are placed in one file, SYSTEM.DAT. Other characteristics, for example the current wallpaper, vary from user to user, that is, from profile to profile. These variable settings are stored in a second file, called USER.DAT.

SYSTEM.DAT stays the same for all users, that is to say profiles, for they are working on the same computer. By contrast, each user has their own USER.DAT with their own user-specific settings. You'll find this file in the directory C:\WINDOWS\PROFILES\USERNAME.

You'll get to know the inner structure of the Registry if you call up the Registry Editor by choosing **Start, Run, Regedit.** A wide, divided screen will appear, which hardly gives you a sense of the possibilities and depth of the system. You'll see just six groups, which are organized underneath the entry "Desktop."

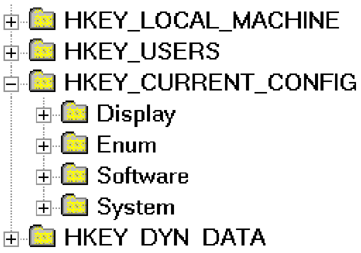
(Homekeys) of the Registry

The six main groups are divided further. You'll see this if you double-click on one of the so-called Homekeys (HKEY). If you do this, several subordinate entries will open up, which in turn branch off again.



My Computer
- HKEY_CLASSES_ROOT
- HKEY_CURRENT_USER
- HKEY_LOCAL_MACHINE
- HKEY_USERS
- HKEY_CURRENT_CONFIG
- HKEY_DYN_DATA

You should think of the whole thing like the directory structure on a hard disk. The Homekeys are like the topmost directories on the hard disk and there are subdirectories underneath these. If you compare the opened-up Registry Editor with the Explorer and its directory tree, you'll see the visual similarity. You can open up the groups just like directories in the Explorer.



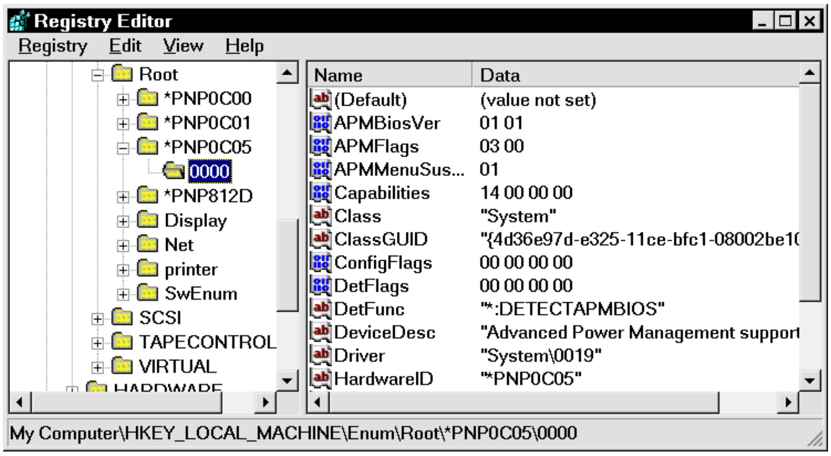
- HKEY_LOCAL_MACHINE
- HKEY_USERS
- HKEY_CURRENT_CONFIG
  - Display
  - Enum
  - Software
  - System
- HKEY DYN DATA

The „Address“ of a Registry Key

When you specify the position of a file on the hard disk, it might appear in the form C:\WINDOWS\HELP\MPLAYER.HLP. You can specify the position of a particular entry in the Registry in exactly the same way:

HKEY_CURRENT_USER\Control Panel\desktop\CursorBlinkRate.

The categories that are subordinate to the Homekeys, in particular the last entries of the Registry tree, are called "Keys." These always appear in the left-hand window.



- Root
  - *PNP0C00
  - *PNP0C01
  - *PNP0C05
    - 0000
  - *PNP812D
  - Display
  - Net
  - printer
  - SwEnum
- SCSI
- TAPECONTROL
- VIRTUAL
- HARDWARE

Name	Data
(Default)	(value not set)
APMBiosVer	01 01
APMFlags	03 00
APMMenuSus...	01
Capabilities	14 00 00 00
Class	"System"
ClassGUID	"{4d36e97d-e325-11ce-bfc1-08002be1(
ConfigFlags	00 00 00 00
DetFlags	00 00 00 00
DetFunc	"*:DETECTAPMBIOS"
DeviceDesc	"Advanced Power Management support
Driver	"System\0019"
HardwareID	"*PNP0C05"

My Computer\HKEY_LOCAL_MACHINE\Enum\Root\*PNP0C05\0000

Illustration 6: You'll recognize character and binary/dword entries by the distinctive icon that precedes them

In the right-hand window of the Registry Editor you'll find the dependent entries, which according to their type are called "binary," "string," or "DWORD." These three entries define a setting since their values determine the result they produce. Note that there is no obvious principle defining when one stores Windows settings as "binary," "string," or "DWORD."

The organization of value names and the values themselves is similar to the INI files; if you remember, there I used the example „NumberColors=256."

In the Registry you'll find, for example, the label "CursorBlinkRate," which defines the blink frequency of the cursor. All the way to the right, that is, next to the label "CursorBlinkRate," you'll find the corresponding value, for example 500. In your head, you can add an equals sign just like in the INI files. If you're thinking this way, you'd have the sequence "CursorBlinkRate=500". Even if you can't see the equals sign in the Registry Editor, adding it in your head may help you make a better connection between the string and the value.

From the relationship string/value, it follows logically that with a key and a string alone (or a DWORD or binary value), one cannot create a setting. The deciding factor is the assigned value, which you'll find all the way to the right in the Registry Editor.

Overview

In the Help file in the Microsoft Resource Kit, to which I've already alluded, you'll find a graphic that provides a visual explanation of the structure of and the difference between the two files. There's the right side with the Homekeys **HKEY_CURRENT_USER** und **HKEY_USERS**, which depicts the entries in USER.DAT. Then there's the left side, which contains the settings that remain the same for all users, which are stored in SYSTEM.DAT.

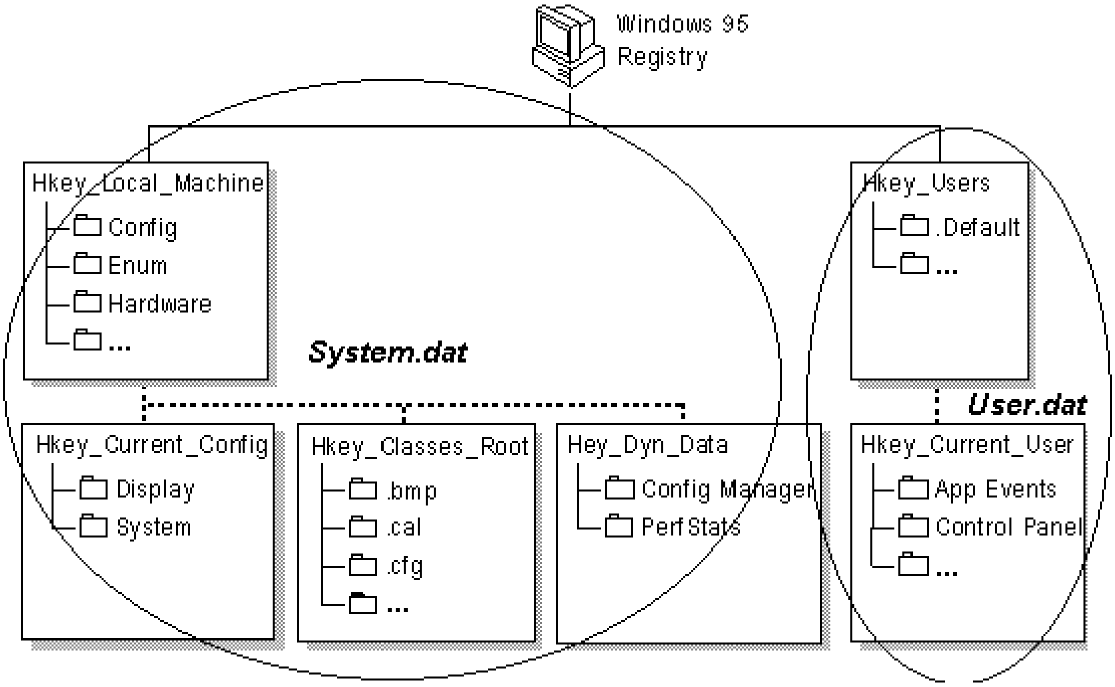


Illustration 7: The MS Resource Kit explains the approximate structure of the Registry

Surely in the restricted space of this booklet, one couldn't compile a complete reference to all the keys in the Registry. But if you're interested, "surf" through the various branches of the Registry databank and you'll be struck by the great variety of the entries and keys. Many of these entries consist of columns of numbers that are hardly comprehensible to the average user, who isn't supposed to make any drastic changes to them. Other entries and whole branches are, by contrast, very interesting, since, with a few small changes one can customize Windows' behavior to suit personal taste. So that you have some idea what the individual main keys for settings are storing, now I'll provide you with several general explanations.

HKEY_CLASSES_ROOT\

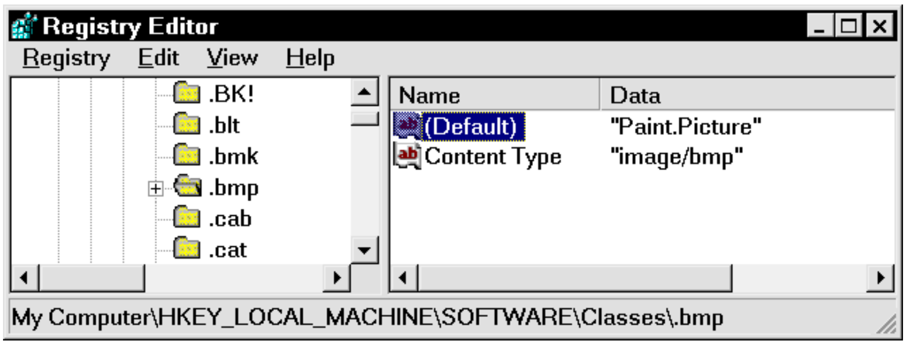
The first main group is the one headed by the main key **HKEY_CLASSES_ROOT**. Windows uses this group to administer all file types. Here, for example, is where the setting that determines that files of the type BMP will be opened with the Windows "Paint" program by default is stored.

Since there are so many different file types, the list of subordinate keys for **HKEY_CLASSES_ROOT** seems very long. If you install additional programs, this list may get even longer. The advantage of the type registry is that if you're working in Explorer or on the Desktop, you only need to double-click on a BMP file in order to open it with a program that can recognize, read, and process it. If the Registry didn't have this feature, you'd have to start up an application and call up the file using **File, Open** each time you wanted to work with it.

Windows' understanding of files relies on a long route through the Registry databank. Therefore, I'll make a short but enlightening "detour":

On Windows 95's Understanding of Files

If you double-click on a file with the extension BMP, Windows will search for a link to an application in **HKEY_CLASSES_ROOT\.bmp**. But such a link isn't to be found there; there, bitmaps (BMP files) are only classified as "Paint Pictures."



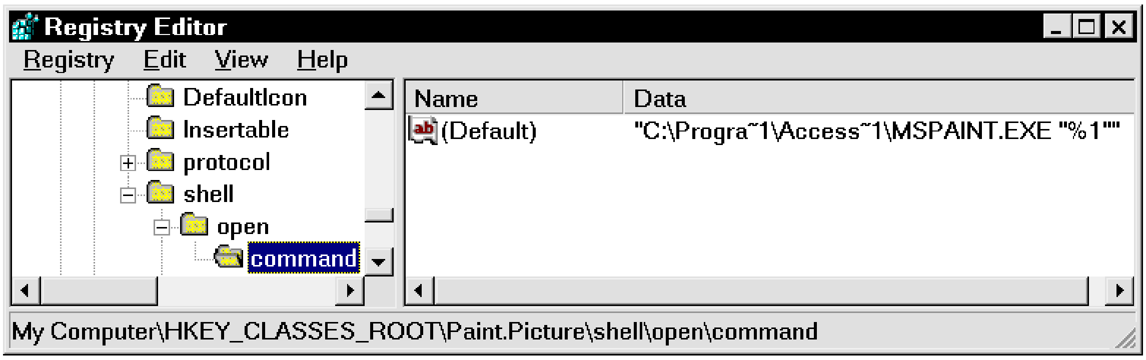
Name	Data
(Default)	"Paint.Picture"
Content Type	"image/bmp"

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Classes\.bmp

Illustration 8: BMP belongs to the group "Paint.Picture"

So Windows searches for more information about "Paint.Picture" and finds a key with the same name. Under the key **HKEY_CLASSES_ROOT\Paint.Picture\shell\open\command\**, it will find a link to the program C:\PROGRA~1\ACCESSORIES\MSPAINT.EXE %1

At first glance, the detour to the key **\Paint.Picture** appears nonsensical, but things are done this way for the following reason: all file types that Paint can process must first be stamped with "Paint.Picture." With this stamp, actions like "open" and "print" can be assigned application-specific instructions. Each graphic-manipulation program other than Paint can read many different filetypes (BMP, GIF, TIF, JPG and PCX). The same path must be blazed and noted for each filetype. This stamping method saves the path just once, which simplifies administration and saves disk space.



Name	Data
(Default)	"C:\Progra~1\Access~1\MSPAINT.EXE "%1""

My Computer\HKEY_CLASSES_ROOT\Paint.Picture\shell\open\command

Illustration 9: The "Open" and "Print" commands in "Paint.Picture"

All other settings that relate to data exchange via OLE (Object Linking & Embedding) are saved under **HKEY_CLASSES_ROOT**. Only with these settings can one embed Excel diagrams in Winword or insert a Paint graphic into an Access file.

HKEY_USERS\

The group **HKEY_USERS** administers all user-specific settings. The settings for each user (if you've set up more than one user) are stored separately according to profile. These profiles contain, among other things, the settings for the wallpaper, sound and color schemes, screen saver, etc. Here you'll also find the user-specific settings for installed software. If you're not working with different user profiles, you'll only see the subkey **.default** under this main key. This is where all settings for a newly-registered user will be saved. If there are multiple users on a single computer, this is where you'll find a list of the individual profiles with their individual settings.

HKEY_CURRENT_USER\

The group **HKEY_CURRENT_USER** is actually part of the group **HKEY_USERS**. **HKEY_CURRENT_USER** keeps track of the current user's settings; it's a kind of mirror. The assignment of a separate homekey for the current user makes a lot of sense. If the current user changes settings such as the wallpaper, for example, Windows or the corresponding program first saves this information under **HKEY_CURRENT_USER**.

Once this has happened, the new settings are saved to the current user's profile in **HKEY_USERS**. This ensures that the changes will only take effect on the current user's desktop and not on other users' desktops. The applications know nothing about profiles; they always save their settings under **HKEY_CURRENT_USER**.

If you haven't set up different users and therefore have no user profiles, the contents of **HKEY_CURRENT_USER** will be identical to the entries in **HKEY_USERS\.default**.

HKEY_LOCAL_MACHINE\

The main group **HKEY_LOCAL_MACHINE** contains all information about installed hardware. All saved settings are the same for all users; these are administered separately from any existing user profiles. Furthermore, this key contains all software settings that cannot vary from user to user.

HKEY_LOCAL_MACHINE\SOFTWARE

The subkey **\SOFTWARE** is an especially important part of **HKEY_LOCAL_MACHINE\**. It contains the registries of all programs sorted by producer. The category **Windows\CurrentVersion\** is of particular interest since it controls the behavior of the operating system and contains important settings that come into play if one wishes to customize Windows.

The subkey **HKEY_LOCAL_MACHINE\SOFTWARE\Classes** has a special meaning: this is the source for the general main key **HKEY_CLASSES_ROOT\**. The reason for double-saving settings has to do with Windows 3.1 and older applications. Windows 3.1 applications stored specific settings in the the Ur-Registry. These only had the main key **HKEY_CLASSES_ROOT\** at their disposal. If you install old Windows applications under Windows 95/98, these programs write their registry precisely in this branch. Therefore, the reasons for the mirroring tactic have to do with compatibility.

Microsoft recommends to software producers that they have their programs for modern Windows software write their entries to the Registry databank in the following form: **HKEY_LOCAL_MACHINE\Software\Producer Name\Product Name\Version**.

HKEY_CURRENT_CONFIG\

The mirroring principle also applies to **HKEY_CURRENT_CONFIG\**. If you haven't set up any additional hardware profiles, "CurrentConfig" (that is, the current configuration) is identical to **HKEY_LOCAL_MACHINE\Config\0001**.

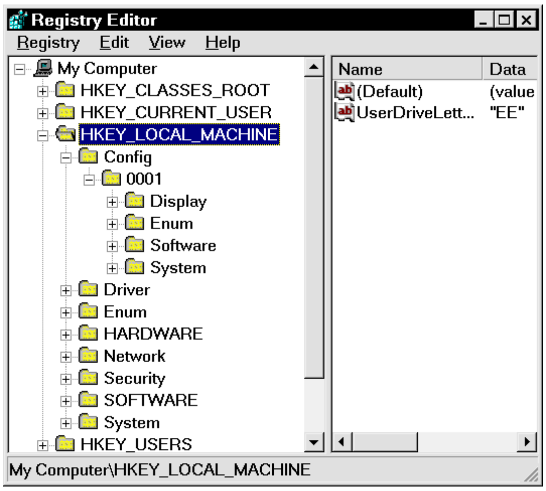


Illustration 10: The source..

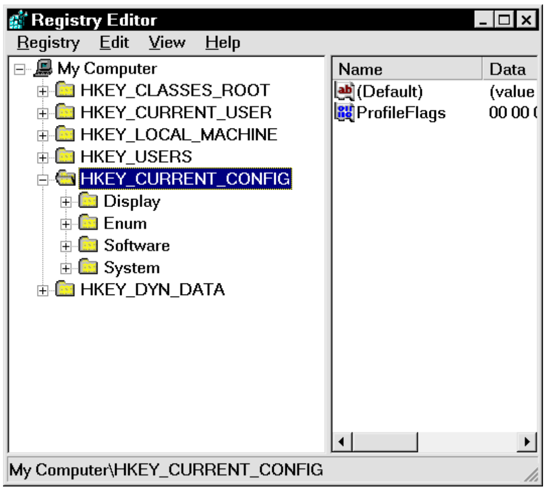
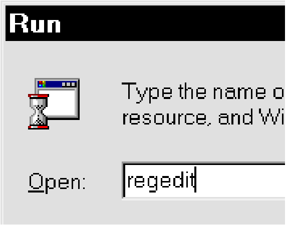


Illustration 11: ... and the mirrored portion

HKEY_DYN_DATA\

As its name suggests, this main key administers so-called "dynamic data." These are primarily data about (Plug&Play) hardware that are accessed so often that Windows stores them in memory while the computer is on. The subkey **\PerfStats** stores data for the Windows System Monitor, a tool that allows you to keep an eye on the actual performance of your processor, among other things. (If you haven't installed System Monitor, you can do this by choosing **Start, Settings, Control Panel, Add/Remove Programs, Windows Setup, Accessories, Details**. You can start it up by choosing **Start, Run, Sysmon**.)

Getting Around in the Registry Editor



Run

Type the name o
resource, and Wi

Open: regedit

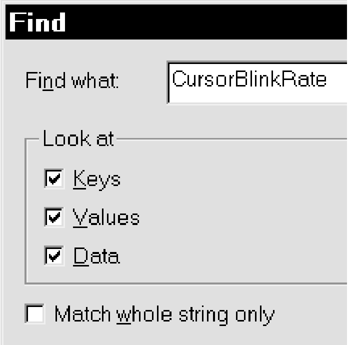
Illustration 12: This is how you call up the Registry Editor

As I've established through experience, it's impossible to avoid the Registry if you really want to customize Windows to suit your particular needs and tastes.

It's well-known that many roads lead to Rome, and so it is also with the Registry. The first way: call up the **Start Menu** and choose **Run**, then type in the command **Regedit**.

The second alternative: since you'll doubtless need the Registry Editor often while you're working your way through this booklet, you should consider creating a shortcut on the Start menu. To do this, open Explorer and find the file REGEDIT.EXE, which should be in the \WINDOWS folder. Click the filename while holding down the right mouse button, then pull REGEDIT.EXE over to the Start button and let the mouse button go.

Searching in the Registry



Find

Find what: CursorBlinkRate

Look at

☑ Keys

☑ Values

☑ Data

☐ Match whole string only

Illustration 13: Without the Search feature, you'd be lost in the Registry

In the Registry Editor, you can use **Edit, Find** or **Ctrl+F** to search for entries. Enter the term you'd like to search for in the dialog box that appears.

In some cases it makes sense to limit the search a bit. For just this purpose, there are various search options available to you. For example, you can limit your search to keys if you're looking for a particular key. If you remember that you've seen a particular setting somewhere near a key named **\Microsoft**, enter the search term and activate the options "Keys" and "Match whole string only." Make sure that the boxes next to "Values" and "Data" aren't checked. Now the chances are pretty good that you'll find what you're looking for. If the "Values" and "Data" boxes were checked and if you hadn't limited the search with "Match whole string only," you'd have had to search through hundreds of results.

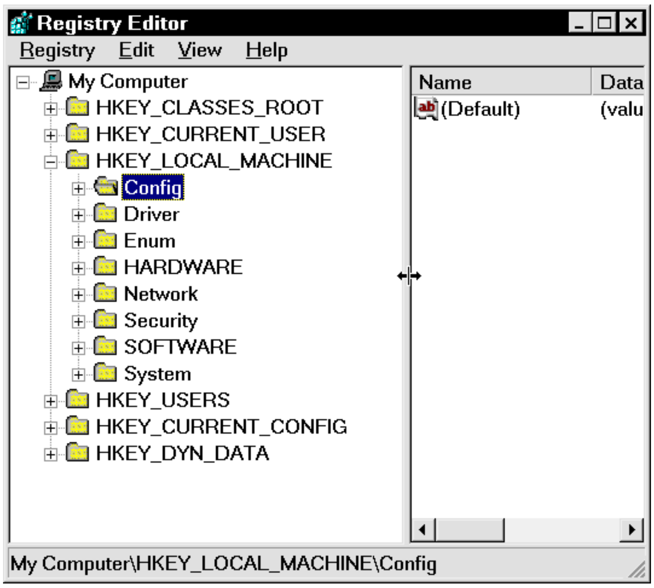
{MM: Please note that I amended this text since the German doesn't apply to the English version of the Registry Editor.} Use the two other options to find "Values," the entries all the way to the right in the Editor, or "Data," the entries in its middle portion, where you'll find the names of strings, binary and Dword values.

By the way, don't lose heart if searches seem to take a long time. I don't understand why Microsoft didn't build in a progress meter as in the installation programs. In any case, your computer hasn't crashed; the searches take a minute or two even on a machine with a very fast CPU since there are so many entries in the Registry.

Tip: if you'd like to get an impression of the variety of keys in the Registry, try clicking the multiplication button on the numerical keypad. Now the Registry Editor will page through all levels, which can take a few minutes.

When you've entered your search term, you can start searching with a click of the **Find Next** button. This will close the Find dialog box and the search will begin. If Windows finds a key or value that matches what you're searching for, it will stop the search and mark the place. Depending on what you're searching for, there might be multiple places where the term that you entered occurs. Make Windows continue the search by pressing **F3**.

There's Room There for the Keys!



```
⊟ My Computer
   ⊞ HKEY_CLASSES_ROOT
   ⊞ HKEY_CURRENT_USER
   ⊟ HKEY_LOCAL_MACHINE
      ⊞ Config
      ⊞ Driver
      ⊞ Enum
      ⊞ HARDWARE
      ⊞ Network
      ⊞ Security
      ⊞ SOFTWARE
      ⊞ System
   ⊞ HKEY_USERS
   ⊞ HKEY_CURRENT_CONFIG
   ⊞ HKEY_DYN_DATA
```

Name	Data
(Default)	(valu

My Computer\HKEY_LOCAL_MACHINE\Config

The deeper you wander into the Registry, the more subkeys you'll find. You'll discover then that the space on the left side of the Registry Editor is no longer sufficient to display the entire Registry tree. But this is exactly why you can adjust the width of the panes. To do this, position the mouse over the vertical bar that divides the Registry Editor window. When the mouse pointer looks like a two-headed arrow, click the left mouse button and adjust the size of the panes to suit your taste. You can perform the same operation using a more complicated method by choosing **View, Split.**

And of course you can also adjust the size of the panes by using the **Ctrl+Left Arrow** or **Ctrl+Right Arrow** key combinations to make hidden keys visible.

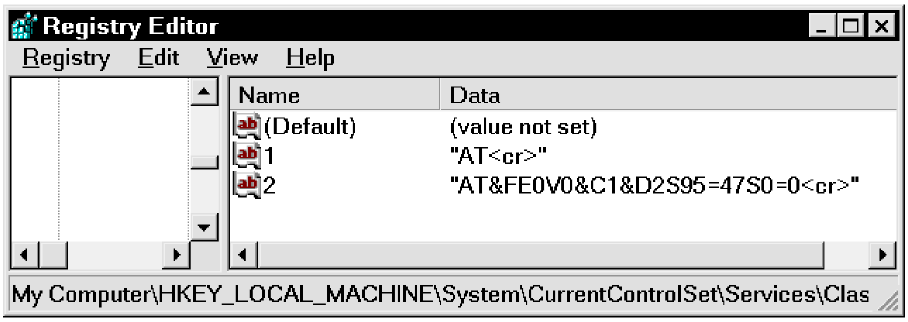
Creating Bookmarks

The complexity and hierarchy of the Registry requires the use of long key braches that one cannot possibly remember. Therefore I have a tip for you that will help you create a kind of bookmark: click the key you'd like to bookmark with the right mouse button and choose **New, String Value.**

As a name, use a keyword that you'll remember easily and that you'll be able to relate to the corresponding key. To avoid confusion with potential future keys or string values with the same name, place a star before and after the keyword. For example: if you need to change the initstring of your modem on a regular basis (the initstring is a string that contains all important modem settings), but you can't remember the "key-worm," then click the key

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Class\Modem\0001\Init

with the right mouse button, then choose **New, String Value** and give this "bookmark" the name ***modeminit***. Now you can use the Find function to jump to this spot anytime if you use ***modeminit*** as your search term.



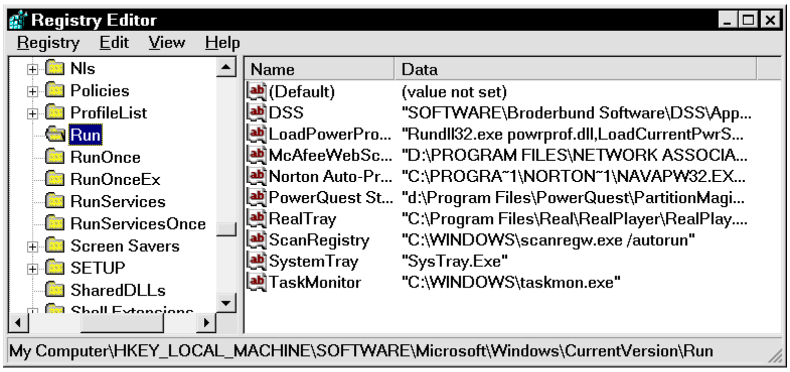
Name	Data
(Default)	(value not set)
1	"AT<cr>"
2	"AT&FE0V0&C1&D2S95=47S0=0<cr>"

My Computer\HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Clas

Illustration 14: With this "bookmark" and the Find function, you'll find your way back here

Making Notes

I'll admit it: the Registry is a veritable "Data Jungle," and you must sometimes search forever for an entry. In addition, given the sheer number of entries, you'll probably find that you have problems remembering which key or entry is responsible for what. So you'll find yourself writing many neat little notes about details of the Registry, which you'll never find on your messy desk (some of you will recognize your desks here.). But these little notes don't have to be; you can make notes to yourself directly in the Registry. To do this, right-click a key, choose **New, String Value**, then type in **Note:** as the name and your text as the value. This method has an advantage and a disadvantage: it's good that you can use your bookmarks as search terms, but it's too bad if all your notes disappear if you have to reinstall Windows. But then again, you probably wouldn't be able to find those little notes after four months either…



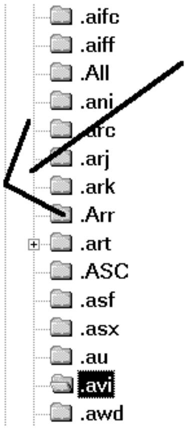
Name	Data
(Default)	(value not set)
DSS	"SOFTWARE\Broderbund Software\DSS\App...
LoadPowerPro...	"Rundll32.exe powrprof.dll,LoadCurrentPwrS...
McAfeeWebSc...	"D:\PROGRAM FILES\NETWORK ASSOCIA...
Norton Auto-Pr...	"C:\PROGRA~1\NORTON~1\NAVAPW32.EX...
PowerQuest St...	"d:\Program Files\PowerQuest\PartitionMagi...
RealTray	"C:\Program Files\Real\RealPlayer\RealPlay....
ScanRegistry	"C:\WINDOWS\scanregw.exe /autorun"
SystemTray	"SysTray.Exe"
TaskMonitor	"C:\WINDOWS\taskmon.exe"

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

Illustration 15: Recognizable notes in the "Key-Salad"

Shortcuts to Make Your Work Go Faster

Whoever has navigated the many branches of the Registry, perhaps in the course of an exhaustive search, has to scroll a pretty long way to return to the beginning. But this doesn't have to be: to jump immediately to the beginning, try the shortcut **Ctrl+Home**, or to jump to the last key, use **Ctrl+End**.



If you want to delete multiple entries at one time, you don't have to click each one and delete it separately; try holding down **Ctrl**, then mark the entries you'd like to delete one after the other. Then use **Delete** to delete them all at once. Or, if the entries you'd like to delete are all next to one another, you can hold down the **Shift** key, then click the first and last entry. The entries in between will be marked automatically.

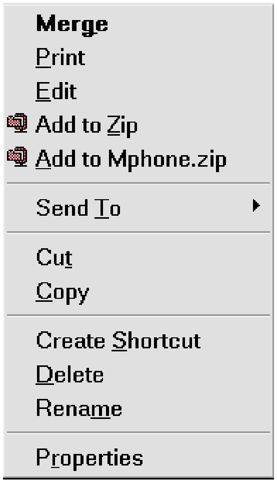
If you've just wandered into the depths of the Registry by accident and you'd like to restore some order to the keys you've opened, it's often the case that you have to scroll a long way upwards until you can see the key that you need to double-click in order to "fold it up." But you can spare yourself the scrolling up and searching for the key one level above; a double-click on the appropriate vertical hierarchy line will close the corresponding branch and help you gain a better overview of things.

The left arrow key is also useful; if you press it once, you'll jump to the next key upwards. If you press it again, the folder below will close itself up and give you an overview. Or, if you'd just like to jump to the key one level up without closing the subkey, use the **Backspace** or **Delete** key.

REG Files

If you want to export the Registry or some part of it, for example a single branch, you'll need to create a file with the extension REG. Such a file is a pure text file that you can load into an editor or Winword if you want to examine it in detail.

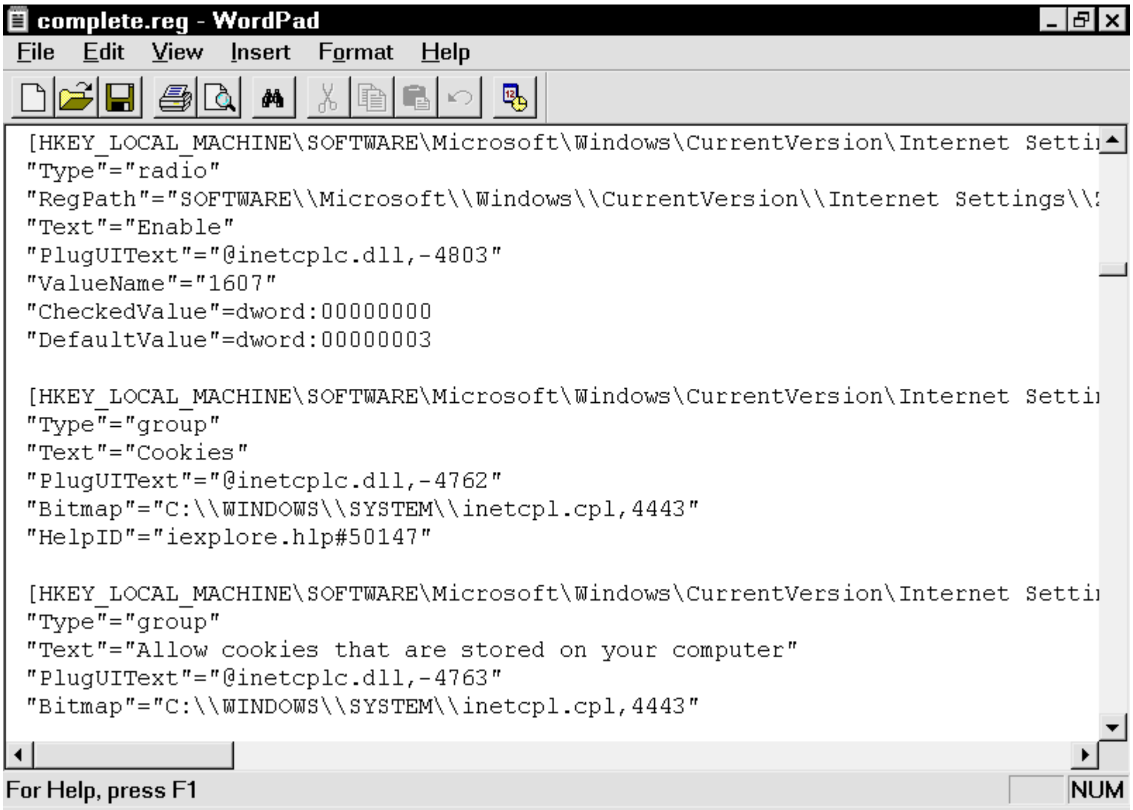
Tips for Working With REG Files



Don't ever click idly or "just for fun" on a random REG file. This could have very unpleasant consequences. Here's an example of what can happen: let's say you've been working with an Excel installation that you've taken the time to customize and configure according to your wishes. One day, without thinking, you double-click on the on the file EXCEL7.REG. The result of this action will be that all the default Excel settings will be written to the Registry and at least a portion of your personal settings and entries will be overwritten.

If you right-click a REG file in Explorer, you'll see an option on the context-menu called **Merge.** Don't click this one; doing so would have the same effect as double-clicking on the file, which would rewrite the settings in the file. If you want to examine the file in a text editor, choose **Edit**. This command is usually linked to Notepad. If the file is too large (if you've exported the whole Registry), then you'll need to use Wordpad instead.

The Structure of REG Files



```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Setti
"Type"="radio"
"RegPath"="SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Internet Settings\\
"Text"="Enable"
"PlugUIText"="@inetcplc.dll,-4803"
"ValueName"="1607"
"CheckedValue"=dword:00000000
"DefaultValue"=dword:00000003

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Setti
"Type"="group"
"Text"="Cookies"
"PlugUIText"="@inetcplc.dll,-4762"
"Bitmap"="C:\\WINDOWS\\SYSTEM\\inetcpl.cpl,4443"
"HelpID"="iexplore.hlp#50147"

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Setti
"Type"="group"
"Text"="Allow cookies that are stored on your computer"
"PlugUIText"="@inetcplc.dll,-4763"
"Bitmap"="C:\\WINDOWS\\SYSTEM\\inetcpl.cpl,4443"
```

Illustration 16: The Registry in text form as a REG file

REG files always follow the same format -- only in this way is it possible to guarantee that all entries in a REG file could be written smoothly to the Registry with a simple double-click. The first line of every REG file is the entry „REGEDIT4“. The key in question is displayed in pointy brackets. It doesn't matter whether or not the key already exists; if it doesn't, it will be created. If there's already a complete branch, then you'll jump directly to the entry in quotation marks and its value.

But quotation marks are not always used. The entry "Default", for example, because it's used so often, is represented by the at-symbol (@). As you'll note, there are no quotation marks in this entry. This is a clever idea since it reduces the number of characters and therefore also the required storage space quite drastically. With values there are also no quotation marks if the value in question is a hexadecimal or Dword value. The following examples are possible:

```
„Entry“=„Wert“
„Entry“=hex:00,00,00,01
„Entry“=dword:00030000
@=„Value“
```

Can I Write REG Files Myself?

If you're familiar with the principle of a REG file, you can easily write one yourself. Rather than explaining to your buddy in great detail what he needs to do to set a particular setting, you can write him such a file. You can send him the file on a diskette or via e-mail and without any further knowledge, the recipient can transfer the setting to the Registry via a double-click. Here's an example:

```
REGEDIT4
[HKEY_CURRENT_USER\Control Panel\desktop]
"CursorBlinkRate"="100"
```

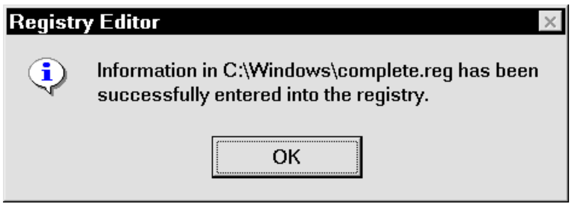
Type this file into Notepad and give it a good, mnemonic name, say CURSORBLINKRATE.REG. As soon as your buddy receives this file, he only has to double-click it. The setting that it changes relates to the blink rate of the cursor. The default value is "500." Smaller numbers make the cursor blink faster, larger ones make it blink slower. So you see: you don't necessarily need the Registry Editor to make changes to the Registry.

The ability to write registries in a text file has, despite the apparent extra work for which it makes, two definite advantages: first, this method is safer, since the entry is only made with a double-click on the complete REG file. In the Registry Editor you're working in real time; each action or change takes effect immediately, that is, it's saved. This explains why Regedit never asks you whether you want to save your changes when you're leaving the program.

The second advantage is for users who work on several computers or who are responsible for them. Instead of changing a setting on each computer on a network directly with its Registry Editor, you can write the file once and then transfer it to each computer in turn.

Shutting Off the Registry Confirmation

Whenever you double-click on a REG file, you'll see -- if everything is running smoothly and the file contains no errors -- a message informing you that the changes were successful. You close this message by clicking "OK."



Registry Editor

Information in C:\Windows\complete.reg has been successfully entered into the registry.

Illustration 17: In the long run, the messages can get on your nerves

In a single case this message may seem reassuring. But if you're writing the REG files of programs anew to the Registry after experiencing problems, you'll probably be annoyed that you have to click OK so often. A tiny intervention before you begin can shut off the Registry confirmation, thereby saving time and sparing your nerves. In the key

HKEY_CLASSES_ROOT\regfile\shell\open\command
you need only to replace „regedit.exe %1“ with the value „regedit.exe /s %1“.

Comparing Registries

The task of comparing two different Registry versions with one another seems quite daunting. You might have to take on this task if some keys are causing problems and you want to close in on the where the mistakes are by using a functional Registry. Or, if you'd like to see for yourself precisely what changes an application's installation program has made, you might want to compare a copy of the backed-up "before" version with the current registry after the installation is complete.

You can make this task easier with Winword, which has a **Compare Version** function that can show you any changes quickly and reliably. I'll explain how this works using the example of the registered username in Windows: click the key **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion** and choose **Registry, Export Registry File.** Save the branch under the name OLDNAME.REG. Now change the value of the entry "RegisteredOwner" by typing another word or name in there. Since this is just the name that you entered when you installed Windows, nothing serious can happen here. Now save (export) the altered key anew, this time under the name NEWNAME.REG.

Now call up Winword and choose **File, Open**. Make sure that under **Filetype** the entry **All Files** is selected; otherwise the files with the extension REG won't appear. Load the file OLDNAME.REG from whatever directory you've stored it in. Now choose **Tools, Track Changes, Compare Documents** (in Word 7/95 choose **Tools, Revisions, Compare Versions**). You'll have to enter the name NEWNAME.REG, then Windows will get right to work comparing the two files. This can take a while (if you're comparing long texts, for example a complete export of the Registry, it can take up to 10 minutes), but when it's done, Winword will show you quite precisely where changes have been made. The old entry is struck through with red, the new one is right next to it.

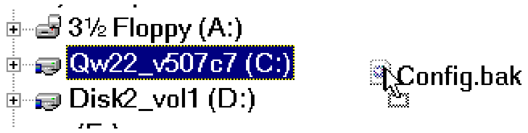
OLE, Drag & Drop, and CLSID

Please don't be afraid of the many, perhaps unfamiliar terms and abbreviations here. All three of these are closely related as far as the Registry is concerned. Let's begin with the simplest, the so-called Drag & Drop.

What in the World is Drag & Drop?

The term "Drag & Drop" refers to one of the features that makes working with graphical user interfaces like Windows so comfortable.

The simplest example of Drag & Drop occurs if you click on a program icon, move it, then let it fall in another place on the Desktop. Some users are so used to working this way that they don't even think about the procedures involved, which, for example, make possible the dragging of a file's icon onto a disk drive in Explorer.



It actually wasn't all that long ago that you had to type in cryptic command sequences in order to copy files or directories, move them, or delete them. Today, depending on your personal preference, you may delete a file by dragging it into the Recycling Bin. You can even drag a document over the Winword icon, whereupon Winword will start and load up the file in question.

So that the operating system is so comfortable to use and so that it can understand and interpret the user's actions correctly, it must administer a whole lot of information about data exchange, OLE, and the functions and characteristics of individual programs.

What in the World is OLE?

You've probably noticed by now -- I wanted to explain something about "Drag & Drop" and I went right on ahead to another new term, namely "OLE." "OLE" stands for "Object Linking and Embedding," that is, the insertion or linking of an object. The difference between inserting and linking can be explained using the following example:

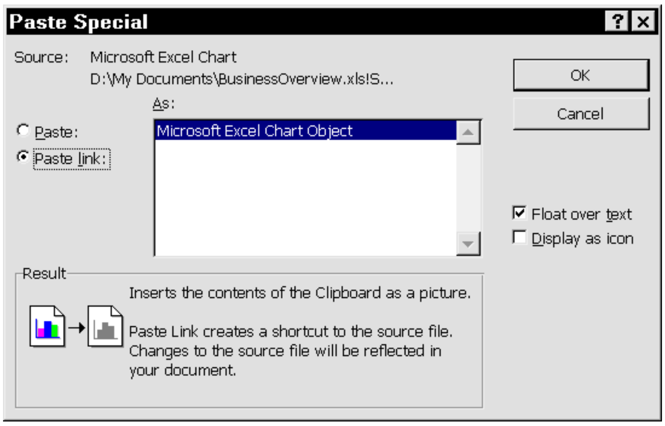
Pasting or Pasting a Link -- a Useful Difference

A) Mark a text in Windows' Notepad and use the key combination **Ctrl+C.** This copies the marked text to the clipboard. Now change to Winword and choose **Edit, Paste**.

B) From Excel, copy a diagram to the clipboard and then choose **Edit, Paste Special** in Winword. You'll see a dialog box that gives you a choice between **Paste** and **Paste link**. Change the setting to **Paste link** and click **OK**.

You've just created a link to the Excel diagram. If you change the underlying data in Excel, the diagram will be changed there. And, the next time you call up the Winword document, the diagram there will be updated too.

You must think of pasting a link as a constant mirroring, whereas pasting is more like taking a single snapshot. A pasted object will not be updated if the data in its source change.



Source: Microsoft Excel Chart
D:\My Documents\BusinessOverview.xls!S...

As:

○ Paste:
● Paste link:

Microsoft Excel Chart Object

☑ Float over text
□ Display as icon

Inserts the contents of the Clipboard as a picture.

Paste Link creates a shortcut to the source file. Changes to the source file will be reflected in your document.

What in the World is a CLSID?

Surely when you made your first lengthy excursion into the Registry you asked yourself with trepidation what those unbelievably long strings like **CLSID\{9E56BE60-C50F-11CF-9A2C-00A0C90A90CE}** could possibly mean.

Actually, this is easy to explain since we've already discussed the terms "OLE" and "Drag & Drop." Each OLE-enabled program, like Winword, and each OLE-enabled object (the aforementioned "Paint.Picture," would be an example) must be identified unambiguously. To do this, you could use the name of the program, but then you would run the risk of giving two programs or objects the same name, which would bring Windows to a screeching halt. In order that nothing like this happens, the practice is to use a long string that identifies each object or program precisely. CLSID stands for "ClassIdentifier."

Here's How to Make Searching for CLSIDs Easier

To find all entries for a program or an object, searching for the text name is not always sufficient. You should also try searching for the CLSID of the program/object. In order to do this, you must know the CLSID in question. But you'll almost always come across this information while you're searching on the name. For objects of the type "Paint.Picture," you'll find the CLSID under **HKEY_CLASSES_ROOT\Paint.Picture\CLSID**.

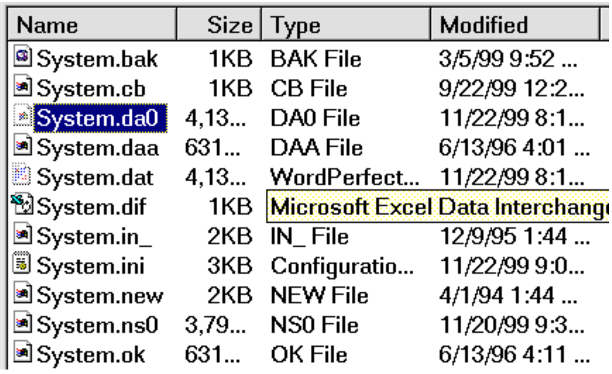
To search on a particular CLSID, you don't have to type the string into the **Find** dialog box manually. The danger of making a typo this way is much too great. To search on the CLSID of "Paint.Picture," double-click on the right-hand side on the entry "Default" of **HKEY_CLASSES_ROOT\Paint.Picture\CLSID**. Copy the marked value to the clipboard with the key combination **Ctrl+C** . Then choose **Edit, Find** and use **Ctrl+V** to insert the CLSID into the **Find** field. This copying and inserting will save you the pain of inserting the CLSID manually.

First Aid for the Registry

Securing the Registry

In contrast to the System Editor, the Registry Editor does not automatically save a backup copy when you make changes. But in the case of the Registry, "securing" things is not much trouble since the Registry databank consists of just two files, USER.DAT and SYSTEM.DAT. Under Windows 95, these files, together with the two backup copies USER.DA0 and SYSTEM.DA0, can be found in the Windows folder. Windows makes the backup copies automatically, so that if the files are damaged or there are other problems, the Registry databank can be restored without too much trouble. Windows 98 does things a slightly different way; when you boot your computer up, the program "Scanreg" creates a CAB archive in the folder \WINDOWS\SYSBCKUP, which secures USER.DAT, SYSTEM.DAT, WIN.INI and SYSTEM.INI in a special form.

Since, for security purposes, these files are endowed with the attributes "hidden," "system," and "write-protected," they are usually not shown by Explorer. In order to change this, you must choose **View, Options, Show All Files**. If you've installed Internet Explorer 4 or Windows 98, you'll find the setting under **View, Folder Options, View, Files and Folders, Hidden Files, Show All Files**.



Name	Size	Type	Modified
System.bak	1KB	BAK File	3/5/99 9:52 ...
System.cb	1KB	CB File	9/22/99 12:2...
System.da0	4,13...	DA0 File	11/22/99 8:1...
System.daa	631...	DAA File	6/13/96 4:01 ...
System.dat	4,13...	WordPerfect...	11/22/99 8:1...
System.dif	1KB	Microsoft Excel Data Interchang	
System.in_	2KB	IN_ File	12/9/95 1:44 ...
System.ini	3KB	Configuratio...	11/22/99 9:0...
System.new	2KB	NEW File	4/1/94 1:44 ...
System.ns0	3,79...	NS0 File	11/20/99 9:3...
System.ok	631...	OK File	6/13/96 4:11 ...

Illustration 18: The backup copy of SYSTEM.DAT under Windows

Copying User.dat and System.dat

Because USER.DAT and SYSTEM.DAT contain all the important settings in Windows, you should copy them to a diskette on a regular basis. Don't let the thought that Windows 95 makes its own backup copies with the extension DAO each time you boot up give you a false sense of security. For if an error occurs immediately after a second start-up, sometimes these files have already been overwritten. This doesn't happen under Windows 98; first of all, the CAB security archive is created only once a day, and secondly, it's numbered and only overwritten after 5 days.

So secure USER.DAT and SYSTEM.DAT by copying them to a diskette regularly; this way you'll ensure that the files are up-to-date should you ever need them. You'll probably need to compress SYSTEM.DAT using a compression program like Winzip since otherwise it won't fit onto a normal disk. You should also know that you should never use your backup copies of these files from Windows 95 with Windows 98 (and vice versa). The Registry in both versions of Windows is mostly identical, but the small differences could cause Windows to function improperly.

Here's the Quickest Way to Copy the Files to a Diskette

By default, Windows offers you an especially quick method to copy data to a diskette: right-click the file and choose the option **Send to, 3.5 Diskette (A)** from the context-sensitive menu.

Partial Export of the Registry

Before you change any entry in the Registry, it's always a good idea to back it up in its original form. This is exactly what the **Registry, Export Registry File** option in the Registry Editor is for.

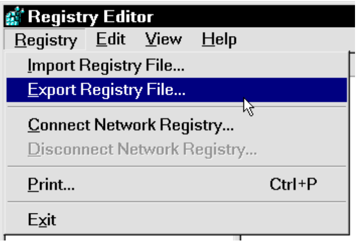
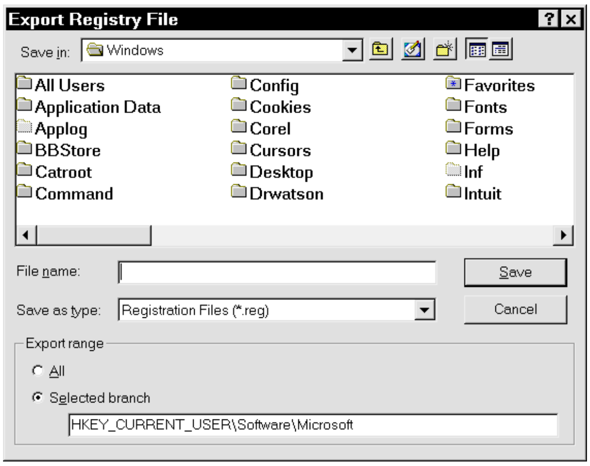


Illustration 19: Securi ng keys quickly

Before you undertake changes to the key **HKEY_CURRENT_USER\Software\Microsoft** or one of its subkeys, click **\Microsoft** and choose **Registry, Export Registry File**. In the dialog box that appears, make sure that the **Selected Branch** option is active and that the name of the selected key appears in the box. Give the exported file a good, mnemonic name. Now you can change the settings of the **\Microsoft** key without having to worry. If something goes wrong while you're working, you can just double-click on the exported REG file to rewrite the settings to the Registry. Alternatively, you can always use the **Import Registry File** command on the **Registry** menu.

The advantage of the exporting method is that instead of exporting the whole Registry to protect against changes, you can back up a single key.

In connection with backing up the Registry, I'd like to make you aware of the chapter "Tools for Working With the Registry." There you'll find an introduction to some programs that can ease or automate the regular backup of the Registry.



Save in: Windows

File name:

Save as type: Registration Files (*.reg)

Export range

○ All

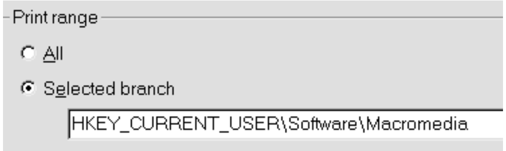
● Selected branch

HKEY_CURRENT_USER\Software\Microsoft

Illustration 20: "Selected Branch" tells you what exactly will be backed up

Printing Portions of the Registry Before Making Changes

Naturally before changing keys and values you don't need to painstakingly write them down on little pieces of paper; otherwise, why would we have printers? But I'd like to warn you not to print the whole Registry.



Print range

○ All

● Selected branch

HKEY_CURRENT_USER\Software\Macromedia

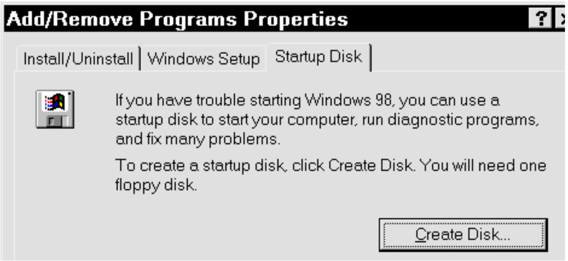
Now I'm next to certain that you'll be running at top speed to the next stationery store to buy a ream of paper. In order to put a particular portion of the Registry on paper before editing it, click the key you'll be editing and choose **Registry, Print**. The **Print Range** box shows you which portion of the Registry (plus all its subordinate branches) will be sent to the printer.

Tip: Whoever likes to work with Shortcuts (key combinations) can use **Ctrl+P** instead of **Registry, Print**.

Creating a Startup Disk

Surely you have a startup disk ready, or maybe not? If you haven't already prepared one or if your startup disk has become obsolete because you've added new hardware to your system, you should create a new one. To do this, choose **Start, Settings, Control Panel, Add/Remove Programs, Startup Disk**. With one click of **Create Disk**, your blank disk will be formatted and made into a bootable startup disk.

The purpose: backing up the files USER.DAT and SYSTEM.DAT is very important, but in case of true emergency, if you have no startup diskette, you may not be able to boot your computer up. For details of this situation, see the section entitled "Troubleshooting -- Nothing Works Anymore."



If you have trouble starting Windows 98, you can use a startup disk to start your computer, run diagnostic programs, and fix many problems.

To create a startup disk, click Create Disk. You will need one floppy disk.

But beware: on the default Windows 95 startup disk, there's no driver for a CD-ROM drive. This means that after booting your computer up, you won't have access to the CD-ROM drive, and thus to the Windows 95 CD. In this situation it's necessary to install this driver manually. In order to access your CD-ROM drive on the DOS level, you'll need two files.

The file A:\CONFIG.SYS contains the actual DOS drivers from the CD-ROM manufacturer or the default device drivers. These can look as follows:

DEVICEHIGH=A:\IDECDROM.SYS /D:MSCDOAK (this is all one line)

In this example, the file IDECDROM.SYS will be loaded from the diskette and the CD-ROM will be given the name "MSCDOAK."

The second necessary file that A:\AUTOEXEC.BAT must load is the file MSCDEX.EXE, which should also be on the diskette. In case this file is not yet on your diskette, you should copy it from the folder C:\WINDOWS\COMMAND\. The call to it will look as follows:

LH A:\MSCDEX.EXE /D:MSCDOAK /M:10 /E (this is all one line)

The name of the CD-ROM must be the same as the one defined in CONFIG.SYS; in this case, the name "MSCDOAK" would be given to the CD-ROM.

To review again: for maximal security, you should have in your possession a startup disk and secure copies of USER.DAT and SYSTEM.DAT. Exporting a particular key or even writing it down on a piece of paper before making changes gives you a quick "out" in case of problems.

Troubleshooting – Nothing Works Anymore

Hopefully your computer will never kick up its heels because of a destroyed or deleted Registry. But in case that should happen, if you've created a startup disk and you have backup copies of USER.DAT and SYSTEM.DAT, things aren't so bad after all.

Individual Errors after Startup

If you've changed a single entry, for example the one that controls the behavior of Winword, but now you're having problems, do the following: as I recommended in the section entitled "Securing the Registry," surely you've exported the relevant branch with its keys. To do this, you had to provide a filename. In order to restore the original form of this key with its strings and values, use Explorer to search for the corresponding file with the extension REG. If, when exporting, you gave the file the name "Backup of Winword Settings," you'll find the file BACKUP OF WINWORD SETTINGS.REG in the folder where you saved it (usually C:\WINDOWS). A double-click on this file will suffice, and all the settings from your original backup file will be written back to the Registry.

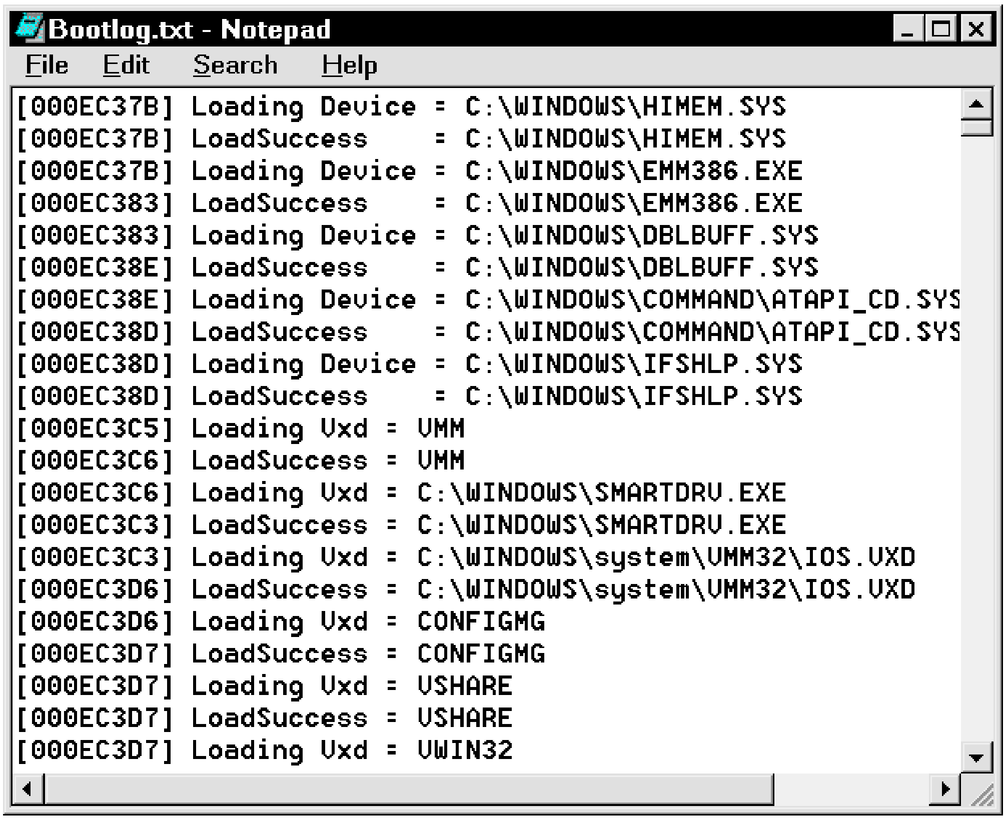
You'll encounter problems if you haven't backed up the key as I've advised you to do. Then you can try to reconfigure the program in question from scratch. If we're using Winword as an example, you'd probably choose **Tools, Options** for this task, since this is where you customize most of the settings. Perhaps this way you'll be able to correct the changes you made to the Registry. If this doesn't work, you may have to install the affected program all over again in order to make Windows write over the mistaken changes you made to Registry entries with correct settings.

Problems at Boot-Up

If Windows gives you one or more error messages on booting, you should examine the report on the booting process. To do this, press **F8** when you see the message "Windows 9x is starting up" in order to reach the boot menu. In Windows 98, you'll need to have set things up so that the boot menu comes up automatically (see p.
"Watch out for the Windows 98 Boot Menu"). Here you'll find the corresponding entry **Report**. After booting, you'll find a file called BOOTLOG.TXT in the main booting directory, usually C:\. This file contains information about the starting of each driver and its success or lack thereof. You should load this file into a text editor and search for the string "Failed." Each occurrence of "Failed" doesn't indicate an actual error. For example, Windows adds the line

```
[00113FFB] Loading Vxd = ebios
[00113FFB] LoadFailed = ebios
```

if it doesn't find an expanded BIOS. But maybe you can find the culpret if there's a missing, perhaps mistakenly deleted file indicated here.



```
[000EC37B] Loading Device = C:\WINDOWS\HIMEM.SYS
[000EC37B] LoadSuccess    = C:\WINDOWS\HIMEM.SYS
[000EC37B] Loading Device = C:\WINDOWS\EMM386.EXE
[000EC383] LoadSuccess    = C:\WINDOWS\EMM386.EXE
[000EC383] Loading Device = C:\WINDOWS\DBLBUFF.SYS
[000EC38E] LoadSuccess    = C:\WINDOWS\DBLBUFF.SYS
[000EC38E] Loading Device = C:\WINDOWS\COMMAND\ATAPI_CD.SYS
[000EC38D] LoadSuccess    = C:\WINDOWS\COMMAND\ATAPI_CD.SYS
[000EC38D] Loading Device = C:\WINDOWS\IFSHLP.SYS
[000EC38D] LoadSuccess    = C:\WINDOWS\IFSHLP.SYS
[000EC3C5] Loading Vxd = VMM
[000EC3C6] LoadSuccess = VMM
[000EC3C6] Loading Vxd = C:\WINDOWS\SMARTDRV.EXE
[000EC3C3] LoadSuccess = C:\WINDOWS\SMARTDRV.EXE
[000EC3C3] Loading Vxd = C:\WINDOWS\system\VMM32\IOS.VXD
[000EC3D6] LoadSuccess = C:\WINDOWS\system\VMM32\IOS.VXD
[000EC3D6] Loading Vxd = CONFIGMG
[000EC3D7] LoadSuccess = CONFIGMG
[000EC3D7] Loading Vxd = VSHARE
[000EC3D7] LoadSuccess = VSHARE
[000EC3D7] Loading Vxd = VWIN32
```

If you see an error message from AUTOEXEC.BAT or CONFIG.SYS that goes by so fast that you can't read it, just press the **Pause** key the next time you boot up. Once you've discovered where the offending line is, boot up using your (current) startup disk. Load the file that contains this line in a text editor and place the word **rem** and an empty space in front of the line in question to deactivate it.

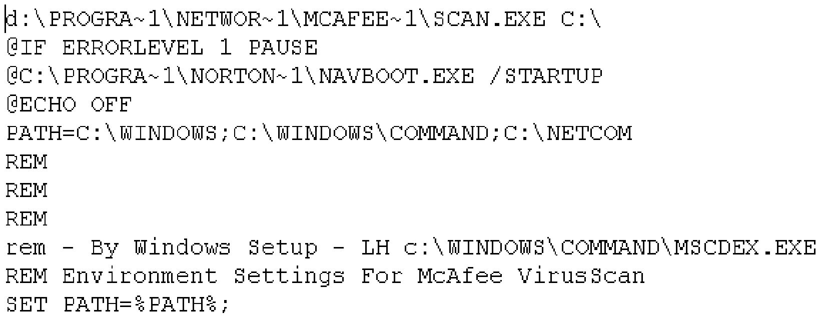
If it appears that the error lies in a faulty Registry entry, boot up with the startup disk and then copy your backup copies of USER.DAT and SYSTEM.DAT into the folder C:\WINDOWS to write over the old files.

If you're unlucky enough not to have a startup disk on hand, you can try pressing the **F8** key when you see the message "Windows 95 is starting up" in order to call up the boot menu. Under Windows 98, use the **Ctrl** key instead. Try to boot using the "XXX," "XXX" or "XXX" options, then try to restore the Registry using your backup copy or the Windows backup files USER.DA0 and SYSTEM.DA0 and make any necessary changes to AUTOEXEC.BAT or CONFIG.SYS.

Error Messages after Uninstalling a Program

You may experience an often harmless but still aggravating problem if a program hasn't been properly uninstalled. This can happen if the uninstallation program doesn't work right and, for example, has deleted the driver files, but hasn't removed the calls to these files from the Registry, that is, from SYSTEM.INI & Co. Or perhaps you've simply deleted the directory for an application and there are still calls to files in this directory hanging around. In both cases, at bootup, as in the case of calls in the system files and the Registry, Windows will try to load particular files that no longer exist.

As a consequence, in most cases you'll see an error message, which you must confirm by clicking **Enter** or **OK** before the booting process can continue. If the missing file causes no problems other than the error message, then you'll only need to remove the call to it. To do this, you'll need to open up the Registry Editor, where you'll use **Edit, Find** to search for the offending filename in the Registry. You should back up the relevant portions of the Registry using **Registry, Export Registry File** and then delete the offending string or key wherever it occurs. If you don't find the filename in the Registry, load up the System editor SYSEDIT.EXE using the **Run** option on the Start menu. Then you'll need to search for the filename in SYSTEM.INI, WIN.INI, AUTOEXEC.BAT and CONFIG.SYS. In the last two files, if you find what you're looking for, deactivate the relevant line by adding the word **rem** and an empty space.



```
d:\PROGRA~1\NETWOR~1\MCAFEE~1\SCAN.EXE C:\
@IF ERRORLEVEL 1 PAUSE
@C:\PROGRA~1\NORTON~1\NAVBOOT.EXE /STARTUP
@ECHO OFF
PATH=C:\WINDOWS;C:\WINDOWS\COMMAND;C:\NETCOM
REM
REM
REM
rem - By Windows Setup - LH c:\WINDOWS\COMMAND\MSCDEX.EXE
REM Environment Settings For McAfee VirusScan
SET PATH=%PATH%;
```

In both INI files, you'll need to add a semicolon (;) in front of the offending line. This is how to deactivate the call to the file. If you do this, the error message should go away.

If the missing file named in an error message creates problems even after you've started Windows up, you must, for better or worse, restore the file to your system. First I'd look in the Recycling Bin; if you deleted the file reasonably recently, it's probably still in there and you'll be able to return it to the "land of the living" by right-clicking on it and choosing **Restore**. Another possibility would be to do a complete reinstallation of the program the file belongs to into the same folder, which would replace all missing files. If you previously uninstalled the program incorrectly by simply deleting its folder, after a reinstallation you can uninstall the whole application cleanly by choosing **Start, Settings, Control Panel, Add/Remove Programs**.

System Tuning with the Registry

Making Windows Faster and Cleaning Up

Creating a Delay in the Start Menu

By default, the entries on the Start menu unfold after a delay. Windows users are divided when it comes to their opinion of this feature: some feel it is a pure waste of time, while beginners find it advantageous that they can use the mouse to choose an entry without having any subgroups unfold immediately. Whichever side you find yourself on, you can make the delay longer or shorter. To do this, choose the Registry path
HKEY_CURRENT_USER\ControlPanel\desktop

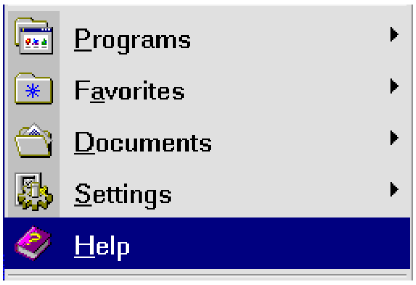
There, in the right-hand window, you'll find the string "MenuShowDelay". If not, right-click the **\desktop** key, choose **New, String Value** and enter the name shown above. A double-click on the (new) entry allows you to type in any number you want as an expression of the delay in milliseconds. If you don't want any delay at all, set this value to 0. For a longer delay, set this value to 500 or even higher.

Start Menu – Slimmed Down

By default, Windows' Start menu contains different entries like **Find** and **Run**. You can't delete these entries, for they don't show up in Explorer under \WINDOWS\STARTMENU. But there's almost nothing that you can't solve with the Registry. Try calling up the key

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\

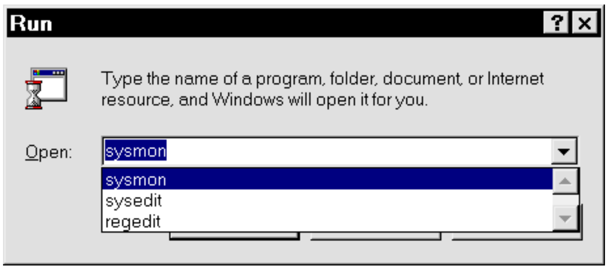
Create a binary value **NoRun**, which, after a double-click you'll assign the value **01** in order to throw the entry **Run** off the Start menu. Similarly, you can delete the entries **Control Panel** and **Printers** from **Start, Settings** if you create a binary value **NoSetFolders** with a value of **01**.



Deleting the Run List

In an effort to make Windows more user-friendly, its creators have become accustomed to storing and reporting on all actions and entries. To this practice belongs, for example, the automatic storage of the last entries in the **Run** dialog box, so that you can choose a value you've already typed in once from a comfortable list box.

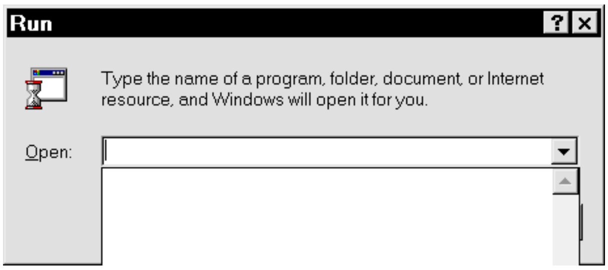
If you'd like to delete this list in order to hide your actions from curious onworkers, you need to look at the Registry key **HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU**
Here you'll need to delete the entries "a", "b", "c", "d", etc. The information will be removed the next time you boot up.



Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: sysmon

sysmon
sysedit
regedit



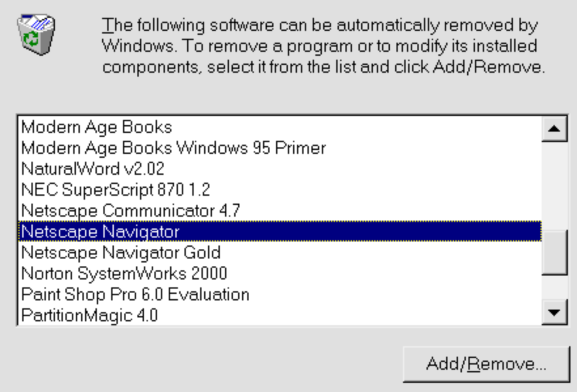
Run
Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.
Open:

Illustration 21: ... now the list is empty

Cleaning Up Your Uninstall List

For every program that you install using Setup, Windows notes the settings for the uninstallation of the program. This is so that little files like old DLL files in the WINDOWS\SYSTEM directory will be removed too. Otherwise, these files would remain behind if you just deleted the program folder. So much for theory -- in practice, the uninstall routines do not always function properly, so sometimes the names of programs remain behind in the uninstall list even after they've been uninstalled. You'll find the uninstall option if you choose **Start, Settings, Control Panel, Add/Remove Programs**. You can modify this list using the Registry Editor. Delete the subkeys of **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\\,** that refer to programs that are no longer present. If you do this, these programs will no longer appear on the uninstall list.

Optical Tuning

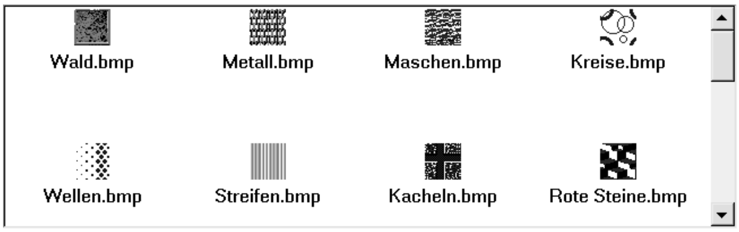


The following software can be automatically removed by Windows. To remove a program or to modify its installed components, select it from the list and click Add/Remove.

- Modern Age Books
- Modern Age Books Windows 95 Primer
- NaturalWord v2.02
- NEC SuperScript 870 1.2
- Netscape Communicator 4.7
- Netscape Navigator
- Netscape Navigator Gold
- Norton SystemWorks 2000
- Paint Shop Pro 6.0 Evaluation
- PartitionMagic 4.0

Displaying Bitmaps as Icons

If in the Explorer or on the Desktop you prefer to represent files with icons, by default you'll see graphics files with the extension BMP displayed with the icon of the application that's designated as the one for this graphic type, for example, with the icon for Paint. With a small intervention in the Registry, however, you can make it so that instead of the Paint icon, you'll see a tiny version of the complete picture as the file icon, as a kind of preview. Change to **HKEY_CLASSES_ROOT\Paint.Picture\DefaultIcon** and, after a double-click, change the value of the string "Default" to %1. This value is a placeholder that causes the file icon to display the contents of its respective bitmap file. In Explorer, you can use the **View** menu to change the representation and thus the size of the icon.



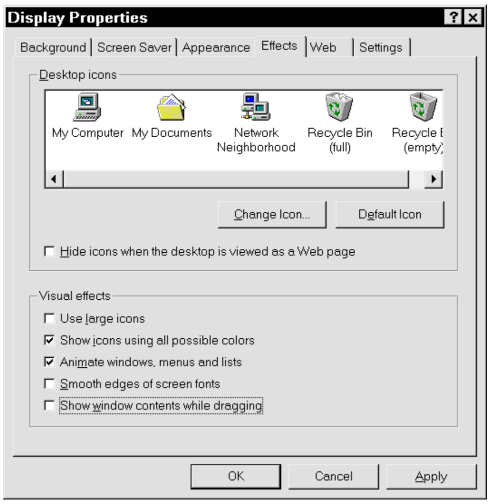
Placing Your Wallpaper Precisely

When you're setting your wallpaper, Windows isn't exactly flexible, even if you're using the Active Desktop in Windows 98 or Internet Explorer 4, which slows down the system quite a bit. There are only two options, **Centered** or **Tiled**, for laying out the whole Desktop. With the help of the Registry Editor, you can, despite the setting **Centered**, place a picture wherever you want it on your screen. Let's take the example of someone who wants to place a firm logo in the lower right-hand corner, which, given the normal settings, isn't possible. Change to the Registry Editor and search for the Registry path **HKEY_CURRENT_USER\ControlPanel\Desktop**. Right-click the **\Desktop** key and choose **New, String**. Create the strings **WallpaperoriginX** and **WallpaperoriginY**. Give these strings the exact pixel values where you'd like the picture to begin. With a screen resolution of 640x480, you might enter the values 550 and 380, depending on the size of the picture. Reboot your computer and check the position of your wallpaper.

By the way, if the active background color doesn't match your background, you can change it by right-clicking the Desktop and choosing **Properties, Appearance.**

Icons in 16 Bit Color

One feature that many users got with the Plus! package for Windows 95 is the ability to display icons in 16-bit color. You can actually activate this feature if you go into the Registry, right-click the key **HKEY_CURRENT_USER\ControlPanel\desktop\WindowMetrics**, and choose **New, String**. Give this string the name **Shell Icon BPP**. Double-click the new entry and enter the value 16, which will set the color to 16-bit per pixel (BPP). In order to enjoy your new and improved icons, your graphics card must be capable of "High Color" and your graphics card driver must be set to 16 or a higher number of colors. Under Windows 98, this is even easier: right-click the Desktop and choose **Properties, Effects, Show icons using all possible colors.**



□ Hide icons when the desktop is viewed as a Web page

□ Use large icons
☑ Show icons using all possible colors
☑ Animate windows, menus and lists
□ Smooth edges of screen fonts
□ Show window contents while dragging

Turning Off „Click Here to Start“

Using the options **Auto hide** and **Always on top**, you can configure the taskbar with the Start button so that it becomes visible only when you move the mouse pointer to the lower edge of the screen. If you're using this setting, then you'll always see a message in the taskbar that says "Click here to Start" with an arrow pointing to the Start button. If you'd like, you can turn this message off. To do this, start the Registry Editor and search for the key **HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies.** Click the subkey **\Explorer** and choose **New, Binary Value**. Here, you'll want to enter a new value, which you should call NoStartBanner. Double-click the new entry and give it the value 01 00 00 00.

Animation When You’re Minimizing and Maximizing

Whenever you minimize or maximize a program window using the appropriate button in the upper right corner of the screen, Windows opens or closes the window using animation. As nice as this feature can be, if you’re using a slower computer, it can slow system performance unnecessarily. To shut off the animation, open the Registry Editor and look at the key **HKEY_CURRENT_USER\ControlPanel\Desktop\WindowMetrics.** Right-click on **\WindowMetrics** and choose **New, String Value**. Give the new string the name **MinAnimate** and give it the value **0**. Now the animation when you’re minimizing or maximizing program windows is shut off; if you’d like to turn it on again, just change the value of **MinAnimate** to **1**.

Icon Resolution and Number of Colors Next to the Clock

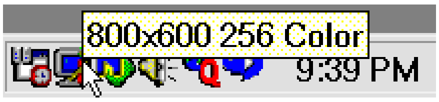
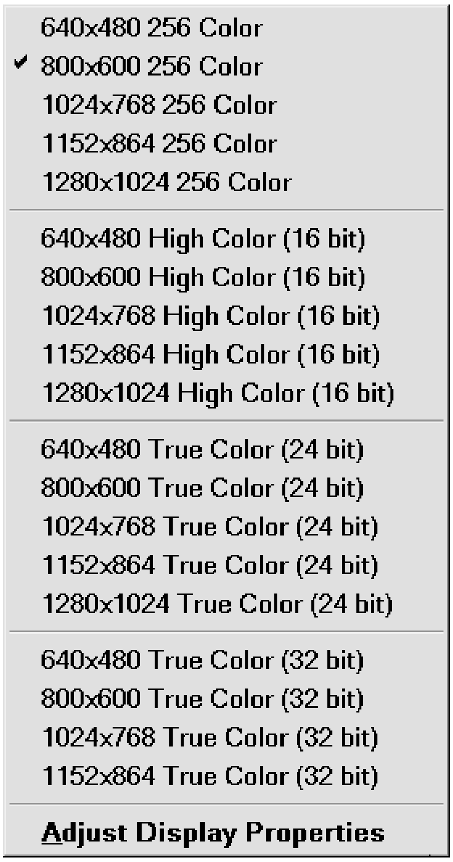


Illustration 22: This icon is the result of our efforts

Using the Registry, you can accomplish yet another very useful thing: with a single string you can install an icon in the System Tray, right next to the clock, that will display all the possible combinations of resolution and colors. This way, you can switch from a resolution of 640x480 with Truecolor (24 bit color depth) to 800x600 in High Color (16 bit) in the blink of an eye. Here’s how to install the icon: right-click the key **HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run** and choose **New, String Value**. Add an entry called **Taskbar Display Controls**. Double-click the new entry and give it the following value: **RunDLL deskcp16.dll,QUICKRES_RUNDLLENTRY**. After you’ve rebooted, you should see the new icon next to the clock. Unfortunately, for reasons that are unclear, this does not work on every system.


640x480 256 Color
✓ 800x600 256 Color
1024x768 256 Color
1152x864 256 Color
1280x1024 256 Color
640x480 High Color (16 bit)
800x600 High Color (16 bit)
1024x768 High Color (16 bit)
1152x864 High Color (16 bit)
1280x1024 High Color (16 bit)
640x480 True Color (24 bit)
800x600 True Color (24 bit)
1024x768 True Color (24 bit)
1152x864 True Color (24 bit)
1280x1024 True Color (24 bit)
640x480 True Color (32 bit)
800x600 True Color (32 bit)
1024x768 True Color (32 bit)
1152x864 True Color (32 bit)
Adjust Display Properties

Customizing Windows with the Registry

Changing User Names

When you’re installing Windows, you’re asked to provide a name and a firm name. Windows stores these names, like nearly all system settings, in the Registry, specifically in the key **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ Windows\CurrentVersion** under „RegisteredOwner“ and „RegisteredOrganization“. If you double-click on these strings, you can change the names if you made a mistake when typing them in or if you’ve changed your name because you got married or whatever.

Customizing the Setup Path

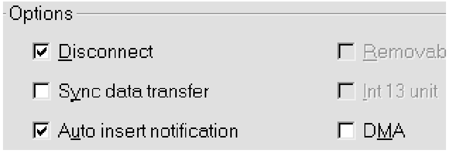
If, for the sake of speedy access, you’ve copied the Windows installation files onto your hard disk, you’ll certainly want to let Windows know that you’ve done this. Otherwise, Windows will always ask for the location of the Windows CD as soon as you use Setup to install or uninstall components. You’ll find this setting under **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Setup**

On the right-hand window you’ll find the string “SourcePath,” which you can edit after double-clicking it. Change the path of your CD-ROM drive by specifying the new position of the installation data on the hard disk. This trick is also useful if the drive letter of your CD-ROM drive has changed because you’ve installed additional hard disks in your computer.

If you’ve installed a new hard disk and thus changed the drive letter of your CD-ROM drive, the installation path for the Plus! package in the Registry won’t be correct anymore. Use the key **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Plus!\Setup** to correct the path by changing the value of “SourcePath.” Here you can provide a path to a directory on the hard disk if that’s where you’ve stored the installation data.

Only Suppress the Autostart of Audio-CDs

Windows recognizes when you’ve put in an audio CD and it starts playing the disk automatically. It does the same thing with data CDs if they have an Autorun file in their root directory. If you’d like to circumvent the Autorun, you can hold down the **Shift** key when you’re putting the CD in the drive. In the long run, however, this is inconvenient, so you might consider using **Control Panel, System, Device Manager, CD ROM, Your CD ROM, Properties** to turn this function off completely. On the **Settings** tab, you’ll find the option **Auto Insert Notification**, which you can turn off by unchecking the box with a click of the mouse.



Options

☑ Disconnect □ Removab

□ Sync data transfer □ Int 13 unit

☑ Auto insert notification □ DMA

But if you’d like to shut off only the Autorun of music CDs, this option is no good for you since it turns off the Autorun feature on data disks too. So change to the Registry editor, where you can specify settings for each type of CD separately. Under the key **HKEY_CLASSES_ROOT\AudioCD\shell\play\command** double-click „Default“ and delete the command C:\WINDOWS\CDPLAYER.EXE /PLAY %1. This way, music CDs won’t Autorun, but the behavior of data CDs won’t be affected. If, later on, you’d like to take back this action, you only need to reinsert the command line you just took out.

You can also change this setting using Explorer. Choose **View, Folder Options, File Types** and double-click the entry **Audio CD**. If you **Remove** the **Play** function, you'll delete the automatic Autorun; later on you can re-insert this function by choosing **New**. After you've reinserted it, click **Set Default** and everything will function as it did from the start.

Preventing the Saving of Settings

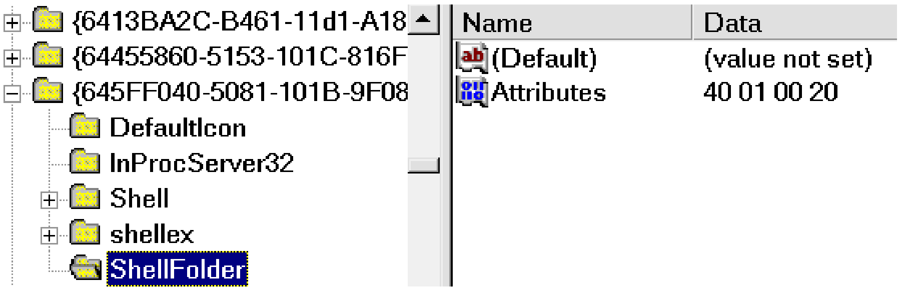
At the end of every session, Windows automatically stores the current settings for the Desktop. These include the arrangement of icons and the size and position of the taskbar. So that a once-neat desk won't look chaotic after accidental or mischevious changes, you can prevent Windows from automatically saving settings. To do this, open the Registry Editor and have a look at the key **KEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\**. Create a binary value "NoSaveSettings," double-click it, and give it the value "01". By doing this, you'll also break Windows of its habit of calling up any programs that you left open at the end of the last session.

Hiding Programs that Start when Booting

If you'd like to call up a particular program automatically each time you start Windows, you can create a shortcut for it in the Autostart folder. If you've created such an Autostart shortcut, you can circumvent it with a simple press of the **Shift** key while the Windows logo is on the screen and then remove it from the Autostart folder. However, the **Run** function is much more reliable, since it can call a given program up after booting, but it's buried deeper. Open the Registry Editor and search for the key **KEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run**. On the right-hand side of the Editor window you'll see the entry "Default" and perhaps there will be other program commands there as well. For example, if you'd like Winword to start up automatically, right-click the **\Run** key and choose **New, String Value**. Name the new string **Winword**, double-click it, and assign it the command line **C:\Programs\Office\Winword\Winword.exe**. Note that you may have to customize the path to suit conditions on your system. A program that's called up by **Run** will be visible in the Autostart folder, and it can be circumvented by pressing the **Shift** key during boot-up. Users in the know probably know the trick, however, which is why you shouldn't rely 100% on the security of this function.

Shutting off the Protection Against Renaming the Recycle Bin

By default, the Recycle Bin is protected against renaming. If you double-click its icon or select it and then press **F2**, you'll hear a voice from the loudspeaker telling you that you can't rename this object. But if you're clever, you can shut off this protection in the Registry in order to rename the Recycle Bin according to your mood. Just consider some of the possibilities: "Data Dumpster," "Digital Recycling," or "File Graveyard." To do this, change to the key **HKEY_CLASSES_ROOT\CLSID\{645FF040-5081-101B-9F08-00AA002F954E}\ShellFolder**. If you click the long string, you'll see the plain text name "Recycling Bin" appear on the right-hand side of Explorer. Now click **\ShellFolder**, double-click "Attributes" and change the first pair of numbers from "40" (the default) to **10.**



```
{6413BA2C-B461-11d1-A18      Name          Data
{64455860-5153-101C-816F     (Default)     (value not set)
{645FF040-5081-101B-9F08     Attributes    40 01 00 20
    DefaultIcon
    InProcServer32
    Shell
    shellex
    ShellFolder
```

Illustration 23: The attributes of the Recycling Bin

Now you can rename the Recycling Bin anytime you want to. The entry "Attributes" is a so-called EditFlag, which determines particular restrictions on objects.

Removing the „Log Off" and „Favorites" Entries from the Start Menu

Windows 98 and Windows 95 plus Internet Explorer ask you each time they start up after installation to provide a username and an optional password. If you're the only person using your PC and you'd rather not have a user profile, you should just enter a name here and forego entering a password. This way, Windows won't bother you with a login procedure; otherwise you must provide a name and password each time you start up. In any case, the next-to-last entry of the Start menu is **Log Off Username**, which you can turn off if you choose to do so.

To do this, right-click

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer

and choose **New, Binary Value**. Name the new entry **NoLogOff** and assign it the value **01**.

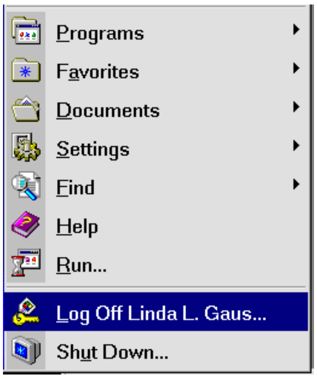


Illustration 24: Here the entries are still there …

In the same key you can also create an entry named **NoFavoritesMenu** after choosing **New, Binary Value**. If you assign this entry the value **01**, the Favorites entry in the Start menu will also disappear.

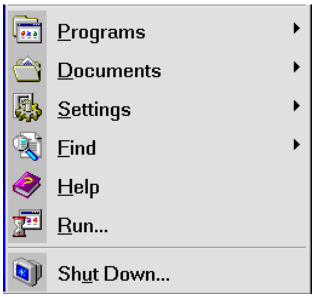


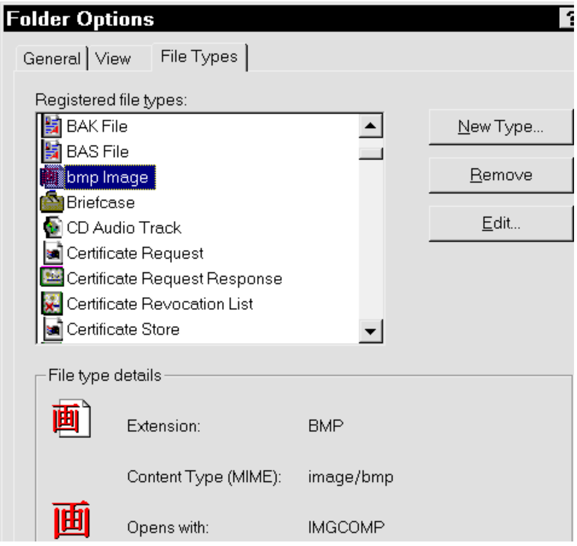
Illustration 25: ... but here they're gone

Improving System Performance with the Registry

Customizing Context-Sensitive Menus

Order of Menu Entries

The topmost entry in a context-sensitive menu is always the command that can be carried out with a double-click. You'll recognize this because this command appears in **boldface**. Thereafter follow alternative commands such as **Print** or **Edit**. The order of items in context-sensitive menus is easy to change using the Registry. In order to change the context-sensitive menu for batch files, which by default starts with **Open**, you need to go to **HKEY_CLASSES_ROOT\batfile\shell**.



Registered file types:

- BAK File
- BAS File
- bmp Image
- Briefcase
- CD Audio Track
- Certificate Request
- Certificate Request Response
- Certificate Revocation List
- Certificate Store

File type details

Extension: BMP

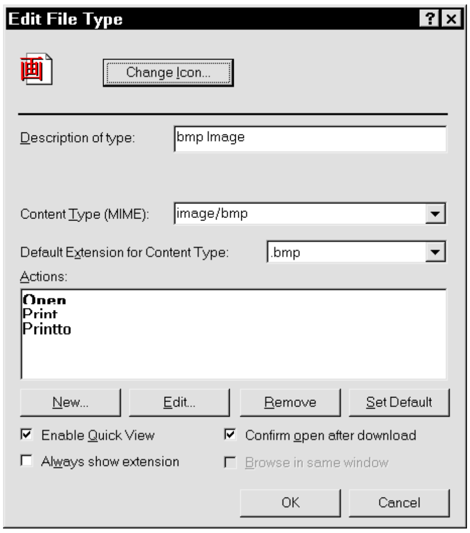
Content Type (MIME): image/bmp

Opens with: IMGCOMP

Illustration 26: Choose the file type, then click "Edit"…

There you'll find the subkeys **\edit**, **\open**, and **\print** for the commands **Edit**, **Open**, and **Print**. If you'd like to make it so that BAT files won't be opened with a double-click, but rather loaded into an editor for processing, you'll need to double-click on the "Default" entry under **\shell** and type in the order in which you'd like these commands to appear: **edit, open, print**. Now, if you look at a context-sensitive menu, you'll see **Edit** in the first place in boldface, then **Open** and **Print**.

The default order of context-sensitive menus can be changed not only in the Registry, but also on a dialog box. In Windows 98, open Explorer and choose **View, Folder Options, File Types**; in Windows 95, you'll find the dialog box under **View, Options, File Types**. If, for example, you'd like to print BMP files when you double-click them instead of opening them (which makes sense if you have a lot of these files to print), click the type "Bitmap" and choose **Edit, Print, Set Default**.



Description of type: bmp Image

Content Type (MIME): image/bmp

Default Extension for Content Type: .bmp

Actions:

Open
Print
Printto

☑ Enable Quick View ☑ Confirm open after download

□ Always show extension □ Browse in same window

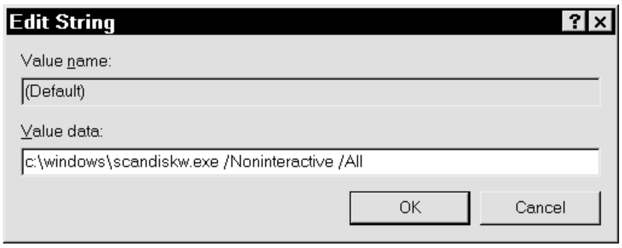
Illustration 27: ... and choose the command as the default

Adding Scandisk to the Context-Sensitive Menus of Drives

If you right-click a drive icon on the Desktop or in Explorer, you'll see a context-sensitive menu that by default contains many entries. If you wish, you can add even more entries to this menu, for example, the command **Scandisk**. Do as follows: choose **Start, Run, Regedit** and then change to the Registry path **HKEY_CLASSES_ROOT\Drive\Shell**. Open the context-sensitive menu by right-clicking on **\Shell**, then choose **New, Key** and create the key **Scandisk**. On the right-hand window you'll see the string "Default," which, if you double-click it, you can change into the context-sensitive entry "Scan&disk".

The businesslike and (&) specifies the letter that you'll use to call up the command from the context-sensitive menu if you're using the keyboard. You can also open the context-sensitive menu with the right mouse button (or with **Shift+F10**), and then close it by pressing **D** on the keyboard. It's also possible to define another letter as the command activator; all you have to do is put another letter in front of the symbol.

Right-click the key **\Scandisk** and create a subkey **Command** by choosing **New, Key**. The subkey **\Command** also contains the string "Default," which you can double-click and assign the command **c:\windows\scandskw.exe**.



Value name:
(Default)

Value data:
c:\windows\scandiskw.exe /Noninteractive /All

Illustration 28: Automate by using Parameters

You can also add parameters to the command string, for example, the parameter **/All**, which automatically selects all drives for inspection. Or you can automate the procedure with the parameter **/Noninteractive**, which causes Windows to start the drive analysis without further intervention on the part of the user.

Collapsing or Expanding the New Menu

You'll find the command **New** in many places, for example on the context-sensitive menu on the Desktop after a right-click or on Explorer's **File** menu. The more applications you install, the longer and more cumbersome the list of file types that you can create anew becomes. To collapse this menu – what else – you must use the Registry Editor. Search for the relevant file type, for example WAV, under the key **HKEY_CLASSES_ROOT**. If the file type supports the **New** function, it will be present in the subkey **\ShellNew** – in our example above, the complete Registry path would be **HKEY_CLASSES_ROOT\.wav\ShellNew**.

Delete the subkey **\ShellNew** and the file type will no longer appear on the **New** menu.

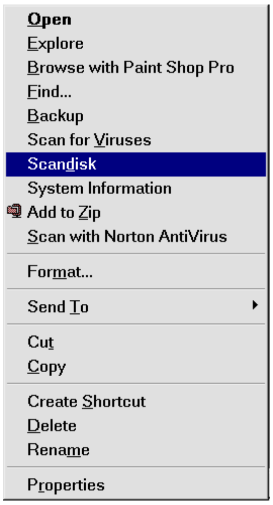


Illustration 29: Scandisk on the context-sensitive menu for all drives

Just as you can collapse the commands on the **New** menu according to the foregoing tip, you can also add new entries to it. Under **HKEY_CLASSES_ROOT**, and its corresponding file type, create a subkey with the name “ShellNew”. Then add a string “NullFile”. If, for example, you’d like to use the **New** menu to create Internet sites easily in HTML, you’d need to use the Registry Editor to create a subkey called “ShellNew” with the string “NullFile” under the key **HKEY_CLASSES_ROOT\.htm**.

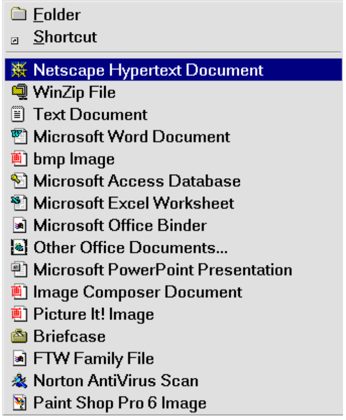


Illustration 31: ... now it’s gone

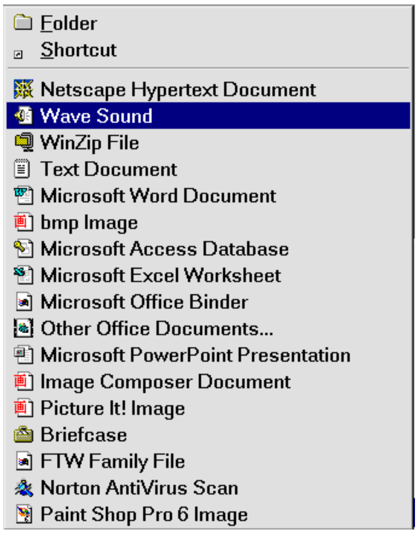


Illustration 30: Here, the WAVE-Audio is still there...

Adding an Antivirus Scan to the Context-Sensitive Menu of a Disk Drive

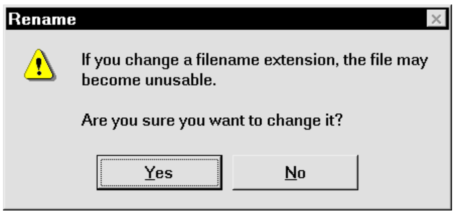
DOS programs are, with very few exceptions, no longer current. One of these exceptions is virus scanners, which, as DOS programs, can easily be started from a startup disk in case your system is ever threatened by an invasion of digital pests. The biggest disadvantage of these DOS helpers as compared to their Windows equivalents is that the DOS virus scanners must be called up with parameters instead of by clicking quickly on an icon. So we'd like to expand the context-sensitive menu of disk drives by one command, which will perform an immediate scan of the affected drive. As an example, we'll use the McAfee virus scanner. Call up the key **HKEY_CLASSES_ROOT\Drive\shell** and create a subkey named "Virusscan" by right-clicking on **\shell** and choosing **New, Key**. In exactly the same way, you should create a subkey of the new key **\Virusscan**, which should be named **Command**. Click **\Command** and assign its "Default" entry the value **e:\antivir\mcafee\scan.exe /all /sub %1**. Naturally the precise form of this command will depend on whether you're really using the MacAfee virus scanner and where you've saved the file. Customize this value to suit your particular circumstances.

Printing the Contents of a Folder or Saving its Contents to a File

To gain an overview of all the files in a folder, you can use Explorer or the command **Dir**. Often you'll want to print the list of files or save this list to a file in order to archive it. Doing this has one major advantage: if, during installation, a program has copied files, for example into the WINDOWS directory, and an uninstall program hasn't removed them, if you have a printed list of files, you can figure out which files you no longer need (if you've "printed" the list to a file, consider using the **Version Compare** function in Winword for doing this) and delete them manually. In order to print such a file list, you need to change to the directory in question and use the command **Dir > LPT1**.

Unfortunately Explorer doesn't offer you the opportunity to print the contents of a directory, but you can use a batch file and an entry in the Registry to change this. Create a new text file called PRINTDIR.BAT in the C:\WINDOWS\COMMAND directory by right-clicking on a free spot in the COMMAND folder and choosing **New, Text Document**. You'll see an error message when you're renaming the file since you're changing its extension from TXT to BAT. Make this message go away by clicking **Yes** and load the file in a text editor.

Type in the command **Dir > LPT1** and save the file. If you'd like to save the contents of this folder to a text file rather than printing it out on paper, try using the command **Dir > Filelist.txt**.



Rename

If you change a filename extension, the file may become unusable.

Are you sure you want to change it?

Illustration 32: You'll see this message when you change the extension from TXT to BAT

Now change to the Registry Editor and search for the key **HKEY_CLASSES_ROOT\Directory\shell**. Right-click **\shell** and choose **New, Key**. Give the new key the name **Print Folder Contents**, right-click it, and choose **New, Key** once again. This time, name the key **Command**. On the right side of **\Command** you'll find the entry called "Default," which you should double-click. Now you just need to enter **c:\windows\command\printdir.bat** into the dialog box. In the future, you'll be able to right-click any folder and choose **Print Folder Contents** from its context-sensitive menu. With this trick, you've taught Explorer a trick that comes from the dark old days of DOS.

Expanding the Context-Sensitive Menu for the Start Button

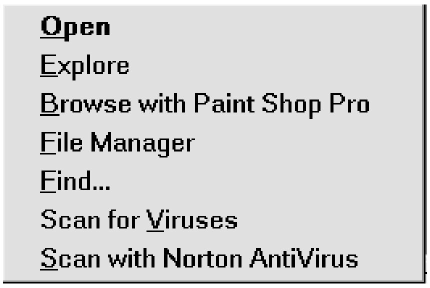


Illustration 33: The context-sensitive menu for the Start button is comparable to the context-sensitive menu of a normal folder

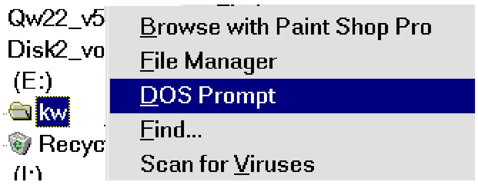
Perhaps you've already discovered that you can right-click the Start button and open a menu that contains various commands. Depending on what software you've installed, you may see entries for **Explorer** and **Open**, as well as **Find**, **Find Viruses**, and more.

In principle, this menu is just like a context-sensitive menu for a folder, since the folder C:\WINDOWS\STARTMENU is behind the Start menu. If you'd like to expand this menu, for example with a file manager, you'll proceed exactly as you would for any other folder's context-sensitive menu. The point in the Registry where you'll need to start is **HKEY_CLASSES_ROOT\Folder\shell**.

Right-click **\shell**, choose **New, Key**, and name the new key **File Manager**. Then create a subkey called **Command** under **\File Manager**. On the right-hand window for **\Command**, you'll find the entry "Default," which you should assign the value **c:\windows\winfile.exe** with a double-click of the mouse.

DOS Prompt Calls with Flexible Directories

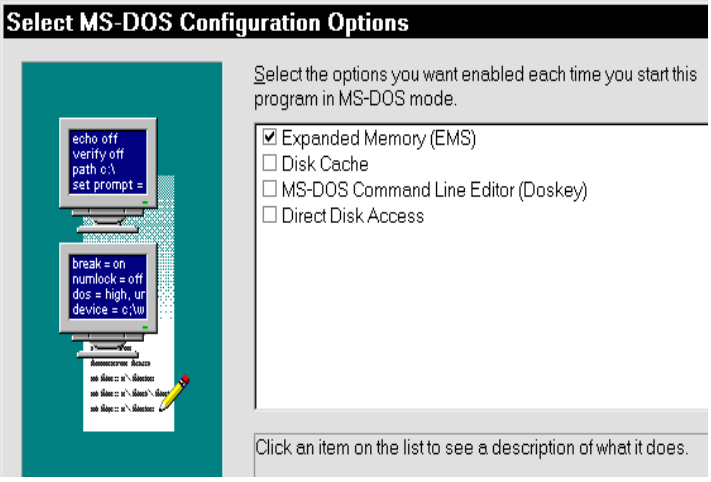
An annoying feature of Windows is that if you start up the MS-DOS command line, you'll always land in the folder C:\WINDOWS\>. This means that you'll have to use the command **CD..** or **CD Directoryname** to change to the folder you really want to be in. With a stroke of genius, you can change this using the power toys that Microsoft has given you. We'll just expand the context-sensitive menu of all folders to include the command **DOS Prompt**, which will give you a command line (the so-called prompt) and change to the folder from whose context-sensitive menu you chose the command.



Here's how: under the key **HKEY_CLASSES_ROOT\Folder\shell**, create a subkey called **DOS Prompt** by choosing **New, Key**. Then, create a subkey under **DOS Prompt** called **Command**. After you've created **\Command**, double-click the string "Default" and assign it the value **Command.com**.

Now each folder's context-sensitive menu will include the useful command **DOS Prompt**.

A Toolbox for Customized DOS Mode



Select MS-DOS Configuration Options

Select the options you want enabled each time you start this program in MS-DOS mode.

☑ Expanded Memory (EMS)
☐ Disk Cache
☐ MS-DOS Command Line Editor (Doskey)
☐ Direct Disk Access

Click an item on the list to see a description of what it does.

Illustration 34: Some important options are missing here

Windows offers you the opportunity to assign each DOS program its own MS-DOS mode with variable settings. This is useful, if, for example, DOS games don't run well with the default settings or you need to load special drivers. In order to set up such a customized MS-DOS mode, right-click the program file with the extension EXE and **choose Properties, Program, Advanced**. Activate the option **MS-DOS Mode**. If you want to set up a special configuration, you have to choose the option **Specify a new MS-DOS Configuration**. Now you'll see Windows' default settings for the special AUTOEXEC.BAT and CONFIG.SYS. You'll see further options if you click the **Configuration** button.

Many games require expanded memory, so you should carry over the already activated EMM386 driver into your configuration. In addition, consider using Smartdrive, which speeds up access to the hard disk by installing a cache system. The configuration options offered here are by no means the only ones that Windows has to offer. There are more possible settings that are deactivated in the Registry by default. To change this, have a look at the key **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\MS-DOSOptions**. The subkeys available there include expanded memory (**\EMS**) and Smartdrive (\ **Smartdrv**) and elements like **\CD-ROM** or **\Mouse**, which are not visible on the settings dialog box. Whether an element appears or not depends on the value of the entry "Flags". Try clicking the subkey **\Smartdrv**, look at the entry "Flags," and you'll see a value of "02 00 00 00" on the right-hand side. The 02 means that this option appears on the configuration window, but it's not activated by default. Under **\EMS**, by contrast, the value is "1b 00 00 00." This means that the entry will appear and that it's already activated.

CD-ROM Support for DOS

DOS games need access to the CD-ROM drive, therefore you need to integrate the CD-ROM drive into your DOS configuration. Click **MS-DOSOptions\CD-ROM** and, on the right-hand side, change the value of "Flags" to 02 00 00 00. Right-click the program file and choose **Properties, Program, Expanded, Configuration** and you'll see that the list has been expanded to include the entry "CD-ROM." Checking the boxes in front of these options doesn't accomplish anything since Windows doesn't know the name and location of the CD-ROM driver and the syntax of the loading command(s) that are added to one of the start files when these options are activated. To specify the CD-ROM driver and loading command(s), right-click **\CD-ROM** and choose **New, String Value**. Give your new string the name CONFIG.SYS. Double-click on the newly-created entry and add the appropriate command line or loading commands. What you'll type here depends on your CD-ROM drive, since each manufacturer gives its driver files different names. Here's an example:

```
DEVICEHIGH=C:\CDROM\MTMCDAI.SYS /D:MTMIDE01
```

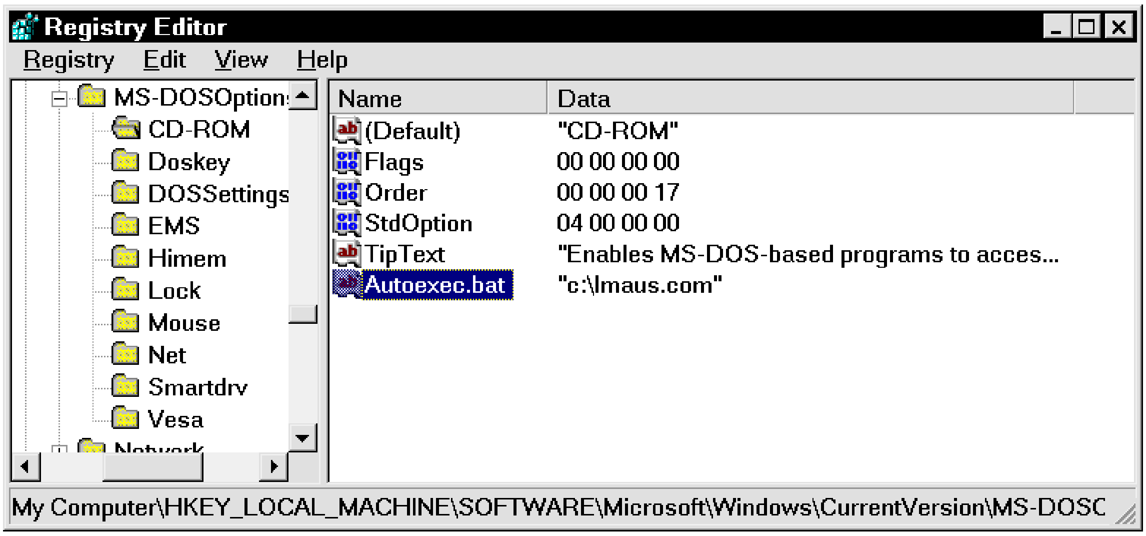
The driver file for this Mitsumi drive is called MTMCDAI.SYS and it's located in the directory C:\CDROM\. The drive is given the name MTMIDE01. Later on, this is important. You should have the driver file available on a diskette. Right-click **\CD-ROM** once more and **choose New, String Value** to create the entry AUTOEXEC.BAT, and then double-click on it. In AUTOEXEC.BAT, you'll need to include the loading instructions for MSCDEX, for example

LH C:\WINDOWS\COMMAND\MSCDEX.EXE /D:MTMIDE01 /M:10

The most important thing is that the same name appear next to /D: in both AUTOEXEC.BAT and CONFIG.SYS; in our example this should be MTMIDE01.

Now, if you've checked an option to activate it when configuring a DOS game (CD-ROM support is a good example), the appropriate commands will be added to the start files. With a single intervention, you've made the future configuration of all DOS applications a lot easier.

Mouse Support for DOS



Name	Data
(Default)	"CD-ROM"
Flags	00 00 00 00
Order	00 00 00 17
StdOption	04 00 00 00
TipText	"Enables MS-DOS-based programs to acces...
Autoexec.bat	"c:\lmaus.com"

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\MS-DOSC

Illustration 35: My mouse driver is called lmaus.com

Mouse support is something else you just can't do without. You can reveal the configuration options that Windows is hiding as follows: on the right-hand side of **\MS-DOSOptions\Mouse**, change the value of "Flags" to "2". Now you're just missing the command with the appropriate mouse driver. This varies according to the mouse. If it has the extension SYS, for example MOUSE.SYS, the call from CONFIG.SYS must follow. After right-clicking **\Mouse** and choosing **New, String Value**, create an entry called **Config.sys**. Double-click it and assign it the command that will call up the mouse driver, for example

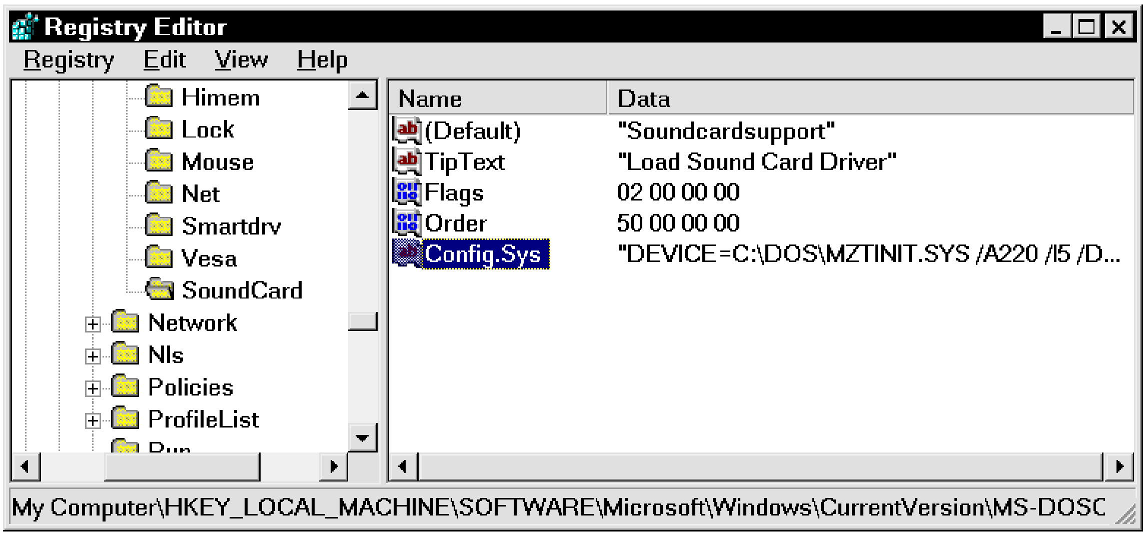
DEVICEHIGH=C:\DRIVER\MOUSE.SYS.

If your mouse driver has the extension COM, it must be loaded through AUTOEXEC.BAT. To do this, choose **New, String Value**, and give the string the name **Autoexec.bat**. The value must be in a form like

LH C:\DRIVER\MOUSE.COM

where of course you'll have to customize the path and filename.

Sound Card Support for DOS



Name	Data
(Default)	"Soundcardsupport"
TipText	"Load Sound Card Driver"
Flags	02 00 00 00
Order	50 00 00 00
Config.Sys	"DEVICE=C:\DOS\MZTINIT.SYS /A220 /I5 /D...

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\MS-DOSC

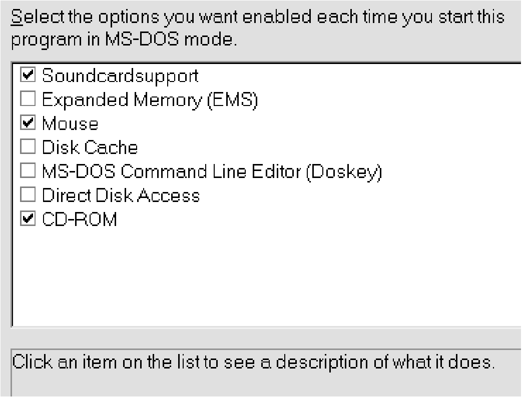
Illustration 36: Without a sound card, games are only half as much fun

Now that we've revealed two unused entries, let's create a completely new entry for the obligatory sound card. Right-click the key **\MS-DOSOptions**, choose **New, Key**, and give the new key the name **Soundcard**.

Double-click the entry "Default" and assign it the value **Soundcardsupport**. Right-click **\Soundcard** and choose **New, String Value** to create the entry **TipText**. Double-click on it and type in a description, for example **Load Sound Card Driver**. Next, you'll need to create a new entry named "Flags" using **New, Binary Value**, and assign it the value **02 00 00 00**. Similarly, add a binary value "Order" and assign it the value **50 00 00 00**. Last but not least, add the command that will load the driver: right-click **\Soundcard**, choose **New, String Value**, and create the entry CONFIG.SYS. Now you have to know what the sound card driver is called, where you've saved it, and what the correct call looks like. To find this information, check in the literature that came with your sound card. Many times there will be text and other help files on the diskette or CD that contains the drivers. Here's an example:

DEVICE=C:\DOS\MZTINIT.SYS /A220 /I5 /D1 /N /G /Q /V5

Once you understand the principle, you can expand the configuration dialog box to your heart's delight. In any case, you now have the most important configuration options for DOS programs (CD-ROM, sound card, and mouse) available to you with a simple click of the mouse button.



Select the options you want enabled each time you start this program in MS-DOS mode.

☑ Soundcardsupport
☐ Expanded Memory (EMS)
☑ Mouse
☐ Disk Cache
☐ MS-DOS Command Line Editor (Doskey)
☐ Direct Disk Access
☑ CD-ROM

Click an item on the list to see a description of what it does.

Illustration 37: And if that wasn't a cool move...

Music Lessons for Your Programs

Windows offers you the opportunity to accompany program events like starting up Explorer with sounds. You'll find these settings under **Control Panel, Sounds**. By default, only a few programs are supported, but you can change this in the Registry. How would you like an accoustic greeting when you start up Winword?

Call up the Registry Editor and change to the key **HKEY_CURRENT_USER\AppEvents**. The subkey **\EventLabels** describes all program events that can be supported by sounds and gives them names. Here, for example, you'll find events like "Open" and "Close," that is, the opening and closing of a program. You'll find an exact description of the individual events in the table below.

EVENT NAME	EXPLANATION
.Default	Default signal
AppGPFault	Program error
Close	Close program
EmptyRecycleBin	Empty the Recycle Bin
Maximize	Maximize program
MenuCommand	Menu command, for example, File, Open
MenuPopup	Opening of a menu (menu bar or context-sensitive menu)
Minimize	Minimize program
Open	Open program
RestoreDown	Make program smaller
RestoreUp	Make program larger
SystemAsterisk	Asterisk (Star), Message from Windows 95
SystemExclamation	Hint
SystemExit	Close Windows
SystemHand	Critical error
SystemQuestion	Question
SystemStart	Start Windows

Table 1: Windows recognizes these events

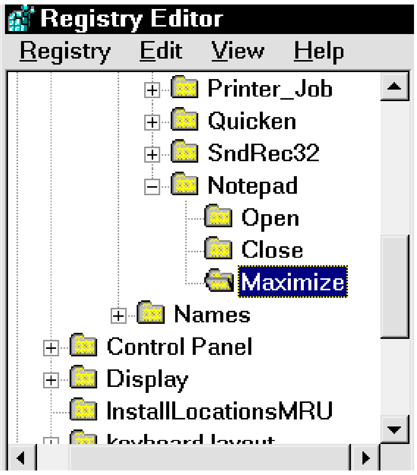
The subkey **\AppEvents\Schemes** is divided into **\Apps** and **\Names**. The latter key administers the individual sound schemes for Windows. By default, Windows has the "Jungle Sound Scheme," "Music Sound Scheme," Robotz Sound Scheme," and the "Utopia Sound Scheme." If you didn't install these when you installed Windows, you can add them using **Control Panel, Add/Remove Programs, Windows Setup, Multimedia**.

If you didn't install some or all of the schemes, don't worry; later on, we'll create our own sound scheme. The key **\Apps** contains the exact settings that determine which sound will accompany which program event. It's here that we must begin if we want to expand the very meager list of programs and events.

It's easiest to explain the principle using the existing subkey **\Sndrec32** as an example. The name stands for the program file SNDREC32.EXE of the Windows Audiorecorder. Here, you'll find two default subkeys, **\Open** and **\Close**. If you click one of these subkeys, you'll see a further breakdown into the individual sound schemes. If you haven't installed any sound schemes, you'll only find the default entry **\.Current**. On the right side you could specify the WAV file to play under "Default," for example C:\SOUNDS\OPEN.WAV-- but that's much easier to do later using the Control Panel.

Example: Teaching the Editor About Sounds

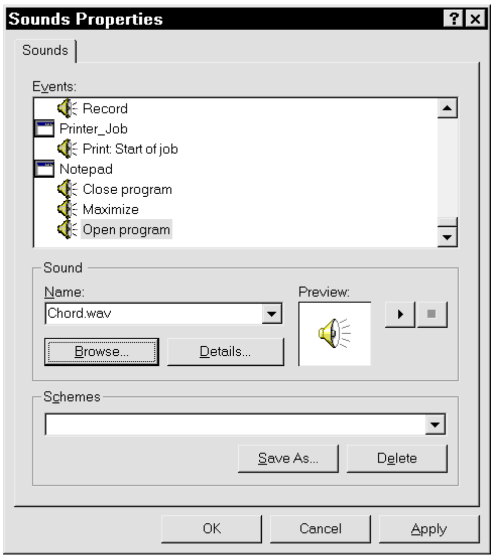
Just as by default there are audible events called "Open" and "Close" for the Audiorecorder, now we'll create something just like them for the editor. Right-click the key **\Apps** and choose **New, Key**. Name the new key **Notepad**, since the program file for the editor is called NOTEPAD.EXE. Next, right-click **\Notepad**, choose **New, Key** again, and call the new key **Open**. In the same way you should create a subkey called **Close**.



Under **Control Panel, Sounds** you'll find the editor **Notepad**, which as a supported event, offers you the entries **Open Program** and **Close Program**. Now, under Name, you can specify a WAV file, whose name you can type in the box or find by clicking the **Browse** button.

Once you understand the principle, you can expand the event list as much as you wish. For example, under the key **HKEY_CURRENT_USER\AppEvents\Schemes\Apps\** you could add the subkey **\Winword**, which could, in turn, contain the subkeys **\Open**, **\Close** or another event such as **\Minimize**. To do this, consult the above table; it contains all possible events.

In doing this, you must be aware of two things: events like Systemstart can't be assigned to a program like Winword – naturally this event marks the startup of Windows.



You should also know that certain events can overlap one another. For example, let's take the events "MenuPopup" (the opening of a menu) and "MenuCommand" (the carrying out of a menu command). After a click on a menu you'll hear one sound, after a click on a menu command, you'll hear another. If the first sound is too long, then the second can't be played on time, so it will be omitted. You should think about this when tinkering with the sounds on your system.

If you've succeeded in arranging all the sounds for programs and events according to your wishes, you can save the whole thing under a name to create a so-called sound scheme. Just try this out; soon you'll have only "musical" programs.

Other Stuff

Opening Unknown File Types with the Editor

If you don’t want to open unregistered file types with the editor by default, you must venture again into the Registry. Call up the Registry Editor, right-click **HKEY_CLASSES_ROOT\Unknown\shell** and choose **New, Key**. Call the subkey you just created **command**. After clicking **\command**, you’ll see the entry “Default” on the right-hand editor window. Double-click this entry and assign it the value **Notepad.exe %1**. In the future, all unknown file types will first be opened with the editor.

Welcome Screen with Tips

By default, each time you start up, Windows presents you with a welcome screen that offers you a new tip about how to work with the operating system. If you’ve deactivated this function by unchecking the option **Show me this message each time I start Windows**, then this adviser has disappeared forever. However, with a venture into the Registry you can conjure him back up onto your screen. To do this, change to the key **HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Tips,** and change the binary value „Show“ from „00 00 00 00“ to „01 00 00 00“.

Your own, customized tips

It’s possible to customize the daily tips on the welcome screen to suit your wishes, or even to add new entries. It’s possible, for example, to insert a collection of sayings so that upon startup, the user will be greeted with a joke. Open the Registry Editor and search for the key **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\explorer\Tips**. If you double-click a number on the right-hand window, you can change the daily tip in the dialog box that appears.

You can also right-click the key **\Tips** and choose **New, String Value** from the context-sensitive menu in order to create a completely new tip with a new number.

Tools for Working With the Registry

Fear of intervening in the system or insufficient knowledge on the part of many users has lead countless shareware and freeware authors to create many programs and tools that you can use to work with the Registry. The advantage for you is that you can get these programs for free from shareware CDs or download them off the Internet. So that this software model won't be misused, let me explain for you precisely what shareware and freeware are.

What's Shareware, What's Freeware?

The principle of shareware is simple: you get a program in one of the abovementioned ways and you install it. Then you have an evaluation period (usually 30 days). If you're satisfied, then you must pay a registration fee, which is usually between 5 and 30 dollars US.

Often evaluation versions contain messages that remind the user to register the program. Unfortunately, some shareware authors build so many hints and obstacles into their software that the program can't be used sensibly. The problem: the ability to evaluate a program before buying it is a clear advantage for the user, since no one likes to buy the cat in the bag. Most of the time, the programmer will have a lot of trouble collecting his well-earned pay. So heed my advice: support the shareware principle and register your tools if you use them a lot.

By contrast, you don't have to register freeware. As with shareware, you can get freeware for nothing on CDs or on the World Wide Web and you can use it as much as you want without having to pay a penny for it. As with Shareware, you can copy the program legally for distribution to your friends and acquaintances.

Shareware and Freeware on the Internet

Once upon a time, one got shareware and freeware on diskettes and CD-ROMs. Today, these media are being supplanted by the Internet, which is why I've researched a few links for you where you can find many programs without searching for a long time:

`www.filepile.com`

`www.winfiles.com`

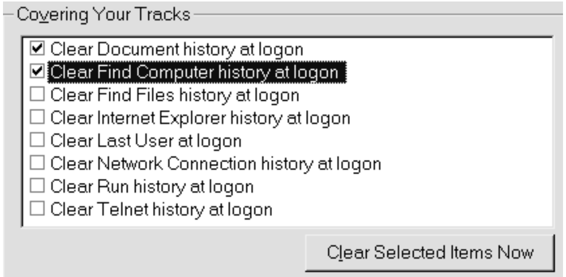
`www.shareware.com`

`www.download.com`

In addition, I would recommend the address www.winfiles.com, since here you'll find all kinds of things that are relevant to Windows.

TweakUI - Microsoft’s Trick Box

TweakUI is one of the Microsoft power toys that are available for free for Windows 95 and 98. Here Windows 98 users have an advantage: the program can be found directly on the Windows CD in the directory \TOOLS\RESKIT\POWERTOY. In addition, the version for Windows 98 has been greatly improved. The program consists only of the file TWEAKUI.INF. All you have to do is right-click it with the mouse and choose **Install**. The tool will then be integrated into the Control Panel, where you can use it to customize hidden settings in thematically sorted registers.



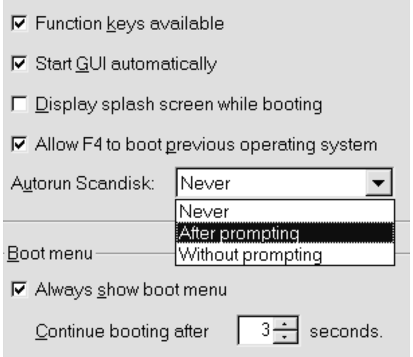
Covering Your Tracks

☑ Clear Document history at logon
☑ Clear Find Computer history at logon
☐ Clear Find Files history at logon
☐ Clear Internet Explorer history at logon
☐ Clear Last User at logon
☐ Clear Network Connection history at logon
☐ Clear Run history at logon
☐ Clear Telnet history at logon

Illustration 38: Automatic Trace Removal

In addition to changing basic settings like the one that controls how fast menus open and the settings for the boot menu, under Windows 98 you can specify, for example, that certain drives will not appear in Explorer and certain elements of the Control Panel.

You can even shut off the annoying feature of Windows that causes it to run Scandisk every time the system crashes. But do remember to run Scandisk manually from time to time.

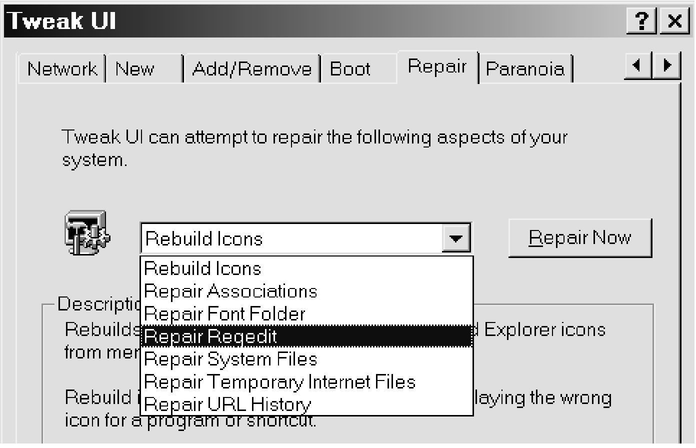


☑ Function keys available
☑ Start GUI automatically
☐ Display splash screen while booting
☑ Allow F4 to boot previous operating system
Autorun Scandisk: Never

Boot menu
☑ Always show boot menu
Continue booting after 3 seconds.

Illustration 39: Boot options under scrutiny

A noteworthy feature of TweakUI for Windows 98 is the various repair features that can help you solve small problems quickly and elegantly. All in all it’s a tool that users of the newer Windows versions should not let rot on their CDs.



Tweak UI can attempt to repair the following aspects of your system.

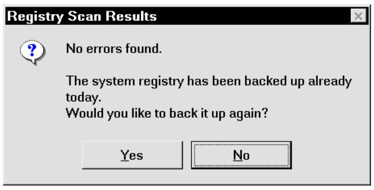
Rebuild Icons

Illustration 40: Windows repairs made easy

Scanreg – The Registry Examiner of Windows 98

As one of the many new tools that come with Windows 98, the operating system now offers a program called “Scanreg,” with which you can scan the Registry for errors and automatically optimize it. Even if you’ve never started the program, it’s working from the first time you boot up each day, in that it creates backup copies of SYSTEM.DAT, USER.DAT, WIN.INI and SYSTEM.INI for you. Note that this doesn’t apply to Windows 95, where the backup copies of USER.DA0 and SYSTEM.DA0 live in the Windows directory (in case you don’t see these files, open Explorer and choose **View, Folder Options, View, Show all Files**).

Windows 98 with Scanreg stores the two Registry files, as well as WIN.INI and SYSTEM.INI, in compressed form as RB000.CAB to RB005.CAB in the folder C:\WINDOWS\SYSBCKUP. Here you’ll find automatically-created backup copes for the last five days, so that you only need to call up the program using **Start, Run, Scanregw** (the **w** stands for the **W**indows version) if you’d like to examine the Registry for errors. After scanning for errors, you’ll see a dialog box stating that you’ve already backed up these files today. If you wish, however, you can back them up again.



No errors found.

The system registry has been backed up already today.
Would you like to back it up again?

While booting, before Windows makes the backup copy, Scanreg examines the Registry. If it finds that the deletion of old entries has created unused but not freed-up space, it will optimize the Registry without the user’s intervention.

Just so you know, CAB is the default format for Windows installation files (see the CD), so Windows 98 has no problems with this file type. If you wish, you can examine the archive with a double-click and extract files without even using a packer like Winzip. Under DOS, you generally need to use the command **Extract** for CAB files, but in the case of the Registry backup, things are easier: start in MS-DOS mode and type **Scanreg /Restore** into the Windows directory. Alternatively, you can use **Scanreg /Fix** to fix mistakes. If you would like to make a backup of DOS, just type in **Scanreg /Backup**.

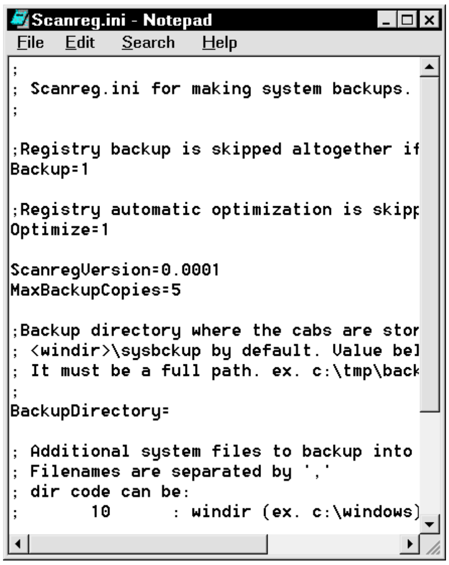
Customizing and Expanding Scanreg

Scanreg even has some further, hidden features up its sleeve: the most important is surely its ability to back up additional files at startup. To do this, you must load the file SCANREG.INI in the WINDOWS folder into the editor.

If, for example, you’d like to back up AUTOEXEC.BAT, CONFIG.SYS and MSDOS.SYS in addition to the CAB archive when you’re booting up, add the following to the last line:

Files=31,config.sys,autoexec.bat

If you’re of the opinion that you don’t need automatic backups, you can change the value *BACKUP=1* to **0**. But I wouldn’t recommend doing this – not even if you’re anxious to save every possible second at bootup in order to make your PC available immediately for incoming faxes. With the entry *MaxBackupCopies=x* you determine how many backup copies will be made before they’re overwritten. Last but not least, *BackupDirectory=* lets you specify a directory other than the default C:\WINDOWS\SYSBCKUP.



```
;
; Scanreg.ini for making system backups.
;

;Registry backup is skipped altogether if
Backup=1

;Registry automatic optimization is skipp
Optimize=1

ScanregVersion=0.0001
MaxBackupCopies=5

;Backup directory where the cabs are stor
; <windir>\sysbckup by default. Value bel
; It must be a full path. ex. c:\tmp\back
;
BackupDirectory=

; Additional system files to backup into
; Filenames are separated by ','
; dir code can be:
;       10      : windir (ex. c:\windows)
```

Illustration 41: As you can see, INI files are largely self-explanatory

Registry Inkognito

Registry Inkognito is *the* program if you're looking to reveal the hidden options of Windows 95. It's only interesting for users of Windows 95 since it has approximately the same features as TweakUI for Windows 98. In various thematically-ordered registers you'll find everything you need to make Windows 95 to suit your personal tastes. Beginning with the deactivation of the logo during bootup, continuing with the setting that controls the pause in the Start menu, and concluding with the cleaning out of the tasklists on the Start menu, here you'll find nearly everything you can think of. The crowning glory here is the mean trick that lets you remove the shutdown command from the Start menu. Have fun…

Registry Inkognito is shareware and the registration fee is about $ 14 US for private users. Commercial users, by contrast, pay about $ 31 US. At least for home users, this is a fair price for a program like this, even if, as a reader of this booklet, you already know nearly all the tips and tricks you need to do the same things.

With the help of Registry Inkognito, you can avoid interventions in the Registry without having to forego the hidden possibilities.

WinHacker 95 – Hacking for Beginners

WinHacker is a useful tool with which you can accomplish and manipulate many things, including hiding selected drives in Explorer and removing the annoying arrows on shortcut icons. You can also teach Windows that the icon captions for shortcuts shouldn't always begin with "Shortcut to…", and much more. There's also the brilliant possibility to expand the context-sensitive menu for folders to include the command **Explore from here**. If you add this command and then right-click a folder and choose it, you'll see a new Explorer window with the selected folder as the topmost entry. Similarly, you can expand the context-sensitive menus to include the command **MS-DOS-Prompt**, which, when you choose it, will produce a DOS window with the selected folder as the active directory. You can even change the icons for system components like the Desktop. With WinHacker 95 you'll get a true power program, that's well worth the $17.95 US registration fee.

Policy Editor

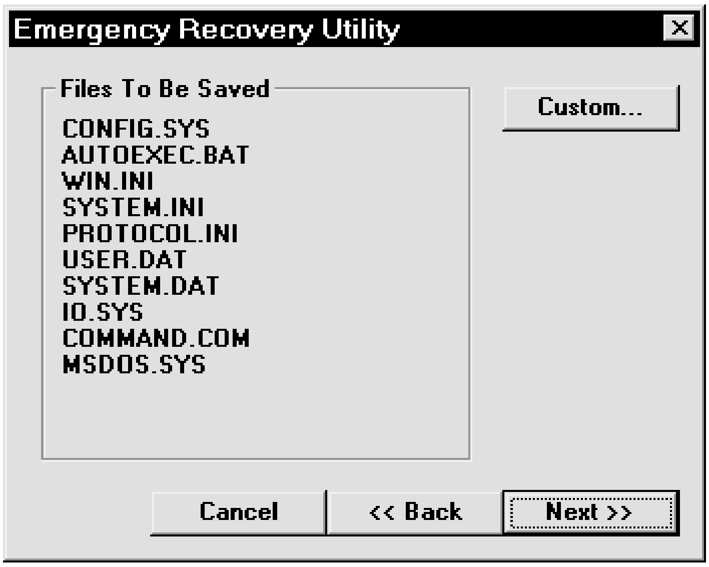
If you work with different user profiles, I would definitely recommend the Policy Editor. You'll find this little program on your Windows 95 CD in the directory \ADMIN\APPTOOLS\POLEDIT. This recommendation applies to Windows 98 too; here you'll find the program in the CD directory \TOOLS\RESKIT\NETADMIN\POLEDIT.

What are User Profiles and System Policies?

If, for example, a father wishes to let his son work on his PC, but he'd like to protect himself against the consequences of childish curiosity, he can "lock up" particular parts of Windows. He can create a user profile for his son and set system policies. He could, for example, "lock" the **Display** icon in the **Control Panel** so that his son could not change the settings for the graphics card. In this manner, you can limit access to the system according to who's using it. The various user profiles are selected when you enter your name at startup, and it is at this time that the system policies are loaded.

ERU - Windows' Free Security Tool

ERU stands for Microsoft's "Emergency Recovery Utility," which you'll find in the \OTHER\MISC\ERU directory on your Windows 95 CD. This little tool will backup system-relevant files in a directory or onto a diskette so that, in case of emergency, you can boot from it. Under Windows 98, there is no more ERU.



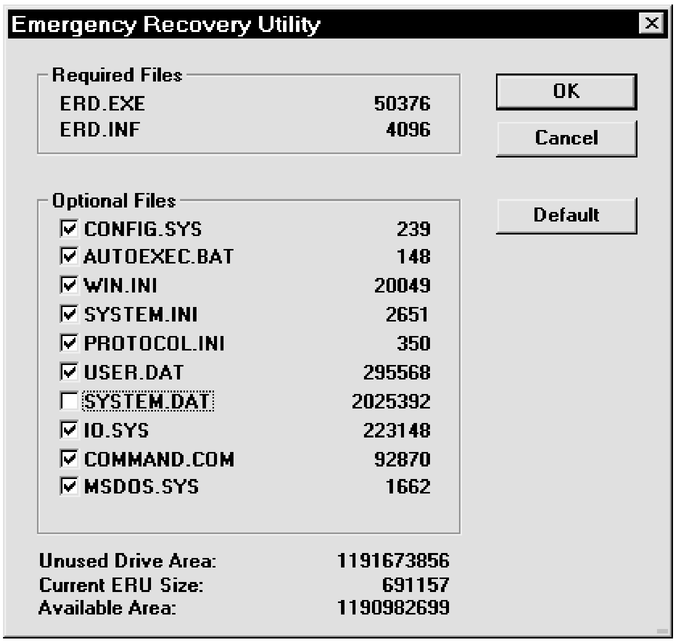
Emergency Recovery Utility

Files To Be Saved

CONFIG.SYS
AUTOEXEC.BAT
WIN.INI
SYSTEM.INI
PROTOCOL.INI
USER.DAT
SYSTEM.DAT
IO.SYS
COMMAND.COM
MSDOS.SYS

Illustration 42: Something's missing here...

ERU will ask you for a formatted system diskette, which, since it's already storing the system files, is often not big enough to hold SYSTEM.DAT. But this differs from computer to computer and is determined primarily by how many programs and how much hardware you've installed, actions which, with their entries, swell the size of the Registry. You can either back these files up on a second diskette (consider compressing them with a program like Winzip) or store them in another directory on the hard disk.



Emergency Recovery Utility

Required Files

ERD.EXE	50376
ERD.INF	4096

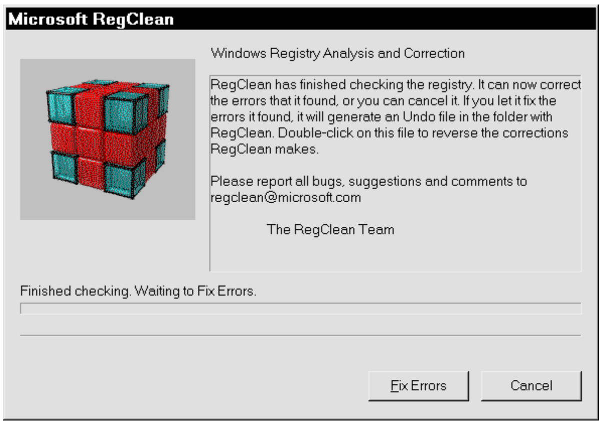
Optional Files

☑ CONFIG.SYS	239
☑ AUTOEXEC.BAT	148
☑ WIN.INI	20049
☑ SYSTEM.INI	2651
☑ PROTOCOL.INI	350
☑ USER.DAT	295568
☐ SYSTEM.DAT	2025392
☑ IO.SYS	223148
☑ COMMAND.COM	92870
☑ MSDOS.SYS	1662

Unused Drive Area:	1191673856
Current ERU Size:	691157
Available Area:	1190982699

Illustration 43: SYSTEM.DAT is too big for this diskette

RegClean – The Free Cleaner



Microsoft RegClean

Windows Registry Analysis and Correction

RegClean has finished checking the registry. It can now correct the errors that it found, or you can cancel it. If you let it fix the errors it found, it will generate an Undo file in the folder with RegClean. Double-click on this file to reverse the corrections RegClean makes.

Please report all bugs, suggestions and comments to regclean@microsoft.com

The RegClean Team

Finished checking. Waiting to Fix Errors.

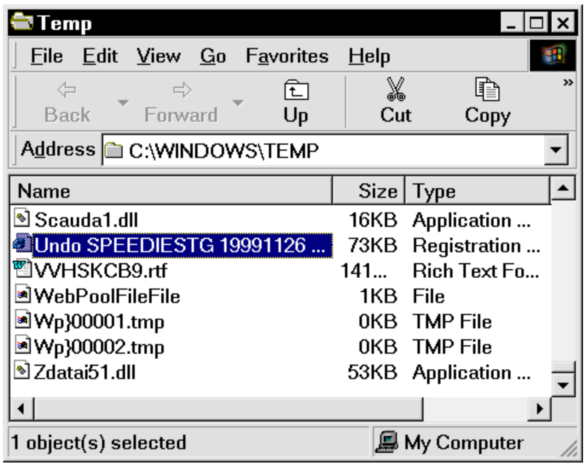
With the constant installation and uninstallation of applications under Windows, the Registry can get bloated with unnecessary entries, which can cost you time at startup and system stability. With RegClean, Microsoft offers the user a tool that can automatically seek out “orphaned” entries in the Registry and remove them. You can get RegClean for free over the Internet, either directly from the Microsoft Web site (www.microsoft.com) or in countless other software archives. These days, you’ll find it on nearly every page that concerns itself with Windows (tools, tips, and solutions to problems).

After a double-click on the program file, all relevant settings will be read in. Then, if you click **Fix Errors**, unnecessary entries will be removed.

The program directory also contains an Undo file, which, if you double-click on it, can undo your changes in case you encounter problems.

Under Windows 98 RegClean will work, but you should also consider using Scanreg to look for errors since RegClean will only find unnecessary entries.

You can download this program from www.download.com and search using the keyword „regclean.” You should find a link that will lead you to a site where you can download the current version.



C:\WINDOWS\TEMP

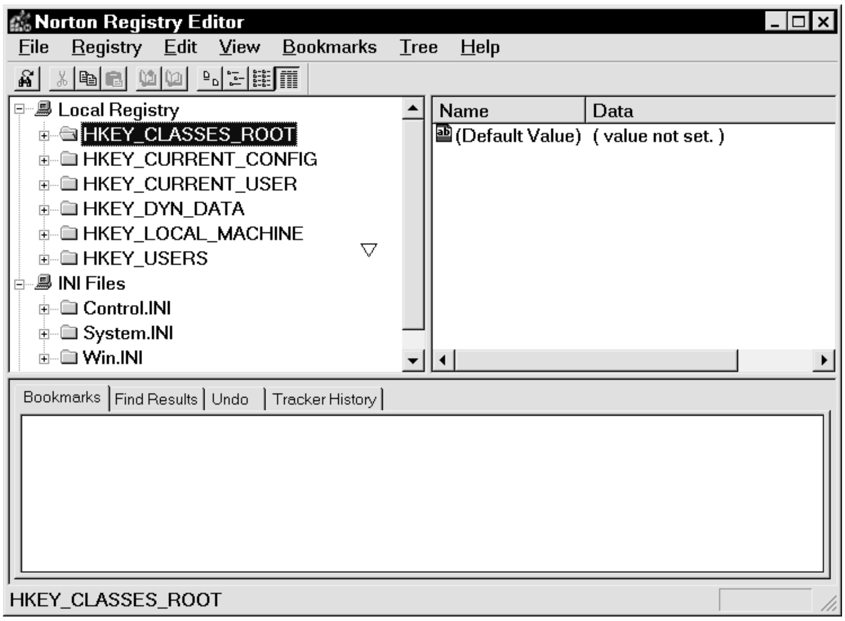
Name	Size	Type
Scauda1.dll	16KB	Application ...
Undo SPEEDIESTG 19991126 ...	73KB	Registration ...
VVHSKCB9.rtf	141...	Rich Text Fo...
WebPoolFileFile	1KB	File
Wp}00001.tmp	0KB	TMP File
Wp}00002.tmp	0KB	TMP File
Zdatai51.dll	53KB	Application ...

1 object(s) selected

My Computer

Illustration 44: The Undo File with Date and Time Stamps

Norton Registry Editor and Registry Tracker



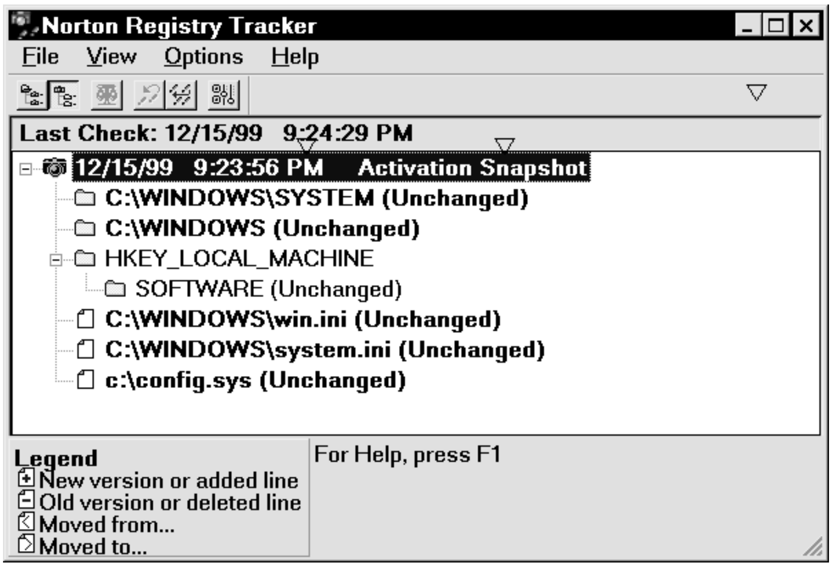
Local Registry
- HKEY_CLASSES_ROOT
- HKEY_CURRENT_CONFIG
- HKEY_CURRENT_USER
- HKEY_DYN_DATA
- HKEY_LOCAL_MACHINE
- HKEY_USERS

INI Files
- Control.INI
- System.INI
- Win.INI

Name	Data
(Default Value)	(value not set.)

HKEY_CLASSES_ROOT

If you own the Norton Utilities, then you've already got both the "Norton Registry Editor" and the "Norton Registry Tracker." The first is an improved version of Microsoft's Regedit, which provides explanatory help text about the most important keys on the bottom half of the screen. This will help orient you in the Registry. In addition, the Norton Registry Editor contains many useful features, such as the ability to create shortcuts within the Registry. For example, you can create a shortcut from the level of the main keys to a key that lies deep in the hierarchy of the Registry so that you don't have to click through the "key jungle" each and every time. Furthermore, the Registry Editor is a good editor for INI files. The program automatically loads the important INI files (WIN.INI, SYSTEM.INI and CONTROL.INI); you can load others permanently into the Editor by loading them just once. The processing proceeds according to the Registry scheme: the [Groups] are shown as subkeys, and the entries and values are displayed on the right-hand side.



Last Check: 12/15/99 9:24:29 PM

12/15/99 9:23:56 PM Activation Snapshot
- C:\WINDOWS\SYSTEM (Unchanged)
- C:\WINDOWS (Unchanged)
- HKEY_LOCAL_MACHINE
  - SOFTWARE (Unchanged)
- C:\WINDOWS\win.ini (Unchanged)
- C:\WINDOWS\system.ini (Unchanged)
- c:\config.sys (Unchanged)

Legend
- New version or added line
- Old version or deleted line
- Moved from...
- Moved to...

For Help, press F1

The second programm, Norton Registry Tracker, reports on changes to the Registry. This is useful if you're installing a new program, since it allows you to compare the original and new versions of a key. It's also very useful for venturing into the Registry in case you've made a mistake. Also useful: you can monitor further files such as CONFIG.SYS.

Last but not Least

I hope that by the end of this booklet you've learned something and that your Windows installation now suits your needs much better. I thank you for buying and reading this booklet – and for being so courageous about exploring the depths of Windows, especially the Registry. Since, as the reader, you're the last and most important link in the chain of all booklets, I'm interested in knowing what you think. Write me if you have praise or criticisms, ideas, or suggestions for improvement. My e-mail address is andre.moritz@gmx.net.

If you have an Internet connection and you're interested, check us out on the Internet at www.knowware.com. There you'll find information about new, old, and planned booklets, online ordering instructions, tips and tricks, and much more. Have fun!

Berlin, September 1998
André Moritz
andre.moritz@gmx.net