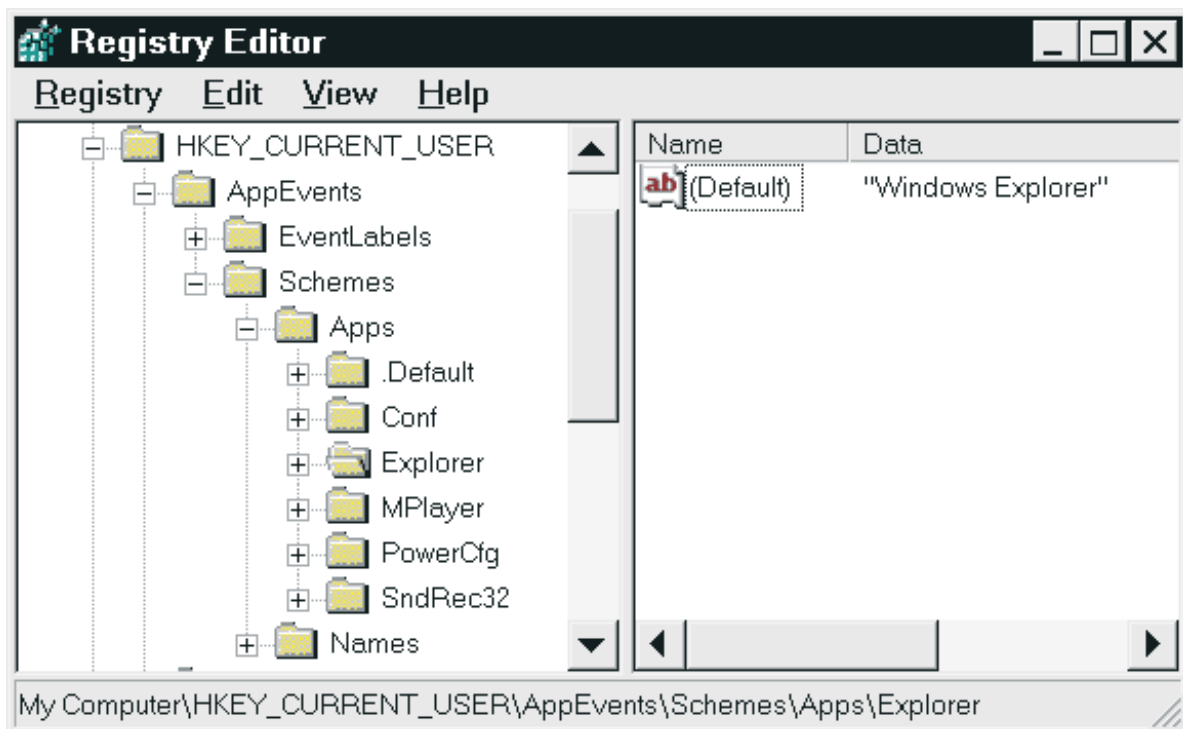


Free!

Windows-Tuning mit der Registry



- * Was ist die Registry?
- * Grundlagen
- * Wege zum Editieren der Registry
- * Wo finde ich welche Einstellungen?
- * Sichern der Windows-Einstellungen
- * Kontextmenüs anpassen und erweitern

- * Troubleshooting - Wenn der Computer streikt
- * Windows aufräumen
- * Optisches Tuning
- * Versteckte Möglichkeiten nutzen
- * Shareware-Tools
- * Viele weitere Tips & Tricks

Bildqualität

Wenn die Bildqualität in dieser Datei gering ist, liegt es an eine Priorität. Wir versuchen die Dateigrösse klein zu halten, um die Downloadzeit zu kürzen - nicht alle haben DSL. Im Heft ist die Qualität selbstverständlich gut ;)

Acrobat Reader: Wie ...

F5/F6 öffnet/schließt die Ansicht **Lesezeichen**

Strg+F sucht

Im Menü Ansicht stellst du ein, wie die Datei gezeigt wird

STRG+0 = Ganze Seite **STRG+1** = Originalgrösse **STRG+2** = Fensterbreite

Im selben Menü kannst du folgendes einstellen:: **Einzelne Seite**, **Fortlaufend** oder **Fortlaufend - Doppelseiten** .. Probiere es aus, um die Unterschiede zu sehen.

Navigation

Pfeil Links/Rechts: eine Seite vor/zurück

Alt+ Pfeil Links/Rechts: Wie im Browser: Vorwärts/Zurück

Strg++ vergrößert und **Strg+-** verkleinert

Bestellung und Vertrieb für den Buchhandel

Bonner Pressevertrieb, Postfach 3920, D-49029 Osnabrück

Tel.: +49 (0)541 33145-20 Fax: +49 (0)541 33145-33

bestellung@knowware.de

Autoren gesucht

Der KnowWare-Verlag sucht ständig neue Autoren. Hast du ein Thema, daß dir unter den Fingern brennt? - ein Thema, das du anderen Leuten leicht verständlich erklären kannst?

Schicke uns einfach ein paar Beispielseiten und ein vorläufiges Inhaltsverzeichnis an folgende Adresse:

lektorat@knowware.de

Wir werden uns deinen Vorschlag ansehen und dir so schnell wie möglich eine Antwort senden.

Grundlagen	4	Hintergrundbild genau plazieren	30
Was ist die Registry?	4	Icons mit 16 Bit Farbtiefe	30
Microsofts Warnung	4	„Klicken Sie hier um zu starten“	
Versteckte Informationen	5	ausschalten	31
Die Geschichte der Registry	5	Animation beim Minimieren und	
INI-Dateien – Was war das gleich noch		Maximieren	31
mal?	5	Icon für Auflösung und Farbanzahl neben	
Erste Anfänge der Registry	7	der Uhr	31
Vorteile des Registry-Prinzips	8	Windows über die Registry anpassen	32
Der Aufbau der Registry	8	Benutzernamen ändern	32
Übersicht	10	Setuppfade anpassen	32
HKEY_CLASSES_ROOT\	11	Autostart nur von Audio-CDs unterdrücken	32
HKEY_USERS\	12	Speichern von Einstellungen verhindern	33
HKEY_CURRENT_USER\	12	Programme beim Booten versteckt starten	33
HKEY_LOCAL_MACHINE\	12	Umbenennen-Schutz des Papierkorbs	
HKEY_CURRENT_CONFIG\	13	ausschalten	33
HKEY_DYN_DATA\	13	„Abmelden“- und „Favoriten“-Eintrag aus	
Rund um den Registriereditor	14	dem Startmenü verbannen	34
Suchen in der Registry	14	Systemausbau mit der Registry	35
Platz da für die Schlüssel!	15	Kontextmenüs anpassen	35
Lesezeichen erstellen	15	Reihenfolge von Menüeinträgen	35
Notizen erstellen	16	Scandisk im Kontextmenü von Laufwerken	35
Shortcuts für schnelleres Arbeiten	16	Neu-Menü entrümpeln oder erweitern	36
REG-Dateien	17	Neuen Ordner über Kontextmenü erstellen	38
Tips zum Umgang mit REG-Dateien	17	Antivirenskan im Kontextmenü von	
Aufbau von REG-Dateien	17	Laufwerken	38
REG-Dateien selber schreiben?	18	Ordnerinhalt ausdrucken oder in Datei	
Registrierungsbestätigung abschalten	19	speichern	38
Registrierungen vergleichen	19	Kontextmenü von Start erweitern	39
OLE, Drag & Drop und CLSID	20	Promptaufwurf mit flexiblem Verzeichnis	40
Was zum Teufel ist Drag & Drop?	20	Baukasten für individuellen DOS-Modus	41
Was zum Teufel ist OLE?	20	CD-ROM-Unterstützung für DOS	41
Was zum Teufel ist eine CLSID?	21	Maus-Unterstützung für DOS	42
Erste Hilfe für die Registry	22	Soundkartenunterstützung für DOS	43
Registry sichern	22	Musikunterricht für Deine Programme	44
User.dat und System.dat kopieren	22	Beispiel: Dem Editor Klänge beibringen	45
Teilexport der Registry	23	Sonstiges	46
Registrierungsabschnitte vor Eingriffen		Unbekannte Dateitypen mit Editor öffnen	46
drucken	23	Willkommensbildschirm mit eigenen Tips	46
Startdiskette erstellen	24	Tools rund um die Registry	47
Troubleshooting – Nichts geht mehr	25	Was ist Shareware, was ist Freeware?	47
Einzelfehler nach dem Start	25	Shareware und Freeware im Internet	47
Probleme beim Start	25	TweakUI - Microsofts Trickkiste	48
Fehlermeldung nach		Scanreg – Der Registryprüfer von	
Programmdeinstallation	26	Windows 98	49
Systemtuning mit der Registry	28	Registry Inkognito	50
Windows schneller machen und		WinHacker 95 – Hacken für Anfänger	51
aufräumen	28	Policy Editor - Systemrichtlinieneditor	51
Verzögerung im Startmenü einstellen	28	ERU - Windows' kostenloses	
Startmenü – abgespeckt	28	Sicherungstool	52
Ausführen-Liste löschen	28	RegClean – Der kostenlose Reinemacher	53
Uninstall-Liste bereinigen	29	Norton Registry Editor und Registry	
Optisches Tuning	29	Tracker	54
Bitmaps als Icon darstellen	29		

Registry – da gehe ich lieber nicht ‘ran!

Hast Du dies vielleicht auch schon einmal gedacht? Kennst Du Ratschläge wie: „Laß’ da lieber die Finger von“? Verständlich ist das irgendwo schon. Der Gedanke, daß in der Registry von Windows 95 respektive Windows 98 alle Einstellungen des Betriebssystems festgehalten werden, läßt vielen Anwendern ein mulmiges Gefühl in der Magengegend hochkommen. Die Angst, mit einer falschen Einstellung das ganze System lahmzulegen, ist auch nicht unbegründet. Mit diesem Heft in der Hand kannst Du jedoch nichts falsch machen.

Wozu dieses Heft?

„Niemand pflegt akrobatische Übungen vorzunehmen, ohne vorher das Netz gespannt zu haben“. Recht hat der kluge Mann, der dies einmal behauptete (das war bestimmt ein Versicherungsvertreter ☺), und deshalb zeige ich Dir ausführlich, wie Du die Registry richtig sicherst. Keine Angst, wenn wirklich etwas schiefgehen sollte, guckst Du einfach im Abschnitt „Troubleshooting – nichts geht mehr“ vorbei. Dort erkläre ich Dir Schritt für Schritt, wie Du Dein System wieder zum Laufen bringst. So kannst Du beruhigt lernen, was für eine enorme Bedeutung die Registry unter Windows 9x hat. Vielleicht weißt Du noch gar nicht so recht, was „Registry“ überhaupt ist. Das stellt überhaupt kein Problem dar – im Gegenteil. Schließlich habe ich mir wie alle KnowWare-Autoren das Ziel gestellt, gerade Anfängern die manchmal wirklich schwierige Problematik Computer näherzubringen.

Auch wenn Du Dich schon ein wenig auskennst, kannst Du mit diesem Heft noch viel über Dein Betriebssystem lernen. Mit den vielen Tips und Tricks im ganzen Heft findest Du eine Menge gebündeltes Praxis-Know-how und viele Möglichkeiten, versteckte Funktionen von Windows zu nutzen. Außerdem möchte ich Dir verschiedene Programme vorstellen, die Dir die Arbeit mit der Registry erleichtern und manuelle Eingriffe in die Registrierdatenbank nicht mehr unbedingt notwendig machen. Trotzdem solltest Du lernen, wozu die verschiedenen Einträge der Registry gedacht sind und wie Du sie an Deine persönlichen Erfordernisse anpassen kannst.

Grundkenntnisse

Um in die Tiefen der Registry abzutauchen, solltest Du zumindest einige Grundkenntnisse im Umgang mit Windows 95/98 besitzen: Du solltest wissen, wie man mit Maus und Tastatur umgeht, was Dateien und Ordner sind und wie man diese kopiert oder löscht.

Falls Du lange nicht mehr mit Windows gearbeitet hast oder gerade eingestiegen bist, empfehle ich Dir die KnowWare-Hefte „Start mit Windows 95“ und „Windows 95 für Einsteiger“, welche das nötige Grundwissen vermitteln oder auffrischen.

Grundlagen

Was ist die Registry?

Wenn man es kurz auf einen Nenner bringen sollte, ist die Registry die Systemdatenbank von Windows 95/98. Sie enthält alle Einstellungen von der Soundkarte über Winword bis hin zur Mauskonfiguration. Ein so fortschrittliches und für den Anwender bequemes Betriebssystem wie Windows (abgesehen von einigen unschönen Macken – but „nobody is perfect, and nobody’s software, too“) muß eine unvorstellbare Menge an Informationen festhalten. Welches Hintergrundbild wünscht der Anwender? Wie schnell führt er den Doppelklick mit der Maus aus? Welches Programm nimmt er zum Schreiben von Texten, und wie lautet seine Lieblingsschriftart? Will der Anwender 16, 256 oder noch mehr Farben? Von welcher Firma ist sein Drucker, und hatte er nicht auch noch einen Scanner...

Du siehst: Fragen über Fragen. Das alles und noch viel mehr muß sich Windows merken. Die Einstellungen können sich jeden Tag ändern. Vielleicht arbeiten an einem PC mehrere Leute, die alle verschiedene Einstellungen bevorzugen. Für jeden müssen separat seine Vorstellungen vom Betriebssystem verwaltet werden.

Alle diese Informationen unter einen Hut zu bringen, ist fast schon ein Kunststück. Alle diese Einstellungen übersichtlich und für den Anwender verständlich und editierbar in einer Datenbank zu verwalten, das gleicht eigentlich schon einem Wunder.

Microsofts Warnung

Anscheinend sieht es Microsoft nicht besonders gern, wenn Endanwender an der Registry herumspielen. Der Registriereditor ist deshalb standardmäßig auch nicht im Startmenü zu finden. Außerdem enthalten die allgemeinen Hinweise, die Microsoft in Form verschiedener Textdateien beilegt, folgende Warnung:

„Manipulationen, die manuell an der Windows 95-Registrierung vorgenommen werden, können schwerwiegende Auswirkungen auf die Windows 95-Installation haben.

Solche Auswirkungen können von der Fehlfunktion einzelner Komponenten bis hin zur Unbenutzbarkeit des Systems reichen und sind gegebenenfalls nur durch eine Neuinstallation von Windows 95 zu beheben. Microsoft kann keine Gewährleistungen oder Support für Probleme übernehmen, welche durch eine Manipulation der Windows 95-Registrierungsdatenbank verursacht wurden. Es ist Ihr eigenes Risiko, den Windows 95-Registrierungseditor (REGEDIT.EXE) oder ähnliche Werkzeuge zur Manipulation der Windows 95-Registrierung zu verwenden.“

Diese Warnung ist sicherlich gut gemeint und auch richtig, dennoch wirkt sie auf unerfahrene Anwender sehr abschreckend. Du als Leser dieses Heftes läßt Dich davon jedoch nicht einschüchtern. Um allen Unglücksfällen vorzubeugen, solltest Du den etwas später folgenden Abschnitt „Registry sichern“ unbedingt lesen.

Daß Microsoft keinerlei Dokumentation zur Registry mitliefert, unterstreicht das Vorhaben, die Endanwender möglichst von der Systemdatenbank fernzuhalten. Doch „keinerlei Dokumentation“ ist eigentlich nicht ganz richtig, wie ich Dir im folgenden Tip zeige:

Versteckte Informationen

Lediglich im Microsoft Resource Kit, das auf Deiner Windows-95-CD im Ordner `\ADMIN\RESKIT\HELPPFILE` zu finden ist, gibt es eine – wenn auch englischsprachige – Dokumentation über die Registry. Die beiden Dateien `WIN95RK.HLP` und `WIN95RK.CNT` kopierst Du Dir am besten in das Verzeichnis `C:\WINDOWS\HELP`.

Wenn Du diese englische Hilfedatei mit in das Gesamthilfesystem von Windows 95 einbinden willst, mußt Du die Datei `WINDOWS.CNT` um die Zeilen

:Index Resource Kit =win95rk.hlp und

:include win95rk.cnt

erweitern. Erstere Zeile gehört hinter den letzten Eintrag, der ebenfalls mit „:Index“ beginnt, die zweite Zeile kannst Du an das Ende der Datei schreiben.

Unter Windows 98 installierst Du das Resource Kit, indem Du die Datei `SETUP.EXE` aus dem Verzeichnis `\TOOLS\RESKIT` Deiner Windows-98-CD aufrufst. Die Hilfedatei findest Du dann unter **Start, Programme, Windows 98 Resource Kit, Resource Kit Online Book**.

Die Geschichte der Registry

Windows mußte sich nicht immer um so viele Konfigurationsdetails kümmern wie heute. Unter Windows 3.1 zum Beispiel hatte man grob gerechnet ein Drittel der Einstellungen von Windows 95 zu verwalten, doch Windows war damals auch noch nicht so leistungsfähig wie heute. Mit zunehmendem Komfort der grafischen Benutzeroberfläche steigt auch exponentiell der Verwaltungsaufwand. Der Internet Explorer 4, den Microsoft in Windows 95 eingebaut und samt einigen weiteren Neuerungen gleich als komplett neues Betriebssystem Windows 98 verkauft hat, bringt weitere Einstellungen mit sich, die sich Windows merken muß.

INI-Dateien – Was war das gleich noch mal?

Unter Windows 3.1 hat man fast alle Einstellungen in Textdateien gespeichert. Bekannt wurden diese unter dem Namen INI-Dateien, weil sie die Endung `INI` trugen. Die beiden bekanntesten unter diesen waren die `WIN.INI` und die `SYSTEM.INI`. Beide zusammen beinhalteten fast alle Einstellungen von Windows, so zum Beispiel die Anzahl der eingestellten Farben, die Monitorauflösung, das Hintergrundbild und die installierten Schriftarten.

Zusätzlich besaß fast jedes Programm auch noch eine eigene `INI`-Datei, um seine eigenen Einstellungen zu speichern. Je nach Anwendung hießen diese dann beispielsweise `WINWORD.INI` oder auch `MPLAYER.INI`.

Der Aufbau von INI-Dateien

Die INIs waren recht einfach aufgebaut: Die übergeordneten Gruppen wurden in eckigen Klammern quasi als Überschrift eines Textblockes geschrieben. Da gab es Gruppen wie [Windows], [Fonts] oder auch [Microsoft Word]. Der Rest bestand aus sogenannten Schlüssel-Werte-Paaren. In jeder Zeile stand ein Schlüssel (ein Wort als Name für eine Funktion von Windows) und ein Gleichheitszeichen gefolgt vom zugewiesenen Wert. Die Form von solchen Zeilen sah meist wie folgt aus: „AnzahlFarben=256“. Mit dieser Beispielzeile merkt sich Windows, daß es mit 256 Farben arbeiten soll.

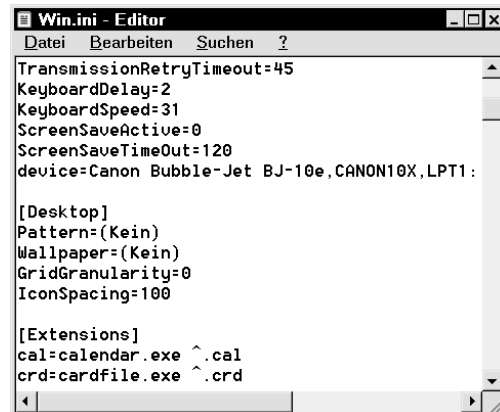


Abbildung 1: Eine klassische WIN.INI von Windows 3.1

INI-Dateien editieren

Wie Du im Verlauf des Heftes erfahren wirst, finden die meisten Konfigurationseinstellungen heute in der Registry statt. Wenn Du jedoch trotzdem einmal eine der beiden entscheidenden INI-Dateien von Windows (WIN.INI und SYSTEM.INI) bearbeiten muß, solltest Du den Systemeditor von Windows kennen. Dieses kleine Tool startest Du mit **Start, Ausführen, Sysedit**.

Das Programm startet und lädt neben weiteren recht wichtigen Dateien die WIN.INI und SYSTEM.INI gleich automatisch mit.

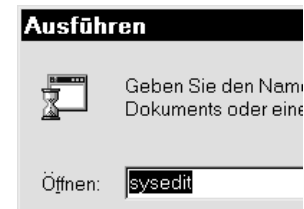


Abbildung 2: So startest Du den praktischen Systemeditor

Das Beste: Sobald Du Änderungen an einer der Dateien vornimmst und diese speicherst, erstellt Sysedit automatisch eine Sicherungskopie der Datei mit der Endung SYD. Wenn Du also die SYSTEM.INI editierst und die neue Version gespeichert hast, findest Du im WINDOWS-Verzeichnis die alte Version unter dem Namen SYSTEM.SYD.

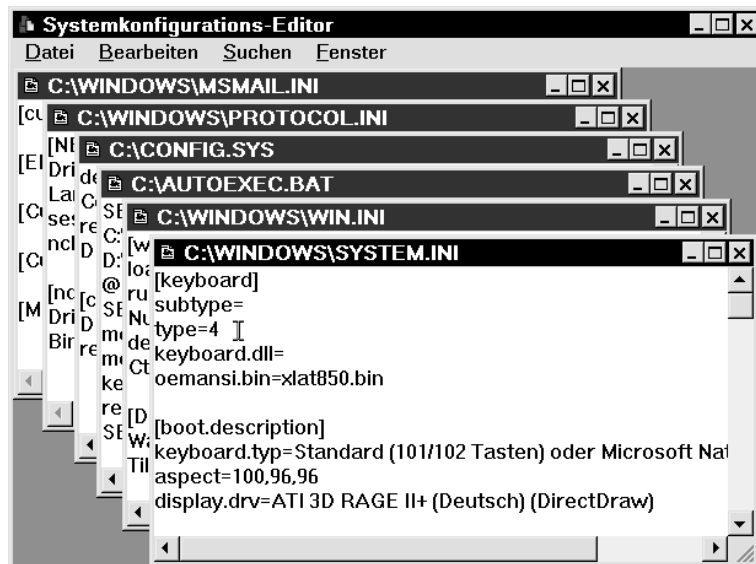


Abbildung 3: Der Systemeditor lädt viele wichtige Systemdateien

Erste Anfänge der Registry

Neben den **INI**-Dateien gab es unter Windows 3.1 jedoch auch schon eine Urform der heutigen Registry, eine Datei namens **REG.DAT**. Diese Datei wurde bei der Verwaltung von Software verwendet, um beispielsweise Informationen für den Datenaustausch vom Programm A zum Programm B zu speichern. Alle anderen Einstellungen befanden sich jedoch wie gesagt in den beiden Dateien **WIN.INI** und **SYSTEM.INI**.

Das Prinzip der **REG.DAT** wurde unter Windows 95 ausgebaut. Ab sofort sollten alle Einstellungen über die Registry verwaltet werden. Doch mitnichten! Damit alte Programme, die man für Windows 3.1 geschrieben hat, auch unter Windows 95/98

laufen, mußte man die beiden Dateien **WIN.INI** und **SYSTEM.INI** in das neue System mitnehmen. Installierst Du ein solches Programm, schreibt es seine Konfigurationseinstellungen nicht etwa in die Registry. Nein, wie zu alten Zeiten landen die meisten Einträge wieder in der **WIN.INI**. Schlimmer noch: Auch die eigenen **INI**-Dateien lassen sich die älteren Programme nicht nehmen.

Leider halten sich auch die Programmierer moderner 32-Bit-Software (Software für Windows 95/98/NT) nicht immer an Microsofts Empfehlung, alle Einstellungen an einer bestimmten Stelle der Registry abzulegen.

Du siehst: An der Registry spalten sich Theorie auf den Blättern Microsofts und die Praxis der weltweiten Computerei, und so kommt es, daß Du beim Durchforsten Deiner Festplatte noch immer auf eine Vielzahl von **INIs** triffst.

Unnötige INI-Dateien suchen und löschen

Probiere die Suche einfach einmal aus, indem Du über **Start, Suchen, Dateien/Ordner** den **Suchen**-Dialog aufrufst. Unter **Name** gibst Du das Suchkriterium ***.ini** ein und beginnst die Suche mit einem Klick auf die Schaltfläche **Starten**.

Auf meiner Festplatte befanden sich sagenhafte 449 dieser **INI**-Dateien; tragen sie den Namen eines längst gelöschten Programmes, kannst Du die **INI**-Datei entfernen. Bist Du Dir nicht sicher, solltest Du die Datei behalten, denn aufgrund ihrer geringen Größe ist sie sicher nicht der Übeltäter, der Deine Festplatte mal wieder zu klein werden läßt. Notfalls kannst Du die Datei auch auf einer Diskette sichern; falls ein Programm eine Fehlermeldung ausgibt, kopierst Du sie einfach zurück.

Dateiname	Größe	Typ
Systemmon.exe	65 KB	Anwendung
System.cb	1 KB	Datei CB
System.da0	2.895 KB	Datei DA0
System.dat	2.895 KB	Datei DAT
System.ini	3 KB	Konfigurati...
system.nu	2.561 KB	Datei NU
System.sysd	3 KB	Datei SYD
Systempr.grp	7 KB	Microsoft...
Taskman.exe	29 KB	Anwendung
Telephon.ini	6 KB	Konfigurati...
Telnet.exe	66 KB	Anwendung
Telnet.hlp	23 KB	Hilfdatei
Time.txt	28 KB	Textdatei

Abbildung 4: Bei Änderungen an der **SYSTEM.INI** hat Sysedit eine Sicherungskopie erstellt

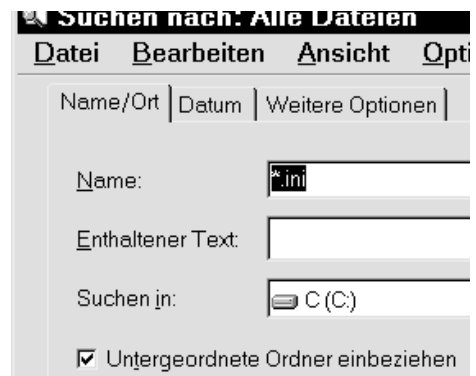


Abbildung 5: Schneller als die Polizei erlaubt – INI-Dateien aufgespürt

Vorteile des Registry-Prinzips

Daß **INI**-Dateien einen sehr einfachen Aufbau haben, weißt Du schon. Mit dieser einfachen Struktur, die letztendlich nur zwei Ebenen hat, wird die Verwaltung von sehr vielen und komplexen Informationen schwierig. Hinzu kam unter Windows 3.1 die Beschränkung, daß **INI**-Dateien nicht größer als 64 KB werden durften, was zumindest im Fall der **WIN.INI** teilweise Probleme bereitete.

Auch konnte man unter Windows 3.1 nur auf sehr kompliziertem Wege verschiedene Benutzerprofile anlegen. Wollte man wirklich unterschiedliche Konfigurationen nebeneinander betreiben, mußten vor dem Starten von Windows immer (zeit)aufwendig die **INI**-Dateien mit den verschiedenen Einstellungen hin- und herkopiert werden. So manch ein Anwender wird sich sicherlich mit Schauern an Dateien wie **WIN.INI**, **WIN.002**, **WIN.003** oder ähnlich erinnern. Man brauchte also für jede Konfiguration eine separate Datei.

Windows 95/98 mit seinem System der Registry kennt diese Probleme nicht. Die Registrierungsdatenbank ist gut durchdacht und klar strukturiert. Dank dem hierarchischen Aufbau ist das System beliebig in die Tiefe ausbaubar. Ähnlich der Organisationsstruktur von Festplatten mit Ordnern und Unterordnern lassen sich nämlich immer weiter Verschachtelungen erstellen. Auch werden in der Registry die Einstellungen nicht mehr nur in reiner Textform festgehalten. Teilweise verwendet man auch sogenannte Binärwerte (die Kolonnen aus Einsen und Nullen), die schneller verarbeitet werden können. Zum Aufbau der Registry kommen wir jedoch etwas später.

Nichtsdestotrotz hat so viel Komplexität und Erweiterbarkeit auch ihren Preis. Anfänger finden sich in der Registry zu Beginn schwerer zurecht als in den einfachen textbasierten **INI**-Dateien. Auch wird der Aufwand für die Programmierer von Anwendungssoftware größer, weshalb – wie schon erwähnt – 32-Bit-Programmierer teilweise immer noch **INI**-Dateien verwenden.

Der Aufbau der Registry

Äußerlich besteht die Registry aus den beiden Dateien **USER.DAT** und **SYSTEM.DAT**, die damit wohl die beiden wichtigsten Systemdateien unter Windows darstellen. Das Fehlen einer Datei würde das gesamte System lahmlegen.

Der Unterschied zwischen System.dat und User.dat

Berechtigterweise wirst Du jetzt vielleicht fragen: „Was soll das? Warum hat Microsoft die Registry auf zwei Dateien aufgeteilt?“ Dies hat folgenden Grund: Windows unterstützt die Verwendung von Benutzerprofilen. Das bedeutet, daß mehrere Benutzer an einem Rechner arbeiten können. Sie melden sich beim Starten mit ihrem Namen und einem Paßwort an und richten dann den Computer nach ihren Wünschen und Vorlieben (Hintergrundbild etc.) ein. Windows erkennt beim Anmelden über den Namen, wer gerade am Rechner sitzt und stellt die entsprechenden Einstellungen bereit.

Die Hardwareeinstellungen und die Registrierung der installierten Software bleiben bei verschiedenen Profilen gleich; deshalb kommen diese Einträge in eine Datei, die **SYSTEM.DAT**. Andere Merkmale, wie beispielsweise das aktuelle Hintergrundbild, variieren von Benutzer zu Benutzer bzw. von Profil zu Profil. Diese variablen Einstellungen werden in einer zweiten Datei gespeichert, der **USER.DAT**.

Die **SYSTEM.DAT** bleibt für alle Benutzer bzw. Profile gleich, denn sie arbeiten ja mit dem gleichen Computer. Jeder Benutzer hat dagegen eine eigene **USER.DAT** mit seinen benutzer-spezifischen Einstellungen. Diese findest Du im Ordner

C:\WINDOWS\PROFILES\BENUTZERNAME.

Den inneren Aufbau der Registry erkennst Du, wenn Du über **Start, Ausführen, Regedit** den Registriereditor aufrufst. Es erscheint ein aufgeräumter, zweigeteilter Bildschirm, der die Möglichkeiten und Tiefen des Systems noch nicht errahnen läßt. Du siehst lediglich sechs Gruppen, die dem Eintrag „Arbeitsplatz“ untergeordnet sind.

(Homekeys) der Registry

Die sechs Hauptschlüssel sind weiter verschachtelt. Das kannst Du leicht erkennen, wenn Du einmal auf einen der sogenannten Homekeys (HKEY) doppelklickst. Es öffnen sich mehrere untergeordnete Einträge, die sich ihrerseits wieder weiter verzweigen.



Abbildung 6: die 6 Hauptgruppen

Man muß sich das Ganze wie die Verzeichnisstruktur auf einer Festplatte denken. Die Homekeys bilden die übergeordneten Verzeichnisse der Festplatte und sind weiter untergliedert. Wenn Du den aufgeklappten Registriereditor mit dem Explorer und seinem Verzeichnisbaum vergleichst, wirst Du auch die optische Ähnlichkeit feststellen.

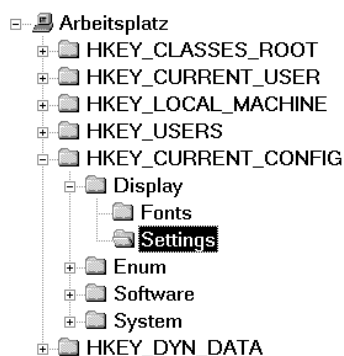


Abbildung 7: Du kannst die Gruppen wie Ordner im Explorer aufklappen

Die „Adresse“ eines Registry-Schlüssels

Wenn Du die Position einer Datei auf der Festplatte angibst, erscheint diese beispielsweise in der Form `C:\WINDOWS\HELP\MPLAYER.HLP`. Genauso kannst Du in der Registry die Position eines bestimmten Eintrages exakt angeben:

HKEY_CURRENT_USER\Control Panel\desktop\CursorBlinkRate.

Die den Homekeys untergeordneten Kategorien, insbesondere die letzten Einträge der Registrierbäume, nennt man „Schlüssel“; sie treten immer im linken Fenster auf.

Im rechten Fenster des Registriereditors findet man die abhängigen Einträge, die je nach Typ „Binärwert“, „Zeichenfolge“ oder „DWord-Wert“ heißen. Diese drei Einträge definieren eine Einstellung, der ihnen zugewiesene Wert beeinflusst das Ergebnis.

Allerdings gibt es kein offensichtliches Prinzip, wann Windows Einstellungen als „Zeichenfolge“ und wann als „Binär-“ oder „DWord-Wert“ speichert.

Die Zuordnung von Wertnamen und Wert ist ähnlich den INI-Dateien; dort habe ich das Beispiel „AnzahlFarben=256“ angeführt, wenn Du Dich erinnerst.

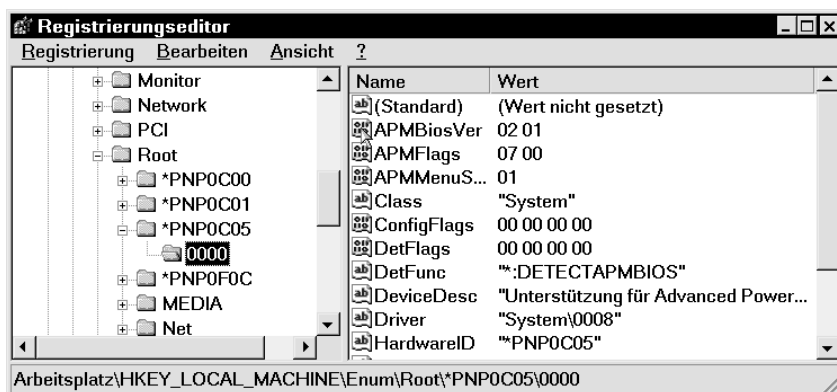


Abbildung 8: Zeichenfolgen und Binär- / Dword-Einträge erkennt man am unterschiedlichen Icon davor

In der Registry findest Du beispielsweise die Zeichenfolge „CursorBlinkRate“, welche die Blinkfrequenz des Cursors definiert. Ganz rechts, also neben der Zeichenfolge „CursorBlinkRate“, findest Du den zugewiesenen Wert, beispielsweise 500. In Gedanken kannst Du Dir auch ein Gleichheitszeichen wie in den **INI**-Dateien denken. In dieser Schreibweise würde das dann wie folgt lauten: „CursorBlinkRate=500“. Auch wenn im Registriereditor dieses Gleichheitszeichen nicht zu sehen ist, kannst Du Dir so die untrennbare Verbindung zwischen Zeichenfolge und Wert besser vorstellen.

Aus der Beziehung Zeichenfolge / Wert folgt logischerweise, daß man mit einem Schlüssel und einer Zeichenfolge (oder einem DWord- bzw. Binärwert) allein noch keine Einstellung treffen kann. Entscheidend ist der zugewiesene Wert, der immer ganz rechts im Registriereditor steht. ⚠

Übersicht

In der Help-Datei des Microsoft Resource Kit, die ich anfangs erwähnte, findest Du eine Grafik, die den Aufbau und den Unterschied zwischen den beiden Dateien optisch verdeutlicht. Der rechte Strang mit den Homekeys **HKEY_CURRENT_USER**\ und **HKEY_USERS**\ stellt die Einträge der **USER.DAT** dar; der linke Strang beinhaltet die für alle Benutzer gleichen Einstellungen der **SYSTEM.DAT**.

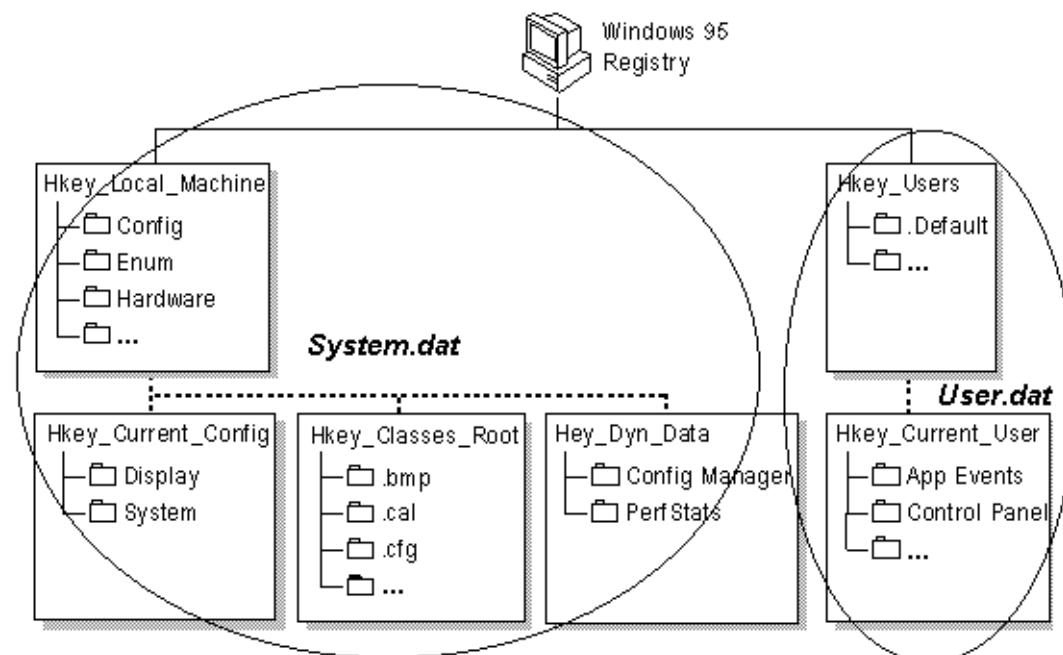


Abbildung 9: Das MS Resource Kit zeigt den groben Aufbau der Registry

Sicherlich kann man im eng gesteckten Rahmen dieses Heftes kaum eine komplette Referenz über alle Schlüssel der Registry erstellen. Wer einmal nur aus Interesse durch die verschiedenen Zweige der Registrierdatenbank „surft“, wird sicherlich von der Vielfalt der Einträge und Schlüssel erschlagen. Viele dieser Einträge bestehen aus für Normalanwender kaum verständlichen Zahlenkolonnen, die für Benutzeränderungen auch nicht vorgesehen sind. Andere Einträge und ganze Zweige sind hingegen sehr interessant, kann man doch dort mit ein paar Änderungen das Verhalten von Windows an persönliche Anforderungen und Wünsche anpassen. Damit Du überhaupt eine Ahnung hast, was die einzelnen Hauptschlüssel für Einstellungen speichern, gebe ich Dir im folgenden einige allgemeine Erklärungen.

HKEY_CLASSES_ROOT\

Die erste Gruppe bildet der Hauptschlüssel **HKEY_CLASSES_ROOT**. In dieser Gruppe verwaltet Windows alle Dateitypen. Hier wird beispielsweise die Einstellung gespeichert, daß der Dateityp **BMP** standardmäßig mit dem Windows-Programm „Paint“ geöffnet wird.

Da es eine Vielzahl unterschiedlicher Dateitypen gibt, fällt die Liste mit den Unterschlüsseln von **HKEY_CLASSES_ROOT** sehr lang aus; bei der Installation zusätzlicher Programme wird sie sogar noch länger. Der Vorteil der Typenregistrierung liegt darin, daß Du eine **BMP**-Datei im Explorer, Arbeitsplatz oder auf dem Desktop nur doppelt anzuklicken brauchst, damit sie mit einem Programm geöffnet wird, das die Datei lesen, anzeigen und bearbeiten kann. Würde es diese Funktion nicht geben, müßtest Du jedes Mal erst umständlich ein Anwendungsprogramm starten, um über **Datei öffnen** die Datei aufzurufen.

Dieses Dateiverständnis von Windows stützt sich auf einen langen Weg durch die Registrierungsdatenbank. Dazu ein kurzer aber aufschlußreicher Exkurs:

So funktioniert das Dateiverständnis von Windows 95

Wenn Du auf eine Datei mit der Endung **BMP** doppelklickst, sucht Windows unter **HKEY_CLASSES_ROOT\.bmp** nach einer Programmanweisung. Die ist an dieser Stelle jedoch noch nicht zu finden. Bitmaps (**BMP**-Dateien) werden hier nur als „Paint.Picture“ klassifiziert.

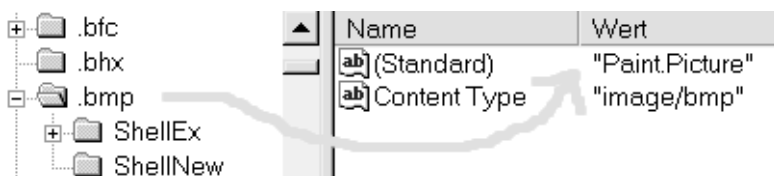


Abbildung 10: BMP gehört zur Gruppe "Paint.Picture"

Windows sucht also nach weiteren Informationen zu „Paint.Picture“ und findet einen gleichnamigen Schlüssel. Unter **HKEY_CLASSES_ROOT\Paint.Picture\shell\open\command** steht dann die Programmanweisung **C:\PROGRA~1\ZUBEHÖR\MSPAINTE.EXE %1**.

Der Umweg über den Schlüssel **\Paint.Picture** erscheint auf den ersten Blick sinnlos, hat jedoch folgenden Grund: Alle Dateitypen, die Paint verarbeiten kann, bekommen erst einmal den Stempel „Paint.Picture“ aufgedrückt. Erst diesem Stempel werden für Aktionen wie „Öffnen“ und „Drucken“ exakte Programmanweisungen zugeordnet. Jedes Bildverarbeitungsprogramm vom einfachen Paint abgesehen kann sehr viele verschiedene Dateitypen lesen (**BMPs**, **GIFs**, **TIFs**, **JPGs** oder auch **PCX**-Bilder). Für jeden Dateityp müßte der jedes Mal gleiche Programmpfad vermerkt werden. Die Stempelmethode speichert diesen nur ein Mal, was die Verwaltung vereinfacht und Speicher spart.

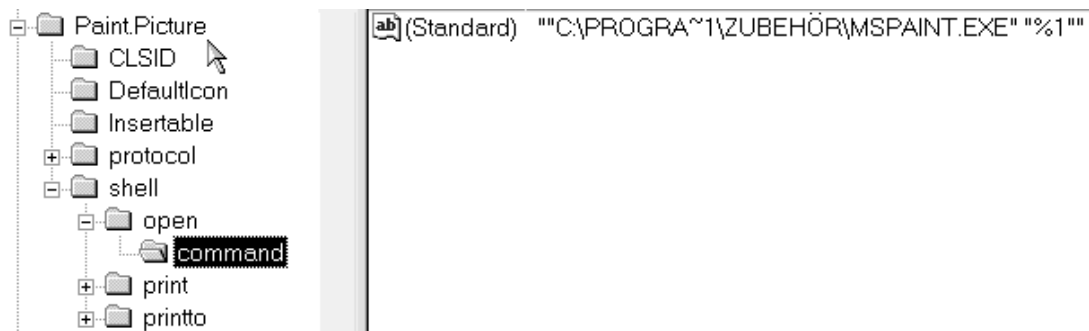


Abbildung 11: Befehle zum Öffnen und Drucken von "Paint.Picture"

Unter **HKEY_CLASSES_ROOT** werden weiterhin alle Einstellungen gespeichert, die den Datenaustausch per OLE (Object Linking & Embedding) betreffen. Nur mit diesen Einstellungen lassen sich beispielsweise Excel-Diagramme in Winword einbetten oder eine Grafik aus Paint in eine Access-Datenbank einfügen.

HKEY_USERS\

Die Gruppe **HKEY_USERS** verwaltet alle Benutzereinstellungen. Die Einstellungen jedes einzelnen Benutzers (sofern Du mehrere Benutzer eingerichtet hast) werden separat nach Profilen gespeichert. Diese Profile enthalten unter anderem die Einstellungen für das Hintergrundbild, die verwendeten Klang- und Farbschemata, Bildschirmschoner und so weiter. Hier findet man auch die benutzerspezifischen Einstellungen der installierten Software. Arbeitest Du nicht mit verschiedenen Benutzerprofilen, erscheint in diesem Hauptschlüssel lediglich der Unterschlüssel **\.default**. Dort werden dann alle Einstellungen gespeichert, die jedem neu angemeldeten User zugeordnet werden. Bei mehreren Anwendern findest Du hier eine Liste der einzelnen Profile mit den jeweils eigenen Einstellungen.

HKEY_CURRENT_USER\

Die Gruppe **HKEY_CURRENT_USER** ist eigentlich ein Teil der Gruppe **HKEY_USERS**, der die Einstellungen des aktuellen Benutzers bereithält, sozusagen eine Spiegelung. Die Zuteilung eines separaten Homekeys für den aktuellen Benutzer ist durchaus sinnvoll. Werden nämlich durch den aktuellen Anwender Einstellungen wie zum Beispiel das Hintergrundbild verändert, speichert Windows oder das entsprechende Programm dies erst unter **HKEY_CURRENT_USER**

Dann wird der Weg zum Profil des aktuellen Anwenders in **HKEY_USERS** gesucht, wo die neuen Einstellungen vermerkt werden. Damit wird sichergestellt, daß die Änderungen sich wirklich nur auf seinen Desktop und nicht auch auf den der anderen Profilanwender auswirken. Die Programme bekommen von den Profilen gar nichts mit; sie legen ihre Einstellungen immer unter **HKEY_CURRENT_USER** ab.

Hast Du keine unterschiedliche Benutzer und damit Benutzerprofile eingerichtet, ist der Inhalt von **HKEY_CURRENT_USER** identisch mit den Einträgen von **HKEY_USERS\.default**.

HKEY_LOCAL_MACHINE\

Die Hauptgruppe **HKEY_LOCAL_MACHINE** enthält alle Informationen über die installierte Hardware. Alle gespeicherten Einstellungen sind für jeden Benutzer gleich, sie werden also unabhängig von eventuell eingerichteten Benutzerprofilen verwaltet. Weiterhin enthält die Rubrik alle Softwareeinstellungen, die nicht von Benutzer zu Benutzer variieren können.

HKEY_LOCAL_MACHINE\SOFTWARE

Ein besonders wichtiger Unterschlüssel von **HKEY_LOCAL_MACHINE** ist der Schlüssel **\SOFTWARE**. Er enthält die Registrierungen aller Programme nach Hersteller sortiert. Von großem Interesse ist sicherlich die Kategorie **Windows\CurrentVersion**, denn diese regelt das Verhalten des Betriebssystems und enthält wichtige Einstellungen, wenn man Windows an persönliche Bedürfnisse anpassen möchte.

Eine eigene Bedeutung kommt dem Unterschlüssel

HKEY_LOCAL_MACHINE\SOFTWARE\Classes zu: Dieser ist die eigentliche Quelle für den gesamten Hauptschlüssel **HKEY_CLASSES_ROOT**. Daß man die Einstellungen gleich doppelt speichert, hat folgenden Grund: Windows-3.1-Anwendungen legten bestimmte Einstellungen in der damaligen Ur-Registry ab. Diese verfügte lediglich über den Hauptschlüssel **HKEY_CLASSES_ROOT**. Wenn Du alte Windows-Anwendungen installierst, schreiben diese ihre Registrierungen unter Windows 95/98 in genau diesen Zweig. Die Spiegelungstaktik hat also reine Kompatibilitätsgründe.

Microsoft empfiehlt Softwareherstellern, ihre Registrierungen für moderne Windows-Software immer nach dem Schema

HKEY_LOCAL_MACHINE\Software\Herstellername\Produktname\Version in die Registrierdatenbank einzutragen.

HKEY_CURRENT_CONFIG

Auch bei **HKEY_CURRENT_CONFIG** finden wir das Spiegelungsprinzip wieder. Hast Du keine weiteren Hardwareprofile eingerichtet, ist „CurrentConfig“ (also die aktuelle Konfiguration) identisch mit **HKEY_LOCAL_MACHINE\Config\0001**.



Abbildung 12: Der Ursprung...

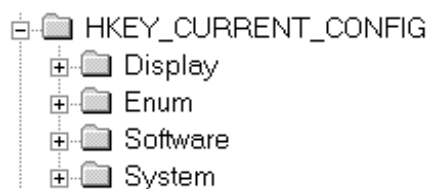


Abbildung 13: ... und der gespiegelte Teil

HKEY_DYN_DATA

Wie der Name schon verrät, verwaltet dieser Hauptschlüssel sogenannte „Dynamische Daten“. Das sind vorwiegend Daten über (Plug&Play-)Hardware, auf die so oft zugegriffen wird, daß Windows sie zur Laufzeit im Speicher hält. Im Unterschlüssel **\PerfStats** werden Daten für den Systemmonitor von Windows gespeichert, ein Tool, mit dem Du beispielsweise die aktuelle Prozessorauslastung im Auge behalten kannst. (Falls Du den Systemmonitor nicht installiert hast, läßt sich dies über **Start, Einstellungen, Systemsteuerung, Software, Windows-Setup, Zubehör, Details** nachholen. Du kannst ihn dann über **Start, Ausführen, Sysmon** starten.)

Rund um den Registriereditor

Wie ich bisher festgestellt habe, kommst Du also um die Registry nicht herum, wenn Du Windows an Deine persönlichen Erfordernisse und Vorlieben anpassen willst.

Bekanntlich führen viele Wege nach Rom, und so ist das auch mit der Registry. Der erste Weg: Rufe aus dem **Startmenü** den Befehl **Ausführen** auf, und gib den Befehl **Regedit** ein.

Die zweite Variante: Da Du den Registriereditor bei der Arbeit mit diesem Heft sicherlich noch öfter benötigen wirst, bietet es sich an, eine Verknüpfung in das Startmenü aufzunehmen. Dafür ziehst Du im Explorer die Datei **REGEDIT.EXE** bei gedrückter rechter Maustaste aus dem Ordner **\WINDOWS** auf den Startknopf und läßt sie los.

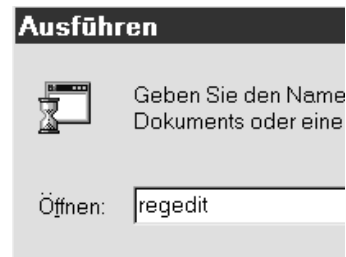


Abbildung 14: So rufst Du den Registriereditor auf

Suchen in der Registry

Im Registriereditor kannst Du über **Bearbeiten, Suchen** oder **Strg+F** (für „find“) nach Einträgen fahnden. Gib im erscheinenden Dialog den zu suchenden Begriff ein.

In manchen Fällen ist es sinnvoll, die Suche ein wenig einzugrenzen. Dazu stehen unter „Suchoptionen“ verschiedene Möglichkeiten zur Verfügung. Du kannst beispielsweise die Suche auf Schlüssel beschränken, wenn Du nach einem bestimmten Schlüssel fahndest. Wenn Du Dich erinnerst, eine bestimmte Einstellung irgendwo bei einem Schlüssel namens **\Microsoft** gefunden zu haben, gibst Du den Suchbegriff ein und aktivierst die Option „Schlüssel“ und „Ganze Zeichenfolge vergleichen“. Vor „Daten“ und „Werte“ darf kein Häkchen stehen. Nun sind die Chancen recht gut, daß Du das Gesuchte findest. Wären „Werte“ und „Daten“ auch aktiv und hättest Du die Suche nicht mit „Ganze Zeichenfolge vergleichen“ eingeschränkt, könntest Du Dich nun sehr lange durch hunderte Fundstellen arbeiten.

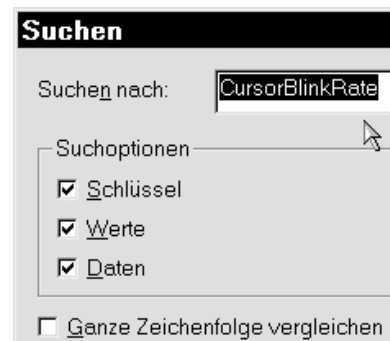


Abbildung 15: Ohne die Suchen-Funktion wärst Du in der Registry verloren

Die beiden anderen Optionen sind in der deutschen Version des Registriereditors vertauscht; wenn Du Werte suchst, also die Einträge ganz rechts im Editor, mußt Du im **Suchen**-Dialog „Daten“ aktivieren. Setzt Du ein Häkchen vor „Werte“, findet Windows nur Treffer im mittleren Teil des Registriereditors, wo die Namen von Zeichenfolgen, Binär- und DWord-Werten stehen.

Du solltest übrigens nicht verzweifeln, wenn die Suchen sehr lange dauern. Ich verstehe auch nicht, warum Microsoft hier nicht ähnlich den Installationsprogrammen eine Fortschrittsanzeige eingebaut hat. Auf jeden Fall ist Dein Rechner nicht abgestürzt; die Suche dauert aufgrund der vielen Einträge selbst auf sehr schnellen CPUs mitunter ein oder zwei Minuten.

Tip: Falls Du einmal einen kleinen Eindruck von der Vielzahl der Schlüssel bekommen möchtest, drückst Du im Registriereditor die Multiplikationstaste auf dem numerischen Zahlenblock. Nun werden alle Ebenen aufgeblättert, was wiederum auch einige Minuten dauern kann.

Wenn Du den gewünschten Suchbegriff eingegeben hast, kannst Du die Suche mit einem Klick auf die Schaltfläche „Weitersuchen“ starten. Damit schließt Du das Dialogfenster, und die Suche beginnt. Findet Windows einen Schlüssel oder Wert, der mit dem von Dir angegebenen Suchbegriff übereinstimmt, stoppt es die Suche und markiert die Fundstelle. Je nach Suchbegriff kann es teilweise mehrere Stellen geben, an denen das gesuchte Wort vorkommt. Du kannst Windows dann weiter fahnden lassen, indem Du **F3** drückst.

Platz da für die Schlüssel!

Je tiefer Du in die Registry vordringst, um so mehr Unterschlüssel wirst Du finden. Dabei reicht der Platz auf der linken Hälfte des Registriereditors meist nicht mehr aus, um den Registrierbaum komplett darzustellen. Du kannst die Breite der beiden Seiten jedoch ändern. Klicke dazu die senkrechte Leiste an, die das Editorfenster teilt. Bei gedrückter Maustaste kannst Du diese dann beliebig verschieben. Eine andere, wenn auch aufwendigere Methode besteht darin, **Ansicht, Teilen** zu verwenden.

Natürlich kannst Du auch die Bildlaufleisten oder die Tasten **Strg+Linkspfeil** oder **Strg+Rechtspfeil** verwenden, um verdeckte Schlüssel sichtbar zu machen.

Lesezeichen erstellen

Die Komplexität und Hierarchie der Registry macht lange Schlüsselzweige notwendig, die man sich nie merken kann. Deshalb habe ich einen Tip für Dich, wie Du eine Art Lesezeichen erstellst: Klicke den Schlüssel, bei dem Du ein Lesezeichen setzen möchtest, mit der rechten Maustaste an, und wähle **Neu, Zeichenfolge**.

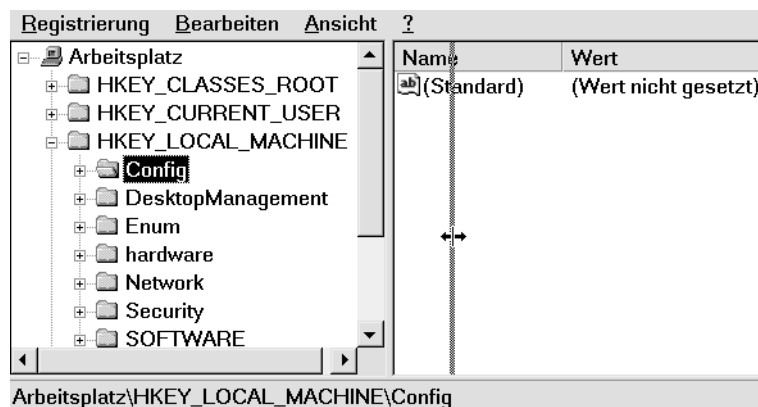


Abbildung 16: Die Fensteraufteilung ist variabel

Als Name nimmst Du ein Stichwort, das Du Dir leicht merken kannst und mit dem Du den entsprechenden Schlüssel in Verbindung bringst. Um Verwechslungen mit eventuell bestehenden, gleichnamigen Schlüsseln oder Zeichenfolgen zu verhindern, setzt Du vor und nach dem Wort jeweils ein Sternchen. Ein Beispiel: Wenn Du den Initstring Deines Modems öfter anpassen mußst (der Initstring ist eine Zeichenfolge, die alle wichtigen Modemeinstellungen festhält), Dir jedoch nicht den „Schlüssel-Wurm“ merken möchtest, klickst Du den Schlüssel

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

mit der rechten Maustaste an, wählst **Neu, Zeichenfolge** und vergibst den Namen ***modeminit***. Nun kannst Du jederzeit über die **Suchen**-Funktion an diese Stelle springen, indem Du als Suchbegriff ***modeminit*** eingibst.

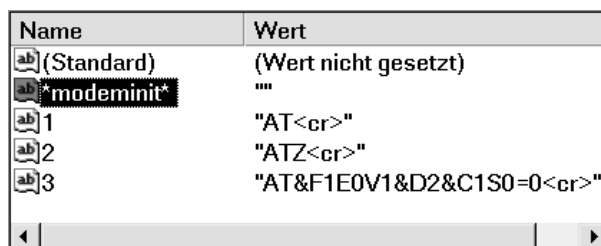


Abbildung 17: Mit diesem „Lesezeichen“ und der Suchen-Funktion findest Du zurück

Notizen erstellen

Ich gebe es zu: Die Registry ist ein ziemlicher „Datenschungel“, und man sucht manchmal ewig nach einem Eintrag. Außerdem hat man bei der Vielzahl der Einträge immer Probleme, sich zu merken, welcher Schlüssel oder Eintrag für was zuständig war. Also schreibt man sich fein säuberlich einige Sachen über die Registry auf viele kleine Zettel, die man auf dem wüsten Schreibtisch dann nie wieder findet (einige werden hier ihren Schreibtisch wiedererkennen.). Die Zettelwirtschaft muß nicht sein, denn Du kannst Notizen auch direkt in der Registry verfassen. Dazu klickst Du einen Schlüssel rechts an, wählst **Neu, Zeichenfolge** und schreibst als Name **Notiz:** und als Wert Deinen Text. Diese Arbeitsweise hat einen Vor- und einen Nachteil: Gut ist, daß Du wie bei den Lesezeichen jetzt nach Stichwörtern suchen kannst; schade allerdings, daß Deine Notizen nach einer Windows-Neuinstallation wieder verschwunden sind. Allerdings findet man die Zettel ja nach 4 Monaten auch nicht mehr...

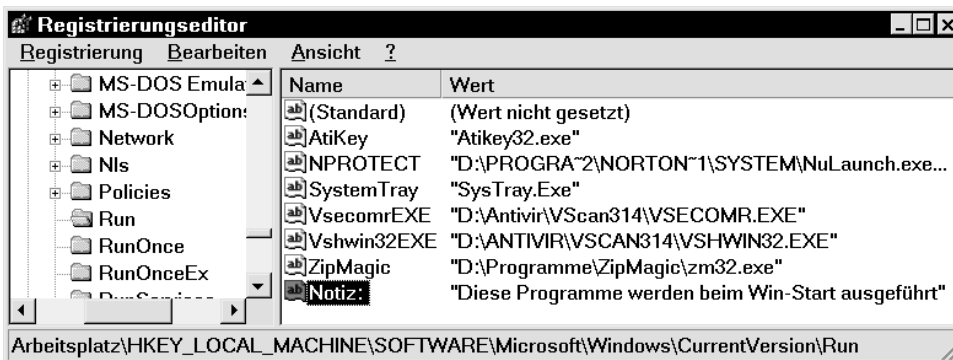


Abbildung 18: Verständliche Notizen im Schlüssel-Salat

Shortcuts für schnelleres Arbeiten

Wer viele Zweige der Registry ausgefahren hat, beispielsweise nach einer ausführlichen Suche, muß ziemlich lange per Bildlaufleiste scrollen, um wieder an den Anfang zu gelangen. Das muß jedoch nicht sein: Für das sofortige Springen an den Beginn gibt es den Shortcut **Strg+Pos1**, während man mit **Strg+Ende** zum letzten Schlüssel gelangt.

Wenn Du mehrere Einträge auf einmal löschen willst, mußt Du nicht jeden einzelnen anklicken und separat löschen: Halte **Strg** gedrückt, und markiere nacheinander alle gewünschten Einträge. Mit **Entf** löschst Du alle in einem Rutsch. Übrigens: Wenn die Einträge alle aufeinanderfolgen, reicht es, bei gedrücktem **Shift** den ersten und letzten anzuklicken. Die Einträge dazwischen werden automatisch markiert.

Wenn Du Dich gerade wieder in den Tiefen der Registry „verirrt“ hast und wieder etwas Ordnung in die geöffneten Schlüssel bringen willst, mußt Du oft lange nach oben scrollen, bis der übergeordnete Schlüssel sichtbar ist, den Du per Doppelklick wieder „einfährst“. Das Hochscrollen und Suchen des übergeordneten Schlüssels kannst Du Dir jedoch sparen: Ein Doppelklick auf die jeweilige senkrechte Hierarchielinie schließt den entsprechenden Zweig und verhilft zu mehr Übersicht.

Nützlich ist auch die linke Pfeiltaste: Beim ersten Druck springt die Markierung zum übergeordneten Schlüssel; ein weiterer Druck fährt die Unterordner ein und schafft Übersicht. Wenn Du einfach nur zum übergeordneten Schlüssel springen willst, ohne die Unterschlüssel zu schließen, verwendest Du die **Backspace**- oder **Rücklösch**-Taste.

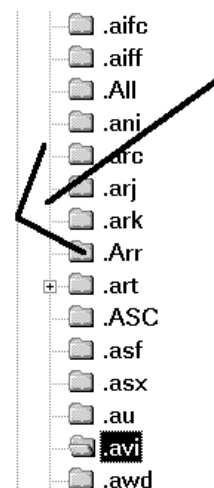


Abbildung 19: Ein Doppelklick fährt die Unterschlüssel ein

REG-Dateien

Wenn Du die Registry oder auch nur Teile wie beispielsweise einen einzelnen Zweig exportierst, erzeugst Du eine Datei mit der Endung **REG**. Eine solche Datei ist eine reine Textdatei, die Du in einen Editor oder auch Winword laden kannst, um sie genauer unter die Lupe zu nehmen.

Tips zum Umgang mit REG-Dateien



Klicke nicht wahllos oder spaßeshalber auf irgendwelche **REG**-Dateien. Dies könnte unangenehme Folgen haben.

Arbeitest Du beispielsweise schon lange mit einer Excel-Installation, hast Du diese bestimmt an Vorlieben angepaßt und nach Deinen Wünschen konfiguriert. Ein unbedachter Doppelklick auf die Datei **EXCEL7.REG** würde alle Standardeinstellungen in die Registry schreiben und damit zumindest Teile Deiner persönlichen Einstellungen und Einträge überschreiben.

Wenn Du eine **REG**-Datei im Explorer mit der rechten Maustaste anklickst, erscheint im Kontextmenü der Befehl **Zusammenführen**. Klicke diesen nicht an; das hätte den Effekt eines Doppelklicks auf die Datei, womit die Einstellungen in der Datei neu registriert werden. Willst Du die Datei im Editor begutachten, wählst Du **Bearbeiten**. Dieser Befehl ist standardmäßig mit dem Editor verknüpft. Sollte die Datei zu groß sein (Komplettextport der Registry), wird statt dessen Wordpad verwendet.



Abbildung 20: Mit **Bearbeiten** kannst Du die Datei editieren und anschauen

Aufbau von REG-Dateien

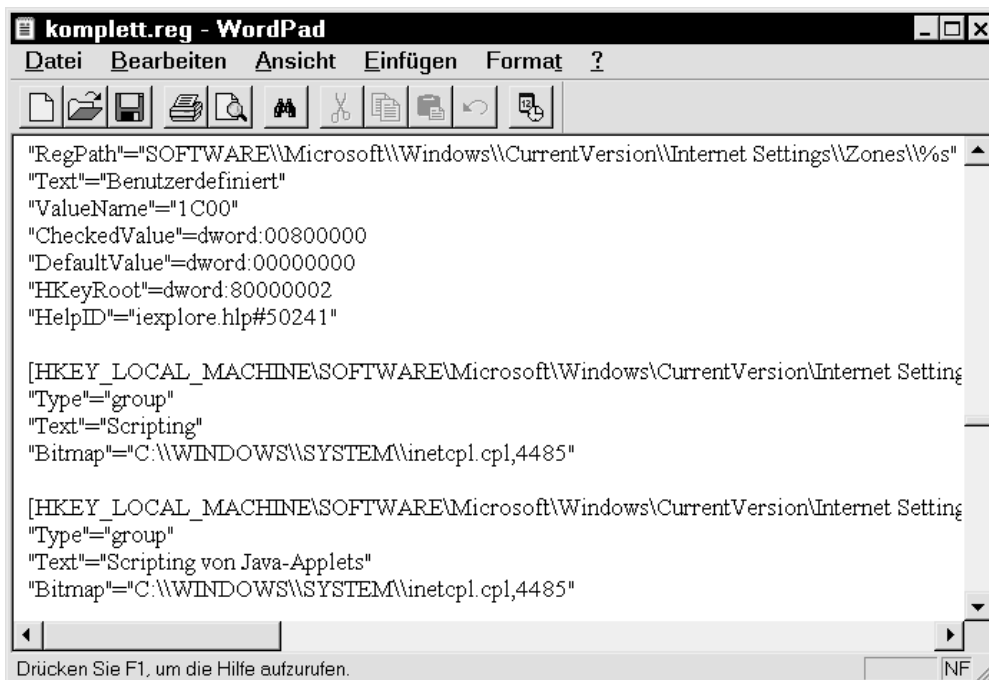


Abbildung 21: Die Registry in Textform als REG-Datei

REG-Dateien liegen immer im gleichen Schema vor – nur so kann garantiert werden, daß alle Einträge der REG-Dateien nach einem Doppelklick problemlos in die Registry geschrieben werden können. Eingeleitet werden sie mit dem Eintrag „REGEDIT4“. Der betroffene Schlüssel wird in eckige Klammern gesetzt. Dabei ist es egal, ob dieser Schlüssel schon existiert. Ist dies nicht der Fall, wird er angelegt. Gibt es den kompletten Zweig schon, wird direkt zu dem in Anführungszeichen gesetzten Eintrag und seinem Wert gesprungen.

Die Anführungszeichen werden jedoch nicht immer verwendet. Der Eintrag „Standard“ wird beispielsweise aufgrund des häufigen Vorkommens durch den kurzen Klammeraffen @ vertreten. Dabei wird auch auf die sonst für Einträge üblichen Anführungszeichen verzichtet. Eine clevere Idee, wird doch dadurch der Zeichen- und damit auch Speicherplatzaufwand insgesamt gesehen drastisch reduziert. Auch bei den Werten verzichtet man auf Anführungszeichen, wenn es ein hexadezimaler oder DWord-Wert ist. Folgende Beispiele sind also möglich:

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

REG-Dateien selber schreiben?

Kennst Du das Prinzip einer REG-Datei, kannst Du leicht auch selbst eine solche Registrierung schreiben. Ehe Du einem Bekannten zeitaufwendig erklärst, was er für eine bestimmte Einstellung alles machen muß, schreibst Du ihm lieber gleich eine solche Datei. Per Diskette versandt oder an eine Email angehängt kann der Empfänger ohne weiterführende Kenntnisse die Einstellung per Doppelklick in die Registry übernehmen. Hierzu ein Beispiel:

REGEDIT4

[HKEY_CURRENT_USER\Control Panel\desktop]

```
"CursorBlinkRate"="100"
```

Schreibe diese Datei im Editor, und gib ihr einen aussagekräftigen Namen, beispielsweise **CURSORBLINKRATE.REG**. Sobald Dein Bekannter die Datei empfangen hat, braucht er sie nur doppelt anzuklicken. Die Einstellung, die damit geändert wird, bezieht sich auf den Blinkrhythmus des Cursors. Der voreingestellte Wert beträgt „500“. Kleinere Zahlen bewirken ein schnelleres Blinken, größere lassen den Cursor einschläfernd wirken. Du siehst also: Man muß für einen Eingriff in die Registry nicht unbedingt den Registrireditor bemühen.

Die Möglichkeit, gewünschte Registrierungen in eine Textdatei zu schreiben, hat trotz des anscheinend höheren Aufwandes zwei unschlagbare Vorteile: Erstens ist diese Arbeitsweise sicherer, denn die Eintragung wird erst nach einem Doppelklick auf die fertige REG-Datei vorgenommen. Im Registrireditor arbeitest Du in Echtzeit, denn jede Aktion und Änderung wird sofort durchgeschrieben, d.h. gespeichert. Dies erklärt auch die Eigenheit von Regedit, beim Verlassen des Programms nicht danach zu fragen, ob Du die Änderungen speichern möchtest.

Der zweite Vorteil kommt gerade Anwendern zugute, die an mehreren Computern arbeiten oder für diese verantwortlich sind. Anstatt eine Einstellung an jedem Computer eines Netzwerkes ein jedes Mal direkt im Registrireditor zu ändern, schreibt man die Datei einmal und führt sie dann nur noch auf jedem Computer aus.

Registrierungsbestätigung abschalten

Wann immer Du auf eine **REG**-Datei doppelklickst, erscheint – vorausgesetzt alles läuft glatt und die Datei ist nicht fehlerhaft – eine Meldung über die erfolgreiche Registrierung, die Du mit „OK“ schließt.



Abbildung 22: Auf die Dauer nerven die Meldungen

Das ist zwar im Einzelfall eine Beruhigung, spätestens jedoch wenn Du die **REG**-Dateien von Programmen nach Problemen neu in die Registry schreiben lassen willst, nervt das ständige OK-Klicken. Ein klitzekleiner Eingriff vorher schaltet die „Meldepflicht“ ab, spart Zeit und schont Deine Nerven. Du mußt lediglich beim Schlüssel

HKEY_CLASSES_ROOT\regfile\shell\open\command

aus der Anweisung „regedit.exe %1“ den Wert „regedit.exe /s %1“ machen.

Registrierungen vergleichen

Eine scheinbar schwierige Aufgabe besteht darin, zwei verschiedene Registry-Versionen miteinander zu vergleichen. Vor diese Aufgabe könntest Du gestellt werden, wenn einige Schlüssel Probleme machen und Du durch den Vergleich mit einer lauffähigen Registry den Fehler einkreisen mußt. Oder Du willst konkret feststellen, welche Änderungen das Installationsprogramm einer Anwendung vorgenommen hat, indem Du eine gesicherte Ausgangsversion mit der aktuellen Registry nach der Installation vergleichst.

Diese Aufgabe kannst Du Dir mit Winword erleichtern. Dort gibt es eine Funktion **Versionsvergleich**, die Dir Änderungen schnell und zuverlässig anzeigt. Wie das geht, erkläre ich Dir am Beispiel des registrierten Anwendernamens von Windows: Klicke den Schlüssel **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion** an und wähle **Registrierung, Registrierungsdatei exportieren**. Speichere den Zweig unter dem Namen **OLDNAME.REG**. Nun änderst Du den Wert des Eintrags „RegisteredOwner“, indem Du irgendein Wort oder Namen einträgst. Da dies nur der Name ist, den Du bei der Installation von Windows angegeben hast, kann hier nichts passieren. Nun speicherst (exportierst) Du den bearbeiteten Schlüssel erneut, dieses Mal jedoch unter dem Namen **NEWNAME.REG**.

Rufe nun Winword auf, und wähle **Datei, Öffnen**. Stelle sicher, daß unter **Dateityp** der Eintrag **Alle Dateien** ausgewählt ist, denn sonst werden die Dateien mit der Endung **REG** nicht angezeigt. Lade die Datei **OLDNAME.REG** aus dem Verzeichnis, in dem Du die Datei abgelegt hast. Nun wählst Du **Extras, Änderungen verfolgen, Dokumente vergleichen** (in Word 7/95 wählst Du **Extras, Überarbeiten, Versionsvergleich**). Du mußt die zweite Datei **NEWNAME.REG** angeben, worauf Winword den Vergleich startet. Das kann mitunter eine Weile dauern (bei größeren Texten wie beispielsweise einem Registry-Komplettexport auch mal 10 Minuten); danach zeigt Dir Winword aber haargenau, an welcher Stelle eine Änderung vorgenommen wurde. Der alte Eintrag ist rot durchgestrichen, der neue steht direkt dahinter.

OLE, Drag & Drop und CLSID

Erschrecke jetzt bitte nicht wegen der vielen, Dir vielleicht unbekanntem Begriffe und Abkürzungen. Alle drei stehen im Hinblick auf die Registry in engem Zusammenhang. Fangen wir mit dem einfachsten an, dem sogenannten Drag & Drop.

Was zum Teufel ist Drag & Drop?

Unter dem englischen Begriff „Drag & Drop“ versteht man das „Ziehen und Fallenlassen“, das die Arbeit unter den grafischen Benutzeroberflächen wie Windows so komfortabel macht.

Dazu gehört als simpelstes Beispiel das Anklicken eines Programmicons, das Verschieben und Fallenlassen an einer anderen Stelle des Desktops. Manch einem Anwender ist diese Arbeitsweise mittlerweile schon so in Fleisch und Blut übergegangen, daß er sich keine Gedanken mehr über die Vorgänge macht, die beispielsweise beim Ziehen einer Datei im Explorer auf das Icon des Diskettenlaufwerkes ablaufen.

Dabei sind die Zeiten noch gar nicht so lange her, als man nur mit kryptischen Befehlszeilen Dateien oder gar ganze Verzeichnisse kopieren, verschieben oder löschen konnte. Das Löschen erledigen wir heute je nach persönlicher Vorliebe zum Beispiel durch das Ziehen einer Datei auf den Papierkorb. Man kann sogar ein Dokument auf das Winword-Programmicon ziehen, wodurch Winword startet und den entsprechenden Text lädt.



Abbildung 23: Einfach und schnell – Drag & Drop

Damit ein Betriebssystem so komfortabel ist und die für den Benutzer völlig klaren Aktionen auch richtig versteht und interpretiert, muß es eine Menge Informationen über Datenaustausch, OLE und die Funktionen und Eigenschaften der einzelnen Programme verwalten.

Was zum Teufel ist OLE?

Du wirst es gemerkt haben – ich wollte etwas über „Drag & Drop“ erklären und bin dabei glatt bei einem neuen Begriff, nämlich „OLE“ gelandet. „OLE“ ist die englische Abkürzung für „Object Linking and Embedding“, also das Einfügen oder Verknüpfen eines Objektes. Der Unterschied zwischen Einfügen und Verknüpfen soll an je einem Beispiel kurz erklärt sein:

Einfügen oder Verknüpfen – ein nützlicher Unterschied

- A) Markiere einen Text im Editor von Windows, und betätige die Tastenkombination **Strg+C**. Damit wird der markierte Text in die Zwischenablage kopiert. Wechsele dann zu Winword und rufe **Bearbeiten, Einfügen** auf.
- B) Kopiere aus Excel ein Diagramm in die Zwischenablage und wähle in Winword **Bearbeiten, Inhalte einfügen**. Es erscheint ein Dialogfenster, in dem Du die Wahl zwischen **Einfügen** und **Verknüpfen** hast. Ändere die Voreinstellung in **Verknüpfen** und klicke auf **OK**.

Du hast nun zum Diagramm in Excel eine Verknüpfung erstellt. Änderst Du die zugrundeliegenden Daten in Excel, so ändert sich erst das Diagramm in Excel. Beim nächsten Aufruf des Winword-Dokumentes wird auch das dortige Diagramm aktualisiert.

Du mußt Dir eine Verknüpfung wie eine ständige Spiegelung vorstellen, während das Einfügen dem einmaligen Schießen eines Fotos gleicht. Ein eingefügtes Objekt wird nicht aktualisiert, wenn sich die Daten in der Quellanwendung ändern.

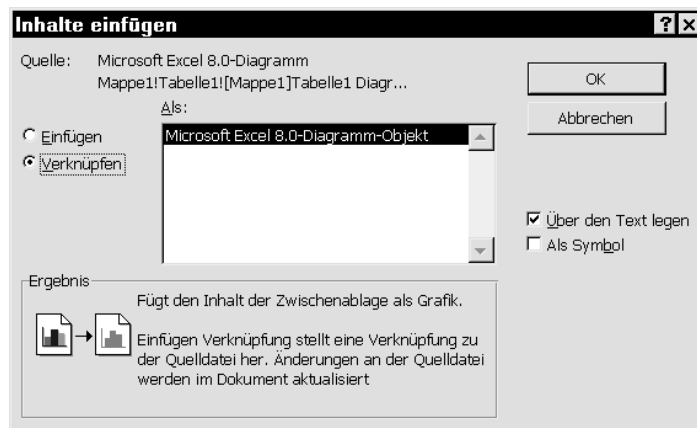


Abbildung 24: Verknüpfen bringt Aktualität

Was zum Teufel ist eine CLSID?

Sicherlich wirst Du Dich bei Deinem ersten intensiven Ausflug in die Registry erschrocken gefragt haben, was diese unglaublich langen Zeichenketten wie beispielsweise `CLSID{9E56BE60-C50F-11CF-9A2C-00A0C90A90CE}` zu bedeuten haben.

Dies ist ziemlich einfach, denn wir haben die dafür wichtigen Begriffe „OLE“ und „Drag & Drop“ bereits geklärt: Jedes OLE-fähige Programm wie beispielsweise Winword oder OLE-taugliche Objekt (zum Beispiel das anfangs angesprochene „Paint.Picture“) muß eindeutig identifiziert werden. Dazu könnte man sicherlich auch den Namen des Programmes verwenden, würde dabei jedoch das Risiko eingehen, daß es zwei Programme oder Objekte gleichen Namens gibt, was Windows ziemlich ins Schleudern bringen würde. Damit so etwas nicht passiert, hat man sich eine so lange Zeichenfolge ausgedacht, die jedes Objekt oder Programm genau identifiziert. CLSID heißt ausgeschrieben „ClassIdentifier“

So machst Du Dir das Suchen nach CLSIDs leichter

Um alle Einträge eines Programms oder Objekts zu finden, reicht die Suche nach dem Klartextnamen nicht immer aus. Du solltest also immer auch nach der CLSID des Programms/Objekts suchen. Dazu mußt Du diese natürlich vorher herausfinden. Du stößt fast immer darauf während der Suche mit dem Namen als Suchbegriff. Für Objekte vom Typ „Paint.Picture“ findest Du die CLSID unter `HKEY_CLASSES_ROOT\Paint.Picture\CLSID`.

Du brauchst bei der Suche mit einer bestimmten CLSID die Zeichenfolge im **Suchen**-Dialog nicht von Hand einzugeben. Die Gefahr eines Fehlers ist hier einfach viel zu groß. Um mit der CLSID von „Paint.Picture“ zu suchen, doppelklickst Du im rechten Bereich von `HKEY_CLASSES_ROOT\Paint.Picture\CLSID` auf den Eintrag „Standard“. Kopiere den markierten Wert mit der Tastenkombination **Strg+C** in die Zwischenablage. Dann wechselst Du zu **Bearbeiten, Suchen** und fügst im Feld **Suchen nach** mittels **Strg+V** die CLSID aus der Zwischenablage ein. Über dieses Kopieren und Einfügen sparst Du Dir die manuelle Eingabe.

Erste Hilfe für die Registry

Registry sichern

Im Gegensatz zum Systemeditor erzeugt der Registriereditor bei Änderungen nicht automatisch eine Sicherungskopie. Doch auch hier ist das Sichern nicht sehr aufwendig, da die Registrierdatenbank nur aus den zwei Dateien **USER.DAT** und **SYSTEM.DAT** besteht. Die Dateien befinden sich unter Windows 95 zusammen mit zwei Sicherungskopien **USER.DA0** und **SYSTEM.DA0** im Windows-Ordner. Die Sicherungskopien werden von Windows automatisch angelegt, damit bei Beschädigungen oder anderen Problemen die Registrierdatenbank problemlos restauriert werden kann. Windows 98 geht einen etwas anderen Weg: Das Programm „Scanreg“ legt beim ersten Start des Tages im Ordner **\WINDOWS\SYBCKUP** ein **CAB-Archiv** an, in dem **USER.DAT**, **SYSTEM.DAT**, **WIN.INI** und **SYSTEM.INI** in komprimierter Form gesichert werden.

Da die Dateien aus Sicherheitsgründen mit den Attributen „Versteckt“, „System“ und „Schreibgeschützt“ versehen sind, werden sie standardmäßig vom Explorer nicht angezeigt. Um dies zu ändern, mußt Du im Explorer **Ansicht, Optionen, Alle Dateien anzeigen** wählen. Hast Du den Internet Explorer 4 oder Windows 98 installiert, findest Du die Einstellung unter **Ansicht, Ordneroptionen, Ansicht, Erweiterte Einstellungen, Versteckte Dateien, Alle Dateien anzeigen**.

Dateiname	Größe	Typ
Sol.exe	168 KB	Anwendung
Stamp1.dat	1 KB	Datei DAT
Streifen.bmp	1 KB	Bitmap
Strohmatte.bmp	1 KB	Bitmap
Support.txt	9 KB	Textdatei
Sysmon.exe	65 KB	Anwendung
System.cb	1 KB	Datei CB
System.da0	2.895 KB	Datei DA0
System.dat	2.895 KB	Datei DAT
System.ini	2 KB	Konfigurat

Abbildung 25: Die Sicherungskopie der **SYSTEM.DAT** unter Windows

User.dat und System.dat kopieren

Da **USER.DAT** und **SYSTEM.DAT** alle wichtigen Einstellungen von Windows beinhalten, sollten sie regelmäßig auf eine Diskette kopiert werden. Der Gedanke, daß Windows 95 nach jedem Start selber Sicherungskopien mit der Endung **DA0** erstellt, sollte Dich nicht in falscher Sicherheit wiegen. Denn stellt sich der Fehler ganz konkret nach einem zweiten Neustart heraus, sind diese Dateien manchmal schon überschrieben. Unter Windows 98 passiert das nicht mehr; erstens wird das **CAB-Sicherungsarchiv** nur ein Mal am Tag angelegt, zweitens wird es nummeriert und für gewöhnlich erst nach 5 Tagen überschrieben.

Also regelmäßig (damit die Dateien bei Bedarf auch aktuell sind) die **USER.DAT** und **SYSTEM.DAT** auf eine Diskette sichern. Die **SYSTEM.DAT** wirst Du in den meisten Fällen zuvor mit einem Packprogramm wie Winzip komprimieren müssen, denn sonst paßt sie nicht auf eine normale Diskette. Es sollte außerdem klar sein, daß Du eine Sicherung von Windows 95 nicht zum Restaurieren von Windows 98 verwendest (andersherum natürlich auch nicht). Die Registry der beiden Windows-Version ist zwar weitestgehend identisch, aber die kleinen Unterschiede würden höchstwahrscheinlich dazu führen, daß Windows nicht mehr korrekt funktioniert.



So kopierst Du Dateien am schnellsten auf Diskette

Windows bietet standardmäßig eine besonders schnelle Option, um Dateien auf Diskette zu kopieren: Klicke die Datei mit der rechten Maustaste an und wähle aus dem Kontextmenü **Senden an, 3.5 Diskette (A)**.

Teilexport der Registry

Bevor Du irgendeinen Eintrag in der Registry änderst, empfiehlt es sich immer, diesen in der Originalform zu sichern. Dazu bietet der Registriereditor die Funktion **Registrierung, Exportieren**.

Bevor Du beispielsweise Änderungen am Schlüssel `HKEY_CURRENT_USER\Software\Microsoft` oder einem seiner Unterschlüssel vornimmst, klickst Du `\Microsoft` an und wählst **Registrierung, Registrierungsdatei exportieren**. Im erscheinenden Dialog ist unter „Exportbereich“ die Option **Gewählter Zweig** aktiviert und zeigt den gewählten Schlüssel an. Gib der Exportdatei einen aussagekräftigen Namen. Nun kannst Du beruhigt die Einstellungen des Schlüssels `\Microsoft` ändern. Sollte dabei etwas schief gehen, schreibst Du durch einen Doppelklick auf die exportierte **REG**-Datei die Einstellungen wieder in die Registry. Alternativ kannst Du natürlich auch den Befehl **Registrierungsdatei importieren** verwenden.

Der Vorteil der Exportieren-Methode ist, daß Du statt der ganzen Registry vor Änderungen ganz gezielt einen einzigen Schlüssel sichern kannst.

Im Zusammenhang mit dem Sichern der Registry möchte ich Dich auf das Kapitel „Tools rund um die Registry“ hinweisen. Dort findest Du einige Programme vorgestellt, die das regelmäßige Sichern der Registry erleichtern oder automatisieren.

Registrierungsabschnitte vor Eingriffen drucken

Selbstverständlich brauchst Du vor dem Ändern von Schlüsseln und Werten diese nicht mühsam und unübersichtlich auf einen kleinen Zettel schreiben; wozu gibt es Drucker?! Allerdings möchte ich Dich vorsorglich davor warnen, die gesamte Registry auszudrucken.

Mit an Sicherheit grenzender Wahrscheinlichkeit würdest Du ziemlich schnell in den nächsten Schreibwarenladen rennen und ein 500er-Pack Papier kaufen. Um einen bestimmten Teil vor dem Editieren zu Papier zu bringen, klickst Du den Schlüssel an und wählst **Registrierung, Drucken**. Unter „Gewählter Zweig“ zeigt Dir der Registriereditor an, welchen Teil (plus aller untergeordneten Zweige) er zum Druck schicken wird.

Tip: Wer lieber mit Shortcuts (Tastenkombinationen) arbeitet, kann statt **Registrierung, Drucken** auch **Strg+P** (für **Print**) drücken.

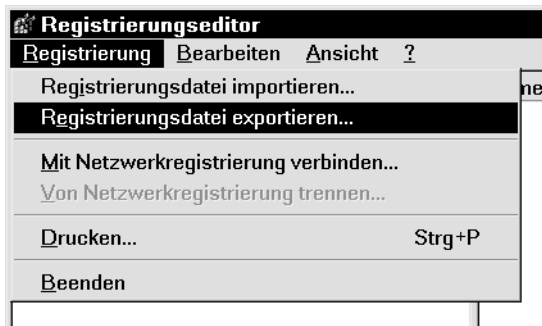


Abbildung 26: Die schnelle Sicherung von Schlüsseln

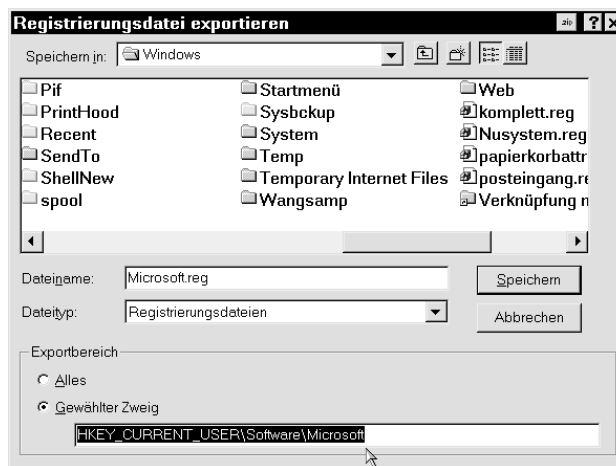


Abbildung 27: „Gewählter Zweig“ zeigt an, was alles gesichert wird.

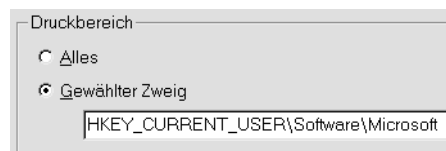


Abbildung 28: Der Ausdruck dauert nicht lange, ist aber bei Problemen eine große Hilfe

Startdiskette erstellen

Du hast doch sicherlich eine Startdiskette parat, oder etwa nicht?! Falls Du das Anfertigen einer solchen versäumt hast oder die Diskette durch das Hinzufügen neuer Hardware nicht mehr aktuell ist, solltest Du eine neue erstellen. Wechsele dazu zu **Start, Einstellungen, Systemsteuerung, Software, Startdiskette**. Ein Klick auf **Diskette erstellen**, und Deine formatierte und leere Diskette wird bootfähig.

Der Sinn: Die Sicherung der Dateien **USER.DAT** und **SYSTEM.DAT** ist zwar ungeheuer wichtig, wenn Du aber im Ernstfall keine Startdiskette hast, kannst Du den Rechner vielleicht nicht mehr booten. Einzelheiten dazu entnimmst Du bitte dem Kapitel „Troubleshooting – Nichts geht mehr“.



Abbildung 29: Eine Startdiskette ist immens wichtig!!

Aber Achtung: Unter Windows 95 befindet sich auf der Startdiskette standardmäßig kein Treiber für ein CD-ROM-Laufwerk. Das bedeutet, daß Du nach dem Booten von Diskette keinen Zugriff auf das CD-ROM-Laufwerk hast und deshalb auch nicht die Win95-CD ansprechen kannst. Hier es nötig, einen solchen manuell zu installieren. Um ein CD-ROM-Laufwerk auf der DOS-Ebene zum Laufen zu bringen, benötigst Du grundsätzlich zwei Dateien.

In der Datei **A:\CONFIG.SYS** wird der eigentliche DOS-Treiber des Herstellers oder ein Standardtreiber geladen. Dies kann zum Beispiel so aussehen:

DEVICEHIGH=A:\IDECDROM.SYS /D:MSCDOAK (alles eine Zeile)

In diesem Beispiel wird der auf der Diskette befindliche Treiber **IDECDROM.SYS** geladen und dem CD-ROM der Name „MSCDOAK“ zugewiesen.

Als zweite notwendige Datei muß aus der **A:\AUTOEXEC.BAT** die Datei **MSCDEX.EXE** geladen werden, ebenfalls von der Diskette. Falls sich die Datei noch nicht auf der Diskette befindet, kopierst Du sie aus dem Ordner **C:\WINDOWS\COMMAND**. Der Aufruf sieht wie folgt oder ähnlich aus:

LH A:\MSCDEX.EXE /D:MSCDOAK /M:10 /E (alles eine Zeile)

Der Name des CD-ROMs muß gleich dem in der **CONFIG.SYS** definierten angegeben werden, also für das hier gewählte CD-ROM „MSCDOAK“.

Es sei noch einmal zusammengefaßt: Für maximale Sicherheit solltest Du eine Startdiskette und einen Satz Sicherungskopien von **USER.DAT** und **SYSTEM.DAT** besitzen. Das Exportieren eines speziellen Schlüssels oder zumindest die Notiz auf dem Zettel vor Änderungen fördern eine schnelle Problemlösung.

Troubleshooting – Nichts geht mehr

Hoffentlich passiert es Dir nie, daß Dein Computer aufgrund einer kaputten oder gelöschten Registry die Hufe hochzieht. Sollte das wirklich einmal passieren, ist das mit einer erstellten Startdiskette und einem Sicherungssatz von **USER.DAT** und **SYSTEM.DAT** jedoch nur halb so schlimm.

Einzelfehler nach dem Start

Wenn Du einen einzelnen Eintrag geändert hast, der beispielsweise das Verhalten von Winword beeinflussen sollte, nun aber nur noch zu Problemen führt, gehst Du wie folgt vor: Wie ich Dir im Abschnitt „Registry sichern“ empfohlen habe, hast Du sicherlich den Zweig mit den betroffenen Schlüsseln exportiert. Dazu mußt Du einen Dateinamen angeben. Um den alten Zustand dieses Schlüssels mit seiner Zeichenfolge und dem zugeordneten Wert wiederherzustellen, mußt Du im Explorer die entsprechende Datei mit der Endung **REG** suchen. Hast Du beim Exportieren den Namen „Sicherung Winwordeinstellung“ vergeben, findest Du in dem Ordner, wo Du die Datei abgelegt hast (das ist üblicherweise **C:\WINDOWS**) die Datei **SICHERUNG WINWORDEINSTELLUNG.REG**. Ein Doppelklick auf diese Datei genügt, und alle Einstellungen des gesicherten Originals werden wieder zurück in die Registry geschrieben.

Problematisch wird es, wenn Du den Schlüssel nicht wie angeraten gesichert hast. Dann kannst Du entweder versuchen, das betroffene Programm wieder neu zu konfigurieren. Am Beispiel von Winword könntest Du beispielsweise über das Menü **Extras, Optionen** gehen. Dort lassen sich die meisten Einstellungen vornehmen; vielleicht ist auch die von Dir in der Registry vorgenommene Änderung hier wieder zu korrigieren. Hilft auch das nichts, bleibt wohl oder übel nur die Neuinstallation des betroffenen Programmes, denn dann werden die von Dir eingesetzten fehlerhaften Einträge mit korrekten Einstellungen überschrieben.

Probleme beim Start

Wenn Windows schon beim Booten eine oder mehrere Fehlermeldungen ausgibt, solltest Du den Bootvorgang protokollieren lassen. Dazu drückst Du bei Erscheinen der Meldung „Windows 9x wird gestartet“ die Taste **F8** oder **Strg**, um in das Bootmenü zu gelangen. Hier findest Du den entsprechenden Eintrag **Protokolliert**. Nach dem Booten gibt es im Hauptverzeichnis des Bootlaufwerks, also üblicherweise **C:**, eine Datei **BOOTLOG.TXT**, die das Starten jedes Treibers und den Erfolg oder Mißerfolg festhält. Du solltest die Datei in den Editor laden und nach der Zeichenfolge „Failed“ suchen. Dabei muß jedoch nicht jedes „Failed“ einen wirklichen Fehler bedeuten. Windows fügt beispielsweise die Zeile

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

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

ein, wenn kein erweitertes BIOS gefunden wurde. Vielleicht kannst Du aber so eine fehlende, versehentlich gelöschte Datei als Übeltäter ausmachen.

Erscheint beim Booten kurz eine Fehlermeldung der **AUTOEXEC.BAT** oder **CONFIG.SYS**, die Du nicht so schnell lesen konntest, drückst Du beim nächsten Booten einfach die **Pause**-Taste. Sofern Du die fehlerverursachende Zeile herausbekommst, startest Du mit Deiner (aktuellen) Bootdiskette. Lade die Datei in einen Editor und setze vor diese Zeile das Wort **rem** sowie ein Leerzeichen, um sie zu deaktivieren.

Liegt der Fehler anscheinend an einem fehlerhaften Registry-Eintrag, kopierst Du nach dem Start mit der Bootdiskette Deine Sicherungskopien von **USER.DAT** und **SYSTEM.DAT** in den Ordner **C:\WINDOWS** und überschreibst die alten Dateien damit.

Solltest Du unglücklicherweise keine Startdiskette zur Verfügung haben, kannst Du bei Erscheinen der Meldung „Windows 95 wird gestartet“ die Taste **F8** drücken und das Bootmenü aufrufen. Unter Windows 98 verwendest Du statt dessen **Strg**. Versuche mit der Option **“Abgesichert”**, **“Abgesichert nur Eingabeaufforderung”** oder **“Nur Eingabeaufforderung”** zu starten, um dann die Registry mittels Deiner Sicherungskopie oder der Windows-Sicherung **USER.DA0** und **SYSTEM.DA0** zu restaurieren bzw. den Eingriff an **AUTOEXEC.BAT** oder **CONFIG.SYS** vorzunehmen.

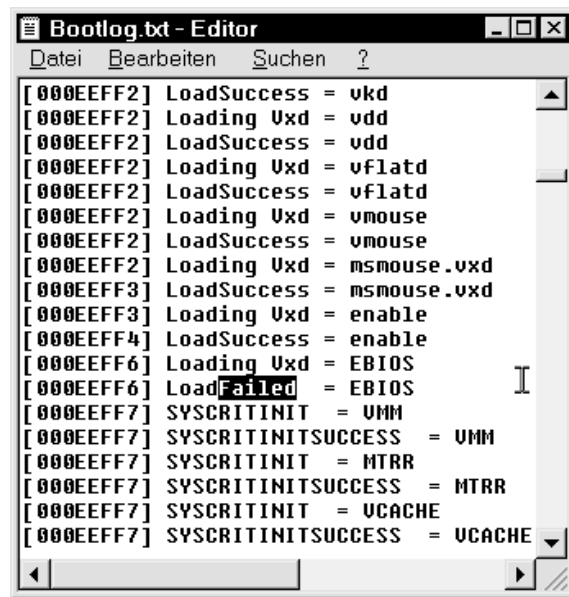


Abbildung 30: Die Bootlog.txt hilft bei der Fehlersuche

Fehlermeldung nach Programmdeinstallation

Ein oft harmloses, aber doch nervendes und sorgenbereitendes Problem kannst Du erleben, wenn ein Programm nicht richtig deinstalliert wurde. Das kann passieren, wenn die Deinstallationsroutine nicht richtig gearbeitet hat und zum Beispiel (Treiber)Dateien gelöscht, aber die Aufrufe aus Registry, **SYSTEM.INI** & Co. nicht entfernt hat. Oder Du hast einfach das Programmverzeichnis einer Anwendung gelöscht und es sind ebenfalls Rückstände in den Systemdateien zurückgeblieben. In beiden Fällen versucht Windows beim Start, analog der Anweisungen in den Systemdateien und der Registry bestimmte Dateien zu laden, die nicht mehr vorhanden sind.

Als Folge erhältst Du in den meisten Fällen eine Fehlermeldung, die erst durch einen Druck auf **Enter** oder einen Klick auf **OK** bestätigt werden muß, bevor der Bootvorgang fortgesetzt wird. Wenn die fehlende Datei bis auf die Fehlermeldung keine weiteren Probleme bereitet, müssen wir nur den Aufruf entfernen. An allererster Stelle würde hier das Starten der Registry stehen, wo Du über **Bearbeiten, Suchen** den Dateinamen eingibst. Fundstellen solltest Du über **Registrierung, Registrierungsdatei exportieren** sichern und dann die entsprechende Zeichenfolge oder den Schlüssel löschen. Läßt sich die problematische Datei in der Registry nicht finden, lädst Du den Systemeditor **SYSEDIT.EXE** über die **Ausführen**-Funktion im Startmenü. Dann suchst Du nach der Datei nacheinander in **SYSTEM.INI, WIN.INI, AUTOEXEC.BAT** und **CONFIG.SYS**. In letzteren beiden setzt Du vor die relevante Zeile das Wort **REM** und ein Leerzeichen.

Systemtuning mit der Registry

Windows schneller machen und aufräumen

Verzögerung im Startmenü einstellen

Die Einträge im Startmenü klappen standardmäßig mit Verzögerung auf. Windows-Anwender sind über diese Funktion geteilter Meinung: Die einen halten es für pure Zeitverschwendung, Anfänger finden es vorteilhaft, daß man langsam mit der Maus über die Einträge fahren kann, ohne daß sofort die Untergruppen aufklappen. Welcher Partei Du auch immer angehörst, Du kannst die Verzögerungszeit entweder verlängern oder verkürzen. Wechsele dazu in den Registrierpfad

HKEY_CURRENT_USER\ControlPanel\desktop

Dort findest Du im rechten Fenster die Zeichenfolge „MenuShowDelay“. Falls nicht, klicke den Schlüssel **desktop** mit der rechten Maustaste an, wähle **Neu, Zeichenfolge**, und gib den obigen Namen ein. Ein Doppelklick auf den (neuen) Eintrag läßt Dich eine beliebige Zahl als Ausdruck für die Verzögerung in Millisekunden eingeben. Soll es keine Verzögerung geben, mußt Du den Wert auf 0 setzen. Für eine längere Verzögerung kannst Du jedoch auch einen Wert von beispielsweise 500 oder höher eingeben.

Startmenü – abgespeckt

Das Startmenü von Windows umfaßt standardmäßig verschiedene Einträge wie **Suchen** und **Ausführen**. Diese Einträge lassen sich nicht löschen, denn sie werden im Explorer auch unter **\WINDOWS\STARTMENÜ** nicht angezeigt. Doch es gibt nichts, was sich mit der Registry nicht lösen ließe: Rufe den Schlüssel

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

Erstelle einen Binärwert **NoRun**, dem Du nach einem Doppelklick den Wert **01** zuweist, um den Eintrag **Ausführen** (=Run) aus dem Startmenü zu werfen. Analog lassen sich die Einträge **Systemsteuerung** und **Drucker** aus **Start, Einstellungen** streichen, wenn Du einen Binärwert **NoSetFolders** samt **01** erstellst.



Abbildung 32: Hier fehlen „Suchen“ und „Ausführen“

Ausführen-Liste löschen

Im Zuge einer gestiegenen Benutzerfreundlichkeit hat sich Windows angewöhnt, alle Aktionen und Eingaben zu protokollieren und zu speichern. Dazu gehört beispielsweise auch die automatische Speicherung der letzten Eingaben im **Ausführen**-Dialog, so daß Du schon einmal eingegebene Werte bequem aus einem Listefeld auswählen kannst.

Willst Du diese Liste löschen, um Deine Aktionen vor neugierigen Mitmenschen zu verstecken, mußt Du in der Registry unter **HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU** die Einträge „a“, „b“, „c“, „d“ usw. löschen. Die Informationen werden nach dem Neustart entfernt.

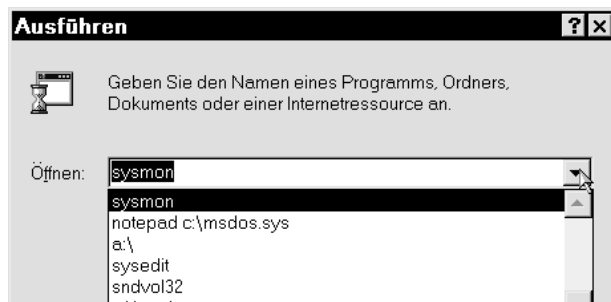


Abbildung 33: „Ausführen“ merkt sich die letzten Eingaben – ein gefundenes Fressen für Neugierige

Uninstall-Liste bereinigen

Windows vermerkt für alle Programme, die Du per Setup installierst, die Einstellungen für die Deinstallation des Programmes. Damit soll sichergestellt werden, daß auch Dateileichen wie alte **DLL**-Dateien im **WINDOWS\SYSTEM-**Verzeichnis mit entfernt werden. Diese Dateien würden sonst zurückbleiben, wenn Du einfach nur den Programmordner löschst. Soweit die Theorie – in der Praxis funktioniert die Uninstall-Routine nicht immer korrekt, so daß manchmal die Namen von Programmen auch nach der Deinstallation in der Softwareliste zurückbleiben. Du findest die Deinstallationsoption unter **Start, Einstellungen, Systemsteuerung, Software**. Diese Liste kannst Du über den Registriereditor modifizieren. Lösche die Unterschlüssel von **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall**, die auf nicht mehr vorhandene Programme hinweisen. Damit erscheinen diese Programme auch nicht mehr in der Uninstall-Liste.

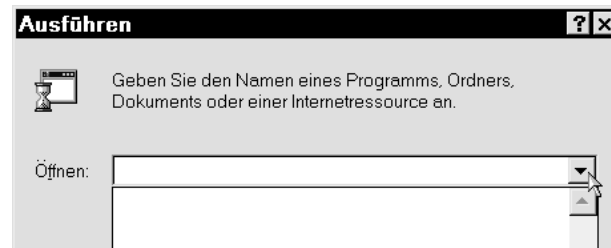


Abbildung 34: ... doch nun ist die Liste leer

Optisches Tuning

Bitmaps als Icon darstellen

Wenn Du im Explorer oder Arbeitsplatz Dateien mit Symbolen anzeigen läßt, erscheint bei Grafikdateien mit der Endung **BMP** standardmäßig das Icon der Anwendung, die dem Grafiktyp zugeordnet ist, also beispielsweise das Icon von Paint. Mit einem kleinen Eingriff in die Registry kannst Du jedoch einstellen, daß statt dessen immer das komplette Bild verkleinert als Icon dargestellt werden soll, sozusagen als Vorschau.

Wechsle zu **HKEY_CLASSES_ROOT\Paint.Picture\DefaultIcon** und ändere den Wert der Zeichenfolge „Standard“ nach einem Doppelklick in **%1** um. Dieser Wert steht als Platzhalter und bewirkt, daß nicht irgendein Icon angezeigt wird, sondern der Inhalt des jeweiligen Bitmaps. Im Explorer kannst Du über das Menü **Ansicht** die Darstellungsart und damit die Größe der Icons beeinflussen.

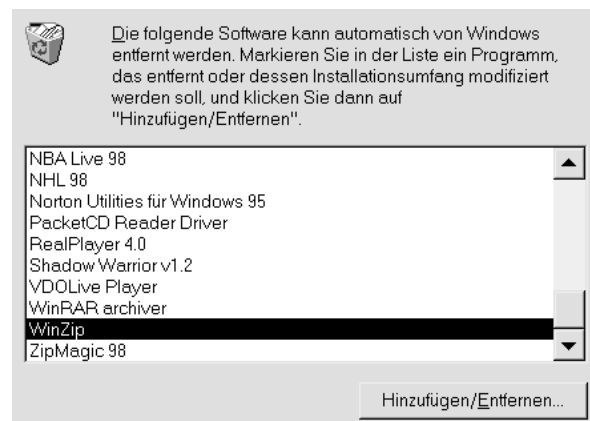


Abb 35: Diese Liste offeriert Programme zum Deinstallieren, die gar nicht mehr auf meiner Festplatte sind

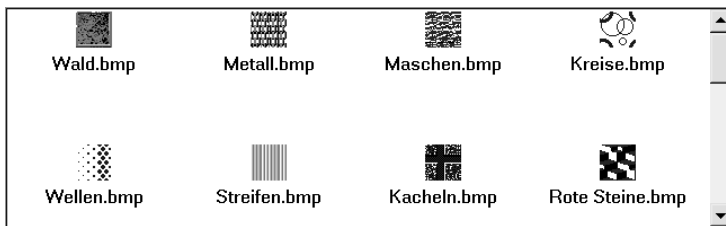


Abbildung 36: Das Icon liefert eine Vorschau auf das Bild

Hintergrundbild genau plazieren

Bei den Einstellungen zum Hintergrundbild ist Windows nicht gerade flexibel, es sei denn, man verwendet den Active Desktop von Windows 98 bzw. des Internet Explorers 4, welcher das System ziemlich bremst. Lediglich die Optionen **Mitte** zur Zentrierung und **Kachel** zum Auslegen des gesamten Desktops werden angeboten. Mit Hilfe des Registrieditors kannst Du trotz der Einstellung **Mitte** ein Bild beliebig auf dem Bildschirm plazieren. Denkbar wäre zum Beispiel ein Anwendungsfall, bei dem das Firmenlogo immer in der rechten unteren Ecke stehen soll, was mit den normalen Einstellungen nicht möglich wäre. Wechsle also in den Registrieditor und suche den Registrierpfad

HKEY_CURRENT_USER\ControlPanel\Desktop. Klicke den Schlüssel **\Desktop** mit der rechten Maustaste an und wähle **Neu, Zeichenfolge**. Erstelle die Zeichenfolgen **WallpaperoriginX** und **WallpaperoriginY**. Weise diesen Zeichenfolgen die genauen Pixelwerte zu, an denen das Bild anfangen soll. Bei einer Bildschirmauflösung von 640x480 wären je nach Größe des Bildes beispielsweise die Werte **550** und **380** einzutragen. Nach einem Neustart des Rechners kannst Du die Position des Hintergrundbildes überprüfen.

Übrigens, wenn die aktive Hintergrundfarbe nicht zu Deinem Hintergrundbild paßt, kannst Du dies mit einem Rechtsklick auf den Desktop und **Eigenschaften, Darstellung** ändern.

Icons mit 16 Bit Farbtiefe

Eine Funktion, die viele User mit dem Plus!-Paket für Windows 95 erhalten haben, ist die Anzeige von Icons mit 16-Bit-Farbtiefe. Die Funktion ist aktivierbar, indem Du in der Registry den Schlüssel **HKEY_CURRENT_USER\ControlPanel\desktop\WindowMetrics** mit der rechten Maustaste anklickst und **Neu, Zeichenfolge** wählst. Als Name trägst Du **Shell Icon BPP** ein. Nach einem Doppelklick auf den neuen Eintrag weist Du den Wert **16** zu, womit die Farbtiefe auf 16 Bit per Pixel (BPP) gesetzt wird. Damit Du in den Genuß der erweiterten Icons kommst, muß Deine Grafikkarte „HighColor“ beherrschen und der Grafikkartentreiber auf diese oder eine höhere Farbtiefe eingestellt sein. Unter Windows 98 ist läßt sich dies einfacher realisieren: Klicke den Desktop mit der rechten Maustaste an und wähle **Eigenschaften, Effekte, Symbole mit höchster Farbanzahl anzeigen**.

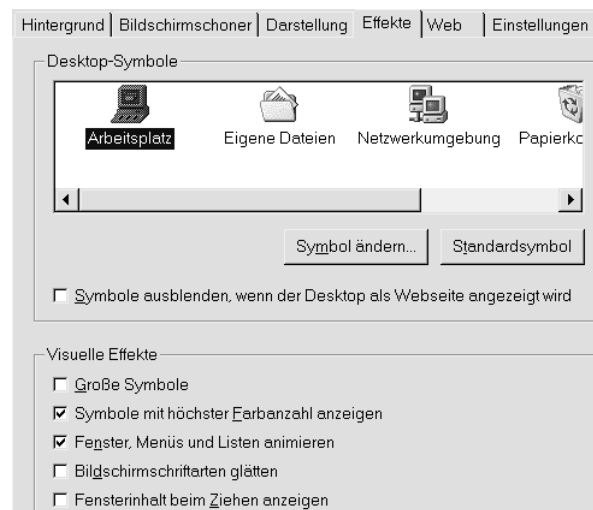


Abbildung 37: Unter Windows 98 lassen sich viele Sachen einfacher einstellen

„Klicken Sie hier um zu starten“ ausschalten

Du kannst die Taskleiste mit dem Startknopf über die Optionen **Automatisch im Hintergrund** und **Immer im Vordergrund** so



konfigurieren, daß sie erst sichtbar wird, wenn Du den Mauszeiger an den unteren Bildschirmrand bewegst. Verwendest Du diese Einstellung, erscheint in der Taskleiste immer ein Hinweis „Klicken Sie hier, um zu starten“, der mit einem Pfeil auf den Startknopf zeigt. Diesen Hinweis kannst Du auch abschalten: Starte den Registriereditor und suche den Schlüssel **HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies**. Klicke den Unterschlüssel **\Explorer** rechts an und wähle **Neu, Binärwert**. Es wird ein neuer Wert angelegt, dem Du den Namen **NoStartBanner** gibst. Klicke den neu geschaffenen Eintrag doppelt an und weise den Wert **01 00 00 00** zu.

Animation beim Minimieren und Maximieren

Wann immer Du ein Programmfenster über die entsprechende Schaltfläche am rechten oberen Bildschirmrand minimierst oder maximierst, läßt Windows das Fenster in einer Animation zu- oder aufklappen. So schön diese Funktion auch sein mag, auf langsamen Rechnern belastet sie das System unnötig. Um die Animation abzuschalten, wechsele in Registriereditor zum Schlüssel **HKEY_CURRENT_USER\ControlPanel\Desktop\WindowMetrics**. Nach einem Rechtsklick auf **\WindowMetrics** und **Neu, Zeichenfolge** nennst Du die neue Zeichenfolge **MinAnimate**. Als Wert gibst **0** ein. Nun ist die Animation beim Minimieren oder Maximieren von Programmfenstern ausgeschaltet; aktivieren läßt sie sich jedoch jederzeit durch Ändern des Wertes auf **1**.

Icon für Auflösung und Farbanzahl neben der Uhr

Eine sehr nützliche Sache läßt sich ebenfalls über die Registry realisieren: Mit einer einzigen Zeichenfolge wird im Systemschacht direkt neben der Uhr ein Icon installiert, das nach dem Anklicken alle möglichen Kombinationen von Auflösung und Farbtiefe anzeigt. Somit kannst Du in Windeseile beispielsweise von der Auflösung 640*480 mit Truecolor (24 Bit Farbtiefe) auf 800*600 in High Color (16 Bit) wechseln. So installierst Du das Icon: Klicke den Schlüssel



Abbildung 38: Dieses Icon ist das Ergebnis unserer Mühe

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run

mit der rechten Maustaste an und erstelle über **Neu, Zeichenfolge** den neuen Eintrag **Taskbar Display Controls**. Nach einem Doppelklick gibst Du folgenden Wert ein:

RunDLL

deskcp16.dll,QUICKRES_RUNDLLENTY. Nach dem nächsten Neustart solltest Du das neue Icon neben der Uhr finden. Leider funktioniert dies unerklärlicherweise nicht auf allen Systemen.



Abbildung 39: ... und so ein praktisches Menü öffnet es auf Klick

Windows über die Registry anpassen

Benutzernamen ändern

Bei der Installation von Windows mußt Du einen Namen und gegebenenfalls eine Firma angeben. Diese Namen speichert Windows wie fast alle Systemeinstellungen in der Registry, und zwar im Schlüssel **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion** unter „RegisteredOwner“ und „RegisteredOrganization“. Durch Doppelklick auf diese Zeichenfolgen kannst Du den Namen ändern, wenn Du Dich bei der Installation verschrieben hast oder beispielsweise durch Heirat den Namen gewechselt hast.

Setuppfade anpassen

Wenn Du zwecks schnellerem Zugriff die Windows-Installationsdateien auf die Festplatte kopiert hast, wirst Du Windows darauf sicherlich hinweisen wollen. Andernfalls fragt Windows ständig nach dem Verbleib der Win-CD, sobald Du über das Setup Komponenten (de)installierst. Die diesbezügliche Einstellung findest Du unter

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Setup

Im rechten Fenster findest Du dort die Zeichenfolge „SourcePath“, die Du nach einem Doppelklick editieren kannst. Ändere den Pfad Deines CD-Laufwerkes, indem Du die neue Position der Installationsdateien auf der Festplatte angibst. Dieser Trick ist auch dann nützlich, wenn sich durch Einbau zusätzlicher Festplatten der Buchstabe des CD-Laufwerkes geändert hat.

Wenn sich durch den Einbau einer neuen Festplatte der Laufwerksbuchstabe Deines CD-ROM-Laufwerks verschoben hat, stimmt außerdem der in der Registry eingetragene Installationspfad des Plus!-Paketes nicht mehr. Die neue Laufwerkskennung kannst Du unter

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Plus!\Setup

im Wert von „SourcePath“ angeben. Du kannst hier auch einen Hinweis auf ein Verzeichnis auf der Festplatte anlegen, wenn Du die Installationsdateien dorthin kopiert hast.

Autostart nur von Audio-CDs unterdrücken

Windows erkennt, wenn Du eine Audio-CD eingelegt hast, und startet die Wiedergabe automatisch. Genauso verfährt es mit Daten-CDs, die eine sogenannte Autorun-Datei im Stammverzeichnis enthalten. Wenn Du den Autostart umgehen willst, kannst Du beim Einlegen der CD die **Shift**-Taste gedrückt halten. Auf Dauer wird dies jedoch zu umständlich, so daß Du über **Systemsteuerung, System, Gerätemanager, CD-ROM, Dein CD-ROM-Laufwerk, Eigenschaften** die Funktion auch gänzlich ausschalten kannst. Hinter dem Register **Einstellungen** findest Du die Option **Automatische Benachrichtigung beim Wechsel**, die Du mit einem Klick auf das Häkchen deaktivierst.

Wenn Du jedoch nur den Autostart von Musik-CDs verhindern willst, nützt Dir diese Option nichts, denn sie blockiert gleichzeitig auch normale Datenscheiben. Wechsele also in den Registriereditor, wo Du die Einstellungen für beide Typen von CDs getrennt festlegen kannst. Unter

HKEY_CLASSES_ROOT\AudioCD\shell\play\command klickst Du doppelt auf „Standard“ und löschst den zugeordneten Befehl **C:\WINDOWS\CDPLAYER.EXE /PLAY %1**. Damit entfällt der Autostart von Musik-CDs, ohne daß das Verhalten von Daten-CDs beeinflusst wird. Um die

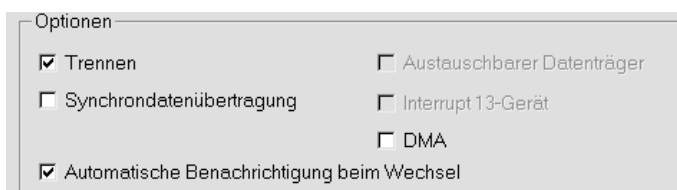


Abbildung 40: Hier schaltest Du sämtliche Automatik ab

Aktion später wieder rückgängig zu machen, brauchst Du nur den Wert wieder mit der Befehlszeile zu versehen.

Diese Einstellung kannst Du übrigens auch über den Explorer vornehmen. Wähle **Ansicht, Optionen, Dateitypen** und klicke den Eintrag **Audio-CD** doppelt an. Durch **Entfernen** des Vorgangs **Wiedergeben** löschst Du die Autorun-Automatik; später kannst Du den Vorgang über **Neu** wieder aktivieren. Nach dem Neueinrichten klickst Du auf **Als Standard**, und alles funktioniert wie gehabt.

Speichern von Einstellungen verhindern

Windows speichert automatisch am Ende jeder Sitzung die aktuellen Einstellungen des Desktops. Dazu zählen unter anderem die Iconanordnungen und die Größe und Position der Taskleiste. Damit ein einmal aufgeräumter Desktop nach versehentlichen oder mutwilligen Änderungen nicht beim nächsten Start chaotisch erscheint, kannst Du die automatische Speicherung der Einstellungen verhindern. Dazu gibt es im Registriereditor den Schlüssel **HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer**. Erstelle einen Binärwert „NoSaveSettings“, dem Du nach einem Doppelklick den Wert „01“ zuweist. Mit dieser Option schaltest Du übrigens auch die Eigenart von Windows ab, die am Ende der letzten Sitzung noch geöffneten Programme beim Start wieder aufzurufen.

Programme beim Booten versteckt starten

Wenn Du willst, daß ein Programm bei jedem Start von Windows automatisch aufgerufen wird, kannst Du eine Verknüpfung dieses Programms im Autostart-Ordner ablegen. Eine solche per Autostart aufgerufene Anwendung kannst Du jedoch durch simples Drücken der **Shift**-Taste während der Anzeige des Windows-Logos umgehen und anschliessend aus dem Autostart-Ordner entfernen. Wesentlich sicherer ist da die Funktion **Run**, die wie Autostart angegebene Programme nach dem Booten aufruft, nur eben versteckter. Suche im Registriereditor den Schlüssel

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

Im rechten Teil des Editorfensters siehst Du den Eintrag „Standard“ und vielleicht auch weitere Programmbefehle. Wenn Du beispielsweise Winword automatisch starten lassen willst, klickst Du den Schlüssel **Run** mit der rechten Maustaste an und wählst **Neu, Zeichenfolge**. Nenne die neue Zeichenfolge **Winword** und weise ihr nach einem Doppelklick die Befehlszeile **C:\Programme\Office\Winword\Winword.exe** zu. Gegebenenfalls muß Du den Pfad an Deine bestehenden Verhältnisse anpassen. Ein per **Run** aufgerufenes Programm ist weder im Autostart-Ordner sichtbar, noch läßt es sich über das Drücken der **Shift**-Taste beim Start umgehen. Gewiefte Anwender kennen den Trick natürlich, weshalb Du Dich nicht 100%ig auf die Sicherheit der Funktion verlassen solltest.

Umbenennen-Schutz des Papierkorbs ausschalten

Standardmäßig ist der Papierkorb vor dem Umbenennen geschützt. Wenn Du das Icon zweimal zeitversetzt anklickst oder markierst und dann **F2** drückst, ertönt nur ein Signalton aus dem Lautsprecher, der Dir anzeigt, daß ein Umbenennen nicht möglich ist. Allerdings kannst Du diese Sperre in der Registry ausschalten, um den Papierkorb nach Tageslaune in beispielsweise „Datenmülltonne“, „Digitalrecycling“ oder „Friedhof der Dateien“ umzubenennen. Wechsle dazu zum Schlüssel

HKEY_CLASSES_ROOT\CLSID\{645FF040-5081-101B-9F08-00AA002F954E}\ShellFolder

Wenn Du die lange Zeichenkette anklickst, erscheint im rechten Teil des Explorers der Klartextname „Papierkorb“. Nun klickst Du jedoch **\ShellFolder** an und änderst nach einem Doppelklick auf „Attributes“ das erste Zahlenpaar von normalerweise „40“ auf **10**.

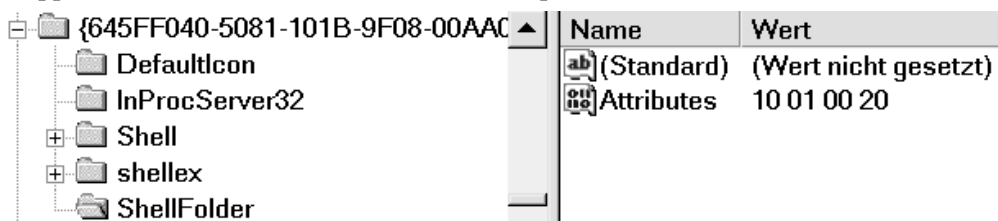


Abbildung 41: Die Attribute des Papierkorbs

Jetzt kannst Du den Papierkorb jederzeit umbenennen. Der Eintrag „Attributes“ ist ein sogenannter EditFlag, der bestimmte Beschränkungen von Objekten festlegt.



„Abmelden“- und „Favoriten“-Eintrag aus dem Startmenü verbannen

Windows 98 beziehungsweise Windows 95 plus Internet Explorer fordern beim ersten Start nach der Installation die Anmeldung mit einem Nutzernamen und optional einem Paßwort. Nutzt Du Deinen PC allein und möchtest deshalb keine Benutzerprofile verwenden, solltest Du hier nur einen Namen eingeben, auf ein Paßwort aber verzichten. Auf diese Weise belästigt Windows Dich nicht mehr mit dem Anmeldeprozedere; andernfalls mußt Du bei jedem Start Name und Paßwort eingeben. Auf jeden Fall erscheint als vorletzter Eintrag des Startmenüs **Benutzername abmelden**, den Du bei Belieben ausblenden kannst.

Dazu mußt Du

HKEY_CURRENT_USER\Software\Microsoft\CurrentVersion\Policies\Explorer

mit der rechten Maustaste anklicken und **Neu, Binärwert** wählen. Nenne den neuen Eintrag **NoLogOff** und weise ihm den Wert **01** zu.

Im gleichen Schlüssel kannst Du auch über **Neu, Binärwert** einen Eintrag namens **NoFavoritesMenu** erstellen. Weist Du hier den Wert **01** zu, verschwindet auch der **Favoriten**-Eintrag im Startmenü.



Abbildung 42: Hier sind die Einträge noch vorhanden ...

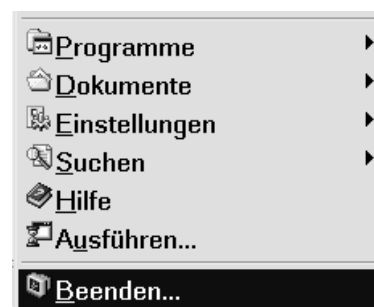


Abbildung 43: ... hier nicht mehr

Systemausbau mit der Registry

Kontextmenüs anpassen

Reihenfolge von Menüeinträgen

Der oberste Eintrag im Kontextmenü ist immer der Befehl, der bei Doppelklick ausgeführt wird. Das erkennst Du daran, daß er **fett** formatiert ist. Danach folgen alternative Befehle wie **Drucken** oder **Bearbeiten**. Die Reihenfolge der Kontextbefehle läßt sich dabei leicht in der Registry umstellen. Um das Kontextmenü von Batchdateien zu verändern, das als Standardbefehl **Öffnen** vorsieht, gehst Du zu

HKEY_CLASSES_ROOT\batfile\shell.

Dort findest Du die Unterschlüssel **\edit**, **\open** und **\print** für die Befehle **Bearbeiten**, **Öffnen** und **Drucken**. Wenn Du nun möchtest, daß **BAT**-Dateien auf Doppelklick nicht ausgeführt (open), sondern in den Editor zum Bearbeiten geladen werden, mußt Du im rechten Bereich von **\shell** doppelt auf „Standard“ klicken und die Reihenfolge **edit**, **open**, **print** eingeben. Nun steht im Kontextmenü an erster Stelle **Bearbeiten** als fetter Eintrag; danach folgen **Öffnen** und **Drucken**.

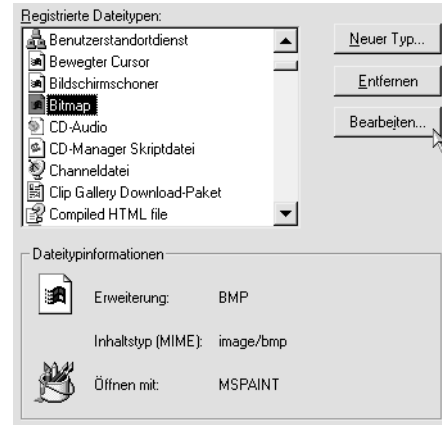


Abbildung 44: Den Dateityp auswählen, „Bearbeiten“ klicken...

Die Reihenfolge von Kontextbefehlen läßt sich zwar nur in der Registry einstellen den Standardbefehl kannst Du aber auch über ein Dialogfenster ändern. In Windows 98 wählst Du im Explorer **Ansicht, Ordneroptionen, Dateitypen**; unter Windows 95 findest Du den Dialog über **Ansicht, Optionen, Dateitypen**. Wenn Du beispielsweise **BMP**-Dateien auf Doppelklick drucken statt öffnen möchtest (was natürlich nur sinnvoll ist, wenn Du viele Bilder drucken mußt), klickst Du den Typ „Bitmap“ an und wählst **Bearbeiten, print, Als Standard**.

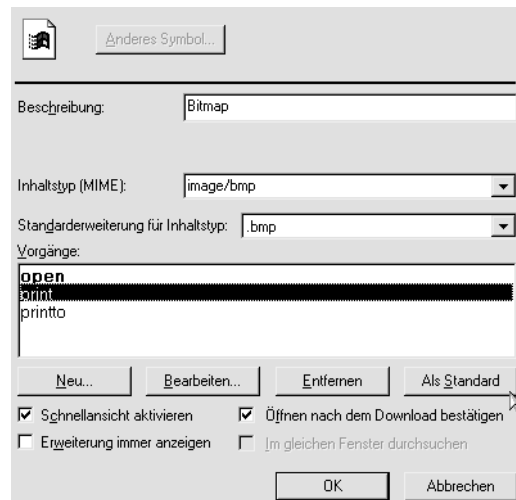


Abbildung 45: ... den gewünschten Befehl und "Als Standard" wählen

Scandisk im Kontextmenü von Laufwerken

Wenn Du ein Laufwerkssymbol im Arbeitsplatz oder Explorer mit der rechten Maustaste anklickst, erscheint ein Kontextmenü, das standardmäßig schon eine Vielzahl von Einträgen enthält. Du kannst das Kontextmenü aber auch noch erweitern, beispielsweise mit dem Befehl **Scandisk**. Gehe dazu wie folgt vor: Rufe **Start, Ausführen, Regedit** auf und wechsele in den Registrierpfad **HKEY_CLASSES_ROOT\Drive\Shell**. Öffne durch Rechtsklick auf **\Shell** das Kontextmenü, wähle **Neu, Schlüssel**, und erstelle den Schlüssel **Scandisk**. Im rechten Fenster erscheint die Zeichenfolge „Standard“, der Du nach einem Doppelklick den Kontexteintrag „Scan&disk“ zuweist.

Das kaufmännische Und (&) gibt den Buchstaben an, mit dem Du den Befehl aus dem Kontextmenü per Tastatur aufrufen kannst. Du kannst also auch mit der rechten Maustaste (oder mit **Shift+F10**) das Kontextmenü öffnen, und dann durch Druck auf **D** den Befehl per Tastatur auslösen. Ebenso ist es möglich, auch einen anderen Buchstaben als Befehlsaktivator zu definieren, indem Du einfach das Zeichen vor einen anderen Buchstaben setzt.

Klicke den Schlüssel **\Scandisk** mit der rechten Maustaste an und erstelle über **Neu, Schlüssel** den Unterschlüssel **Command**. Auch der Unterschlüssel **\Command** enthält die Zeichenfolge „Standard“, der Du nach einem Doppelklick den Befehl **c:\windows\scandiskw.exe** zuweist.

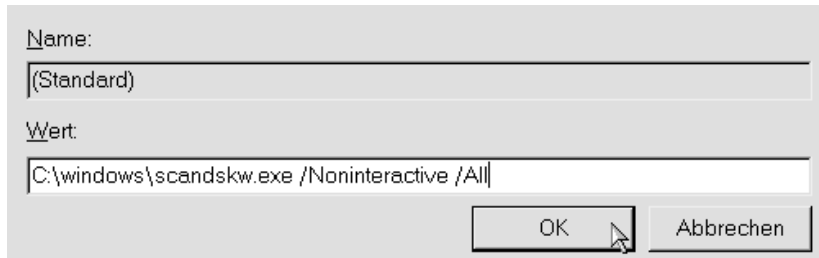


Abbildung 46: Automatik per Parameter

Du kannst dem Befehlsstring auch Parameter anhängen, zum Beispiel den Parameter **/All**, der automatisch alle Laufwerke zur Überprüfung auswählt. Oder Du automatisierst den Vorgang mit dem Parameter **/Noninteractive**, durch den Windows die Datenträgeranalyse sofort ohne weiteres Zutun des Anwenders startet.

Neu-Menü entrümpeln oder erweitern

Den Befehl **Neu** findest Du an vielen Stellen, beispielsweise nach einem Rechtsklick im Kontextmenü des Desktops oder auch im Menü **Datei** des Explorers. Je mehr Anwendungsprogramme Du installierst, um so länger und unübersichtlicher wird dabei die Liste der Dateitypen, die Du neu erstellen kannst. Um die Liste zu entrümpeln, mußt Du – wie könnte es anders sein – den Registriereditor bemühen. Suche den relevanten Dateityp, zum Beispiel **WAV** unter **HKEY_CLASSES_ROOT**.

Unterstützt der Dateityp die Funktion **Neu**, ist im Registriereditor jeweils ein Unterschlüssel **\ShellNew** vorhanden – im gewählten Beispiel heißt der komplette Registrierpfad **HKEY_CLASSES_ROOT\wav\ShellNew**.

Lösche den Unterschlüssel **\ShellNew**, und der Dateityp erscheint zukünftig nicht mehr im **Neu**-Menü.

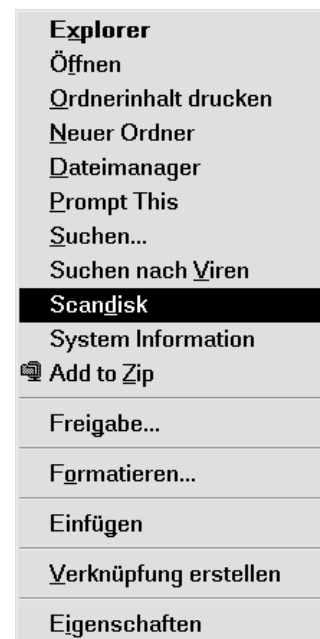


Abbildung 47: Scandisk im Kontextmenü aller Laufwerke

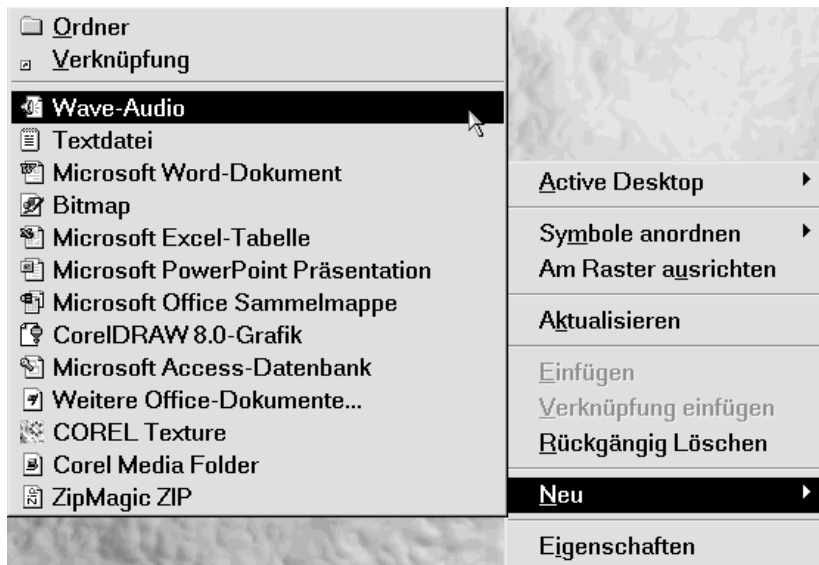


Abbildung 48: Hier ist WAVE-Audio noch da...

Genauso wie Du nach obigem Tip das Menü des Befehls **Neu** entrümpeln kannst, ist es auch möglich, neue Einträge hinzuzufügen. Erstelle unter **HKEY_CLASSES_ROOT** und dem entsprechenden Dateityp einen Unterschlüssel mit dem Namen „ShellNew“. Dann legst Du die Zeichenfolge „NullFile“ an. Willst Du beispielsweise über das **Neu**-Menü bequem Internetseiten in HTML erstellen, mußt Du im Registrireditor unter **HKEY_CLASSES_ROOT\htm** den Unterschlüssel „ShellNew“ mit der Zeichenfolge „NullFile“ erstellen.

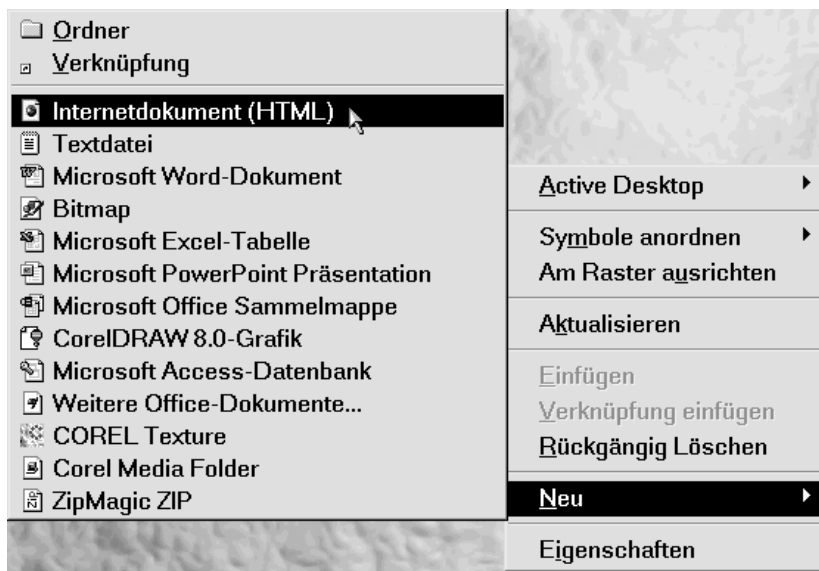


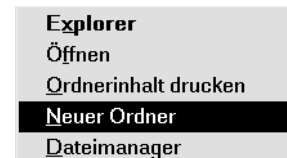
Abbildung 49: ... nun ist er weg und ein neuer Eintrag da

Neuen Ordner über Kontextmenü erstellen

Wenn Du im Explorer einen neuen Ordner erstellen möchtest, wirst Du als eingefleischter „Rechtsklicker“ einen entsprechenden Befehl im Kontextmenü vermissen. Das ist ein Manko, das seit den ersten Versionen von Windows bekannt und trotzdem nie behoben wurde. Die einzige Möglichkeit scheint, erst den Ordner anzuklicken und dann im rechten Bereich an einer freien Stelle einen Rechtsklick gefolgt von **Neu, Ordner** zu tätigen. Möchtest Du aber auch Verzeichnisse in der linken Hälfte des Explorers anklicken und **Neuer Ordner** wählen können, ist es Zeit, selbst Hand anzulegen.

Dazu gehst Du in die Registry zum Schlüssel

HKEY_CLASSES_ROOT\Folder, den Du mit der rechten Maustaste anklickst und dann **Neu, Schlüssel** wählst. Der neue Schlüssel soll **Neuer Ordner** heißen. Als nächsten Schritt klickst Du **\Neuer Ordner** rechts an und wählst noch einmal **Neu, Schlüssel**.



Jetzt vergibst Du den Namen **command**. Im rechten Teil von **\command** klickst Du doppelt auf den Eintrag „Standard“ und weist ihm den Wert **command.com /c md neu** zu. Nun kannst Du im Explorer jedes beliebige Verzeichnis mit der rechten Maustaste anklicken. Wenn Du den Befehl **Neuer Ordner** wählst, wird ein Unterordner namens **\NEU** angelegt, den Du durch zwei zeitversetzte Klicks oder durch Drücken der Taste **F2** umbenennst.

Antivirensan im Kontextmenü von Laufwerken

DOS-Programme sind bis auf wenige Ausnahmen eigentlich nicht mehr zeitgemäß. Eine dieser Ausnahmen sind zum Beispiel Virens Scanner, die sich als DOS-Anwendung gut von einer Startdiskette starten lassen, wenn Du von einer Invasion der digitalen Plagegeister bedroht bist. Der größte Nachteil der DOS-Helfer gegenüber ihren Windows-Pendants ist meist die Notwendigkeit, den Scanner mit Parametern aufrufen zu müssen, statt schnell auf ein Icon zu klicken. Also wollen wir das Kontextmenü von Laufwerken um einen Befehl erweitern, der sofort einen Scan des betreffenden Laufwerkes auslöst. Als Beispiel nehmen wir den Virens Scanner von McAfee. Rufe **HKEY_CLASSES_ROOT\Drive\shell** auf und erstelle einen Unterschlüssel namens „Virensan“, indem Du **\shell** mit der rechten Maustaste anklickst und **Neu, Schlüssel** wählst. Genauso erstellst Du einen Unterschlüssel des neuen **Virensan**, der den Namen **Command** tragen muß. Klicke **\Command** an, und gib auf der rechten Seite als Wert des Eintrags „Standard“ die Programmanweisung **E:\ANTIVIR\MCAFEE\SCAN.EXE /ALL /SUB %1** ein. Diese Anweisung hängt natürlich davon ab, ob Du wirklich den Scan von McAfee einsetzt und wo Du die Datei gespeichert hast. Passe den Wert also entsprechend an.

Ordnerinhalt ausdrucken oder in Datei speichern

Um sich einen Überblick über alle in einem Ordner vorhandenen Dateien zu verschaffen, kann man den Explorer oder den Befehl **Dir** an der Eingabeaufforderung verwenden. Oftmals möchte man die Dateiliste ausdrucken oder in einer Datei speichern, um sie zu archivieren. Dies hat einen entscheidenden Vorteil: Wenn ein Programm bei seiner Installation Dateien beispielsweise in das **WINDOWS**-Verzeichnis kopiert hat, die Deinstallationsroutine diese jedoch nicht entfernt hat, kannst Du anhand Deiner ausgedruckten Dateiliste die entsprechenden Dateien ausfindig machen (beim „Druck“ in eine Datei hilft auch die Funktion **Versionsvergleich** von Word) und per Hand löschen. Um eine solche Dateiliste an der Eingabeaufforderung auszu-drucken, mußt Du in das entsprechende Verzeichnis wechseln und den Befehl **Dir > LPT1** eingeben.

Der Explorer bietet den Ausdruck des Verzeichnisinhaltes leider nicht an; Windows läßt sich aber mit einer Batchdatei und einem Eintrag in der Registry zu diesem Komfort überreden.

Erstelle im Verzeichnis `C:\WINDOWS\COMMAND` eine neue Textdatei `PRINTDIR.BAT`, indem Du auf eine freie Stelle im `COMMAND`-Ordner mit der rechten Maustaste klickst und den Befehl **Neu, Textdatei** wählst. Es erscheint eine Fehlermeldung bei der Namengebung, da Du die Erweiterung von `TXT` auf `BAT` änderst. Klicke die Meldung mit **Ja** weg, und lade die Datei in den Editor.

Gib den Befehl **Dir > LPT1** ein, und speichere die Datei. Falls Du den Ordnerinhalt nicht zu Papier bringen möchtest, sondern ihn in eine Textdatei schreiben lassen willst, verwendest Du statt dessen **Dir > Filelist.txt**.

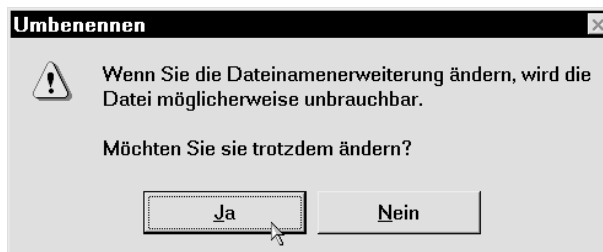


Abbildung 50: Diese Meldung erscheint beim Ändern der Endung von `TXT` auf `BAT`

Nun wechselst Du in den Registriereditor und suchst nach dem Schlüssel

`HKEY_CLASSES_ROOT\Directory\shell`. Klicke `\shell` mit der rechten Maustaste an und wähle **Neu, Schlüssel**. Den neuen Schlüssel nennst Du **Drucken Ordnerinhalt**, klickst ihn erneut mit der rechten Maustaste an, und wählst wieder **Neu, Schlüssel**. Vergib diesmal den Schlüsselnamen **Command**. Im rechten Bereich von `\Command` findest Du den Eintrag „Standard“, den Du bitte doppelt anklickst. Nun brauchst Du nur noch `c:\windows\command\printdir.bat` in das Dialogfenster einzugeben. Zukünftig kannst Du jeden Ordner mit der rechten Maustaste anklicken und aus dessen Kontextmenü den Befehl **Drucken Ordnerinhalt** auswählen. Du hast mit diesem Trick dem Explorer eine Funktion beigebracht, die man eigentlich schon aus alten DOS-Zeiten her kannte.

Kontextmenü von Start erweitern

Vielleicht ist Dir schon einmal aufgefallen, daß man den Startknopf mit der rechten Maustaste anklicken kann und dann ein Menü mit verschiedenen Befehlen erscheint. Je nach installierter Software können hier neben den Einträgen **Explorer** und **Öffnen** auch **Suchen**, **Suchen nach Viren** oder weitere vorhanden sein. Dieses Menü ist im Grund nur das Kontextmenü von Ordnern, denn dem Startmenü liegt der Ordner `C:\WINDOWS\STARTMENÜ` zugrunde. Wenn Du das Menü erweitern möchtest, beispielsweise mit dem Dateimanager, gilt diese Änderung gleichsam für alle Ordner-Kontextmenüs. Der Punkt, an dem Du in der Registry angreifen mußt, ist `HKEY_CLASSES_ROOT\Folder\shell`. Klicke `\shell` mit der rechten Maustaste an, wähle **Neu, Schlüssel** und gib dem neuen Schlüssel den Namen **Dateimanager**. Dann erstellst Du unter `\Dateimanager` einen Unterschlüssel namens **Command**. Im rechten Fensterteil von `\Command` findest Du wieder den Eintrag „Standard“, dem Du nach einem Doppelklick den Wert `c:\windows\winfile.exe` zuweist.

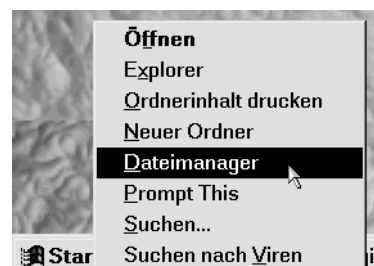


Abbildung 51: Das Kontextmenü des Startknopfes läßt sich mit dem Kontextmenü normaler Ordner vergleichen

Promptaufruf mit flexiblem Verzeichnis

Eine nervige Angewohnheit von Windows ist, daß man beim Start der MS-DOS-Eingabeaufforderung immer im Order `C:\WINDOWS\>` landet. Nun muß man zeitaufwendig mittels des Befehls `CD..` beziehungsweise `CD Verzeichnisname` in den gewünschten Ordner wechseln. Eine geniale Idee, die man auch fertig mit den PowerToys von Microsoft erhält: Wir erweitern das Kontextmenü aller Ordner um einen Befehl **Prompt DOS**, der direkt die Eingabeaufforderung (den sogenannten Prompt) startet und in das Verzeichnis wechselt, aus dessen Kontextmenü wir den Befehl wählen.

So geht's: Unter `HKEY_CLASSES_ROOT\Folder\shell` erstellst Du über **Neu, Schlüssel** den Unterschlüssel **Prompt DOS**, der wiederum einen Unterschlüssel **Command** enthalten muß. Nachdem Du `\Command` erstellt hast, klickst Du die Zeichenfolge „Standard“ doppelt an und gibst als Befehl / Wert **Command.com** ein. Nun hast Du in jedem Ordner-Kontextmenü den nützlichen Befehl **Prompt DOS**.

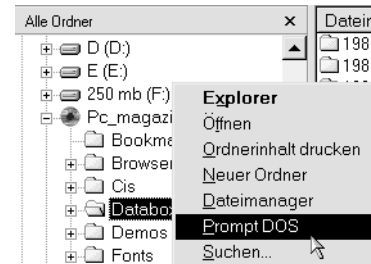


Abbildung 52: **Prompt DOS** aus dem Kontextmenü von `J:\DATABOX...`



Abbildung 53: ... und schon sind wir an der Eingabeaufforderung im richtigen Verzeichnis

Baukasten für individuellen DOS-Modus

Windows bietet die Möglichkeit, jedem DOS-Programm einen eigenen MS-DOS-Modus mit variablen Einstellungen zuzuweisen. Das ist nützlich, wenn beispielsweise DOS-Spiele mit den Standardeinstellungen nicht klar kommen oder Du spezielle Treiber laden mußt. Um einen solchen individuellen MS-DOS-Modus einzurichten, klickst Du die Programmdatei mit der Endung

EXE mit der rechten Maustaste an und wählst **Eigenschaften, Programm, Erweitert**. Aktiviere die Option **MS-DOS-Modus**. Zum Zuweisen einer speziellen Konfiguration mußt Du die Option **Neue MS-DOS-Konfiguration angeben** wählen. Du siehst nun Standardvorschläge von Windows für die spezielle **AUTOEXEC.BAT** und **CONFIG.SYS**. Weitere Optionen eröffnen sich nach einem Klick auf die Schaltfläche **Konfiguration**.

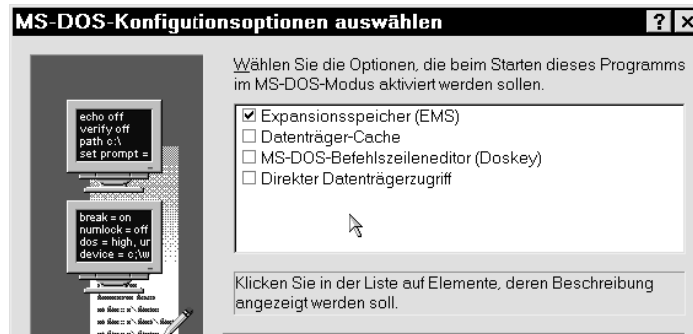


Abbildung 54: Hier fehlen einige wichtige Optionen

Viele Spiele benötigen Expansionspeicher, deshalb solltest Du den bereits aktivierten EMM386-Treiber auf jeden Fall in die Konfiguration nehmen. Außerdem empfiehlt sich die Integration von SmartDrive, das durch Installation eines Cachesystems Zugriffe auf die Festplatte beschleunigt. Die hier angebotenen Konfigurationsoptionen sind jedoch bei weitem nicht alles, was Windows zu bieten hat. Es gibt weitere Einstellmöglichkeiten, die allerdings standardmäßig in der Registry deaktiviert sind. Um das zu ändern, wechselst Du zum Schlüssel **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\MS-DOSOptions**.

Die dort vorhandenen Unterschlüssel zeigen neben dem Expansionspeicher (**EMS**) und Smartdrive (**Smartdrv**) auch Elemente wie **CD-ROM** oder **Mouse**, die im Einstellungsfenster nicht sichtbar waren. Ob ein Element angezeigt wird oder nicht, hängt von dem Wert des Eintrags „Flags“ ab. Klickst Du einmal den Unterschlüssel **Smartdrv** an, siehst Du im rechten Teil den Eintrag „Flags“ mit dem Wert „02 00 00 00“. Die 02 bedeutet, daß die Option im Konfigurationsfenster angezeigt, jedoch standardmäßig nicht aktiviert wird. Unter **EMS** ist der Wert hingegen „1b 00 00 00“. Das bedeutet, daß der Eintrag angezeigt und auch gleich aktiviert wird.

CD-ROM-Unterstützung für DOS

DOS-Spiele brauchen Zugriff auf das CD-ROM-Laufwerk; also mußt Du den CD-ROM-Treiber für DOS in die Konfiguration integrieren. Klicke **MS-DOSOptions\CD-ROM** an und ändere im rechten Teil den Wert von „Flags“ auf **02 00 00 00**. Nach einem Rechtsklick auf die Programmdatei und **Eigenschaften, Programm, Erweitert, Konfiguration** ist die Liste um den Eintrag „CD-ROM“ erweitert. Das Setzen eines Häkchens vor die Option bringt allerdings noch nichts, da Windows gar nicht weiß, wie der CD-ROM-Treiber und Ladebefehl lautet, der bei Aktivierung in eine der Startdateien eingefügt wird. Dafür klickst Du **CD-ROM** mit der rechten Maustaste an und wählst **Neu, Zeichenfolge**. Vergib als Namen **CONFIG.SYS**. Doppelklicke auf den neu erstellten Eintrag, und gib die Befehlszeile bzw. Ladeanweisung ein. Die Zeile hängt vom CD-ROM-Laufwerk ab, denn jeder Hersteller nennt die Treiberdatei anders. Deshalb hier nur ein Beispiel:

DEVICEHIGH=C:\CDROM\MTMCDAL.SYS /D:MTMIDE01

Die Treiberdatei dieses Mitsumi-Laufwerkes heißt **MTMCDAL.SYS** und liegt im Verzeichnis **C:\CDROM**. Dem Laufwerk wird der Name **MTMIDE01** zugewiesen. Das ist später wichtig. Die Treiberdatei sollte auf einer dem CD-ROM beigelegten Diskette vorhanden sein. Über einen weiteren Rechtsklick auf **\CD-ROM** und **Neu, Zeichenfolge** erstellst Du den Eintrag **AUTOEXEC.BAT** und doppelklickst darauf. Als Zeile für die **AUTOEXEC.BAT** muß die Ladeanweisung für MSCDEX angegeben werden, die beispielsweise

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

lauten kann. Entscheidend ist, daß sowohl in **AUTOEXEC.BAT** als auch **CONFIG.SYS** jeweils der gleiche Name hinter **/D:** angegeben wird, im Beispiel **MTMIDE01**.

Name	Wert
(Standard)	"CD-ROM"
Autoexec.bat	"LH C:\WINDOWS\COMMAND\mscdex.exe /D:IDECD001 /M:10"
Config.sys	"DEVICEHIGH=C:\WINDOWS\CDROM\aoatapi.sys /D:IDECD001 /P:1F0,14 ..."
Flags	02 00 00 00
Order	00 00 00 17
StdOption	04 00 00 00
TipText	"Ermöglicht Programmen für MS-DOS den direkten Zugriff auf CD-ROM-Lauf..."

Abbildung 55: Dieser Treiber heißt AOATAPISYS

Wenn Du nun bei der Konfiguration eines DOS-Spiels die CD-ROM-Unterstützung durch ein Häkchen aktiviert hast, werden die eben angegebenen Befehlszeilen in die Startdateien eingefügt. Damit hast Du Dir durch einen einmaligen Eingriff die Konfiguration aller DOS-Anwendungen zukünftig entscheidend erleichtert.

Maus-Unterstützung für DOS

Ebenso unverzichtbar ist meist die Maus-Unterstützung. Die von Windows versteckte Konfigurationsoption bringst Du folgendermaßen ans Licht: Ändere im rechten Bereich von **\MS-DOSOptions\Mouse** den Wert von „Flags“ in „2“. Nun fehlt nur noch die Befehlszeile mit dem passenden Maustreiber. Dieser variiert je nach Maus. Hat er die Endung **SYS**, also beispielsweise **MOUSE.SYS**, muß der Aufruf aus der **CONFIG.SYS** erfolgen. Nach einem Rechtsklick auf **\Mouse** und **Neu, Zeichenfolge** erstellst Du den Eintrag **Config.sys**. Nach dem Doppelklick darauf gibst Du den Befehl zum Aufruf des Maustreibers ein, beispielsweise

Name	Wert
(Standard)	"Maus"
Autoexec.bat	"C:\lmaus.com"
Flags	0x00000002 (2)
Order	00 00 00 12
StdOption	01 00 00 00
TipText	"Stellt Mausunterstützung für Programme für..."

Abbildung 56: Mein Maustreiber heißt Imaus.com

DEVICEHIGH=C:\TREIBER\MOUSE.SYS.

Hat Dein Maustreiber jedoch die Endung **COM**, muß er über die **AUTOEXEC.BAT** geladen werden. Dazu wählst Du auch **Neu, Zeichenfolge**, vergibst jedoch als Namen **Autoexec.bat**. Der Wert muß dann in der Form

LH C:\TREIBER\MOUSE.COM

vorliegen, wobei Du Pfad und Dateinamen natürlich anpassen mußt.

Soundkartenunterstützung für DOS

Name	Wert
(Standard)	"Soundkartenunterstützung"
Config.sys	"DEVICE=C:\WINDOWS\MZTINIT.SYS /A220 /
Flags	02 00 00 00
Order	50 00 00 00
TipText	"Lädt Soundkartentreiber"

Abb 57: Ohne Soundkarte machen Spiele doch nur halb so viel Spaß

Nachdem wir zwei ungenutzte Einträge sichtbar gemacht haben, erstellen wir nun für den obligatorischen Sound einen gänzlich neuen Eintrag. Klicke den Schlüssel **MS-DOSOptions** mit der rechten Maustaste an, wähle **Neu, Schlüssel** und gib diesem den Namen **Soundkarte**.

Weise im rechten Teil dem Eintrag „Standard“ nach einem Doppelklick den Wert **Soundkartenunterstützung** zu. Über einen Rechtsklick auf **Soundkarte** und **Neu, Zeichenfolge** erstellst Du den Eintrag **TipText**. Doppelklicke darauf und gib eine Beschreibung ein, zum Beispiel: **Lädt Soundkartentreiber**. Als nächstes erstellst Du über **Neu, Binärwert** einen Eintrag namens „Flags“, dem Du den Wert **02 00 00 00** zuweist. Analog legst Du einen Binärwert „Order“ an und weist **50 00 00 00** zu. Fehlt zu guter Letzt nur noch die Befehlszeile zum Laden des Treibers: Erstelle durch Rechtsklick auf **Soundkarte** und Wahl von **Neu, Zeichenfolge** den Eintrag **CONFIG.SYS**. Nun mußt Du wissen, wie der Soundkartentreiber heißt, wo Du ihn gespeichert hast und wie der korrekte Aufruf lautet. Hier hilft oft ein Blick in das Handbuch. Vielfach finden sich auf der Treiberdiskette oder -CD auch mehrere Text- und Hilfedateien, die Rat geben. Hier ein beispielhafter Wert:

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

Hast Du das Prinzip einmal verstanden, kannst Du den Konfigurationsdialog beliebig ausbauen. Auf jeden Fall hast Du nun bei der Konfiguration von DOS-Anwendungen sofort alle wichtigen Optionen wie CD-ROM, Soundkarte und Maus mit einem einfachen Klick zur Verfügung.

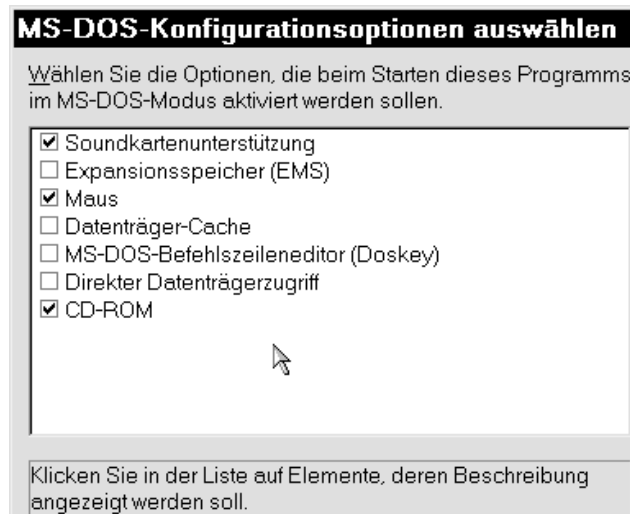


Abbildung 58: Wenn das nicht ein cooler Eingriff war...

Musikunterricht für Deine Programme

Windows bietet Dir die Möglichkeit, Programmereignisse wie beispielsweise das Starten des Explorers mit Klängen zu unterlegen. Die Einstellungen dazu findest Du unter **Systemsteuerung, Akustische Signale**. Standardmäßig werden nur ganz wenige Programme unterstützt, was Du aber über die Registry ändern kannst. Wie wär's etwa mit einer akustischen Begrüßung beim Start von Winword?

Rufe den Registriereditor auf, und wechsele zum **HKEY_CURRENT_USER\AppData**. Dieser Name steht für den englischen Ausdruck *ApplicationEvents*, was man frei in programmbezogene Ereignisse übersetzen kann. Der Unterschlüssel **\EventLabels** beschreibt alle mit Klängen unterlegbaren Ereignisse und weist ihnen einen Namen zu. So findet man dort beispielsweise Ereignisse wie „Open“ und „Close“, also das Öffnen und Schließen eines Programmes. Eine genaue Beschreibung der einzelnen möglichen Ereignisse findest Du in der Tabelle.

EREIGNISNAME	ERKLÄRUNG
.Default	Standardsignal
AppGPFault	Programmfehler
Close	Programm schließen
EmptyRecycleBin	Leerer Papierkorb
Maximize	Programm maximieren
MenuCommand	Menübefehl, beispielsweise Datei, Öffnen
MenuPopup	Aufklappen eines Menüs (Menüleiste oder Kontextmenü)
Minimize	Programm minimieren
Open	Programm öffnen
RestoreDown	Programm verkleinern
RestoreUp	Programm vergrößern
SystemAsterisk	Stern, Meldung von Windows 95
SystemExclamation	Hinweis
SystemExit	Windows beenden
SystemHand	Kritischer Abbruch
SystemQuestion	Frage
SystemStart	Windows starten

Tabelle 1: Diese Ereignisse kennt Windows

Der Unterschlüssel **\AppEvents\Schemes** untergliedert sich in **\Apps** und in **\Names**. Letzterer Schlüssel verwaltet die einzelnen Klangschemas von Windows. Standardmäßig verfügt Windows über „Jungle-Audioschema“, „Musica-Audioschema“, „Robotz-Audioschema“ und „Utopia-Audioschema“. Solltest Du diese beim Setup von Windows nicht installiert haben, kannst Du dies über **Systemsteuerung, Software, Windows-Setup, Multimedia** nachholen.

Falls Du keine oder nicht alle Schemata installiert hast, ist das auch nicht so schlimm, denn wir werden später ein eigenes Klangschema erstellen. Der Schlüssel **\Apps** enthält die genauen Einstellungen dafür, welche Klänge bei welchem Programmereignis gespielt werden sollen. Hier müssen wir ansetzen, wenn wir die ziemlich mickrige Liste von möglichen Programmen und Ereignissen erweitern wollen.

Das Prinzip sei am bestehenden Unterschlüssel **\Sndrec32** erklärt. Der Name steht für die Programmdatei **SNDREC32.EXE** des Audiorecorders von Windows. Es sind dort standardmäßig zwei Unterschlüssel **\Open** und **\Close** vorhanden. Klickst Du einen dieser Unterschlüssel an, siehst Du eine weitere Untergliederung in die einzelnen Klangschemas. Hast Du keine Klangschemas installiert, befindet sich hier nur der Standardeintrag **\Current**. Im rechten

Bereich könntest Du unter „Standard“ die abzuspielende **WAV**-Datei angeben, beispielsweise **C:\SOUNDS\OPEN.WAV** – das läßt sich über die Systemsteuerung nachher aber leichter erledigen.

Beispiel: Dem Editor Klänge beibringen

So wie es für den Audiorecorder die vertonbaren Ereignisse „Open“ und „Close“ standardmäßig schon gibt, erstellen wir diese nun nach gleichem Muster für den Editor. Klicke den Schlüssel **\Apps** rechts an, und wähle **Neu, Schlüssel**. Benenne den neuen Schlüssel als **Notepad**, denn die Programmdatei des Editors heißt **NOTEPAD.EXE**. Dann klickst Du **\Notepad** mit der rechten Maustaste an, wählst erneut **Neu, Schlüssel** mit dem Namen **Open**. Auf gleiche Weise erstellst Du auch einen Unterschlüssel **Close**.

Unter **Systemsteuerung, Akustische Signale** findest Du nun den Editor **Notepad**, der als unterstützte Ereignisse die beiden Einträge **Programm öffnen** und **Programm schließen** anbietet. Nun kannst Du unter Name eine **WAV**-Datei angeben, die Du entweder aus dem Listefeld oder nach einem Klick auf die Schaltfläche **Durchsuchen** auswählst.

Wenn Du das Prinzip einmal verstanden hast, läßt sich die Ereignisliste beliebig erweitern. Du kannst unter **HKEY_CURRENT_USER\AppData\Schemes\Apps** beispielsweise den Unterschlüssel **\Winword** anlegen, der dann wiederum die Unterschlüssel **\Open**, **\Close** oder auch ein anderes Ereignis wie beispielsweise **\Minimize** enthält. Konsultiere dazu die obige Tabelle; sie enthält alle möglichen Ereignisse.

Dabei mußt Du jedoch zwei Dinge beachten: Ereignisse wie beispielsweise Systemstart kannst Du natürlich nicht einem Programm wie Winword zuweisen – dieses Ereignis kennzeichnet nämlich den Start von Windows.

Wissen solltest Du außerdem, daß sich bestimmte Ereignisse überschneiden können. Verwendest Du zum Beispiel die Ereignisse „MenuPopup“ (Aufklappen eines Menüs) und „MenuCommand“ (Ausführen eines Menübefehls), wird nach dem Klick auf ein Menü ein Klang ausgegeben, beim Klicken auf einen Menübefehl ein anderer.

Ist der erste Sound zu lang, kann der zweite nicht rechtzeitig abgespielt werden und fällt deshalb aus. Dies sollte man bei der Einrichtung der akustischen Signale bedenken.

Hast Du alle Klänge nach Deinen Wünschen auf verschiedene Programme und Ereignisse verteilt, kannst Du das Ganze noch unter einem Namen, einem sogenannten Schema, abspeichern. Probiere einfach ein wenig; bald hast Du lauter „musikalische“ Programme.

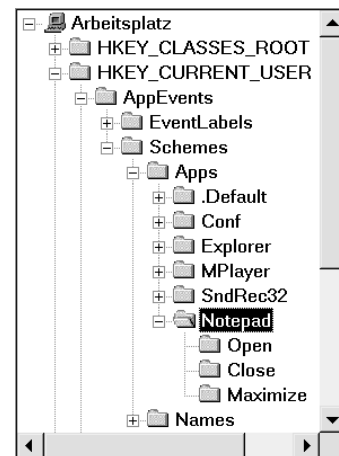


Abb 59: Einige neue Schlüssel machen Windows musikalisch

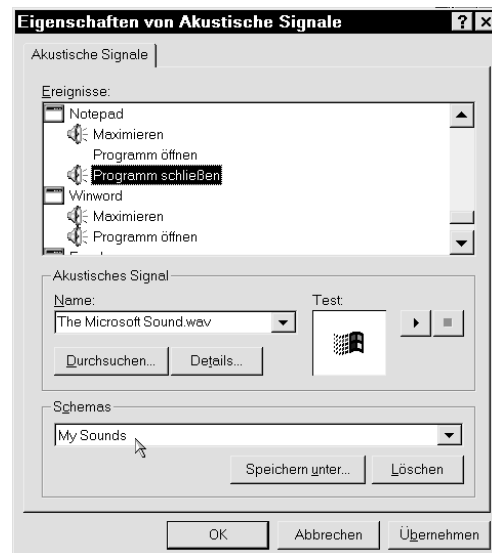


Abbildung 60: Nun hörst Du endlich mal was von Windows

Sonstiges

Unbekannte Dateitypen mit Editor öffnen

Wenn Du nicht registrierte Dateitypen standardmäßig mit dem Editor öffnen willst, mußt Du ebenfalls einen Eingriff in die Registry vornehmen. Rufe den Registriereditor auf, klicke **HKEY_CLASSES_ROOT\Unknown\shell** mit der rechten Maustaste an, und wähle **Neu, Schlüssel**. Nenne den neuen Schlüssel **open**, und führe auch hier einen Rechtsklick und **Neu, Schlüssel** aus. Nenne den so erstellten Unterschlüssel **command**. Nach einem Klick auf **\command** siehst Du im rechten Editorfenster den Eintrag „Standard“, den Du doppelt anklickst. Trage den Wert **Notepad.exe %1** ein. Zukünftig werden alle unbekanntes Dateitypen erst einmal mit dem Editor geöffnet.

Willkommensbildschirm mit eigenen Tips

Standardmäßig präsentiert Windows bei jedem Start einen Willkommensbildschirm, der Dir jedes Mal einen anderen Tip zum Umgang mit dem Betriebssystem gibt. Hast Du diese Funktion jedoch durch das Entfernen des Häkchens vor der Option **Dieses Dialogfeld bei jedem Start von Windows anzeigen** deaktiviert, ist der Ratgeber ein für alle Mal verschwunden. Mit einem Eingriff in die Registry kannst Du ihn jedoch wieder auf den Bildschirm zaubern. Wechsele dafür zum Schlüssel **HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Tips**, und ändere den Binärwert „Show“ von „00 00 00 00“ zu „01 00 00 00“.

Eigene oder angepaßte Tips

Es besteht die Möglichkeit, die Tagetips aus dem Willkommensbildschirm an persönliche Wünsche anzupassen oder sogar neue Einträge hinzuzufügen. Damit ist es zum Beispiel möglich, eine komplette Sprüchesammlung anzulegen, so daß man bei jedem Start von Windows mit einem lockeren Spruch begrüßt wird. Suche im Registriereditor den Schlüssel **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\explorer\Tips**. Klickst Du eine Zahl im rechten Fenster doppelt an, kannst Du im erscheinenden Dialog den Tagetip entsprechend ändern.

Du kannst den Schlüssel **\Tips** auch mit der rechten Maustaste anklicken und dann aus dem Kontextmenü die Befehlsfolge **Neu, Zeichenfolge** auswählen, um mit einer neuen Nummer einen völlig neuen Tip zu erstellen.

Tools rund um die Registry

Die Angst vor Systemeingriffen oder auch die fehlende Kenntnis vieler Anwender hat zahlreiche Shareware- und Freeware-Autoren dazu veranlaßt, zahlreiche Programme und Tools rund um die Registry zu entwerfen. Vorteilhaft für Dich ist, daß Du die Programme kostenlos von Shareware-CDs oder auch aus Mailboxen und dem Internet beziehen kannst. Damit dieses Softwaremodell jedoch nicht mißbraucht wird, erkläre ich Dir im folgenden einmal genau, was Shareware und Freeware ist.

Was ist Shareware, was ist Freeware?

Das Prinzip von Shareware ist einfach: Du erhältst ein Programm über eine der oben genannten Quellen und installierst es. Dann hast Du eine Testzeit (meist 30 Tage). Bist Du zufrieden, mußt Du eine Registriergebühr bezahlen, welche meist zwischen 10 und 70 DM liegt.

Oft sind in die Testversionen Meldungen eingebaut, die den Anwender an die Registrierung erinnern. Leider bauen einige Sharewareautoren so viele Hinweise und Behinderungen ein, daß das Programm nicht mehr sinnvoll nutzbar ist. Das Problem: Das Testen eines Programmes vor dem Kauf ist für den Anwender ein klarer Vorteil, denn niemand kauft gern die Katze im Sack. Der Programmierer hat jedoch meist große Mühe, seinen in den meisten Fällen wohlverdienten Obolus einzuziehen. Also: unterstütze das Shareware-Prinzip und laß Dich bei längerer Nutzung eines Tools registrieren.

Freeware mußt Du im Gegensatz zu Shareware nicht registrieren lassen. Du erhältst Freeware ebenso kostenlos wie Shareware von CDs oder aus dem World Wide Web und kannst sie beliebig nutzen, ohne dafür auch nur einen Pfennig zu zahlen. Ebenso wie bei Shareware kannst Du das Programm völlig legal für Freunde und Bekannte kopieren und weitergeben.

Shareware und Freeware im Internet

Shareware und Freeware hat man früher über Disketten und CD-ROMs bekommen. Heute werden diese Medien zunehmend durch das Internet abgelöst, weshalb ich für Dich einige Links recherchiert habe, wo Du ohne langes Suchen Zugriff auf viele Programme hast:

www.filepile.com

www.shareware.com

www.download.com

www.freeware.de

Außerdem kann man jedem Windows-Anwender die Adresse www.winfiles.com empfehlen, denn hier findest Du alles rund um Windows.



Abbildung 61: Viele Hilfen und Programme zu Windows

TweakUI - Microsofts Trickkiste

TweakUI ist eines der Microsoft Powertoys, die kostenlos sowohl für Windows 95 als auch 98 erhältlich sind. Windows 98-Nutzer haben es dabei aber besser: Hier ist das Programm erstens direkt auf der Windows-CD im Verzeichnis

`\TOOLS\RESKIT\POWERTOY`

vorhanden; außerdem ist die 98er-Version noch wesentlich verbessert worden. Das Programm besteht

lediglich aus der Datei `TWEAKUI.INF`,

die Du nur mit der rechten Maustaste anklicken und dann **Installieren** wählen mußt. Das Tool wird daraufhin in die Systemsteuerung integriert, wo man in thematisch sortierten Registern versteckte Einstellungen anpassen kann.

Neben grundlegenden Einstellungen wie der Aufklappgeschwindigkeit von Menüs und Einstellungen zum Bootmenü kannst Du unter Windows 98 zum Beispiel festlegen, daß bestimmte Laufwerke aus dem Explorer und ausgewählte Elemente der Systemsteuerung ausgeblendet werden.

Auch die lästige Eigenheit von Windows, nach jedem Absturz Scandisk auszuführen, läßt sich abschalten.

Allerdings solltest Du dann von Zeit zu Zeit das Programm manuell starten.

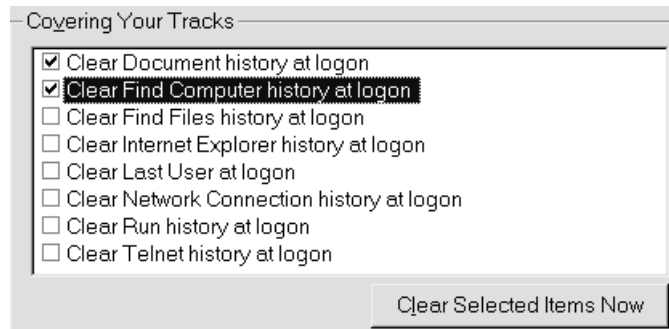


Abbildung 62: Automatische Spurenbeseitigung

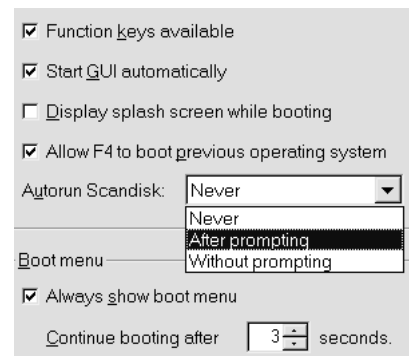


Abbildung 63: Boot-Optionen unter Kontrolle

Als besonderes Schmankerl verfügt TweakUI für Windows 98 über verschiedene Reparaturmechanismen, um kleinere Probleme elegant und schnell aus der Welt zu schaffen. Alles in allem also ein Tool, daß insbesondere Anwender der neuen Windows-Version nicht auf ihrer CD dahinsiechen lassen sollten.

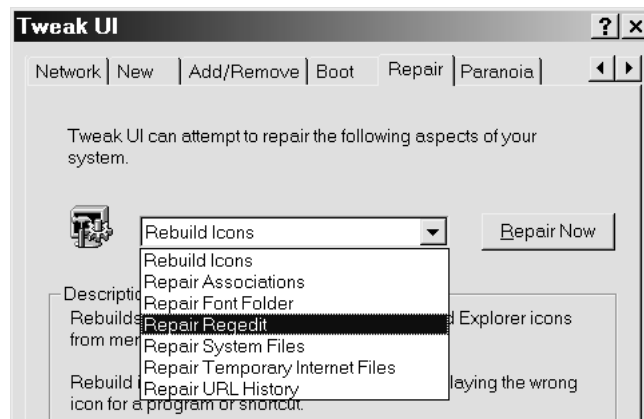


Abbildung 64: Windows-Reparatur leichtgemacht

Scanreg – Der Registryprüfer von Windows 98

Als eines der vielen neuen Tools von Windows 98 verfügt das Betriebssystem nun über ein Programm namens „Scanreg“, mit dem Du die Registry auf Fehler überprüfen und automatisch optimieren lassen kannst. Auch wenn Du das Programm noch nie gestartet hast, arbeitet es trotzdem beim jeweils ersten Booten eines Tages, indem es eine Sicherungskopie von **SYSTEM.DAT**, **USER.DAT**, **WIN.INI** und **SYSTEM.INI** für Dich anlegt. Das geschieht allerdings nicht wie bei Windows 95, wo die Sicherung aus **USER.DA0** und **SYSTEM.DA0** im Windows-Verzeichnis besteht (falls Du die Dateien nicht siehst, mußt Du **Explorer, Ansicht, Optionen, Alle Dateien anzeigen** wählen).

Windows 98 mit Scanreg legt die beiden Registry-Dateien samt **WIN.INI** und **SYSTEM.INI** komprimiert als **RB000.CAB** bis **RB005.CAB** im Ordner **C:\WINDOWS\SYSBCKUP** ab. Damit hast Du automatisch immer Sicherungskopien der letzten fünf Tage, so daß Du das Programm über **Start, Ausführen, Scanregw** (das **w** steht für die **Windows**-Version) nur aufzurufen brauchst, wenn

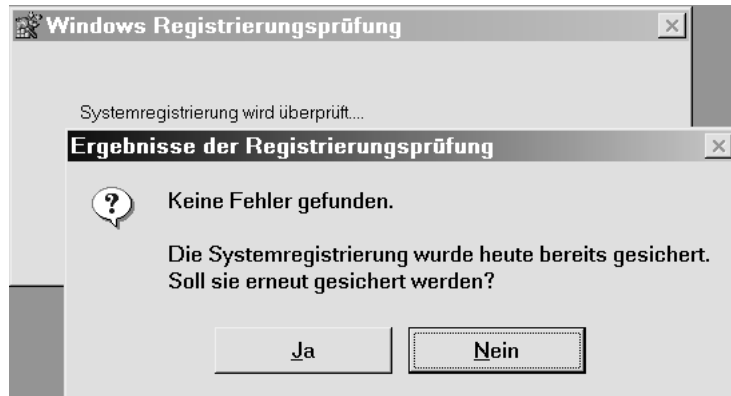


Abbildung 65: Zum Glück kein Fehler...

Du zwischenzeitlich eine Fehlerüberprüfung vornehmen möchtest. Nach der Überprüfung erscheint dann auch ein Dialog mit der Meldung, daß heute schon eine Sicherung vorgenommen wurde. Falls gewünscht, kannst Du aber auch noch eine weitere erstellen.

Bevor Windows beim Booten übrigens die Sicherungskopie anlegt, überprüft Scanreg die Registry. Findet es durch das Löschen von alten Einträgen ungenutzten, aber nicht freigegebenen Platz, optimiert es die Registry selbständig dahingehend.

Übrigens ist **CAB** das Standardformat von Windows-Installationsdateien (siehe CD), das heißt Windows 98 kommt damit gut zurecht. Du kannst per Doppelklick problemlos in die Archive schauen und Dateien wieder extrahieren, ohne dafür einen Packer wie Winzip zu verwenden. Unter DOS brauchst Du für das Entpacken von **CAB**-Dateien normalerweise den Befehl **Extract**, allerdings geht es im Fall der Registrysicherung auch einfacher: Starte im MS-DOS-Modus und gib im Windows-Verzeichnis **Scanreg /Restore** ein. Alternativ verwendest Du **Scanreg /Fix**, um Fehler zu beheben. Falls Du einmal von DOS aus eine Sicherung machen willst, gibst Du einfach **Scanreg /Backup** ein.

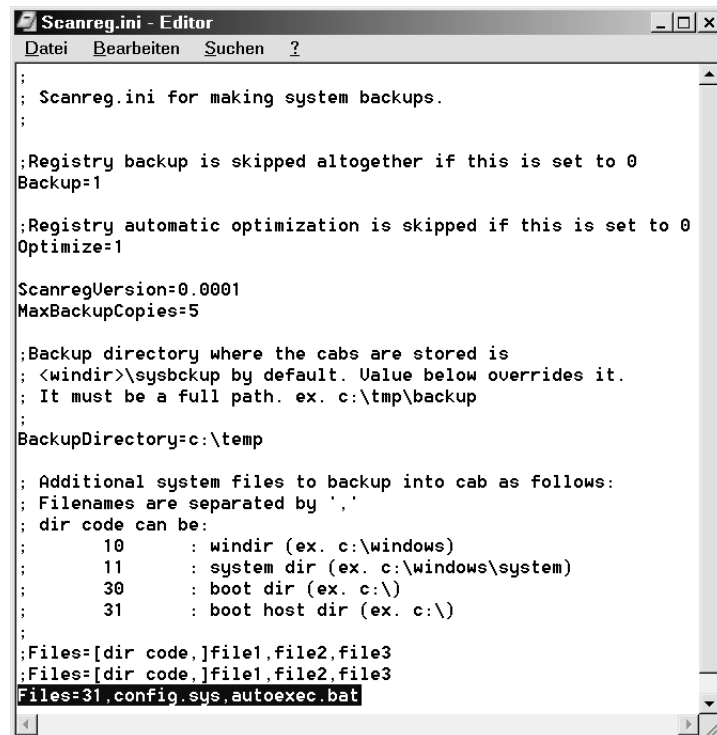
Scanreg anpassen und ausbauen

Scanreg hat sogar einige weitere, versteckte Features auf Lager: Das wichtigste ist sicherlich die Möglichkeit, weitere Dateien beim Start gleich in die Sicherung einzubeziehen. Dazu mußt Du die Datei **SCANREG.INI** im **WINDOWS**-Ordner in den Editor laden.

Wenn Du beispielsweise in das CAB-Archiv beim Booten außer der Registry-Daten auch **AUTOEXEC.BAT**, **CONFIG.SYS** und **MSDOS.SYS** sichern willst, schreibst Du in die letzte Zeile:

Files=31,config.sys,autoexec.bat

Bist Du der Meinung, Du brauchst die automatische Sicherung nicht, kannst Du den Wert **BACKUP=1** in **0** ändern. Das empfehle ich Dir aber nicht, es sei denn, Du möchtest beim Start jede mögliche Sekunde einsparen, weil beispielsweise Dein PC bei ankommenden Faxen besonders schnell zur Verfügung stehen muß. Über **MaxBackupCopies=x** stellst Du die Anzahl der Sicherungskopien ein, bis die erste wieder überschrieben wird. Last but not least läßt sich über **BackupDirectory=** auch ein anderes Verzeichnis als das standardmäßige **C:\WINDOWS\SYSBCKUP** definieren.



```

Scanreg.ini - Editor
Datei Bearbeiten Suchen ?
:
; Scanreg.ini for making system backups.
;
;Registry backup is skipped altogether if this is set to 0
Backup=1
;Registry automatic optimization is skipped if this is set to 0
Optimize=1
ScanregVersion=0.0001
MaxBackupCopies=5
;Backup directory where the cabs are stored is
; <windir>\sysbckup by default. Value below overrides it.
; It must be a full path. ex. c:\tmp\backup
;
BackupDirectory=c:\temp
; Additional system files to backup into cab as follows:
; Filenames are separated by ','
; dir code can be:
;      10 : windir (ex. c:\windows)
;      11 : system dir (ex. c:\windows\system)
;      30 : boot dir (ex. c:\)
;      31 : boot host dir (ex. c:\)
;
;Files=[dir code,]file1,file2,file3
;Files=[dir code,]file1,file2,file3
Files=31,config.sys,autoexec.bat

```

Abbildung 66: Wie Du siehst, ist die INI-Datei weitgehend selbsterklärend

Registry Inkognito

Registry Inkognito ist das Programm schlechthin, wenn es um versteckte Optionen von Windows 95 geht. Es ist nur für Nutzer von Windows 95 interessant, denn es hat ungefähr die gleichen Features wie TweakUI von Windows 98. In verschiedenen, thematisch geordneten Registern findest Du alles, womit Du Dein Windows 95 an persönliche Vorlieben anpassen kannst. Beginnend bei der Deaktivierung des Logos beim Start über die Einstellung der Verzögerung im Startmenü bis hin zum Bereinigen der Ausführen-Liste des Startmenüs läßt sich hier alles nur erdenkliche einstellen. Die Krönung setzt der fiese Trick auf, einfach den Befehl Beenden aus dem Startmenü zu entfernen. Prost Mahlzeit...

Registry Inkognito ist Shareware und kostet für Privatanwender 26 DM Registriergebühr. Kommerzielle Anwender zahlen hingegen schon 56 DM. Zumindest für den Heimanwender ein fairer Preis für ein solches Programm, auch wenn Du als Leser dieses Heftes fast alle Einstellungen mit den vielen Tips & Tricks von Hand vornehmen kannst.

Mit Hilfe von Registry Inkognito kann man den Eingriff in die Registry umgehen, ohne dabei auf die versteckten Möglichkeiten verzichten zu müssen.

WinHacker 95 – Hacken für Anfänger

WinHacker ist ein nützliches Tool, mit dem man unglaublich viele Dinge realisieren und manipulieren kann: Verstecke einfach ausgewählte Laufwerke im Explorer, entferne die lästigen Pfeile auf den Verknüpfungen von Icons. Bringe Windows bei, daß die Iconunterschriften von Verknüpfungen nicht immer mit „Verknüpfung mit...“ anfangen sollen und so weiter. Genial ist auch die Möglichkeit, das Kontextmenü von Ordnern mit dem Eintrag **Explore from here** erweitern zu lassen. Klickst Du einen Ordner mit der rechten Maustaste an und wählst diesen Befehl, öffnet sich ein neues Explorerfenster mit diesem Ordner als oberstem Eintrag. Ganz ähnlich kannst Du das Kontextmenü um den Eintrag **MS-DOS-Prompt** erweitern lassen, der bei Auswahl eine DOS-Box mit dem angeklickten Ordner als aktives Verzeichnis aufruft. Sogar die Icons von Systemteilen wie dem Arbeitsplatz lassen sich ändern. Mit WinHacker 95 erhältst Du ein echtes Powerprogramm, das seine 17,95 US\$ Registrierungsgebühr wert ist.

Policy Editor - Systemrichtlinieneditor

Wenn Du mit verschiedenen Benutzerprofilen arbeitest, kann ich Dir den Policy Editor empfehlen. Dieses kleine Programm befindet sich auf Deiner Windows-95-CD im Verzeichnis `\ADMIN\APPTOOLS\POLEDIT`. Unter Windows 98 gilt diese Empfehlung natürlich auch, hier findest Du das Programm im CD-Verzeichnis `\TOOLS\RESKIT\NETADMIN\POLEDIT`.

Was sind Benutzerprofile und Systemrichtlinien?

Wenn beispielsweise der Vater seinem Sohn die Arbeit am PC erlauben möchte, dabei jedoch Folgen kindlicher Neugierde verhindern möchte, so kann er gewisse Teile von Windows sperren. Er erstellt also ein Benutzerprofil für seinen Sohn und legt die Systemrichtlinien fest. So kann er beispielsweise den Punkt **Anzeige** in der **Systemsteuerung** sperren, so daß Sohnemann nicht einfach die Einstellungen der Grafikkarte ändert. Auf diese Weise läßt sich der Zugriff auf das System je nach Anwender limitieren. Die verschiedenen Benutzerprofile werden durch den beim Einloggen angegebenen Namen ausgewählt und die zugehörigen Systemrichtlinien geladen.

ERU - Windows' kostenloses Sicherungstool

ERU steht abkürzend für das „Emergency Recovery Utility“ von Microsoft, das Du auf Deiner Win95-CD im Verzeichnis `\OTHER\MISC\ERU` findest. Dieses kleine Tool sichert für Dich systemrelevante Dateien in ein Verzeichnis oder auf eine Diskette, so daß Du diese im Notfall von dieser zurückspielen kannst. Unter Windows 98 gibt es ERU nicht mehr.

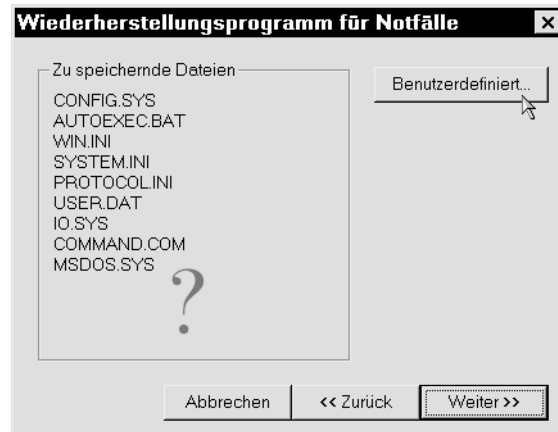



Abbildung 67: Hier fehlt doch was...

 ERU verlangt eine formatierte Systemdiskette, die aufgrund des durch die Systemdateien verbrauchten Speicherplatzes die Datei `SYSTEM.DAT` oft nicht mehr faßt. Das ist jedoch von Computer zu Computer unterschiedlich und hängt primär davon ab, wieviele Programme und was für Hardware Du installiert hast, die durch ihre Einträge die Registry aufblähen. Sichere diese Datei entweder auf eine zweite Diskette (eventuell vorher mit einem Packprogramm wie Winzip komprimieren) oder lege sie in einem anderen Verzeichnis der Festplatte ab.

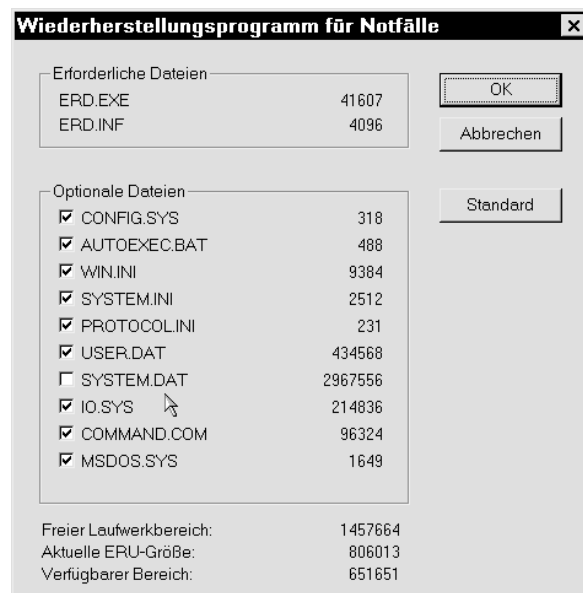


Abb 68: Die `SYSTEM.DAT` ist zu groß für die Diskette

RegClean – Der kostenlose Reinemacher

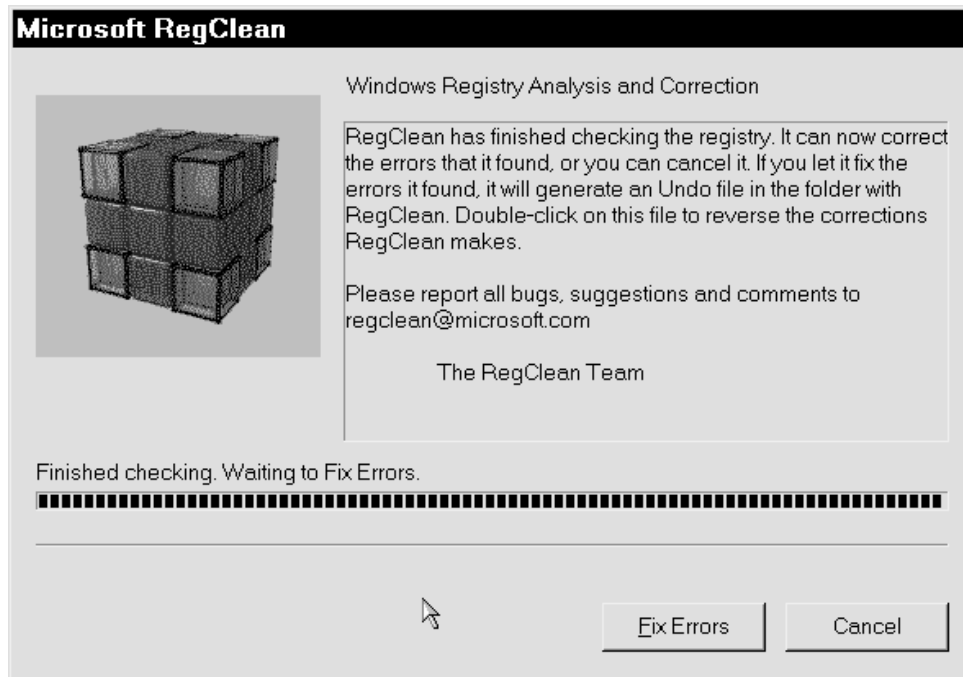


Abbildung 69: Schnelle und zuverlässige Reinigung

Durch das ständige Installieren und Deinstallieren von Anwendungen unter Windows kann sich die Registry mit unnötigen Einträgen ziemlich aufblähen, was Zeit beim Starten und Systemstabilität kostet. Mit RegClean stellt Microsoft dem Anwender ein Tool zur Verfügung, das die Registry automatisch nach verwaisten Einträgen absucht und diese entfernt. RegClean ist kostenlos im Internet erhältlich; entweder direkt auf der Web-Site von Microsoft www.microsoft.com oder auch in zahlreichen Softwarearchiven. Weiterhin kann man es auf fast allen Seiten finden, die sich mit Windows (Tools, Tips und Problemlösungen) beschäftigen.

Nach einem Doppelklick auf die Programmdatei werden alle relevanten Einstellungen eingelesen. Mit einem Klick auf **Fix Errors** (Fehler beheben) werden unnötige Einträge entfernt.

Im Programmverzeichnis wird dabei eine Undo-Datei erstellt, die auf Doppelklick die Änderungen bei Problemen rückgängig macht.

Unter Windows 98 funktioniert RegClean auch, allerdings solltest Du hier auch noch Scanreg in Betracht ziehen, wenn es um das Aufspüren von Fehlern geht. Unnütze Einträge findet hingegen nur RegClean.

Dateiname	Größe	Typ
loadist.exe	479 KB	Anwendung
README.txt	8 KB	Textdatei
RegClean.exe	818 KB	Anwendung
Undo AM 19980616 170633.Reg	0 KB	Registrier...

Abbildung 70: Die Undo-Datei mit Datum & Zeitangabe

Du kannst das Programm im Internet downloaden. Gehe dazu zu www.download.com und gib das Stichwort „regclean“ ein. Du bekommst einen aktuellen Link, von wo Du die aktuelle Version des Programms laden kannst.

Norton Registry Editor und Registry Tracker

Wenn Du Besitzer der Norton Utilities bist, verfügst Du über die beiden Programme „Norton Registry Editor“ und „Norton Registry Tracker“. Ersteres ist eine verbesserte Version von Microsofts Regedit, bei dem zu den wichtigsten Schlüsseln im unteren Bildschirmbereich erklärende Hilfetexte erscheinen. Das erleichtert Dir die Orientierung in der Registry ungemein. Hinzu kommen viele nützliche Features wie beispielsweise das Erstellen von Verknüpfungen innerhalb der Registry. So kannst Du beispielsweise zu einem in der Hierarchie sehr tief liegenden Schlüssel eine Verknüpfung auf der Ebene der Hauptschlüssel erstellen, so daß Du Dich nicht jedesmal ewig durch den „Schlüsseldschungel“ klicken mußt. Weiterhin ist der Registry Editor von Norton ein guter Editor für **INI**-Dateien. Die wichtigen (**WIN.INI**, **SYSTEM.INI** und **CONTROL.INI**) lädt das Programm automatisch, weitere lassen sich durch einmaliges Hinzufügen dauerhaft in den Editor einbinden. Das Bearbeiten erfolgt nach Registryschema: Die [Gruppen] werden als Unterschlüssel, die Einträge und Werte dann im rechten Bereich angezeigt.

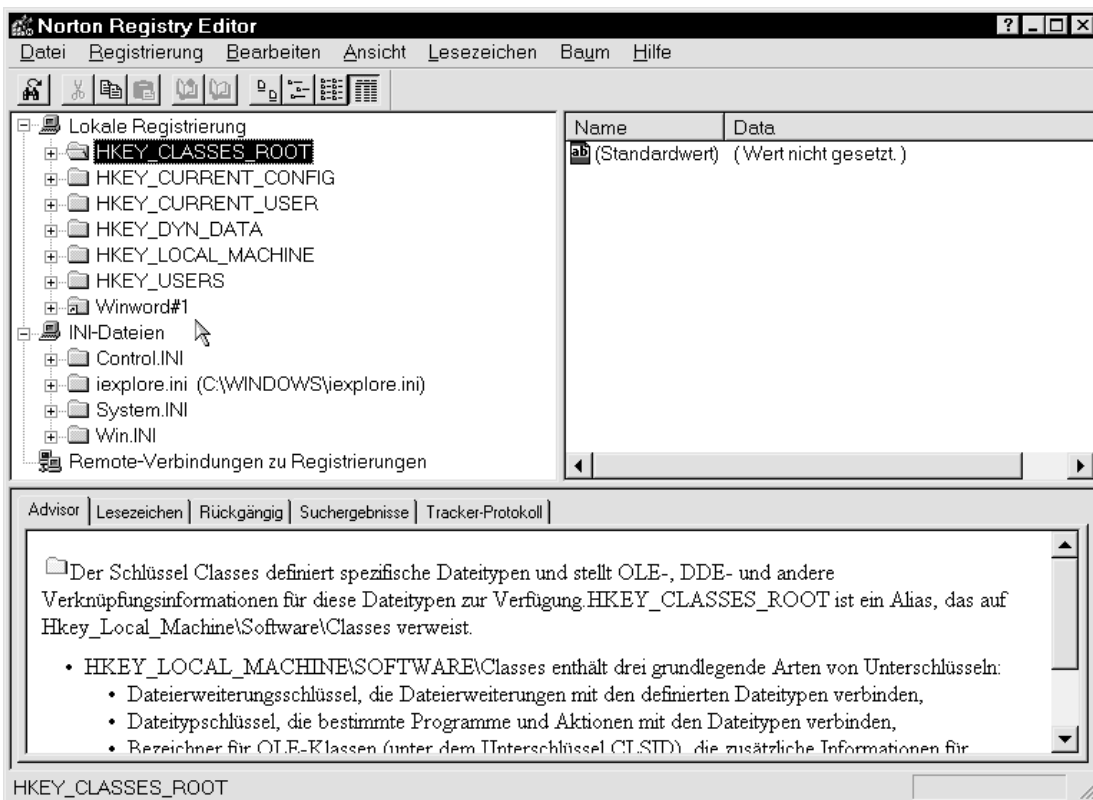


Abbildung 71: Norton Registry Editor - die bessere Alternative

Das zweite Programm, Norton Registry Tracker, protokolliert Änderungen an der Registry. Das ist zum einen bei Installationen nützlich, denn man kann Ursprungszustand und neue Version eines Schlüssels vergleichen. Auch für eigene Eingriffe ist das recht nützlich, falls man mal einen Fehler gemacht hat. Auch nützlich: Du kannst weitere Dateien wie **CONFIG.SYS** zur Überwachung angeben.

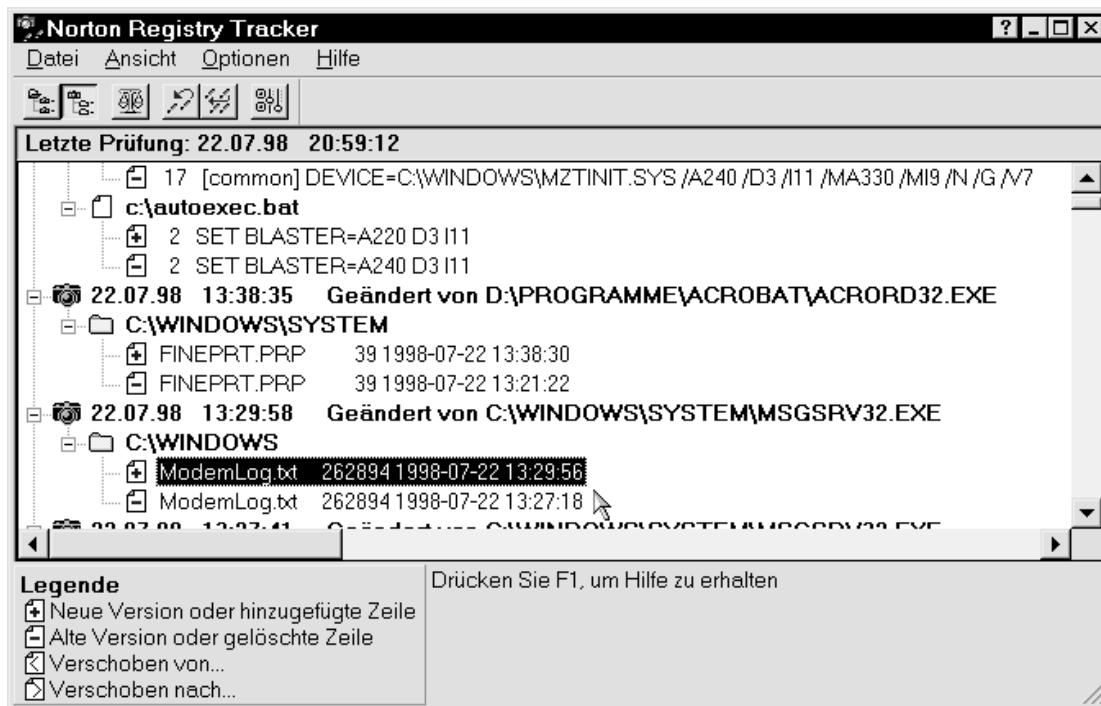


Abbildung 72: Der Registry-Tracker überwacht alle Änderungen

Zu guter Letzt

Ich hoffe, daß Du am Ende dieses Heftes etwas dazugelernt hast und Dein Windows jetzt viel besser an Deine Bedürfnisse angepaßt ist. Ich danke Dir für den Kauf und das Lesen dieses Heftes – dafür, daß Du so mutig warst, Dich mit den Tiefen von Windows, sprich der Registry, auseinanderzusetzen. Da Du als Leser das letzte und wichtigste Glied in der Kette aller Hefte bist, würde mich Deine Meinung interessieren. Schreib mir, wenn Du Lob oder Kritik, Anregungen und Verbesserungsvorschläge hast. Zum Zeitpunkt des Druckes lautet meine E-Mail-Adresse: andre.moritz@gmx.net

Falls Du einen Internetzugang besitzt und Lust hast, kannst Du uns auch im Internet unter www.knowware.dk/de besuchen. Dort gibt es Informationen zu neuen, alten und geplanten Heften, Online-Bestellmöglichkeiten, Tips & Tricks und vieles andere mehr. Viel Spaß!

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

- Abgesichert, Start, 26
- Abmelden, 34
- Animation, minimieren, 31
- Attribute, 22
- Audioschema, 44
- Auflösung, 31
- Autorun, 32
- Autostart, 32; 33
- Benutzername, 32
- Benutzerprofile, 8; 51
- Binärwert, 9
- Bootlog.txt, 25
- Bootmenü, 25; 26
- ClassIdentifier, 21
- CLSID, 20; 21; 33
- Dateiverständnis, 11
- Deinstallation, 29
- DWord, 9; 14; 18
- EditFlag, 34
- Einstellungen, speichern, 33
- ERU, 52
- Farbtiefe, 31
- Freeware, 47
- Hauptschlüssel
 - HKEY_CLASSES_ROOT, 11
 - HKEY_CURRENT_CONFIG, 13
 - HKEY_CURRENT_USER, 12
 - HKEY_DYN_DATA, 13
 - HKEY_LOCAL_MACHINE, 12
 - HKEY_USERS, 12
- Hilfdatei, einbinden, 5
- Hintergrundbild, Position, 30
- Icons, BMP-Vorschau, 29
- Icons, mehr Farben, 30
- INI-Datei, 5; 7
- Mscdex.exe, 24
- MS-DOS-Modus, 41
- Neu-Menü, 36
- OLE, 12; 20
- Papierkorb, umbenennen, 33
- PowerToys, 40
- Protokollieren, Start, 25
- RegClean, 53
- REG-Datei, 19
- REG-Dateien
 - Aufbau, 17
 - Bestätigung abschalten, 19
 - selber schreiben, 18
 - Umgang mit, 17
- Registriereditor
 - Shortcuts, 16
- Registrierung
 - Aufbau, 8
 - exportieren, importieren, 23
 - Lesezeichen, 15
 - Notizen, 16
 - sichern, 22
 - Suchen, 14
- Resource Kit, 5
- Scanreg, 49
- Setuppfade, 32
- Shareware, 47
- Sicherungskopien, 22
- Startdiskette, 24
- Startmenü, 28
- System.dat, 8; 10; 22; 52
- Systemeditor, 6; 26
- Systemmonitor, 13
- Systemrichtlinien, 51
- Tagestips, 46
- TweakUI, 48
- Uninstall, 29
- User.dat, 8; 10; 22
- vergleichen, Registrierungen, 19
- Versionsvergleich, 19; 38
- Virensan, 38
- Willkommensbildschirm, 46
- Zusammenführen, 17