WinHex 9.72

Allgemeines

Arbeiten mit dem Hex-Editor

Menüreferenz

Extras

WinHex 9.72

Autor: Stefan Fleischmann
E-Mail: mail@sf-soft.de

Programmiert und weiterentwickelt seit 1995, letzte Aktualisierung im Januar 2001.

Unterstützte Betriebssysteme:

- Windows 95/98/Me
- Windows NT 4.0
- Windows 2000

Die jeweils neueste Version dieses Programms finden Sie immer auf den Web-Sites http://www.sf-soft.de und http://www.winhex.de. Besuchen Sie auch das WinHex-Forum unter http://www.winhex.net.

Wertung der ZDNet Software-Library: 5 von 5 Punkten!

Zu den registrierten Benutzern gehören Universitäts- und nationale Forschungseinrichtungen (z. B. das Institut für Informatik der Technischen Universität München, die Technische Versuchs- und Forschungsanstalt der Technischen Universität Wien, das Oak Ridge National Laboratory in Tennessee, USA), Behörden wie die Kriminalpolizei Passau, diverse militärische Einrichtungen mehrerer Staaten sowie Unternehmen aus den verschiedensten Branchen, z. B. Toshiba Europe, Siemens Business Services, Mannesmann VDO AG, DePfa Deutsche Pfandbriefbank AG, Analytik Jena AG, INTERNOLIX AG (Abt. Entwicklung MINI MARKET), Ontrack Data International Inc., Novell Inc., Password Crackers Inc. Bestellen auch Sie die Vollversion!

Französische Übersetzung: Jérôme Broutin und Henri Pouzoullic, aktualisiert von Bernard Leprêtre
Spanische Übersetzung: José María Tagarro Martí
Italienische Übersetzung: Luca Cantarini
Portugiesische Übersetzung: Heyder Lino Ferreira
Kryptographische Beratung: Alexandre Pukall

Die Algorithmen Pukall Cipher 1 (PC 1) und Pukall Stream Cipher Hash Function wurden von Alexandre Pukall entwickelt. Quellcode erhältlich unter http://www.freecode.com, http://www.multimania.com/cuisinons/progs/ und unter http://www.multimania.com/ pc1/.

Der MD5 Message-Digest wurde entwickelt von RSA Data Security Inc.

Die „zlib"-Datenkompression mit den Algorithmen Deflate und Inflate wurde entwickelt von Jean-loup Gailly und Mark Adler. ftp://ftp.cdrom.com/pub/infozip/zlib/zlib.html

Werkzeug Hex-Editor

Ein Hexadezimal-Editor ist in der Lage, den Inhalt einer Datei jedes Typs vollständig anzuzeigen. Im Gegensatz zu einem Text-Editor kann er **alle** Bytes einer Datei darstellen, auch Steuerzeichen (für Zeilenumbruch, Tabulator usw.) und Programmcode, und zwar unter Angabe einer zweistelligen Zahl des Hexadezimalsystems (16er-System).

Ein Byte ist eine Kombination aus 8 Bits. Jedes Bit enthält entweder eine 0 oder eine 1, hat also einen von zwei möglichen Zuständen. Ein Byte kann daher einen von 2 hoch 8 (=256) verschiedenen Werten annehmen. Da 256 das Quadrat von 16 ist, kann jedes Byte durch eine zweistellige Zahl aus dem Hexadezimalsystem repräsentiert werden. Jede der beiden Stellen steht für eine Tetrade (auch: ein Nibble) eines Bytes, d. h. 4 Bits. Die möglichen Ziffern dabei sind 0-9 und A-F. Durch Änderung dieser Ziffern kann man einem Byte einem neuen Wert zuweisen.

Genauso ist es möglich, die Zeichen zu editieren, die jedem Byte zugeordnet sind (Textmodus, s. a. »Zeichen eingeben«). Diese Zeichen können z. B. Buchstaben oder Satzzeichen sein. Beispiel: Ein Byte, das den dezimalen Wert 65 hat, wird vom Hex-Editor in der Hexadezimal-Schreibweise mit 41 angeben (**4**•16+**1**=65) und in der Zeichenschreibweise mit dem Buchstaben »A«. Die Zuordnung von Zeichen gibt der »Zeichensatz« an.

Entscheidend beim Editieren einer Programmdatei (z. B. EXE-Datei) ist, daß nicht die Länge der Datei (die Anzahl der Bytes, die sie enthält) und damit die relativen Positionen von Programmcode und Daten verändert werden. Dies würde die Ausführbarkeit des Programmcodes beeinträchtigen. Es ist generell zu beachten, daß Änderungen an Dateiinhalten zu anormalen Verhaltensweise der zugehörigen Programme führen können. Für viele Zwecke genügt es, sich auf das Editieren des in einer Datei vorkommenden Textes beschränken. Es ist in jedem Fall ratsam, vor dem Bearbeiten eine Sicherung der Datei anzulegen.

Sie werden feststellen, daß WinHex vor der Benutzung aller entscheidenden Funktionen Sicherheitsabfragen durchführt, die Fehlbedienungen vorbeugen.

Bestellung

Die Lizenzgebühr hängt davon ab, ob Sie WinHex privat einsetzen oder in einem Unternehmen, einer Behörde oder einer sonstigen Organisation. Wenn Sie die Vollversion an mehr als einem Rechner betreiben möchten, benötigen Sie entsprechend zusätzliche Lizenzen.

Basislizenz:	EUR 31,90 / DM 62,39 (privat) EUR 56,90 / DM 111,29 (gewerblich/behördlich)
Zusätzliche Lizenzen:	je EUR 16,90 / DM 33,05 (privat) je EUR 29,90 / DM 58,48 (gewerblich/behördlich)

Alle Preise incl. 16% Mwst. bei Erwerb in Deutschland sowie bei Erwerb durch Privatpersonen in der EU. Wenn Sie in US-Dollar bezahlen möchten, beachten Sie bitte die englische Anleitung.

• Überweisen Sie die Lizenzgebühr auf das nachstehende Konto. Wenn Sie von außerhalb Deutschlands überweisen, addieren Sie bitte EUR 5 / DM 10 hinzu. *oder*
• Schicken Sie mir einen Scheck. Wenn Sie mir von außerhalb Deutschlands einen Scheck schicken, der kein ec-Scheck (eurocheque) ist, addieren Sie bitte EUR 6 / DM 12. *oder*
• Schicken Sie Bargeld (auf eigenes Risiko).

Bei Ihrer Bestellung (schriftlich oder per E-Mail) nennen Sie bitte die Programmbezeichnung »WinHex 9.72« und Ihre Adresse. Über Möglichkeiten zur **Online-Bestellung** (etwas teurer als o. a., Bezahlung per Kreditkarte) erfahren Sie mehr unter http://www.sf-soft.de oder http://www.winhex.de.

Firmen, Behörden und Institutionen können bei u. a. Adresse auf Rechnung bestellen. Bitte geben Sie dabei, wenn vorhanden, eine E-Mail-Adresse an.

Nach Erhalt der Lizenzgebühr schicke ich Ihnen Freischaltcodes, damit Sie von sämtlichen Shareware-Hinweisen befreit werden, Dateien speichern können, die größer als 250 KB sind, mit dem Disk-Editor Sektoren schreiben und virtuellen Arbeitsspeicher editieren können. Alle späteren Versionen, die innerhalb von 12 Monaten nach Erscheinen *dieser* Version veröffentlicht werden, sind im Preis enthalten (evtl. noch mehr).

Anschrift:
Stefan Fleischmann
Carl-Diem-Str. 32
D-32257 Bünde

Kontonummer: 1208127686
Sparkasse Herford, BLZ: 494 501 20

Homepage: http://www.sf-soft.de und http://www.winhex.de
E-Mail: mail@sf-soft.de

Bitte besuchen Sie meine Homepage, um herauszufinden, ob es bereits eine neuere Version dieses Programms gibt.

Vielen Dank für Ihre Bestellung!

Ganzzahlige Datentypen

Format/Typ	Bereich	Beispiel
8 Bit, vorzeichenbehaftet	-128...127	FF = -1
8 Bit, vorzeichenlos	0...255	FF = 255
16 Bit, vorzeichenbehaftet	-32.768...32.767	00 80 = -32.768
16 Bit, vorzeichenlos	0...65.535	00 80 = 32.768
32 Bit, vorzeichenbehaftet	-2.147.483.648...2.147.483.647	00 00 00 80 = -2.147.483.648
32 Bit, vorzeichenlos	0...4.294.967.295	00 00 00 80 = 2.147.483.648
64 Bit, vorzeichenbehaftet	-2^63...2^63-1	00 00 00 00 00 00 00 80 = -2^63

Sofern nicht anders angegeben, sind ganzzahlige Datentypen im Little-Endian-Format gespeichert. D. h. das erste Byte einer Zahl ist das niederstwertige und das letzte Byte ist das höchstwertige. Dies ist das gebräuchliche Format für Computer, auf denen Windows läuft.

Wenn beispielsweise in einer Datei die Hex-Werte 10 27 stehen, so entspricht dies als numerischer 16-Bit-Wert der Hexadezimal-Zahl 2710 (was ins Dezimalsystem umgerechnet 10000 bedeutet). Ebenso erscheint die Hexadezimal-Zahl 123 als 23 01. Das Byte mit dem Wert 23 ist das niederwertige (es enthält die Einer- und die 16er-Stelle der Zahl) und kommt daher zuerst.

Eine weitere Besonderheit ist beim Interpretieren von Daten-Bytes als numerische Werte zu beachten: Zahlen, die größer als die Hälfte der Maximalzahl verschiedener Werte eines Zahlentyps sind (8 Bit: 2 hoch 8=256, 16 Bit: 2 hoch 16=65536), können als negative Zahlen übersetzt werden. Der Hex-Wert 8235 (der in einer Datei als 35 82 erscheint, s. o.), kann ins Dezimalsystem zu 33333 umgerechnet werden. Ein Programm, das den 16-Bit-Wert aber vorzeichenbehaftet liest, erhält aber die Zahl "-32203". Diese zweite Möglichkeit ergibt sich, wenn von der Übersetzung als vorzeichenloser Wert die Maximalzahl verschiedener numerischer Werte des Zahlentyps subtrahiert wird (Beispiel: 33333-65536=-32203).

Die Darstellung in der Statusleiste, der Daten-Dolmetscher (der Daten in allen obigen Formaten auf einmal übersetzen kann) und die Funktion »Ganze Zahl suchen« im Suchen-Menü berücksichtigen die genannten Besonderheiten automatisch.

Gleitkomma-Datentypen

Typ	Bereich	signifikante Stellen	Bytes
float (single)	± 1,5e-45..3,4e38	7-8	4
real	± 2,9e-39..1,7e38	11-12	6
double (double)	± 5,0e-324..1,7e308	15-16	8
long double (extended)	± 3,4e-4932..1,1e4932	19-20	10

Die Bezeichnungen stammen aus der Programmiersprache C, in Klammern ist die entsprechende Pascal-Bezeichnung angegeben. Der Typ real in nur in Pascal vorhanden.

Die Gleitkommazahlen werden im Computer unter Zuhilfenahme von Zweierpotenzen abgebildet. Gespeichert werden die Mantisse m und der Exponent e aus der Darstellung m × (2 hoch e). Beide Werte enthalten ein Vorzeichen. Die Datentypen unterscheiden sich in ihrem Wertebereich (=der Anzahl der für den Exponenten reservierten Bits) und der Genauigkeit der Werte (=der Anzahl der für die Mantisse reservierten Bits).

Rechenoperationen mit Gleitkommazahlen werden in Intel-Architekturen vom mathematischen Koprozessor ausgeführt während der Hauptprozessor wartet. Der Intel 80x87 rechnet mit einer Genauigkeit von 80 Bit, RISC-Prozessoren häufig mit 64 Bit.

Hexadezimal-Werte in einem Editierfenster können vom Daten-Dolmetscher in alle vier Gleitkomma-Datentypen übersetzt werden.

ANSI-/IBM-ASCII

ANSI-ASCII ist der Zeichensatz, der in Windows-Anwendungen verwendet wird (genormt vom American National Standards Institute). MS-DOS benutzt den IBM-ASCII-Zeichensatz (auch als OEM-Format bezeichnet). Diese Zeichensätze unterscheiden sich in der Zuordnung von Zeichen, deren ASCII-Wert über 127 liegt. Wenn Sie einen Text zum Beispiel mit dem Windows-Notizblock (notepad.exe) schreiben und ihn sich später mit dem Editor von MS-DOS ansehen (edit.com), dann werden Umlaute und Sonderzeichen nicht richtig dargestellt.

Schalten Sie daher die Option »ANSI-Zeichensatz« ab, wenn Sie mit WinHex eine Datei editieren, die zu einem DOS-Programm gehört. Sie sehen dann die in der Datei enthaltenen Texte wie sie auch in diesem Programm erscheinen. Die von ihnen eingegebenen Zeichen werden dann umgekehrt auch richtig in diesem DOS-Programm dargestellt. Wenn Sie hingegen eine typische Windows-Datei bearbeiten (Initialisierungsdateien von Windows-Programmen, Windows-Programmdateien usw.), sollten Sie die Option »ANSI-Zeichensatz« aktivieren.

Mit der Funktion »Konvertieren« im Bearbeiten-Menü können Textdateien von einem Zeichensatz in den anderen konvertiert werden.

Die ersten 32 ASCII-Zeichen sind weder Buchstaben oder Zahlen noch Satzzeichen. Es handelt sich um Steuerzeichen.

Hex	Steuerzeichen
0	Null
1	Start of Header
2	Start of Text
3	End of Text
4	End of Transmission
5	Enquiry
6	Acknowledge
7	Bell
8	Backspace
9	Horizontal Tab
A	Line Feed
B	Vertical Tab
C	Form Feed
D	Carriage Return
E	Shift Out
F	Shift In
10	Data Link Escape
11	Device Control 1 (XON)
12	Device Control 2
13	Device Control 3 (XOFF)
14	Device Control 4
15	Negative Acknowledge
16	Synchronous Idle
17	End of Transmission Block
18	Cancel
19	End of Medium
1A	Substitute
1B	Escape
1C	File Separator
1D	Group Separator

1E Record Separator
1F Unit Separator

Prüfsummen

Eine Prüfsumme ist eine Kennzahl zur möglichst eindeutigen Identifizierung von Daten. Zwei Datensätze mit der gleichen Prüfsumme sind mit hoher Wahrscheinlichkeit exakt (Byte für Byte) gleich. Es kann z. B. sinnvoll sein, die Prüfsumme von Daten *vor* und *nach* einer möglicherweisen fehlerbehaften Übertragung zu berechnen. Ist sie in beiden Fällen gleich, dann sind die Daten mit hoher Wahrscheinlichkeit unverändert geblieben. Allerdings können Daten mit bösartiger Absicht so manipuliert werden, daß ihre Prüfsumme trotz Änderung gleich bleibt. Dadurch wird die Manipulation nicht bemerkt. Diese Möglichkeit schließen Digests aus.

Prüfsummen können in WinHex beim Öffnen einer Datei (s. Optionen) und mit der Datenanalyse (im Extra-Menü) berechnet werden. Durch Drücken der Tastenkombination Alt+F2 wird die in der Informationspalte angezeigte Prüfsumme neu berechnet, wenn an einer Datei Änderungen vorgenommen wurden.

Die **Standard-Prüfsumme** ist einfach die Summe aller Bytes einer Datei auf einem 32-Bit-Akkumulator. Die **CRC32-Prüfsumme** (**C**yclic **R**edundancy **C**ode) wird mit einem komplizierteren, auf Polynomdivision beruhenden Algorithmus berechnet, der *sicherer* ist. Das drückt sich in einer niedrigeren Wahrscheinlichkeit dafür aus, für zwei verschiedene Dateien durch Zufall dieselbe Prüfsumme zu erhalten.

Beispiel: Wenn in einer Datei durch fehlerhafte Übertragung zwei Bytes verfälscht werden, sich die Abweichungen aber genau ausgleichen (z. B. erstes Byte +1, zweites Byte -1), dann bleibt die Standard-Prüfsumme im Gegensatz zur CRC32-Prüfsumme unverändert.

Technische Hinweise

Platzbedarf im Arbeitsspeicher:	0,5 MB
Speicherbedarf pro Routine:	0,5 KB
Maximalzahl geöffneter Fenster:	1000 (Windows NT/2000), 500 (Windows 9x)
Maximale Datei- u. Datenträgergröße:	ca. 1024 GB
Max. Anz. paralleler Instanzen:	99
Max. Anz. von Routinen:	100
Max. umkehrbare Tastatureingaben:	65535
Verschlüsselungstiefe:	128 Bit
Digest in Sicherungsdateien:	256 Bit
Zeichensätze der Textdarstellung:	ANSI-/IBM-ASCII, EBCDIC
Offset-Darstellung:	hexadezimal/dezimal

• Die Fortschrittsanzeige bei länger andauernden Operation zeigt in Prozent den Anteil des Vorgangs an, der bereits erledigt ist. Bei allen Suchen- und Ersetzen-Operationen zeigt sie jedoch die relative Position in der aktuellen Datei an. Dies entspricht dem bereits erledigten Anteil des Vorgangs, wenn in der gesamten Datei gesucht wird, also die Option »Nur im Block suchen« nicht verwendet wird.

• Zur optimalen Darstellung aller Schriftzeichen in WinHex sollte Ihr Windows-System *keine* extragroßen Systemschriften benutzen.

• WinHex wurde ausschließlich für Computer im »Little-Endian«-Modus konzipiert.

• Such- und Ersetzen-Funktionen laufen generell schneller ab, wenn kein Jokerzeichen verwendet und (bei Text-Suche) nach Groß- und Kleinschreibung unterschieden wird. Außerdem gilt: Je länger die Such-Zeichenfolge, desto schneller die Such-Funktion.

• Beim Suchen mit aktivierter Option »Vorkommen zählen« und beim Ersetzen ohne Bestätigung bieten sich für einen Suchalgorithmus zwei Alternativen für das Verhalten bei Fundstellen an, die in Sonderfällen zu unterschiedlichen Ergebnissen führen. Dies soll anhand eines Beispiels verdeutlicht werden:

In der Zeichenfolge »ananas« wird nach »ana« gesucht; das Vorkommnis beim ersten Zeichen wurde gefunden.
1. Möglichkeit: Ab dem zweiten Zeichen wird wieder nach »ana« gesucht. Beim dritten Zeichen wird dann ein Vorkommnis registriert.
2. Möglichkeit: Die drei mit der Suchzeichenfolge übereinstimmenden Zeichen werden übersprungen. »ana« wird erst wieder ab dem vierten Zeichen gesucht, in »nas« also nicht mehr gefunden.

In WinHex wird der zweiten Alternative gefolgt, da sie für das Zählen von Vorkommen und das Ersetzen ohne Bestätigung meistens sinnvollere Ergebnisse liefert. (Wenn Sie normale Suchvorgänge mit F3 fortsetzen oder Ersetzen *mit* Bestätigung wählen, wird nach der ersten Methode verfahren.)

• Hier erfahren Sie etwas über den Aufbau des Master-Boot-Sektors einer Festplatte, den Sie mit dem Disk-Editor editieren können.

• Weitere technische Informationen erhalten Sie auf der WinHex-Homepage unter http://www.winhex.com.

Rechtliche Hinweise

Allgemeine Optionen

1. Spalte:

• Wenn Sie die Option »**WinHex als Standardzuordnung**« aktivieren, wird in der Systemregistrierung der Schlüssel »HKEY_CLASSES_ROOT\Unknown\shell\Open\Command« angelegt bzw. angepaßt. Sie können dann die Dateien eines nicht registrierten Dateityps durch einen Doppelklick im Explorer mit WinHex öffnen. Bei allen anderen Dateien (selbst ausführbaren Dateien) ist nur die Shift-Taste zusätzlich zu drücken.

• **WinHex** kann sich **im Windows-Kontextmenü** eintragen. Das Kontextmenü sehen Sie, wenn Sie im Windows-Explorer oder auf dem Desktop mit der rechten Maustaste ein Objekt anklicken. Wenn Sie die Option nur halb aktivieren, gibt es keinen Kontextmenü-Eintrag für einzelne Dateien.

• Sie können Winhex auf Wunsch **mehrfach zugleich ausführen**. In der Voreinstellung jedoch wird eine bereits geladene Instanz des Programms aktiviert statt eine neue erzeugt.

• Wenn die Option »**Dateidatum und -zeit beibehalten**« aktiviert ist, werden Datum und Uhrzeit der letzten Änderung einer Datei beim Speichern auf dem Stand belassen, den die Datei zum Zeitpunkt des Öffnens hatte.

• Geben Sie an, wie lang die **Liste der zuletzt geöffneten Dateien** am Ende des Dateimenüs sein soll.

• Die **Symbolleiste** wird optional angezeigt.

• Eine **Tab-Leiste**, die es erlaubt, alle Editierfenster mit einem einfachen Mausklick anzuwählen, wird ebenfalls optional angezeigt.

• Die **Informationsspalte,** die Details über das editierte Objekt (Datei, Datenträger, RAM) aufführt, wird auch optional angezeigt.

• Sofern die Option »**Datei-Icons anzeigen**« aktiviert ist, werden ab einer Bildschirmauflösung von 800×600 die in der aktuellen Datei enthalten Windows-Icons unterhalb der Informationsspalte angezeigt. Je nach Anzahl der zu ladenden Icons nimmt das Öffnen von Dateien mehr Zeit und Arbeitsspeicher in Anspruch. Enthält eine Datei keine Icons, so wird das dem Dateityp zugeordnete Icon dargestellt, es sei denn, die Option ist nur »halb« markiert

• Entscheiden Sie, ob das Betätigen der **»Enter«**-Taste Hex-Werte in die zu editierende Datei schreiben soll. In der Voreinstellung sind dies 0x0D0A (=Zeilenende-Zeichen). Sie können bis zu vier zweistellige Hex-Werte angeben.

• Auf Wunsch können Sie mit der Tabulator-Taste auf Ihrer Tastatur das **Tabulator-Zeichen** erzeugen (0x09). Um dann vom Hexadezimal-Modus in den Text-Modus und umgekehrt zu wechseln, müssen Sie die Tabulator-Taste zusammen mit der Umschalt-Taste drücken.

2. Spalte:

• Ändern Sie wenn nötig den **Ordner**, in dem die **temporären Dateien** angelegt werden. Voreingestellt ist der Ordner, den die Umgebungsvariablen »TEMP« Ihres Systems definiert.

• Ebenfalls wählen können Sie den **Ordner**, in dem die **Sicherungsdateien** angelegt werden. Normalerweise ist er identisch mit dem für temporäre Dateien.

• Geben Sie den Pfad und den Dateinamen eines **Texteditors** oder Textverarbeitungsprogramms ein. Die von WinHex erstellten Berichte über Dateivergleiche können Sie dann mit diesem Programm einsehen. Der Windows-Editor *notepad.exe* kann nur Textdateien bis zu einer Größe von 64 KB anzeigen.

• Bestimmen Sie, ob die **Offsets** (Byte-Adressen) in **dezimaler** oder **hexadezimaler** Schreibweise angegeben und zur Eingabe verlangt werden. Diese Einstellung gilt für den gesamten Umfang des Programms.

• Auf Wunsch können beim Benutzen des RAM-Editors anstelle von Null-basierten Offsets **virtuelle Adressen** angezeigt werden. Dies geschieht grundsätzlich in hexadezimaler Schreibweise. Im Dialogfenster der Funktion »Offset aufsuchen« sind dann auch virtuelle Adressen einzugeben.

• »**0x01-0x1F als Punkt darstellen**« bewirkt, daß im ANSI-ASCII-Zeichensatz Hex-Werte aus dem Bereich von 01 bis 1F ebenso wie der Wert 00 als Punkt [.] in der Textdarstellung angezeigt werden.

• Geben Sie an, wie viele **Bytes** in einem Editierfenster **pro Zeile** dargestellt werden sollen. Standardeinstellungen sind 16 oder 32 Bytes, je nach Bildschirmauflösung.

• Geben Sie an, wie viele **Bytes** als **Gruppe** zusammenhängend angezeigt werden sollen. I. d. R. empfiehlt sich eine Zweier-Potenz.

3. Spalte:

• Normalerweise stellt WinHex den **Cursor doppelt** dar (in beiden Hälften des Fensters). Im Text-Modus befindet sich der Haupt-Cursor auf der Hexadezimal-Hälfte, im Hex-Modus auf der Text-Hälfte. Der Zweit-Cursor in der jeweils anderen Hälfte steht in zwei Arten zur Verfügung.

• Bei zeilenweisem Rollen können **Seiten**- und Sektor**trennlinien angezeigt** oder ausgeblended werden. Wenn die Option nur halb gewählt ist, werden nur Sektortrennlinien angezeigt.

• Wenn die Option »**Zeilenweises Rollen**« eingeschaltet ist, können Sie im Editorfenster Zeile für Zeile statt seitenweise vor und zurückgehen.

• Sie können die Anzahl der **Zeilen** bestimmen, die mit Hilfe des **Mausrades** (falls vorhanden) vorwärts und zurück **gerollt** werden können.

• Bei aktivierter Option »**Windows-Standardfarben benutzen**« wird das Editorfenster in den Farben angezeigt, die in der Windows-Systemsteuerung eingestellt sind. Andernfalls werden die Standardfarben von WinHex verwendet.

• Sie haben die Möglichkeit, die **Hintergrundfarbe** des Blocks zu bestimmen, wenn die Option »Windows-Standardfarben benutzen« nicht aktiviert ist.

• Wählen Sie eine **Schriftart** für die Darstellung im ANSI-ASCII-Format aus. Das Verwenden der WinHex-Schriftart stellt sicher, daß auch Sonderzeichen in der Textdarstellung angezeigt werden (z. B. die Symbole TM und Euro sowie echte Anführungszeichen).

• Die Option »**Sound in Meldungsfenstern**« bestimmt, ob beim Anzeigen von Meldungsfenstern die in der Windows-Systemsteuerung festgelegten Klänge abgespielt werden.

• Bei aktivierter Option »**Windows-Fortschrittsanzeige**« wird während lang dauernden Vorgängen die Windows-typische anstatt der WinHex-eigenen Fortschrittsanzeige verwendet.

Bestimmen Sie außerdem das Aussehen der **Dialog-** und Meldungs**fenster** in WinHex. Sie haben die Wahl zwischen vier verschiedenen **Stil**en.

Die Voreinstellungen sämtlicher Optionen können durch Benutzen der Funktion »Initialisieren« im Hilfe-Menü wiederhergestellt werden.

Zeichen eingeben

Mit der Tastatur lassen sich im Hex-Modus nur Hexadezimal-Zeichen eingeben ('0' bis '9' und 'A' bis 'F').

Im Text-Modus lassen sich dagegen alle Zeichen eingeben: Buchstaben, Zahlen, Satzzeichen und auch Sonderzeichen (wie '»', ']' und '^'). Durch Benutzen des Windows-Programms »Zeichentabelle« kann man herausfinden, durch welche Tastenkombinationen evtl. erwünschte Sonderzeichen zu erzeugen sind (z. B. Alt-1-7-5 für '»').

Editier-Modi

Standard-Editiermodus: Im voreingestellten Standard-Editiermodus werden Änderungen, die Sie an einer geöffneten Datei vornehmen, in einer temporären Datei gespeichert. Entweder diese wird dynamisch verwaltet oder bereits beim Öffnen der Originaldatei erzeugt (s. Option »Schnelles Öffnen von Dateien«). Beim Speichern werden die Änderungen dann in die Originaldatei übertragen.

View-Modus: Dateien, die im View-Modus geöffnet werden, können nicht editiert, sondern nur eingesehen werden. Dies entspricht der Möglichkeit in anderen Programmen, Dateien »schreibgeschützt« zu öffnen.

In-Place-Modus: Verwenden Sie diesen Modus mit Vorsicht. *Sämtliche* Änderungen (Tastatureingaben, Füllen/Entfernen des Blocks, Schreiben des Zwischenspeichers, Ersetzen-Vorgänge, ...) werden direkt in die Originaldatei (»in-place«) geschrieben. Dies geschieht dynamisch, spätestens aber, wenn das Editierfenster geschlossen wird. Es ist daher nicht erforderlich, den Menüpunkt »Speichern« im Dateimenü aufzurufen, es sei denn, Sie möchten sicherstellen, daß alle Änderungen zu einem bestimmten Zeitpunkt geschrieben werden, wenn das Editierfenster noch geöffnet ist.

Dieser Modus empfiehlt sich, wenn das im Standard-Editiermodus obligate Übertragen von Daten aus der Originaldatei in die Temporärdatei und umgekehrt zu zeitaufwendig wäre und zuviel Festplattenspeicherplatz verbräuchte. Dies kann z. B. dann der Fall sein, wenn in großen Dateien viele Änderungen vorgenommen werden sollen. Da im In-Place-Modus keine Daten in temporären Dateien gespeichert werden, ist dieser Editiermodus generell schneller als der Standard-Editiermodus. Der In-Place-Modus ist der einzige Modus, in dem der RAM-Editor benutzt werden kann.

Hinweis: Auch im In-Place-Modus muß eine temporäre Datei angelegt werden, wenn die *Größe* der Originaldatei geändert wird.

Statusleiste

Die Statusleiste zeigt beim Einsehen einer Datei folgende Informationen an:

1. Feld: aktuelle Seite und Anzahl der Seiten, auf denen die aktuelle Datei dargestellt wird
2. Feld: Cursorposition (Offset in der Datei)
3. Feld: ins Dezimalsystem übersetzte Hex-Werte an der Cursorposition
4. Feld: Blockanfang und -ende (falls festgelegt)
5. Feld: Größe des Blocks in Byte (dto.)

Durch einen Klick der linken Maustaste läßt sich...
• im 1. Feld eine andere Seite aufschlagen,
• im 2. Feld den Cursor zu einem bestimmten Offset bewegen,
• im 3. Feld das Format festlegen, in dem die Hex-Werte als Zahlen des Dezimalsystems interpretiert werden, und
• im 4. und 5. Feld den Block neu definieren.

Klicken Sie mit der rechten Maustaste, um in einem Feld der Statusleiste angezeigte Informationen in die Zwischenablage zu kopieren.

Durch einen Mausklick rechts im 2. Feld der Statusleiste können Sie von absoluter Offsetdarstellung (Standard) auf relative Datensatz-Offsets umschalten. Dies ist nützlich, wenn die von Ihnen im Hex-Editor untersuchten Daten aus gleich langen Datensätzen bestehen. Nachdem Sie deren Länge angegeben haben, wird Ihnen für die aktuelle Cursorposition anstelle des absoluten Offsets jeweils die Nummer des Datensatzes und der relative Offset darin angezeigt.

Ein Rechts-Klick auf das 3. Feld der Statusleiste erlaubt es außerdem, die vier Hex-Werte an der aktuellen Cursorposition in umgekehrter Reihenfolge in die Zwischenablage zu kopieren. Dies ist nützlich beim Verfolgen von Zeigern.

Arbeitserleichterungen

- Linke Maustaste — Blockanfang festlegen (Doppelklick)
- Rechte Maustaste — Blockende festlegen
- Rechte Maustaste — Blockmarkierung aufheben (Doppelklick)
- Shift+Pfeiltasten — Block markieren
- Alt+1 — Blockanfang setzen
- Alt+2 — Blockende setzen
- Tabulatortaste — zwischen Text- und Hexmodus umschalten
- Strg+Q — alle Fenster schließen
- (Strg+)Enter — Fenster-Manager aufrufen
- ESC — aktuellen Vorgang abbrechen bzw. Blockauswahl aufheben
- PAUSE — aktuellen Vorgang anhalten bzw. fortsetzen
- F11 — »Offset aufsuchen« wiederholen
- Shift+F7 — Zeichensatz wechseln
- (Shift+)Alt+F11 — »Block verschieben« wiederholen
- Alt+F2 — Prüfsumme neu berechnen

• Schalten Sie die ersten beiden allgemeinen Optionen ein, um WinHex voll in die Windows-Oberfläche zu integrieren.

• WinHex akzeptiert Dateinamen als Startparameter und öffnet Dateien, die per Drag&Drop (mit der Maus) in das Programmfenster gezogen werden.

• Der Einsatz von Routinen kann Ihr Arbeiten mit WinHex effizienter machen.

• Als Parameter wird auch die Nummer einer Routine akzeptiert (s. dort).

• Die Offset-Schreibweise (dezimal oder hexadezimal) läßt sich durch einen Mausklick auf die Offsetdarstellung im Editorfenster umstellen. Die dezimale Schreibweise ist mit oder ohne führende Nullen verfügbar (Mausklick rechts).

• Bei einer Bildschirmauflösung von mind. 800×600 und maximiertem Rahmenfenster kann man mit der Maus die untere Begrenzung der Statusleiste und der Informationsspalte nach oben und unten ziehen. Damit läßt sich die Anzahl der Zeilen im Editorfenster bzw. die Länge der Informationsspalte festlegen.

• Klicken Sie probehalber auf die Statusleiste (linke und rechte Maustaste).

• Durch einen Mausklick unter die Informationsspalte rechts im Editorfenster (ab Auslösung 800×600) kann das Optionen-Dialogfenster aufgerufen werden.

Routinen

Wenn Sie in bestimmten Dateien häufig dieselben Änderungen durchführen, können Sie versuchen, diesen Vorgang kann unter Einsatz einer Routine zu automatisieren. WinHex vermag bis zu 100 Routinen zu verwalten. Die entworfenen Routinen werden bei Programmende in der Datei *Routines.dat* gespeichert. Im Routinen-Manager können Sie Routinen entwerfen und bearbeiten.

1. Um eine Routine zu entwerfen, müssen Sie zuerst festlegen, welche Datei(en) bearbeitet werden soll(en). Entweder die Routine bezieht sich auf die zum Zeitpunkt der Ausführung im Editorfenster dargestellte Datei, auf *sämtliche* in WinHex geöffneten Dateien oder auf eine feststehende, mit Dateinamen und Pfad angegebene Datei. An Dateien, die im View-Modus geöffnet wurden, kann eine Routine *nicht* ausgeführt werden.

2. Wenn Sie an bestimmten Stellen der Datei einzelne Bytes auf einen festen Wert setzen wollen, so tragen Sie jeweils den Offset und den Wert, den das Byte annehmen soll, in die dafür vorgesehenen Editierfelder ein. Es können bis zu fünf Bytes direkt adressiert und geändert werden Der neue Byte-Wert muß in hexadezimaler Form eingegeben werden.

3. Wenn Sie Text oder Hex-Werten ersetzen möchten (dies ist hier nur ohne Bestätigung möglich), so aktivieren Sie den entsprechenden Schalter. Tragen Sie die Zeichenfolge, die gesucht werden soll, und ihren Ersatz ein. Sie können einen durch Offsets definierten Bereich in der Datei angeben, auf den der Vorgang beschränkt werden soll. Geben Sie auf Wunsch einen Joker an und aktivieren Sie ggf. die Optionen »Groß- und Kleinschreibung beachten« sowie »Nur ganze Wörter ersetzen« (s. Optionen des Ersetzens).

4. Legen Sie fest, ob zu Beginn und am Ende der Ausführung Hinweise erscheinen sollen. Warnungen und Fehlermeldungen werden ungeachtet dessen auf jeden Fall angezeigt. Eine weitere Option ist die automatische Speicherung der bearbeiteten Dateien nach Beendigung der Routine.
Außerdem können Sie eine *Folgeroutine* angeben, die direkt im Anschluß ausgeführt werden soll. Wenn diese sich auf dieselbe Datei beziehen soll, können Sie als zu bearbeitende Datei »Aktuelle Datei« angeben. Die Folgeroutine einer auf alle geöffneten Dateien einwirkenden Routine aber kann sich *nicht* auf eine »aktuelle« Datei beziehen, da sonst unklar ist, welche Datei gemeint ist. Es ist möglich, mehrere Routinen verketten. Wenn sich jedoch Routinen wechselseitig aufrufen, gerät das Programm in eine Endlosschleife!
Als letztes geben Sie der Routine eine aussagekräftigen Bezeichnung.

Tips:

• WinHex bietet die Möglichkeit, zuvor erstellte Routinen (ggf. jeweils mitsamt Folgeroutinen) automatisch bei Programmstart auszuführen. Übergeben Sie einfach die Nummer der Routine als Parameter in der Kommandozeile (z. B. »winhex 4«). Wenn die genannte Routine existiert, lädt WinHex die in der Routine angegebene Datei, führt die Änderungen durch und speichert sie, falls die entsprechende Option gewählt wurde.

• Wenn vor dem Aufruf einer Routine über die Kommandozeile bereits Dateien in WinHex geöffnet wurden, können diese von der Routine bearbeitet werden, sofern sie sich auf »alle geöffneten Dateien« bezieht. Entweder WinHex läuft bereits mit geöffneten Dateien oder diese werden vor der Nummer der Routine angegeben.

• Wenn Sie mehrere Routinen, die sich auf geöffnete Dateien beziehen, nacheinander ausführen und die Änderungen sofort speichern lassen möchten, muß nur in der zuletzt ausgeführten Routine die Option »Änderungen automatisch speichern« gewählt sein.

• Wird als letzter Parameter »auto« übergeben, dann beendet sich WinHex automatisch.

Beispiel:

Der Aufruf »winhex c:\datei1.dat 1 d:\datei2.dat 4 auto« bewirkt folgendes: Erst wird die Datei c:\datei1.dat geöffnet und dann die Routine Nr. 1 auf diese Datei angewandt. Anschließend wird zusätzlich die Datei d:\datei2.dat geöffnet und die Routine Nr. 4 an beiden Dateien ausgeführt. Am Ende wird WinHex geschlossen. Damit der genannte Aufruf die gewünschten Resultate erzielt, müssen sich beide Routinen auf »alle geöffneten Dateien« beziehen. Die zweite ist auf automatische Speicherung einzustellen.

Disk-Editor

Im Extra-Menü finden Sie die Funktion »Disk-Editor«. Der Disk-Editor ermöglicht es, den Inhalt einer Diskette oder Festplatte ohne Rücksicht auf die Dateistruktur direkt einzusehen. Dazu wird auf jeder Seite genau ein Sektor dargestellt. Wählen Sie zunächst aus einer Liste mit den auf Ihrem System installierten Laufwerken einen Datenträger aus. Sie können auf einen Datenträger logisch (vom Betriebssystem gesteuert) oder physisch (vom BIOS gesteuert) zugreifen. Auf den meisten Computersystemen können Sie sogar CD-ROMs und DVDs einsehen.

Disk Editor Questions & Answers

Bitte beachten Sie die folgenden Einschränkungen bzw. Voraussetzungen:

• Um unter Windows NT auf Festplatten zugreifen zu können, sind Administrator-Rechte erforderlich.
• Die Ersetzen-Funktionen sind im Disk-Editor generell nicht verfügbar.
• WinHex kann auf CD-ROMs und DVDs nicht schreiben.
• Der Disk-Editor kann nicht auf Netzlaufwerke zugreifen.

Beachten Sie bitte die Option »Alleinige Datenträgerkontrolle«.

Freien Speicher des Datenträgers editieren (Windows 95/98)

Unter Windows 95/98 ist es möglich, den unbenutzten Speicher eines logischen Datenträgers einzusehen und zu editieren. Dabei entfallen die o. g. Einschränkungen. Es wird eine Datei angelegt, die den gesamten freien Speicher auf dem gewählten Datenträger belegt. In dieser Datei können Sie nun im In-Place-Editiermodus Änderungen vornehmen. Dies kann die Integrität der Daten in benutzten Bereich des Datenträgers *nicht* beeinflussen.

Anwendungsbeispiele *unter Windows 95/98*:
• Sie können mit Hilfe dieser Funktion versehentlich gelöschte Daten, die noch nicht von neuen Dateien überschrieben worden sind, wiederherstellen, z. B. indem Sie sie erst suchen, dann als Block markieren und kopieren.
• Ebenso ist die Funktion nützlich, um den unbenutzten Speicher eines Datenträgers aus Sicherheitsgründen mit Nullen zu überschreiben. Grund: Vertrauliche Informationen könnten durch normale Lösch- und Kopiervorgänge in momentan unbenutzten Bereichen des Datenträgers liegen. Füllen Sie den Bereich zu diesem Zweck einfach mit dem Hex-Wert 00.

Unbenutzten Speicher des Datenträgers initialisieren (Windows NT)

Unter Windows NT kann mit dieser Funktion der unbenutzte Speicher eines Datenträgers (z. B. aus Sicherheitsgründen) mit Nullen initialisiert werden. Dies verhindert die Wiederherstellung von Daten aus diesem Bereich des Datenträgers.

Anmerkung: Daten aus Dateien, die mit der Funktion »Sicheres Löschen« von WinHex gelöscht wurden, befinden sich natürlich nicht mehr in unbenutzten Bereichen des Datenträgers.

Auf Disk schreiben: Entspricht dem Befehl »Speichern« für Dateien und befindet sich an dessen Stelle im Menü. Schreibt die von Ihnen vorgenommenen Änderungen auf den Datenträger. Bitte beachten Sie, daß Sie damit einen äußerst kritischen Eingriff in die Integrität des Datenträgers vornehmen. Sofern die entsprechende Rückgängig-Option eingeschaltet ist, wird von den betroffenen Sektoren vor dem Überschreiben eine Sicherung angelegt.

Die Funktion nur in der Vollversion benutzbar.

Hier erfahren Sie etwas über den Aufbau des Master-Boot-Record einer Festplatte, den Sie mit dem Disk-Editor editieren können.

Datei-Menü

Neu: Hier können Sie eine neue Datei anlegen, deren Inhalt mit Null-Bytes initialisiert wird. Es ist die gewünschte Größe der Datei in Bytes anzugeben (>0). Die neue Datei wird prinzipiell im Standard-Editiermodus geöffnet.

Öffnen: In einem Dateiauswahlfenster markieren Sie eine oder mehrere Dateien, die Sie mit dem Hex-Editor einsehen oder bearbeiten möchten. Sofern Sie WinHex nicht schon im Extra-Menü als Viewer oder In-Place-Editor eingestellt haben, können Sie einen der drei Editier-Modi zum Öffnen der Datei(en) wählen.

Speichern: Hier speichern Sie ein zuvor geöffnete Datei mit allen von Ihnen vorgenommenen Änderungen, nachdem Sie eine Sicherheitsabfrage mit »Ja« beantwortet haben. Im In-Place-Editiermodus ist das Aufrufen dieses Befehls nicht notwendig. Beim Benutzen des Disk-Editors heißt dieser Befehl »Auf Disk schreiben«.

Speichern unter: Speichert eine Datei unter einem neuen Namen oder in einem anderen Ordner. Existiert bereits eine Datei mit diesem Namen, so werden Sie gefragt, ob die vorhandene Datei überschrieben werden soll.

Sicherung anlegen

Sicherung laden: Wählen Sie eine Sicherungsdatei (=WHX-Datei) aus, deren Inhalt (eine Datei oder Datenträger-Sektoren) Sie wiederherstellen möchten.

Sicherungs-Manager

Ausführen: Führt die aktuell dargestellte Datei mit allen evtl. vorgenommenen Änderungen aus. Es muß sich entweder um eine unter DOS oder Windows ausführbare EXE- oder COM-Datei handeln oder der Dateityp muß unter Windows mit einer Anwendung verknüpft worden sein. Dann wird dieses Programm gestartet und die aktuelle Datei geladen. Sie können mit dieser Funktion z. B. überprüfen, ob die vorgenommenen Änderungen in einer Programmdatei ihre Ausführbarkeit beeinträchtigt haben.

Drucken

Eigenschaften: Hier können die Größe, Datum und Uhrzeit der Erzeugung, der letzten Änderung und des letzten Zugriffs sowie Attribute (A: zu archivierend, S: System, H: versteckt, R: schreibgeschützt) einer Datei (unter Windows NT auch eines Verzeichnisses) eingesehen und editiert werden. Nach Eingabe neuer Werte in einem der drei Bereiche betätigen Sie die Enter-Taste, damit die Änderungen in kraft treten.

Erweitertes Öffnen: Wählen Sie einen Ordner aus, dessen Dateien Sie öffnen möchten. Wahlweise werden auch die Dateien in untergeordneten Ordnern berücksichtigt. Sie können einen Dateifilter verwenden (z. B. »w*.exe«) und einen Editiermodus auswählen, wenn Sie WinHex nicht schon im Extra-Menü als Viewer oder In-Place-Editor eingestellt haben. Optional werden nur solche Dateien geöffnet, die einen bestimmten Text oder bestimmte Hex-Werte enthalten. In diesem Fall stehen Ihnen noch weitere Suchoptionen zur Verfügung.

Geänderte speichern: All die von WinHex geöffneten Dateien, an denen Änderungen vorgenommen wurden, werden mit ihrem aktuellen Inhalt gespeichert. Es erfolgen keine weitere Sicherheitsabfragen. Daher ist diese Funktion »mit Vorsicht zu genießen«.

Alle speichern: Sämtliche von WinHex nicht im View-Modus geöffneten Dateien werden mit ihrem aktuellen Inhalt gespeichert. Es erfolgen keine weitere Sicherheitsabfragen.

Beenden: Hier können Sie WinHex schließen. Sie erhalten noch einmal die Möglichkeit, Änderungen an Dateien und Datenträgern zu speichern.

Bearbeiten-Menü

Rückgängig: Erlaubt Ihnen, Tastatureingaben und die Anwendung sonstiger Funktionen ungeschehen zu machen. Dazu müssen die entsprechenden Optionen aktiviert sein.

Ausschneiden: Bewirkt, daß der aktuelle Block aus der Datei entfernt und in die Zwischenablage kopiert wird. Der dahinter liegende Teil der Datei wird entsprechend vorgezogen.

Block/Alles/Sektor kopieren
- normal: Kopiert den markierten Block bzw. den gesamten Dateiinhalt bzw. den aktuellen Sektor in die Zwischenablage, so daß er später wieder eingefügt werden kann.
- in neue Datei: Kopiert die Daten direkt in eine neue Datei (nicht über den Umweg Zwischenablage). Mit dieser Funktion kann man z. B. beliebige Daten von einem Datenträger schnell in Dateien umwandeln.
- Hex-Werte: Kopiert die Daten im Hexadezimal-Format in die Zwischenablage.
- C-Quellcode: Kopiert die Daten im C-Quelltext-Format in die Zwischenablage.
- Editoranzeige: Kopiert die Daten als Text so formatiert in die Zwischenablage, wie sie auch im Hex-Editor erscheinen, d. h. mit eine Offset-, einer Hex- und einer ASCII-Text-Spalte.

Zwischenspeicher einfügen: Fügt den Inhalt der Zwischenablage, sofern er in einem kompatiblen Format vorliegt, an der aktuellen Cursorposition ein. Der Teil der Datei, der dahinter liegt, wird hinter die Einfügung versetzt.

Zwischenspeicher schreiben: Überträgt den Inhalt der Zwischenablage an die aktuelle Cursorposition und *überschreibt* dabei die Bytes der Datei, die dahinter folgen. Falls dabei das Dateiende erreicht wird, wird die Datei so weit wie erforderlich verlängert, damit die Daten Platz finden.

Zwischenspeicher in neue Datei schreiben: Legt eine neue Datei mit dem aktuellen Inhalt der Zwischenablage an.

Zwischenspeicher freigeben: Löscht den Inhalt der Zwischenablage gibt den von ihm genutzten Teil des Arbeitsspeichers wieder frei.

Entfernen: Löscht den aktuellen Block aus der Datei. Der hintere Teil der Datei wird dann entsprechend vorgezogen. Der gelöschte Block wird *nicht* in die Zwischenablage kopiert. Wenn in allen geöffneten Dateien der Block gleich definiert ist (also an den gleichen Offsets beginnt und endet), können Sie diese Funktion wahlweise auch auf alle geöffneten Dateien anwenden.

Nullbytes einfügen: Läßt Sie eine bestimmte Anzahl von Bytes mit dem Wert Null an der aktuellen Cursor-Position einfügen.

Block festlegen: In einem Dialogfenster kann man die Offsets einstellen, die den Beginn und das Ende des aktuellen Blocks markieren. Diese Funktion ist auch über die Statusleiste zugänglich. Sie läßt sich wahlweise auch auf alle geöffneten Dateien anwenden.

Alles auswählen: Legt den Dateianfang als Blockanfang und das Dateiende als Blockende fest.

Konvertieren

Daten modifizieren

Block/Datei/Sektoren füllen

Suchen-Menü

Text suchen: Diese Funktion sucht Vorkommen einer max. 50stelligen Zeichenfolge in der aktuellen Datei (s. a. Suchoptionen).

Hex-Werte suchen: Sucht Vorkommen einer Kombination von max. 50 jeweils zweistelligen Hex-Werten (s. a. Suchoptionen).

Text ersetzen: Diese Funktion ersetzt Vorkommen einer Zeichenfolge in der Datei durch eine andere (s. a. Ersetzen-Optionen).

Hex-Werte ersetzen: Funktioniert genau wie der Befehl »Text ersetzen«, wird aber auf eine Folge von Hex-Werten angewandt (s. a. Ersetzen-Optionen).

Kombinierte Suche: Mit dieser besonderen Funktion können Sie eine komplexe Suche durchführen: In der aktuell angezeigten und einer auf einem Datenträger bestehenden Datei wird ein gemeinsamer Offset gesucht, an dem die beiden Dateien bestimmte Daten enthalten. Wählen Sie zunächst den Hex-Wert, der in aktuellen Datei an der gesuchten Position stehen soll. Geben Sie dann den Namen der zweiten Datei und den in ihr zu suchenden Hex-Wert an. WinHex sucht nun eine Stelle, an der in jeder Datei der jeweilige Hex-Wert steht.

Ganze Zahl suchen: Geben Sie eine natürliche Zahl (in den Grenzen eines vorzeichenbehafteter 64-Bit-Integer-Wertes) an. Die Funktion sucht dann diejenigen Bytes in der Datei, die als diese Zahl interpretiert werden könnten. Ist sie fündig geworden, gibt sie den Fundort und die entsprechenden Hex-Werte an und nennt das Format, in dem die Hexadezimal-Werte der eingegebenen Zahl entsprechen (s. a. Suchoptionen).

Gleitkommazahl suchen: Geben Sie eine Dezimalzahl (z. B. 12,34 = 0,1234 * (10 hoch 2) = 0,1234E2) und den Fließkomma-Datentyp an. Die Funktion sucht dann diejenigen Bytes in der Datei, die als diese Zahl interpretiert werden könnten. Ist sie fündig geworden, gibt sie den Fundort und die entsprechenden Hex-Werte an.

Textpassagen suchen: Sucht in der Datei einen Bereich mit aufeinanderfolgenden Buchstaben (a-z, A-Z; äöüß im ANSI-ASCII-Zeichensatz), Ziffern (0-9) und/oder Satz- und Leerzeichen. Diese Funktion erfüllt zum Beispiel dann ihren Zweck, wenn Sie in einer Programmdatei den sporadisch zwischen den Steuerzeichen vorkommenden Text finden möchten.
Regeln Sie, wie »sensibel« WinHex nach Vorkommen von Text sucht, indem Sie angeben, wie lang der Text sein muß, damit er als solcher erkannt wird.
Viele Dateitypen neueren Datums, darunter 32-Bit-Programmdateien, reservieren zwei Bytes für ein Zeichen statt eins (Unicode-Zeichensatz). Die Option »Unicode-Zeichen tolerieren« bedeutet, daß auch alphanumerische ASCII-Zeichen, zwischen denen jeweils ein Byte mit dem Wert Null steht, als Text erkannt werden.

Globale Suche fortsetzen: Setzt einen bereits begonnenen globalen, d.h einen mit Option »In allen geöffneten Dateien suchen« durchgeführten Suchvorgang, nach Anzeigen einer Fundstelle in der *nächsten* Datei fort. Soll zunächst in derselben Datei noch weiter gesucht werden, muß die Funktion »Suche fortsetzen« benutzt werden.

Suche fortsetzen: Führt einen bereits begonnenen Suchvorgang, auch nach Vorkommen von Text, aber keinen Ersetzen-Vorgang, von der aktuellen Cursor-Position an fort.

Position-Menü

Offset aufsuchen: Setzt den Cursor auf einen von Ihnen gewünschten Offset, d. h. eine Position in der Datei. Gewöhnlich wird diese relativ zum Anfang der Datei (Offset 0) angegeben. Sie können den Cursor aber auch relativ von der aktuellen Position vorwärts und rückwärts und vom Dateiende aus rückwärts bewegen. Die Maßeinheit ist entweder ein Byte, ein Word (2 Bytes) oder ein DoubleWord (4 Bytes). Verwenden Sie F11, um die gewählte Positionsveränderung zu wiederholen.

Seite/Sektor aufsuchen: Schlägt die von Ihnen angegebene Seite auf bzw. springt im Fall eines Datenträgers zum gewählten Sektor/Cluster. Bitte beachten Sie, daß der Datenbereich auf FAT-Laufwerken mit der Cluster-Nr. 2 beginnt.

Block verschieben: Verschiebt die aktuelle Block-*Markierung* (nicht die *Daten* im Block) nach vorne oder hinten. Geben Sie die Distanz in Byte an. Verwenden Sie Alt+F11, um die gewählte Blockverschiebung zu wiederholen, und Shift+Alt+F11, um in die jeweils umgekehrte Richtung zu verschieben. Diese Funktion kann z. B. beim Editieren einer Datei von Nutzen sein, die aus mehreren gleichartigen Datenfeldern (Records) derselben Länge besteht.

Anzeige aktualisieren: Erneuert die Anzeige im aktiven Editierfenster.

Dateianfang: Zeigt die erste Seite der Datei an und setzt den Cursor auf den Anfang der Datei (Offset 0).

Dateiende: Zeigt die letzte Seite der Datei an und setzt den Cursor auf das Ende der Datei (letztes Byte, Offset=Dateigröße-1).

Blockanfang: Setzt den Cursor auf den aktuellen Blockanfang.

Blockende: Setzt den Cursor auf das aktuellen Blockende.

Position markieren: Markiert die aktuelle Position optisch.

Markierung löschen: Löscht eine zuvor gesetzte Markierung vom Bildschirm.

Markierung aufsuchen: Setzt den Cursor auf die zuvor markierte Position.

Positions-Manager

Fenster-Menü

Fenster-Manager: Listet alle Edtierfenster auf und gibt Ihnen die Möglichkeit, schell zwischen verschiedenen Fenstern zu wechseln. Sie können im Fenster-Manager auch einzelne Fenster schließen und geänderte Dateien speichern.

Alle schließen: Schließt alle geöffneten Fenster und damit alle momentan in WinHex dargestellten Dateien und Datenträger.

Ohne Abfragen schließen: Funktioniert wie »Alle schließen«, ohne Ihnen jedoch die Möglichkeit zu geben, eventuelle Änderungen zu speichern.

Übereinander/Horizontal/Vertikal: Ordnet die Editierfenster wie beschrieben an.

Rollen synchronisieren: Synchronisiert bis zu 4 Fenster.

Synchronisieren und vergleichen: Synchronisiert zwei Fenster und zeigt unterschiedliche Bytewerte gesondert an.

Minimieren: Verkleinert alle Editierfenster.

Symbole anordnen: Richtet verkleinert dargestellte Fenster ordentlich am unteren Rand des Rahmenfensters aus.

Extra-Menü

Disk-Editor

Clusterketten inspizieren: Verfügbar für FAT16- und FAT32-Laufwerke. WinHex durchläuft alle Clusterketten und kann dadurch für jeden Sektor/Cluster angeben, was in ihm gespeichert ist bzw. ob er unbelegt ist. Durch Dateioperationen auf dem betreffenden Laufwerk veralten diese Informationen allerdings, und ein erneutes Aufrufen dieser Funktion bietet sich an. Vgl. Sicherheitsoptionen.

Datei-/Verzeichniscluster auflisten: Verfügbar für FAT16-, FAT32- und NTFS-Laufwerke. Setzt im Fall von FAT-Laufwerken voraus, daß die durch die Funktion »Clusterketten inspizieren« gewonnenen Informationen auf dem aktuellen Stand sind. WinHex sucht Cluster, die einer von Ihnen angegebenen Datei oder einem Verzeichnis zugeordnet sind. Die gefundenen Cluster werden in einem separaten Fenster aufgelistet. Klicken Sie auf einen Listeneintrag, um zum betreffenden Cluster zu springen.

Plattenparameter eingeben: Benutzen Sie diese Funktion für einen physischen Datenträger, um die von WinHex erkannte Zahl der Zylinder, Köpfe und Sektoren pro Spur anzupassen. Dies kann nützlich sein, um auf die überschüssigen Sektoren am Ende des Datenträgers unter Windows NT/2000 zugreifen zu können (unter Windows 9x/Me sollten diese bereits in der Ansicht enthalten sein) oder um das CHS-Koordinatensystem nach Ihren Wünschen zu ändern. Im Fall eines logischen Datenträgers können Sie die Zahl der zu erkennenden Cluster selbsten bestimmen, was erforderlich werden kann, wenn etwa eine große DVD von Windows 9x nur als 2 GB erkannt wird.

Dateien extrahieren: Eine einfache Funktion zur Dateiwiederherstellung, die Dateien auf einem beliebigen Datenträger (oder in einer Image-Datei) sucht und anhand eines bestimmten Datei-Headers erkennt. Es werden entweder Dateien fixer Größe wiederhergestellt oder es wird nach zugehörigen Datei-Footern gesucht, die das Dateiende markieren. Header und Footer müssen in hexadezimaler Schreibweise angegeben werden. Im Ausgabeverzeichnis wird auch eine Protokolldatei angelegt, die über die die Offsets der erkannten und verwendeten Header und Footer Aufschluß gibt. Die resultierenden Dateien werden nach einem Muster, das Sie bestimmen können, benannt. Beachten Sie, daß diese Funktion u. U. überhaupt nicht richtig funktioniert, wenn die Dateien stark fragmentiert sind, also Header und Footer nicht abwechselnd auftreten.

Datenträger klonen

RAM-Editor

Text-Editor aufrufen: Startet den von Ihnen unter Allgemeine Optionen angegebenen Text-Editor und lädt darin die aktuelle Datei. Wenn Sie mit dem Text-Editor Änderungen an dieser Daten vornehmen, können diese anschließend von WinHex übernommen werden.

Rechner: Startet den Windows-Rechner für sonstige Berechnungen. Dazu muß sich die Datei »calc.exe« im Windows-Ordner befinden.

Umrechnung: Diese Funktion können Sie benutzen, um Zahlen des Hexadezimal-Systems ins Dezimal-System oder umgekehrt zu übersetzen. Nach der Eingabe der Zahl betätigen Sie die ENTER-Taste. Bitte beachten Sie die Hinweise unter »Ganzzahlige Datentypen«.

Tabellen: Diese Funktion stellt Ihnen Übersichtstabellen zur Verfügung, in denen Sie zu Hexadezimal-Werten von 0 bis FF die Entsprechungen in Dezimalschreibweise, im IBM-ASCII-, ANSI-ASCII- und EBCDIC-Format ablesen können.

Block/Datei/Sektoren analysieren: Die Daten im aktuellen Block, in der gesamten Datei bzw. auf dem gesamten Datenträger werden statistisch ausgewertet und das Ergebnis in einem Fenster graphisch veranschaulicht. WinHex ermittelt dazu die Häufigkeiten des Vorkommens aller 256 möglichen Bytewerte und bildet sie proportional in vertikalen Linien entsprechender Länge ab. Dabei wird die Höhe des Fensters optimal genutzt, d. h. die längste Linie (die den häufigsten Bytewert repräsentiert) reicht von unten bis zur Titelleiste des Fensters. Unter der Titelleiste können Sie abhängig von der Mauscursor-Position den relativen Anteil und die absolute Anzahl eines jeden Bytewerts ablesen.
Diese Funktion kann z. B. dazu eingesetzt werden, um Datenmaterial unbekannter Art zu analysieren. Audio-Daten, komprimierte Daten, ausführbarer Code u. a. lassen sich an charakteristische Grafiken erkennen. Im Systemmenü des Fensters läßt sich einstellen, ob Bytes mit dem Wert Null unberücksichtigt bleiben sollen. Dies kann in vielen Fällen die Aussagekraft der Grafik stark erhöhen.
Außerdem wird eine Standard-Prüfsumme (Summe aller Bytes) und der sicherere CRC32 angegeben.

Digest berechnen: Berechnet für die aktuelle Datei, den aktuellen Datenträger bzw. den gegenwärtig definierten Block den 128 Bit großen, verbreitet angewandten MD5 Message-Digest. Verwenden Sie die Tastenkombination Shift+F2, um statt dessen den 256 Bit großen PSCHF-Digest zu berechnen.

Routinen-Manager: s. u. »Routinen-Manager«

Routine ausführen: Läßt sie eine Routine auswählen, die ausgeführt wird. D. h. der in der Routine definierte automatisierte Vorgang wird gestartet. Die angegebene Datei wird wie gewohnt geladen und alle von Ihnen angegebenen Änderungen werden sofort vorgenommen.

Optionen-Menü

Allgemeine Optionen

Rückgängig-Optionen

Sicherheitsoptionen

Daten-Dolmetscher-Optionen

Viewer statt Editor: Wenn Sie WinHex lediglich zum Einsehen von Dateien benutzen möchten, also ohne die Möglichkeit zum Editieren, dann können Sie diese Menüoption aktivieren. Alle Dateien werden dann im View-Modus geöffnet, nur neu angelegte Dateien sind weiterhin editierbar.

In-Place-Editor: Alle Dateien werden im In-Place-Editiermodus geöffnet, d. h. alle Änderungen, die Sie vornehmen, werden sofort auf den Datenträger geschrieben.

Zeichensatz: An diesem Menüeintrag oder mit der Tastenkombination Shift+F7 kann eingestellt werden, ob für die Textdarstellung der ANSI-ASCII-, der IBM-ASCII- oder der EBCDIC-Zeichensatz verwendet wird. Wie unter Punkt »ANSI-/IBM-ASCII« erläutert, ist der ANSI-Zeichensatz bei der Bearbeitung von Windows-Dateien vorzuziehen. EBCDIC ist auf IBM-Mainframe-Rechnern gebräuchlich. EBCDIC kann in WinHex nicht zum Drucken verwendet werden.

Nur Text-Anzeige: Blendet die Hexadezimal-Anzeige aus und verwendet die gesamte Breite des Editorfensters für die Text-Anzeige.

Nur Hex-Anzeige: Blendet die Text-Anzeige aus und verwendet die gesamte Breite des Editorfensters für die Hexadezimal-Anzeige.

Dateimanager

Ausführen: Sie können eine bestehende Datei wählen, die ausgeführt werden soll. Wenn es sich nicht um eine ausführbare Datei handelt, so wird die Anwendung geladen, mit der sie unter Windows verknüpft ist.

Zerlegen: Wählen Sie eine bestehende Datei, aus der Sie mehrere neue Dateien bilden möchten. Geben Sie für jede Zieldatei den Dateinamen an und den Offset der Quelldatei, an dem die Trennung vorgenommen werden soll. Die Quelldatei bleibt durch diese Funktion unberührt.

Verketten: Diese Funktion läßt Sie eine beliebige Anzahl bestehender Dateien auswählen, die aneinandergehängt eine Zieldatei bilden.

Verschmelzen: Geben Sie die Namen zweier Quelldateien und einer Zieldatei an. Die Bytes bzw. Words der Quelldateien werden abwechselnd in die Zieldatei geschrieben (wobei das erste Byte/Word aus der ersten Quelldatei stammt). Auf diese Weise lassen sich die in getrennten Dateien enthaltenen Odd- und Even-Bytes bzw. -Words zu einer Datei zusammenfügen (z. B. in der EPROM-Programmierung).

Aufspalten: Geben Sie die Namen einer Quelldatei und zweier Zieldateien an. Die Bytes bzw. Words der Quelldatei werden abwechselnd in die Zieldateien geschrieben (wobei das erste Byte/Word in die zuerst ausgewählte Zieldatei gelangt). Auf diese Weise lassen sich Odd- und Even-Bytes bzw. -Words in zwei separate Dateien überführen (z. B. in der EPROM-Programmierung).

Vergleichen: Wählen Sie zwei bestehende Dateien aus, die Sie Byte für Byte vergleichen möchten. Geben Sie außerdem den Namen der Datei an, in die der Bericht geschrieben werden soll. Bestimmen Sie, ob nach *Unterschieden* oder nach *Übereinstimmungen* gesucht werden soll. Es ist möglich, eine Anzahl von Unterschieden/Übereinstimmungen anzugeben, bei deren Erreichen der Vergleich abgebrochen werden soll. Als letztes ist der Bereich einzustellen, in dem verglichen werden soll. Sobald Sie Dateien ausgewählt haben, wird als Ende dieses Bereichs automatisch das Ende der kleineren Datei eingetragen.
WinHex erstellt einen Bericht in Form einer Textdatei, den Sie mit dem unter Optionen gewählten Texteditor einsehen können. Bei großen Vergleichsbereichen und vielen Unterschieden/Übereinstimmungen kann diese Textdatei sehr groß werden.

Kopieren: Kopiert eine bestehende Datei in einen anderen Ordner.

Verschieben: Bewegt eine bestehende Datei in einen anderen Ordner und/oder ändert ihren Namen. Dabei wird die Ursprungsdatei entfernt.

Sicheres Löschen: Löscht eine Datei definitiv, so daß ihr Inhalt mit Undelete-Programmen nicht rekonstruiert werden kann. Jede gewählte Datei wird (in ihrer aktuellen Größe) mit Nullen überschrieben, auf die Länge Null gekürzt und dann gelöscht. Ihr Name wird unkenntlich gemacht. Selbst manuelle Wiederherstellungsversuche führen i. d. R. zu keinem Ergebnis. »Sicheres Löschen« eignet sich daher für Dateien mit vertraulichen Informationen, die vernichtet werden sollen. Die *absolute* Sicherheit dieser Funktion kann aber *nicht* garantiert werden, insbesondere dann nicht, wenn residente Undelete- und Backup-Programme mit ausgeklügelten Mechanismen die Wiederherstellbarkeit von Dateien sicherzustellen versuchen.

Hilfe-Menü

Inhalt: Ruft die Inhaltsübersicht dieser Hilfe-Datei auf.

Sprache wechseln: Schaltet zwischen Englisch und Deutsch um.

Initialisieren: Mit dieser Funktion können Sie die Voreinstellungen sämtlicher Optionen wiederherstellen. Alternativ dazu können Sie die Datei »winhex.cfg« löschen, bevor Sie WinHex starten.

Deinstallieren: Mit dieser Funktion können Sie die WinHex von Ihrer Festplatte entfernen, selbst wenn Sie nicht das Setup-Programm zur Installation verwendet haben.

Homepage: Lädt die WinHex-Homepage (Internet-Adresse http://www.winhex.com) in Ihrem Browser.

Drucken

Mit dieser Funktion des Datei-Menüs können Sie einen Ausschnitt aus einem Editierfenster drucken. Geben Sie den Druckbereich in Form von Offsets an. Sie haben die Möglichkeit, einen Drucker auszuwählen und ihn einzurichten.

Bestimmen Sie den Zeichensatz für den Druck, ändern Sie ggf. die vorgeschlagene Schriftgröße und tragen Sie auf Wunsch einen Kommentar, der am Ende des Ausdruckes erscheinen soll, in das dafür vorgesehene Feld ein. Die empfohlene Schriftgröße berechnet sich als Druckauflösung (z. B. 720 dpi) geteilt durch 6 (z. B. 120).

Wenn Ihnen das Drucken mit WinHex nicht flexibel genug ist, können Sie auch einen Block definieren, ihn mit »Bearbeiten->Kopieren->Editoranzeige« als Hex-Editor-formatierten Text in die Zwischenablage kopieren und in einem Textverarbeitungsprogramm weiterverwenden. Dort eignet sich dann besonders die Schriftart »Courier New«, Größe 10, zum Ausdruck auf DIN A4.

Block

Als »Block« wird ein ausgewählter Bereich bezeichnet, der für jede in WinHex geöffnete Datei festgelegt werden kann. Dieser Bereich ist Gegenstand vieler Funktionen im Bearbeiten-Menü, genau wie Markierungen in anderen Windows-Programmen. Wenn kein Block definiert ist, beziehen sich diese Funktionen gewöhnlich auf den gesamten Dateiinhalt.

Die aktuelle Lage und Größe des Blocks werden in der Statusleiste angezeigt. Durch Drücken der Escape-Taste oder durch einen Doppelklick mit der rechten Maustaste hebt man die Blockmarkierung auf.

Daten modifizieren

Mit dieser Funktion können Sie die Daten im aktuellen Block bzw. in der gesamten Datei (falls kein Block definiert ist) verändern. Entweder Sie *addieren* zu jedem Element der Daten eine Zahl, Sie *invertieren* die Bits, Sie führen eine XOR-Operation mit einer Konstanten aus (eine einfache Art der Verschlüsselung), Sie shiften Bits logisch oder Sie *vertauschen* Bytes paarweise. Durch das Shiften (Verschieben) von Bits können Sie das Einfügen oder Entfernen eines einzelnen Bits am Anfang des Blockes simulieren.

Bytes vertauschen

Vertauscht benachbarte Bytes paarweise (16-Bit-Vertauschung) oder in 4er-Gruppen (32-Bit-Vertauschung) innerhalb des aktuellen Blocks bzw. innerhalb der gesamten Datei, wenn kein Block definiert ist. Der Bereich muß dazu ein Vielfaches von 2 (16-Bit-Vertauschung) bzw. 4 (32-Bit-Vertauschung) Bytes enthalten. Mit dieser Funktion können Sie »Big-Endian«-Daten in »Little-Endian«-Daten verwandeln.

Addition

Geben Sie einen positiven oder negativen, dezimalen oder hexadezimalen Summanden an, der jedem Datenelement des Blockes hinzuaddiert werden soll. Der numerische Datentyp bestimmt Größe (1, 2 oder 4 Bytes) und Art (vorzeichenbehaftet oder vorzeichenlos) eines Elements.

Es werden zwei Möglichkeiten angeboten, wie WinHex verfahren soll, wenn durch die Addition der Wertebereich des Formats über- oder unterschritten würde. Entweder der Wertebereich wird nicht verlassen, d. h. das Maximum bzw. Minimum des Wertebereichs wird als neuer Wert angenommen (I), oder die Addition wird dennoch durchgeführt und der entstehende Übertrag ignoriert (II).

Beispiel: 8 Bit, vorzeichenlos
I. FF + 1 = FF (255 + 1 = 255)
II. FF + 1 = 00 (255 + 1 = 0)

Beispiel: 8 Bit, vorzeichenbehaftet
I. 80 - 1 = 80 (-128 - 1 = -128)
II. 80 - 1 = 7F (-128 - 1 = +127)

Hinweise:

• Bei Verwendung der ersten Methode erhalten Sie nach Abschluß der Operation eine Meldung, wie oft die Addition nicht durchgeführt werden konnte.
• Wenn Sie die zweite Methode verwenden, ist der Vorgang umkehrbar. Geben Sie einfach die Gegenzahl des zuvor benutzten Summanden bei gleichem Zahlenformat ein. Sie erhalten dann exakt die ursprünglichen Daten.
• Bei Wahl der zweiten Methode ist es egal, ob Sie ein vorzeichenbehaftetes oder vorzeichenloses Format angeben.

Konvertierungen

WinHex erlaubt es, mit dem Befehl »Konvertieren« im Bearbeiten-Menü Daten in andere Formate umzuwandeln, zu verschlüsseln und zu entschlüsseln. Die Konvertierung kann optional in allen in WinHex geöffneten Dateien statt nur in der aktuellen Datei durchgeführt werden. Die mit einem Stern gekennzeichneten Formate können nie blockweise, sondern nur dateiweise konvertiert werden. Die folgenden Formate werden unterstützt:

• ANSI-ASCII, IBM-ASCII (zwei sich teilweise unterscheidende ASCII-Zeichensätze)
• EBCDIC (ein IBM-Mainframe-Zeichensatz)
• Groß-/Kleinbuchstaben (ANSI-ASCII)
• Binär* (Rohdaten)
• Hex-ASCII* (Hexadezimal-Darstellung von Rohdaten als ASCII-Text)
• Intel-Hex* (=Extended Intellec; Hex-ASCII-Daten in einem speziellen Format, incl. Prüfsummen etc.)
• Motorola-S* (=Extended Exorcisor; dto.)

Bitte beachten Sie:
• Beim Konvertieren von Intel-Hex oder Motorola-S in ein anderes Format werden die in den Daten enthaltenen Prüfsummen nicht auf Korrektheit überprüft.
• In Abhängigkeit von der Dateigröße wird der kleinstmögliche Subtyp in der Ausgabe verwendet: Intel-Hex: 20-Bit oder 32-Bit. Motorola-S: S1, S2 oder S3.
• Einige Konvertierungsarten können nur auf eine ganze Datei angewandt werden.

Verschlüsselung/Entschlüsselung

Es wird empfohlen, einen Schlüssel zu verwenden, der aus mind. 8 Zeichen besteht. Widerstehen Sie der Versuchung, ein Wort aus einer belieben Sprache zu wählen. Am besten ist eine zufällige Kombination von Buchstaben, Satzzeichen und Ziffern. Beachten Sie, daß bei Groß- und Kleinbuchstaben unterschieden werden. Es ist unmöglich, ohne den richtigen Schlüssel die verschlüsselten Daten wiederherstellen zu können. Der zur Entschlüsselung eingegebene Schlüssel wird nicht auf Korrektheit überprüft.

Als Verschlüsselungsalgorithmus wird »Pukall Cipher 1« (PC 1) mit einem 128-Bit-Schlüssel benutzt (dem 128-Bit-Digest des von Ihnen angegebenen Schlüssels).

Suchoptionen

Groß-/Kleinschreibung beachten: Wenn diese Option gewählt ist, wird der Text immer in genau der Schreibweise gesucht, in der Sie ihn vorgeben. Z. B. wird »Beispiel« nicht bei der Vorgabe »beispiel« gefunden. Wenn Sie die Option nicht wählen, so wird WinHex selbst bei »bEIsPiEl« fündig. Auch deutsche Umlaute, die im ANSI-ASCII-Format vorliegen, sind dann in der Groß- und Kleinschreibung austauschbar, sonstige sprachspezifischen Buchstaben (çåê...) allerdings nicht.

Im Unicode-Zeichensatz suchen: Der Text wird im Unicode-Zeichensatz gesucht. Dieser Zeichensatz reserviert 16 Bit je Zeichen, wobei die ersten 256 Unicode-Zeichen den ANSI-ASCII-Zeichen entsprechen. Das höherwertige Byte ist dabei Null. In 32-Bit-Programmdateien z. B. sind Texte teilweise im Unicode-Zeichensatz gespeichert.

Sie können ein frei wählbares **Jokerzeichen** (ein Zeichen bzw. ein zweistelliger Hex-Wert) verwenden, das genau ein Byte abdecken kann. Z. B. kann man mit der Such-Zeichenfolge »Sp?ck« sowohl »Speck« als auch »Spock« finden.

Nur ganze Wörter suchen: Die zu suchende Zeichenfolge wird nur erkannt, wenn sie als einzelnes Wort vorkommt, also von anderen Buchstaben (z. B. durch Leer- oder Steuerzeichen) getrennt ist.

Suchrichtung: Bestimmen Sie, ob von vorne bis hinten oder von der aktuellen Position an ab- oder aufwärts gesucht werden soll.

Bedingung: Offset modulo x = y: Der Suchalgorithmus erfaßt nur Vorkommnisse an Offsets, die die genannte Bedingung erfüllen. Wenn Sie bspw. Daten suchen, von denen Sie wissen, daß sie an Position 10 eines Festplatten-Sektors stehen, geben Sie x=512, y=10 an. Wenn Sie DWORD-ausgerichtete Daten suchen, verwenden Sie x=4, y=0, um irrelevante Treffer auszuschließen.

Nur im Block suchen: Es wird nur derjenige Teil der Datei/des Datenträgers/des virtuellen Speichers durchsucht, der innerhalb des Blockes liegt.

In allen geöffneten Dateien suchen: Die Suche wird der Reihe nach in allen von WinHex geöffneten Dateien durchgeführt. Wird der Hex-Editor in einer Datei fündig, kann die Suche danach in derselben Datei normal fortgesetzt werden (durch F3); zur nächsten Datei geht WinHex mit der Funktion »Globale Suche fortsetzen« (F4) über. Wenn »Nur im Block suchen« aktiviert ist, wird in jeder Datei nur der dort festgelegte Block durchsucht.

Fundstellen zählen (und speichern): Die Anzahl der Vorkommnisse des gesuchten Texts/der gesuchten Hex-Werte in der Datei/auf dem Datenträger/im virtuellen Speicher wird ermittelt. Die Positionen der Vorkommnisse werden ggf. im Positions-Manager gespeichert, so daß sie zu einem späteren Zeitpunkt wiedergefunden und bearbeitet werden können.

Suchen-Menü
Optionen des Ersetzens
Technische Hinweise

Ersetzen-Optionen

Auf Bestätigung warten: An jeder Fundstelle entscheiden Sie, ob dort ersetzt und ob der Vorgang evtl. abgebrochen werden soll.

Alles ersetzen: Alle Vorkommnisse werden automatisch ersetzt.

Groß-/Kleinschreibung beachten: Bei der Suche nach der zu ersetzenden Zeichenfolge kann nach Groß- und Kleinschreibung unterschieden werden (s. a. Suchoptionen). WinHex verwendet die Ersatz-Zeichenfolge natürlich in jedem Fall in der von Ihnen gewählten Schreibweise.

Unicode-Zeichensatz verwenden: Der Text wird im Unicode-Zeichensatz gesucht. Dieser Zeichensatz reserviert 16 Bit je Zeichen, wobei die ersten 256 Unicode-Zeichen den ANSI-ASCII-Zeichen entsprechen. Das höherwertige Byte ist dabei Null. In 32-Bit-Programmdateien beispielsweise sind Texte teilweise im Unicode-Zeichensatz gespeichert.

Sie können ein beliebiges Zeichen bzw. einen beliebigen zweistelligen Hex-Wert als **Jokerzeichen** verwenden. Z. B. kann man mit der Such-Zeichenfolge »Sp?ck« sowohl »Speck« als auch »Spock« finden.

In der Ersatz-Zeichenfolge kann das Jokerzeichen verwendet werden, um an den betreffenden Stellen das bestehende Zeichen nicht zu ändern. Auf diese Weise kann man bspw. »Huhn« und »Hahn« in einem Schritt durch »Hund« und »Hand« ersetzen (entsprechende Eingabe: »H?hn« ersetzen durch »H?nd«).
Ein Jokerzeichen, das im überstehenden Teil einer Ersatz-Zeichenfolge steht, die länger als die zugehörige Such-Zeichenfolge ist, wird selbst als Ersatz in die Datei geschrieben, da es kein bereits bestehendes Zeichen in der Datei gibt, das sich dem Jokerzeichen zuordnen läßt.

Ganze Wörter: Die zu suchende Zeichenfolge wird nur erkannt, wenn sie als einzelnes Wort vorkommt, also von anderen Buchstaben (z. B. durch Leer- oder Steuerzeichen) getrennt ist. Wenn diese Option gewählt ist, wird z. B. »Tomate« nicht in »Automaten« gefunden.

Suchrichtung: Bestimmen Sie, ob von vorne bis hinten oder von der aktuellen Position an ab- oder aufwärts ersetzt werden soll.

Nur im Block suchen: Es wird nur derjenige Teil der Datei/des virtuellen Speichers durchsucht, der innerhalb des Blockes liegt.

In allen geöffneten Dateien ersetzen: Der Vorgang wird der Reihe nach in allen von WinHex geöffneten Dateien durchgeführt (sofern sie nicht im View-Modus geöffnet wurden). Wenn »Nur im Block suchen« aktiviert ist, wird in jeder Datei nur im dort festgelegten Block ersetzt.

Mit WinHex sind Sie in der Lage, eine Zeichenfolge durch eine andere Zeichenfolge unterschiedlicher Länge zu ersetzen. Solche Vorgänge benötigen allerdings mehr Zeit und im Ersetzen-Modus mit Bestätigung sind die Änderungen nicht sofort sichtbar. Immer, wenn Sie diese Möglichkeit nutzen möchten, können Sie bestimmen, auf welche Art dies geschehen soll:

1. Die Dateiinhalte hinter einem Vorkommnis der Suchzeichenfolge werden entsprechend der Längendifferenz von Such- und Ersatzzeichenfolge nach vorne oder hinten verschoben. Die Größe der Datei ändert sich. Viele Arten von Dateien (darunter ausführbare Dateien) werden dadurch unbrauchbar. Es ist sogar möglich, nichts als Ersatz-Zeichenfolge anzugeben. Jedes Vorkommen der Such-Zeichenfolge wird dann aus der Datei entfernt!

2. Die Ersatzzeichenfolge wird ungeachtet ihrer Länge dort in die Datei geschrieben, wo die Suchzeichenfolge gefunden wurde. Wenn die Ersatzzeichenfolge kürzer als Suchzeichenfolge ist, bleibt der hintere Teil des Vorkommnisses der Suchzeichenfolge in der Datei unverändert. Ist die Ersatzzeichenfolge länger, werden auch noch Daten hinter dem Vorkommnis mit dem überstehenden Teil der Ersatzzeichenfolge überschrieben (sofern das Dateiende nicht erreicht ist). Die Größe der Datei bleibt unverändert

Suchen-Menü
Suchoptionen
Technische Hinweise

Rückgängig-Optionen

Für den Befehl »Rückgängig« stehen Ihnen folgende Optionen zur Auswahl:

• Bestimmen Sie, wie viele nacheinander ausgeführte Aktionen ungeschehen gemacht werden können. Wichtig: Dies hat keinen Einfluß auf die Anzahl der umkehrbaren Tastatureingaben, die nur vom Arbeitsspeicher limitiert wird.

• Um Zeit und Speicherplatz auf der Festplatte zu sparen, können Sie ein Dateigrößenlimit angeben, oberhalb dessen keine Sicherungen mehr durchgeführt werden, so daß der »Rückgängig«-Befehl nur noch nach Tastatureingaben zur Verfügung steht.

• Automatisch angelegte Sicherungen für die Benutzung durch den »Rückgängig«-Befehl werden von WinHex selbständig beim Schließen der Datei gelöscht, falls die betreffende Option voll aktiviert ist. Ist sie nur halb aktiviert, werden sie erst bei Programmende gelöscht.

• Geben Sie für alle Arten von Editiervorgängen an, ob sie rückgängig gemacht werden können.

Positions-Manager

In dem »Positions-Manager« genannten Dialogfenster können unbegrenzt viele Datei- und Datenträger-Offsets mit Beschreibungen verwaltet werden. Wenn Sie etwa in einer Datei eine markante Stelle ausfindig gemacht haben, die evtl. häufig geändert werden muß, dann lohnt es sich, diese Stelle im Positions-Manager einzutragen. Sie können sie dann später schnell wiederfinden, ohne sie sich merken zu müssen. Klicken Sie auf »Neu«, geben Sie den Offset und anschließend eine Beschreibung (etwa »Hier ein Byte auf Null setzen!«) ein.

Klicken Sie die rechte Maustaste, um ein Kontextmenü zu erzeugen. Darin können Sie Positionen löschen, aus einer Datei laden oder in eine Datei speichern (letzteres auch als HTML). Wenn die Daten des Positions-Managers geändert wurden, werden sie nach dem Beenden von WinHex grundsätzlich in der Datei *WinHex.pos* im WinHex-Verzeichnis gespeichert.

Das Positions-Manager-Fenster läßt sich verkleinern. Daraufhin können Sie zwischen den Positionen in der gewählten Reihenfolge im Editierfenster mit den Tasten Strg+Links und Strg+Rechts hin- und herwechseln.

Das POS-Dateiformat ist auf der WinHex-Homepage http://www.winhex.com vollständig dokumentiert.

Routinen-Manager

Der Routinen-Manager ist ein multifunktionales Dialogfenster, das Ihnen alle existierenden Routinen anzeigt. Die Liste kann sortiert werden nach der Routinen-Nr., der Routinen-Bezeichnung und nach den Dateien, auf die sich die Routinen beziehen. Sie haben die Möglichkeit, bestehende Routinen zu bearbeiten, zu kopieren und zu löschen und können neue Routinen entwerfen.

Wenn Sie eine Routine löschen, werden diejenigen Routinen, die sich in den Speicherplätzen mit höherer Routinennummer befinden, um eine Position vorgezogen. Verweise auf diese Routinen als Folgeroutinen werden entsprechend korrigiert.

Sicherungs-Manager

In der wahlweise nach dem Erstellungszeitpunkt, dem Dateinamen oder dem Pfad geordneten Liste können Sie Sicherungen auswählen, die Sie wiederherstellen möchten. Ein neues Editierfenster zeigt daraufhin den Datei- bzw. Sektorinhalt vom Zeitpunkt der Sicherung an. Damit die Datei bzw. der Sektor auf dem Datenträger tatsächlich wiederhergestellt wird, muß sie bzw. er noch gespeichert werden. Im Fall von Sektoren haben Sie aber auch die Möglichkeiten, die Wiederherstellung direkt auf dem Datenträger zu vollziehen oder die Sektoren in eine Datei zu schreiben. Wahlweise können Sie auch das Ziel der Wiederherstellung von Sektoren (Datenträger und Sektornummer) ändern. Sie können außerdem optional nur einen Teil der Sektoren aus der Sicherung extrahieren (Sektoren am Anfang einer komprimierten Sicherung können allerdings nicht übersprungen werden). Wenn die Sicherung mit einer Prüfsumme und/oder einem Digest versehen war, werden die Daten erst auf Authentizität überprüft, bevor sie direkt auf den Datenträger geschrieben werden.

Mit Hilfe des Sicherungs-Managers können Sie außerdem Sicherungen löschen, die Sie nicht mehr benötigen. Die automatisch erzeugten Sicherungsdateien für die »Rückgängig«-Funktion werden von WinHex standardmäßig selbständig gelöscht (s. u. Rückgängig-Optionen).

Die Sicherungsdateien, die vom Backup-Manager verwaltet werden, heißen »Saved???.whx« und befinden sich in dem unter Allgemeine Optionen gewählten Ordner. An die Stelle von ??? tritt eine aus drei Ziffern bestehende eindeutige Identifikationsnummer, die im Sicherungs-Manager in der letzten Spalte angegeben ist. Eine vollständige Dokumenation des WHX-Dateiformats ist auf der WinHex-Homepage http://www.winhex.com verfügbar.

Daten-Dolmetscher

Der Daten-Dolmetscher ist ein kleines Fenster, das »Übersetzungsmöglichkeiten« für die Daten an der aktuellen Cursorposition anzeigt. In den Optionen können Sie einstellen, welche Datentypen zu berücksichtigen sind. Zur Verfügung stehen sieben ganzzahlige Datentypen, die Bit-Darstellung eines Bytes (Binärformat), vier Gleitkomma-Datentypen, Assembler-Opcodes (Intel) und fünf Datumstypen.

Der Dolmetscher kann alle Datentypen (außer Assembler-Opcodes) auch rückwärts wieder in Hex-Werte übersetzen. Doppelklicken Sie dazu im Daten-Dolmetscher auf die Darstellung eines Datentyps, tragen Sie den gewünschten Wert ein und bestätigen Sie mit ENTER. Daraufhin schreibt der Daten-Dolmetscher die entsprechenden Hex-Werte an der aktuellen Position in das Editierfenster.

Mit einem Klick der rechten Maustaste können Sie ein Kontextmenü im Daten-Dolmetscher aufrufen und darin einstellen, ob die ganzzahligen und Gleitkomma-Datentypen im Little- oder Big-Endian-Format übersetzt werden sollen.

Hinweise:

Nicht alle Hex-Werte können in Gleitkomma-Zahlen übersetzt werden. Wenn eine Übersetzung nicht möglich ist, erscheint die Angabe NAN (»**n**ot **a n**umber«) im Daten-Dolmetscher.

Ebensowenig können alle Hex-Werte als Datumswerte jeden Typs übersetzt werden. Manche Datumstypen haben stark eingeschränkte gültige Wertebereiche.

Redundanzen im Befehlssatz der Intel-Prozessoren schlagen sich in mehrfach vorkommenden Opcodes und mnemonischen Abkürzungen nieder. Floating-Point-Befehle werden im Daten-Dolmetscher nur als F*** angezeigt.

Beschreibungen der den mnemonischen Abkürzungen entsprechenden Befehle können von Intel über das Internet bezogen werden. Das Dokument heißt »Intel Architecture Software Developer’s Manual Volume 2: Instruction Set Reference« und liegt im PDF-Format vor.

RAM-Editor

Im Extra-Menü finden Sie die Funktion »RAM-Editor«. Der RAM-Editor ermöglicht es, den virtuellen Arbeitsspeicher eines in der Ausführung befindlichen Programms (= eines Prozesses) direkt einzusehen. Dazu werden alle von dem Prozeß belegten Seiten im Arbeitspeicher als zusammenhängender Speicherbereich abgebildet. Ungenutzte (leere oder nur reservierte) Blöcke im Speicher werden von WinHex ignoriert.

Wählen Sie zunächst aus einer Liste aller laufenden Prozesse den zu untersuchenden Prozeß aus. Sie können entweder auf den sog. Primärspeicher oder den Gesamtspeicher eines Prozesses oder auf einzelne von diesem Prozeß geladene Module zugreifen. Unter Windows 95/98 werden Systemmodule optional augelistet. Als Systemmodule werden diejenigen Module bezeichnet, die stets oberhalb von 2 GB geladen werden (wie z. B. kernel32.dll, gdi32.dll usw.). Als Primärspeicher wird derjenige Bereich bezeichnet, den Programme vorrangig für verschiedenste Zwecke nutzen. Zumindest das Hauptmodul eines Prozesses (die EXE-Datei) ist i. d. R. ebenfalls im Primärspeicher enthalten. Der Gesamtspeicher umfaßt den gesamten virtuellen Speicher eines Prozesses einschließlich dem gemeinsamen Speicherbereich aller Prozesse, bis auf die Systemmodule.

Bitte beachten Sie die folgenden Einschränkungen:

• Vorsicht: Rückgängig gemacht werden können *ausschließlich* Tastatureingaben!
• Der virtuelle Arbeitsspeicher von 16-Bit-Prozessen kann unter Windows 95/98 nur unvollständig, unter NT gar nicht erfaßt werden.
• Das Editieren ist nur im In-Place-Editiermodus möglich.
• Systemmodule von Windows 95/98 können nur im View-Modus *eingesehen*, nicht editiert werden.
• Die Demo-Version erlaubt generell nur den View-Modus! Bestellen Sie die Vollversion.

Beachten Sie bitte die Optionen »Auf Änderungen im Speicher prüfen« (Sicherheitsoptionen) und »Virtuelle Adressen« (Allgemeine Optionen).

Kontextmenü

Das Kontextmenü sehen Sie, wenn Sie im Windows-Explorer oder auf dem Desktop ein Objekt mit der rechten Maustaste anklicken. WinHex erscheint im Kontextmenü nur, wenn die entsprechenden Optionen eingeschaltet sind.

Editieren mit WinHex: Öffnet die gewählte Datei in WinHex.

Ordner in WinHex öffnen: Läßt Sie alle Dateien des gewählten Ordners in WinHex öffnen (wie „Erweitertes Öffnen" im Datei-Menü)

Datenträger editieren: Öffnet den gewählten Datenträger im Disk-Editor von WinHex. Wenn Sie die Shift-Taste gedrückt halten, wird statt des logischen Laufwerks der zugehörige physische Datenträger geöffnet (letzteres nur unter Windows 95/98).

WinHex stellt in der Statusleiste, im Daten-Dolmetscher und im Positions-Manager eigene Kontextmenüs zur Verfügung.

Schlüssel

Als Schlüssel geben Sie eine Zeichenfolge aus 1-16 Zeichen ein. Je mehr Zeichen Sie eingeben, umso sicherer ist die Verschlüsselung. Der Schlüssel wird nicht direkt als für Ver- und Entschlüsselung benutzt, sondern ist nur Datenmaterial für einen Digest. Er wird nicht auf der Festplatte gespeichert. Fall die entsprechende Sicherheitsoption gewählt ist, wird er in verschlüsselter Form im Arbeitsspeicher gehalten, solange WinHex läuft.

Sicherung anlegen

Dieser Befehl stellt Ihnen ein Dialogfenster zu Verfügung, das Sie eine Sicherheitskopie von der Datei bzw. des Datenträgers (Drive-Imaging) erstellen läßt. Die Sicherheitskopie wird in Form einer WHX-Datei angelegt.

Im Falle eines Datenträgers können Sie die zu sichernden Sektoren bestimmen. Standardmäßig sind alle Sektoren von der aktuellen Position bis zum Ende des Datenträgers eingetragen. Wahlweise läßt sich eine Datenträger-Sicherung in mehrere Archive aufteilen. Teilsicherungen der Größe 650 MB bspw. eignen sich zur Archivierung auf CD-R. Wenn Sie die Sicherung möglichst platzsparend anlegen möchten, initialisieren Sie vorher unbenutzte Bereiche des Datenträgers, denn mit Nullen gefüllte Sektoren beanspruchen bei eingeschalteter Komprimierung praktisch keinen Platz in der Sicherung.

Hinweis zu Disk-Cloning & -Imaging

Im Falle eines Datenträgers wird der gegenwärtig dargestellte Sektor gesichert. Wenn Sie mehr als einen Sektor sichern lassen, werden zusätzlich die folgenden Sektoren gesichert. Wenn Sektor 0 der aktuelle Sektor ist und sie so viele Sektoren sichern lassen, wie auf dem Datenträger enthalten sind, wird der Datenträger komplett gesichert.

Wenn Sie den Namen der WHX-Datei automatisch vergeben lassen (Format »Saved???.whx«), wird sie im Verzeichnis für Sicherungsdateien erstellt (s. Allgemeine Optionen). Bei Bedarf kann das Original dann mit dem Sicherungsmanager wiederhergestellt werden. Wenn Sie selbst Dateinamen und Pfad angeben, kann die WHX-Datei mit dem Menübefehl »Sicherung laden« wiederhergestellt werden.

Sie können der Sicherung eine textuelle Beschreibung hinzufügen.

Das WHX-Format kann außerdem eine Prüfsumme (CRC32) und einen Digest der zu sichernden Daten aufnehmen. Die zu sichernden Daten können optional verschlüsselt und/oder komprimiert werden. Die Berechnung eines Digests und die Verschlüsselung verlangsamen das Anlegen der Sicherung sowie die spätere Wiederherstellung. Beide Funktionen sollten nur benutzt werden, wenn es zu Sicherheitszwecken erforderlich ist. Wenn Sie ein Backup wiederherstellen, das mit einem Digest versehen wurde, und WinHex dabei keine Warnung ausgibt, können Sie völlig sicher sein, daß die Sicherung nach Erstellung nicht manipuliert wurde.

Als Verschlüsselungsalgorithmus wird »Pukall Cipher 1« (PC 1) mit einem 128-Bit-Schlüssel benutzt. Der 128-Bit-Schlüssel ist der Digest aus der 256-Bit-Konkatenation, die aus dem 128-Bit-Digest des von Ihnen angegebenen Schlüssels und einer 128-bittigen Zufallszahl besteht. Die 128-bittige Zufallszahl wird in der WHX-Datei für die Entschlüsselung gespeichert.

Zum Komprimieren wird der verbreitete Deflate-Algorithmus der zlib-Bibliothek verwendet. Er basiert auf LZ77-Kompression und Huffman-Codierung. Die Kompressionsrate ist dieselbe wie bei der ZIP-Komprimierung.

Das WHX-Dateiformat ist vollständig dokumentiert (s. WinHex-Homepage http://www.winhex.com).

Ihre Freischaltcodes galten wahrscheinlich für eine *frühere* Version von WinHex. Für die vorliegende Version sind sie *nicht mehr gültig*.

Hinweise zu Updates finden Sie unter http://www.winhex.com/winhex/upgrade-d.html. Wenn Sie registrierter Benutzer sind und für WinHex 9.0 oder neuer bezahlt haben, ist das Update auf die vorliegende Version für Sie kostenlos. In diesem Fall wenden Sie sich bitte unter Angabe Ihres Namens und Ihrer Anschrift per E-Mail an den Autor (mail@sf-soft.de).

Digests

Ein »Digest« (engl.) ist ähnlich einer Prüfsumme eine Kennzahl zur eindeutigen Identifizierung von Daten. Digests sind aber mehr als Prüfsummen. Es handelt sich um »starke« Einweg-Hashcodes, die Datenintegrität mit extrem hoher Sicherheit garantieren. Daten können mit computerunterstütztem Rechenaufwand in bösartiger Absicht so manipuliert werden, daß ihre Prüfsumme trotz Änderung gleich bleibt. Dies kann fälschlicherweise zu der Annahme verleiten, die Daten seien noch im Originalzustand. Diese Möglichkeit schließen Digests aus. Es lassen sich mit vorstellbarem computerunterstütztem Rechenaufwand keine Daten finden, die denselben Digest besitzen wie vorgegebene andere Daten.

Natürlich können durch Verwendung von Digests auch zufällige, etwa durch fehlerhafte Übertragung entstandene Datenveränderungen festgestellt werden, aber dafür reichen Prüfsummen aus, die viel schneller berechnet werden können.

WinHex verwendet 128-Bit- und 256-Bit-Digests als Schlüssel für Datenverschlüsselung (s. Konvertierungen u. Sicherung anlegen) sowie 256-Bit-Digests zum Zwecke der Datenauthentizitätsprüfung in Sicherungsdateien. Als Algorithmus wird die »Pukall Stream Cipher«-Hashfunktion mit folgendem fixen 128-Bit-Schlüssel benutzt: F6 C7 24 95 17 9F 3F 03 C6 DE F1 56 F8 2A 85 38.

WinHex beherrscht darüber hinaus den 128 Bit großen MD5 Message-Digest.

Sicherheitsoptionen

• Die Option »**Alleinige Datenträgerkontrolle**« betrifft den Disk-Editor. Sie sorgt dafür, daß WinHex einen Datenträger während des direkten Zugriffs für andere Programme *grundsätzlich* sperrt (nicht nur, wenn unbedingt erforderlich). Dies kann den Disk-Editor stark verlangsamen.

• »**Caching beim Lesen von Sektoren**« beschleunigt den sequentiellen Datenträgerzugriff mit dem Disk-Editor. Diese Option empfiehlt sich insbes. beim Durchsehen von CD-ROM- und Disketten-Sektoren, da sie die Zahl der erforderlichen physischen Zugriffe stark herabsetzt.

• Die Option »**Clusterketten automatisch einlesen**« sorgt dafür, daß WinHex die Cluster eines FAT16- oder FAT32-Laufwerks selbständig untersucht, wenn ein solches Laufwerk geöffnet wird und die benötigten Informationen noch nicht vorliegen. Dadurch kann WinHex anzeigen, wofür Sektoren/Cluster verwendet werden. Benutzen Sie die Funktion »Cluster inspizieren« im Extra-Menü, um diese Informationen zu aktualisieren.

• Wenn die Option »**Daten über Clusterketten speichern**« eingeschaltet ist, bleiben die Informationen, die WinHex über die Clusterketten von FAT16- und FAT32-Laufwerken gesammelt hat, beim Beenden von WinHex im Ordner für temporäre Dateien erhalten. WinHex kann sie dann beim nächsten Programmstart wiederverwenden.

• Die Option »**Auf Änderungen im Speicher prüfen**« betrifft den RAM-Editor. Sie sorgt dafür, daß WinHex vor jedem Lesen und Beschreiben des virtuellen Speichers erst prüft, ob sich dessen Größe und Zusammensetzung geändert hat. Ist dies der Fall, wird der Speicher in WinHex neu abgebildet und damit ein möglicher Lesefehler vermieden. Besonders unter Windows NT kann diese Einstellung den RAM-Editor stark verlangsamen. Beim Editieren des *Gesamtspeichers* eines Prozesses wird unabhängig von der gewählten Einstellung nicht auf Änderungen geprüft.

• Auf Wunsch wird beim Öffnen einer Datei eine Standard- oder CRC32-Prüfsumme berechnet und in der Informationsspalte rechts angezeigt. **Prüfsummen** werden auch in der Funktion »Block analysieren« berechnet.

• In der Voreinstellung müssen Sie das **Speichern von** Änderungen an existierenden **Dateien bestätigen**. Wenn Sie diese Option ausschalten, entfällt die Sicherheitsabfrage.

• Beim manuellen Wiederherstellen von Sicherungen wird ein Bericht i. d. R. nur dann angezeigt, wenn die Sicherungsdatei einen Digest enthält oder fehlerhaft ist. Sie können sich jedoch auch grundsätzlich einen **Bericht anzeigen** lassen. Wenn diese Option gewählt ist, wird Ihnen auch der Digest angezeigt.

• Den für Verschlüsselung und Entschlüsselung erforderliche **Schlüssel** können Sie entweder in ein normales Editierfeld **eingeben** oder **blind** (es erscheinen nur Sternchen). In letzterem Fall müssen Sie den Schlüssel bestätigen, um Eingabefehler zu vermeiden.

• Standardmäßig wird der **Schlüssel** selbst verschlüsselt **im** Arbeits**speicher** ge**halten**, solange WinHex läuft, damit Sie ihn nicht mehrmals eingeben müssen, wenn Sie ihn mehrmals verwenden möchten. Möglicherweise ziehen Sie es aber vor, daß WinHex sich den Schlüssel **nicht** merkt.

Endian-ness

Mikroprozessoren unterscheiden sich in der Position des niederwertigsten Bytes. In Systemen mit Intel®- und MIPS®-Processoren steht das niederwertigsten Byte an erster Stelle. Daten eines aus mehreren Bytes bestehender Datentyps (z. B. 32-Bit-Integertyp, Unicode-Zeichen) stehen im Speicher beginnend mit dem niederwertigsten (»little end«) und endend mit dem höherwertigstem Bytes. Zum Beispiel wird die Hexadezimalzahl 12345678 als 78 56 34 12 gespeichert. Dies wird das **Little-Endian**-Format genannt.

Motorola-Processoren setzen voraus, daß das niederwertigste Byte an hinterster Stelle steht. Mehrfach-Byte-Daten werden beginnend mit dem höchstwertigen Byte (»big end«) und endend mit dem niederstwertigem Byte gespeichert. Zum Beispiel wird die Hexadezimalzahl 12345678 als 12 34 56 78 gespeichert. Dies wird das **Big-Endian**-Format genannt.

Datumstypen

Die folgenden Datumsformate werden vom Daten-Dolmetscher unterstützt.

MS-DOS Datum & Zeit (4 Bytes)

Das niederwertige Word bestimmt die Zeit, das höherwertige das Datum. Wird von zahlreichen DOS-Funktionen und von allen FAT-Dateisystemen benutzt.

Bits	Inhalt
0-4	Sekunden geteilt durch 2
5-10	Minuten (0-59)
11-15	Stunde (0-23)
16-20	Tag (1-31)
21-24	Monat (1=Januar, 2=Februar usw.)
25-31	Jahre seit 1980

Win32 FILETIME (8 Bytes)

Ein ganzzahliger 64-Bit-Wert, der die Anzahl der seit dem 1. Januar 1601 vergangenen 100-Nanosekunden-Intervalle angibt. Wird in der Win32-API benutzt.

OLE 2.0 Datum & Uhrzeit (8 Bytes)

Ein Gleitkommawert (double), dessen ganzzahliger Bestandteil die Zahl der seit dem 30. Dezember 1899 vergangenen Tage angibt (Datum). Der Bruchanteil wird als die Uhrzeit interpretiert (z. B. 1/4 = 6:00 Uhr). Dies ist der OLE-2.0-Standarddatumstyp. Er wird bspw. auch von MS Excel verwendet.

ANSI SQL Datum & Uhrzeit (8 Bytes)

Zwei aufeinanderfolgende ganzzahlige 32-Bit-Werte. Der erste gibt die Anzahl der seit dem 17. November 1858 vergangenen Tage an (Datum). Der zweite bestimmt die Anzahl der seit Mitternacht vergangenen 100-Mikrosekunden-Intervalle (Uhrzeit). Dieser Datumstyp ist ANSI-SQL-Standard und wird in Datenbanken verwendet (u. a. in InterBase 6.0).

UNIX/C Datum & Uhrzeit (4 Bytes)

Ein ganzzahliger 32-Bit-Wert, der die Anzahl der seit dem 1. Januar 1970 vergangenen Sekunden angibt. Dieser Datumstyp wird bzw. wurde in UNIX, in C und C++ (»time_t«) sowie in FORTRAN-Programmen in den 80er Jahren verwendet. Gelegentlich ist er auch definiert als die Anzahl der seit dem 1. Januar 1970 vergangenen *Minuten*. In den Optionen des Daten-Dolmetschers läßt sich die verwendete Zeiteinheit einstellen.

Master-Boot-Record

Der **Master-Boot-Record** befindet sich am physischen Anfang einer Festplatte (editierbar mit dem Disk-Editor). Er besteht aus einem 446 Bytes langen **Master-Bootstrap-Loader-Code** und vier aufeinanderfolgenden, identisch aufgebauten **Partitions-Records**. Abschließend folgt die Hexadezimal-Signatur 55AA, die einen gültigen Master-Boot-Record kennzeichnet.

Das Format eines Partitions-Record sieht wie folgt aus:

Offset	Größe	Beschreibung
0	8 Bit	Der Hexadezimal-Wert 80 kennzeichnet eine aktive Partition.
1	8 Bit	Startkopf der Partition
2	8 Bit	Startsektor der Partition (Bits 0-5)
3	8 Bit	Startspur der Partition (Bits 8, 9 in den Bits 6, 7 vom Sektor)
4	8 Bit	Betriebssystem-Kennung (s. u.)
5	8 Bit	Endkopf der Partition
6	8 Bit	Endsektor der Partition (Bits 0-5)
7	8 Bit	Endspur der Partition (Bits 8, 9 in den Bits 6, 7 vom Sektor)
8	32 Bit	Anzahl der Sektoren vor der Partition
C	32 Bit	Anzahl der Sektoren der Partition

Betriebssystem-Kennungen (Auswahl):
(hexadezimal)

00 Leerer Partitionstabellen-Eintrag
01 DOS 12-Bit FAT
04 DOS 16-Bit FAT (max. 32 MB)
05 DOS 3.3+ erweiterte Partition
06 DOS 3.31+ Large File System (16-Bit FAT, > 32 MB)
07 OS/2 HPFS, Windows NT NTFS, Advanced Unix
08 OS/2 v1.0-1.3, AIX bootable partition, SplitDrive
09 AIX Datenpartition
0A OS/2 Boot Manager
0B Windows 95 with 32-Bit FAT
0C Windows 95 with 32-Bit FAT (LBA-Modus INT 13 Erweiterungen verwendend)
0E Logical-Block-addressierbares VFAT (wie 06, aber LBA-Modus INT 13 verwendend)
0F Logical-Block-addressierbares VFAT (wie 05, aber LBA-Modus INT 13 verwendend)
17 Versteckte NTFS-Partition
1B Versteckte Windows 95 FAT 32-Partition
1C Versteckte Windows 95 FAT 32-Partition (LBA-Modus INT 13 Erweiterungen verwendend)
1E Versteckte LBA VFAT-Partition
50 OnTrack Disk Manager, schreibgeschützte Partition
51 OnTrack Disk Manager
81 Linux
82 Linux Swap-Partition, Solaris (Unix)
83 Linux natives Dateisystem (ext2fs/xiafs)
85 Linux EXT
86 FAT 16 Volume/Stripe-Set (Windows NT)
87 HPFS fehlertolerante, gespiegelte Partition, NTFS Volume/Stripe-Set
BE Solaris Boot-Partition
C0 DR-DOS/Novell DOS gesicherte Partition
C6 FAT 16 Volume/Stripe-Set (Windows NT), "corrupted"
C7 NTFS Volume/Stripe-Set, "corrupted"
F2 DOS 3.3+ Sekundärpartition

Datei/Block/Sektoren füllen

Füllen mit Hex-Werten: Geben Sie 1-5 jeweils zweistellige Hex-Werte an, die aneinandergehängt in den aktuellen Block bzw. in die Datei kopiert werden.

Erzeugung zufälliger Bytes: Geben Sie ein Intervall innerhalb von 0-255 (dezimal) an, aus dem zufällig jedem einzelnen Byte des aktuellen Blocks bzw. des gesamten Dateiinhalts ein Wert zugeordnet wird. Jeder Wert aus dem Intervall wird mit gleicher Wahrscheinlichkeit ausgewählt.

Chaotische Zahlenfolge generieren: Aus einem (zufälligen) Startwert wird nach einem Mixmaster-Algorithmus eine chaotische Zahlenfolge generiert, die in den aktuellen Block bzw. in die ganze Datei kopiert wird. Der Startwert, den Sie auch selbst bestimmen können, muß mind. eine Nachkommastelle haben und kleiner als 256 sein.

Auf Wunsch kann diese Funktion **in allen geöffneten Dateien** ausgeführt werden. Dazu muß in allen Dateien entweder ein Block definiert oder in allen Dateien *kein* Block definiert sein.

Disk Editor Questions & Answers

Wie kann ich auf CD-ROM- und DVD-Sektoren unter Windows 9x zugreifen?

1. Es muß ein Windows-Treiber für das CD-ROM-Laufwerk installiert sein. Ein MS-DOS-Treiber genügt nicht.
2. Es muß eine ASPI-Schnittstelle installiert sein. Ggf. müssen Sie die Datei wnaspi32.dll von Hand in Ihr Windows\System-Verzeichnis kopieren. Die Datei befindet sich auf Ihrer Windows-Installations-CD. Zum Extrahieren aus einem CAB-Archiv empfiehlt sich das Shareware-Programm WinZip (erhältlich von http://www.winzip.com).
3. Das CD-ROM-Laufwerk muß die von WinHex benutzte Zugriffsart unterstützen. Dies ist bei den meisten heutigen ATAPI- und SCSI-Laufwerken der Fall.

Was muß ich tun, damit WinHex eine installierte PC Card ATA Flash Disk bzw. ein PCMCIA-Laufwerk als physischen Datenträger unter Windows 9x anzeigt?

Windows-Systemsteuerung -> System -> Gerätemanager -> Wählen Sie das PCMCIA-Laufwerk -> Klicken Sie auf „Eigenschaften" -> Suchen Sie nach einer Option names "Interrupt 13 Gerät" o. ä. Je nach Windows-Version kann das Auswahlfeld auch in einem anderen Bereich zu finden sein. Wenn möglich, schalten Sie diese Option *ein* und starten Sie Ihren Computer neu.

Editieren mit Schablonen

Eine Schablone ("Template") ist ein Dialogfenster, das die Mittel zum Editieren maßgeschneiderter Datenstrukturen zur Verfügung stellt. Im Vergleich zum reinen Hex-Editieren ist das Editieren mit Schablonen komfortabler und weniger fehleranfällig. Hier werden Änderungen in getrennten Editierfeldern vorgenommen und mit der ENTER-Taste bestätigt (oder beim Schließen der Schablone). Die zu editierenden Daten können von einer Datei, von Datenträger-Sektoren oder aus dem virtuellen Arbeitsspeicher stammen. Insbesondere beim Editieren von Datenbanken empfiehlt sich das Benutzen von Schablonen aufgrund des leichteren Datenzugriffs.

Eine Schablonen-Definition wird als Textdatei gespeichert. Der Schablonen-Editor ermöglicht es Ihnen, solche Definitionen zu verfassen und deren Syntax zu prüfen. Eine Schablonen-Definition enthält hauptsächlich Variablen-Deklarationen, ähnlich wie die in Programmiersprachen. Die Syntax finden Sie hier erläutert. Zu den unterstützten Datentypen gehören alle geläufigen Integer-, Gleitkomma- und Boolean-Varianten, fünf Datumstypen, Hex-Werte, Binärwerte, Zeichen und Strings. Man kann Arrays (Felder) sowohl von einzelnen Variablen als auch von ganzen Blöcken definieren.

Die Möglichkeit, beim Interpretieren von Daten mit einer Schablone die aktuelle Position frei zu bestimmen machen das Editieren mit Schablonen besonders flexibel:
• Dieselbe Variable kann in Form von unterschiedlichen Typen interpretiert und manipuliert werden.
• Irrelevante Datenbereiche können übersprungen werden.

Der Schablonen-Manager listet alle Textdateien im WinHex-Verzeichnis, die Schablonen-Definitionen enthalten, auf. Er zeigt die Bezeichnung der Schablone, eine Beschreibung, den Dateinamen und den Zeitpunkt der letzten Änderung an. Klicken Sie auf den „Anwenden"-Schalter, um unter Verwendung der ausgewählten Schablonen-Definition eine Schablone zum Editieren der Daten im aktuellen Editorfenster an der aktuellen Position anzuzeigen. Sie können im Schablonen-Manager auch neue Definitionen erstellen oder vorhandene Definitionen löschen oder mit dem Schablonen-Editor bearbeiten.

WinHex ist werkseitig mit mehreren Beispiel-Schablonen ausgestattet.

Schablonen-Definition

Eine Schablonen-Definition besteht aus einem Kopf und einem Rumpf.

Syntax des Kopfes
Variablen-Deklarationen im Rumpf
Fortgeschrittene Befehle im Rumpf

Schablonen-Definition: Kopf

Der Kopf einer Schablonen-Definition hat das folgende Format. Die Ausdrücke in Klammern sind optional. Die Reihenfolge der Ausdrücke ist nicht von Bedeutung.

template "*Titel*"
[description "*Beschreibung*"]
[appliesto (file/disk/RAM)]
[sector-aligned]
[requires *Offset* "*Hex-Werte*"]
[big-endian]
[hexadecimal]
[read-only]
[multiple [*fixe Gesamtgröße*]]
// *Hier ist Platz für allgemeine Kommentare*.
begin
    Variablen-Deklarationen
end

Ausdrücke müssen nur in Hochkommata eingeschlossen werden, wenn sie Leerzeichen enthalten. Kommentare dürfen überall in einer Schablonen-Definition auftauchen; Zeichen, die einem doppelten Schrägstrich folgen, werden vom Parser ignoriert.

Dem Schlüsselwort "appliesto" muß genau eins der Wörter file, disk oder RAM folgen. WinHex gibt eine Warnmeldung aus, wenn Sie eine auf diese Weise gekennzeichnete Schablone auf Daten von einer anderen Quelle anwenden.

Wendet man eine Schablone auf einen Datenträger an, so stellt das Schlüsselwort "sector-aligned" sicher, daß sie ungeachtet der exakten Cursor-Position auf den Anfang des aktuellen Sektors bezogen wird.

Ähnlich wie ein "appliesto"-Ausdruck ermöglicht es die "requires"-Anweisung WinHex, eine unabsichtliche Anwendung einer Schablonen-Definition auf nicht auf sie passende Daten zu verhindern. Geben Sie hinter "requires" einen Offset und eine Hex-Wert-Kette beliebiger Länge an. Dies soll die Daten, für die die Schablone konzipiert wurde, identifizieren. Zum Beispiel läßt sich ein gültig Master-Boot-Record an den Hex-Werten 55 AA an Offset 0x1FE erkennen, eine ausführbare Datei an den Hex-Werten 4D 5A ("MZ") an Offset 0x0. Es dürfen mehrere "appliesto"-Anweisungen im Definitionskopf vorkommen, die alle berücksichtigt werden.

Das Schlüsselwort "big-endian" sorgt dafür, daß alle aus mehreren Bytes bestehende Integer- und Boolean-Variablen in Big-Endian-Reihenfolge gelesen und geschrieben werden (höchst-wertiges Byte vorn).

Das Schlüsselwort "hexadecimal" bewirkt, daß Integer-Variablen innerhalb der Schablonen-Definition in hexadezimaler Schreibweise angezeigt werden.

Das Schlüsselwort "read-only" stellt sicher, daß die Schablone nur benutzt werden kann, um Datenstrukturen einzusehen, nicht um sie zu manipulieren. Die Editierfelder der Schablone erscheinen dann grau.

Wenn das Schlüsselwort "multiple" im Definitionskopf angegeben wird, erlaubt WinHex das Wechseln zu benachbarten Datensätzen derselben Struktur. Das erfordert, daß WinHex die Größe eines Datensatzes kennt. Sofern diese nicht fest als Parameter der "multiple"-Anweisung angegeben wurde, nimmt WinHex an, daß die Gesamtgröße sich berechnet als die aktuelle Position nach der Anwendung der Schablonen-

Definition minus Startposition. Wenn dies eine variable Größe ergibt, d. h. Array-Größen oder "move"-Parameter sich dynamisch aus den Werten von Variablen bestimmen, kann WinHex nicht zu vorgelagerten Datensätzen wechseln.

Variablen-Deklarationen

Der Rumpf eine Schablonen-Definition besteht im wesentlichen aus Variablen-Deklarationen, ähnlich wie die in Programmiersprachen. Eine Deklaration hat folgende Gestalt:

type "Bezeichnung"

wobei type einer der folgenden Datentypen sein kann:

• int8, uint8 = byte, int16, uint16, int32, uint32, int64,
• binary,
• float = single, real, double, longdouble = extended,
• char, char16, string, string16,
• boole8 = boolean, boole16, boole32
• hex,
• DOSDateTime, FileTime, OLEDateTime, SQLDateTime, UNIXDateTime = time_t

Die Bezeichnung der Variablen muß nur dann in Hochkammata gesetzt werden, wenn sie Leerzeichen enthält. Sie darf nicht nur aus Ziffern bestehen. WinHex unterscheidet nicht zwischen Groß- und Kleinschreibung. Maximal werden zur Identifikation einer Variablen 41 Zeichen verwendet.

title kann jeweils maximal ein Koeffizient der folgenden Koeffizientenpaare vorangestellt werden:

big-endian	little-endian	(s. Endian-ness)
hexadecimal	decimal	
read-only	read-write	

Diese Koeffizienten wirken sich nur auf die unmittelbar folgende Variable aus. Sie sind redundant, wenn sie bereits im Definition-Kopf angegeben werden.

Die Nummern am Ende der Typnamen bezeichnet die Größe einer Variablen dieses Typs (Strings: eines Zeichens) in Bits. Mit den Typen "char16" und "string16" unterstützt WinHex Unicode-Zeichen und -Strings. Höhere Unicode-Zeichen als die ersten 256 ANSI-äquivalenten werden allerdings nicht unterstützt. Es können außerdem maximal Strings einer Größe von 8192 Bytes editiert werden.

Die Typen "string", "string16" und "hex" erfordern einen zusätzlichen Parameter, der die Anzahl der Elemente angibt. Dieser Parameter kann eine Konstante oder eine zuvor deklarierte Variable sein. Wenn es sich um eine Konstante handelt, kann sie entweder dezimal oder hexadezimal geschrieben werden, im zweiten Fall muß ihr "0x" vorangestellt werden.

Sie können Arrays (Felder) deklarieren, indem Sie in eckigen Klammern die gewünschte Größe angeben, entweder hinter der Typangabe oder hinter der Variablenbezeichnung. Bspw. deklarieren die folgenden zwei Zeilen einen ASCII-String, dessen Länge dynamisch von der vorherigen Variable bestimmt wird:

```
uint8          "Länge"
char[Länge] "Ein String"
```

Dasselbe Ergebnis könnte mit folgenden zwei Deklarationen erzielt werden:

```
byte           "Länge"
string Länge          "Ein String"
```

Eine Tilde ("~") kann als Platzhalter eingesetzt werden, um zur Laufzeit mit der tatsächlichen Array-Elementnummer ersetzt werden (s. u. Fortgeschrittene Befehle). Dies trifft nicht auf Arrays des Typs char

zu, da diese von WinHex automatisch in einen String übersetzt werden.

Bitte beachten Sie, daß Schablonen in der aktuellen Version nicht fähig sind zu „rechnen", so daß Operatoren wie "+" und "*" nicht in Parametern oder Ausdrücken für die Bestimmung von Array-Größen benutzt werden können.

Fortgeschrittene Befehle

Variablendeklarationen können in geschweiften Klammern eingeschlossen werden, so daß sie einen Block bilden, der als ganzes wiederholt eingesetzt werden kann. Beachten Sie aber, daß Blöcke in der aktuellen Implementation nicht verschachtelt werden dürfen. Eine Tilde ("~") kann als Platzhalter für eine spätere Ersetzung mit dem aktuellen Stand des Wiederholungszählers in Variablennamen verwendet werden. Die optionale numbering-Anweisung legt dabei fest, mit welcher Nummer die Zählung begonnen werden soll (standardmäßig mit Null).

```
numbering   1
{
byte          "Länge"
string Länge          "String Nr. ~"
} [10]
```

In diesem Beispiel werden die tatsächlichen Variablennamen in der Schablone "String Nr. 1", "String Nr. 2", ..., "String No. 10" lauten.

Um die Übersichtlichkeit einer Schablone zu verbessern, lassen sich Gruppen von Variablen auch visuell bilden, so daß die zugehörigen Editierfelder durch freien Raum im Dialogfenster voneinander getrennt erscheinen.:

```
section "...Bezeichnung des Bereichs..."
...
endsection
```

Die Anweisungen "section", "endsection" und "numbering" haben keinen Einfluß auf die aktuelle Position der Datenauswertung durch die Schablone.

Es gibt noch zwei weitere Befehle, die auch keine Variablen deklarieren, aber explizit benutzt werden, um die aktuelle Position zu manipulieren. Dies kann z. B. geschehen, um irrelevante Daten zu überspringen (Vorwärtsbewegung) oder um bestimmte Variablen mehrfach in Form von unterschiedlichen Datentypen erfassen zu können (Rückwärtsbewegung). Benutzen Sie die "move n"-Anweisung, um n Bytes von der aktuellen Position aus zu überspringen, wobei n auch negativ sein darf. "goto n" setzt die aktuelle Position auf n, einen absoluten (positiven) Offset auf die Basisposition, auf die die Schablone angewandt wird.

Das folgende Beispiel demonstriert den Zugriff auf 4 Bytes an Daten als 32-Bit-Integer und als eine Kette von 4 Hex-Werten:

```
int32       "Seriennummer des Datenträgers (dezimal)"
move -4
hex 4       "Seriennummer des Datenträgers (hexadezimal)"
```

Datenträger klonen

Läßt Sie eine bestimmte Anzahl von Sektoren von einem Quell- auf einen Zieldatenträger kopieren. Dazu müssen beide Datenträger dieselbe Sektorgröße aufweisen. Mit dieser Funktion können Sie exakte Duplikate ganzer Festplatten herstellen, indem Sie einfach *alle* Sektoren kopieren. Aktivieren Sie die entsprechende Option, damit die richtigen Zahlen automatisch für Sie eingetragen werden. Der Zieldatenträger darf nicht kleiner als der Quelldatenträger sein.

Die Funktion »Datenträger klonen« bietet verschiedene Möglichkeiten zu verfahren, wenn defekte Sektoren auf dem Quelldatenträger angetroffen werden:
• Standardmäßig werden Sie benachrichtigt und gefragt, ob der Vorgang abgebrochen oder dennoch fortgesetzt werden soll. Bei eingeschalteter Option „Protokolldatei anlegen" werden Informationen über die gesamte Operation in eine Logdatei geschrieben. Darin sind auch die Nummern etwaiger unlesbarer Sektoren enthalten. Diese Option verhindert, daß WinHex jeden defekten Sektor während des Vorgangs einzeln meldet und kann sich z. B. für forensische Anwendungen als nützlich erweisen.
• WinHex kann die Zielsektoren, die mit dem Inhalt unlesbarer Quellsektoren beschrieben werden müßten, entweder unverändert lassen oder mit Nullbytes auffüllen.

Das konventionelle Klonen ist bei austauschbaren Datenträgern (wie Disketten) nicht möglich, wenn nur *ein* entsprechendes Laufwerk installiert ist. Eine geeignete Vorgehensweise für diesen Fall ist *Disk Imaging*, also eine Art »verzögertes« Klonen. Ein Disk-Image kann auf einen anderen Datenträger zurückgespielt werden. Das Ergebnis ist dann dasselbe wie beim Klonen.

Es gibt zwei Möglichkeiten, ein Abbild eines Datenträger zu schaffen:
• Wenn die Einfachheit der Benutzerung Priorität hat, benutzen Sie die Funktionalität, die WinHex mit Sicherungen bietet. Eine Sicherungsdatei speichert auch die Informationen, welche Sektornummern von welchem Laufwerk sie enthält.
• Der Dialog »Datenträger klonen« erlaubt es, Sektoren von einem Datenträger in eine rohe, originalgetreue Imagedatei zu kopieren (und später zurück). Zusammen mit dem stillen »Protokolldatei-Modus« ist die dem Erstellen einer Sicherung vorzuziehen, wenn es defekte Sektoren auf dem Quelldatenträger gibt.

Hinweis zu Disk-Cloning & -Imaging

Das Klonen oder Sichern des Laufwerks, auf dem die aktive Windows-Installation enthalten ist, kann eine inkonsistente Kopie zur Folge haben. In jedem Fall stellen Sie bitte sicher, daß das Ursprungslaufwerk während des Klonvorgangs nicht von anderen Programmen oder von Windows beschrieben wird. Es wird empfohlen, das von der Umgebungsvariable TEMP angegebene Verzeichnis ggf. auf ein anderes Laufwerk zu verlagern. Die Auslagerungsdatei von Windows sollte ebenfalls auf einem anderen Laufwerk liegen. Alternativ können Sie Swapping in der Systemsteuerung auch völlig ausschalten.